SUSE Linux Reference

www.novell.com

10.0 | 09/12/2005

SUSE

A NOVELL BUSINESS

Reference

List of Authors: Jörg Arndt, Stefan Behlert, Frank Bodammer, James Branam, Volker Buzek, Klara Cihlarova, Stefan Dirsch, Olaf Donjak, Roman Drahtmüller, Thorsten Dubiel, Torsten Duwe, Thomas Fehr, Stefan Fent, Werner Fink, Kurt Garloff, Joachim Gleißner, Carsten Groß, Andreas Grünbacher, Berthold Gunreben, Franz Hassels, Andreas Jaeger, Jana Jaeger, Klaus Kämpf, Andi Kleen, Hubert Mantel, Lars Marowsky-Bree, Chris Mason, Johannes Meixner, Lars Müller, Matthias Nagorni, Anas Nashif, Siegfried Olschner, Edith Parzefall, Peter Pöml, Thomas Renninger, Hannes Reinecke, Thomas Rölz, Heiko Rommel, Marcus Schäfer, Thomas Schraitle, Klaus Singvogel, Hendrik Vogelsang, Klaus G. Wagner, Rebecca Walter, Christian Zoz

All information found in this book has been compiled with utmost attention to detail. However, this does not guarantee complete accuracy. Neither SUSE LINUX GmbH, the authors, nor the translators shall be held liable for possible errors or the consequences thereof.

Many of the software and hardware descriptions cited in this book are registered trademarks. All trade names are subject to copyright restrictions and may be registered trade marks. SUSE LINUX GmbH essentially adheres to the manufacturer's spelling. Names of products and trademarks appearing in this book (with or without specific notation) are likewise subject to trademark and trade protection laws and may thus fall under copyright restrictions.

Please direct suggestions and comments to `documentation@suse.de`.

Contents

About This Guide

This manual gives you a general understanding of SUSE Linux. It is intended mainly for system administrators and home users with basic system administration knowledge. This manual presents a selection of applications needed in everyday life and provides in-depth desciptions of advanced installation and configuration scenarios.

Advanced Deployment Scenarios
Learn how to deploy SUSE Linux in complex environments.

Internet, Multimedia, Office, and Graphics
Get a tour of the most important applications a home user might need.

Mobility
Get an introduction to mobile computing with SUSE Linux and learn how to configure the various options for wireless computing, power management, and profile management.

Administration
Learn how to make your SUSE Linux secure, how to deal with file system access controls, and get to know some important utilities for Linux administrators.

System
Get an introduction to the components of your Linux system and a deeper understanding of their interaction.

Services
Learn how to configure the various network and file services that come with SUSE Linux.

1 Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Please use the User Comments feature at the bottom of each page of the online documentation, or go to `http://www.novell.com/documentation/feedback.html` and enter your comments there.

2 Additional Documentation

There are other manuals available on this SUSE Linux product, either online at `http://www.novell.com/documentation/` or in your installed system under `/usr/share/doc/manual/`:

Start-Up
: This guide covers your first steps with SUSE Linux. An online version of this document can be found at `http://www.novell.com/documentation/suse10/`.

Novell AppArmor Powered by Immunix 1.2 Installation and QuickStart Guide
: This guide outlines the initial installation procedure for the *AppArmor* product. An online version of this document can be found at `http://www.novell.com/documentation/apparmor/`.

Novell AppArmor Powered by Immunix 1.2 Administration Guide
: This guide contains in-depth information on the use of *AppArmor* in your environment. An online version of this document can be found at `http://www.novell.com/documentation/apparmor/`.

3 Documentation Conventions

The following typographical conventions are used in this manual:

- `/etc/passwd`: filenames and directory names
- *`placeholder`*: replace *`placeholder`* with the actual value
- `PATH`: the environment variable PATH
- `ls`, `--help`: commands, options, and parameters
- `user`: users or groups
- Alt, Alt + F1: a key to press or a key combination
- *File*, *File* → *Save As*: menu items, buttons

- *Dancing Penguins* (Chapter Penguins, ↑*Reference*): This is a reference to a chapter in another book.

4 Acknowledgment

With a lot of voluntary commitment, the developers of Linux cooperate on a global scale to promote the development of Linux. We thank them for their efforts—this distribution would not exist without them. Furthermore, we thank Frank Zappa and Pawar. Special thanks, of course, go to Linus Torvalds.

Have a lot of fun!

Your SUSE Team

Part I Advanced Deployment Scenarios

1 Remote Installation

SUSE Linux can be installed in several different ways. As well as the usual CD or DVD installation covered in Chapter *Installation with YaST* (↑Start-Up), you can choose from various network-based approaches or even take a completely hands-off approach to the installation of SUSE Linux.

Each method is introduced by means of two short check lists: one listing the prerequisites for this method and the other illustrating the basic procedure. More detail is then provided for all the techniques used in these installation scenarios.

NOTE

In the following sections, the system to hold your new SUSE Linux installation is referred to as *target system* or *installation target*. The term *installation source* is used for all sources of installation data. This includes physical media, such as CD and DVD, and network servers distributing the installation data in your network.

1.1 Installation Scenarios for Remote Installation

This section introduces the most common installation scenarios for remote installations. For each scenario, carefully check the list of prerequisites and follow the procedure outlined for this scenario. If in need of detailed instructions for a particular step, follow the links provided for each one of them.

IMPORTANT

The configuration of the X Window System is not part of any remote installation process. After the installation has finished, log in to the target system as root, enter `init 3`, and start SaX2 to configure the graphics hardware as described in Section 35.1, “X11 Setup with SaX2” (page 509).

1.1.1 Simple Remote Installation via VNC—Static Network Configuration

This type of installation still requires some degree of physical access to the target system to boot for installation. The installation itself is entirely controlled by a remote workstation using VNC to connect to the installation program. User interaction is required as with the manual installation in Chapter *Installation with YaST* (↑Start-Up).

For this type of installation, make sure that the following requirements are met:

- Remote installation source: NFS, HTTP, FTP, or SMB with working network connection

- Target system with working network connection

- Controlling system with working network connection and VNC viewer software or Java-enabled browser (Firefox, Konqueror, Internet Explorer, or Opera)

- Physical boot medium (CD, or DVD) for booting the target system

- Valid static IP addresses already assigned to the installation source and the controlling system

- Valid static IP address to assign to the target system

To perform this kind of installation, proceed as follows:

1 Set up the installation source as described in Section 1.2, “Setting Up the Server Holding the Installation Sources” (page 30).

2 Boot the target system using the first CD or DVD of the SUSE Linux media kit.

3. When the boot screen of the target system appears, use the boot options prompt to set the appropriate VNC options and the address of the installation source. This is described in detail in Section 1.4, "Booting the Target System for Installation" (page 49).

   The target system boots to a text-based environment, giving the network address and display number under which the graphical installation environment can be addressed by any VNC viewer application or browser. VNC installations announce themselves over OpenSLP and can be found using Konqueror in `service://` or `slp://` mode.

4. On the controlling workstation, open a VNC viewing application or Web browser and connect to the target system as described in Section 1.5.1, "VNC Installation" (page 54).

5. Perform the installation as described in Chapter *Installation with YaST* (↑Start-Up).

   You will need to reconnect to the target system after it reboots for the final part of the installation.

6. Finish the installation.

1.1.2 Simple Remote Installation via VNC—Dynamic Network Configuration via DHCP

This type of installation still requires some degree of physical access to the target system to boot for installation. The network configuration is made with DHCP. The installation itself is entirely controlled from a remote workstation using VNC to connect to the installer, but still requires user interaction for the actual configuration efforts.

For this type of installation, make sure that the following requirements are met:

- Remote installation source: NFS, HTTP, FTP, or SMB with working network connection

- Target system with working network connection

- Controlling system with working network connection and VNC viewer software or Java-enabled browser (Firefox, Konqueror, Internet Explorer, or Opera)

- Physical boot medium (CD, DVD, custom boot disk) for booting the target system

- Running DHCP server providing IP addresses

To perform this kind of installation, proceed as follows:

1 Set up the installation source as described in Section 1.2, “Setting Up the Server Holding the Installation Sources” (page 30). Choose an NFS, HTTP, or FTP network server. For a SMB installation source, refer to Section 1.2.5, “Managing a SMB Installation Source” (page 38).

2 Boot the target system using the first CD or DVD of the SUSE Linux media kit.

3 When the boot screen of the target system appears, use the boot options prompt to set the appropriate VNC options and the address of the installation source. This is described in detail in Section 1.4, “Booting the Target System for Installation” (page 49).

   The target system boots to a text-based environment, giving the network address and display number under which the graphical installation environment can be addressed by any VNC viewer application or browser. VNC installations announce themselves over OpenSLP and can be found using Konqueror in `service://` or `slp://` mode.

4 On the controlling workstation, open a VNC viewing application or Web browser and connect to the target system as described in Section 1.5.1, “VNC Installation” (page 54).

5 Perform the installation as described in Chapter *Installation with YaST* (↑Start-Up).

   You will need to reconnect to the target system after it reboots for the final part of the installation.

6 Finish the installation.

1.1.3 Remote Installation via VNC—PXE Boot and Wake on LAN

This type of installation is completely hands-off. The target machine is started and booted remotely. User interaction is only needed for the actual installation. This approach is suitable for cross-site deployments.

To perform this type of installation, make sure that the following requirements are met:

- Remote installation source: NFS, HTTP, FTP, or SMB with working network connection
- TFTP server
- Running DHCP server for your network
- Target system capable of PXE boot, networking, and Wake on LAN, plugged in and connected to the network
- Controlling system with working network connection and VNC viewer software or Java-enabled browser (Firefox, Konqueror, Internet Explorer, or Opera)

To perform this type of installation, proceed as follows:

1. Set up the installation source as described in Section 1.2, “Setting Up the Server Holding the Installation Sources” (page 30). Choose an NFS, HTTP, FTP network server or configure a SMB installation source as described in Section 1.2.5, “Managing a SMB Installation Source” (page 38).

2. Set up a TFTP server to hold a boot image that can be pulled by the target system. This is described in Section 1.3.2, “Setting Up a TFTP Server” (page 41).

3. Set up a DHCP server to provide IP addresses to all machines and reveal the location of the TFTP server to the target system. This is described in Section 1.3.1, “Setting Up a DHCP Server” (page 40).

4. Prepare the target system for PXE boot. This is described in further detail in Section 1.3.5, “Preparing the Target System for PXE Boot” (page 48).

5. Initiate the boot process of the target system using Wake on LAN. This is described in Section 1.3.7, “Wake on LAN” (page 48).

6. On the controlling workstation, open a VNC viewing application or Web browser and connect to the target system as described in Section 1.5.1, “VNC Installation” (page 54).

7. Perform the installation as described in Chapter *Installation with YaST* (↑Start-Up).

   You will need to reconnect to the target system after it reboots for the final part of the installation.

8. Finish the installation.

1.1.4 Simple Remote Installation via SSH—Static Network Configuration

This type of installation still requires some degree of physical access to the target system to boot for installation and to determine the IP address of the installation target. The installation itself is entirely controlled from a remote workstation using SSH to connect to the installer. User interaction is required as with the regular installation described in Chapter *Installation with YaST* (↑Start-Up).

For this type of installation, make sure that the following requirements are met:

- Remote installation source: NFS, HTTP, FTP, or SMB with working network connection

- Target system with working network connection

- Controlling system with working network connection and VNC viewer software or Java-enabled browser (Firefox, Konqueror, Internet Explorer, or Opera)

- Physical boot medium (CD, DVD, custom boot disk) for the target system

- Valid static IP addresses already assigned to the installation source and the controlling system

- Valid static IP address to assign to the target system

To perform this kind of installation, proceed as follows:

1. Set up the installation source as described in Section 1.2, “Setting Up the Server Holding the Installation Sources” (page 30).

2. Boot the target system using the first CD or DVD of the SUSE Linux media kit.

3. When the boot screen of the target system appears, use the boot options prompt to set the appropriate parameters for network connection, address of the installation source, and SSH enablement. This is described in detail in Section 1.4.3, “Using Custom Boot Options” (page 51).

   The target system boots to a text-based environment, giving the network address under which the graphical installation environment can be addressed by any SSH client.

4. On the controlling workstation, open a terminal window and connect to the target system as described in Section “Connecting to the Installation Program” (page 56).

5. Perform the installation as described in Chapter *Installation with YaST* (↑Start-Up).

   You will need to reconnect to the target system after it reboots for the final part of the installation.

6. Finish the installation.

1.1.5 Simple Remote Installation via SSH—Dynamic Network Configuration via DHCP

This type of installation still requires some degree of physical access to the target system to boot for installation and determine the IP address of the installation target. The installation itself is entirely controlled from a remote workstation using VNC to connect to the installer, but still requires user interaction for the actual configuration efforts.

For this type of installation, make sure that the following requirements are met:

- Remote installation source: NFS, HTTP, FTP, or SMB with working network connection
- Target system with working network connection
- Controlling system with working network connection and VNC viewer software or Java-enabled browser (Firefox, Konqueror, Internet Explorer, or Opera)
- Physical boot medium (CD or DVD) for booting the target system
- Running DHCP server providing IP addresses

To perform this kind of installation, proceed as follows:

1. Set up the installation source as described in Section 1.2, “Setting Up the Server Holding the Installation Sources” (page 30). Choose an NFS, HTTP, or FTP network server. For a SMB installation source, refer to Section 1.2.5, “Managing a SMB Installation Source” (page 38).

2. Boot the target system using the first CD or DVD of the SUSE Linux media kit.

3. When the boot screen of the target system appears, use the boot options prompt to pass the appropriate parameters for network connection, location of the installation source, and SSH enablement. See Section 1.4.3, “Using Custom Boot Options” (page 51) for detailed instructions on the use of these parameters.

   The target system boots to a text-based environment, giving you the network address under which the graphical installation environment can be addressed by any SSH client.

4. On the controlling workstation, open a terminal window and connect to the target system as described in Section “Connecting to the Installation Program” (page 56).

5. Perform the installation as described in Chapter *Installation with YaST* (↑Start-Up).

   You will need to reconnect to the target system after it reboots for the final part of the installation.

6. Finish the installation.

1.1.6 Remote Installation via SSH—PXE Boot and Wake on LAN

This type of installation is completely hands-off. The target machine is started and booted remotely.

To perform this type of installation, make sure that the following requirements are met:

- Remote installation source: NFS, HTTP, FTP, or SMB with working network connection
- TFTP server
- Running DHCP server for your network, providing a static IP to the host to install
- Target system capable of PXE boot, networking, and Wake on LAN, plugged in and connected to the network
- Controlling system with working network connection and SSH client software

To perform this type of installation, proceed as follows:

1. Set up the installation source as described in Section 1.2, "Setting Up the Server Holding the Installation Sources" (page 30). Choose an NFS, HTTP, or FTP network server. For the configuration of a SMB installation source, refer to Section 1.2.5, "Managing a SMB Installation Source" (page 38).

2. Set up a TFTP server to hold a boot image that can be pulled by the target system. This is described in Section 1.3.2, "Setting Up a TFTP Server" (page 41).

3. Set up a DHCP server to provide IP addresses to all machines and reveal the location of the TFTP server to the target system. This is described in Section 1.3.1, "Setting Up a DHCP Server" (page 40).

4. Prepare the target system for PXE boot. This is described in further detail in Section 1.3.5, "Preparing the Target System for PXE Boot" (page 48).

5 Initiate the boot process of the target system using Wake on LAN. This is described in Section 1.3.7, “Wake on LAN” (page 48).

6 On the controlling workstation, start a VNC client and connect to the target system as described in Section 1.5.2, “SSH Installation” (page 55).

7 Perform the installation as described in Chapter *Installation with YaST* (↑Start-Up).

You will need to reconnect to the target system it reboots for the final part of the installation.

8 Finish the installation.

1.2 Setting Up the Server Holding the Installation Sources

Depending on the operating system running on the machine to use as network installation source for SUSE Linux, there are several options for the server configuration. The easiest way to set up an installation server is to use YaST on SUSE LINUX Enterprise Server 9 or SUSE Linux 9.3 and higher. On other versions of SUSE LINUX Enterprise Server or SUSE Linux, set up the installation source manually.

TIP

You can even use a Microsoft Windows machine as installation server for your Linux deployment. See Section 1.2.5, “Managing a SMB Installation Source” (page 38) for details.

1.2.1 Setting Up an Installation Server Using YaST

YaST offers a graphical tool for creating network installation sources. It supports HTTP, FTP, and NFS network installation servers.

1 Log in as root to the machine that should act as installation server.

2 Start *YaST* → *Miscellaneous* → *Installation Server*.

3 Select the server type (HTTP, FTP, or NFS).

The selected server service is started automatically every time the system starts. If a service of the selected type is already running on your system and you want to configure it manually for the server, deactivate the automatic configuration of the server service with *Do not configure any network services*. In both cases, define the directory in which the installation data should be made available on the server.

4 Configure the required server type.

This step relates to the automatic configuration of server services. It is skipped when automatic configuration is deactivated. Define an alias for the root directory of the FTP or HTTP server on which the installation data should be found. The installation source will later be located under `ftp://Server-IP/Alias/Name` (FTP) or under `http://Server-IP/Alias/Name` (HTTP). `Name` stands for the name of the installation source, which is defined in the following step. If you selected NFS in the previous step, define wild cards and exports options. The NFS server will be accessible under `nfs://Server-IP/Name`. Details of NFS and exports can be found in Chapter 42, *Sharing File Systems with NFS* (page 623).

5 Configure the installation source.

Before the installation media are copied to their destination, define the name of the installation source (ideally, an easily remembered abbreviation of the product and version). YaST allows providing ISO images of the media instead of copies of the installation CDs. If you want this, activate the relevant check box and specify the directory path under which the ISO files can be found locally. Depending on the product to distribute using this installation server, it might be that more add-on CDs or service pack CDs are required to install the product completely. If you activate *Prompt for Additional CDs*, YaST automatically reminds you to supply these media. To announce your installation server in the network via OpenSLP, activate the appropriate option.

TIP

Consider announcing your installation source via OpenSLP if your network setup supports this option. This saves you from entering the network installation path on every target machine. The target systems are just booted using the SLP boot option and will find the network installation source without any further configuration. For details on this option, refer to Section 1.4, “Booting the Target System for Installation” (page 49).

6 Upload the installation data.

The most lengthy step in configuring an installation server is copying the actual installation CDs. Insert the media in the sequence requested by YaST and wait for the copying procedure to end. When the sources have been fully copied, return to the overview of existing information sources and close the configuration by selecting *Finish*.

Your installation server is now fully configured and ready for service. It is automatically started every time the system is started. No further intervention is required. You only need to configure and start this service correctly by hand if you have deactivated the automatic configuration of the selected network service with YaST as an initial step.

To deactivate an installation source, select *Change* in the overview to reach a list of all available installation sources. Choose the entry to remove then select *Delete*. This delete procedure only relates to the deactivation of the server service. The installation data itself remains in the directory chosen. However, you can remove it manually.

If your installation server should provide the installation data for more than one product of product version, start the YaST installation server module and select *Configure* in the overview of existing installation sources to configure the new installation source.

1.2.2 Manual Setup of an NFS Installation Source

Setting up an NFS source for installation is basically done in two steps. In the first step, create the directory structure holding the installation data and copy the installation media over to this structure. Second, export the directory holding the installation data to the network.

To create a directory holding the installation data, proceed as follows:

1 Log in as root.

2 Create a directory that should later hold all installation data and change into this directory. For example:

```
mkdir install/product/productversion
```

```
cd install/product/productversion
```

Replace *`product`* with an abbreviation of the product name (in this case SUSE Linux) and *`productversion`* with a string that contains the product name and version.

3 For each CD contained in the media kit execute the following commands:

a Copy the entire content of the installation CD into the installation server directory:

```
cp -a /media/path_to_your_CD-ROM_drive .
```

Replace *`path_to_your_CD-ROM_drive`* with the actual path under which your CD or DVD drive is addressed. Depending on the type of drive used in your system, this can be `cdrom`, `cdrecorder`, `dvd`, or `dvdrecorder`.

b Rename the directory to the CD number:

```
mv path_to_your_CD-ROM_drive CDx
```

Replace *`x`* with the actual number of your CD.

To export the installation sources via NFS using YaST, proceed as follows:

1 Log in as root.

2 Start *YaST* → *Network Services* → *NFS Server*.

3 Select *Start NFS Server* and *Open Port in Firewall* and click *Next*.

4 Select *Add Directory* and enter the path to the directory holding the installation data. In this case, it is `/productversion`.

5 Select *Add Host* and enter the hostnames of the machines to which to export the installation data. Instead of specifying hostnames here, you could also use wild cards, ranges of network addresses, or just the domain name of your network. Enter the appropriate export options or leave the default, which works fine in most setups. For more information about the syntax used in exporting NFS shares, read the `exports` man page.

6 Click *Finish*.

The NFS server holding the SUSE Linux installation sources is automatically started and integrated into the boot process.

If you prefer to manually export the installation sources via NFS instead of using the YaST NFS Server module, proceed as follows:

1 Log in as root.

2 Open the file `/etc/exports` and enter the following line:

```
/productversion *(ro,root_squash,sync)
```

This exports the directory `/productversion` to any host that is part of this network or to any host that can connect to this server. To limit the access to this server, use netmasks or domain names instead of the general wild card `*`. Refer to the `export` man page for details. Save and exit this configuration file.

3 To add the NFS service to the list of servers started during system boot, execute the following commands:

```
insserv /etc/init.d/nfsserver
insserv /etc/init.d/portmap
```

4 Start the NFS server using the following command:

```
rcnfsserver start
```

If you need to change the configuration of your NFS server later, modify the configuration file and restart the NFS daemon with `rcnfsserver restart`.

Announcing the NFS server via OpenSLP makes its address known to all clients in your network.

1. Log in as root.

2. Enter the directory /etc/slp.reg.d/.

3. Create a configuration file called install.suse.nfs.reg containing the following lines:

   ```
   # Register the NFS Installation Server
   service:install.suse:nfs://$HOSTNAME/path_instsource/CD1,en,65535
   description=NFS Installation Source
   ```

   Replace *path_instsource* with the actual path to the installation source on your server.

4. Save this configuration file and start the OpenSLP daemon using the following command:

   ```
   rcslpd start
   ```

For more information about OpenSLP, refer to the package documentation located under /usr/share/doc/packages/openslp/ or refer to Chapter 39, *SLP Services in the Network* (page 589).

1.2.3 Manual Setup of an FTP Installation Source

Creating an FTP installation source is very similar to creating an NFS installation source. FTP installation sources can be announced over the network using OpenSLP as well.

1. Create a directory holding the installation sources as described in Section 1.2.2, "Manual Setup of an NFS Installation Source" (page 32).

2. Configure the FTP server to distribute the contents of your installation directory:

   a. Log in as root and install the package pure-ftpd (a lean FTP server) using the YaST package manager.

   b. Enter the FTP server root directory:

```
cd/srv/ftp
```

c Create a subdirectory holding the installation sources in the FTP root directory:

```
mkdir instsource
```

Replace `instsource` with the product name.

d Copy the contents of all installation CDs into the FTP server's root directory (similar to the procedure described in Section 1.2.2, “Manual Setup of an NFS Installation Source” (page 32), Step 3 (page 33)).

Alternatively, mount the contents of the already existing installation repository into the change root environment of the FTP server:

```
mount --bind path_to_instsource /srv/ftp/instsource
```

Replace `path_to_instsource` and `instsource` with values matching your setup. If you need to make this permanent, add it to `/etc/fstab`.

e Start pure-ftpd:

```
pure-ftpd &
```

3 Announce the installation source via OpenSLP, if this is supported by your network setup:

a Create a configuration file called `install.suse.ftp.reg` under `/etc/slp/reg.d/` that contains the following lines:

```
# Register the FTP Installation Server
service:install.suse:ftp://$HOSTNAME/srv/ftp/instsource/CD1,en,65535
description=FTP Installation Source
```

Replace `instsource` with the actual name to the installation source directory on your server. The `service:` line should be entered as one continuous line.

b Save this configuration file and start the OpenSLP daemon using the following command:

```
rcslpd start
```

1.2.4 Manual Setup of an HTTP Installation Source

Creating an HTTP installation source is very similar to creating an NFS installation source. HTTP installation sources can be announced over the network using OpenSLP as well.

1 Create a directory holding the installation sources as described in Section 1.2.2, “Manual Setup of an NFS Installation Source” (page 32).

2 Configure the HTTP server to distribute the contents of your installation directory:

   a Log in as root and install the package `apache2` using the YaST package manager.

   b Enter the root directory of the HTTP server (`/srv/www/htdocs`) and create a subdirectory that will hold the installation sources:

   ```
   mkdir instsource
   ```

   Replace *`instsource`* with the product name.

   c Create a symbolic link from the location of the installation sources to the root directory of the Web server (`/srv/www/htdocs`):

   ```
   ln -s /path_instsource /srv/www/htdocs/instsource
   ```

   d Modify the configuration file of the HTTP server (`/etc/apache2/default-server.conf`) to make it follow symbolic links. Replace the following line:

   ```
   Options None
   ```

   with

   ```
   Options Indexes FollowSymLinks
   ```

   e Restart the HTTP server using `rcapache2 restart`.

3. Announce the installation source via OpenSLP, if this is supported by your network setup:

   a. Create a configuration file called `install.suse.http.reg` under `/etc/slp/reg.d/` that contains the following lines:

      ```
      # Register the HTTP Installation Server
      service:install.suse:http://$HOSTNAME/srv/www/htdocs/instsource/CD1/,en,65535
      description=HTTP Installation Source
      ```

      Replace *`path_to_instsource`* with the actual path to the installation source on your server. The `service:` line should be entered as one continuous line.

   b. Save this configuration file and start the OpenSLP daemon using `rcslpd restart`.

1.2.5 Managing a SMB Installation Source

Using SMB (Samba), you can import the installation sources from a Microsoft Windows server and start your Linux deployment even with no Linux machine around.

To set up an exported Windows Share holding your SUSE Linux installation sources, proceed as follows:

1. Log in to your Windows machine.

2. Start Explorer and create a new folder that will hold the entire installation tree and name it `INSTALL`, for example.

3. Export this share according the procedure outlined in your Windows documentation.

4. Enter this share and create a subfolder, called *`product`*. *`product`* needs to be replaced with the actual product name (SUSE Linux in this case).

5. Copy each SUSE Linux CD into a separate folder and name these folders `CD1`, `CD2`, `CD3`, etc.

6. Enter the top directory of the exported share (`INSTALL`, in this example) and copy the following files and folders from *`product`*`/CD1` to this folder: `content`, `media.1`, `control.xml`, and `boot`.

7. Create a new folder under `INSTALL` and name it `yast`.

   Enter the `yast` folder and create the files `order` and `instorder`.

8. Open the `order` file and enter the following line:

   ```
   /NLD/CD1 smb://user:password@hostname/productCD1
   ```

   Replace *`user`* with the username you use on the Windows machine or use `Guest` to enable guest login to this share. *`password`* should be replaced either with your login password or any other string for guest login. *`hostname`* should be replaced with the network name of your Windows machine.

9. Open the `instorder` file and add the following line:

   ```
   /product/CD1
   ```

To use a SMB mounted share as installation source, proceed as follows:

1. Boot the installation target.

2. Select *Installation*.

3. Press F4 for a selection of installation sources.

4. Choose SMB and enter the Windows machine's name or IP address, the share name (`INSTALL`, in this example), username, and password.

   After you hit Enter, YaST starts and you can perform the installation.

1.3 Preparing the Boot of the Target System

This section covers the configuration tasks needed in complex boot scenarios. It contains ready-to-apply configuration examples for DHCP, PXE boot, TFTP, and Wake on LAN.

1.3.1 Setting Up a DHCP Server

The setup of a DHCP server on SUSE Linux is done by manually editing the appropriate configuration files. This section covers extending an existing DHCP server configuration to provide the data needed to serve in a TFTP, PXE, and WOL environment.

Manual Setup of a DHCP Server

All the DHCP server needs to do, apart from providing automatic address allocation to your network clients, is to announce the IP address of the TFTP server and the file that should be pulled in by the installation routines on the target machine.

1. Log in as root to the machine hosting the DHCP server.

2. Append the following lines to your DHCP server's configuration file located under `/etc/dhcpd.conf`:

   ```
   group {
     # PXE related stuff
     #
     # "next server" defines the tftp server that will be used
     next server ip_tftp_server:
     #
     # "filename" specifiies the pxelinux image on the tftp server
     # the server runs in chroot under /srv/tftpboot
     filename  "pxelinux.0";
   }
   ```

   Replace *`ip_of_the_tftp_server`* with the actual IP address of the TFTP server.

   For more information about the options available in `dhcpd.conf`, refer to the `dhcpd.conf` manual page.

3 Restart the DHCP server by executing `rcdhcpd restart`.

If you plan on using SSH for the remote control of a PXE and Wake on LAN installation, explicitly specify the IP address DHCP should provide to the installation target. To achieve this, modify the above mentioned DHCP configuration according to the following example:

```
group {
  # PXE related stuff
  #
  # "next server" defines the tftp server that will be used
  next server ip_tftp_server:
  #
  # "filename" specifiies the pxelinux image on the tftp server
  # the server runs in chroot under /srv/tftpboot
  filename "pxelinux.0";
  host test { hardware ethernet mac_address;
              fixed-address some_ip_address; }
      }
```

The host statement introduces the hostname of the installation target. To bind the hostname and IP address to a specific host, you have to know and specify the system's hardware (MAC) address. Replace all the variables used in this example with the actual values that match your environment.

After restarting the DHCP server, it provides a static IP to the host specified, enabling you to connect to the system via SSH.

1.3.2 Setting Up a TFTP Server

Set up a TFTP server with YaST or manually on any other Linux operating system that supports xinetd and tftp. The TFTP server delivers the boot image to the target system once it boots and sends a request for it.

Setting Up a TFTP Server Using YaST

1 Log in as root.

2 Start *YaST* → *Network Services* → *TFTP Server* and install the requested package.

3. Click *Enable* to make sure that the server is started and included in the boot routines. No further action from your side is required to secure this. xinetd starts tftpd at boot time.

4. Click *Open Port in Firewall* to open the appropriate port in the firewall running on your machine. If there is no firewall running on your server, this option is not available.

5. Click *Browse* to browse for the boot image directory.

   The default directory `/tftpboot` is created and selected automatically.

6. Click *Finish* to apply your settings and start the server.

Manual Setup of a TFTP Server

1. Log in as root and install the packages `tftp` and `xinetd`.

2. If unavailable, create `/srv/tftpboot` and `/srv/tftpboot/pxelinux.cfg` directories.

3. Add the appropriate files needed for the boot image as described in Section 1.3.3, “PXE Boot” (page 43).

4. Modify the configuration of xinetd located under `/etc/xinetd.d/` to make sure that the tftp server is started on boot:

   a. If it does not exist, create a file called `tftp` under this directory with `touch tftp`. Then run `chmod 755 tftp`.

   b. Open the file `tftp` and add the following lines:

```
service tftp
{
        socket_type             = dgram
        protocol                = udp
        wait                    = yes
        user                    = root
        server                  = /usr/sbin/in.tftpd
        server_args             = -s /tftpboot
        disable                 = no
}
```

c Save the file and restart xinetd with `rcxinetd restart`.

1.3.3 PXE Boot

Some technical background information as well as PXE's complete specifications are available in the Preboot Execution Environment (PXE) Specification (`ftp://download.intel.com/labs/manage/wfm/download/pxespec.pdf`).

1 Change to the directory of your installation repository and copy the `linux`, `initrd`, `message`, and `memtest` files to the `/srv/tftpboot` directory by entering the following:

```
cp -a boot/loader/linux boot/loader/initrd
      boot/loader/message boot/loader/memtest /srv/tftpboot
```

2 Install the `syslinux` package directly from your installation CDs or DVDs with YaST.

3 Copy the `/usr/share/syslinux/pxelinux.0` file to the `/srv/tftpboot` directory by entering the following:

```
cp -a /usr/share/syslinux/pxelinux.0 /srv/tftpboot
```

4 Change to the directory of your installation repository and copy the `isolinux.cfg` file to `/srv/tftpboot/pxelinux.cfg/default` by entering the following:

```
cp -a boot/loader/isolinux.cfg /srv/tftpboot/pxelinux.cfg/default
```

5 Edit the `/srv/tftpboot/pxelinux.cfg/default` file and remove the lines beginning with `gfxboot`, `readinfo`, and `framebuffer`.

6 Insert the following entries in the append lines of the default `failsafe` and `apic` labels:

`insmod=e100`

By means of this entry, the kernel module for an Intel 100MBit/s network card is loaded on the PXE clients. This entry depends on the client's hardware and must be adapted accordingly. In the case of a Broadcom GigaBit network card, this entry should read `insmod=bcm5700`.

`netdevice=eth0`

This entry defines the client's network interface that must be used for the network installation. It is only necessary if the client is equipped with several network cards and must be adapted accordingly. In case of a single network card, this entry can be omitted.

`install=nfs://`*`ip_instserver`*`/`*`path_instsource`*`/CD1`

This entry defines the NFS server and the installation source for the client installation. Replace *`ip_instserver`* with the actual IP address of your installation server. *`path_instsource`* should be replaced with the actual path to the installation sources. HTTP, FTP, or SMB sources are addressed in a similar manner, except for the protocol prefix, which should read `http`, `ftp`, or `smb`.

> **IMPORTANT**
>
> If you need to pass other boot options to the installation routines, such as SSH or VNC boot parameters, append them to the `install` entry. An overview of parameters and some examples are given in Section 1.4, “Booting the Target System for Installation” (page 49).

An example `/srv/tftpboot/pxelinux.cfg/default` file follows. Adjust the protocol prefix for the installation source to match your network setup and specify your preferred method of connecting to the installer by adding the `vnc` and `vncpassword` or the `ssh` and `sshpassword` options to the `install` entry. The lines separated by \ must be entered as one continuous line without a line break and without the \.

```
default linux

# default
label linux
  kernel linux
     append initrd=initrd ramdisk_size=65536 insmod=e100 \
     install=nfs://ip_instserver/path_instsource/product

# failsafe
label failsafe
  kernel linux
  append initrd=initrd ramdisk_size=65536 ide=nodma apm=off acpi=off \
  insmod=e100 install=nfs://ip_instserver/path_instsource/product

# apic
label apic
  kernel linux
  append initrd=initrd ramdisk_size=65536 apic insmod=e100 \
  install=nfs://ip_instserver/path_instsource/product

# manual
label manual
  kernel linux
  append initrd=initrd ramdisk_size=65536 manual=1

# rescue
label rescue
  kernel linux
  append initrd=initrd ramdisk_size=65536 rescue=1

#  memory test
label memtest
  kernel memtest

# hard disk
label harddisk
  kernel
  linux append SLX=0x202

implicit     0
display      message
prompt       1
timeout      100
```

Replace *ip_instserver* and *path_instsource* with the values used in your setup.

The following section serves as a short reference to the PXELINUX options used in this setup. More information about the options available can be found in the documentation of the syslinux package located under /usr/share/doc/packages/syslinux/.

1.3.4 PXELINUX Configuration Options

The options listed here are a subset of all the options available for the PXELINUX configuration file.

DEFAULT *kernel options...*

Sets the default kernel command line. If PXELINUX boots automatically, it acts as if the entries after DEFAULT had been typed in at the boot prompt, except the auto option is automatically added, indicating an automatic boot.

If no configuration file is present or no DEFAULT entry is present in the configuration file, the default is the kernel name "linux" with no options.

APPEND *options...*

Add one or more options to the kernel command line. These are added for both automatic and manual boots. The options are added at the very beginning of the kernel command line, usually permitting explicitly entered kernel options to override them.

LABEL *label* KERNEL *image* APPEND *options...*

Indicates that if `label` is entered as the kernel to boot, PXELINUX should instead boot `image` and the specified `APPEND` options should be used instead of the ones specified in the global section of the file (before the first `LABEL` command). The default for `image` is the same as `label` and, if no `APPEND` is given, the default is to use the global entry (if any). Up to 128 `LABEL` entries are permitted.

Note that GRUB uses the following syntax:

```
title mytitle
  kernel my_kernel my_kernel_options
  initrd myinitrd
```

while PXELINUX uses the following syntax:

```
label mylabel
  kernel mykernel
  append myoptions
```

Labels are mangled as if they were filenames and they must be unique after mangling. For example, the two labels "v2.1.30" and "v2.1.31" would not be distinguishable under PXELINUX because both mangle to the same DOS filename.

The kernel does not have to be a Linux kernel; it can be a boot sector or a COMBOOT file.

APPEND -

Append nothing. `APPEND` with a single hyphen as argument in a `LABEL` section can be used to override a global `APPEND`.

`LOCALBOOT` *`type`*

On PXELINUX, specifying `LOCALBOOT 0` instead of a `KERNEL` option means invoking this particular label and causes a local disk boot instead of a kernel boot.

Argument	Description
`0`	Perform a normal boot
`4`	Performs a local boot with the Universal Network Driver Interface (UNDI) driver still resident in memory
`5`	Performs a local boot with the entire PXE stack, including the UNDI driver, still resident in memory

All other values are undefined. If you do not know what the UNDI or PXE stacks are, specify `0`.

`TIMEOUT` *`time-out`*

Indicates how long to wait at the boot prompt until booting automatically, in units of 1/10 second. The time-out is cancelled as soon as the user types anything on the keyboard, the assumption being that the user completes the command begun. A time-out of zero disables the time-out completely (this is also the default).

The maximum possible time-out value is 35996 (just less than one hour).

`PROMPT` *`flag_val`*

If `flag_val` is 0, displays the boot prompt only if Shift or Alt is pressed or Caps Lock or Scroll lock is set (this is the default). If `flag_val` is 1, always displays the boot prompt.

```
F2  filename
F1  filename
..etc...
F9  filename
F10filename
```

Displays the indicated file on the screen when a function key is pressed at the boot prompt. This can be used to implement preboot online help (presumably for the kernel command line options.) For backward compatibility with earlier releases, F10 can be also entered as F0. Note that there is currently no way to bind filenames to F11 and F12.

1.3.5 Preparing the Target System for PXE Boot

Prepare the system's BIOS for PXE boot by including the PXE option in the BIOS boot order.

WARNING

Do not place the PXE option ahead of the hard disk boot option in the BIOS. Otherwise this system would try to reinstall itself every time you boot it.

1.3.6 Preparing the Target System for Wake on LAN

Wake on LAN (WOL) requires the appropriate BIOS option to be enabled prior to the installation. Also, note down the MAC address of the target system. This data is needed to initiate Wake on LAN.

1.3.7 Wake on LAN

Wake on LAN allows a machine to be powered on via a special network packet that is sent containing the machine's MAC address. Because every machine in the world has a unique MAC identifier, you do not need to worry about accidentally powering on the wrong machine.

IMPORTANT

If the controlling machine is not located in the same network segment as the installation target that should be awakened, either configure the WOL requests to be sent as multicasts or remotely control a machine on that network segment to act as the sender of these requests.

1.3.8 Manual Wake on LAN

1. Log in as root.

2. Start *YaST* → *Software Management* and install the package `netdiag`.

3. Open a terminal and enter the following command as root to wake the target:

   ```
   ether-wakemac_of_target
   ```

   Replace *`mac_of_target`* with the actual MAC address of the target.

1.4 Booting the Target System for Installation

Basically, there are two different ways to customize the boot process for installation apart from those mentioned under Section 1.3.7, "Wake on LAN" (page 48) and Section 1.3.3, "PXE Boot" (page 43). You can either use the default boot options and F keys or use the boot options prompt of the installation boot screen to pass any boot options that the installation kernel might need on this particular hardware.

1.4.1 Using the Default Boot Options

The boot options have already been described in detail in Chapter *Installation with YaST* (↑Start-Up).

Generally, just selecting *Installation* starts the installation boot process. If problems occur, the *Installation—ACPI Disabled* or *Installation—Safe Settings* options might come in handy.

For more information about troubleshooting the installation process, refer to Section "Installation Problems" (Chapter 9, *Common Problems and Their Solutions*, ↑Start-Up).

1.4.2 Using the F Keys

The menu bar at the bottom screen offers some advanced functionality needed in some setups. Using the F keys, you can specify additional options to pass to the installation routines without having to know the detailed syntax of these parameters you would need if you entered them as boot options (see Section 1.4.3, “Using Custom Boot Options” (page 51)).

See the table below for a complete set of the options available.

Table 1.1 *F Keys During Installation*

Key	Purpose	Available Options	Default Value
F1	Provide help	None	None
F2	Select the installation language	All supported languages	English
F3	Change screen resolution for installation	• Text mode • VESA • resolution #1 • resolution #2 • ...	• Default value depends on your graphics hardware
F4	Select the installation source	• CD-ROM/DVD • SLP • FTP • HTTP • NFS • SMB	CD-ROM/DVD

Key	Purpose	Available Options	Default Value
		- Hard Disk	
F5	Apply driver update disk	Driver	None

1.4.3 Using Custom Boot Options

Using the appropriate set of boot options helps facilitate your installation procedure. Many parameters can also be configured later using the linuxrc routines, but using the boot options is easier. In some automated setups, the boot options can be provided with `initrd` or an `info` file.

The following table lists all installation scenarios mentioned in this chapter with the required parameters for booting and the corresponding boot options. Just append all of them in the order they appear in this table to get one boot option string that is handed to the installation routines. For example (all in one line):

```
install=... netdevice=... hostip=...netmask=... vnc=... vncpassword=...
```

Replace all the values (...) in this string with the values appropriate for your setup.

***Table 1.2** Installation (Boot) Scenarios Used in This Chapter*

Installation Scenario	Parameters Needed for Booting	Boot Options
Chapter *Installation with YaST* (↑Start-Up)	None: system boots automatically	None needed
Section 1.1.1, "Simple Remote Installation via VNC—Static Network Configuration" (page 22)	• Location of the installation server • Network device • IP address • Netmask • Gateway • VNC enablement • VNC password	• `install=(nfs,http,ftp,smb)://`*`path_to_instmedia`* • `netdevice=`*`some_netdevice`* (only needed if several network devices are available)

Installation Scenario	Parameters Needed for Booting	Boot Options
		• `hostip=`*`some_ip`* • `netmask=`*`some_netmask`* • `gateway=`*`ip_gateway`* • `vnc=1` • `vncpassword=`*`some_password`*
Section 1.1.2, "Simple Remote Installation via VNC—Dynamic Network Configuration via DHCP" (page 23)	• Location of the installation server • VNC enablement • VNC password	• `install=(nfs,http,ftp,smb)://`*`path_to_instmedia`* • `vnc=1` • `vncpassword=`*`some_password`*
Section 1.1.3, "Remote Installation via VNC—PXE Boot and Wake on LAN" (page 25)	• Location of the installation server • Location of the TFTP server • VNC enablement • VNC password	Not applicable; process managed through PXE and DHCP
Section 1.1.4, "Simple Remote Installation via SSH—Static Network Configuration" (page 26)	• Location of the installation server • Network device • IP address • Netmask • Gateway • SSH enablement • SSH password	• `install=(nfs,http,ftp,smb)://`*`path_to_instmedia`* • `netdevice=`*`some_netdevice`* (only needed if several network devices are available) • `hostip=`*`some_ip`* • `netmask=`*`some_netmask`* • `gateway=`*`ip_gateway`*

Installation Scenario	Parameters Needed for Booting	Boot Options
		• `usessh=1` • `sshpassword=`*`some_password`*
Section 1.1.5, “Simple Remote Installation via SSH—Dynamic Network Configuration via DHCP” (page 27)	• Location of the installation server • SSH enablement • SSH password	• `install=(nfs,http,ftp,smb)://`*`path_to_instmedia`* • `usessh=1` • `sshpassword=`*`some_password`*
Section 1.1.6, “Remote Installation via SSH—PXE Boot and Wake on LAN” (page 29)	• Location of the installation server • Location of the TFTP server • SSH enablement • SSH password	Not applicable; process managed through PXE and DHCP

TIP

Find more information about the linuxrc boot options used for booting a Linux system in `/usr/share/doc/packages/linuxrc/linuxrc.html`.

1.5 Monitoring the Installation Process

There are several options for remotely monitoring the installation process. If the proper boot options have been specified while booting for installation, either VNC or SSH can be used to control the installation and system configuration from a remote workstation.

1.5.1 VNC Installation

Using any VNC viewer software, you can remotely control the installation of SUSE Linux from virtually any operating system. This section introduces the setup using a VNC viewer application or a Web browser.

Preparing for VNC Installation

All you need to do on the installation target to prepare for a VNC installation is to provide the appropriate boot options at the initial boot for installation (see Section 1.4.3, "Using Custom Boot Options" (page 51)). The target system boots into a text-based environment and waits for a VNC client to connect to the installation program.

The installation program announces the IP address and display number needed to connect for installation. If you have physical access to the target system, this information is provided right after the system booted for installation. Enter this data when your VNC client software prompts for it and provide your VNC password.

Because the installation target announces itself via OpenSLP, you can retrieve the address information of the installation target via an SLP browser without the need for any physical contact to the installation itself provided your network setup and all machines support OpenSLP:

1. Start the KDE file and Web browser Konqueror.

2. Enter `service://yast.installation.suse` in the location bar.

   The target system then appears as an icon in the Konqueror screen. Clicking this icon launches the KDE VNC viewer in which to perform the installation. Alternatively, run your VNC viewer software with the IP address provided and add `:1` at the end of the IP address for the display the installation is running on.

Connecting to the Installation Program

Basically, there are two ways to connect to a VNC server (the installation target in this case). You can either start an independent VNC viewer application on any operating system or connect using a Java-enabled Web browser.

Using VNC, you can control the installation of a Linux system from any other operating system, including other Linux flavors, Windows, or Mac OS.

On a Linux machine, make sure that the package `tightvnc` is installed. On a Windows machine, install the Windows port of this application, which can be obtained at the TightVNC home page (http://www.tightvnc.com/download.html).

To connect to the installation program running on the target machine, proceed as follows:

1 Start the VNC viewer.

2 Enter the IP address and display number of the installation target as provided by the SLP browser or the installation program itself:

```
ip_address:display_number
```

A window opens on your desktop displaying the YaST screens as in a normal local installation.

Using a Web browser to connect to the installation program makes you totally independent of any VNC software or the underlying operating system. As long as the browser application has Java support enabled, you can use any browser (Firefox, Internet Explorer, Konqueror, Opera, etc.) to perform the installation of your Linux system.

To perform a VNC installation, proceed as follows:

1 Launch your preferred Web browser.

2 Enter the following at the address prompt:

```
http://ip_address_of_target:5801
```

3 Enter your VNC password when prompted to do so. The browser window now displays the YaST screens as in a normal local installation.

1.5.2 SSH Installation

Using SSH, you can remotely control the installation of your Linux machine using any SSH client software.

Preparing for SSH Installation

Apart from installing the appropriate software package (OpenSSH for Linux and PuTTY for Windows), you just need to pass the appropriate boot options to enable SSH for installation. See Section 1.4.3, "Using Custom Boot Options" (page 51) for details. OpenSSH is installed by default on any SUSE Linux based operating system.

Connecting to the Installation Program

1 Retrieve the installation target's IP address.

If you have physical access to the target machine, just take the IP address the installation routine provides at the console after the initial boot. Otherwise take the IP address that has been assigned to this particular host in the DHCP server configuration.

2 At a command line enter the following command:

```
ssh -X root@ip_address_of_target
```

Replace `ip_address_of_target` with the actual IP address of the installation target.

3 When prompted for a username, enter `root`.

4 When prompted for password, enter the password that has been set via the SSH boot option.

After you have successfully authenticated, a command line prompt for the installation target appears.

5 Enter `yast` to launch the installation program.

A window opens showing the normal YaST screens as described in Chapter *Installation with YaST* (↑Start-Up).

Advanced Disk Setup 2

Sophisticated system configurations require particular disk setups. To get persistent device naming with SCSI devices, use a specific start-up script. Logical Volume Management (LVM) is a disk partitioning scheme that is designed to be much more flexible than the physical partitioning used in standard setups. Its snapshot functionality enables you to create data backups easily. Redundant Array of Independent Disks (RAID) offers increased data integrity, performance, and fault tolerance.

2.1 Permanent Device Names for SCSI Devices

When the system is booted, SCSI devices are assigned device filenames in a more or less dynamic way. This is no problem as long as the number or configuration of the devices does not change. However, if a new SCSI hard disk is added and the new hard disk is detected by the kernel before the old hard disk, the old disk is assigned a new name and the entry in the mount table `/etc/fstab` no longer matches.

To avoid this problem, the system start-up script `boot.scsidev` could be used. Enable this script using `/sbin/insserv` and set parameters for it in `/etc/sysconfig/scsidev`. The script `/etc/rc.d/boot.scsidev` handles the setup of the SCSI devices during the boot procedure and enters permanent device names under `/dev/scsi/`. These names can then be used in `/etc/fstab`. In addition, `/etc/scsi.alias` can be used to define persistent names for the SCSI configuration. The naming scheme of the devices in `/etc/scsi` is explained in `man scsidev`.

In the expert mode of the runlevel editor, activate `boot.scsidev` for level `B`. The links needed for generating the names during the boot procedure are then created in `/etc/init.d/boot.d`.

TIP: Device Names and udev

For SUSE Linux, although `boot.scsidev` is still supported, the preferred way to create persistent device names is to use udev to create device nodes with persistent names in `/dev/by-id/`.

2.2 LVM Configuration

This section briefly describes the principles behind LVM and its basic features that make it useful under many circumstances. In Section 2.2.2, “LVM Configuration with YaST” (page 60), learn how to set up LVM with YaST.

WARNING

Using LVM might be associated with increased risk, such as data loss. Risks also include application crashes, power failures, and faulty commands. Save your data before implementing LVM or reconfiguring volumes. Never work without a backup.

2.2.1 The Logical Volume Manager

The Logical Volume Manager (LVM) enables flexible distribution of hard disk space over several file systems. It was developed because sometimes the need to change the segmentation of hard disk space arises only after the initial partitioning during installation has already been done. Because it is difficult to modify partitions on a running system, LVM provides a virtual pool (volume group, VG for short) of memory space from which logical volumes (LVs) can be created as needed. The operating system accesses these LVs instead of the physical partitions. Volume groups can span more than only one disk so that several disks or parts of them may constitute one single VG. This way, LVM provides a kind of abstraction from the physical disk space that allows its segmentation to be changed in a much easier and safer way than physical repartitioning does. Background information regarding physical partitioning can be found in Section “Partition Types" (Chapter 1, *Installation with YaST*, ↑Start-Up) and Section "Partitioner" (Chapter 3, *System Configuration with YaST*, ↑Start-Up).

Figure 2.1 *Physical Partitioning versus LVM*

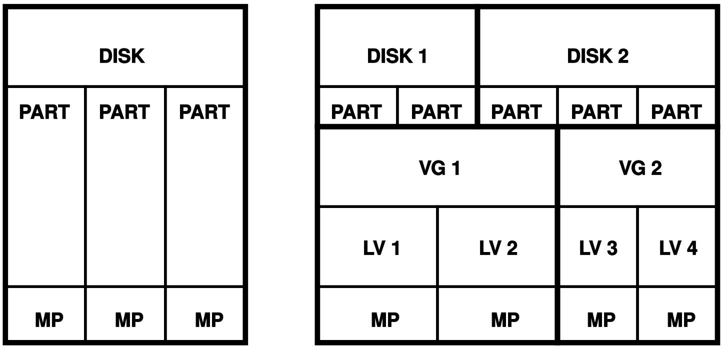


Figure 2.1, "Physical Partitioning versus LVM" (page 59) compares physical partitioning (left) with LVM segmentation (right). On the left side, one single disk has been divided into three physical partitions (PART), each with a mount point (MP) assigned so that the operating system can access them. On the right side, two disks have been divided into two and three physical partitions each. Two LVM volume groups (VG 1 and VG 2) have been defined. VG 1 contains two partitions from DISK 1 and one from DISK 2. VG 2 contains the remaining two partitions from DISK 2. In LVM, the physical disk partitions that are incorporated in a volume group are called physical volumes (PVs). Within the volume groups, four logical volumes (LV 1 through LV 4) have been defined, which can be used by the operating system via the associated mount points. The border between different logical volumes need not be aligned with any partition border. See the border between LV 1 and LV 2 in this example.

LVM features:

- Several hard disks or partitions can be combined in a large logical volume.

- Provided the configuration is suitable, an LV (such as `/usr`) can be enlarged when the free space is exhausted.

- Using LVM, even add hard disks or LVs in a running system. However, this requires hot-swappable hardware that is capable of such actions.

- It is possible to activate a "striping mode" that distributes the data stream of a logical volume over several physical volumes. If these physical volumes reside on different disks, this can improve the reading and writing performance just like RAID 0.

- The snapshot feature enables consistent backups (especially for servers) in the running system.

With these features, using LVM already makes sense for heavily used home PCs or small servers. If you have a growing data stock, as in the case of databases, music archives, or user directories, LVM is just the right thing for you. This would allow file systems that are larger than the physical hard disk. Another advantage of LVM is that up to 256 LVs can be added. However, keep in mind that working with LVM is different from working with conventional partitions. Instructions and further information about configuring LVM is available in the official LVM HOWTO at `http://tldp.org/HOWTO/LVM-HOWTO/`.

Starting from kernel version 2.6, LVM version 2 is available, which is downward-compatible with the previous LVM and enables the continued management of old volume groups. When creating new volume groups, decide whether to use the new format or the downward-compatible version. LVM 2 does not require any kernel patches. It makes use of the device mapper integrated in kernel 2.6. This kernel only supports LVM version 2. Therefore, when talking about LVM, this section always refers to LVM version 2.

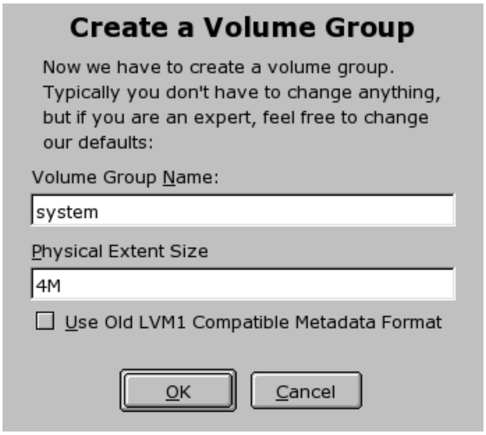
2.2.2 LVM Configuration with YaST

The YaST LVM configuration can be reached from the YaST Expert Partitioner (see Section “Partitioner” (Chapter 3, *System Configuration with YaST*, ↑Start-Up)). This professional partitioning tool enables you to edit and delete existing partitions and create new ones that should be used with LVM. There, create an LVM partition by first clicking *Create* → *Do not format* then selecting *0x8E Linux LVM* as the partition identifier. After creating all the partitions to use with LVM, click *LVM* to start the LVM configuration.

Creating Volume Groups

If no volume group exists on your system yet, you are prompted to add one (see Figure 2.2, “Creating a Volume Group” (page 61)). It is possible to create additional groups with *Add group*, but usually one single volume group is sufficient. `system` is suggested as a name for the volume group in which the SUSE Linux system files are located. The physical extent size defines the size of a physical block in the volume group. All the disk space in a volume group is handled in chunks of this size. This value is normally set to 4 MB and allows for a maximum size of 256 GB for physical and logical volumes. The physical extent size should only be increased, for example, to 8, 16, or 32 MB, if you need logical volumes larger than 256 GB.

Figure 2.2 *Creating a Volume Group*



Create a Volume Group

Now we have to create a volume group. Typically you don't have to change anything, but if you are an expert, feel free to change our defaults:

Volume Group Name: system

Physical Extent Size: 4M

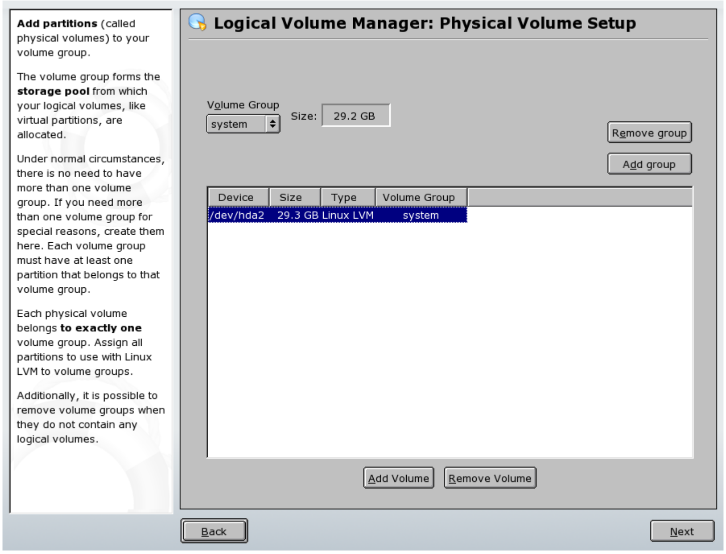
□ Use Old LVM1 Compatible Metadata Format

Configuring Physical Volumes

Once a volume group has been created, the following dialog lists all partitions with either the “Linux LVM” or “Linux native” type. No swap or DOS partitions are shown. If a partition is already assigned to a volume group, the name of the volume group is shown in the list. Unassigned partitions are indicated with “--”.

If there are several volume groups, set the current volume group in the selection box to the upper left. The buttons in the upper right enable creation of additional volume groups and deletion of existing volume groups. Only volume groups that do not have any partitions assigned can be deleted. All partitions that are assigned to a volume group are also referred to as a physical volumes (PV).

Figure 2.3 *Physical Volume Setup*



Add partitions (called physical volumes) to your volume group.

The volume group forms the **storage pool** from which your logical volumes, like virtual partitions, are allocated.

Under normal circumstances, there is no need to have more than one volume group. If you need more than one volume group for special reasons, create them here. Each volume group must have at least one partition that belongs to that volume group.

Each physical volume belongs **to exactly one** volume group. Assign all partitions to use with Linux LVM to volume groups.

Additionally, it is possible to remove volume groups when they do not contain any logical volumes.

Logical Volume Manager: Physical Volume Setup

Volume Group: system

Size: 29.2 GB

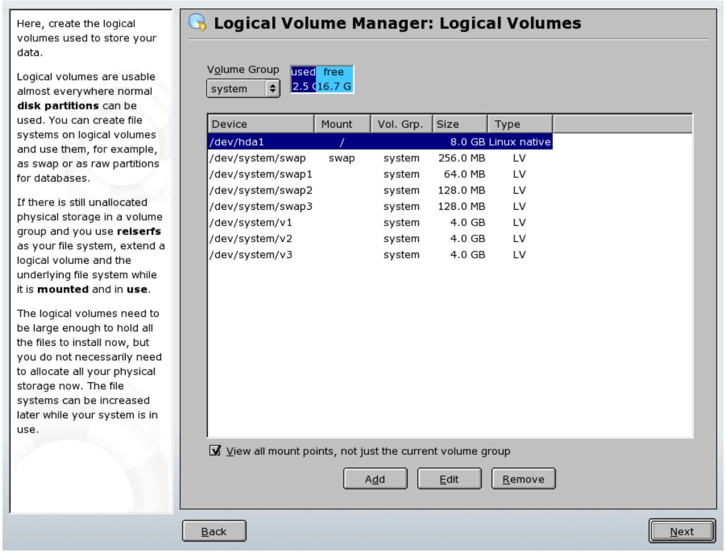
Device	Size	Type	Volume Group
/dev/hda2	29.3 GB	Linux LVM	system

To add a previously unassigned partition to the selected volume group, first click the partition then *Add Volume*. At this point, the name of the volume group is entered next to the selected partition. Assign all partitions reserved for LVM to a volume group. Otherwise, the space on the partition remains unused. Before exiting the dialog, every volume group must be assigned at least one physical volume. After assigning all physical volumes, click *Next* to proceed to the configuration of logical volumes.

Configuring Logical Volumes

After the volume group has been filled with physical volumes, define the logical volumes the operating system should use in the next dialog. Set the current volume group in a selection box to the upper left. Next to it, the free space in the current volume group is shown. The list below contains all logical volumes in that volume group. All normal Linux partitions to which a mount point is assigned, all swap partitions, and all already existing logical volumes are listed here. *Add*, *Edit*, and *Remove* logical volumes as needed until all space in the volume group has been exhausted. Assign at least one logical volume to each volume group.

Figure 2.4 *Logical Volume Management*



Here, create the logical volumes used to store your data.

Logical volumes are usable almost everywhere normal **disk partitions** can be used. You can create file systems on logical volumes and use them, for example, as swap or as raw partitions for databases.

If there is still unallocated physical storage in a volume group and you use **reiserfs** as your file system, extend a logical volume and the underlying file system while it is **mounted** and in **use**.

The logical volumes need to be large enough to hold all the files to install now, but you do not necessarily need to allocate all your physical storage now. The file systems can be increased later while your system is in use.

Logical Volume Manager: Logical Volumes

Volume Group: system

used	free
2.5 G	16.7 G

Device	Mount	Vol. Grp.	Size	Type
/dev/hda1	/		8.0 GB	Linux native
/dev/system/swap	swap	system	256.0 MB	LV
/dev/system/swap1		system	64.0 MB	LV
/dev/system/swap2		system	128.0 MB	LV
/dev/system/swap3		system	128.0 MB	LV
/dev/system/v1		system	4.0 GB	LV
/dev/system/v2		system	4.0 GB	LV
/dev/system/v3		system	4.0 GB	LV

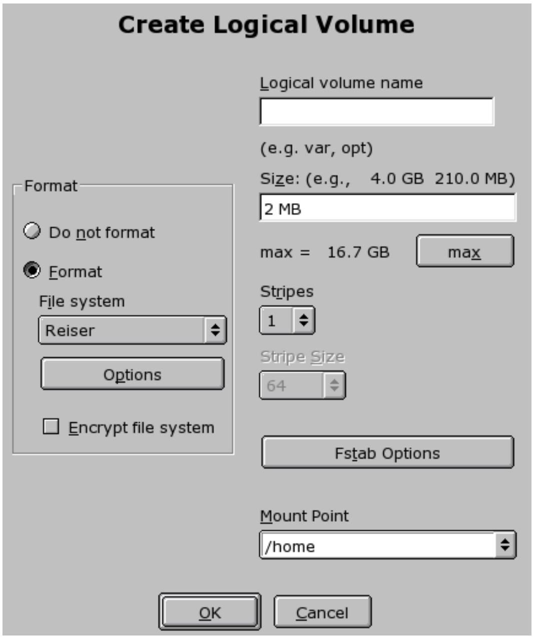
☑ View all mount points, not just the current volume group

To create a new logical volume, click *Add* and fill out the pop-up that opens. As for partitioning, enter the size, file system, and mount point. Normally, a file system, such as reiserfs or ext2, is created on a logical volume and is then designated a mount point. The files stored on this logical volume can be found at this mount point on the installed system. Additionally it is possible to distribute the data stream in the logical volume among several physical volumes (striping). If these physical volumes reside on different hard disks, this generally results in a better reading and writing performance (like RAID 0). However, a striping LV with `n` stripes can only be created correctly if the hard disk space required by the LV can be distributed evenly to `n` physical volumes. If, for example, only two physical volumes are available, a logical volume with three stripes is impossible.

WARNING: Striping

YaST has no chance at this point to verify the correctness of your entries concerning striping. Any mistake made here is apparent only later when the LVM is implemented on disk.

Figure 2.5 *Creating Logical Volumes*



Create Logical Volume

Format

○ Do not format

● Format

File system: Reiser

☐ Encrypt file system

Logical volume name:

(e.g. var, opt)

Size: (e.g., 4.0 GB 210.0 MB)

2 MB

max = 16.7 GB

Stripes: 1

Stripe Size: 64

Mount Point: /home

If you have already configured LVM on your system, the existing logical volumes can be entered now. Before continuing, assign appropriate mount points to these logical volumes too. With *Next*, return to the YaST Expert Partitioner and finish your work there.

Direct LVM Management

If you already have configured LVM and only want to change something, there is an alternative way to do that. In the YaST Control Center, select *System* → *LVM*. Basically this dialog allows the same actions as described above with the exception of physical partitioning. It shows the existing physical volumes and logical volumes in two lists and you can manage your LVM system using the methods already described.

2.3 Soft RAID Configuration

The purpose of RAID (redundant array of inexpensive disks) is to combine several hard disk partitions into one large *virtual* hard disk to optimize performance, data security, or both. Using this method, however, one advantage is sacrificed for another. Most RAID controllers use the SCSI protocol because it can address a larger number of hard disks in a more effective way than the IDE protocol and is more suitable for parallel processing of commands. There are some RAID controllers that support IDE or SATA hard disks. Refer to the Hardware Database at `http://cdb.suse.de`.

2.3.1 Soft RAID

Like a RAID controller, which can often be quite expensive, soft RAID is also able to take on these tasks. SUSE Linux offers the option of combining several hard disks into one soft RAID system with the help of YaST—a very reasonable alternative to hardware RAID. RAID implies several strategies for combining several hard disks in a RAID system, each of them having different goals, advantages and characteristics. These variations are commonly known as *RAID levels*.

Common RAID levels are:

RAID 0

This level improves the performance of your data access by spreading out blocks of each file across multiple disk drives. Actually, this is not really a RAID, because it does not provide data backup, but the name *RAID 0* for this type of system has become the norm. With RAID 0, two or more hard disks are pooled together. The performance is very good, but the RAID system is destroyed and your data lost if even one hard disk fails.

RAID 1

This level provides adequate security for your data, because the data is copied to another hard disk 1:1. This is known as *hard disk mirroring*. If a disk is destroyed, a copy of its contents is available on another one. All of them except one could be damaged without endangering your data. The writing performance suffers a little in the copying process compared to when using single disk access (ten to twenty percent slower), but read access is significantly faster in comparison to any one of the normal physical hard disks, because the data is duplicated so can be parallel scanned. Generally it can be said that Level 1 provides nearly twice the read transaction rate of single disks and almost the same write transaction rate as single disks.

RAID 2 and RAID 3

These are not typical RAID implementations. Level 2 stripes data at the bit level rather than the block level. Level 3 provides byte-level striping with a dedicated parity disk and cannot service simultaneous multiple requests. Both levels are only rarely used.

RAID 4

Level 4 provides block-level striping just like Level 0 combined with a dedicated parity disk. In the case of a data disk failure, the parity data is used to create a replacement disk. However, the parity disk may create a bottleneck for write access. Nevertheless, Level 4 is sometimes used.

RAID 5

RAID 5 is an optimized compromise between Level 0 and Level 1 in terms of performance and redundancy. The hard disk space equals the number of disks used minus one. The data is distributed over the hard disks as with RAID 0. *Parity blocks*, created on one of the partitions, are there for security reasons. They are linked to each other with XOR, enabling the contents, via XOR, to be reconstructed by the corresponding parity block in case of system failure. With RAID 5, no more than one hard disk can fail at the same time. If one hard disk fails, it must be replaced as soon as possible to avoid the risk of losing data.

Other RAID Levels

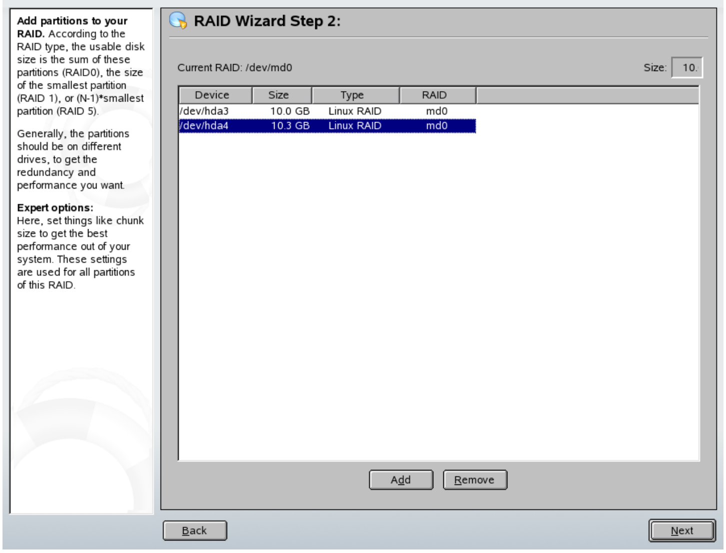
Several other RAID levels have been developed (RAIDn, RAID 10, RAID 0+1, RAID 30, RAID 50, etc.), some of them being proprietary implementations created by hardware vendors. These levels are not very widespread, so are not explained here.

2.3.2 Soft RAID Configuration with YaST

The YaST soft RAID configuration can be reached from the YaST Expert Partitioner, described in Section “Partitioner” (Chapter 3, *System Configuration with YaST*, ↑Start-Up). This professional partitioning tool enables you to edit and delete existing partitions and create new ones that should be used with soft RAID. There, create RAID partitions by first clicking *Create → Do not format* then selecting *0xFD Linux RAID* as the partition identifier. For RAID 0 and RAID 1, at least two partitions are needed—for RAID 1, usually exactly two and no more. If RAID 5 is used, at least three partitions are required. It is recommended to take only partitions of the same size. The RAID partitions should be stored on different hard disks to decrease the risk of losing data if one is defective (RAID 1 and 5) and to optimize the performance of RAID 0. After creating all the partitions to use with RAID, click *RAID* → *Create RAID* to start the RAID configuration.

In the next dialog, choose between RAID levels 0, 1, and 5 (see Section 2.3.1, “Soft RAID” (page 65) for details). After *Next* is clicked, the following dialog lists all partitions with either the “Linux RAID” or “Linux native” type (see Figure 2.6, “RAID Partitions” (page 67)). No swap or DOS partitions are shown. If a partition is already assigned to a RAID volume, the name of the RAID device (e.g., `/dev/md0`) is shown in the list. Unassigned partitions are indicated with “--”.

Figure 2.6 *RAID Partitions*



Add partitions to your RAID. According to the RAID type, the usable disk size is the sum of these partitions (RAID0), the size of the smallest partition (RAID 1), or (N-1)*smallest partition (RAID 5).

Generally, the partitions should be on different drives, to get the redundancy and performance you want.

Expert options: Here, set things like chunk size to get the best performance out of your system. These settings are used for all partitions of this RAID.

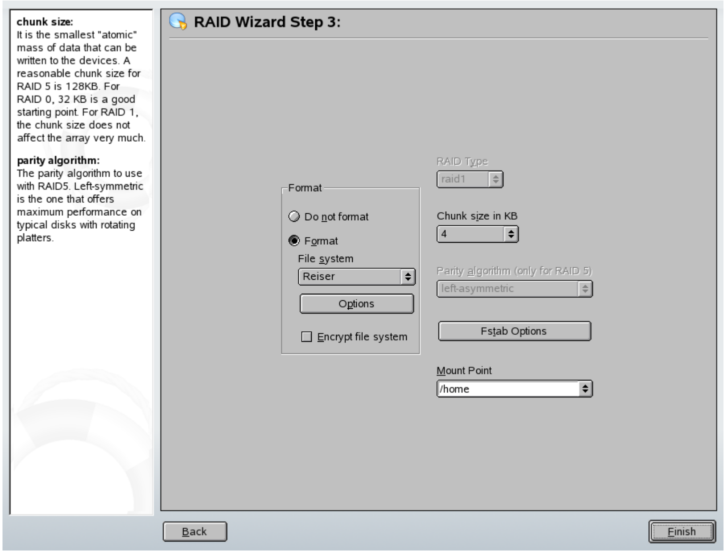
RAID Wizard Step 2:

Current RAID: /dev/md0 Size: 10.

Device	Size	Type	RAID
/dev/hda3	10.0 GB	Linux RAID	md0
/dev/hda4	10.3 GB	Linux RAID	md0

To add a previously unassigned partition to the selected RAID volume, first click the partition then *Add*. At this point, the name of the RAID device is entered next to the selected partition. Assign all partitions reserved for RAID. Otherwise, the space on the partition remains unused. After assigning all partitions, click *Next* to proceed to the settings dialog where you can fine-tune the performance (see Figure 2.7, “File System Settings” (page 68)).

Figure 2.7 *File System Settings*



chunk size:
It is the smallest "atomic" mass of data that can be written to the devices. A reasonable chunk size for RAID 5 is 128KB. For RAID 0, 32 KB is a good starting point. For RAID 1, the chunk size does not affect the array very much.

parity algorithm:
The parity algorithm to use with RAID5. Left-symmetric is the one that offers maximum performance on typical disks with rotating platters.

RAID Wizard Step 3:

Format
○ Do not format
● Format
File system: Reiser
□ Encrypt file system

RAID Type: raid1
Chunk size in KB: 4
Parity algorithm (only for RAID 5): left-asymmetric
Mount Point: /home

As with conventional partitioning, set the file system to use as well as encryption and the mount point for the RAID volume. Checking *Persistent Superblock* ensures that the RAID partitions are recognized as such when booting. After completing the configuration with *Finish*, see the `/dev/md0` device and others indicated with *RAID* in the expert partitioner.

2.3.3 Troubleshooting

Check the file `/proc/mdstats` to find out whether a RAID partition has been destroyed. In the event of a system failure, shut down your Linux system and replace the defective hard disk with a new one partitioned the same way. Then restart your system and enter the command `mdadm /dev/mdX --add /dev/sdX`. Replace 'X' with your particular device identifiers. This integrates the hard disk automatically into the RAID system and fully reconstructs it.

2.3.4 For More Information

Configuration instructions and more details for soft RAID can be found in the HOWTOs at:

- /usr/share/doc/packages/raidtools/Software-RAID.HOWTO.html

- http://en.tldp.org/HOWTO/Software-RAID-HOWTO.html

Linux RAID mailing lists are also available, such as http://www.mail-archive.com/linux-raid@vger.rutgers.edu.

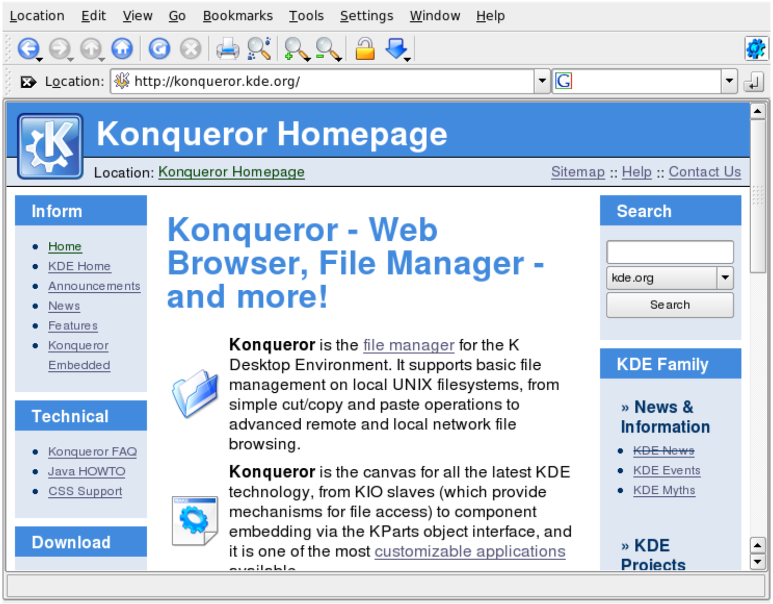
Part II Internet

3

The Web Browser Konqueror

Konqueror is not only a versatile file manager. It is also a modern Web browser. If you start the browser with the icon in the panel, Konqueror opens with the Web browser profile. As a browser, Konqueror offers tabbed browsing, the possibility of saving Web pages with graphics, Internet keywords, bookmarks, and support for Java and JavaScript.

Figure 3.1 *The Browser Window of Konqueror*



Konqueror - Web Browser, File Manager - and more!

Konqueror is the file manager for the K Desktop Environment. It supports basic file management on local UNIX filesystems, from simple cut/copy and paste operations to advanced remote and local network file browsing.

Konqueror is the canvas for all the latest KDE technology, from KIO slaves (which provide mechanisms for file access) to component embedding via the KParts object interface, and it is one of the most customizable applications available

Start Konqueror from the main menu or by entering the command `konqueror`. To load a Web page, enter its address in the location bar, for example, http://www.suse.com. Konqueror now tries to reach the address and display the page. Entering the protocol at the beginning of the address (`http://` in this case) is not strictly required. The program is able to complete the address automatically, but this only works reliably with Web addresses. For an FTP address, always enter `ftp://` at the beginning of the input field.

3.1 Tabbed Browsing

If you often use more than one Web page at a time, tabbed browsing may make it easier to switch between them. Load Web sites in separate tabs within one window. The advantage is that you keep more control over your desktop because you only have one main window. After logout, the KDE session management allows for saving your Web session in Konqueror. The next time you log in, Konqueror loads the exact URLs visited last time.

To open a new tab, select *Window* → *New Tab* or press Ctrl + Shift + N. To change the behavior of tabs, go to *Settings* → *Configure Konqueror*. In the dialog box that opens, select *Web Behavior* → *Tabbed Browsing*. To open new tabs instead of windows, enable *Open links in new tab instead of in new window*. You can also hide the tab bar with *Hide the tab bar when only one tab is open*. To see more options, press *Advanced Options*.

You can save your tabs with URLs and the position of the window in a profile. This is a bit different from the session managment mentioned above. With profiles, you have your saved tabs at hand and without intensive start-up time like with session management.

In Konqueror, go to *Settings* → *Configure View Profiles* and give your profile a name. You can save the window size in the profile, too, with the respective option. Make sure that *Save URLs in profile* is selected. Approve with *Save*. Next time you need your “tab collection,” go to *Settings* → *Load View Profile* and see the name listed in the menu. After selecting the name, Konqueror restores your tabs.

3.2 Saving Web Pages and Graphics

As in other browsers, you can save Web pages. To do this, select *Location* → *Save as* and specify a name for your HTML file. However, images are not saved. To archive an entire Web page including the images, select *Tools* → *Archive Web Page*. Konqueror suggests a filename that you can usually accept. The filename ends with `.war`, the extension for Web archives. To view the saved Web archive later, simply click the respective file and the Web page is displayed in Konqueror along with its images.

3.3 Internet Keywords

Searching the Web using Konqueror is very easy. Konqueror defines over 70 search filters for you, all with a specific shortcut. To search for a certain topic on the Internet, enter the shortcut and the keyword separated by a colon. The relevant page containing the search results is then displayed.

To see the already definied shortcuts, go to *Settings* → *Configure Konqueror*. In the dialog box that appears, select *Web Shortcuts*. Now you can see the names of the search providers and the shortcuts. Konqueror definies lots of search filters: the “classical” search engines, like Google, Yahoo, and Lycos, and a number of filters for less common purposes, like an acronym database, the Internet movie database, or KDE application searches.

If you do not find your preferred search engine here, easily define a new one. For example, to search our support database for some interesting articles, normally go to `http://portal.suse.com/`, find the search page, and enter your query. This can be simplified by using shortcuts. In the mentioned dialog box, select *New* and give your shortcut a name in *Search provider name*. Enter your abbreviations in *URI shortcuts*. There can more than one, separated by commas. The important text field is *Search URI*. Pressing Shift + F1 and clicking in the field opens a small help. The search query is specified as `\{@}`. The challenge is inserting this at the correct position. In this case, the settings for the SUSE support database looks like this: *Search provider name* is `SUSE Support Database`, *Search URI* is (one line) `https://portal.suse.com/PM/page/search.pm?q=\{@}&t=optionSdbKeywords&m=25&l=en&x=true`, and *URI shortcuts* is `sdb_en`.

After approving with *Ok* two times, enter your query in Konqueror's location bar, for example, `sdb_en:kernel`. The result is displayed in the current window.

3.4 Bookmarks

Instead of remembering and reentering addresses for sites visited often, you can bookmark these URLs using the *Bookmark* menu. Apart from Web page addresses, you can also bookmark any directories of your local disk in this way.

To create a new bookmark in Konqueror, click *Bookmarks* → *Add Bookmark*. Any bookmarks added previously are included as items in the menu. It is a good idea to arrange the bookmark collection by subjects in a hierarchical structure, so that you do not lose track of the different items. Create a new subgroup for your bookmarks with *New Bookmark Folder*. Selecting *Bookmarks* → *Edit Bookmarks* opens the bookmark editor. Use this program to organize, rearrange, add, and delete bookmarks.

If you are using Netscape, Mozilla, or Firefox as additional browsers, it is not necessary to recreate your bookmarks. *File* → *Import* → *Import Netscape Bookmarks* in the bookmark editor enables you to integrate your Netscape and Mozilla bookmarks into your most current collection. The reverse is also possible via *Export as Netscape Bookmarks*.

Change your bookmarks by right-clicking the entry. A pop-up menu appears in which to select the desired action (cut, copy, delete, etc.). When you are satisfied with the result, save the bookmarks with *File* → *Save*. If you only want to change the name or link, just right-click the entry in the bookmark toolbar and select *Properties*. Change the name and location and confirm with *Update*.

To save your bookmark list and have instant access to it, make your bookmarks visible in Konqueror. Select *Settings* → *Toolbars* → *Bookmark Toolbar (Konqueror)*. A bookmark panel is automatically displayed in the current Konqueror window.

3.5 Java and JavaScript

Do not confuse these two languages. Java is an object-oriented, platform-independent programming language from Sun Microsystems. It is frequently used for small programs (applets), which are executed over the Internet for things like online banking, chatting, and shopping. JavaScript is an interpreted scripting language mainly used for the dynamic structuring of Web pages, for example, for menus and other effects.

Konqueror allows you to enable or disable these two languages. This can even be done in a domain-specific way, which means that you can permit access for some hosts and block access for others. Java and JavaScript are often disabled for security reasons. Unfortunately, some Web pages require JavaScript for correct display.

3.6 For More Information

For any questions or problems that arise when working with Konqueror, refer to the application's handbook, which is available from the *Help* menu. Konqueror also has a Web page, located at `http://www.konqueror.org`.

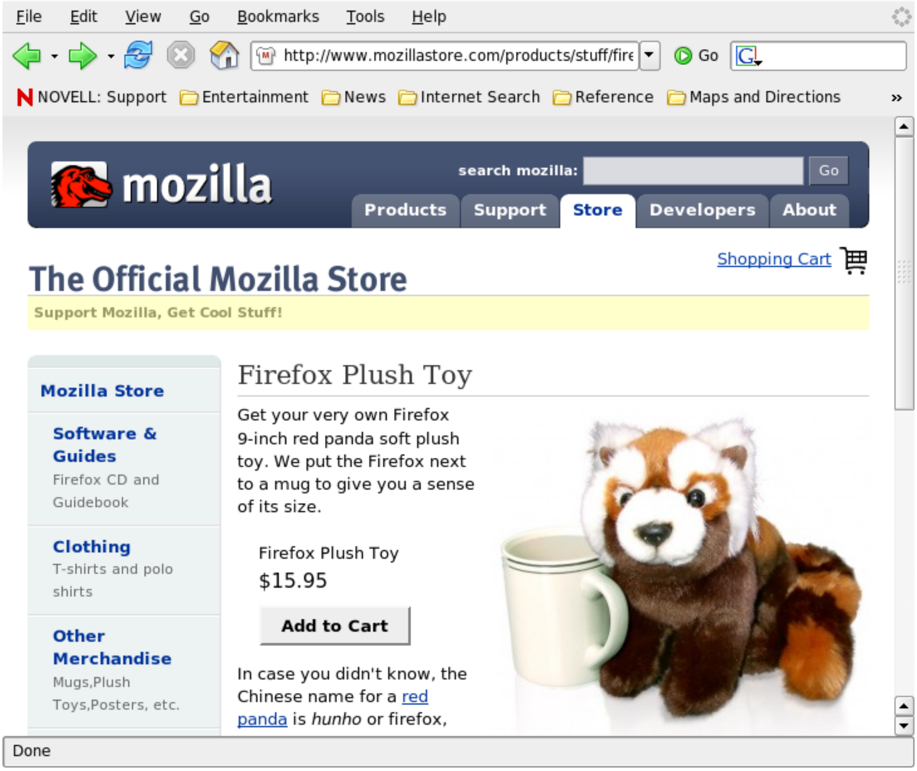
4 Firefox

Included with your SUSE Linux is the Mozilla Firefox Web browser. With features like tabs, pop-up window blocking, and download and image management, Firefox combines the latest Web technologies. View more than one Web page in a single window. Suppress annoying advertisements and disable images that only slow you down. Its easy access to different search engines helps you find the information you need. Start the program from the main menu or by entering the command `firefox`. The main program features are described in the following sections.

4.1 Navigating Web Sites

Firefox has much the same look and feel as other browsers. It is shown in Figure 4.1, "The Browser Window of Firefox" (page 80). The navigation toolbar includes *Forward* and *Back* and a location bar for a Web address. Bookmarks are also available for quick access. For more information about the various Firefox features, use the *Help* menu.

Figure 4.1 *The Browser Window of Firefox*



The Official Mozilla Store

Support Mozilla, Get Cool Stuff!

Firefox Plush Toy

Get your very own Firefox 9-inch red panda soft plush toy. We put the Firefox next to a mug to give you a sense of its size.

Firefox Plush Toy
$15.95

In case you didn't know, the Chinese name for a red panda is *hunho* or firefox,

4.1.1 Tabbed Browsing

If you often use more than one Web page at a time, tabbed browsing may make it easier to switch between them. Load Web sites in separate tabs within one window.

To open a new tab, select *File* → *New Tab*. This opens an empty tab in the Firefox window. Alternatively, right-click a link and select *Open link in new tab*. Right-click the tab itself to access more tab options. You can create a new tab, reload one or all existing tabs, or close them.

4.1.2 Using the Sidebar

Use the left side of your browser window for viewing bookmarks or the browsing history. Extensions may add new ways to use the sidebar as well. To display the Sidebar, select *View* → *Sidebar* and select the desired contents.

4.2 Finding Information

There are two ways to find information in Firefox: the search bar and the find bar. The search bar looks for pages while the find bar looks for things on the current page.

4.2.1 Using the Search Bar

Firefox has a search bar that can access different engines, like Google, Yahoo, or Amazon. For example, if you want to find information about SUSE using the current engine, click in the search bar, type `SUSE`, and hit Enter. The results appear in your window. To choose your search engine, click the icon in the search bar. A menu opens with a list of available search engines.

4.2.2 Using the Find Bar

To search inside a Web page, click *Edit* → *Find in This Page* or press Ctrl + F and the find bar opens. Usually, it is displayed at the bottom of a window. Type your query in the input field. Firefox highlights all occurrences of this phrase. With *Highlight*, enable and disable the highlighting.

4.3 Managing Bookmarks

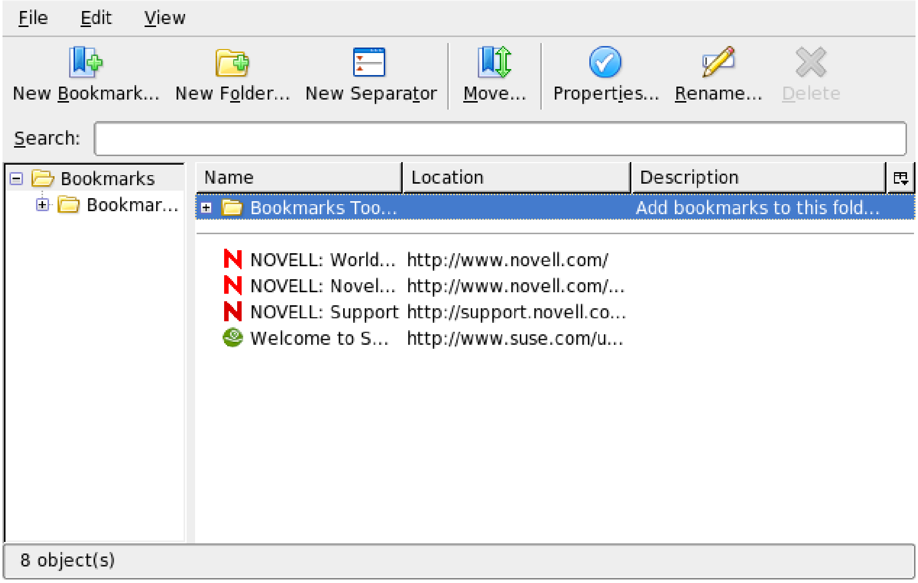
Bookmarks offer a convenient way of saving links to your favorite Web sites. To add the current Web site to your list of bookmarks, click *Bookmarks* → *Bookmark This Page*. If your browser currently displays multiple Web sites on tabs, only the URL on the currently selected tab is added to your list of bookmarks.

When adding a bookmark, you can specify an alternative name for the bookmark and select a folder where Firefox should store it. To remove a Web site from the bookmarks list, click *Bookmarks*, right-click the bookmark in the list, then click *Delete*.

4.3.1 Using the Bookmark Manager

The bookmark manager can be used to manage the properties (name and address location) for each bookmark and organize the bookmarks into folders and sections. It resembles Figure 4.2, "Using the Firefox Bookmark Manager" (page 82).

Figure 4.2 *Using the Firefox Bookmark Manager*



Name	Location	Description
Bookmarks Too...		Add bookmarks to this fold...
NOVELL: World...	http://www.novell.com/	
NOVELL: Novel...	http://www.novell.com/...	
NOVELL: Support	http://support.novell.co...	
Welcome to S...	http://www.suse.com/u...	

8 object(s)

To open the bookmark manager, click *Bookmark* → *Manage Bookmarks*. A window opens and displays your bookmarks. With *New Folder*, create a new folder with a name and a description. If you need a new bookmark, click *New Bookmark*. This let you insert the name, location, keywords, and also a description. The keyword is a shortcut to your bookmark. If you need your newly created bookmark in the sidebar, check *Load this bookmark in the sidebar*.

4.3.2 Migrating Bookmarks

If you used a different browser in the past, you probably want to use your preferences and bookmarks in Firefox, too. At the moment, you can import from Netscape 4.x, 6, 7, Mozilla 1.x, and Opera.

To import your settings, click *File* → *Import*. Select the browser from which to import settings. After you click *Next*, your settings are imported. Find your imported bookmarks in a newly created folder, beginning with `From`.

4.3.3 Live Bookmarks

Live Bookmarks displays headlines in your bookmarks menu and keep you up to date with the latest news. This enables you to save time with one glance from your favorite sites.

Many sites and blogs support this format. A Web site indicates this by showing an orange rectangle with RSS inside on the bottom right corner. Click it and choose *Subscribe to NAME OF THE FEED*. A dialog box opens where you can select the name and location of your Live Bookmark. Confirm with *Add*.

Some sites do not tell Firefox that they support a news feed, although they actually do. To manually add a Live Bookmark, you need the URL of the feed. Do the following:

Procedure 4.1 *Adding a Live Bookmark Manually*

1. Open the bookmark manager with *Bookmarks* → *Manage Bookmarks*. A new window opens.

2. Select *File* → *New Live Bookmark*. A dialog box opens.

3. Insert a name for the Live Bookmark and add your URL, for example, `http://www.novell.com/newsfeeds/rss/coolsolutions.xml`. Firefox updates your Live Bookmarks.

4. Close your bookmark manager.

4.4 Using the Download Manager

With the help of the download manager, keep track of your current and past downloads. To open the download manager, click *Tools* → *Downloads*. Firefox opens a window with your downloads. While downloading a file, see a progress bar and the current file. Pause the download and resume it later, if necessary. To open a downloaded file, click *Open*. With *Remove*, erase it from the medium. If you need information about the file, right-click the filename and choose *Properties*.

If you need further control of the Download Manager, open the configuration window from *Edit* → *Preferences* and go to the *Downloads* tab. Here, determine the download folder, how the manager behaves, and some configuration of file types.

4.5 Customizing Firefox

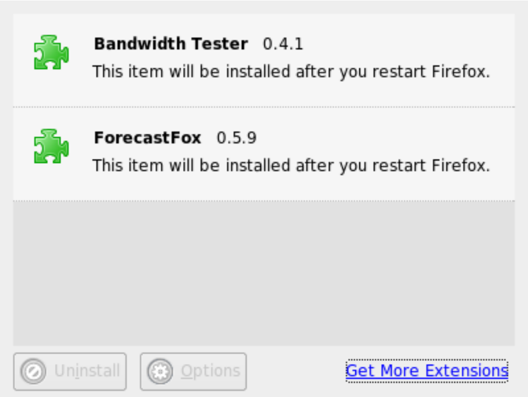
With the ability to install extensions, change themes, and add smart keywords for your online searches, Firefox can be customized extensively.

4.5.1 Extensions

Mozilla Firefox is a multifunctional application, which means that you can download and install add-ons, known as extensions. For example, add a new download manager and mouse gestures. This has the advantage that Firefox itself stays small and unbloated.

To add an extension, click *Tools* → *Extensions*. In the bottom-right corner, click *Get More Extensions* to open the Mozilla extensions update Web page where you can choose from a variety of available extensions. Click the extension to install then click the install link to download and install it. When you restart Firefox, the new extension is functional. You can also look at the various extensions at `http://update.mozilla.org/`
.

Figure 4.3 *Installing Firefox Extensions*



Bandwidth Tester 0.4.1
This item will be installed after you restart Firefox.

ForecastFox 0.5.9
This item will be installed after you restart Firefox.

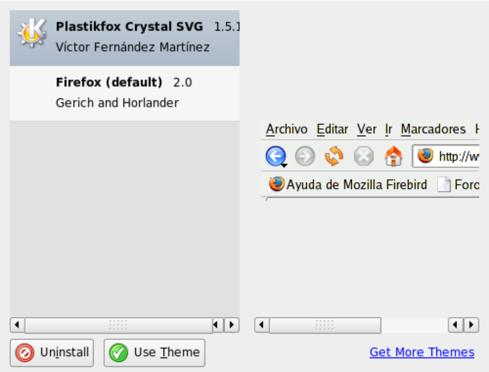
4.5.2 Changing Themes

If you do not like the standard look and feel of Firefox, install a new *theme*. Themes do not change the functionality, only the appearance of the browser. When installing a theme, Firefox asks for confirmation first. Allow the installation or cancel it. After a successful installation, you can enable the new theme.

1. Click *Tools* → *Theme*.

2. A new dialog appears. Click *Get More Themes*. If you already installed a theme, find it in the list, as in Figure 4.4, “Installing Firefox Themes” (page 86).

Figure 4.4 *Installing Firefox Themes*


Plastikfox Crystal SVG 1.5.1
Víctor Fernández Martínez
Firefox (default) 2.0
Gerich and Horlander
Archivo Editar Ver Ir Marcadores
http://w
Ayuda de Mozilla Firebird
Foro
Uninstall
Use Theme
Get More Themes

3 A new window appears with the Web site `https://update.mozilla.org`.

4 Choose a theme and click *Install Now*.

5 Confirm the download and installation.

6 After downloading the theme, a dialog appears and informs you about your list of themes. Activate the new theme with *Use Theme*.

7 Close the window and restart Firefox.

If a theme is installed, you can always switch to a different theme without restarting by clicking *Tools* → *Themes* then *Use Theme*. If you do not use a theme anymore, you can delete it in the same dialog with *Uninstall*.

4.5.3 Adding Smart Keywords to Your Online Searches

Searching the Internet is one of the main tasks a browser can perform for you. Firefox lets you define your own *smart keywords*: abbreviations to use as a “command” for searching the Web. For example, if you use Wikipedia often, use a smart keyword to simplify this task:

1. Go to `http://en.wikipedia.org`.

2. After Firefox displays the Web page, see the search text field. Right-click it then choose *Add a Keyword for this Search* from the menu that opens.

3. The *Add Bookmark* dialog appears. In *Name*, name this Web page, for example, *Wikipedia (en)*.

4. For *Keyword*, enter your abbreviation of this Web page, for example, *wiki*.

5. With *Create in*, choose the location of the entry in your bookmarks section. You can put it into *Quick Searches*, but any other level is also appropriate.

6. Finalize with *Add*.

You have successfully generated a new keyword. Whenever you need to look into Wikipedia, you do not have to use the entire URL. Just type `wiki Linux` to view an entry about Linux.

4.6 Printing from Firefox

Configure the way Firefox prints the content it displays using the *Page Setup* dialog. Click *File* → *Page Setup* then go to the *Format & Options* tab to select the orientation of your print jobs. You can scale or make it adjust automatically. To print a background, select *Print Background (colors & images)*. Click the *Margins & Header/Footer* tab to adjust margins and select what to include in the headers and footers.

After you configured your settings, print a Web page with *File* → *Print*. Select the printer or a file in which to save the output. With *Properties*, set the paper size, specify the print command, choose grayscale or color, and determine the margins. When satisfied with your settings, approve with *Print*.

4.7 For More Information

Get more information about Firefox from the official home page at `http://www.mozilla.org/products/firefox/`. Refer to the integrated help to find out more about certain options or features.

Linphone—VoIP for the Linux Desktop

5

Linphone is a small Web phone application for your Linux desktop. It allows you to make two-party calls over the Internet. There is no need for special hardware items: a standard workstation with a properly configured sound card, microphone, and speakers or headphones is all you need to get started with Linphone.

5.1 Configuring Linphone

Before you start using Linphone there are some basic decisions to make and some configuration tasks to complete. First, determine and configure the run mode of Linphone, determine the connection type to use, then start the Linphone configuration (*Go → Preferences*) to make the necessary adjustments.

5.1.1 Determining the Run Mode of Linphone

Linphone can be run in two different different modes, depending on the type of desktop you run and on its configuration.

Normal Application
After the Linphone software has been installed, it can be started via the GNOME and KDE application menus or via the command line. When Linphone is not running, incoming calls cannot be received.

GNOME Panel Applet

Linphone can be added to the GNOME panel. Right-click an empty area in the panel, select *Add to Panel*, and select Linphone. Linphone is then permanently added to the panel and automatically started on login. As long as you do not receive any incoming calls, it runs in the background. As soon as you get an incoming call, the main window opens and you can receive the call. To open the main window to call someone, just click the applet icon.

5.1.2 Determining the Connection Type

There are several different ways to make a call in Linphone. How you make a call and how you reach the other party is determined by the way you are connected to the network or the Internet.

Linphone uses the session initiation protocol (SIP) to establish a connection with a remote host. In SIP, each party is identified by a SIP URL:

```
sip:username@hostname
```

`username` is your login on your Linux machine and *`hostname`* the name of the computer you are using. If you use a SIP provider, the URL would look like the following example:

```
sip:username@sipserver
```

`username` is the username chosen when registering at a SIP server. *`sipserver`* is the address of the SIP server or your SIP provider. For details on the registration procedure, refer to Section 5.1.5, “Configuring the SIP Options” (page 92) and check the provider's registration documentation. For a list of providers suitable for your purpose, check the Web pages mentioned in Section 5.8, “For More Information” (page 99).

The URL to use is determined by the type of connection you choose. If you chose to call another party directly without any further routing by a SIP provider, you would enter a URL of the first type. If you chose to call another party via a SIP server, you would enter a URL of the second type.

Calling in the Same Network

If you intend to call a friend or coworker belonging to the same network, you just need the correct username and hostname to create a valid SIP URL. The same applies if this person wants to call you. As long as there is no firewall between you and the other party, no further configuration is required.

Calling across Networks or the Internet (Static IP Setup)

If you are connected to the Internet using a static IP address, anyone who wants to call you just needs your username and the hostname or IP address of your workstation to create a valid SIP URL, as described in Section “Calling in the Same Network” (page 90). If you or the calling party are located behind a firewall that filters incoming and outgoing traffic, open the SIP port (`5060`) and the RTP port (`7078`) on the firewall machine to enable Linphone traffic across the firewall.

Calling across Networks or the Internet (Dynamic IP Setup)

If your IP setup is not static—if you dynamically get a new IP address every time you connect to the Internet—it is impossible for any caller to create a valid SIP URL based on your username and an IP address. In these cases, either use the services offered by a SIP provider or use a DynDNS setup to make sure that an external caller gets connected to the right host machine. More information about DynDNS can be found at `http://en.wikipedia.org/wiki/Dynamic_DNS`.

Calling across Networks and Firewalls

Machines hidden behind a firewall do not reveal their IP address over the Internet. Thus, they cannot be reached directly from anyone trying to call a user working at such a machine. Linphone supports calling across network borders and firewalls by using a SIP proxy or relaying the calls to a SIP provider. Refer to Section 5.1.5, “Configuring the SIP Options” (page 92) for a detailed description of the necessary adjustments for using an external SIP server.

5.1.3 Configuring the Network Parameters

Most of the settings contained in the *Network* tab do not need any further adjustments. You should be able to make your first call without changing them.

NAT Traversal Options

Enable this option only if you find yourself in a private network behind a firewall and if you do not use a SIP provider to route your calls. Select the check box and enter the IP address of the firewall machine in dot notation, for example, `192.168.34.166`.

RTP Properties

Linphone uses the real-time transport protocol (RTP) to transmit the audio data of your calls. The port for RTP is set to `7078` and should not be modified, unless you have another application using this port. The jitter compensation parameter is used to control the number of audio packages Linphone buffers before actually playing them. By increasing this parameter, you improve the quality of transmission. The more packages buffered, the greater a chance for “late comers” to be played back. On the other hand increasing the number of buffered packages also increases the latency—you hear the voice of your counterpart with a certain delay. When changing this parameter, carefully balance these two factors.

Other

If you use a combination of VoIP and landline telephony, you might want to use the dual tone multiplexed frequency (DTMF) technology to trigger certain actions, like a remote check of your voice mail just by punching certain keys. Linphone supports two protocols for DTMF transmission, SIP INFO and RTP rfc2833. If you need DTMF functionality in Linphone, choose a SIP provider that supports one of these protocols. For a comprehensive list of VoIP providers, refer to Section 5.8, “For More Information” (page 99).

5.1.4 Configuring the Sound Device

Once your sound card has been properly detected by Linux, Linphone automatically uses the detected device as the default sound device. Leave the value of *Use sound device* as it is. Use *Recording source* to determine which recording source should be used. In most cases, this would be a microphone (`micro`). To select a custom ring sound, use *Browse* to choose one and test your choice using *Listen*. Click *Apply* to accept your changes.

5.1.5 Configuring the SIP Options

The *SIP* dialog contains all SIP configuration settings.

SIP Port

Determine on which port the SIP user agent should run. The default port for SIP is `5060`. Leave the default setting unchanged unless you know of any other application or protocol that needs this port.

Identity

Anyone who wants to call you directly without using a SIP proxy or a SIP provider needs to know your valid SIP address. Linphone creates a valid SIP address for you.

Remote Services

This list holds one or more SIP service providers where you have created a user account. Server information can be added, modified, or deleted at any time. See Adding a SIP Proxy and Registering at a Remote SIP Server (page 93) to learn about the registration procedure.

Authentication Information

To register at a remote SIP server, provide certain authentication data, such as a password and username. Linphone stores this data once provided. To discard this data for security reasons, click *Clear all stored authentification data*.

The *Remote services* list can be filled with several addresses of remote SIP proxies or service providers.

Procedure 5.1 *Adding a SIP Proxy and Registering at a Remote SIP Server*

1. Choose a suitable SIP provider and create a user account there.

2. Start Linphone.

3. Go to *Go* → *Preferences* → *SIP*.

4. Click *Add proxy/registrar* to open a registration form.

5. Fill in the appropriate values for *Registration Period*, *SIP Identity*, *SIP Proxy* and *Route*. If working from behind a firewall, always select *Send registration* and enter an appropriate value for *Registration Period*. This resends the original registration data after a given time to keep the firewall open at the ports needed by Linphone. Otherwise, these ports would automatically be closed if the firewall did not receive any more packages of this type. Resending the registration data is also needed to keep the SIP server informed about the current status of the

connection and the location of the caller. For *SIP identity*, enter the SIP URL that should be used for local calls. To use this server also as a SIP proxy, enter the same data for *SIP Proxy*. Finally, add an optional route, if needed, and leave the dialog with *OK*.

5.1.6 Configuring the Audio Codecs

Linphone supports a several codecs for the transmission of voice data. Set your connection type and choose your preferred codecs from the list window. Codecs not suitable for your current connection type are red and cannot be selected.

5.2 Testing Linphone

Check your Linphone configuration using `sipomatic`, a small test program that can answer calls made from Linphone.

Procedure 5.2 *Testing a Linphone Setup*

1. Open a terminal.

2. Enter `sipomatic` at the command line prompt.

3. Start Linphone.

4. Enter `sip:robot@127.0.0.1:5064` as *SIP address* and click *Call or Answer*.

5. If Linphone is configured correctly, you will hear a phone ringing and, after a short while, you will hear a short announcement.

If you successfully completed this procedure, you can be sure that your audio setup and the network setup are working. If this test fails, check whether your sound device is correctly configured and whether the playback level is set to a reasonable value. If you still fail to hear anything, check the network setup including the port numbers for SIP and RTP. If any other application or protocol uses the defaults ports for these as proposed by Linphone, consider changing ports and retry.

5.3 Making a Call

Once Linphone is configured appropriately, making a call is straightforward. Depending on the type of call (see Section 5.1.2, "Determining the Connection Type" (page 90) for reference), the calling procedures differ slightly.

1. Start Linphone using the menu or a command line.

2. Enter the SIP address of the other party at the *SIP address* prompt. The address should look like `sip:username@domainname` or `username@hostname` for direct local calls or like `username@sipserver` or `userid@sipserver` for proxied calls or calls using the service of a SIP provider.

3. If using a SIP service provider or a proxy, select the appropriate proxy or provider from *Proxy to use* and provide the authentication data requested by this proxy.

4. Click *Call or Answer* and wait for the other party to pick up the phone.

5. Once you are done or wish to end the call, click *Release or Refuse* and leave Linphone.

If you need to tweak the sound parameters during a call, click *Show more* to show four tabs holding more options. The first one holds the *Sound* options for *Playback level* and *Recording level*. Use the sliders to adjust both volumes to fit your needs.

The *Presence* tab lets you set your online status. This information can be relayed to anyone who tries to contact you. If you are permanently away and wish to inform the calling party of this fact, just check *Away*. If you are just busy, but want the calling party to retry, check *Busy, I'll be back in ... min* and specify how long you will not be reachable. Once you are reachable again, set the status back to the default (*Reachable*). Whether another party can check your online status is determined by the *Subscribe Policy* set in the address book, as described in Section 5.5, "Using the Address Book" (page 96). If any party listed in your address book published their online status, you can monitor it using the *My online friends* tab.

The *DTMF* tab can be used to enter DTMF codes for checking voice mail. To check your voice mail, enter the appropriate SIP address and use the keypad in the *DTMF* tab to enter the voice mail code. Finally, click *Call or Answer* as if you were making an ordinary call.

5.4 Answering a Call

Depending on the run mode selected for Linphone, there are several ways you would notice an incoming call:

Normal Application

Incoming calls can only be received and answered if Linphone is already running. You then hear the ring sound on your headset or your speakers. If Linphone is not running, the call cannot be received.

GNOME Panel Applet

Normally, the Linphone panel applet would run silently without giving any notice of its existence. This changes as soon as a call comes in: the main window of Linphone opens and you hear a ring sound on your headset or speakers.

Once you have noticed an incoming call, just click *Call or Answer* to pick up the phone and start talking. If you do not want to accept this call, click *Release of Refuse*.

5.5 Using the Address Book

Linphone offers to manage your SIP contacts. Start the address book with *Go → Address book*. An empty list window opens. Click *Add* to add a contact.

The following entries need to be made for a valid contact:

Name

Enter the name of your contact. This may be a full name, but you can also use a nickname here. Choose something you easily remember this person as. If you choose to see this person's online status, this name is shown in the *My online friends* tab of the main window.

SIP Address

Enter a valid SIP address for your contact.

Proxy to Use

If needed, enter the proxy to use for this particular connection. In most cases, this would just be the SIP address of the SIP server you use.

Subscribe Policy

Your subscribe policy determines whether your presence or absence can be tracked by others.

To call any contact from the address book, select this contact with the mouse, click *Select* to make the address appear in the address field of the main window, and start the call with *Call or Answer* as usual.

5.6 Troubleshooting

I try to call someone, but fail to establish a connection.

There are several reasons why a call could fail:

Your connection to the Internet is broken.

Because Liphone uses the Internet to relay your calls, make sure that your computer is properly connected to and configured for the Internet. This can easily be tested by trying to view a Web page using your browser. If the Internet connection works, the other party might not be reachable.

The person you are calling is not reachable.

If the other party refused your call, you would not be connected. If Linphone is not running on the other party's machine while you are calling, you will not be connected. If the other party's Internet connection is broken, you cannot make the connection.

My call seems to connect, but I cannot hear anything.

First, make sure that your sound device is properly configured. Do this by launching any other application using sound output, such as a media player. Make sure that Linphone has sufficient permissions to open this device. Close all other programs using the sound device to avoid resource conflicts.

If the above checks were successful, but you still fail to hear anything, raise the recording and playback levels under the *Sound* tab.

The voice output on both ends sounds strangely clipped.

Try to adjust the jitter buffer using *RTP properties* in *Preferences* → *Network* to compensate for delayed voice packages. When doing this, be aware that it increases the latency.

DTMF does not work.

You tried to check your voice mail using the DTMF pad, but the connection could not be established. There are three different protocols used for the transmission of DTMF data, but only two of these are supported by Linphone (SIP INFO and RTP rfc2833). Check with your provider whether it supports one of these. The default protocol used by Linphone is rfc2833, but if that fails you can set the protocol to SIP INFO in *Preferences* → *Network* → *Other*. If it does not work with either of them, DTMF transmission cannot be done using Linphone.

5.7 Glossary

Find some brief explanation of the most important technical terms and protocols mentioned in this document:

VoIP

VoIP stands for *voice over Internet protocol*. This technology allows the transmission of ordinary telephone calls over the Internet using packet-linked routes. The voice information is sent in discrete packets like any other data transmitted over the Internet via IP.

SIP

SIP stands for *session initiation protocol*. This protocol is used to establish media sessions over networks. In a Linphone context, SIP is the magic that triggers the ring at your counterpart's machine, starts the call, and also terminates it as soon as one of the partners decides to hang up. The actual transmission of voice data is handled by RTP.

RTP

RTP stands for *real-time transport protocol*. It allows the transport of media streams over networks and works over UDP. The data is transmitted by means of discrete packets that are numbered and carry a time stamp to allow correct sequencing and the detection of lost packages.

DTMF

A DTMF encoder, like a regular telephone, uses pairs of tones to represent the various keys. Each key is associated with a unique combination of one high and one low tone. A decoder then translates these touch-tone combinations back into numbers. Linphone supports DTMF signalling to trigger remote actions, such as checking voice mail.

codec

Codecs are algorithms specially designed to compress audio and video data.

jitter

Jitter is the variance of latency (delay) in a connection. Audio devices or connection-oriented systems, like ISDN or PSTN, need a continuous stream of data. To compensate for this, VoIP terminals and gateways implement a jitter buffer that collect the packets before relaying them onto their audio devices or connection-oriented lines (like ISDN). Increasing the size of the jitter buffer decreases the likelihood of data being missed, but the latency of the connection is increased.

5.8 For More Information

For general information about VoIP, check the VoIP Wiki at `http://voip-info.org/tiki-index.php`. For a comprehensive list of providers offering VoIP services in your home country, refer to `http://voip-info.org/wiki-VOIP+Service+Providers+Residential`.

Encryption with KGpg

6

KGpg is an important component of the encryption infrastructure on your system. With the help of this program, generate and manage all needed keys, use its editor function for the quick creation and encryption of files, or use the applet in your panel to encrypt or decrypt by dragging and dropping. Other programs, such as your mail program (Kontact or Evolution), access the key data to process signed or encrypted contents. This chapter covers the basic functions needed for daily work with encrypted files.

6.1 Generating a New Key Pair

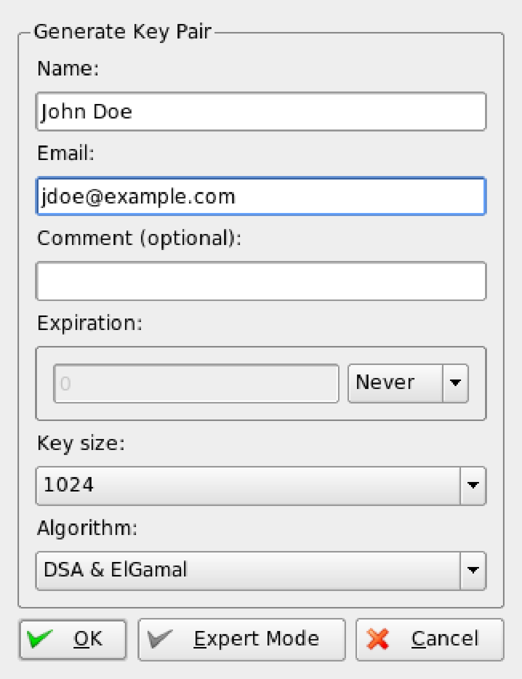
To be able to exchange encrypted messages with other users, first generate your own key pair. One part of it—the *public key*—is distributed to your communication partners, who can use it to encrypt the files or e-mail messages they send. The other part of the key pair—the *private key*—is used to decrypt the encrypted contents.

IMPORTANT: Private Key versus Public Key

The public key is intended for the public and should be distributed to all your communication partners. However, only you should have access to the private key. Do not grant other users access to this data.

Start KGpg from the main menu by selecting *Utilities* → *KGpg* or enter `kgpg` on the command line. When you start the program for the first time, an assistant appears to guide you through the configuration. Follow the instructions up to the point where you are prompted to create a key. Enter a name, an e-mail address, and, optionally, a comment. If you do not like the default settings provided, also set the expiration time for the key, the key size, and the encryption algorithm used. See Figure 6.1, “KGpg: Creating a Key” (page 102).

Figure 6.1 *KGpg: Creating a Key*



Generate Key Pair

Name: John Doe

Email: jdoe@example.com

Comment (optional):

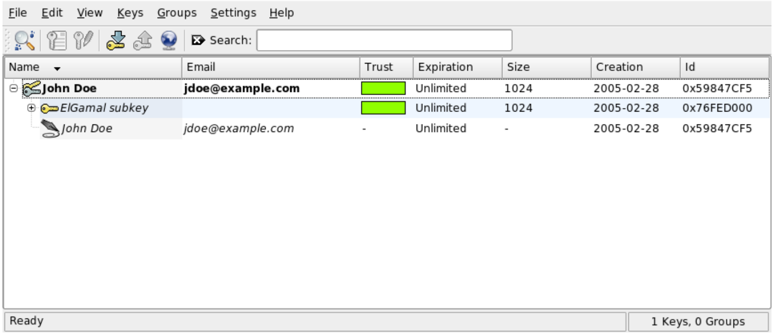
Expiration: 0 Never

Key size: 1024

Algorithm: DSA & ElGamal

Confirm your settings with *OK*. The next dialog prompts you to enter a password twice. The program then generates the key pair and displays a summary. It is a good idea to save or print a revocation certificate right away. Such a certificate will be needed if you forget the password for your private key so need to revoke it. After you confirm with *OK*, KGpg displays its main window. See Figure 6.2, “The Key Manager” (page 103).

Figure 6.2 *The Key Manager*



Name	Email	Trust	Expiration	Size	Creation	Id
John Doe	**jdoe@example.com**	🟩	Unlimited	1024	2005-02-28	0x59847CF5
ElGamal subkey		🟩	Unlimited	1024	2005-02-28	0x76FED000
John Doe	*jdoe@example.com*	-	Unlimited	-	2005-02-28	0x59847CF5

Ready

1 Keys, 0 Groups

6.2 Exporting the Public Key

After generating your key pair, make the public key available to other users. This enables them to use it to encrypt or sign the messages or files they send you. To make the public key available for others, select *Keys* → *Export Public Key(s)*. The dialog that opens offers four options:

Email

Your public key is sent to a recipient of your choice by e-mail. If you activate this option and confirm with *OK*, the dialog for creating a new e-mail message with KMail appears. Enter the recipient and click *Send*. The recipient receives your key and can then send you encrypted contents.

Clipboard

You can place your public key here before continuing to process it.

Default Key Server

To make your public key available to a wide audience, export it to one of the key servers on the Internet. For more information, refer to Section 6.4, “The Key Server Dialog” (page 105).

File

If you prefer to distribute your key as a file on a data medium instead of sending it by e-mail, click this option, confirm or change the file path and name, and click *OK*.

6.3 Importing Keys

If you receive a key in a file (for example, as an e-mail attachment), integrate it in your key ring with *Import Key* and use it for encrypted communication with the sender. The procedure is similar to the procedure for exporting keys already described.

6.3.1 Signing Keys

Keys can be signed like every other file to guarantee their authenticity and integrity. If you are absolutely sure an imported key belongs to the individual specified as the owner, express your trust in the authenticity of the key with your signature.

> **IMPORTANT: Establishing a Web of Trust**
>
> Encrypted communication is only secure to the extent that you can positively associate public keys in circulation with the specified user. By cross-checking and signing these keys, you contribute to the establishment of a web of trust.

Select the key to sign in the key list. Select *Keys* → *Sign Keys*. In the following dialog, designate the private key to use for the signature. An alert reminds you to check the authenticity of this key before signing it. If you have performed this check, click *Continue* and enter the password for the selected private key in the next step. Other users can now check the signature by means of your public key.

6.3.2 Trusting Keys

Normally, you are asked by the corresponding program whether you trust the key (whether you assume it is really used by its authorized owner). This happens each time a message needs to be decrypted or a signature must be checked. To avoid this, edit the trust level of the newly imported key.

Right-click the newly imported key to access a small context menu for key management. Select *Edit Key in Terminal* from it. KGpg opens a text console in which to set the trust level with a few commands.

At the prompt of the text console (`Command >`), enter `trust`. On a scale between `1` (unsure) and `5` (complete trust) make an estimate of how much you trust that the signers of the imported key have checked the true identity of the key owner. Enter the selected value at the prompt (`Your decision?`). If you are really sure about the signers' trustworthiness, enter `5`. Answer the following question by entering `y`. Finally, enter `quit` to exit the console and return to the list of keys. The key now has the trust level `Ultimate`.

The trust level of the keys in your key ring is indicated by a colored bar next to the key name. The lower the trust level is, the less you trust the signer of the key to have checked the true identity of the keys signed. You may be entirely sure about the signer's identity, but he may still be lazy in regard to checking other people's identities before signing their keys. Therefore, you could still trust him and his own key, but assign lower trust levels to the keys of others that have been signed by him. The trust level's purpose is solely one of a reminder. It does not trigger any automatic actions by KGpg.

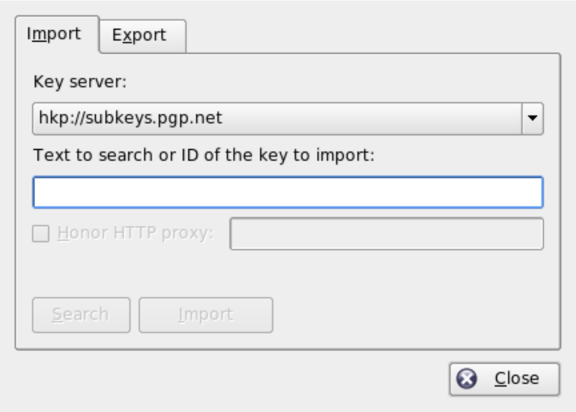
6.4 The Key Server Dialog

Several Internet-based key servers offer the public keys of many users. To engage in encrypted communication with a large number of users, use these servers to distribute your public key. For this purpose, export your public key to one of these servers. Similarly, KGpg enables you to search one of these servers for the keys of certain people and import their public keys from the server. Open the key server dialog with *File → Key Server Dialog*.

6.4.1 Importing a Key from a Key Server

By means of the *Import* tab in the key server dialog, import public keys from one of the Internet-based key servers. Use the drop-down menu to select one of the preconfigured key servers and enter a search string (e-mail address of the communication partner) or the ID of the key to find. When you click *Search*, your system connects to the Internet and searches the specified key server for a key that matches your specifications. Refer to Figure 6.3, “Search Screen for Importing a Key” (page 106).

Figure 6.3 *Search Screen for Importing a Key*



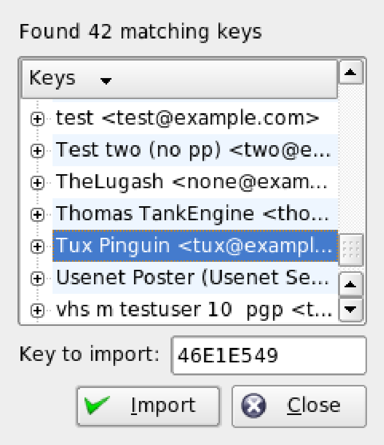
Key server: hkp://subkeys.pgp.net

Text to search or ID of the key to import:

□ Honor HTTP proxy:

If your search on the key server is successful, a list of all retrieved server entries is displayed in a new window. Select the key to include in your key ring and click *Import*. See Figure 6.4, “Hits and Import” (page 106). Confirm the following message with *OK* then exit the key server dialog with *Close*. The imported key then appears in the main overview of the key manager and is ready for use.

Figure 6.4 *Hits and Import*


Found 42 matching keys

Keys

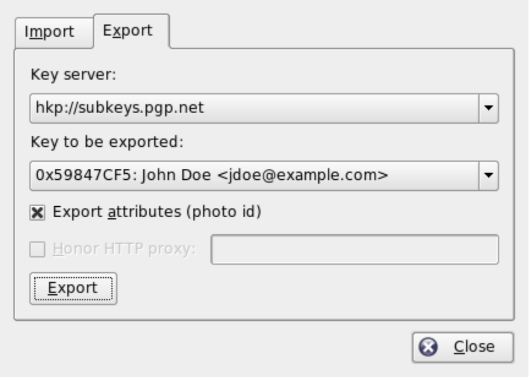
- test <test@example.com>
- Test two (no pp) <two@e...
- TheLugash <none@exam...
- Thomas TankEngine <tho...
- Tux Pinguin <tux@exampl...
- Usenet Poster (Usenet Se...
- vhs m testuser 10 pgp <t...

Key to import: 46E1E549

6.4.2 Exporting Your Keys to a Key Server

To export your key to one of the freely accessible key servers on the Internet, select the *Export* tab in the key server dialog. Designate the target server and the key to export by means of two drop-down menus. Then start the export with *Export*.

Figure 6.5 *Exporting a Key to a Key Server*



Key server:
hkp://subkeys.pgp.net
Key to be exported:
0x59847CF5: John Doe <jdoe@example.com>
☒ Export attributes (photo id)
□ Honor HTTP proxy:

6.5 Text and File Encryption

KGpg also offers the possibility to encrypt text or clipboard contents. Click the padlock icon and find the options *Encrypt clipboard* and *Decrypt clipboard* as well as the option for opening the integrated editor.

6.5.1 Encrypting and Decrypting the Clipboard

Files copied to the clipboard can easily be encrypted with a few clicks. Open the function overview by clicking the KGpg icon. Select *Encrypt Clipboard* and designate the key to use. A status message about the encryption procedure is displayed on the desktop. The encrypted contents can now be processed from the clipboard as needed. The decryption of clipboard contents is just as easy. Simply open the menu on the panel, select *Decrypt Clipboard*, and enter the password associated with your private key. The decrypted version is now available for processing in the clipboard and in the KGpg editor.

6.5.2 Encrypting and Decrypting by Dragging and Dropping

To encrypt or decrypt files, click the icons on the desktop or in the file manager, drag them to the padlock in the panel, and drop them there. If the file is not encrypted, KGpg asks for the key to use. As soon as you select a key, the file is encrypted without any further messages. In the file manager, encrypted files are designated with the suffix `.asc` and the padlock icon. These files can be decrypted by clicking the file icon, dragging it to the KGpg symbol in the panel, and dropping it there. Then select whether the file should be decrypted and saved or displayed in the editor.

6.5.3 The KGpg Editor

Instead of creating contents for encryption in an external editor then encrypting the file with one of the methods described above, you can use the integrated editor of KGpg to create the file. Open the editor (*Open Editor* from the context menu), enter the desired text, and click *Encrypt*. Then select the key to use and complete the encryption procedure. To decrypt files, use *Decrypt* and enter the password associated with the key.

Generating and checking signatures is just as easy as encrypting directly from the editor. Go to *Signature* → *Generate Signature* and select the file to sign from the file dialog. Then designate the private key to use and enter the associated password. KGpg informs about the successful generation of the signature. Files can also be signed from the editor by simply clicking *Sign/Verify*. To check a signed file, go to *Signature* → *Verify Signature* and select the file to check in the following dialog. After you confirm the selection, KGpg checks the signature and reports the result of the operation. Another possibility is to load the signed file into the editor and click *Sign/Verify*.

6.6 For More Information

For theoretical background information about the encryption method, refer to the brief and clear introduction on the GnuPG project pages at `http://www.gnupg.org/documentation/howtos.html.en`. This document also provides a list of further information sources.

Part III Multimedia

Sound in Linux

7

Linux includes a wide range of sound and multimedia applications. Some of these applications are part of one of the main desktop environments. With the applications described here, control the volume and balance of playback, play CDs and music files, and record and compress your own audio data.

7.1 Mixers

Mixers provide a convenient means of controlling the volume and balance of the sound output and input of computers. The main difference between the various mixers is the outer appearance of the user interface. However, there are a number of mixers that are designed for specific hardware. One example is envy24control, a mixer for the Envy 24 sound chip. Another one is hdspmixer, which is for RME Hammerfall cards. From the mixers available, select the one that best suits your needs.

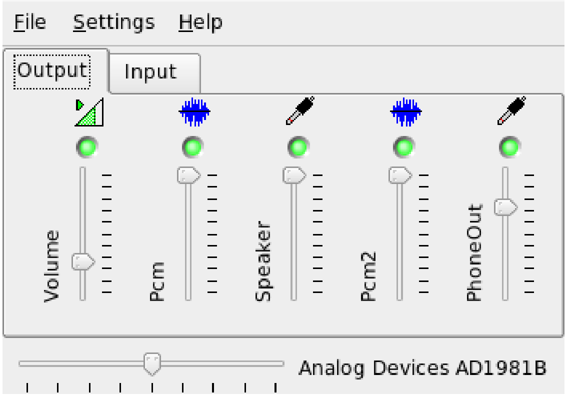
TIP: Test your Mixer

Generally, it is advisable to open a mixer application before opening other sound applications. Use the mixer to test and adjust the control settings for the input and output of the sound card.

7.1.1 The KDE Mixer Applet

KMix is the KDE mixer application. It is integrated into the KDE panel as a small panel applet located in the system tray. Click the panel icon to control the volume of your speakers with a control slider. If you right-click the icon, the context menu of KMix appears. Select *Mute* to switch off the sound output. The panel icon then changes its appearance. Clicking *Mute* again unmutes the volume. To fine-tune your sound settings, select *Show Mixer Window* and configure *Output*, *Input*, and *Switches*. Each of the devices featured there has its own context menu that is opened by a right-clicking the device icon. You can mute or hide each one of them separately.

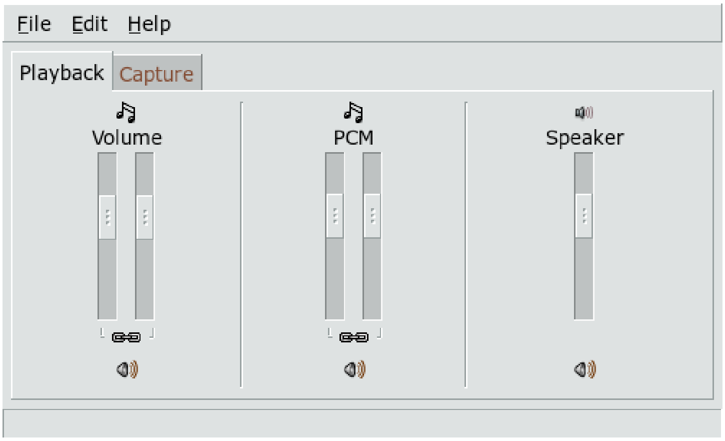
Figure 7.1 *The Mixer KMix*



7.1.2 The GNOME Mixer Applet

GMix, the volume control applet for the GNOME desktop, is integrated into the GNOME panel. Click the panel icon to control the volume of your speakers with a simple control slider. To switch off the sound output, right-click the icon and select *Mute*. The volume control icon then changes its appearance. To unmute the sound output, right-click the icon again and select *Mute* from the menu. Select *Open Volume Control* to access the more advanced mixer features, shown in Figure 7.2, “The GNOME Mixer Applet” (page 113). Each sound device has its own mixer tab.

Figure 7.2 *The GNOME Mixer Applet*



7.1.3 alsamixer

alsamixer can be run from the command line without the X environment, so is entirely controlled via keyboard shortcuts. An alsamixer window always consists of the following elements: a top row holding basic information on card and chip type, the selected view type, and the mixer item then the volume bars below the information area. Use ← and → to scroll left or right if the controls cannot be displayed in one screen. The names of the controls appear below the controls and the currently selected control is colored in red. Toggle between muted and unmuted state of any mixer control using M. A muted control has *MM* written below its name. Any control that has capture (recording) capabilities has a red capture flag.

alsamixer has three different view modes: *Playback*, *Capture*, and *All*. By default, alsamixer is started in *playback* mode, displaying only those mixer controls relevant for playback (Master Volume, PCM, CD, etc.). *Capture* displays only those controls used for recording. *All* displays all controls available. Switch the view modes using F3, F4, and F5.

Select channels with → and ← or N and P. Use ↑ and ↓ or + and - to increase and decrease the volume. Stereo channels can be controlled independently, using Q, W, and E for increasing the volume and Z, X, and C for decreasing the volume. The number keys between 0 and 9 can be used to change the absolute volume quickly. These correspond to zero to ninety percent of full volume.

7.1.4 Look and Feel of Mixer Applications

The look and feel of mixer applications depends on the type of sound card used. Some drivers, like SB Live!, have many controllable (tunable) mixer elements while the drivers for professional sound cards may have elements with totally different names.

On-Board Sound Chip

Most of the PCI on-board sound chips are based on AC97 codec. *Master* controls the main volume from the front speakers. *Surround*, *Center*, and *LFE* control the rear, center, and bass-boost speakers. Each of them has a mute switch. In addition to that, some boards have individual *Headphone* and *Master Mono* volumes. The latter is used for the built-in speaker on some laptops.

PCM controls the internal volume level of digital WAVE playback. PCM is an acronym for Pulse Code Modulation, one of the digital signal formats. This control has also an individual mute switch.

Other volumes, like *CD*, *Line*, *Mic*, and *Aux*, control the loopback volume from the corresponding input to the main output. They do not influence the recording level, only the playback volumes.

For recording, turn on the *Capture* switch. This is the master recording switch. The *Capture* volume is the input gain for recording. By default, this switch is set to zero. Choose a recording source like *Line* or *Mic*. The recording source is exclusive, so you cannot choose two of them at the same time. *Mix* is a special recording source. You can record the currently played signal from this source.

Depending on the AC97 codec chip, special effects, like 3D or bass/treble, are available, too.

SoundBlaster Live! and Audigy Family

SoundBlaster Live! and SB Audigy1 have numerous mixer controls for their AC97 codec chip and DSP engine. In addition to the controls already described, they have *Wave*, *Music*, and *AC97* volumes to control the internal signal routing and attenuation for PCM, WaveTable MIDI, and AC97 mixing. Keep the volume at 100% to hear all of them. SB Audigy2 (depending on the model) has less controls than SB Live, but still has *Wave* and *Music* controls.

The recording on SB Live is similar to on-board chip. You can choose *Wave* and *Music* as the additional recording source to record the played PCM and WaveTable signals.

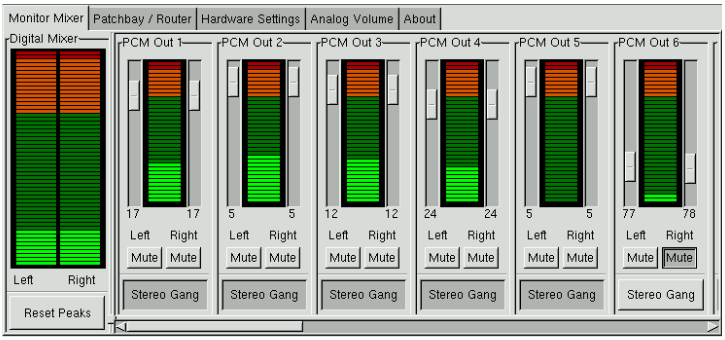
USB Audio Devices

USB audio devices usually have a small number of mixer controls. Sometimes they even have none at all. Most devices either have a *Master* or *PCM* control switch to control the playback volume.

7.1.5 The Mixer for the Sound Chip Envy24

envy24control is a mixer application for sound cards using the Envy24 (ice1712) chip. The flexibility of the Envy24 chip can result in varying functionalities in different sound cards. The latest details on this sound chip are available in `/usr/share/doc/packages/alsa-tools/envy24control`.

Figure 7.3 *Monitor and Digital Mixer of envy24control*



The *Monitor Mixer* of envy24control shows the signal levels that can be mixed digitally in the sound card. The signals designated as *PCM Out* are generated by applications that send PCM data to the sound card. The signals of the analog inputs are shown under *H/W In*. The *S/PDIF* inputs are shown to the right. Set the input and output levels of the analog channels under *Analog Volume*.

Use the *Monitor Mixer* sliders for digital mixing. The respective levels are displayed in the *Digital Mixer*. For each output channel, the *Patchbay* contains a row of radio buttons for selecting the desired channel source.

Adjust the amplification for the analog-to-digital and digital-to-analog converters under *Analog Volume*. Use the *DAC* sliders for the output channels and the *ADC* sliders for the input channels.

The S/PDIF channel settings are made under *Hardware Settings*. The Envy24 chip reacts to volume changes with a delay that can be configured with *Volume Change*.

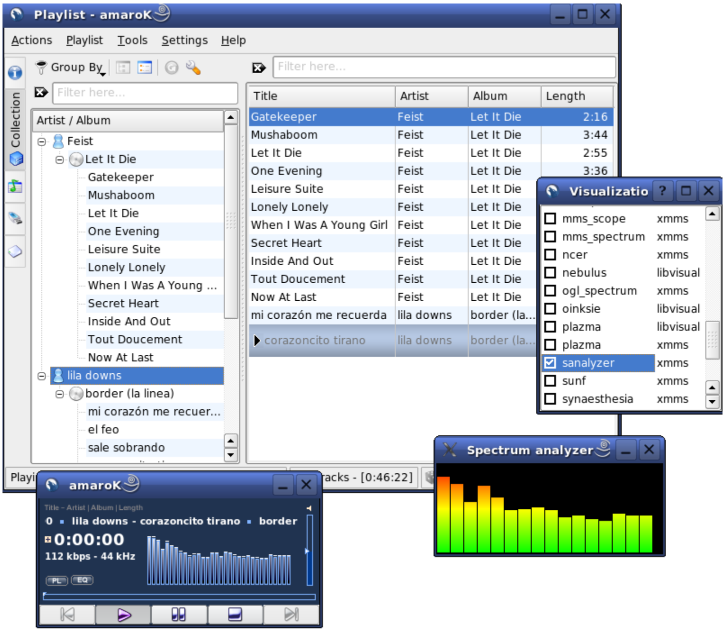
7.2 Multimedia Players

7.2.1 amaroK

The amaroK media player handles various audio formats and plays the streaming audio broadcasts of radio stations on the Internet. The program handles all file types supported by the sound server acting as a back-end—currently aRts or GStreamer.

On first start, amaroK launches a *First-Run Wizard*, which helps set up amaroK. In the first step, configure your preferred look and feel for amaroK. Choose to display player and playlist in separate windows (see Figure 7.4, “The amaroK Media Player” (page 117)) or combine their functionality in one single window. In the second step, determine where amaroK should look for your music collection. amaroK scans these folders for playable media. By default, amaroK is configured to scan the selected folders recursively (to include all their subdirectories in the scan), monitor changes to the content of the selected directories, and import any playlists located there. All the settings made with the wizard can be modified later by starting the wizard again with *Tools* → *First-Run Wizard*.

Figure 7.4 The amaroK Media Player



Artist / Album

- Feist
  - Let It Die
    - Gatekeeper
    - Mushaboom
    - Let It Die
    - One Evening
    - Leisure Suite
    - Lonely Lonely
    - When I Was A Young ...
    - Secret Heart
    - Inside And Out
    - Tout Doucement
    - Now At Last
- lila downs
  - border (la linea)
    - mi corazón me recuer...
    - el feo
    - sale sobrando

Title	Artist	Album	Length
Gatekeeper	Feist	Let It Die	2:16
Mushaboom	Feist	Let It Die	3:44
Let It Die	Feist	Let It Die	2:55
One Evening	Feist	Let It Die	3:36
Leisure Suite	Feist	Let It Die	
Lonely Lonely	Feist	Let It Die	
When I Was A Young Girl	Feist	Let It Die	
Secret Heart	Feist	Let It Die	
Inside And Out	Feist	Let It Die	
Tout Doucement	Feist	Let It Die	
Now At Last	Feist	Let It Die	
mi corazón me recuerda	lila downs	border (la...	
corazoncito tirano	lila downs	border (la...	

Playi... acks - [0:46:22]

□ mms_scope	xmms
□ mms_spectrum	xmms
□ ncer	xmms
□ nebulus	libvisual
□ ogl_spectrum	xmms
□ oinksie	libvisual
□ plazma	libvisual
□ plazma	xmms
☑ sanalyzer	xmms
□ sunf	xmms
□ synaesthesia	xmms

Title - Artist | Album | Length
0 ▪ lila downs - corazoncito tirano ▪ border
0:00:00
112 kbps - 44 kHz

Managing Playlists

On start-up, amaroK scans the file system for multimedia files according to the settings made in the wizard. The right part of the playlist window lists any playlists found. Play titles listed in it in the order of your choice. If no playlist is found, create one. The best way to do this is by using the sidebar to the left of the window. To the far left, there are a number of tabs that can be used to open different views. From each of these views, drag individual titles or entire directories and drop them into the playlist to include them in the list. The following is a description of the function of each tab.

Context

With this tab, view information about your collection and the current artist. For example, the view informs you about your favorite titles, the newest titles added to the collection, and other details. The *Home* view provides statistics on your listening habits, listing your favorite, newest, and least-played tracks. *Current Track* provides data related to the track currently being played, such as the album cover (see Section “The Cover Manager” (page 119)), the listening statistics related to this track, and much more. If you are interested in the lyrics of the track, display them using the *Lyrics* tab.

Collection Browser

Use this view to manage and display your personal collection of titles. The collection view may include files from different locations. The wrench icon in the toolbar lets you determine what locations should be scanned for music files. Once you select the directories, the scan starts automatically. The result is displayed as a tree structure. Using *Primary* and *Secondary*, organize the two top branches of the tree according to the criteria *Album*, *Artist*, *Genre*, and *Year*. Once the tree view is ready, find titles simply by typing them into the input field. The selection in the tree view jumps to the first matching entry automatically as you type. To update your collection data, initiate a rescan of the file system using *Tools → Rescan Collection*.

Playlist Browser

The playlist browser is divided into two parts. The upper part lists all your custom playlists created by dragging tracks into the playlist window and clicking *Save Playlist As*. View the contents of them by clicking the + next to the playlist's name. Modify these playlists using drag and drop. To load one of them, double-click the playlist.

IMPORTANT: Sharing Playlists with Other Players

Save playlists in `m3u` or `pls` format, so you can share them with any other players using these formats.

amaroK can compile useful playlists (“Smart Playlists”) on the fly. Use the bottom part of the playlist browser to select one of the smart playlists or click *Create Smart Playlist* to define a custom smart playlist. Enter a name, search criteria, order, and optional track limit.

File Browser

This tab opens a file browser. It corresponds to the standard KDE file selector dialog with the usual controls for navigating the file system. Enter a URL or directory directly into the text input field. From the contents displayed, drag elements to the playlist to include them. You can also perform a recursive search for a file in a given directory. To do so, enter a text string for the title and the location at which to start the search. Then select *Search* and wait for the results to appear in the lower section of the window.

The Cover Manager

amaroK features a cover manager to enable you to keep matching music and image data on the albums you play. Start the *Cover Manager* with *Tools → Cover Manager*. A tree view in the left part of the window lists all the albums of your collection. The covers retrieved from Amazon are displayed in the right part of the window. With *View*, choose what is displayed in the cover list view. *All albums* lists all albums of your collection, regardless of whether they have a cover image. *Albums with cover* lists only those with a cover and *Albums without cover* lists those lacking a cover. To retrieve cover data, choose your *Amazon Locale* then click *Fetch Missing Covers*. amaroK then tries to get covers for all albums contained in your collection.

Effects

Select the *FX* button in the player window or use the amaroK application menu to open a dialog in which to enable and configure several sound effects, such as an equalizer, the stereo balance, and a hall effect. Select the desired effects and adjust the settings, if available, for each of them.

Visualizations

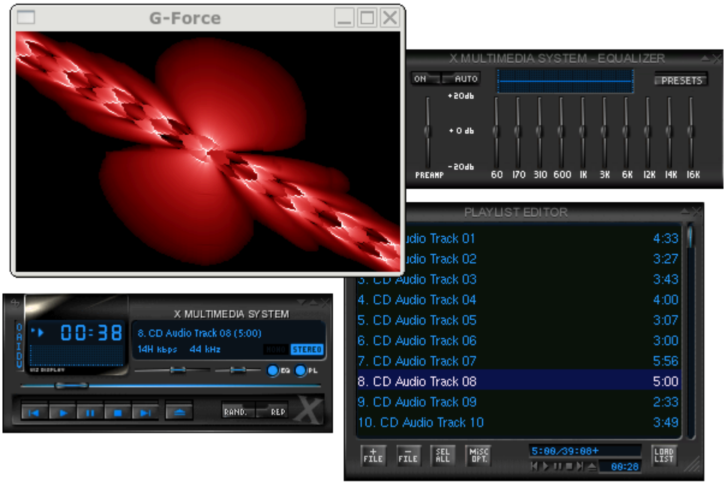
amaroK comes with a number of visualizations that display a graphical effect for the music played. Native amaroK visualizations are displayed in the player window. Cycle through the various available display modes by clicking the animation.

In addition to the above, amaroK also supports the visualization plug-ins of the XMMS media player. To use these, first install the `xmms-plugins` package then select *Visualizations* from the amaroK menu. This opens a window listing the available plug-ins. XMMS plug-ins are always displayed in an extra window. In some cases, there is an option to display them in fullscreen mode. For some plug-ins, you may not get a smooth visual effect unless you use a 3D-accelerated graphics card.

7.2.2 XMMS

XMMS is another full-featured media player with robust audio support, so that pops or breaks during playback should be very rare. The application is easy to use. The button for displaying the menu is located in the upper left corner of the program window. For those preferring a GNOME-like look and feel, there is a GTK2 version of XMMS available, the Beep Media Player. Just install the package `bmp`. However, not all XMMS plug-ins are supported by this port of XMMS.

Figure 7.5 *XMMS with Equalizer, OpenGL Spectrum Analyzer, and Infinity Plug-Ins*



udio Track 01 4:33
udio Track 02 3:27
3. CD Audio Track 03 3:43
4. CD Audio Track 04 4:00
5. CD Audio Track 05 3:07
6. CD Audio Track 06 3:00
7. CD Audio Track 07 5:56
8. CD Audio Track 08 5:00
9. CD Audio Track 09 2:33
10. CD Audio Track 10 3:49

Select the output plug-in module with *Options* → *Preferences* → *Audio I/O Plugins*. If the `xmms-kde` package is installed, the aRts sound server can be configured here.

IMPORTANT: Using the Disk Writer Plug-In

XMMS automatically redirects its output to the *Disk Writer Plugin* if it is not able to find a configured sound card. In this case, the played files are written to the hard disk as WAV files. The time display then runs faster than when playing the output through a sound card.

Start various visualization plug-ins with *Options* → *Preferences* → *Visualization Plugins*. If you have a graphics card with 3D acceleration, select an application such as the OpenGL spectrum analyzer. If the `xmms-plugins` package is installed, try the Infinity plug-in.

To the left under the menu button, there are five buttons with different letters on them. These buttons allow quick access to additional menus, dialog, and configurations. Open the playlist with *PL* and the equalizer with *EQ*.

7.3 CDs: Playback and Ripping

There are many ways to listen to your favorite music tracks. Either play a CD or play digitalized versions of them. The following section features some CD player applications as well as some applications that can be used for the ripping and encoding of audio CDs.

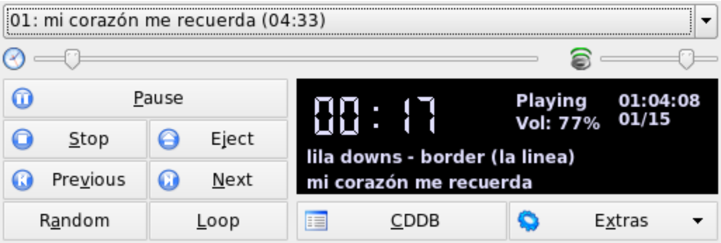
IMPORTANT: CDDA and Analog CD Playback

There are two different ways of playing back audio CDs. CD and DVD drives capable of analog CD playback read out the audio data and send it to the sound output device. Some external drives connected via PCMCIA, FireWire, or USB need to use CDDA (Compact Disk Digital Audio) to extract the audio data first then play it as digital PCM. The players featured in the following sections do not support CDDA. Use XMMS if you need CDDA support.

7.3.1 KsCD—Audio CD Player

KsCD is an easy-to-use audio CD player. It integrates into the KDE taskbar and can be configured to start playing automatically after a CD has been inserted. To access the configuration menu, select *Extras* → *Configure KsCD*. Fetch album and track information from a CDDB server on the Internet if KsCD is configured accordingly. You can also upload CDDB information to share it with others. Use the *CDDB* dialog for information retrieval and upload.

Figure 7.6 *The KsCD User Interface*



01: mi corazón me recuerda (04:33)

00 : 17
Playing 01:04:08
Vol: 77% 01/15
lila downs - border (la linea)
mi corazón me recuerda

7.3.2 GNOME CD Player Applet

This is a simple applet that integrates into a GNOME panel. Using the tools icon, configure its behavior and select a theme. Control the playback with the buttons at the bottom of the player window or using the context menu opened by right-clicking the panel icon or player window.

7.3.3 Compressing Audio Data

Audio compression can be handled by various tools. The following sections feature a command-line approach to encoding and playing audio data as well as some graphical applications capable of audio compression.

Command Line Tools for Encoding and Playback of Audio Data

Ogg Vorbis (package `vorbis-tools`) is a free audio compression format that is now supported by the majority of audio players and even portable MP3 players. The Web page of the project is `http://www.xiph.org/ogg/vorbis`.

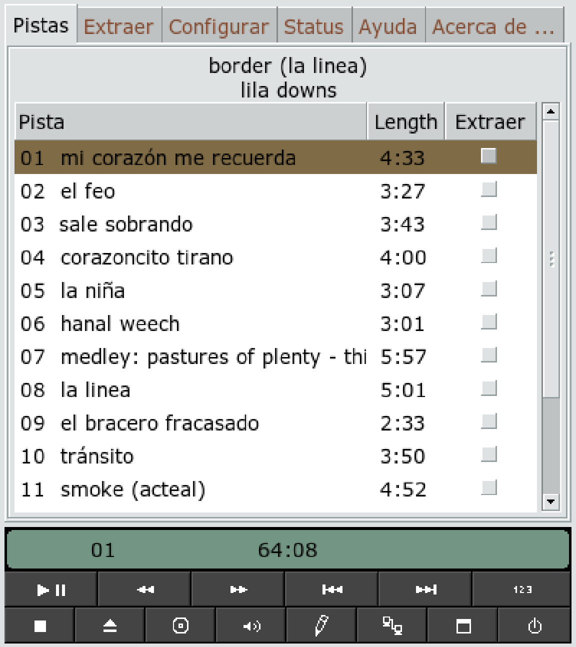
SUSE Linux comes with several tools supporting Ogg Vorbis. `oggenc` is a command line tool used for encoding WAV files to Ogg. Just run `oggenc` *`myfile.wav`* to transform a given `.wav` file into Ogg Vorbis. The option `-h` displays an overview of the other parameters. Oggenc supports encoding with a variable bit rate. In this way, an even higher degree of compression can be achieved. Instead of the bit rate, specify the desired quality with the parameter `-q`; `-b` determines the average bit rate; `-m` and `-M` specify the minimum and maximum bit rate.

ogg123 is a command-line Ogg player. Start it with a command like `ogg123` *`mysong.ogg`*.

Compressing Audio Data Using Grip

Grip is a GNOME CD player and ripper (see Figure 7.7, “Ripping Audio CDs with Grip” (page 123)). The CD player functionality is entirely controlled by the buttons in the bottom part of the window. Control the ripping and encoding functionality using the tabs at the top of the window. To view and edit the track and album information or to select the tracks to rip, open the *Tracks* tab. Select a track by clicking the check box next to the track title. To edit the track information, click *Toggle disc editor* and submit your modifications. The *Rip* tab selects the preferred rip mode and controls the ripping process. Access the entire Grip configuration under the *Config* tab. Use *Status* to check the status of the application.

Figure 7.7 *Ripping Audio CDs with Grip*



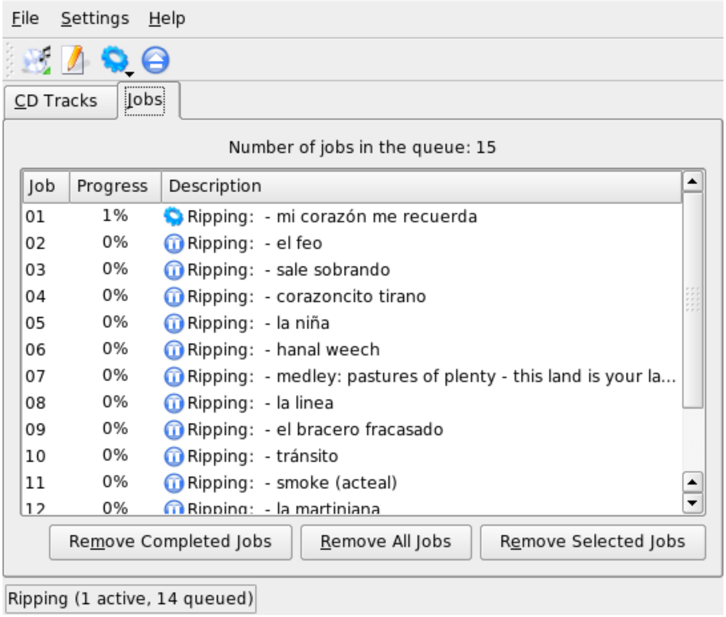
border (la linea)
lila downs

Pista	Length	Extraer
01 mi corazón me recuerda	4:33	□
02 el feo	3:27	□
03 sale sobrando	3:43	□
04 corazoncito tirano	4:00	□
05 la niña	3:07	□
06 hanal weech	3:01	□
07 medley: pastures of plenty - thi	5:57	□
08 la linea	5:01	□
09 el bracero fracasado	2:33	□
10 tránsito	3:50	□
11 smoke (acteal)	4:52	□

Compressing Audio Data Using KAudioCreator

KAudioCreator is a lean CD ripper application (see Figure 7.8, “Ripping Audio CDs with KAudioCreator” (page 124)). Once started, it lists all the tracks of your CD in the *CD Tracks* tab. Select the tracks to rip and encode. To edit the track information, use the *Album Editor* under *File* → *Edit Album*. Otherwise just start the ripping and encoding with *File* → *Rip Selection*. Watch the progress of these jobs using the *Jobs* tab. If configured accordingly, KAudioCreator also generates playlist files for your selection that can be used by players, like amaroK or XMMS.

Figure 7.8 *Ripping Audio CDs with KAudioCreator*



Number of jobs in the queue: 15

Job	Progress	Description
01	1%	Ripping: - mi corazón me recuerda
02	0%	Ripping: - el feo
03	0%	Ripping: - sale sobrando
04	0%	Ripping: - corazoncito tirano
05	0%	Ripping: - la niña
06	0%	Ripping: - hanal weech
07	0%	Ripping: - medley: pastures of plenty - this land is your la...
08	0%	Ripping: - la linea
09	0%	Ripping: - el bracero fracasado
10	0%	Ripping: - tránsito
11	0%	Ripping: - smoke (acteal)
12	0%	Ripping: - la martiniana

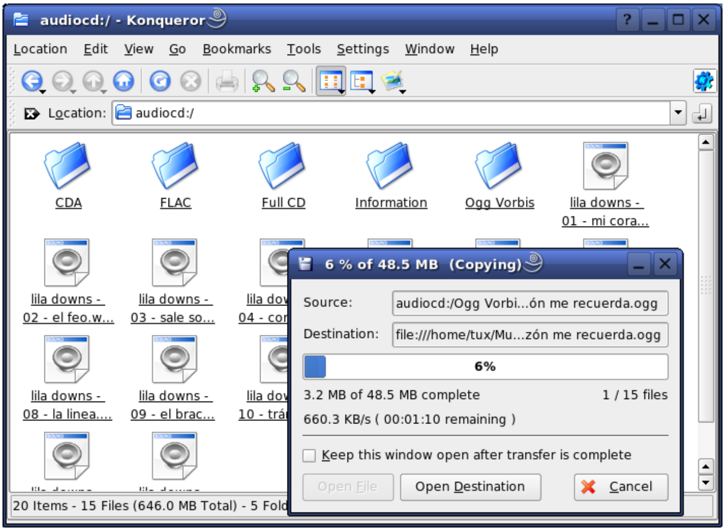
Ripping (1 active, 14 queued)

Compressing Audio CDs Using Konqueror

Before you start the actual ripping process with Konqueror, configure the handling of audio CDs and the Ogg Vorbis encoder in the KDE Control Center. Select *Sound & Multimedia* → *Audio CDs*. The configuration module is divided into three tabs: *General*, *Names*, and *Ogg Vorbis Encoder*. Normally, a suitable CD device is detected automatically. Do not change this default setting unless the autodetection failed and you need to set the CD device manually. Error correction and encoder priority can also be set here. The tab *Ogg Vorbis Encoder* determines the quality of the encoding. To configure online lookup of album, track, and artist information for your ripped audio data, select *Add Track Information*.

Start the ripping process by inserting the CD into the CD-ROM drive and enter `audiocd:/` at the *Location* bar. Konqueror then lists the tracks of the CD and some folders (see Figure 7.9, "Ripping Audio Data with Konqueror" (page 125)).

Figure 7.9 *Ripping Audio Data with Konqueror*


audiocd:/ - Konqueror
Location Edit View Go Bookmarks Tools Settings Window Help
Location: audiocd:/
CDA
FLAC
Full CD
Information
Ogg Vorbis
lila downs - 01 - mi cora...
lila downs - 02 - el feo.w...
lila downs - 03 - sale so...
lila dow 04 - cor
lila downs - 08 - la linea....
lila downs - 09 - el brac...
lila dow 10 - trá
6 % of 48.5 MB (Copying)
Open File
Open Destination
Cancel

Source: audiocd:/Ogg Vorbi...ón me recuerda.ogg

Destination: file:///home/tux/Mu...zón me recuerda.ogg

6%

3.2 MB of 48.5 MB complete 1 / 15 files

660.3 KB/s (00:01:10 remaining)

□ Keep this window open after transfer is complete

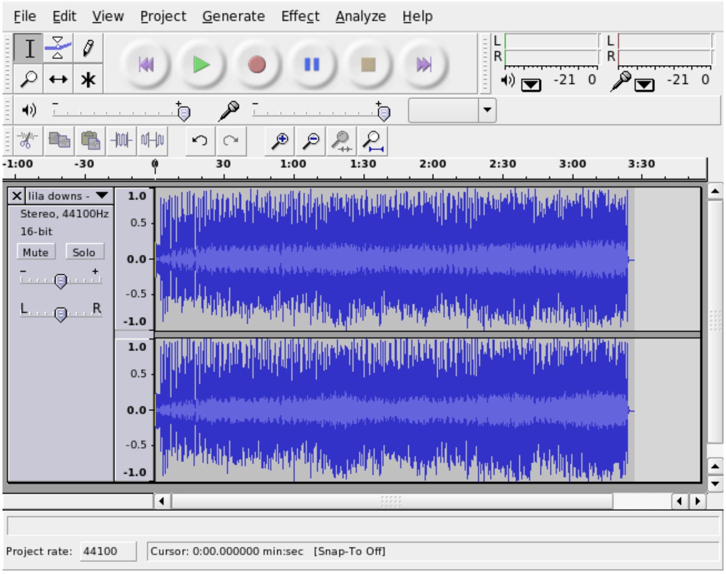
20 Items - 15 Files (646.0 MB Total) - 5 Fold

To keep uncompressed audio data on your disk, just select the `.wav` files and drag them into another Konqueror window to copy them over to their final destination. To start the Ogg Vorbis encoding, drag the `Ogg Vorbis` folder to another Konqueror window. The encoding starts as soon as you drop the Ogg Vorbis folder to its destination.

7.4 Hard Disk Recording with Audacity

With audacity (package `audacity`), record and edit audio files. This is called hard disk recording. When you start the program for the first time, select a language. At other times, change the language setting under *File* → *Preferences* → *Interface*. The language change is then effective the next time you start the program.

Figure 7.10 *Spectral View of the Audio Data*



Stereo, 44100Hz
16-bit

Project rate: 44100

Cursor: 0:00.000000 min:sec [Snap-To Off]

7.4.1 Recording WAV Files and Importing Files

Click the red recording button to create an empty stereo track and start the recording. To change the standard parameters, make the desired settings under *File* → *Preferences*. *Audio I/O* and *Quality* are important for the recording. Even if tracks already exist, pressing the recording button creates new tracks. Initially, this may be confusing, because these tracks cannot be seen in the standard-size program window.

To import audio files, select *Project* → *Import Audio*. The program supports the WAV format and the compressed Ogg Vorbis format. See Section 7.3.3, “Compressing Audio Data” (page 122) for more information about this format.

7.4.2 Editing Audio Files

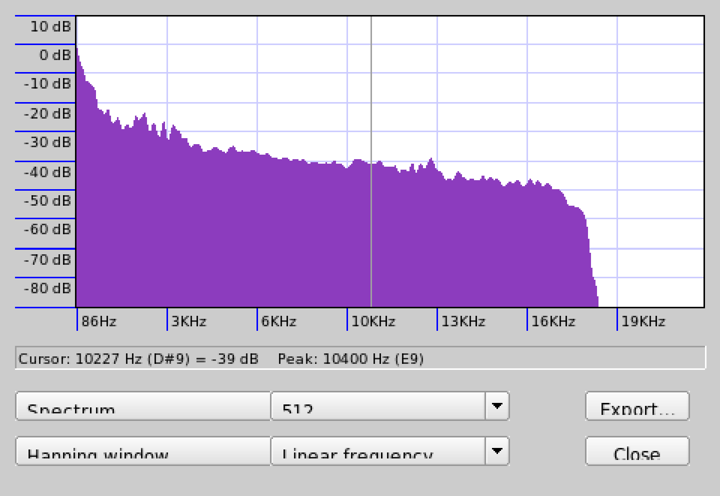
Open the *AudioTrack* menu to the left of the track. This menu offers various options for different views and basic editing operations. To rename the track, select *Name* and enter a new name. The different view modes offered by Audacity include *Waveform*, *Waveform (dB)*, *Spectrum*, and *Pitch*. Choose the one suiting your needs. If you want to edit each channel of a stereo track separately, select *Split Track*. Each channel can then be treated as a separate track. Set *Sample Format* (in bit) and *Sample Rate* (in Hz) for each track.

Before you can use most of the tools offered in the *Edit* menu, first select the channel and the segment of the track to edit. After making your selection, you can apply all kinds of modifications and effects to it.

Depending on the chosen file type, various view formats for segment selections are offered under *View* → *Set Selection Format*. With *Set Snap-To Mode*, the segment boundaries can automatically be adapted to the selected view format. For example, if you select *PAL frames* as the view format and activate *Snap-To*, the segment boundaries are always selected in multiples of frames.

All editing tools come with tool tips, so should be easy to use. The *Undo History* function, accessed with *View* → *History*, is a useful feature for viewing recent editing steps and undoing them by clicking in the list. Use *Discard* with caution, because it deletes editing steps from the list. Once discarded, these steps can no longer be undone.

Figure 7.11 *The Spectrum*



Cursor: 10227 Hz (D#9) = -39 dB   Peak: 10400 Hz (E9)

The built-in spectrum analyzer assists in quickly tracking down any noises. View the spectrum of the selected segment with *View* → *Plot Spectrum*. Select a logarithmic frequency scale in octaves with *Log frequency*. If you move the mouse pointer within the spectrum, the frequencies of the peaks are automatically displayed together with the respective notes.

Remove unwanted frequencies with *Effect* → *FFT Filter*. In connection with the filtering process, it may be necessary to readjust the signal amplitude with *Amplify*. Additionally, use *Amplify* to check the amplitude. By default, the *New Peak Amplitude* is set to 0.0 dB. This value represents the highest possible amplitude in the selected audio format. *Amplification* shows the value needed to amplify the selected segment to this peak amplitude. A negative value indicates overamplification.

7.4.3 Saving and Exporting

To save the entire project, select *File* → *Save Project* or *Save Project As*. This generates an XML file with the extension `.aup`, which describes the project. The actual audio data is saved in a directory named after the project with `_data` appended.

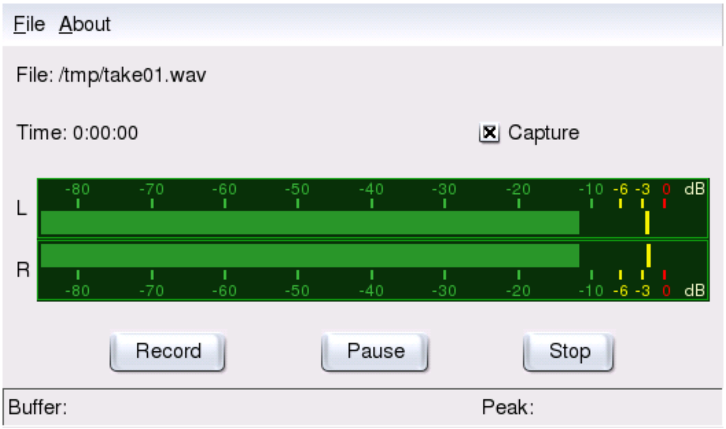
The entire project or the currently selected segment can also be exported as a stereo WAV file. To export the project in Ogg Vorbis format, refer to the information in Section 7.3.3, "Compressing Audio Data" (page 122).

7.5 Direct Recording and Playback of WAV Files

`arecord` and `aplay` from the `alsa` package provide a simple and flexible interface to the PCM devices. `arecord` and `aplay` can be used to record and play audio data in the WAV format and other formats. The command `arecord -d 10 -f cd -t wav mysong.wav` records a WAV file of 10 seconds in CD quality (16 bit, 44.1 kHz). List all options of `arecord` and `aplay` by running them with the `--help` option.

qaRecord (package `kalsatools`) is a simple recording program with a graphical interface and level display. Because this program uses an internal buffer of about 1 MB (configurable with `--buffersize`), it enables uninterrupted recordings even on slow hardware, especially if it is run with real-time priority. During the recording, the currently-used buffer size is displayed in the status line under *Buffer* and the maximum buffer size required so far for this recording is displayed under *Peak*.

Figure 7.12 *QARecord—A Simple Hard Disk Recording Application*


File About
L
R
-80 -70 -60 -50 -40 -30 -20 -10 -6 -3 0 dB
-80 -70 -60 -50 -40 -30 -20 -10 -6 -3 0 dB
Record
Pause
Stop

File: /tmp/take01.wav

Time: 0:00:00

☒ Capture

Buffer:

Peak:

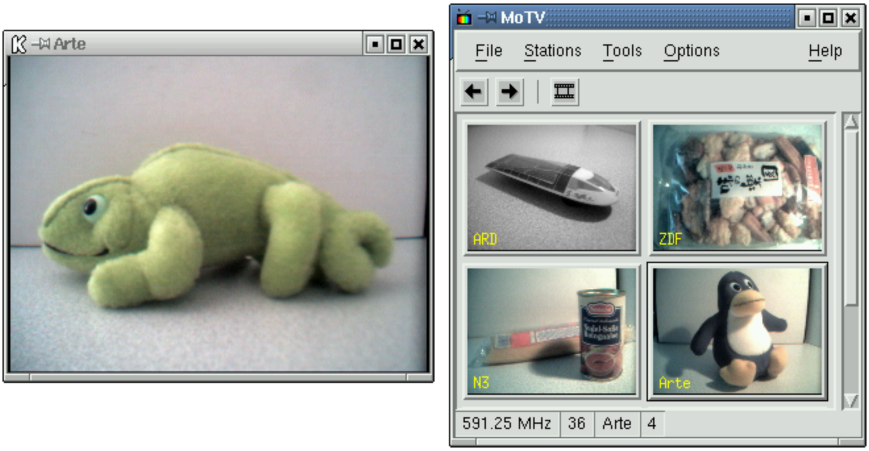
8 TV, Video, Radio, and Webcam

This chapter introduces some basic Linux video, radio, and webcam applications. Learn how to configure and use motv for watching analog TV, using a webcam, and browsing video text. Use xawtv4 for digital video broadcasts. Webcams can be run using gqcam. EPG information can be accessed using nxtvepg or xawtv4.

8.1 Watching TV with motv

motv is an improved successor to xawtv. It incorporates all essential functions into the user interface. Start the application with *Multimedia → Video → motv*. Start it at the command line with `motv`. Initially, only a TV window appears after the application starts. Open a menu window by right-clicking it.

Figure 8.1 *The TV Application motv*



8.1.1 Video Source and Network Search

In *Settings* → *Input*, select the video source. If you select *Television* here, set up the broadcasting network before starting the application. This automatically takes place with the network search, also found under the *Settings* menu. If you click *Save settings*, the network found is entered into the `.xawtv` file in your home directory and will be available the next time you start the application.

TIP: Selecting Channels

If you do not want to browse for all available channels, find the next channel with Ctrl + ↑. If needed, subsequently adjust the broadcast frequency with ← or →.

8.1.2 Retrieving Audio Data

The audio output of the TV card is connected to the line input of your sound card, to the speakers, or to an amplifier. Some TV cards can change the volume of the audio output. The volume can then be set with the sliders that appear after selecting *Settings* → *Slider*. This window also provides the sliders for brightness, contrast, and color.

To use your sound card for audio playback, check the mixer settings using gamix, described in Section 7.1, “Mixers” (page 111). For sound cards meeting the AC97 specifications, set *Input-MUX* to *Line*. The volume can then be adjusted with the *Master* and *Line* sliders.

8.1.3 Screen Proportions and Full-Screen Mode

Most television images have a height and width ratio of 4:3. These proportions can be set with *Tools* → *Screen Dimensions*. If *4:3* is selected here (this is the default setting), the screen dimensions are retained automatically, even when the display size is changed.

With F or *Tools* → *Fullscreen*, switch to full-screen mode. If the TV image in full-screen mode is not scaled to the full monitor size, some fine-tuning is required. Many graphics cards can scale the full-screen mode television image to the full monitor size without changing the graphical mode. If your card does not support this function, the graphics mode must be switched to 640x480 for the full-screen mode. Create the related configuration in *Settings* → *Configuration*. After restarting motv, the monitor mode is also changed if you have switched to full-screen mode.

TIP: Storing the Configuration in .xawtv

The `.xawtv` file is created automatically and updated by clicking *Settings* → *Save settings*. Here, the broadcasters are saved along with the configuration. More information about the configuration file can be found in the man page for `xawtvrc`.

8.1.4 The Launcher Menu

Use the launcher menu to start other applications to use with motv. Start the audio mixer gamix and the video text application alevt, for example, using a keyboard shortcut. Applications to launch from motv must be entered in the `.xawtv` file. The entries should look like this:

```
[launch] Gamix = Ctrl+G, gamix AleVT = Ctrl+A, alevt
```

The shortcut then the command used to start the application should follow the application name itself. Start the applications entered under [launch] via the *Tool* menu.

8.2 Video Text Support

Use alevt to browse video text pages. Start the application with *Multimedia* → *Video* → *alevt* or at the command line with `alevt`.

The application saves all the pages of the selected station just activated with motv. Browse pages by entering the desired page number or by clicking a page number. Move forward or backward through the pages by clicking << or >>, located in the lower window margin.

Recent versions of motv and its successor xawtv4 include their own video text viewer applications: mtt (motv) and mtt4 (xawtv4). mtt4 even supports DVB cards.

8.3 Webcams and motv

If your webcam is already supported by Linux, access it with motv. Find a summary of the supported USB devices at `http://www.linux-usb.org`. If you have already used motv to access the TV card prior to accessing the webcam, the bttv driver is loaded. The webcam driver is loaded automatically when your webcam is connected to the USB. Start motv at the command line with the parameter `-c /dev/video1` to access the webcam. Access the TV card with `motv -c /dev/video0`.

When connecting the webcam to the USB before the bttv driver has been automatically loaded (for example, by starting a TV application), `/dev/video0` is reserved for the webcam. In this case, if you start motv with the `-c /dev/video1` parameter to access the TV card, you might get an error message, because the bttv driver was not automatically loaded. Solve this problem by loading the driver separately with `modprobe bttv` as the user `root`. Access an overview of the configurable video devices on your system with `motv -hwscan`.

8.4 nxtvepg—The TV Magazine for Your PC

From some broadcasters, an EPG signal (Electronic Program Guide) is transmitted along with the video text signal. Easily view this electronic guide using the program nxtvepg. To do this, however, you must have a TV card supported by the bttv driver and be able to receive one of the channels broadcast with an EPG.

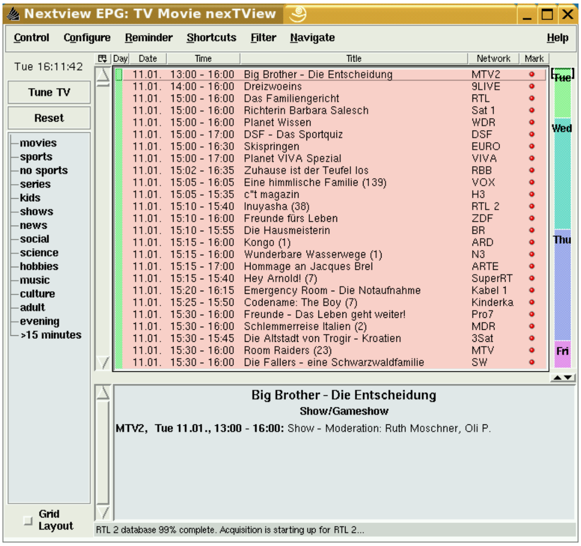
With nxtvepg, the broadcasts are sorted according to channel and topic, such as *movie* and *sport*, and filtered according to criteria, such as *Live*, *Stereo*, or *Subtitle*. Start the application with *Multimedia* → *Video* → *nxtvepg* or at the command line with `nxtvepg`.

8.4.1 Importing the EPG Database

To set up and update the program database via the EPG signal, set the tuner of your TV card to a station that broadcasts EPG. This can be done using a TV application, such as motv or nxtvepg. Only one application at a time can access the tuner.

If you set an EPG broadcaster in motv, nxtvepg immediately begins importing the current list of TV programs. The progress is displayed.

Figure 8.2 *The Electronic TV Magazine nxtvepg*



Day	Date	Time	Title	Network	Mark
	11.01.	13:00 - 16:00	Big Brother - Die Entscheidung	MTV2	●
	11.01.	14:00 - 16:00	Dreizwoeins	9LIVE	●
	11.01.	15:00 - 16:00	Das Familiengericht	RTL	●
	11.01.	15:00 - 16:00	Richterin Barbara Salesch	Sat 1	●
	11.01.	15:00 - 16:00	Planet Wissen	WDR	●
	11.01.	15:00 - 17:00	DSF - Das Sportquiz	DSF	●
	11.01.	15:00 - 16:30	Skispringen	EURO	●
	11.01.	15:00 - 17:00	Planet VIVA Spezial	VIVA	●
	11.01.	15:02 - 16:35	Zuhause ist der Teufel los	RBB	●
	11.01.	15:05 - 16:05	Eine himmlische Familie (139)	VOX	●
	11.01.	15:05 - 15:35	c"t magazin	H3	●
	11.01.	15:10 - 15:40	Inuyasha (38)	RTL 2	●
	11.01.	15:10 - 16:00	Freunde fürs Leben	ZDF	●
	11.01.	15:10 - 15:55	Die Hausmeisterin	BR	●
	11.01.	15:15 - 16:00	Kongo (1)	ARD	●
	11.01.	15:15 - 16:00	Wunderbare Wasserwege (1)	N3	●
	11.01.	15:15 - 17:00	Hommage an Jacques Brel	ARTE	●
	11.01.	15:15 - 15:40	Hey Arnold! (7)	SuperRT	●
	11.01.	15:20 - 16:15	Emergency Room - Die Notaufnahme	Kabel 1	●
	11.01.	15:25 - 15:50	Codename: The Boy (7)	Kinderka	●
	11.01.	15:30 - 16:00	Freunde - Das Leben geht weiter!	Pro7	●
	11.01.	15:30 - 16:00	Schlemmerreise Italien (2)	MDR	●
	11.01.	15:30 - 15:45	Die Altstadt von Trogir - Kroatien	3Sat	●
	11.01.	15:30 - 16:00	Room Raiders (23)	MTV	●
	11.01.	15:30 - 16:00	Die Fallers - eine Schwarzwaldfamilie	SW	●

Big Brother - Die Entscheidung

Show/Gameshow

MTV2, Tue 11.01., 13:00 - 16:00: Show - Moderation: Ruth Moschner, Oli P.

RTL 2 database 99% complete. Acquisition is starting up for RTL 2...

If you have not started a TV application, let nxtvepg search for EPG broadcasters. To do this, use *Configure* → *Provider scan*. *Use .xatv* is activated by default. This indicates that nxtvepg is accessing the broadcasters saved in this file.

TIP: Troubleshooting

If there are problems, check to see if the proper video source has been chosen under *TV card input*.

Select from the EPG providers found in *Configure* → *Select Provider*. *Configure* → *Merge Providers* even creates flexible associations between the various provider databases.

8.4.2 Sorting the Programs

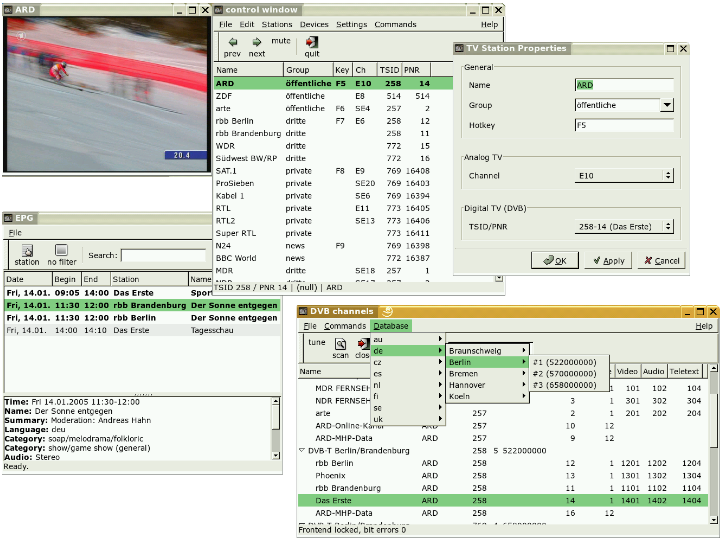
nxtvepg provides a convenient filter function for managing even the most extensive program offerings. Activate a network selection list with *Configure* → *Show networks*. The *Filter* menu offers plenty of filter functions. Right-click the program list to open a special filter menu in which to activate contextual filter functions.

Of particular interest is the *Navigate* menu. This is built directly from the EPG data. It appears in the language provided by the network.

8.5 Watching Digital Video Broadcasts with xawtv4

As your hardware has been properly configured with YaST, start xawtv4 from the main menu (*Multimedia* → *Video* → *xawtv4*). Before you can actually start watching your favorite broadcasts, build a database of DVB stations.

Figure 8.3 *Running xawtv4*



Name	Group	Key	Ch	TSID	PNR
ARD	öffentliche	F5	E10	258	14
ZDF	öffentliche		E8	514	514
arte	öffentliche	F6	SE4	257	2
rbb Berlin	dritte	F7	E6	258	12
rbb Brandenburg	dritte			258	11
WDR	dritte			772	15
Südwest BW/RP	dritte			772	16
SAT.1	private	F8	E9	769	16408
ProSieben	private		SE20	769	16403
Kabel 1	private		SE6	769	16394
RTL	private		E11	773	16405
RTL2	private		SE13	773	16406
Super RTL	private			773	16411
N24	news	F9		769	16398
BBC World	news			772	16387
MDR	dritte		SE18	257	1

TSID 258 / PNR 14 | (null) | ARD

General

Name: ARD

Group: öffentliche

Hotkey: F5

Analog TV

Channel: E10

Digital TV (DVB)

TSID/PNR: 258-14 (Das Erste)

Date	Begin	End	Station	Name
Fri, 14.01.	09:05	14:00	Das Erste	Sport
Fri, 14.01.	11:30	12:00	rbb Brandenburg	Der Sonne entgegen
Fri, 14.01.	11:30	12:00	rbb Berlin	Der Sonne entgegen
Fri, 14.01.	14:00	14:10	Das Erste	Tagesschau

Time: Fri 14.01.2005 11:30-12:00
Name: Der Sonne entgegen
Summary: Moderation: Andreas Hahn
Language: deu
Category: soap/melodrama/folkloric
Category: show/game show (general)
Audio: Stereo
Ready.

Name						Video	Audio	Teletext
MDR FERNSEH					1	101	102	104
NDR FERNSEH				3	1	301	302	304
arte		257		2	1	201	202	204
ARD-Online-Kanal	ARD	257		10	12			
ARD-MHP-Data	ARD	257		9	12			
DVB-T Berlin/Brandenburg		258	5 522000000					
rbb Berlin	ARD	258		12	1	1201	1202	1204
Phoenix	ARD	258		13	1	1301	1302	1304
rbb Brandenburg	ARD	258		11	1	1101	1102	1104
Das Erste	ARD	258		14	1	1401	1402	1404
ARD-MHP-Data	ARD	258		16	12			

Frontend locked, bit errors 0

Right-click the start window to open the control window (see Figure 8.3, "Running xawtv4" (page 137)). Start a scan for available DVB stations with *Edit* → *Scan DVB*. A channel scanner and browser window open. Select a bouquet to prepare the scan. This can be done manually with *Commands* → *Tune manually* if you already know the tuning parameters of the bouquet or by requesting them from a built-in database of xawtv4 via *Database* → *_country_* → *_channel number_* (replace `_country_` and `_channel_number_`) by the actual values for your location.

As soon as the scanner is tuned-in, the first data is displayed in the browser window. Launch a full scan of all available stations with *Command* → *Full Scan*. While the scanner is running, you can select your favorite stations and add them to the station list by simply dragging them into the control window. Leave the channel scanner and select one of the channels to start watching the broadcast.

TIP: Editing the Station List

Using keyboard shortcuts, control the channel selection using your keyboard. To set a shortcut for any station contained in your station list, select the station, click *Edit* → *Edit Station*. A dialog called *TV Station Properties* opens. Enter the shortcut and leave the dialog with *OK*. This dialog also allows you to define submenus holding groups of stations (such as "news" or "private").

The xawtv4 software package contains several more useful stand-alone multimedia applications:

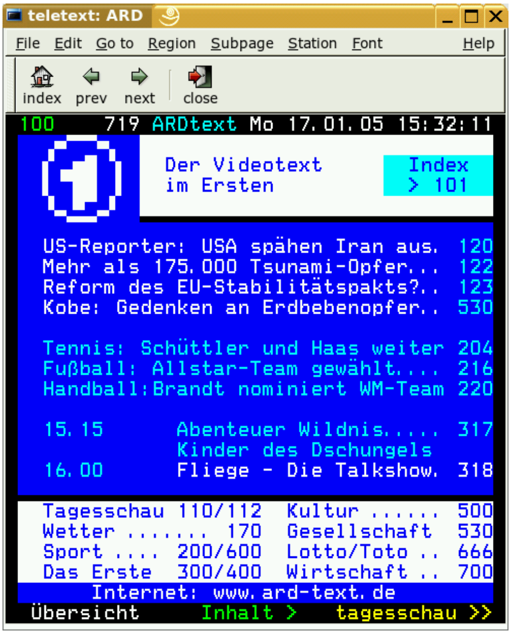
pia4

A lean command-line–controlled movie player that can be used to play any movie streams recorded by xawtv4.

mtt4

A video text browser (see Figure 8.4, “The mtt4 Video Text Browser” (page 138)).

Figure 8.4 *The mtt4 Video Text Browser*



```
100      719 ARDtext Mo 17.01.05 15:32:11
            Der Videotext       Index
            im Ersten           > 101

 US-Reporter: USA spähen Iran aus. 120
 Mehr als 175.000 Tsunami-Opfer... 122
 Reform des EU-Stabilitätspakts?.. 123
 Kobe: Gedenken an Erdbebenopfer.. 530

 Tennis: Schüttler und Haas weiter 204
 Fußball: Allstar-Team gewählt.... 216
 Handball:Brandt nominiert WM-Team 220

 15.15      Abenteuer Wildnis..... 317
            Kinder des Dschungels
 16.00      Fliege - Die Talkshow. 318

 Tagesschau 110/112  Kultur ...... 500
 Wetter ....... 170  Gesellschaft  530
 Sport .... 200/600  Lotto/Toto .. 666
 Das Erste  300/400  Wirtschaft .. 700
      Internet: www.ard-text.de
 Übersicht     Inhalt >   tagesschau >>
```

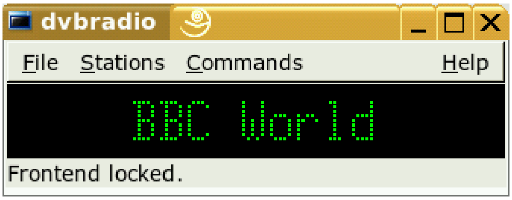
alexplore

A stand-alone DVB channel scanner application. Its functionality is integrated into xawtv4.

dvbradio

A DVB radio player. Use it to listen to DVB-S radio streams after you have completed the initial station scan (see Figure 8.5, “DVB Radio” (page 139)).

Figure 8.5 *DVB Radio*



Frontend locked.

dvbrowse

An EPG browser application. Use it to get EPG information after you have completed the initial station scan.

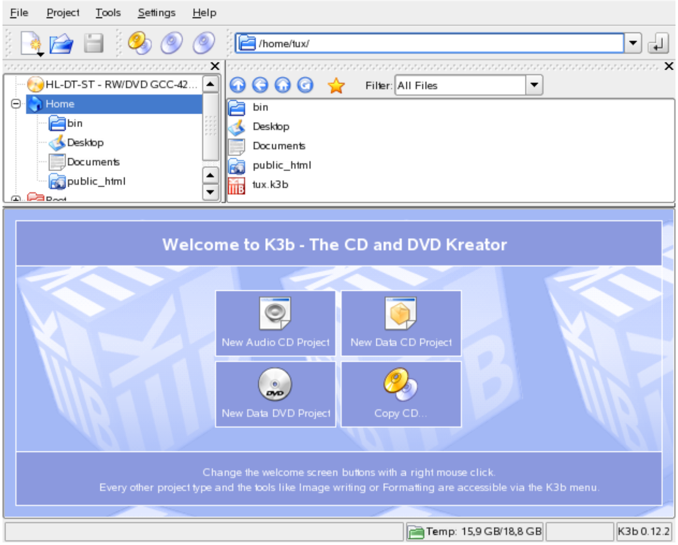
9 K3b—Burning CDs or DVDs

K3b is a comprehensive program for writing data and audio CDs and DVDs. Start the program from the main menu or by entering the command `k3b`. The following sections brief you on how to start a basic burning process to get your first Linux-made CD or DVD.

9.1 Creating a Data CD

To create a data CD, go to *File* → *New Project* → *New Data Project*. The project view appears in the lower part of the window, as shown in Figure 9.1, "Creating a New Data CD" (page 142). Drag the desired directories or individual files from your home directory to the project folder and drop them there. Save the project under a name of your choice with *File* → *Save as*.

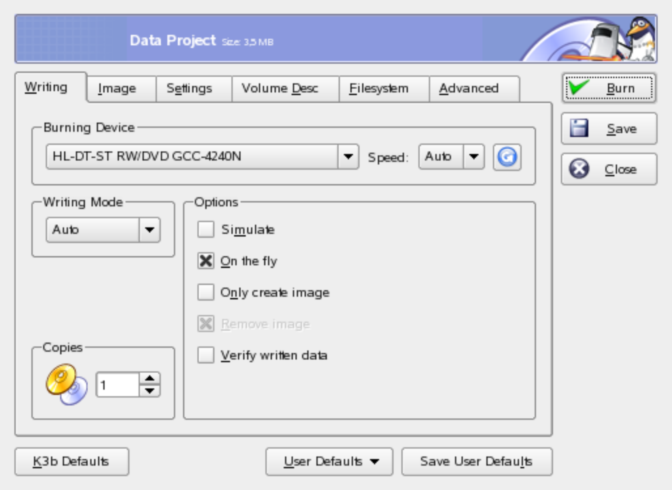
Figure 9.1 *Creating a New Data CD*



Change the welcome screen buttons with a right mouse click.
Every other project type and the tools like Image writing or Formatting are accessible via the K3b menu.

Then select *Burn* from the toolbar or hit Ctrl + B. A dialog with six tabs offering various options for writing the CD opens. See Figure 9.2, "Customizing the Burning Process" (page 142).

Figure 9.2 *Customizing the Burning Process*



Burning Device
HL-DT-ST RW/DVD GCC-4240N
Speed: Auto

Writing Mode
Auto

Options
□ Simulate
☒ On the fly
□ Only create image
☒ Remove image
□ Verify written data

Copies
1

The *Writing* tab has various settings for the burning device, the speed, and the burning options. The following options are offered here:

Burning Device
The detected writer is displayed under this pop-up menu. You can select the speed here too.

WARNING: Select Writing Speed with Care

Normally, you should select *Auto*, which chooses the maximum writing speed possible. However, if you increase this value but your system is not able to send the data fast enough, the likelihood of buffer underruns increases.

Writing Mode
This option determines how the laser writes a CD. In DAO (disk at once) mode, the laser is not deactivated while the CD is written. This mode is recommended for the creation of audio CDs. However, it is not supported by all CD writers. In the TAO mode (track at once), a separate write process is used for each individual track. The RAW mode is not used very often, because the writer does not perform any data corrections. The best setting is *Auto*, because it allows K3b to use the most suitable settings.

Simulate
This function can be used to check if your system supports the selected writing speed. The writing is performed with the laser deactivated to test the system.

On the Fly
Burns the desired data without first creating an image file (do not use this feature on low-performance machines). An image file—also known as an ISO image—is a file containing the entire CD content that is subsequently written to the CD exactly as it is.

Only Create Image
This option creates an image file. Set the path for this file under *Temporary File*. The image file can be written to CD at a later time. To do this, use *Tools → CD → Burn CD Image*. If this option is used, all other options in this section are deactivated.

Remove Image
Remove the temporary image file from hard disk when finished.

Verify Written Data

Check the integrity of the written data by comparing the MD5 sums of the original and the burned data.

The *Image* tab is only accessible if the option *Only create image* from the previous tab is selected. If this is the case, you can determine the file where the ISO is written.

The *Settings* tab contains two options: *Datatrack Mode* and *Multisession Mode*. The *Datatrack Mode* options contains configuration of how data tracks may be written. In general, *auto* is considered the best suited method. The *Multisession Mode* is used to append data to a already written, but not finalized, CD.

In the *Volume Desc* tab, enter some general information that can be used to identify this particular data project, its publisher and preparer, and the application and operating system used in the creation of this project.

Under *File system*, specify settings for the file system on the CD (RockRidge, Joliet, UDF). Also determine how symbolic links, file permissions, and blanks are treated. In the *Advanced* tab, experienced users can make additional settings.

After adjusting all settings to your needs, start the actual burning process using *Burn*. Alternatively, save these settings for future use and adjustment with *Save*.

9.2 Creating an Audio CD

Basically, there are no significant differences between creating an audio CD and creating a data CD. Select *File* → *New Audio Project*. Drag and drop the individual audio tracks to the project folder. The audio data must be in WAV or Ogg Vorbis format. Determine the sequence of the tracks by moving them up or down in the project folder.

With the help of CD Text, you are able to add certain text information to a CD, such as CD title, artist name, and track name. CD players that support this feature can read and display this information. To add CD Text information to your audio tracks, select the track first. Right-click and select *Properties*. A new window opens in which to enter your information.

The dialog for burning an audio CD is not very different from the dialog for burning a data CD. However, the *Disc at once* and the *Track at once* modes have greater importance. The *Track at once* mode inserts an intermission of two seconds after each track.

TIP: Preserving Data Integrity

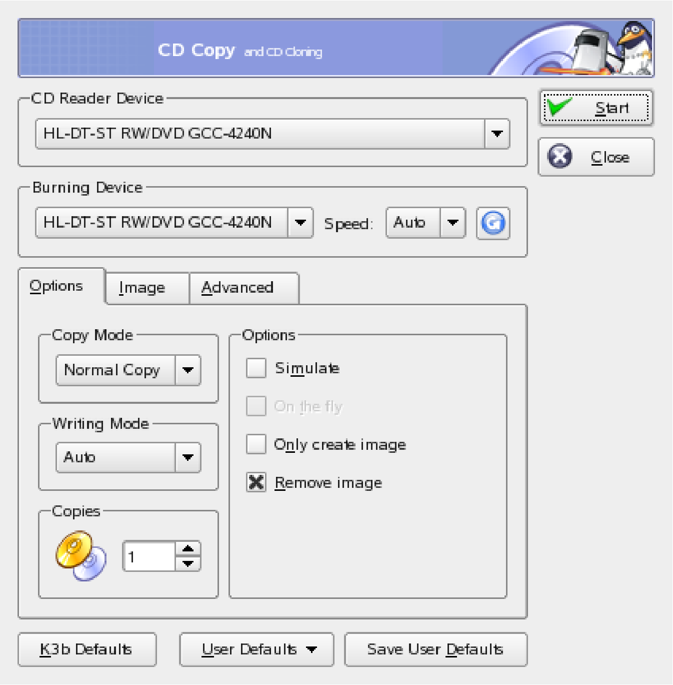
When burning audio CDs, choose a lower burning speed to reduce the risk of burning errors.

After adjusting all settings to your needs, start the actual burning process using *Burn*. Alternatively, save these settings for future use and adjustment with *Save*.

9.3 Copying a CD or DVD

Select *Tools* → *Copy CD* or *Tools* → *Copy DVD* depending on your media. In the dialog that opens, make the settings for the reading and writing device as shown in Figure 9.3, "Copying a CD" (page 145). The writing options discussed are also available here. An additional function enables the creation of several copies of the CD or DVD.

Figure 9.3 *Copying a CD*



CD Reader Device: HL-DT-ST RW/DVD GCC-4240N

Burning Device: HL-DT-ST RW/DVD GCC-4240N Speed: Auto

Copy Mode: Normal Copy

Writing Mode: Auto

Copies: 1

Options
- □ Simulate
- □ On the fly
- □ Only create image
- ☒ Remove image

Check *On the fly* to burn the CD as soon as it has been read or select *Only create image* to create an image in the path specified in *Temp Directory* → *Write image file to* and burn the image later.

9.4 Writing ISO Images

If you already have an ISO image, go to *Tools* → *CD* → *Burn CD image*. A window opens in which to enter the location of the *Image to Burn*. K3b calculates a check sum and displays it in *MD5 Sum*. If the ISO file was downloaded from the Internet, this sum shows if the download was successful.

Use the *Options* and *Advanced* tabs to set your preferences. To burn the CD, click *Start*.

9.5 Creating a Multisession CD or DVD

Multisession discs can be used to write data in more than one burning session. This is useful, for example, for writing backups that are smaller than the media. In each session, you can add another backup file. The interesting part is that you are not only limited to data CDs or DVDs. You can also add audio sessions in a multisession disc.

To start a new multisession disc, do the following:

1. Create your data disc first and add all your files. You cannot start with an audio CD session. Make sure that you do not fill up the whole disc, because otherwise you cannot append a new session.

2. Burn your data with *Project* → *Burn*. A dialog box appears.

3. Go to the tab *Settings* and select *Start Multisession*.

4. Configure other options if needed. See also Section 9.1, "Creating a Data CD" (page 141).

5. Start the buring session with *Burn*.

After a successful burning process, you have created a multisession disc. As long as the media contains enough space, you can append more sessions if you like. Finish discs only if you are sure you do not need any new sessions or the space is occupied.

NOTE: About Storage Space on Multisession Discs

Be aware that multisession discs need space for bookkeeping all the entries from your sessions. This leads to a smaller amount of available space on your disc. The amount depends on the number of sessions.

9.6 For More Information

Apart from the two main functions described above, K3b offers other functions, such as the creation of DVD copies, reading audio data in WAV format, rewriting CDs, and playing music with the integrated audio player. A detailed description of all available program features is available at `http://k3b.sourceforge.net`.

Part IV Office

The OpenOffice.org Office Suite 10

OpenOffice.org is a powerful office suite that offers tools for all types of office tasks, such as writing texts, working with spreadsheets, or creating graphics and presentations. With OpenOffice.org, use the same data across different computing platforms. You can also open and edit files in Microsoft Office formats then save them back to this format, if needed. This chapter only covers the basic skills needed to get started with OpenOffice.org. Start the application from the SUSE menu or using the command `ooffice`.

OpenOffice.org consists of several application modules (subprograms), which are designed to interact with each other. They are listed in Table 10.1, "The OpenOffice.org Application Modules" (page 151). The discussion in this chapter is focused on Writer. A full description of each module is available in the online help, described in Section 10.6, "For More Information" (page 157).

Table 10.1 *The OpenOffice.org Application Modules*

Writer	Powerful word processor application
Calc	Spreadsheet application that includes a chart utility
Draw	Drawing application for creating vector graphics
Math	Application for generating mathematical formulas
Impress	Application for creating presentations
Base	Database application

The appearance of the application varies depending on which desktop or window manager is used. Additionally, the open and save dialog formats for your desktop are used. Regardless of the appearance, the basic layout and functions are the same.

10.1 Compatibility with Other Office Applications

OpenOffice.org is able to work with Microsoft Office documents, spreadsheets, presentations, and databases. They can be seamlessly opened like other files and saved back to that format. Because the Microsoft formats are closed and the specifics are not available to other applications, there are occasionally formatting issues. If you have problems with your documents, consider opening them in the original application and resaving in an open format, such as RTF for text documents or CSV for spreadsheets.

To convert a number of documents, such as when first switching to the application, select *File → Wizard → Document Converter*. Choose the file format from which to convert. There are several StarOffice and Microsoft Office formats available. After selecting a format, click *Next* then specify where OpenOffice.org should look for templates and documents to convert and in which directory the converted files should be placed. Before continuing, make sure that all other settings are appropriate. Click *Next* to see a summary of the actions to perform, which gives another opportunity to check whether all settings are correct. Finally, start the conversion by clicking *Convert*.

IMPORTANT: Finding Windows Files

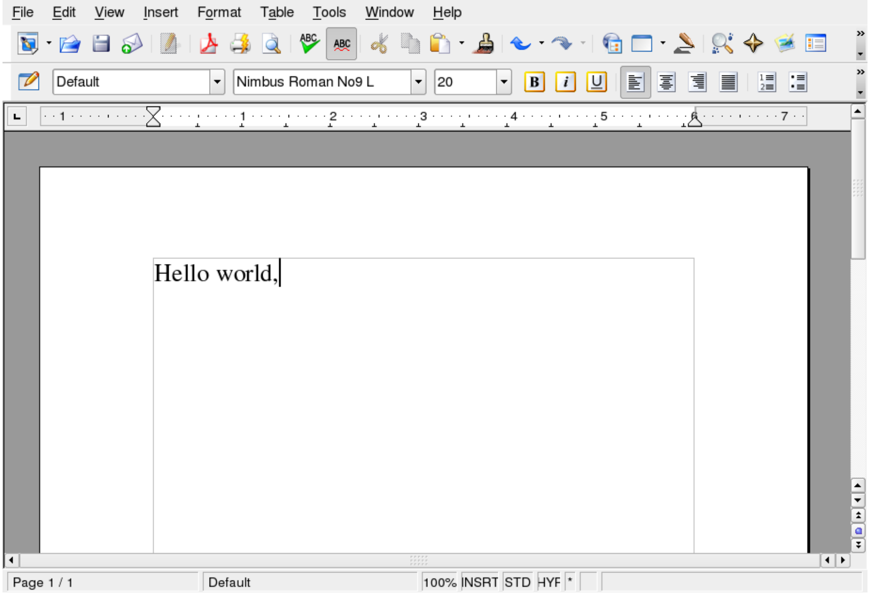
Documents from a Windows partition are usually in a subdirectory of `/windows`.

When sharing documents with others, you have several options. If the recipient only needs to read the document, export it to a PDF file with *File → Export as PDF*. PDF files can be read on any platform using a viewer like Adobe Acrobat Reader. To share a document for editing, use one of the regular document formats. The default formats comply with the OASIS standard XML format, making them compatible with a number of applications. TXT and RTF formats, although limited in formatting, might be a good option for text documents. CSV is useful for spreadsheets. OpenOffice.org might also offer your recipient's preferred format, especially Microsoft formats.

OpenOffice.org is available for a number of operating systems. This makes it an excellent tool when a group of users frequently need to share files and do not use the same system on their computers.

10.2 Word Processing with Writer

***Figure 10.1** The OpenOffice.org Writer*



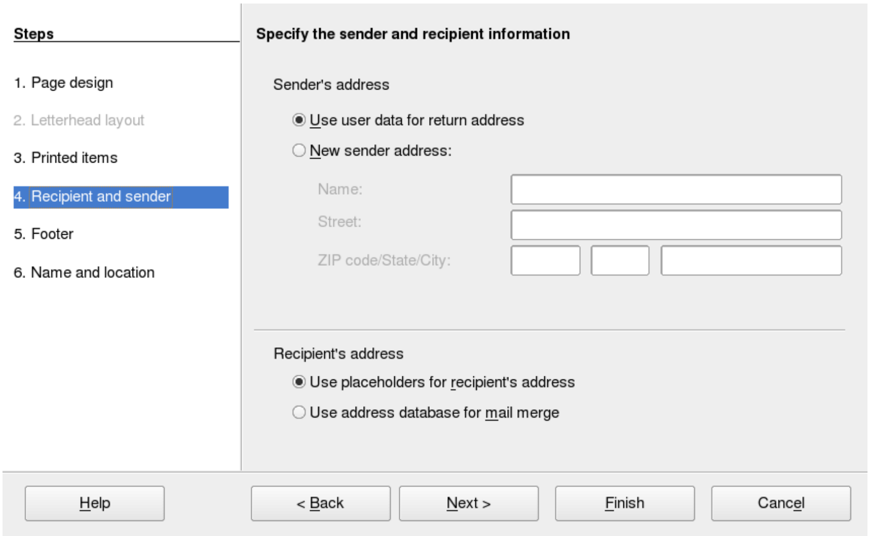
Default | Nimbus Roman No9 L | 20

Hello world,

Page 1 / 1 | Default | 100% INSRT STD HYP *

There are two ways to create a new document. To create a document from scratch, use *File* → *New* → *Text Document*. To use a standard format and predefined elements for your own documents, try a wizard. Wizards are small utilities that let you make some basic decisions then produce a ready-made document from a template. For example, to create a business letter, select *File* → *Wizards* → *Letter*. Using the wizard's dialogs, easily create a basic document using a standard format. A sample wizard dialog is shown in Figure 10.2, “An OpenOffice.org Wizard” (page 154).

Figure 10.2 *An OpenOffice.org Wizard*



Steps

1. Page design
2. Letterhead layout
3. Printed items
4. Recipient and sender
5. Footer
6. Name and location

Specify the sender and recipient information

Sender's address

● Use user data for return address

○ New sender address:

Name:

Street:

ZIP code/State/City:

Recipient's address

● Use placeholders for recipient's address

○ Use address database for mail merge

Enter text in the document window as desired. Use the *Formatting* toolbar or the *Format* menu to adjust the appearance of the document. Use the *File* menu or the relevant buttons in the toolbar to print and save your document. With the options under *Insert*, add extra items to your document, such as a table, picture, or chart.

10.2.1 Selecting Text

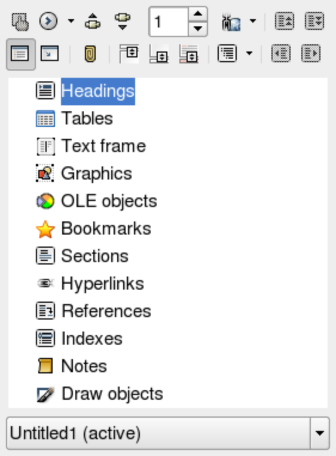
To select text, click the desired beginning of the selection and, keeping the mouse button pressed, move the cursor towards the end of the range (which can be characters, lines, or entire paragraphs). Release the button when all desired text is selected. While selected, text is displayed in inverted colors. Open a context menu by right-clicking the selection. Use the context menu to change the font, the font style, and other text properties.

Selected text can be cut or copied to the clipboard. Cut or copied text can be pasted back into the document at another location. Use the context menu, *Edit*, or the relevant toolbar icons to access these functions.

10.2.2 Navigating in Large Documents

The Navigator displays information about the contents of a document. It also enables you to jump quickly to the different elements included. For example, use the Navigator to get a quick overview of all the chapters or to see a list of the images included in the document. Open it by selecting *Edit → Navigator*. Figure 10.3, "The Navigator in Writer" (page 155) shows the Navigator in action. The elements listed in the Navigator vary according to the document loaded in Writer.

Figure 10.3 *The Navigator in Writer*



10.2.3 Formatting with Styles

The dialog opened with *Format → Styles and Formatting* can help you format text in a number of ways. If you set the drop-down list at the bottom of this dialog to *Automatic*, OpenOffice.org tries to offer a selection of styles adapted to the task at hand. If you select *All Styles*, the Stylist offers all styles from the currently active group. Select groups with the buttons at the top.

By formatting your text with this method, called *soft formatting*, text is not formatted directly. Instead, a style is applied to it. The style can be modified easily, automatically resulting in a formatting change of all the text to which it is assigned.

To assign a style to a paragraph, select the style to use then click the paint bucket icon in *Styles and Formatting*. Click the paragraphs to which to assign the style. Stop assigning the style by pressing Esc or clicking the paint bucket icon again.

Easily create your own styles by formatting a paragraph or a character as desired using the *Format* menu or toolbar. Select the formatted item from which to copy the style. Then click and hold the button to the right of the bucket in *Styles and Formatting* and select *New Style from Selection* from the menu that opens. Enter a name for your style and click *OK*. This style can then be applied to other texts.

Change details of a style by selecting it in the list, right-clicking, and selecting *Modify* from the menu. This opens a dialog in which all the possible formatting properties are available for modification.

10.3 Introducing Calc

Calc is OpenOffice.org's spreadsheet application. Create a new spreadsheet with *File → New → Spreadsheet* or open one with *File → Open*. Calc can read and save in Microsoft Excel's format.

In the spreadsheet cells, enter fixed data or formulas. A formula can manipulate data from other cells to generate a value for the cell in which it is inserted. You can also create charts from cell values.

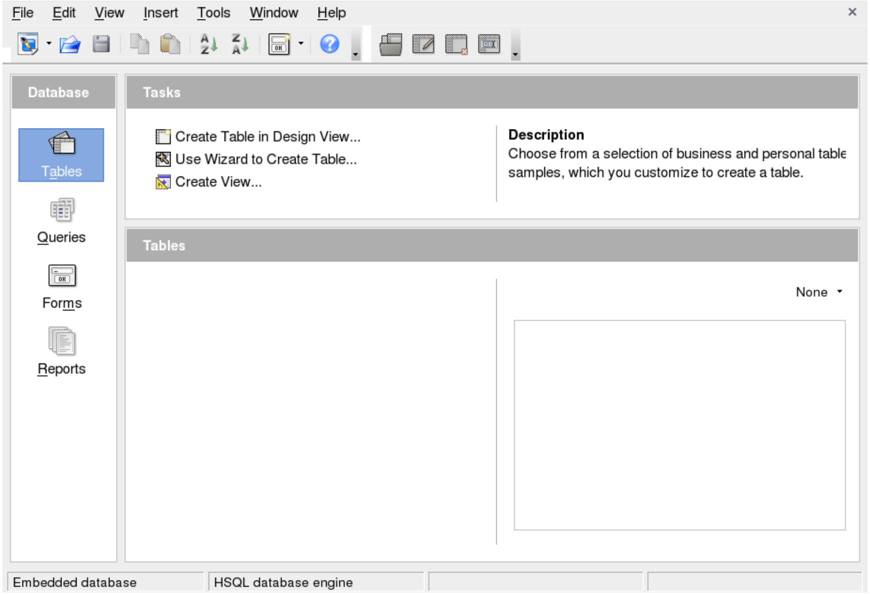
10.4 Introducing Impress

Impress is designed for creating presentations for screen display or printing, such as on transparencies. Create a presentation from scratch with *File → New → Presentation*. To create one with the assistance of a wizard, use *File → Wizards → Presentation*. Open an existing presentation with *File → Open*. Impress can open and save Microsoft PowerPoint presentations.

10.5 Introducing Base

OpenOffice 2.0 introduces a new database module. Create a database with *File → New → Database*. A wizard opens to assist in creating the database. Base can also work with Microsoft Access databases.

Figure 10.4 *Base—Databases in OpenOffice.org*



Description
Choose from a selection of business and personal table samples, which you customize to create a table.

Embedded database | HSQL database engine

Tables, forms, queries, and reports can be created manually or using convenient wizards. For example, the table wizard contains a number of common fields for business and personal use. Databases created in Base can be used as data sources, such as when creating form letters.

10.6 For More Information

OpenOffice.org includes a number of information options that provide different levels of information. To get thoroughly acquainted with a topic, select *Help → OpenOffice.org Help*. The help system provides in-depth information about each of the modules of OpenOffice.org (Writer, Calc, Impress, etc.).

When the application is first started, it provides *Tips*, short information about buttons when the mouse hovers over them, and the *Help Agent*, information based on actions performed. To get more extensive information about buttons than the *Tips* provide, use *Help* → *What's This* then hover over the desired buttons. To end *What's This* mode, click. If you frequently need this function, consider enabling the *Extended Tips* in *Tools* → *Options* → *OpenOffice.org* → *General*. The *Help Agent* and *Tips* can also be enabled and disabled here.

The OpenOffice.org Web site is `http://www.openoffice.org`. There, find mailing lists, articles, and bug information. This site provides the versions for various operating systems for download.

Evolution: An E-Mail and Calendar Program 11

Evolution is a groupware suite that offers the usual e-mail features along with extended features, like task lists and a calendar. The application also provides a complete address book that includes the ability to send contact information to others in vCard format.

Start Evolution from the main menu or with `evolution`. When started for the first time, Evolution offers a configuration assistant. Its use is described in Section 11.3.1, "Configuring Accounts" (page 161).

IMPORTANT: Microsoft Exchange Accounts

To use Evolution with Microsoft Exchange, you must install the `ximian-connector` package. Install it with YaST.

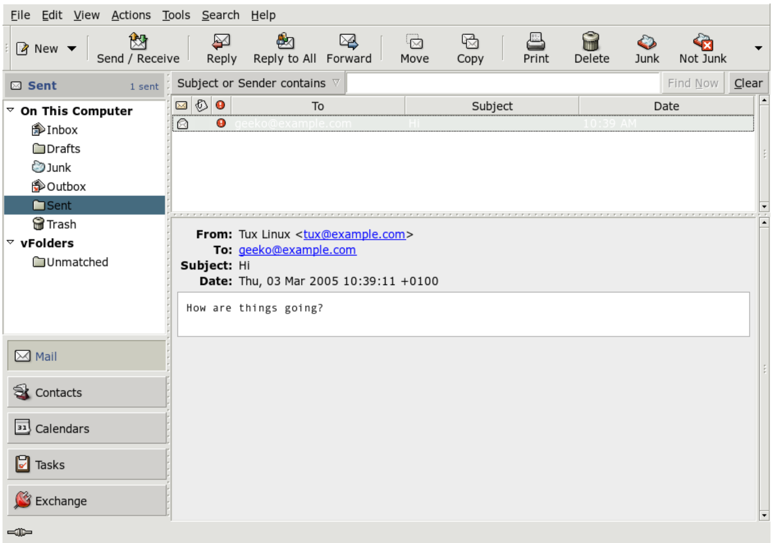
11.1 Importing E-Mail from Other Mail Programs

To import e-mail from other e-mail programs, such as Netscape, select *File* → *Import*. For mbox formats, select *Import a single file*. For Netscape, select *Import data and settings from older programs*. To work with data from programs using the maildir format, such as KMail, configure an account that accesses the mail directory.

11.2 Evolution Overview

The default window view is shown in Figure 11.1, "The Evolution Window with Mail" (page 160). The available menus, menu items, and the icons in the toolbar vary depending on the open component. Use the left frame to select which information to display in the right frame. Adjust the size of the frames by dragging the dividing bars.

Figure 11.1 The Evolution Window with Mail



To	Subject	Date
geeko@example.com	Hi	10:39 AM

From: Tux Linux <tux@example.com>
To: geeko@example.com
Subject: Hi
Date: Thu, 03 Mar 2005 10:39:11 +0100

```
How are things going?
```

11.2.1 Mail

In this view, the upper half of the window shows the contents of the current folder. The lower half is a preview pane used to display the selected mail message. To display a different folder, select a folder from the folder list in the left frame.

Use the search bar to search the messages in a folder. To sort messages by a table header, click the desired header. The arrow to the right shows whether the column is sorted in ascending or descending order. Click the column header until the messages are sorted in the desired direction.

11.2.2 Contacts

This view shows all the addresses in your address book. To locate a particular address, use the search bar or click the button to the right displaying the first letter of the contact's last name. Add contacts or lists with the toolbar.

11.2.3 Calendar

The initial display shows a view of the current day with the month and a task list shown in an additional pane to the right. Week, work week, and month views are also available from the toolbar or the *View* menu. Use the search bar to find an appointment that has been entered in the calendar. Add appointments and tasks using the buttons in the toolbar. You can also use the toolbar to page through the calendar or jump to a specific date.

11.2.4 Tasks

Tasks provides a list of tasks. Details of the selected task are shown in the lower part of the window. Use *File* → *New* → *Task* to add a new task. Search the tasks with the search bar. Assign tasks to others by right-clicking the task and selecting *Assign Task*. *Open* the task to add more details, such as a due date and completion status.

11.3 Mail

The Evolution mail component can work with multiple accounts in a variety of formats. It offers useful features, such as virtual folders for showing search results and filtering for junk mail. Configure the application in *Edit* → *Preferences*.

11.3.1 Configuring Accounts

Evolution can retrieve e-mail from multiple mail accounts. The account you want to send e-mail from can be selected when you compose a message. Configure mail accounts in *Edit* → *Preferences* → *Mail Accounts*. To modify an existing configuration, select it and click *Edit*. To delete an account, select it and click *Delete*.

To add a new account, click *Add*. This opens the configuration assistant. Click *Forward* to use it. Enter your name and your e-mail address in the respective fields. Enter the optional information if desired. Check *Make this my default account* to use this account by default when writing mails. Click *Forward*.

Select the appropriate incoming e-mail format for this address in *Server Type*. *POP* is the most common format for downloading mail from a remote server. *IMAP* works with mail folders on a special server. Obtain this information from your ISP or server administrator. Complete the other relevant fields displayed when the server type is selected. Click *Forward* when finished. Select the desired *Receiving Options*, if available. Click *Forward*.

Next, configure the mail delivery options. To submit outgoing e-mail to the local system, select *Sendmail*. For a remote server, select *SMTP*. Get the details from your ISP or server administrator. For SMTP, complete the other fields displayed after selection. Click *Forward* when finished.

By default, the e-mail address is used as the name to identify the account. Enter another name if desired. Click *Forward*. Click *Apply* to save your account configuration.

To make an account the default account for sending e-mail, select the desired account then press *Default*. To disable the retrieving of e-mail from an account, select the account then click *Disable*. A disabled account can still be used as the address for sending, but that account is not checked for incoming e-mail. If necessary, reactivate the account with *Enable*.

11.3.2 Creating Messages

To compose a new message, click *New* → *Mail Message*. Replying to or forwarding a message opens the same message editor. Next to *From*, select from which account to send the message. In the recipient fields, enter an e-mail address or part of a name or address in your address book. If Evolution can match what you enter to something in the address book, a selection list is displayed. Click the desired contact or complete your input if there are no matches. To select a recipient directly from the address book, click *To* or *CC*.

Evolution can send e-mail as plain text or HTML. To format HTML mail, select *Format* in the toolbar. To send attachments, select *Attach* or *Insert* → *Attachment*.

To send your message, click *Send*. If not ready to send it immediately, make another selection under *File*. For example, save the message as a draft or send it later.

11.3.3 Encrypted E-Mail and Signatures

Evolution supports e-mail encryption with PGP. It can sign e-mail and check signed e-mail messages. To use these features, generate and manage keys with an external application, such as gpg or KGpg.

To sign an e-mail message before sending it, select *Security → PGP sign*. When you click *Send*, a dialog prompts for the password of your secret key. Enter the password and exit the dialog with *OK* to send the signed e-mail. To sign other e-mail messages in the course of this session without needing to "unlock" the secret key repeatedly, check *Remember this password for the remainder of this session*.

When you receive signed e-mail from other users, a small padlock icon appears at the end of the message. If you click this symbol, Evolution starts an external program (gpg) to check the signature. If the signature is valid, a green check mark appears next to the padlock symbol. If the signature is invalid, a broken padlock appears.

The encryption and decryption of e-mail is just as easy. After composing the e-mail message, go to *Security → PGP encrypt* and send the e-mail message. When you receive encrypted messages, a dialog asks for the password of your secret key. Enter the passphrase to decrypt the e-mail message.

11.3.4 Folders

It is often convenient to sort e-mail messages into a variety of folders. Your folder tree is shown in the left frame. If accessing mail over IMAP, the IMAP folders are also shown in this folder bar. For POP and most other formats, your folders are stored locally, sorted under *Local Folders*.

Several folders are included by default. *Inbox* is where new messages fetched from a server are initially placed. *Sent* is used for saving copies of sent e-mail messages. The *Outbox* provides temporary storage for e-mail that has not yet been sent. It is useful if working offline or if the outgoing mail server is temporarily unreachable. *Drafts* is used for saving unfinished e-mail messages. The *Trash* folder is intended for temporary storage of deleted items. *Junk* is for Evolution's junk mail filtering feature.

New folders can be created under *On This Computer* or as subfolders of existing folders. Create as complex a folder hierarchy as desired. To create a new folder, select *File → New → Mail Folder*. In the *Mail Folder* dialog, enter a name for the new folder. Use the mouse to determine the parent folder under which to place the new folder. Exit the dialog with *OK*.

To move a message into a folder, select the message to move. Right-click to open the context menu. Select *Move to Folder* and, in the dialog that opens, the destination folder. Click *OK* to move the message. The message header in the original folder is shown with a line through it, meaning that message is marked for deletion from that folder. The message is stored in the new folder. Messages can be copied in a similar manner.

Manually moving a number of messages into different folders can be time-consuming. Filters can be used to automate this procedure.

11.3.5 Filters

Evolution offers a number of options for filtering e-mail. Filters can be used to move a message into a specific folder or to delete a message. Messages can also be moved directly to the trash with a filter. There are two options for creating a new filter: creating a filter from scratch or creating a filter based on a message to filter. The latter is extremely useful for filtering messages sent to a mailing list.

Setting Up a Filter

Select *Tools → Filters*. This dialog lists your existing filters, which can be edited or deleted. Click *Add* to create a new filter. Alternatively, to create a filter based on a message, select the message then *Tools → Create Filter from Message*.

Enter a name for the new filter in *Rule Name*. Select the criteria to use for the filter. Options include sender, recipients, source account, subject, date, and status. The drop-box showing *Contains* provides a variety of options, such as *contains*, *is*, and *is not.* Select the appropriate condition. Enter the text for which to search. Click *Add* to add more filter criteria. Use *Execute actions* to determine if all or only some of the criteria must be met to apply the filter.

In the lower part of the window, determine the action to take when the filter criteria are met. Messages can, for example, be moved or copied to a folder or assigned a special color. When moving or copying, click to select the destination folder. In the folder list that appears, select the folder. To create a new folder, click *New*. Click *OK* when the correct folder is selected. When finished creating the filter, click *OK*.

Applying Filters

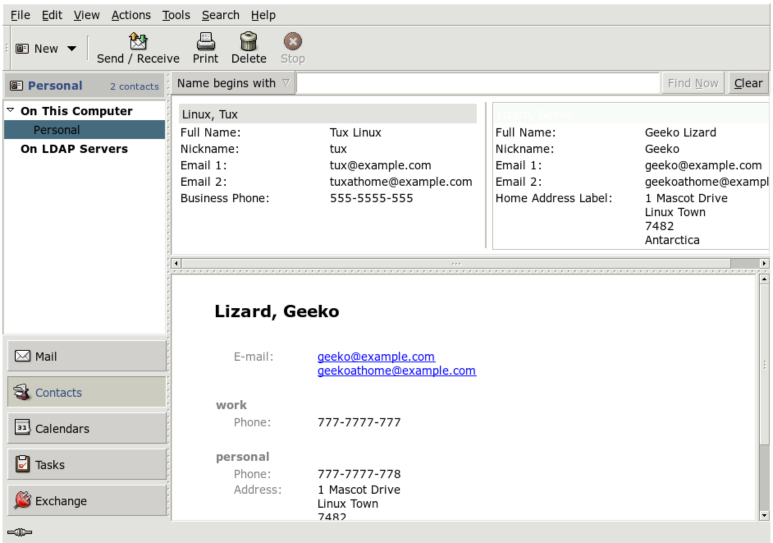
Filters are applied in the order listed in the dialog accessed with *Tools* → *Filters*. Change the order by highlighting a filter and clicking *Up* or *Down*. Click *OK* to close the filter dialog when finished.

Filters are applied to all new mail messages. They are not applied to mail already in your folders. To apply filters to messages already received, select the desired messages then select *Actions* → *Apply Filters*.

11.4 Contacts

Evolution can use several different address books. Available books are listed in the left frame. Search for a particular contact using the search bar. Add contacts in several formats to the Evolution address book using *File* → *Import*. Right-click a contact to open a menu in which to select from a variety of options, such as forwarding the contact or saving it as a vCard. Double-click a contact to edit it.

Figure 11.2 *The Evolution Address Book*



Linux, Tux
Full Name: Tux Linux
Nickname: tux
Email 1: tux@example.com
Email 2: tuxathome@example.com
Business Phone: 555-5555-555

Full Name: Geeko Lizard
Nickname: Geeko
Email 1: geeko@example.com
Email 2: geekoathome@exampl
Home Address Label: 1 Mascot Drive
Linux Town
7482
Antarctica

Lizard, Geeko

E-mail: geeko@example.com
geekoathome@example.com

work
Phone: 777-7777-777

personal
Phone: 777-7777-778
Address: 1 Mascot Drive
Linux Town
7482

11.4.1 Adding Contacts

Along with the name and e-mail address, Evolution can store other address and contact information about a person. Quickly add the e-mail address of a sender by right-clicking the marked address in the message preview. To enter a completely new contact, click *New Contact* in the *Contacts* view. Both methods open a dialog in which to enter contact information.

In the *Contact* tab, enter the contact's name, e-mail addresses, telephone numbers, and instant messaging identities. *Personal Information* is for Web addresses and other detailed information. Enter the contact's other address information in *Mailing Address*. After entering all desired details for the contact, click *OK* to add it to the address book.

11.4.2 Making a List

If you frequently send e-mail messages to a group of people, you can simplify the process by creating a list containing those addresses. Click *File* → *New* → *Contact List*. The contact list editor opens. Enter a name for the list. Add addresses by typing the address in the box and clicking *Add* or by dragging contacts from the *Contacts* view and dropping them in the box. Toggle *Hide addresses* to select whether the recipients can see who else has received the mail. Click *OK* when finished. The list is now one of your contacts and appears in the composition window after the first few letters are typed.

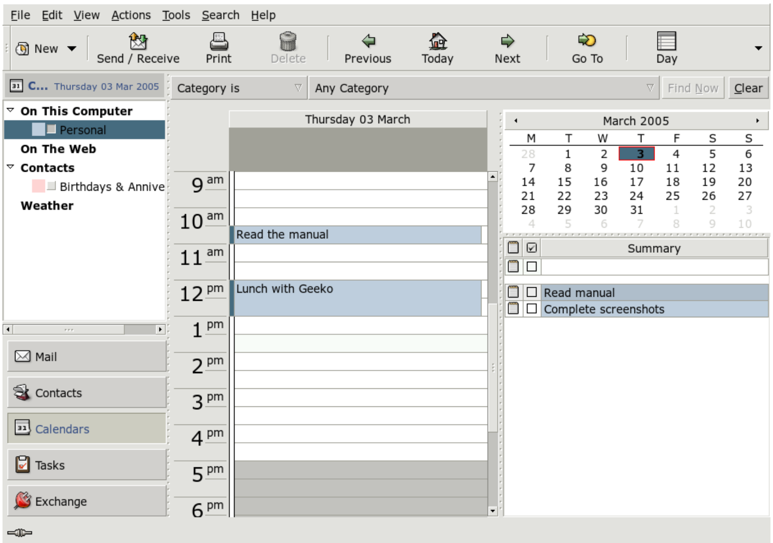
11.4.3 Adding Address Books

Configure additional GroupWise and Exchange address books in the account configuration for that account. To add other local or LDAP books, select *File* → *New* → *Address Book*. In the dialog that opens, select the type of address book and enter the required information.

11.5 Calendars

Evolution can work with multiple calendars. With *File* → *Import*, import calendars in iCalendar format. Use the calendar to enter appointments and schedule meetings with others. If desired, set reminders to let you know when your scheduled appointments are going to start.

Figure 11.3 *The Evolution Calendar*



Read the manual

Lunch with Geeko

March 2005

M	T	W	T	F	S	S
28	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

☐ Read manual
☐ Complete screenshots

11.5.1 Adding Appointments

To add a new appointment to your calendar, click *File* → *New* → *Appointment*. Under the *Appointment* tab, enter the details for the appointment. Select a category, if desired, to ease searching and sorting later. Optionally, use *Alarm* to set an alarm so Evolution will remind you before your appointment starts. If the appointment occurs regularly, set the recurring dates under *Recurrence*. Click *OK* after all settings are made. The new appointment is then shown in your calendar.

11.5.2 Scheduling a Meeting

To schedule a meeting with other people, select *File* → *New* → *Meeting*. Enter information as you would for an appointment. Add the attendees in *Invitations* or *Scheduling*. To enter attendees from your address book, use *Contacts* to open a list of the contacts in your address book. *Scheduling* can also be used to schedule a time that fits all attendees. Press *Autopick* after configuring participants to automatically find a time.

11.5.3 Adding Calendars

GroupWise and Exchange calendars should be configured in the account configuration. To add additional local or Web calendars, select *File* → *New* → *Calendar*. Select the desired type and enter the required information.

11.6 Syncing Data with a Handheld

Evolution is designed so its data can be synced with handheld devices, such as a Palm. The synchronization uses GNOME Pilot. Select *Tools* → *Pilot Settings* to open the configuration wizard. Refer to the help for more information.

11.7 Evolution for GroupWise Users

GroupWise users should have little trouble using Evolution to access their GroupWise accounts. Evolution and GroupWise use very similar terminology. Users familiar with one system should be able to learn the other with minimal effort.

11.7.1 Configuring Evolution to Access Your GroupWise System

Use the Evolution Mail Configuration Assistant to configure Evolution to access your GroupWise system. To start the Evolution Mail Configuration Assistant, click *Preferences → Mail Accounts → Add* then click *Forward*.

On the *Identity* page, provide your e-mail address in the GroupWise system (for example, joe@example.com), then click *Forward*.

On the *Receiving Email* page, select *IMAP* in *Server Type*, specify the hostname of your GroupWise server in *Host*, set the other settings on the *Receiving Options* page as appropriate for your system, then click *Forward*.

On the *Sending Email* page, select *SMTP* in *Server Type*, specify the hostname of your GroupWise server in *Host*, set the other *Sending Email* options as appropriate for your system, then click *Forward*.

On the *Account Management* page, specify the name you want to use to identify this account on the *Evolution Settings* page then click *Forward*.

Click *Apply* to create the GroupWise account. Your GroupWise mailbox now appears in the list of available e-mail accounts.

11.8 For More Information

Evolution offers extensive internal help pages. Use the *Help* menu to access this information. For more information about Evolution, refer to the project's Web site at `http://www.gnome.org/projects/evolution/`.

Kontact: An E-Mail and Calendar Program 12

Kontact combines the functionality of a number of KDE applications into a convenient, single interface for personal information management. These applications include KMail for e-mail, KOrganizer for the calendar, KAddressbook for contacts, and KNotes for notes. It is also possible to sync data with external devices, such as a PalmPilot or other handheld device. Kontact integrates easily with the rest of the KDE desktop and connects to a variety of groupware servers. It includes extra features, such as spam and virus filtering and an RSS reader.

Start Kontact from the main menu with *Office* → *Kontact (Personal Information Manager)*. Alternatively, enter `kontact` in a command line. You can also open the individual components instead of the combined application if you only need partial functionality.

12.1 Importing E-Mail from Other Mail Programs

To import e-mail from other applications, select *Tools* → *Import Messages* from the mail view in Kontact. It currently features import filters for Outlook Express, the mbox format, e-mail text format, Pegasus Mail, Opera, Evolution, and more. The import utility can also be started separately with the command `kmailcvt`.

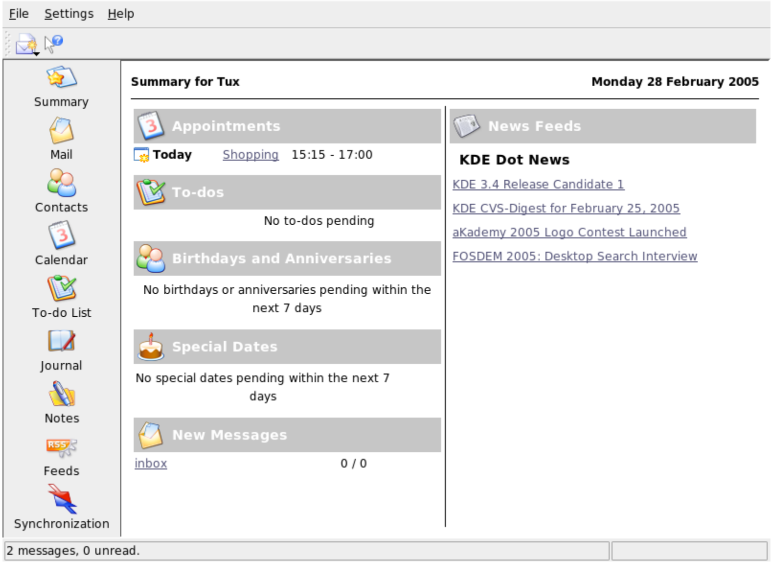
Select the corresponding application and confirm with *Continue*. A file or a folder must be provided, depending on the selected type. Kontact then completes the process.

12.2 Kontact Overview

The default window view, which shows the *Summary*, is shown in Figure 12.1, “The Kontact Window Showing the Summary” (page 172). Use the buttons in the left section to access the different components.

The *Summary* provides basic information, including upcoming birthdays and to-dos, weather, and the status of KPilot. The news section can access RSS feeds to provide updated news of interest to you. Use *Settings* → *Configure Summary View* to configure the information displayed.

Figure 12.1 *The Kontact Window Showing the Summary*



Summary for Tux Monday 28 February 2005

Appointments

Today Shopping 15:15 - 17:00

To-dos

No to-dos pending

Birthdays and Anniversaries

No birthdays or anniversaries pending within the next 7 days

Special Dates

No special dates pending within the next 7 days

New Messages

inbox 0 / 0

News Feeds

KDE Dot News

KDE 3.4 Release Candidate 1

KDE CVS-Digest for February 25, 2005

aKademy 2005 Logo Contest Launched

FOSDEM 2005: Desktop Search Interview

2 messages, 0 unread.

12.2.1 Mail

The folder area to the left contains a list of your mail folders (mail boxes) indicating the total number of messages and how many are still unread. To select a folder, simply click it. The messages in that folder appear in the top right frame. The number of messages in that folder is also shown in the status bar at the bottom of the application window.

The subject, sender, and time of receipt of each message are listed in the header area to the right. Click a message to select it and display it in the message window. Sort the messages by clicking one of the column headers (subject, sender, date, etc.). The contents of the currently selected message are displayed in the message frame of the window. Attachments are depicted as icons at the end of the message, based on the MIME type of the attachment, or they can be displayed inline.

Messages can be marked with different status flags. Change the status with *Message → Mark Message*. You can use this feature to assign a status to a message, such as important or ignored. For example, you can highlight important messages that you do not want to forget. Display only messages with a certain status using *Status* in the search bar.

12.2.2 Contacts

The upper left frame of this component shows all addresses in the currently activated address books. The lower left frame lists your address books and shows whether each one is currently active. The right frame shows the currently selected contact. Use the search bar at the top to find a particular contact.

12.2.3 To-Do List

To-do List shows your list of tasks. Click the field at the top to add a new item to the list. Right-click in a column of an existing item to make changes to the value in that column. An item can be broken into several subitems. Right-click and select *New Sub-to-do* to create a subitem. You can also assign to-dos to other people.

12.2.4 Calendar

The calendar view is divided into a number of frames. By default, view a small calendar of this month and a week view of the current week. Also find a list of to-dos, a detailed view of the current event or to-do, and a list of calendars with the status of each. Select a different view from the toolbar or the *View* menu.

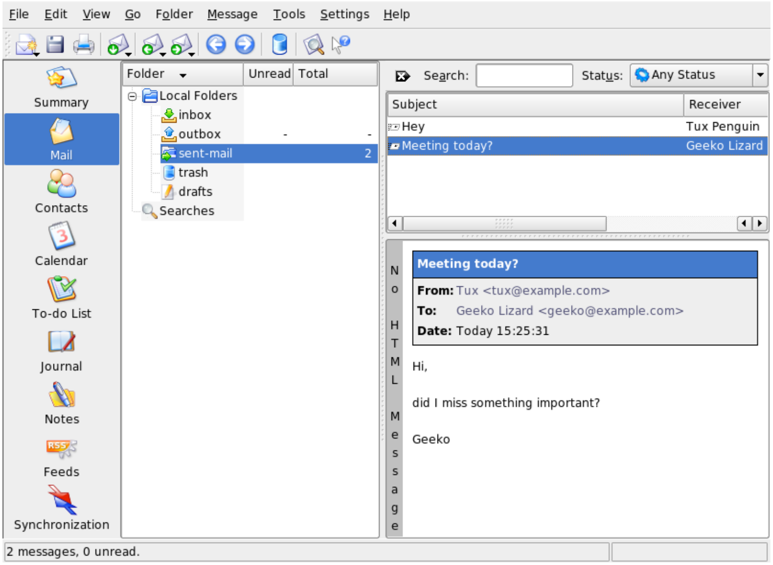
12.2.5 Notes

Use the Notes component to keep sticky notes to yourself. If you are using KDE, use the KNote icon in the system tray to make your notes visible on the desktop.

12.3 Mail

Kontact uses KMail as its e-mail component. To configure it, open the mail component then select *Settings* → *Configure KMail*. KMail is a fully-featured e-mail client that supports a number of protocols. *Tools* contains several useful tools for managing unwanted e-mails. Use *Find* to perform a detailed search for messages. *Anti-Spam Wizard* can help manage tools for filtering unwanted commercial e-mails. *Anti-Virus Wizard* helps manage e-mail virus scanners. These two wizards work with external spam and virus software. If the options are disabled, install additional packages for protection against spam and viruses.

***Figure 12.2** The Kontact Mail Component*



Status: Any Status

Folder	Unread	Total
Local Folders		
inbox		
outbox	-	-
sent-mail		2
trash		
drafts		
Searches		

Subject	Receiver
Hey	Tux Penguin
Meeting today?	Geeko Lizard

Meeting today?

From: Tux <tux@example.com>
To: Geeko Lizard <geeko@example.com>
Date: Today 15:25:31

Hi,

did I miss something important?

Geeko

2 messages, 0 unread.

12.3.1 Configuring Accounts

Kontact can manage multiple e-mail accounts, such as your private e-mail address and your business address. When writing an e-mail, select one of the identities previously defined by clicking *View* → *Identity*. To create a new identity profile, select *Settings* → *Configure KMail* then *Identities* → *New*. In the dialog that opens, give the new identity a name, such as "private" or "office." Click *OK* to open a dialog in which to enter additional information. You can also assign an identity to a folder so that, when replying to a message in that folder, the assigned identity is selected.

Under the *General* tab, enter your name, organization, and e-mail address. Under *Cryptography*, select your keys to send digitally signed or encrypted messages. For the encryption features to work, first create a key with KGpg, described in Chapter 6, *Encryption with KGpg* (page 101).

Under *Advanced*, you can enter a reply-to and a blind carbon-copy address, choose a dictionary, select the folders for drafts and sent messages, and define how messages should be sent. Under *Signature*, decide if and how each of your messages should be signed with an extra block of text at the end. For example. you might sign each e-mail with your contact information. To activate this option, select *Enable Signature* and decide whether to obtain the signature from a file, an input field, or the output of a command. When you are finished with all your identity settings, confirm with *OK*.

The settings under *Network* decide how Kontact receives and sends e-mail. There are two tabs, one each for sending and for receiving mail. Many of these settings vary depending on the system and network in which your mail server is located. If you are not sure about the settings or items to select, consult your ISP or system administrator.

To create outgoing mail boxes under the *Sending* tab, click *Add*. Choose between the SMTP and sendmail transport types. SMTP is the correct choice in most cases. After making this selection, a window appears in which to enter SMTP server data. Provide a name and enter the server address (as given to you by your ISP). If the server wants you to authenticate yourself, enable *Server requires authentication*. Security settings are under the *Security* tab. Specify your preferred encryption method here.

Make settings for receiving e-mail under the *Receiving* tab. Use *Add* to create a new account. Choose between different methods for retrieving mail, such as local (stored in Mbox or Maildir format), POP3, or IMAP. Make the settings appropriate for your server.

12.3.2 Creating Messages

To compose new messages, select *Message* → *New Message* or click the corresponding icon in the toolbar. To send messages from different e-mail accounts, select one of the identities as described in Section 12.3.1, "Configuring Accounts" (page 175). In *To*, enter an e-mail address or part of a name or address in your address book. If Kontact can match what you enter to something in the address book, a selection list opens. Click the desired contact or complete your input if none matches. To select directly from the address book, click the ... button next to the Address field.

To attach files to your message, click the paperclip icon and select the file to attach. Alternatively, drag a file from the desktop or another folder to the *New Message* window or select one of the options in the *Attach* menu. Normally, the format of a file is recognized correctly. If the format is not recognized, right-click the icon. From the menu that appears, select *Properties*. Set the format and filename in the next dialog and add a description. In addition, decide whether the attached file should be signed or encrypted.

When you are finished composing your message, send it immediately with *Message* → *Send* or move it to the outbox with *Message* → *Queue*. If you send the e-mail, the message is copied to `sent-mail` after having been sent successfully. Messages moved to the `outbox` can be edited or deleted.

12.3.3 Encrypted E-Mail and Signatures

To encrypt your e-mail, first generate a key pair as described in Chapter 6, *Encryption with KGpg* (page 101). To configure the details of the encryption procedure, select *Settings* → *Configure KMail* → *Identities* to specify the identity under which to send encrypted and signed messages. Then press *Modify* . After confirming with *OK*, the key should be displayed in the corresponding field. Close the configuration dialog with *OK*.

12.3.4 Folders

Message folders help to organize your messages. By default, they are located in the directory `~/.kde/share/apps/kmail/mail`. When starting KMail for the first time, the program creates several folders. `inbox` is where new messages fetched from a server are initially placed. `outbox` is used for temporary storage of messages queued for sending. `sent-mail` is for copies of messages sent. `trash` contains copies of all e-mails deleted with Del or *Edit → Delete*. `drafts` is where you can save unfinished messages. If you are using IMAP, the IMAP folders are listed below the local folders. Each incoming mail server, for example local or IMAP, has its folders in the Folder list.

If you want to organize your messages in additional folders, create new folders by selecting *Folder → New Folder*. This opens a window in which to specify the name and format of the new folder.

Right-click the folder for a context menu offering several folder operations. Click *Expire* to specify the expiration date for read and unread messages, what should happen with them after expiration, and whether expired messages should be deleted or moved to a folder. If you intend to use the folder to store messages from a mailing list, set the necessary options under *Folder → Mailing List Management*.

To move one or several messages from one folder to another, highlight the messages to move then press M or select *Message → Move to*. In the list of folders that appears, select the folder to which to move your messages. Messages can also be moved by dragging them from the upper window and dropping them into the appropriate folder in the left window.

12.3.5 Filters

Filters are a convenient method of automatically processing incoming mail. They use aspects of the mail, such as sender or size, to move mail to certain folders, delete unwanted mails, bounce mails back to the sender, or perform a number of other actions.

Setting Up a Filter

To create a filter from scratch, select *Settings → Configure Filters*. To create a filter based on an existing message, select the desired message in the Header list, then select *Tools → Create Filter* and the desired filter criteria.

Select the match method for filter criteria (all or any). Then select criteria that applies only to the desired messages. In *Filter Actions*, set what the filter should do to the messages that meet the criteria. *Advanced Options* provides control over when the filter is applied and whether additional filters should be considered for these messages.

Applying Filters

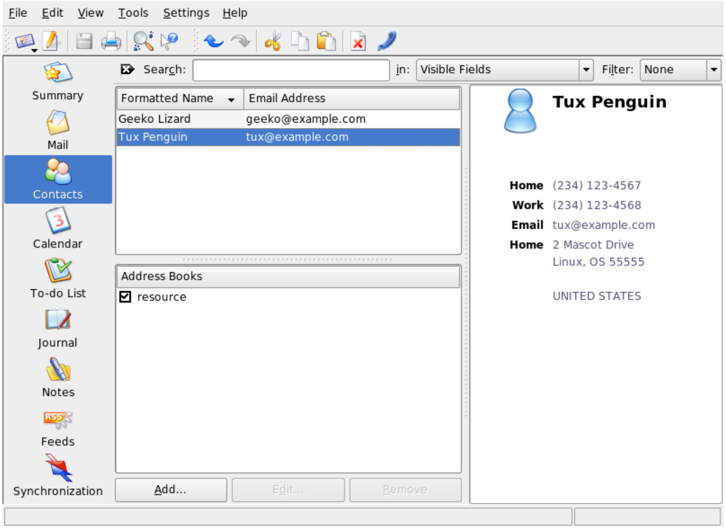
Filters are applied in the order listed in the dialog accessed with *Settings* → *Configure Filters*. Change the order by selecting a filter and clicking the arrow buttons. Filters are only applied to new incoming messages or sent messages as specified in the filter's advanced options. To apply filters to existing messages, select the desired messages then *Message* → *Apply Filters*.

If your filters do not act as expected, monitor them with *Tools* → *Filter Log Viewer*. When logging is enabled in this dialog, it shows how messages are processed by your filters and can help locate the problem.

12.4 Contacts

The contacts component uses KAddressBook. Configure it with *Settings* → *Configure KAddressBook*. To search for a particular contact, use the search bar. With *Filter*, select to display only contacts in a certain category. Right-click a contact to open a menu in which to select from a variety of options, such as sending the contact information in an e-mail.

Figure 12.3 *The Kontact Address Book*



Search: in: Visible Fields Filter: None

Formatted Name	Email Address
Geeko Lizard	geeko@example.com
Tux Penguin	tux@example.com

☑ resource

Tux Penguin

Home	(234) 123-4567
Work	(234) 123-4568
Email	tux@example.com
Home	2 Mascot Drive Linux, OS 55555 UNITED STATES

12.4.1 Adding Contacts

To add a contact with the name and e-mail address from an e-mail, right-click the address in the mail component and select *Open in Address Book*. To add a new contact without using an e-mail, select *File* → *New Contact* in the address component. Both methods open a dialog in which to enter information about the contact.

In the *General* tab, enter basic contact information, such as name, e-mail addresses, and telephone numbers. Categories can be used to sort addresses. *Details* contains more specific information, such as birthday and spouse's name.

If your contact uses an instant messenger, you can add these identities in *IM Addresses*. If you do this and have Kopete or another KDE chat program running at the same time as Kontact, view status information about these identities in Kontact. In *Crypto Settings*, enter the contact's encryption data, such as public key.

Misc has additional information, such as a photograph and the location of the user's Free/Busy information. Use *Custom Fields* to add your own information to the contact or address book.

Contacts can also be imported in a variety of formats. Use *File* → *Import* and select the desired format. Then select the file to import.

12.4.2 Making a Distribution List

If you frequently send e-mail messages to the same group of people, a distribution list enables you to store multiple e-mail addresses as a single contact item so that you do not have to enter each name individually in every e-mail you send to that group. First, click *Settings* → *Show Extension Bar* → *Distribution List Editor*. In the new section that appears, click *New List*. Enter a name for the list then click *OK*. Add contacts to the list by dragging them from the address list and dropping them in the distribution list window. Use this list like you would an individual contact when creating an e-mail.

12.4.3 Adding Address Books

IMPORTANT: Groupware Address Books

The best way to add groupware resources is with the Groupware Wizard, a separate tool. To use it, close Kontact then run `groupwarewizard` in a command line or from the Office group of the KDE menu. Select the server type, such as SLOX, GroupWise, or Exchange, from the list offered then enter the address and authentication data. The wizard then adds the available resources to Kontact.

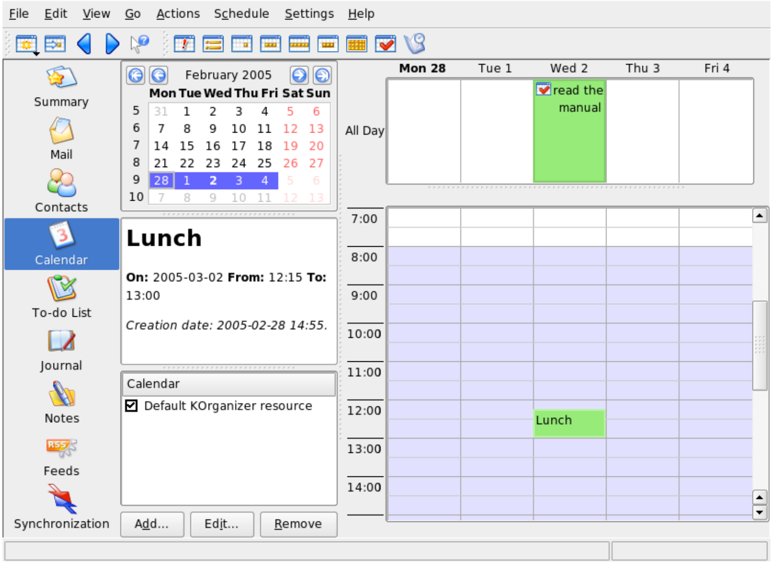
Kontact can access multiple address books, such as shared ones offered by Novell GroupWise or an LDAP server. Select *Settings* → *Show Extension Bar* → *Address Books* to view the current address books. Press *Add* to add one then select the type and enter the required information.

The check boxes in front of the books show the activation status of each address book. To prevent the display of a book without deleting it, uncheck it. *Remove* deletes the selected book from the list.

12.5 Calendar

Kontact uses KOrganizer as its calendar component. To configure it, use *Settings* → *Configure KOrganizer*. With the calendar, enter appointments and schedule meetings with others. If desired, you can be reminded of upcoming events. You can also import, export, and archive calendars with the options in *File*.

Figure 12.4 The Kontact Calendar



February 2005

	Mon	Tue	Wed	Thu	Fri	Sat	Sun
5	31	1	2	3	4	5	6
6	7	8	9	10	11	12	13
7	14	15	16	17	18	19	20
8	21	22	23	24	25	26	27
9	28	1	2	3	4	5	6
10	7	8	9	10	11	12	13

Lunch

On: 2005-03-02 **From:** 12:15 **To:** 13:00

Creation date: 2005-02-28 14:55.

Calendar

☑ Default KOrganizer resource

☑ read the manual

Lunch

12.5.1 Scheduling an Event

Add a new event or meeting with *Actions* → *New Event*. Enter the desired details. Under *Reminder*, specify the exact time (minutes, hours, or days in advance) when the attendees should be reminded of the event. If an event recurs, specify the appropriate interval. Another way to create an event at a specific point in the calendar is to double-click the corresponding field in one of the program's calendar views. This opens the same dialog window as that available from the menu. Alternatively, select a time range in the Calendar view and right-click.

Specify the attendees of an event by entering their data manually in the dialog or by inserting data from the address book. To enter data manually, select *New*. To import data from the address book, click *Select Addressee* then select the corresponding entries from the dialog. To schedule the event based on the participants' availability, go to *Free/Busy* and click *Pick Date*.

Use the *Recurrence* tab to configure an event that happens on a regular basis. *Attachments* can be convenient for linking other information with the event, such as an agenda for a meeting.

12.5.2 Adding Calendars

IMPORTANT: Groupware Calendars

The best way to add groupware resources is with Groupware Wizard, a separate tool. To use it, close Kontact then run `groupwarewizard` in a command line or from the Office group of the KDE menu. Select the server type, such as SLOX, GroupWise, or Exchange, from the list offered then enter the address and authentication data. The wizard adds the available resources to Kontact.

The calendar module can connect to multiple calendars simultaneously. This is useful, for example, to combine a personal calendar with an organizational one. To add a new calendar, click *Add* then select the calendar type. Complete the necessary fields.

The check boxes in front of the calendars show the activation status of each. To prevent the display of a calendar without deleting it, uncheck it. *Remove* deletes the selected calendar from the list.

12.6 Syncing Data with a Handheld

Kontact is designed so its data can be synced with handheld devices, such as a Palm. View information about the status of KPilot in the summary. Refer to Chapter 13, *Synchronizing a Handheld Computer with KPilot* (page 185) for information about configuring and using KPilot.

12.7 Kontact for GroupWise Users

If you are used to working in GroupWise, you should have very little trouble adjusting to Kontact. The two programs share many concepts and provide many of the same services. This section discusses notable terminology differences, as well as some tips to help GroupWise users make the most of Kontact.

12.7.1 Terminology Differences

The following table lists some key terminology differences between Kontact and GroupWise.

Table 12.1 Kontact and GroupWise Terminology Differences

GroupWise	Kontact
Appointments	Events
Busy search	Free/Busy
Notes	Journal entries
Posted, nonposted items	An event without attendees is posted. If an event has attendees, it is a Sent item.
Tasks	To-dos

12.7.2 Tips for GroupWise Users

This section contains hints to help GroupWise users work with some of the differences between GroupWise and Kontact.

Contact Information

You can add your GroupWise Messenger and e-mail contacts to your Kontact contact information. Then you can create an e-mail or open an instant messaging session with that contact by right-clicking the name in the Contact view.

Color Coding

It is helpful to color code GroupWise items, as well as items from other sources. Color coding makes it easy to scan your e-mails, contacts, and other information for items from a particular source.

Inviting Attendees to Events

Unlike GroupWise, Kontact does not automatically enter you as an attendee for events you schedule. Make sure that you remember to invite yourself.

12.8 For More Information

Kontact includes help for itself and its various components. Access it with *Help* → *Kontact Handbook*. The project's Web page, `http://www.kontact.org`, is also informative.

Synchronizing a Handheld Computer with KPilot

13

Handheld computers are in widespread use among users who need to have their schedules, to-do lists, and notes with them everywhere they go. Often users want the same data to be available both on the desktop and on the portable device. This is where KPilot comes in—it is a tool to synchronize data on a handheld with that used by the KDE applications KAddressBook, KOrganizer, and KNotes, which are part of Kontact.

The main purpose of KPilot is to allow sharing of data between the applications of a handheld computer and their KDE counterparts. KPilot does come with its own built-in memo viewer, address viewer, and file installer, but these cannot be used outside the KPilot environment. Independent KDE applications are available for all these functions except the file installer.

For communication between the handheld and the different desktop programs, KPilot relies on conduits. KPilot itself is the program that oversees any data exchange between the two computer devices. Using a particular function of the handheld on your desktop computer requires that the corresponding conduit is enabled and configured. For the most part, these conduits are designed to interact with specific KDE programs, so in general they cannot be used with other desktop applications.

The time synchronization conduit is special in that there is no user-visible program for it. It is activated in the background with each sync operation, but should only be enabled on computers that use a network time server to correct their own time drift.

When a synchronization is started, the conduits are activated one after another to carry out the data transfer. There are two different sync methods: a HotSync operation only synchronizes the data for which any conduits have been enabled while a backup operation performs a full backup of all data stored on the handheld.

Some conduits open a file during a sync operation, which means the corresponding program should not be running at the given time. Specifically, KOrganizer should not be running during a sync operation.

13.1 Conduits Used by KPilot

The conduits used by KPilot can be enabled and configured after selecting *Settings → Configure KPilot*. The following is a list of some important conduits:

Address Book
: This conduit handles the data exchange with the handheld's address book. The KDE counterpart for managing these contacts is KAddressBook. Start it from the main menu or with the command `kaddressbook`.

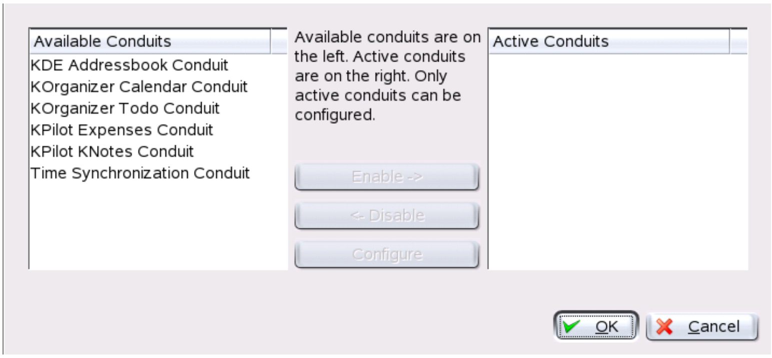
KNotes/Memos
: This conduit allows you to transfer notes created with KNotes to the handheld's memo application. Start the KDE application from the main menu or with the command `knotes`.

Calendar (KOrganizer)
: This conduit is responsible for syncing the appointments (events) of the hendheld. The desktop equivalent is KOrganizer.

ToDos (KOrganizer)
: This conduit is responsible for syncing to-do items. The desktop counterpart is KOrganizer.

Time Synchronization Conduit
: Enabling this conduit adjusts the handheld's clock to that of the desktop computer during each sync operation. This is only a good idea if the clock of the desktop computer itself is corrected by a time server at fairly frequent intervals.

Figure 13.1 *Configuration Dialog with the Available Conduits*



Available conduits are on the left. Active conduits are on the right. Only active conduits can be configured.

13.2 Configuring the Handheld Connection

To be able to use KPilot, first set up the connection with the handheld computer. The configuration depends on the type of cradle (docking unit) used with the handheld. There are two types of these: USB cradles or cables and serial cradles or cables.

13.2.1 Configuring the Connection from within KPilot

The easiest way to set up the connection is by using the configuration assistant. Select *Settings* → *Configuration Assistant* to start the assistant. In the first step, enter your username and the name of the device to which the handheld is connected. The assistant attempts to detect them itself if you select *Autodetect Handheld & Username*. If the autodetection fails, refer to Section 13.2.2, “Creating a /dev/pilot Link” (page 188).

After confirming with *Next*, the assistant prompts you to specify the applications that should be used for synchronization. You can choose among the KDE application suite (default), Evolution, and none. After selecting, close the window with *Finish*.

13.2.2 Creating a /dev/pilot Link

The setup of the connection with a serial handheld cradle is different from that of a USB cradle. Depending on which cradle is used, you may or may not need to create a symbolic link named /dev/pilot.

USB
Normally, a USB cradle is autodetected and there should be no need to create the symbolic link mentioned.

Serial
With a serial cradle, you need to know to which serial port it is actually connected. Serial devices are named /dev/ttyS?, starting from /dev/ttyS0 for the first port. To set up a cradle connected to the first serial port, enter the command:

```
ln -s /dev/ttyS0 /dev/pilot
```

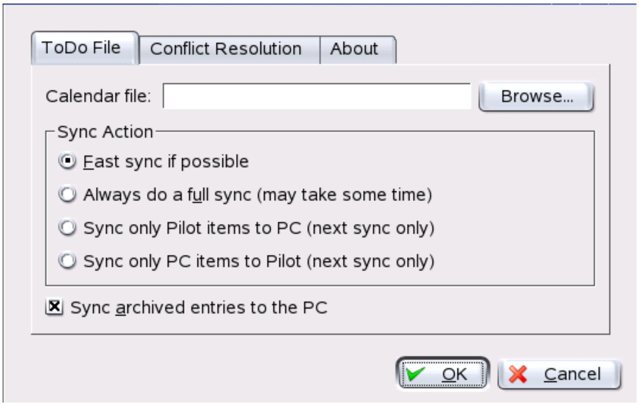
13.3 Configuring the KAddressBook Conduit

Initially, it should be sufficient to enable the KAddressBook conduit without changing any of the defaults. After the data has been synchronized for the first time, configure the details: what to do in case of conflicts, the way in which backup databases are saved, and how certain fields as stored on the handheld should be assigned to the fields expected by KAddressBook.

13.4 Managing To-Do Items and Events

On the KDE desktop, to-dos (tasks) and events (appointments) are managed with KOrganizer. Start the application from the main menu, with the command korganizer, or as part of Kontact. After enabling the calendar and the to-do conduit of KPilot, set some configuration options before using them.

Figure 13.2 *KPilot Configuration*



Calendar file:

Sync Action

● Fast sync if possible

○ Always do a full sync (may take some time)

○ Sync only Pilot items to PC (next sync only)

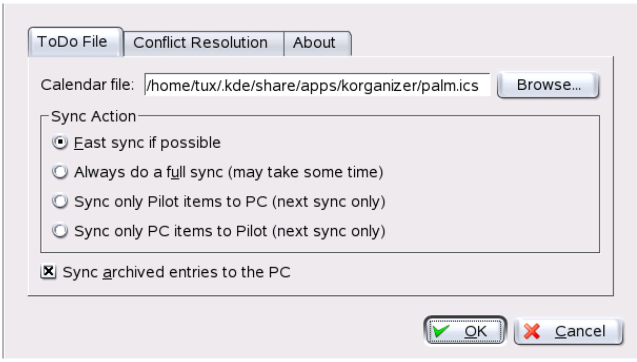
○ Sync only PC items to Pilot (next sync only)

☒ Sync archived entries to the PC

KOrganizer stores its files in the directory `~/.kde/share/apps/korganizer`. However, given that the directory `.kde/` begins with a dot, it may not be shown by the file selection dialog. In this case, enter the complete path manually or explicitly toggle the display of hidden files (dot files) in the file selection dialog. The default shortcut for this is F8.

After opening the directory `~/.kde/share/apps/korganizer`, select a file that can be used as a calendar file by KOrganizer. In this example, this is the file `palm.ics`. In the case of a user called `tux`, the complete path and filename would be `/home/tux/.kde/share/apps/korganizer/palm.ics`, as shown in Figure 13.3, "Dialog Showing the Path to a KOrganizer Calendar File" (page 189).

Figure 13.3 *Dialog Showing the Path to a KOrganizer Calendar File*



Calendar file: /home/tux/.kde/share/apps/korganizer/palm.ics

Sync Action

● Fast sync if possible

○ Always do a full sync (may take some time)

○ Sync only Pilot items to PC (next sync only)

○ Sync only PC items to Pilot (next sync only)

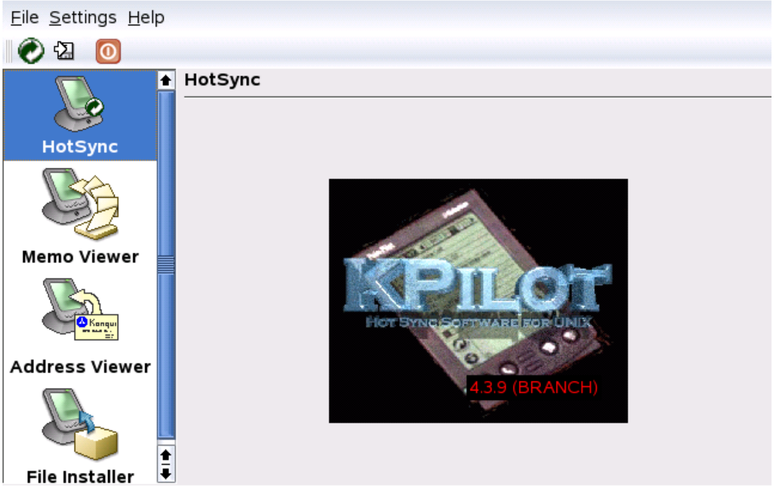
☒ Sync archived entries to the PC

KOrganizer should not be running when data is being exchanged with the handheld. Otherwise KPilot fails to carry out the sync operation.

13.5 Working with KPilot

Synchronizing the data of KDE applications with those of the handheld computer is quite easy. Simply start KPilot then press the HotSync button on the cradle or cable to initiate the sync operation.

Figure 13.4 *The Main Window of KPilot*



13.5.1 Backing Up Data from the Handheld

To do a full backup, select *File* → *Backup*. The backup is performed during the next sync operation. After that, switch back by selecting *File* → *HotSync* from the menu. Otherwise, the time-consuming full backup will be performed again during the next sync operation.

After a full backup, all copies of the handheld's programs and databases are found in `~/.kde/share/apps/kpilot/DBBackup/USERNAME`, where *`USERNAME`* is the name of the user registered on the handheld.

The two built-in KPilot viewers can be used for a quick lookup of addresses or memos, but they are not designed to actually manage this data. The KDE applications mentioned above are much more suited for these tasks.

13.5.2 Installing Programs on the Handheld

The *File Installer* module is an interesting and useful tool for the installation of handheld programs. These programs normally have the extension `.prc` and they are ready to start immediately after uploading them to the handheld. Before using such add-on programs, read their licenses as well as the instructions included.

13.5.3 Synchronizing Your Address Books and Calendars

To synchronize your calendars and addresses, use the KDE tools MultiSynK. Start the tool with the command `multisynk`. Create a Konnector pair before you synchronize your data. Go to *File* → *New* and select your Konnectors. Leave it with *Ok*.

The name is listed in the main window. To synchronize with your handheld computer go to *File* → *Sync*.

14

Using Beagle

Beagle is a search tool that indexes your personal information space to help you find whatever you are looking for. You can use Beagle to find documents, e-mails, Web history, Instant Messenger and ITC conversations, source code, images, music files, applications, and much more.

Beagle supports the following data sources:

- File system
- Application launchers
- Evolution mail and address book
- Gaim instant messaging logs
- Firefox Web pages (as you view them)
- Blam and Liferea RSS aggregators
- Tomboy notes

It also supports the following file formats:

- OpenOffice.org
- Microsoft Office (doc, ppt, xls)
- HTML

- PDF
- Images (jpeg, png)
- Audio (mp3, ogg, flac)
- AbiWord
- Rich Text Format (rtf)
- Texinfo
- Man pages
- Source code (C, C++, C#, Fortran, Java, JavaScript, Pascal, Perl, PHP, Python)
- Plain text

Beagle automatically indexes everything in your home directory, but you can choose to exclude certain files or directories. Beagle also includes a variety of tools that you can use to search your data.

14.1 Indexing Data

The Beagle daemon (`beagled`) automatically performs all indexing. By default, everything in your home directory is indexed. Beagle detects changes made to your home directory and reindexes the data accordingly.

- Files are immediately indexed when they are created, are reindexed when they are modified, and are dropped from the index when they are deleted.
- E-mails are indexed upon arrival.
- IM conversations are indexed as you chat, one line at a time.

Indexing your data requires a fair amount of computing power, but the Beagle daemon tries to be as unobtrusive as possible. It contains a scheduler that works to prioritize tasks and control CPU usage, based on whether you are actively using your workstation.

14.1.1 Preventing Files and Directories from Being Indexed

If you want to prevent a directory (and all of its subdirectories) from being indexed, create an empty file named `.noindex` and place it in the directory. You can add a list of files and directories to the `.noindex` file to prevent those files and directories from being indexed. Wild cards are permitted in the `.noindex` file.

You can also put a `.neverindex` file in your home directory with a list of files that should never be indexed. Wild cards are also allowed in this file. Use the same wild cards as you use for `glob` (for example, `f*le??.txt`). You can also use more powerful regular expressions by adding a forward slash both before and after your pattern (for example, `/file.*.txt/`). For more information, see the glob-UNIX Web site (`http://docs.python.org/lib/module-glob.html`).

14.1.2 Indexing Manually

Beagle has an effective system for determining when to index your files and it tries to not interfere with other applications you might be running. It intentionally times its indexing based on load and whether your system is idle, so as not to adversely affect your desktop experience. However, if you want to index your home directory right away, enter the following command in a terminal window before running Beagle:

```
export BEAGLE_EXERCISE_THE_DOG=1
```

14.1.3 Checking the Status of Your Index

Beagle includes the following commands to let you see the current indexing status:

`beagle-index-info`
Displays how many documents have been indexed and what type of documents have been indexed.

`beagle-status`
Displays the current work the Beagle daemon is doing (on an ongoing basis).

14.2 Searching Data

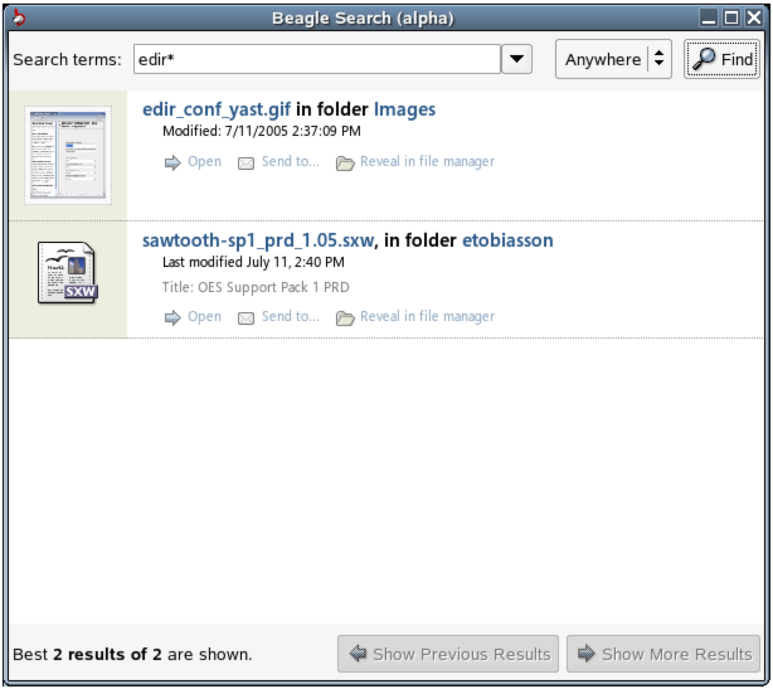
Beagle offers the following tools that let you search through the data that you have indexed.

14.2.1 Best

Best (Bleeding Edge Search Tool) is a graphical tool that searches through your indexed information. Best does not query the index directly; it passes the search terms to the Beagle daemon, which sends any matches back to Best. Best then renders the results and allows you to perform actions on the matching objects.

To open Best in KDE, click *K Menu* → *System* → *File System* → *Beagle Search*. In GNOME, click *Applications* → *System* → *File System* → *Beagle Search*.

Figure 14.1 *Beagle Search*



Search terms: edir*

Anywhere

edir_conf_yast.gif **in folder** Images
Modified: 7/11/2005 2:37:09 PM

sawtooth-sp1_prd_1.05.sxw**, in folder** etobiasson
Last modified July 11, 2:40 PM
Title: OES Support Pack 1 PRD

Best **2 results of 2** are shown.

To use Best, simply type your search text in the entry box at the top then press Enter or click *Find*. Best queries your indexed files and returns the results.

You can use the results list to open a file, mail a file, send an instant message, replay to a file, forward a file, or display a file in your file manager. The options available for each file depend on its type.

You can also use *Anywhere* to limit your search to files in a specific location, such as your address book or Web pages, or to display only a specific type of file in your results list.

14.2.2 beagle-query

Beagle has a command line tool you can use to search your Beagle index. To use this tool, enter the following command in a terminal window:

```
beagle-querysearch
```

Replace `search` with the text to find and the beagle-query tool returns results. You can use wild cards with this command.

Use `beagle-query --verbose search` to display detailed information about the search results.

Part V Graphics

Digital Cameras and Linux 15

Managing photos from your camera can be fun if you have the right tools. Linux offers several handy utilities for sorting and organizing your photographs. These include gphoto2, Konqueror, Digikam, and f-spot.

A comprehensive list of supported cameras is available at `http://www.gphoto.org/proj/libgphoto2/support.php`. If gphoto2 is installed, retrieve the list with the command `gphoto2 --list-cameras`. Get information about the available commands with `gphoto2 --help`.

TIP: Unsupported Cameras

If you do not find your camera in the list from gphoto, do not despair. It is very likely that your camera is supported as a USB mass storage device. Find more information in Section 15.2, “Accessing the Camera” (page 202).

15.1 Connecting to the Camera

The fastest and most convenient way to connect digital cameras to the computer is USB, provided the kernel, the camera, and the computer support it. The standard SUSE kernel provides this support. A suitable cable is also required.

Simply connect the camera to the USB port and turn on the camera. You may need to switch your camera to a special data transfer mode. For this procedure, consult the manual of your digital camera.

15.2 Accessing the Camera

There are three possibilities for accessing the pictures on the camera. It depends on your camera and which protocol it supports. Usually it is USB mass storage, which is handled by the hotplug system, or PTP (also known as PictBridge). Some camera models do not work with either protocol. To support these, gphoto2 includes specific drivers.

It is easiest if your camera supports USB mass storage. Read the documentation of your camera if you are unsure if this is possible. Some support two protocols, like both PTP and USB mass storage. Unfortunately, there are also some that communicate with a proprietary protocol, which can complicate the tasks. If your camera does not support USB mass storage or PTP, the following descriptions will not work. Try `gphoto2 --list-cameras` and the information at `http://www.gphoto.org/`.

If your camera can be switched to a USB mass storage device, select this option. After you connect it with the USB port of your computer and turn it on, it is detected by the hotplug system. This takes care of mounting the device automatically, so it is easily accessible. The KDE desktop shows a camera icon after a successful mount.

After the camera is successfully mounted, see a new directory under `/media`, beginning with `usb` and lots of numbers. Each vendor and product has a number, so when you connect a device on your computer it has always the same name. Depending on what you have connected to the USB bus, find different entries. The only problem left is to find the correct entry for your camera. Try to list one of these directories (`DCIM/`*`xxx`*) and see what happens. Each camera has a different tree structure, so there is no general rule. If you can see JPEG files in a directory, you probably found it.

After you find your correct directory, you can copy, move, or delete the files from your camera with a file manager, such as Konqueror, or simple shell commands (see Section 27.3, "Important Linux Commands" (page 393) and the *Reference*).

15.3 Using Konqueror

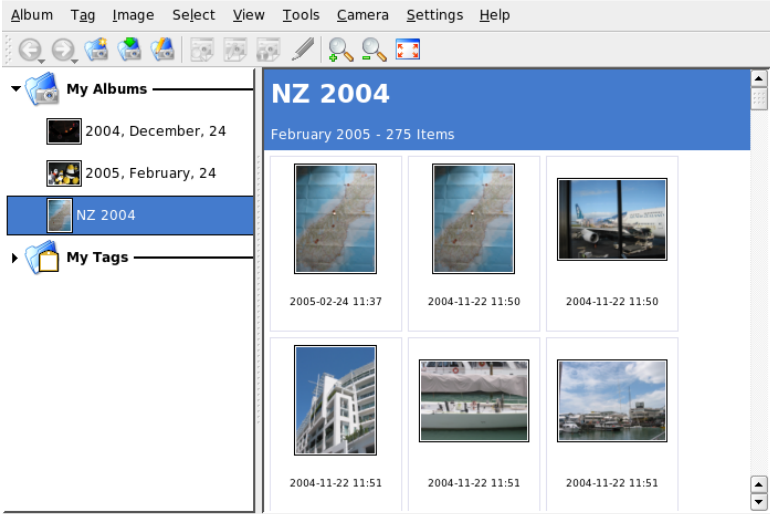
KDE users can easily access digital cameras by means of the familiar Konqueror interface. Connect your camera to the USB port. A camera icon should appear on the desktop. Click this icon to open the camera in Konqueror. The camera can also be accessed by entering the URL `camera:/` in Konqueror. Navigate through the camera's directory structure until the files are shown. Use the usual Konqueror file management features to copy the files as desired. More information about using Konqueror is available in Chapter 3, *The Web Browser Konqueror* (page 73).

15.4 Using Digikam

Digikam is a KDE program for downloading photographs from digital cameras. The first time it is run, Digikam asks where to store your photo album. If you enter a directory that already contains a collection of photographs, Digikam treats each subfolder as an album.

On start-up, Digikam presents a window with two sections: your albums are displayed to the left and the photographs of the current album are displayed to the right. See Figure 15.1, "The Main Window of Digikam" (page 203).

Figure 15.1 *The Main Window of Digikam*



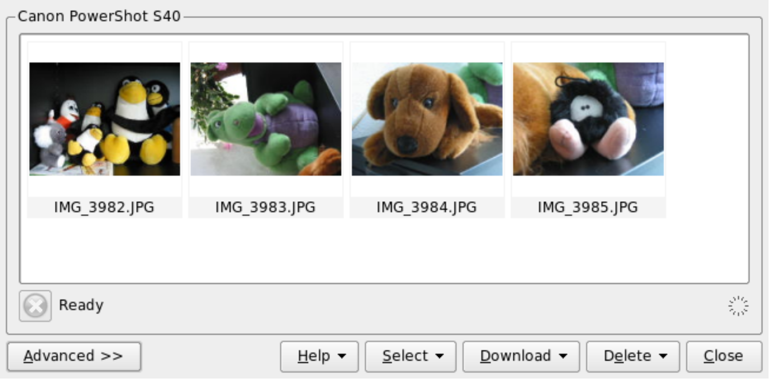
15.4.1 Configuring Your Camera

To set up a camera in Digikam, select *Camera* → *Add Camera*. First, try to autodetect the camera with *Auto-Detect*. If this fails, browse the list for your model with *Add*. If your camera model is not included in the list, try an older model or use *USB/IEEE mass storage camera*. Confirm with *Ok*.

15.4.2 Downloading Pictures from Your Camera

After your camera has been configured correctly, connect to your camera with the *Camera* menu and the name that you gave in the dialog from Section 15.4.1, "Configuring Your Camera" (page 203). Digikam opens a window and begins to download thumbnails and displays them as in Figure 15.2, "Downloading Pictures from Camera" (page 204). Right-click an image to open a pop-up menu with the options to *View*, display some *Properties* or *EXIF Information*, *Download*, or *Delete* the image. With *Advanced*, select renaming options and how the camera-provided information (EXIF) should be handled.

Figure 15.2 *Downloading Pictures from Camera*



The renaming options can be very convenient if your camera does not use meaningful filenames. You can let Digikam rename your photographs automatically. Give a unique prefix and, optionally, a date, time, or sequence number. The rest is done by Digikam.

Select all photographs to download from the camera by pressing the left mouse button or clicking individual photographs with Ctrl pressed. Selected photographs appear with inverted colors. Click *Download*. Select the destination from the list or by creating a new album with *New Album*. This automatically suggests a filename with the current date. Confirm with *Ok* to start the download process.

15.4.3 Getting Information

Getting information about the photograph is not difficult. A short summary is displayed as a tool tip if you point with the mouse cursor at the thumbnail. For longer information, right-click the photograph and choose *Properties* from the menu. A dialog box opens with three tabs, *General*, *EXIF*, and *Histogram*.

General lists the name, type, owner, and some other basic information. The more interesting part is the *EXIF* tab. The camera stores some metadata for each photograph. Digikam reads these properties and displays them in this list. Find the exposure time, pixel dimensions, and others. To get more information for the selected list entry, press Shift + F1. This shows a small tool tip. The last tab, *Histogram*, shows some statistical information.

15.4.4 Managing Albums

Digikam inserts a *My Albums* folder by default, which collects all your photographs. You can store these into subfolders later. The albums can be sorted by their directory layout, by the collection name that has been set in the album properties or by the date that the albums were first created (this date can also be changed in the properties of each album).

To create a new album, you have some possibilities:

- Uploading new photographs from the camera
- Creating a new album by clicking the *New Album* button in the toolbar
- Importing an existing folder of photographs from your hard disk (select *Album → Import → Import Folders*)
- Right-clicking *My Albums* and selecting *New Album*

After selecting to create an album in your preferred way, a dialog box appears. Give your album a title. Optionally, choose a collection, insert some comments, and select an album date. The collection is a way of organizing your albums by a common label. This label is used when you select *View → Sort Albums → By Collection*. The comment is shown in the banner at the top of the main window. The album date is used when you select *View → Albums → By Date*.

Digikam uses the first photograph in the album as the preview icon in the *My Albums* list. To select a different one, right-click the respective photograph and select *Set as Album Thumbnail* from the context menu.

15.4.5 Managing Tags

Managing lots of different photographs with different albums can sometimes be complex. To organize individual photographs, Digikam provides the *My Tag* system.

For example, you have photographed your friend John at different times and you want to collect all images, independent of your album. This let you find all photographs very easily. First, create a new tag by clicking *My Tags → People*. From the context menu, choose *New Tag*. In the dialog box that appears, enter *John* as title and optionally set an icon. Confirm with *Ok*.

After creating your tag, assign it to the desired pictures. Go to each album and select the respective photographs. Right-click and choose *Assign Tag → People → John* from the menu that appears. Alternativly, drag the photographs to the tag name under *My Tags* and drop them there. Repeat as necessary with other albums. View all the images by clicking *My Tags → People → John*. You can assign more than one tag to each photograph.

Editing tags and comments can be tedious. To simplify this task, right-click a photograph and select *Edit Comments & Tags*. This opens a dialog box with a preview, a comment field, and a tag list. Now you can insert all the needed tags and add a comment. With *Forward* and *Back*, navigate in your album. Store your changes with *Apply* and leave with *Ok*.

15.4.6 Exporting Image Collections

Digikam provides several export options that help you archive and publish your personal image collections. It offers archiving to CD or DVD (via k3b), HTML export, and export to a remote gallery.

To save your image collection to CD or DVD, proceed as follows:

1. Select *File → Export → Archive to CD/DVD*.

2. Make your adjustments in the *Create CD/DVD Archive* dialog using its various submenus. After that, click *OK* to initiate the burning process.

   a. *Selection*: Determine which part of your collection should be archived by selecting albums and tags.

   b. *HTML Interface*: Decide whether your image collection should be accessible via an HTML interface and whether autorun functionality should be added to your CD/DVD archive. Set a selection title and image, font, and background properties.

   c. *Media Volume Descriptor*: Change the settings for volume description, if necessary.

   d. *Media Burning*: Adjust the burning options to your needs, if necessary.

To create an HTML export of your image collection, proceed as follows:

1. Select *File* → *Export* → *HTML Export*.

2. Adjust the settings in *Create Image Galleries* to your needs, using the various submenus. When you are done, click *OK* to initiate the gallery creation.

   a. *Selection*: Determine which part of your collection should be archived by selecting albums and tags.

   b. *Look*: Set the title and appearance of your HTML gallery.

   c. *Album*: Determine the location of the gallery on disk as well as image size, compression, format, and the amount of metadata displayed in the resulting gallery.

   d. *Thumbnails*: As with the target images, specify size, compression and file type for the thumbnails used for gallery navitation.

To export your collection to an external image gallery on the Internet, proceed as follows:

1. Get an account for an external web site holding your gallery.

2. Select *File* → *Export* → *Export to Remote Gallery* and provide URL, username, and password for the external site when asked for them.

   Digikam establishes a connection to the site specified and opens a new window called *Gallery Export*.

3. Determine the location of your new album inside the gallery.

4. Click *New Album* and provide the information requested by Digikam.

5. Upload the images to the new album with *Add Photos*.

15.4.7 Useful Tools

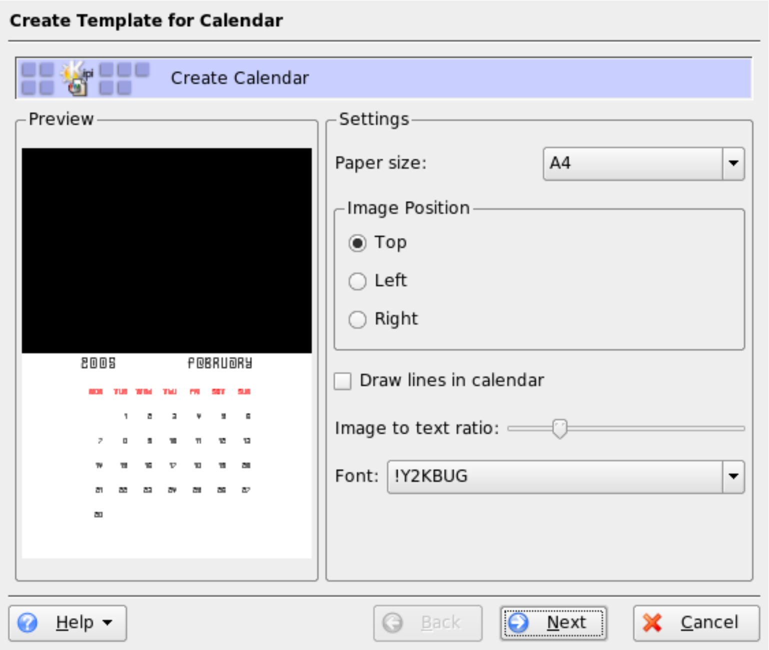
Digikam provides several tools to simplify some tasks. Find them in the *Tools* menu. The following is a small selection of the available tools.

Creating a Calendar

If you want to please someone, a custom calendar can be a nice gift. Go to *Tools* → *Create Calendar*, which opens a wizard dialog like that in Figure 15.3, "Creating a Template for a Calendar" (page 209).

Customize the settings (paper size, image position, font, etc.) and confirm with *Next*. Now you can enter the year and select the images to use. After clicking *Next* again, see a summary. The final *Next* opens the KDE Printer dialog. Here, decide if you want to see a preview, save as PDF, or just print directly.

Figure 15.3 *Creating a Template for a Calendar*



Create Template for Calendar

Create Calendar

Paper size: A4

Image Position

● Top

○ Left

○ Right

□ Draw lines in calendar

Image to text ratio:

Font: !Y2KBUG

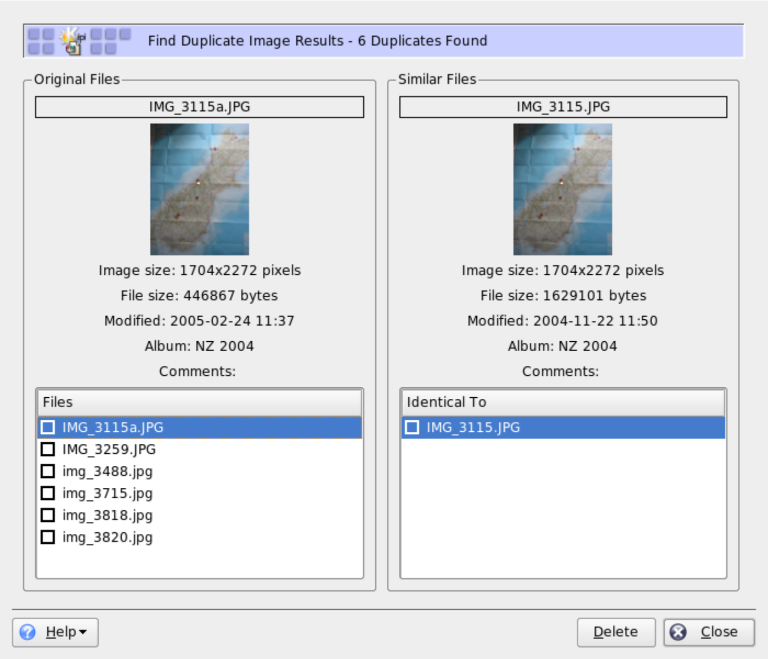
Finding Duplicate Photographs

Sometimes you photograph similar scenes repeatedly and want to keep only the best shots. This is the perfect task for the *Find Duplicate* plug-in.

Go to *Tools* → *Find Duplicate Images*. Select the albums or tags to handle. Under *Method & Cache*, choose the search method: a more accurate or a faster method. After you confirm with *Ok*, Digikam proceeds with the investigation.

If it finds some duplicates, it shows the result in a window like Figure 15.4, “Results of Find” (page 210). Decide which images to delete by activating the desired check boxes then clicking *Delete*. Leave the window with *Close*.

Figure 15.4 *Results of Find*



Find Duplicate Image Results - 6 Duplicates Found

Original Files

IMG_3115a.JPG

Image size: 1704x2272 pixels
File size: 446867 bytes
Modified: 2005-02-24 11:37
Album: NZ 2004
Comments:

Files

- □ IMG_3115a.JPG
- □ IMG_3259.JPG
- □ img_3488.jpg
- □ img_3715.jpg
- □ img_3818.jpg
- □ img_3820.jpg

Similar Files

IMG_3115.JPG

Image size: 1704x2272 pixels
File size: 1629101 bytes
Modified: 2004-11-22 11:50
Album: NZ 2004
Comments:

Identical To

- □ IMG_3115.JPG

Batch Processes

Digikam also provides some batch processes that perform a specific task on lots of files. This can be renaming, converting, resizing, and much more. Find them under *Tools* → *Batch Processes*.

15.4.8 Basic Image Viewing and Editing with Digikam

Digikam includes its own lean image viewing and editing program. It automatically opens if you double-click an image's thumbnail.

Use this tool to do some basic image editing on the images you just downloaded from your camera. You can crop, rotate or flip the image, do some basic color adjustments, apply various colored filters (for example, to export a colored image to black and white), and efficiently reduce red eyes in portrait shots.

The most important menus are:

Image

Use *Edit Comments & Tags* to enter comments to a particular image and to assign a tag (category) to this image. *Properties* takes you to a window consisting of three tabs providing general information, EXIF information, and the histogram of this image.

Fix

This menu contains some of the editing functions most needed in digital photography. *Colors* takes you to a submenu where you can modify all basic color settings. You can also blur or sharpen either the entire picture or just a part of the image you selected. To reduce red eyes in a portrait shot, roughly select the eye region of the face by just clicking and holding the left mouse pointer and gradually expanding the selection, select *Red Eye Reduction* and choose either mild or aggressive reduction depending on whether you selected a whole region or just the eyes.

Transform

The *Transform* menu offers the crop, rotate, flip, and resize functions. You can also use the *Aspect Ratio Crop* option to produce crops in a fixed aspect ratio.

Filters

If you need to transform your color shots into black and white or want to achieve an aged look in your photographs, check out the *Filters* menu and choose from the various export options.

A more detailed description of this tool can be found in Digikam's online help in *digiKam Image Editor*, which can be reached with the *Help* button in Digikam's menu bar.

TIP: Advanced Image Processing

Professional image editing can be done with the GIMP. More information about The GIMP can be found in Chapter 17, *Manipulating Graphics with The GIMP* (page 225).

15.5 Using f-spot

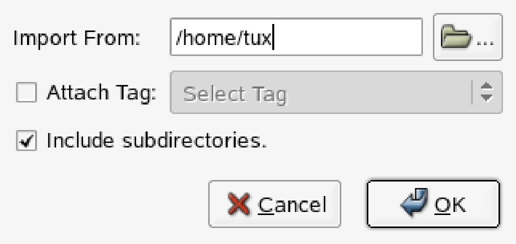
f-spot is a managment tool for your collection of digital images tailored for the GNOME desktop. It allows you to assign different tags to your images in order to categorize them and offers various neat image editing options.

The first time you run f-spot, tell it where to find the images to import to your f-spot collection. If you already have a collection of images stored on your hard drive, enter the path to the respective directory and optionally include subfolders. f-spot imports these images into its database.

TIP: Tagging Images on Import

If all the images you are importing belong to the same category, you can attach the appropriate tag on import. Select *Attach Tag* and choose the matching tag from the drop down menu.

Figure 15.5 *Importing Images to f-spot*



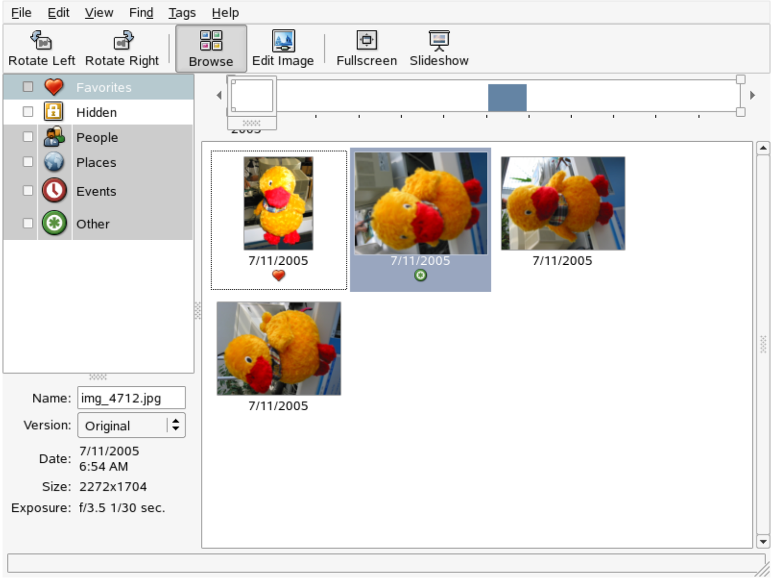
Import From: /home/tux

□ Attach Tag: Select Tag

☑ Include subdirectories.

f-spot's main window is divided into three main areas. Categories, tags, and detailed information for the selected images are displayed in a sidebar to the left and a thumbnails of all images bearing the selected tag or category or, if none is selected, the entire collection is displayed in the right part of the window.

Figure 15.6 *f-spot's Main Window*



Name: img_4712.jpg
Version: Original
Date: 7/11/2005 6:54 AM
Size: 2272x1704
Exposure: f/3.5 1/30 sec.

A menu bar right at the top of the window allows you to access the main menus. A toolbar below offers several different functions depicted by a matching icon:

Rotate (Left or Right)
Use this shortcut to change an image's orientation.

Browse
The *Browse* mode allows you to view and search you entire collection or tagged subsets of it. You can also use the time line to search images by creation date.

Edit Image
This mode allows you to select one image and do some basic image processing. Details are available in Section 15.5.6, "Basic Image Processing with f-spot" (page 217).

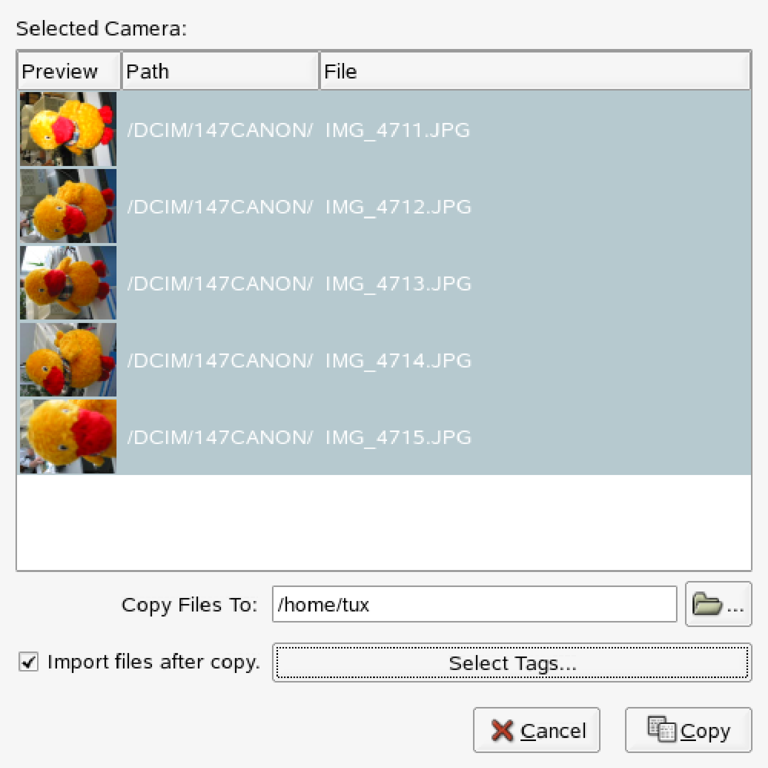
Fullscreen
Switch to fullscreen display mode.

Slideshow
Start a slide show.

15.5.1 Downloading Pictures From Your Camera

Import new images from your digital camera connected to the USB port of your computer using *File → Import from Camera*. The type of camera is detected automatically.

Figure 15.7 *Import from Camera*



Selected Camera:

Preview	Path	File
	/DCIM/147CANON/	IMG_4711.JPG
	/DCIM/147CANON/	IMG_4712.JPG
	/DCIM/147CANON/	IMG_4713.JPG
	/DCIM/147CANON/	IMG_4714.JPG
	/DCIM/147CANON/	IMG_4715.JPG

Copy Files To: /home/tux

☑ Import files after copy.

f-spot launches a preview window displaying all the images that are available for download from camera. The files are copied to the target directory specified via *Copy Files to*. If *Import files after copy* is selected, all images copied from camera are automatically imported to f-spot's database. Tagging can be done on import, if you select the appropriate tag with *Select Tags*. If you do not want to import all images on your camera to your database, just deselect the unwanted one in the preview window.

15.5.2 Getting Information

Once you select an image, some basic statistical information on this image is displayed in the lower left part of the window. This includes the filename, its version (copy or original image), the date of creation, its size and the exposure which was used in creating this particular image. View the EXIF data associated with the image file with *View → EXIF Data*.

15.5.3 Managing Tags

Use tags to categorize any of your images to create manageable subsets of your collection. If, for example, you would like to get some sort of order in your collection of portrait shots of your loved ones, proceed like this:

1. Select the *Browse* mode of f-spot.

2. In the left frame of the f-spot window, select the *People* category, right-click it, then choose *Create New Tag*. The new tags then appear as subcategories below the *People* category:

   a. Create a new tag called `Friends`.

   b. Create a new tag called `Family`.

3. Now attach tags to images or groups of selected images. Right-click an image, choose *Attach Tag*, and select the appropriate tag for this image. To attach a tag to a group of images, click the first one then press Shift and select the other ones without releasing the Shift key. Right-click for the tag menu and select the matching category.

After the images have been categorized, you can browse your collection by tag. Just check *People → Family* and the displayed collection is limited to the images tagged `Family`. Searching your collection by tag is also possible through *Find → Find by Tag*. The result of your search is displayed in the thumbnail overview window.

Removing tags from single images or groups of images works similarly to attaching them. The tag editing functions are also accessible via the *Tags* menu in the top menu bar.

15.5.4 Search and Find

As mentioned in Section 15.5.3, “Managing Tags” (page 215), tags can be used as a means to find certain images. Another way, which is quite unique to f-spot, is to use the *Timeline* below the toolbar. By dragging the little frame along this time line, limit the images displayed in the thumbnail overview to those taken in the selected time frame. f-spot starts with a sensibly chosen default time line, but you can always edit the time span by moving the sliders to the right and left ends of the time line.

15.5.5 Exporting Image Collections

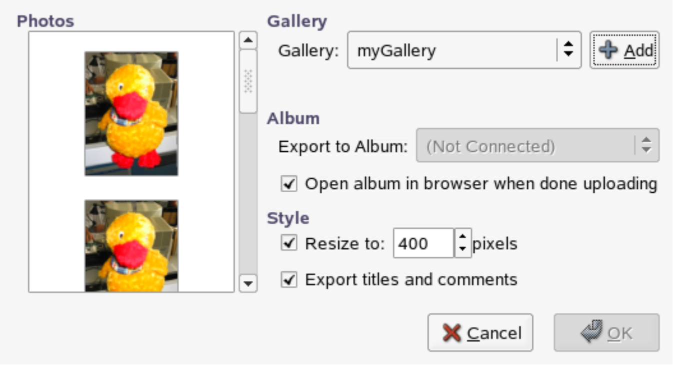
f-spot offers a range of different export functions for your collections under *File* → *Export*. Probably the most often used of these are *Export to Web Gallery* and *Export to CD*.

To export a selection of images to a web gallery, proceed as follows:

1. Select the images to export.

2. Click *File* → *Export* → *Export to Web Gallery* and select a gallery to which to export your images or add a new one. f-spot establishes a connection to the Web location entered for your web gallery. Select the album to which to export the images and decide whether to scale the images automatically and export titles and comments.

Figure 15.8 *Exporting Images to a Web Gallery*



Gallery: myGallery

Export to Album: (Not Connected)

☑ Open album in browser when done uploading

☑ Resize to: 400 pixels

☑ Export titles and comments

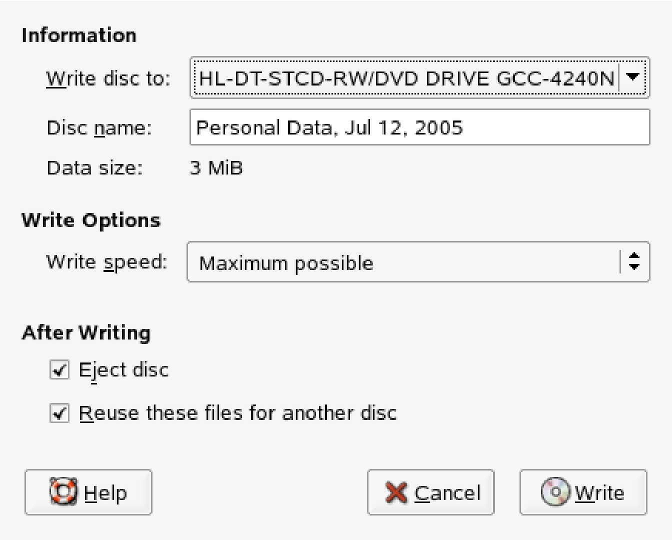
To export a selection of images to CD, proceed as follows:

1. Select the images to export.

2. Click *File* → *Export* → *Export to CD* and click *OK*.

   f-spot copies the files and opens the CD writing dialog. Assign a name to your image disk and determine the writing speed. Click *Write* to start the CD writing process.

Figure 15.9 *Exporting Images to CD*



Information

Write disc to: HL-DT-STCD-RW/DVD DRIVE GCC-4240N

Disc name: Personal Data, Jul 12, 2005

Data size: 3 MiB

Write Options

Write speed: Maximum possible

After Writing

☑ Eject disc

☑ Reuse these files for another disc

15.5.6 Basic Image Processing with f-spot

f-spot offers several very basic image editing functionalities. Enter the edit mode of f-spot by clicking the *Edit Image* icon in the toolbar or by double-clicking the image to edit. Switch images using the arrow keys at the bottom right. Choose from the following edit functions:

Sharpen
: Access this function with *Edit* → *Sharpen*. Adjust the values for *Amount*, *Radius*, and *Threshold* to your needs and click *OK*.

Crop Image
To crop the image to a selection you made, either choose a fixed ratio crop or the *No Constraint* option from the drop-down menu at the bottom left, select the region to crop, and click the scissor icon next to the ratio menu.

Red Eye Reduction
In a portrait shot, select the eye region of the face and click the red eye icon.

Adjust Color
View the histogram used in the creation of the shot and correct exposure and color temperature if necessary.

TIP: Advanced Image Processing

Professional image editing can be done with the GIMP. More information about The GIMP can be found in Chapter 17, *Manipulating Graphics with The GIMP* (page 225).

15.6 For More Information

For more information about using digital cameras with Linux, refer to the following Web sites:

- `http://digikam.sourceforge.net/`—Information about Digikam
- `http://www.gphoto.org`—Information about gPhoto2
- `http://www.gphoto.org/proj/libgphoto2/support.php`—A comprehensive list of supported cameras
- `http://www.thekompany.com/projects/gphoto/`—Information about Kamera, a KDE front-end for gPhoto2

Kooka—A Scanning Application 16

Kooka is a KDE application for scanning. This chapter explains the user interface and the functionality of the application. In addition to creating image files from printed media, like photographs or magazines, Kooka has character recognition capabilities. This means it can help convert written text to a text file that can be edited.

Start Kooka from the main menu or enter the command `kooka`. When started, Kooka opens a three-frame window with a menu bar to the upper left and a toolbar directly below it. All windows can be freely readjusted or rearranged with the mouse. It is also possible to completely detach single frames from the Kooka window for deliberate placement on the desktop. To move the frames, click and drag the thin double line right above the frame. Any frame, except the main window, can be placed within any other frame aligned to the left, right, top, bottom, or center. Centered windows have the same size, are stacked, and can be brought to the foreground with tabs.

The *Image Viewer* and the *Scan Preview* frames share a window by default. Tabs allow switching between them. The left frame provides the gallery. This is a small file browser for accessing the scanned images. The frame to the lower right is shared by OCR (optical character recognition) and the thumbnails, which can be loaded into the image viewer with a simple click of the mouse. See Figure 16.1, “The Kooka Main Window” (page 220).

Figure 16.1 *The Kooka Main Window*

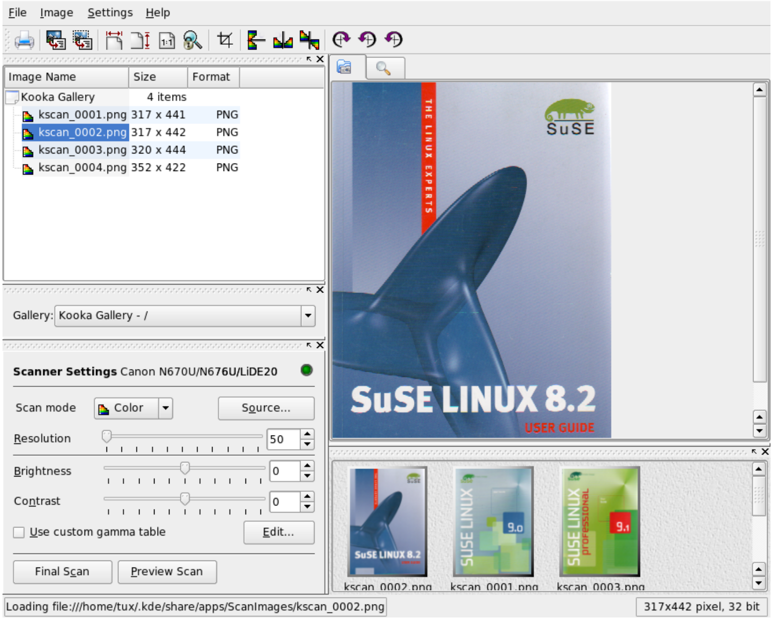


Image Name	Size	Format
Kooka Gallery	4 items	
kscan_0001.png	317 x 441	PNG
kscan_0002.png	317 x 442	PNG
kscan_0003.png	320 x 444	PNG
kscan_0004.png	352 x 422	PNG

Gallery: Kooka Gallery - /

Scanner Settings Canon N670U/N676U/LiDE20

Scan mode: Color

Resolution: 50

Brightness: 0

Contrast: 0

□ Use custom gamma table

Loading file:///home/tux/.kde/share/apps/ScanImages/kscan_0002.png

317x442 pixel, 32 bit

16.1 The Preview

A preview should always be created when the object to scan is smaller than the total scanning area. Set a few parameters to the left of the preview frame. Select the scanning size with *Custom* or one of the standard formats. See Figure 16.2, “The Kooka Preview Window” (page 221). The *Custom* setting is the most flexible, because it allows selection of the desired area with the mouse. Once the settings have been made, request the preview of the image to scan by clicking *Preview Scan* in *Scan Parameters*.

***Figure 16.2** The Kooka Preview Window*

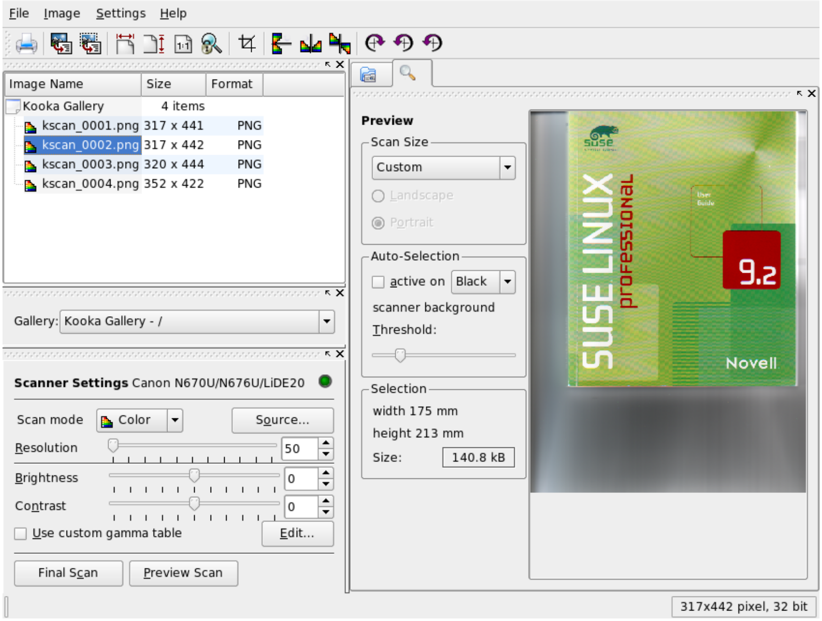


Image Name	Size	Format
Kooka Gallery	4 items	
kscan_0001.png	317 x 441	PNG
kscan_0002.png	317 x 442	PNG
kscan_0003.png	320 x 444	PNG
kscan_0004.png	352 x 422	PNG

Gallery: Kooka Gallery - /

Scanner Settings Canon N670U/N676U/LiDE20

Scan mode: Color

Resolution: 50

Brightness: 0

Contrast: 0

□ Use custom gamma table

Scan Size: Custom

○ Landscape

● Portrait

Auto-Selection: □ active on Black scanner background

Threshold:

Selection: width 175 mm, height 213 mm

Size: 140.8 kB

317x442 pixel, 32 bit

16.2 The Final Scan

If you selected *Custom* for the scanning size, use the mouse to select the rectangular area to scan. The selected area is confined by a dotted border.

Choose between color and black-and-white scanning and set the resolution with the slider. See Figure 16.3, “The Kooka Scanning Parameters” (page 222). The higher the resolution, the better the quality of the scanned image is. However, this also results in a correspondingly larger file and the scanning process can take a very long time at high resolutions. Activate *Use custom gamma table* and click *Edit* to change the settings for brightness, contrast, and gamma.

Figure 16.3 *The Kooka Scanning Parameters*

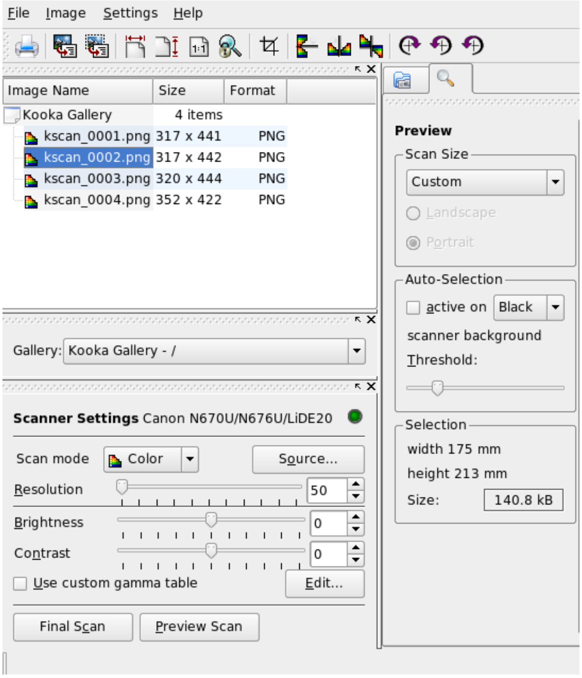


Image Name	Size	Format
Kooka Gallery	4 items	
kscan_0001.png	317 x 441	PNG
kscan_0002.png	317 x 442	PNG
kscan_0003.png	320 x 444	PNG
kscan_0004.png	352 x 422	PNG

Gallery: Kooka Gallery - /

Scanner Settings Canon N670U/N676U/LiDE20

Scan mode: Color
Resolution: 50
Brightness: 0
Contrast: 0
□ Use custom gamma table

Preview

Scan Size: Custom
○ Landscape
● Portrait

Auto-Selection
□ active on Black
scanner background
Threshold:

Selection
width 175 mm
height 213 mm
Size: 140.8 kB

Once all settings have been made, click *Final Scan* to scan the image. The scanned image is then displayed in the image viewer and as a thumbnail. When prompted, select the format in which to save the image. To save all the future images in that same format, check the corresponding box. Confirm with *OK*.

16.3 The Menus

Some of the functions of the toolbar are also available in the *File* and *Image* menus. Modify preference settings for Kooka in *Settings*.

File

Use this menu to start the KPrinter printing assistant, create a new folder for your images, and save, delete, and close files. The OCR results of a scanned text document can be saved here. Also use this menu to close Kooka.

Image

The *Image* menu allows starting a graphics application for postprocessing or optical character recognition of an image. The recognized text from an OCR operation is displayed in its own frame. Various tools for scaling, rotating, and flipping an image are available. These functions can also be accessed from the toolbar. *Create From Selection* allows saving an area of an image previously marked with the mouse.

Settings

Settings adjusts of the look and feel of Kooka. The toolbar and status bar can be switched on and off and keyboard shortcuts for menu entries can be defined. *Configure Toolbars* provides a list of all the functions available to the toolbar. *Configure Kooka* opens a configuration dialog in which to modify the look and feel of Kooka. Normally, however, the defaults are sufficient. In *Tool Views*, enable and disable the thumbnail viewer, the preview, the gallery, the scanning parameters, and the OCR result window.

Help

The *Help* menu provides access to the online help manual for Kooka. Also use it to access a feedback channel for problems and wishes. It also provides information about the version, authors, and license of Kooka and KDE.

16.4 The Gallery

The gallery window shows the default folder where Kooka stores all its image files. An example is shown in Figure 16.4, "The Kooka Gallery" (page 223). To save an image to your personal home directory, click the thumbnail to select it then select *File* → *Save Image*. Then enter your personal home directory and give the file a descriptive name.

Figure 16.4 *The Kooka Gallery*

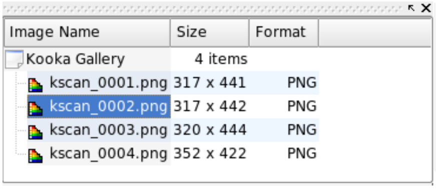


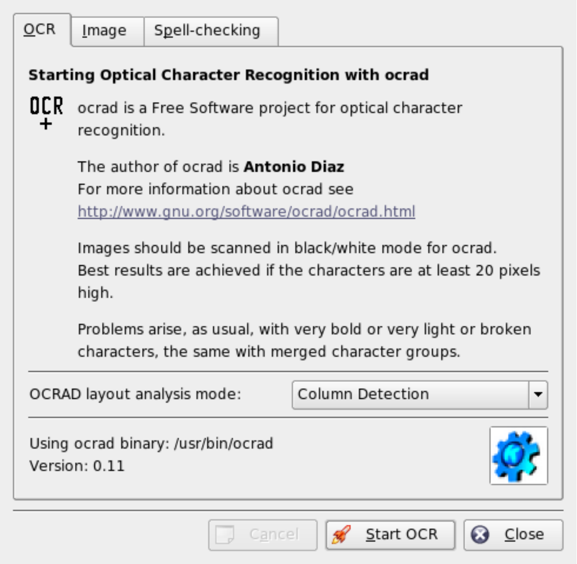
Image Name	Size	Format
Kooka Gallery	4 items	
kscan_0001.png	317 x 441	PNG
kscan_0002.png	317 x 442	PNG
kscan_0003.png	320 x 444	PNG
kscan_0004.png	352 x 422	PNG

To add images to the gallery, simply drag and drop them from Konqueror. Start Konqueror, navigate to the directory containing the images to add to the gallery, and drag them with the mouse to a folder of the Kooka gallery.

16.5 Optical Character Recognition

If the character recognition module is installed, documents can be scanned in *lineart* mode, saved in the proposed format, then processed for text recognition from the *Image* menu. Process the entire document or only a previously selected area. A configuration dialog tells the module whether the original text is in printed type, handwriting, or standardized type. Also set the language so the module can process the document correctly. See Figure 16.5, "OCR with Kooka" (page 224).

Figure 16.5 *OCR with Kooka*



Starting Optical Character Recognition with ocrad

ocrad is a Free Software project for optical character recognition.

The author of ocrad is **Antonio Diaz**
For more information about ocrad see
http://www.gnu.org/software/ocrad/ocrad.html

Images should be scanned in black/white mode for ocrad. Best results are achieved if the characters are at least 20 pixels high.

Problems arise, as usual, with very bold or very light or broken characters, the same with merged character groups.

OCRAD layout analysis mode: Column Detection

Using ocrad binary: /usr/bin/ocrad
Version: 0.11

Switch to the *OCR Result Text* window and check the text, which may need to be proofread. To do this, save the text with *File* → *Save OCR Result Text*. The text can then be processed with OpenOffice.org or KWrite.

Manipulating Graphics with The GIMP 17

The GIMP (*The GNU Image Manipulation Program*) is a program for creating and editing pixel graphics. In most aspects, its features are comparable to those of Adobe Photoshop and other commercial programs. Use it to resize and retouch photographs, design graphics for Web pages, make covers for your custom CDs, or almost any other graphics project. It meets the needs of both amateurs and professionals.

Like many other Linux programs, The GIMP is developed as a cooperative effort of developers worldwide who volunteer their time and code to the project. The program is under constant development, so the version included in your SUSE Linux may vary slightly from the version discussed here. The layout of the individual windows and window sections is especially likely to vary.

The GIMP is an extremely complex program. Only a small range of features, tools, and menu items are discussed in this chapter. See Section 17.6, “For More Information” (page 232) for ideas of where to find more information about the program.

17.1 Graphics Formats

There are two main formats for graphics—pixel and vector. The GIMP works only with pixel graphics, which is the normal format for photographs and scanned images. Pixel graphics consist of small blocks of color that together create the entire image. The files can easily become quite large because of this. It is also not possible to increase the size of a pixel image without losing quality.

Unlike pixel graphics, vector graphics do not store information for all individual pixels. Instead, they store information about how image points, lines, or areas are grouped together. Vector images can also be scaled very easily. The drawing application of OpenOffice.org, for example, uses this format.

17.2 Starting GIMP

Start GIMP from the main menu. Alternatively, enter `gimp &` in a command line.

17.2.1 Initial Configuration

When starting GIMP for the first time, a configuration wizard opens for preparatory configuration. The default settings are acceptable for most purposes. Press *Continue* in each dialog unless you are familiar with the settings and prefer another setup.

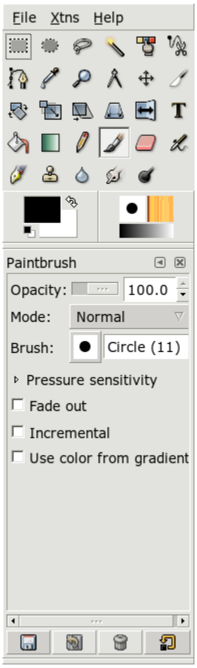
17.2.2 The Default Windows

Three windows appear by default. They can be arranged on the screen and, except the toolbox, closed if no longer needed. Closing the toolbox closes the application. In the default configuration, GIMP saves your window layout when you exit. Dialogs left open reappear when you next start the program.

The Toolbox

The main window of GIMP, shown in Figure 17.1, “The Main Window” (page 227), contains the main controls of the application. Closing it exits the application. At the very top, the menu bar offers access to file functions, extensions, and help. Below that, find icons for the various tools. Hover the mouse over an icon to display information about it.

Figure 17.1 *The Main Window*



Opacity: 100.0

Mode: Normal

Brush: Circle (11)

▸ Pressure sensitivity

□ Fade out

□ Incremental

□ Use color from gradient

The current foreground and background color are shown in two overlapping boxes. The default colors are black for the foreground and white for the background. Click the box to open a color selection dialog. Swap the foreground and background color with the bent arrow symbol to the upper right of the boxes. Use the black and white symbol to the lower left to reset the colors to the default.

To the right, the current brush, pattern, and gradient are shown. Click the displayed one to access the selection dialog. The lower portion of the window contains allows configuration of various options for the current tool.

Layers, Channels, Paths, Undo

In the first section, use the drop-down box to select the image to which the tabs refer. By clicking *Auto*, control whether the active image is chosen automatically. By default, *Auto* is enabled.

Layers shows the different layers in the current images and can be used to manipulate the layers. *Channels* shows and can manipulate the color channels of the image.

Paths are an advanced method of selecting parts of an image. They can also be used for drawing. *Paths* shows the paths available for an image and provides access to path functions. *Undo* shows a limited history of modifications made to the current image.

The bottom portion of the window contains three tabs. With them, select the current brush, gradient, and pattern.

17.3 Getting Started in GIMP

Although GIMP can be a bit overwhelming for new users, most quickly find it easy to use once they work out a few basics. Crucial basic functions are creating, opening, and saving images.

17.3.1 Creating a New Image

To create a new image, select *File* → *New* or press Ctrl + N. This opens a dialog in which to make settings for the new image. If desired, use *Template* to select a template on which to base the new image. The GIMP includes a number of templates, ranging from an A4 sheet of paper to a CD cover, from which to choose. To create a custom template, select *File* → *Dialogs* → *Templates* and use the controls offered by the window that opens.

In the *Image Size* section, set the size of the image to create in pixels or another unit. Click the unit to select another unit from the list of available units. The ratio between pixels and a unit is set in *Resolution*, which appears when the *Advanced Options* section is open. A resolution of 72 pixels per inch corresponds to screen display. It is sufficient for Web page graphics. A higher resolution should be used for images to print. For most printers, a resolution of 300 pixels per inch results in an acceptable quality.

In *Colorspace*, select whether the image should be in color (*RGB*) or *Grayscale*. Select the *Fill Type* for the new image. *Foreground Color* and *Background Color* use the colors selected in the toolbox. *White* uses a white background in the image. *Transparent* creates a clear image. *Transparency* is represented by a gray checkerboard pattern. Enter a comment for the new image in *Comment*.

When the settings meet your needs, press *OK*. To restore the default settings, press *Reset*. Pressing *Cancel* aborts creation of a new image.

17.3.2 Opening an Existing Image

To open an existing image, select *File* → *Open* or press Ctrl + O. In the dialog that opens, select the desired file. Click *OK* to open the selected image. Press *Cancel* to skip opening an image.

17.3.3 The Image Window

The new or opened image appears in its own window. The menu bar in the top of the window provides access to all image functions. Alternatively, access the menu by right-clicking the image or clicking the small arrow button in the left corner of the rulers.

File offers the standard file options, such as *Save* and *Print*. *Close* closes the current image. *Quit* closes the entire application.

With the items in the *View* menu, control the display of the image and the image window. *New View* opens a second display window of the current image. Changes made in one view are reflected in all other views of that image. Alternate views are useful for magnifying a part of an image for manipulation while seeing the complete image in another view. Adjust the magnification level of the current window with *Zoom*. When *Shrink Wrap* is selected, the image window is resized to fit the current image display exactly.

17.4 Saving Images

No image function is as important as *File* → *Save*. It is better to save too often than too rarely. Use *File* → *Save as* to save the image with a new filename. It is a good idea to save image stages under different names or make backups in another directory so you can easily restore a previous state.

When saving for the first time or using *Save as*, a dialog opens in which to specify the filename and type. Enter the filename in the field at the top. For *Save in folder*, select the directory in which to save the file from a list of commonly used directories. To use a different directory or create a new one, open *Browse for other folders*. It is recommended to leave *Select File Type* set to *By Extension*. With that setting, GIMP determines the file type based on the extension appended to the filename. The following file types are frequently useful:

XCF

This is the native format of the application. It saves all layer and path information along with the image itself. Even if you need an image in another format, it is usually a good idea to save a copy as XCF to simplify future modifications.

PAT

This is the format used for GIMP patterns. Saving an image in this format enables using the image as a fill pattern in GIMP.

JPG

JPG or JPEG is a common format for photographs and Web page graphics without transparency. Its compression method enables reduction of file sizes, but information is lost when compressing. It may be a good idea to use the preview option when adjusting the compression level. Levels of 85% to 75% often result in an acceptable image quality with reasonable compression. Saving a backup in a lossless format, like XCF, is also recommended. If editing an image, save only the finished image as JPG. Repeatedly loading a JPG then saving can quickly result in poor image quality.

GIF

Although very popular in the past for graphics with transparency, GIF is less often used now because of license issues. GIF is also used for animated images. The format can only save *indexed* images. The file size can often be quite small if only a few colors are used.

PNG

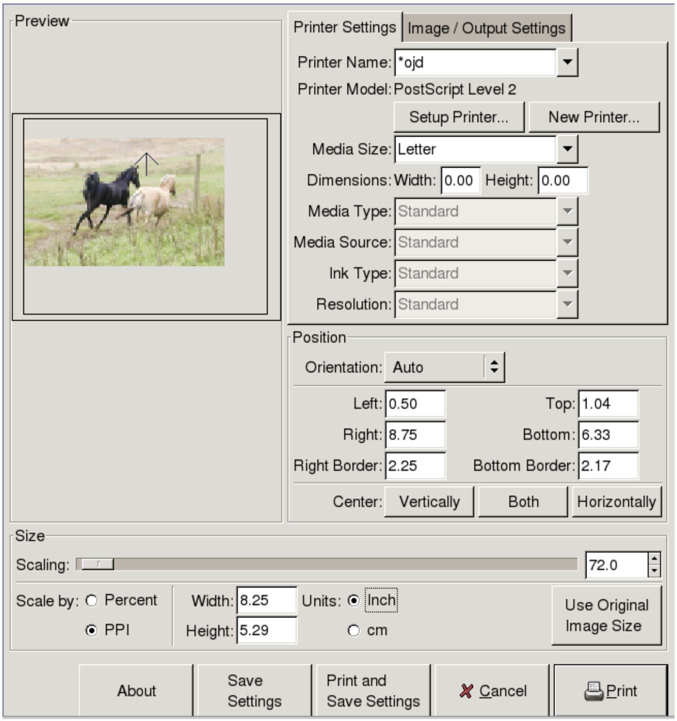
With its support for transparency, lossless compression, free availability, and increasing browser support, PNG is replacing GIF as the preferred format for Web graphics with transparency. An added advantage is that PNG offers partial transparency, which is not offered by GIF. This enables smoother transitions from colored areas to transparent areas (*antialiasing*).

To save the image in the chosen format, press *Save*. To abort, press *Cancel*. If the image has features that cannot be saved in the chosen format, a dialog appears with choices for resolving the situation. Choosing *Export*, if offered, normally gives the desired results. A window then opens with the options of the format. Reasonable default values are provided.

17.5 Printing Images

To print an image, select *File* → *Print* from the image menu. If your printer is configured in SUSE Linux, it should appear in the list. In some cases, it may be necessary to select an appropriate driver with *Setup Printer*. Select the appropriate paper size with *Media Size* and the type in *Media Type*. Other settings are available in the *Image / Output Settings* tab.

Figure 17.2 *The Print Dialog*



Printer Name: *ojd
Printer Model: PostScript Level 2
Media Size: Letter
Dimensions: Width: 0.00 Height: 0.00
Media Type: Standard
Media Source: Standard
Ink Type: Standard
Resolution: Standard

Orientation: Auto
Left: 0.50 Top: 1.04
Right: 8.75 Bottom: 6.33
Right Border: 2.25 Bottom Border: 2.17

Scaling: 72.0
Scale by: ○ Percent ● PPI
Width: 8.25 Height: 5.29
Units: ● Inch ○ cm

In the bottom portion of the window, adjust the image size. Press *Use Original Image Size* to take these settings from the image itself. This is recommended if you set an appropriate print size and resolution in the image. Adjust the image's position on the page with the fields in *Position* or by dragging the image in *Preview*.

When satisfied with the settings, press *Print*. To save the settings for future use, instead use *Print and Save Settings*. *Cancel* aborts printing.

17.6 For More Information

The following are some resources that may be useful for a GIMP user. Unfortunately, many resources apply to older versions.

- *Help* provides access to the internal help system. This documentation is also available in HTML and PDF formats at `http://docs.gimp.org`.
- The GIMP User Group offers an informative and interesting Web site at `http://gug.sunsite.dk`.
- `http://www.gimp.org` is the official home page of The GIMP.
- *Grokking the GIMP* by Carey Bunks is an excellent book based on an older GIMP version. Although some aspects of the program have changed, it can provide excellent guidance for image manipulation. An online version is available at `http://gimp-savvy.com/BOOK/`.
- `http://gimp-print.sourceforge.net` is the Web page for the GIMP print plug-in. The user manual available from the site provides detailed information about configuring and using the program.

Part VI Mobility

Mobile Computing with Linux 18

This chapter provides an overview of the various aspects of using Linux for mobile computing. The various fields of use are briefly introduced and the essential features of the employed hardware are described. Software solutions for special requirements and options for maximum performance are covered along with possibilities to minimize power consumption. An overview of the most important sources of information about the subject concludes the chapter.

Most people associate mobile computing with laptops, PDAs, and cellular phones and the data exchange between them. This chapter extends the focus to mobile hardware components, such as external hard disks, flash drives, or digital cameras, which can be connected to laptops or desktop systems.

18.1 Laptops

The hardware of laptops differs from that of a normal desktop system. This is because criteria like exchangeability, occupied space, and power consumption are relevant properties. The manufacturers of mobile hardware have developed the PCMCIA (Personal Computer Memory Card International Association) standard. This standard covers memory cards, network interface cards, ISDN and modem cards, and external hard disks. How the support for such hardware is implemented in Linux, what needs to be taken into account during configuration, what software is available for the control of PCMCIA, and how to troubleshoot any possible problems is described in Chapter 19, *PCMCIA* (page 245).

18.1.1 Power Conservation

The inclusion of energy-optimized system components when manufacturing laptops contributes to their suitability for use without access to the electrical power grid. Their contribution towards conservation of power is at least as important as that of the operating system. SUSE Linux supports various methods that influence the power consumption of a laptop and have varying effects on the operating time under battery power. The following list is in descending order of contribution towards power conservation:

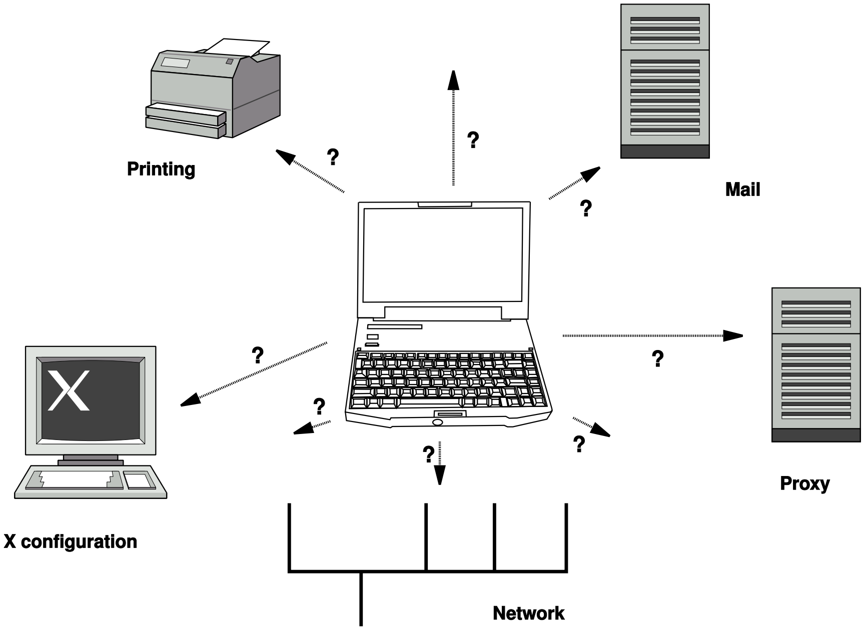
- Throttling the CPU speed
- Switching off the display illumination during pauses
- Manually adjusting the display illumination
- Disconnecting unused, hotplug-enabled accessories (USB CD-ROM, external mouse, unused PCMCIA cards, etc.)
- Spinning down the hard disk when idling

Detailed background information about power management in SUSE Linux and about operating the YaST power management module is provided in Chapter 21, *Power Management* (page 259).

18.1.2 Integration in Changing Operating Environments

Your system needs to adapt to changing operating environments when used for mobile computing. A lot of services depend on the environment and the underlying clients must be reconfigured. SUSE Linux takes over this job for you.

Figure 18.1 Integrating a Laptop in a Network



The services affected in the case of a laptop commuting back and forth between a small home network and an office network are:

Network Configuration
This includes IP address assignment, name resolution, Internet connectivity, and connectivity to other networks.

Printing
A current database of available printers and an available print server must be present, depending on the network.

E-Mail and Proxies
As with printing, the list of the corresponding servers must be current.

Configuring X
If your laptop is temporarily connected to a beamer or an external monitor, the different display configurations must be available.

SUSE Linux offers two ways of integrating a laptop into existing operating environments. They can be combined.

SCPM

SCPM (system configuration profile management) allows storage of arbitrary configuration states of a system into a kind of “snapshot” called a *profile*. Profiles can be created for different situations. They are useful when a system is operated in changing environments (home network, office network). It is always possible to switch between profiles. Information about SCPM can be found in Chapter 20, *System Configuration Profile Management* (page 247). The kicker applet Profile Chooser in KDE allows switching between profiles. The application requires the root password before switching.

SLP

The service location protocol (SLP) simplifies the connection of a laptop to an existing network. Without SLP, the administrator of a laptop usually requires detailed knowledge of the services available in a network. SLP broadcasts the availability of a certain type of service to all clients in a local network. Applications that support SLP can process the information dispatched by SLP and be configured automatically. SLP can even be used for the installation of a system and spare the effort of searching for a suitable installation source. Detailed information about SLP can be found in Chapter 39, *SLP Services in the Network* (page 589).

The emphasis of SCPM lies on enabling and maintaining reproducible system conditions. SLP makes configuration of a networked computer a lot easier by automating much of it.

18.1.3 Software Options

There are various special task areas in mobile use that are covered by dedicated software: system monitoring (especially the battery charge), data synchronization, and wireless communication with peripherals and the Internet. The following sections cover the most important applications that SUSE Linux provides for each task.

System Monitoring

Two KDE system monitoring tools are provided by SUSE Linux. The pure status display of the rechargeable battery of the laptop is handled by the applet KPowersave in the kicker. Complex system monitoring is performed by KSysguard. When using GNOME, the described functions are provided by GNOME ACPI (as panel applet) and System Monitor.

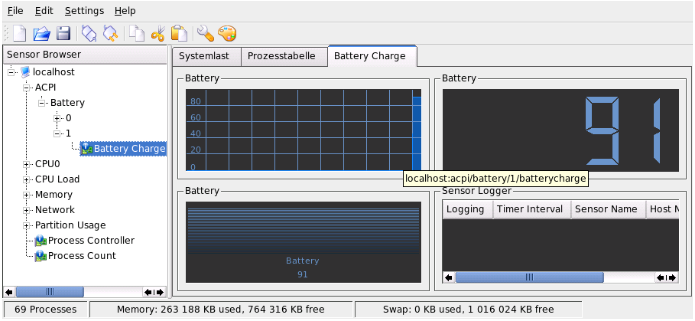
KPowersave

KPowersave is an applet that displays the state of the rechargeable battery in the control panel. The icon adjusts to represent the type of power supply. When working on AC power, a small plug icon is displayed. When working on batteries, the icon changes to a battery. The corresponding menu opens the YaST module for power management after requesting the root password. This allows setting the behavior of the system under different types of power supply. Information about power management and about the corresponding YaST module can be found in Chapter 21, *Power Management* (page 259).

KSysguard

KSysguard is an independent application that gathers all measurable parameters of the system into one monitoring environment. KSysguard has monitors for ACPI (battery status), CPU load, network, partitioning, and memory usage. It can also watch and display all system processes. The presentation and filtering of the collected data can be customized. It is possible to monitor different system parameters in various data pages or collect the data of various machines in parallel over the network. KSysguard can also run as a daemon on machines without a KDE environment. More information about this program is provided in its integrated help function or in the SUSE help pages.

Figure 18.2 *Monitoring the Battery State with KSysguard*



69 Processes | Memory: 263 188 KB used, 764 316 KB free | Swap: 0 KB used, 1 016 024 KB free

Synchronizing Data

When switching between working on a mobile machine disconnected from the network and working at a networked workstation in an office, it is necessary to keep processed data synchronized across all instances. This could include e-mail folders, directories, and individual files that need to be present for work on the road as well as at the office. The solution in both cases is as follows:

Synchronization of E-Mail

Use an IMAP account for storing your e-mails in the office network. The e-mails are then accesssed from the workstation using any disconnected IMAP–enabled e-mail client, like Mozilla Thunderbird Mail, Evolution, or KMail as described in the *Start-Up*. The e-mail client must be configured so that the same folder is always accessed for `Sent messages`. This ensures that all messages are available along with their status information after the synchronization process has completed. Use an SMTP server implemented in the mail client for sending messages instead of the systemwide MTA postfix or sendmail to receive reliable feedback about unsent mail.

Synchronizing Files and Directories

There are several utilities suitable for synchronizing data between a laptop and a workstation. For detailed information, refer to Chapter 47, *File Synchronization* (page 715).

Wireless Communication

As well as connecting to a home or office network with a cable, a laptop can also wirelessly connected to other computers, peripherals, cellular phones, or PDAs. Linux supports three types of wireless communication:

WLAN

With the largest range of these wireless technologies, WLAN is the only one suitable for the operation of large and sometimes even spatially disjointed networks. Single machines can connect with each other to form an independent wireless network or access the Internet. Devices called access points act as base stations for WLAN-enabled devices and act as intermediate for access to the Internet. A mobile user can switch among access points depending on location and which access point is offering the best connection. Like in cellular telephony, a large network is available to WLAN users without binding them to a specific location for accessing it. Details about WLAN can be found in Section 22.1, “Wireless LAN” (page 283).

Bluetooth

Bluetooth has the broadest application spectrum of all wireless technologies. It can be used for communication between computers (laptops) and PDAs or cellular phones, as can IrDA. It can also be used to connect various computers within visible range. Bluetooth is also used to connect wireless system components, like a keyboard or mouse. The range of this technology is, however, not sufficient to connect remote systems to a network. WLAN is the technology of choice for communicating through physical obstacles like walls. More information about Bluetooth, its applications, and configuration can be found in Section 22.2, “Bluetooth” (page 293).

IrDA

IrDA is the wireless technology with the shortest range. Both communication parties must be within viewing distance of each other. Obstacles like walls cannot be overcome. One possible application of IrDA is the transmission of a file from a laptop to a cellular phone. The short path from the laptop to the cellular phone is then covered using IrDA. The long range transport of the file to the recipient of the file is handled by the mobile network. Another application of IrDA is the wireless transmission of printing jobs in the office. More information about IrDA can be found in Section 22.3, “Infrared Data Transmission” (page 304).

18.1.4 Data Security

Ideally, you protect data on your laptop against unauthorized access in multiple ways. Possible security measures can be taken in the following areas:

Protection against Theft

Always physically secure your system against theft whenever possible. Various securing tools, like chains, are available in retail stores.

Securing Data on the System

Important data should not only be encrypted during transmission, but also on the hard disk. This ensures its safety in case of theft. The creation of an encrypted partition with SUSE Linux is described in Section 23.3, “Encrypting Partitions and Files” (page 325).

IMPORTANT: Data Security and Suspend to Disk

Encrypted partitions are not unmounted during a suspend to disk event. Thus, all data on these partitions is available to any party who manages to steal the hardware and issue a resume of the hard disk.

Network Security

Any transfer of data should be secured, no matter how it takes place. General security issues regarding Linux and networks can be found in Section 23.4, “Security and Confidentiality” (page 328). Security measures related to wireless networking are provided in Chapter 22, *Wireless Communication* (page 283).

18.2 Mobile Hardware

SUSE Linux supports the automatic detection of mobile storage devices over firewire (IEEE 1394) or USB. The term *mobile storage device* applies to any kind of firewire or USB hard disk, USB flash drive, or digital camera. These devices are automatically detected and configured via hotplug as soon as they are connected with the system over the corresponding interface. subfs and submount ensure that the devices are mounted to the corresponding locations in the file system. The user is completely spared the manual mounting and unmounting that was found in previous versions of SUSE Linux. A device can simply be disconnected as soon as no program accesses it.

External Hard Disks (USB and Firewire)

As soon as an external hard disk has been correctly recognized by the system, its icon appears in *My Computer* (KDE) or *Computer* (GNOME) in the list of mounted drives. Clicking the icon displays the contents of the drive. It is possible to create folders and files here and edit or delete them. To rename a hard disk from the name it had been given by the system, select the corresponding menu item from the menu that opens when the icon is right-clicked. This name change is limited to display in the file manager. The descriptor by which the device is mounted in `/media/usb-xxx` or `/media/ieee1394-xxx` remains unaffected by this.

USB Flash Drives

These devices are handled by the system just like external hard disks. It is similarly possible to rename the entries in the file manager.

Digital Cameras (USB and Firewire)

Digital cameras recognized by the system also appear as external drives in the overview of the file manager. KDE allows reading and accessing the pictures at the URL `camera:/`. The images can then be processed using Digikam or The GIMP. When using GNOME, Nautilus displays the pictures in their own folder. A simple image processing and management utility is f-spot. Advanced photo processing is done with The GIMP. For more details on digital cameras and image management, refer to Chapter 15, *Digital Cameras and Linux* (page 201).

18.3 Cellular Phones and PDAs

A desktop system or a laptop can communicate with a cellular phone via Bluetooth or IrDA. Some models support both protocols and some only one of the two. The usage areas for the two protocols and the corresponding extended documentation has already been mentioned in Section “Wireless Communication” (page 240). The configuration of these protocols on the cellular phones themselves is described in their manuals. The configuration of the Linux side is described in Section 22.2, “Bluetooth” (page 293) and Section 22.3, “Infrared Data Transmission” (page 304).

The support for syncronizing with handheld devices manufactured by Palm, Inc., is already built into Evolution and Kontact. Initial connection with the device is, in both cases, easily performed with the assistance of a wizard. Once the support for Palm Pilots is configured, it is necessary to determine which type of data should be synchronized (addresses, appointments, etc.). Both groupware applications are described in the *Start-Up*.

The program KPilot as integrated in Kontact is also available as an independent utility. It is described in the *Start-Up*. The program KitchenSync is also available for synchronizing address data.

18.4 For More Information

The central point of reference for all questions regarding mobile devices and Linux is `http://tuxmobil.org/`. Various sections of that Web site deal with the hardware and software aspects of laptops, PDAs, cellular phones, and other mobile hardware.

A similar approach to that of http://tuxmobil.org/ is made by http://www.linux-on-laptops.com/. Information about laptops and handhelds can be found here.

SUSE maintains a mailing list in German dedicated to the subject of laptops. See http://lists.suse.com/archive/suse-laptop/. On this list, users and developers discuss all aspects of mobile computing with SUSE Linux. Postings in English are answered, but the majority of the archived information is only available in German.

In the case of problems with power management with SUSE Linux on laptops, it is advisable to read the file README in /usr/share/doc/packages/powersave. This directory often contains last minute feedback by testers and developers, so provides valuable hints for the solution of problems.

19 PCMCIA

This section covers special aspects of PCMCIA hardware and software as found in laptops. PCMCIA stands for *Personal Computer Memory Card International Association* and is used as a collective term for all related hardware and software.

19.1 Hardware

The most important component is the PCMCIA card. There are two types of PCMCIA cards:

PC Cards
These cards have been around since the dawn of PCMCIA. They use a 16-bit bus for the data transmission and are usually quite inexpensive. Some modern PCMCIA bridges have difficulties detecting these cards. Nevertheless, once they are detected, they usually run smoothly and do not cause any problems.

CardBus Cards
This is a more recent standard. They use a 32-bit bus, which makes them faster but also more expensive. They are integrated in the system like PCI cards and also run smoothly.

The second important component is the PCMCIA controller or the PC card or CardBus bridge, which establishes the connection between the card and the PCI bus. All common models are supported. If it is a built-in PCI device, the command `lspci -vt` provides further information.

19.2 Software

With the current kernel, PCMCIA bridges and PCMCIA cards are handled by the hotplug subsystem. There are pcmcia_socket events for every bridge and pcmcia events. udevd loads all needed modules and calls the necessary tools to set up these devices. These actions are defined in /etc/udev/rules.d/.

/etc/pcmcia/config.opts is used for resource configuration. The needed driver is determined by device tables in the drivers. Information about the current state of the sockets and the cards can be found in /sys/class/pcmcia_socket/ and via pccardctl.

Because there are ongoing changes in the PCMCIA system, this documentation is incomplete. For a comprehensive overview, refer to /usr/share/doc/packages/pcmciautils/README.SUSE.

System Configuration Profile Management 20

With the help of SCPM (system configuration profile management), adapt the configuration of your computer to different operating environments or hardware configurations. SCPM manages a set of system profiles for the different scenarios. SCPM enables easy switching between system profiles, eliminating the need for manually reconfiguring the system.

Some situations require a modified system configuration. This would mostly be the case for mobile computers that are operated in varying locations. If a desktop system should be operated temporarily using other hardware components than usual, SCPM comes in handy. Restoring the original system configuration should be easy and the modification of the system configuration can be reproduced. With SCPM, any part of the system configuration can be kept in a customized profile.

The main field of application of SCPM is network configuration on laptops. Different network configurations often require different settings of other services, such as e-mail or proxies. Then other elements follow, like different printers at home and at the office, a customized X server configuration for the multimedia projector at conferences, special power-saving settings for the road, or a different time zone at an overseas subsidiary.

20.1 Terminology

The following are some terms used in SCPM documentation and in the YaST module.

- The term *system configuration* refers to the complete configuration of the computer. It covers all fundamental settings, such as the use of hard disk partitions, network settings, time zone selection, and keyboard mappings.

- A *profile*, also called *configuration profile*, is a state that has been preserved and can be restored at any time.

- *Active profile* refers to the profile last selected. This does not mean that the current system configuration corresponds exactly to this profile, because the configuration can be modified at any time.

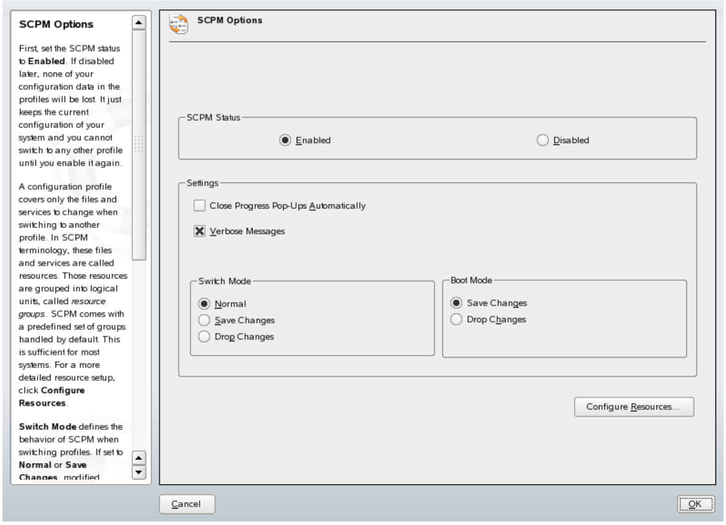
- A *resource* in the SCPM context is an element that contributes to the system configuration. This can be a file or a softlink including metadata (like the user), permissions, or access time. This can also be a system service that runs in this profile, but is deactivated in another one.

- Every resource belongs to a certain *resource group*. These groups contain all resources that logically belong together—most groups would contain both a service and its configuration files. It is very easy to assemble resources managed by SCPM because this does not require any knowledge of the configuration files of the desired service. SCPM ships with a selection of preconfigured resource groups that should be sufficient for most scenarios.

20.2 Using the YaST Profile Manager

Start the YaST profile manager from the YaST control center with *System → Profile Manager*. On first start, explicitly enable SCPM by selecting *Enabled* in the *SCPM Options* dialog, shown in Figure 20.1, “YaST SCPM Options” (page 249). In *Settings*, determine whether progress pop-ups should be closed automatically and whether to display verbose messages about the progress of your SCPM configuration. For *Switch Mode*, determine whether modified resources of the active profile should be saved or discarded when the profile is switched. If *Switch Mode* is set to *Normal*, all changes in the active profile are saved when switched. To define the behavior of SCPM at boot time, set *Boot Mode* to *Save Changes* (default setting) or to *Drop Changes*.

Figure 20.1 *YaST SCPM Options*



SCPM Options

First, set the SCPM status to **Enabled**. If disabled later, none of your configuration data in the profiles will be lost. It just keeps the current configuration of your system and you cannot switch to any other profile until you enable it again.

A configuration profile covers only the files and services to change when switching to another profile. In SCPM terminology, these files and services are called resources. Those resources are grouped into logical units, called *resource groups*. SCPM comes with a predefined set of groups handled by default. This is sufficient for most systems. For a more detailed resource setup, click **Configure Resources**.

Switch Mode defines the behavior of SCPM when switching profiles. If set to **Normal** or **Save Changes**, modified

SCPM Options

SCPM Status: ● Enabled ○ Disabled

Settings

- ☐ Close Progress Pop-Ups Automatically
- ☒ Verbose Messages

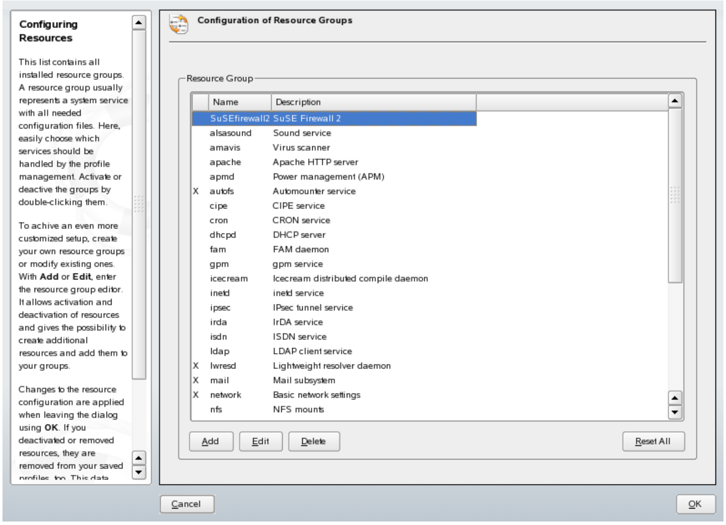
Switch Mode: ● Normal ○ Save Changes ○ Drop Changes

Boot Mode: ● Save Changes ○ Drop Changes

20.2.1 Configuring Resource Groups

To make changes to the current resource configuration, choose *Configure Resources* in the *SCPM Options* dialog. The next dialog, shown in Figure 20.2, “Configuring Resource Groups” (page 250), lists all resource groups available on your system. To add or edit a resource group, specify or modify *Resource Group* and *Description*. For an LDAP service, for example, enter `ldap` as *Resource Group* and `LDAP client service` as *Description*. Then enter the appropriate resources (services, configuration files, or both) or modify the existing ones. Delete those that are not used. To reset the status of the selected resources—discard any changes made to them and return to the initial configuration values—choose *Reset Group*. Your changes are saved to the active profile.

Figure 20.2 *Configuring Resource Groups*



Configuring Resources

This list contains all installed resource groups. A resource group usually represents a system service with all needed configuration files. Here, easily choose which services should be handled by the profile management. Activate or deactive the groups by double-clicking them.

To achive an even more customized setup, create your own resource groups or modify existing ones. With **Add** or **Edit**, enter the resource group editor. It allows activation and deactivation of resources and gives the possibility to create additional resources and add them to your groups.

Changes to the resource configuration are applied when leaving the dialog using **OK**. If you deactivated or removed resources, they are removed from your saved profiles, too. This data

Configuration of Resource Groups

Resource Group

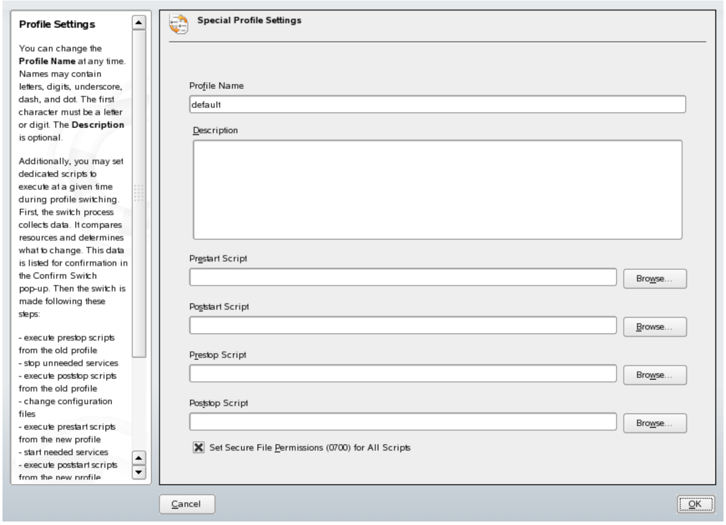
	Name	Description
	SuSEfirewall2	SuSE Firewall 2
	alsasound	Sound service
	amavis	Virus scanner
	apache	Apache HTTP server
	apmd	Power management (APM)
X	autofs	Automounter service
	cipe	CIPE service
	cron	CRON service
	dhcpd	DHCP server
	fam	FAM daemon
	gpm	gpm service
	icecream	Icecream distributed compile daemon
	inetd	inetd service
	ipsec	IPsec tunnel service
	irda	IrDA service
	isdn	ISDN service
	ldap	LDAP client service
X	lwresd	Lightweight resolver daemon
X	mail	Mail subsystem
X	network	Basic network settings
	nfs	NFS mounts

20.2.2 Creating a New Profile

To create a new profile, click *Add* in the start dialog (*System Configuration Profile Management*). In the window that opens, select whether the new profile should be based on the current system configuration (SCPM automatically retrieves the current configuration and writes it to your profile) or on an existing profile. If you use the current system configuration as the base of the new profile, you can mark the new profile as the new active profile. This makes no changes to the old profile and does not start or stop any services.

Provide a name and a short description for the new profile in the following dialog. For SCPM to execute special scripts on a switch of profiles, enter the paths to each executable (see Figure 20.3, “Special Profile Settings” (page 251)). Refer to Section 20.3.4, “Advanced Profile Settings” (page 254) for more information. SCPM runs a check for the resources of the new profile. After this test has been successfully completed, the new profile is ready for use.

Figure 20.3 *Special Profile Settings*



Profile Settings

You can change the **Profile Name** at any time. Names may contain letters, digits, underscore, dash, and dot. The first character must be a letter or digit. The **Description** is optional.

Additionally, you may set dedicated scripts to execute at a given time during profile switching. First, the switch process collects data. It compares resources and determines what to change. This data is listed for confirmation in the Confirm Switch pop-up. Then the switch is made following these steps:

- execute prestop scripts from the old profile
- stop unneeded services
- execute poststop scripts from the old profile
- change configuration files
- execute prestart scripts from the new profile
- start needed services
- execute poststart scripts from the new profile

Special Profile Settings

Profile Name: default

Description:

Prestart Script:

Poststart Script:

Prestop Script:

Poststop Script:

☒ Set Secure File Permissions (0700) for All Scripts

20.2.3 Modifying Existing Profiles

To modify an existing profile, choose *Edit* in the start dialog (*System Configuration Profile Management*). Then modify the name, description, scripts, and resources according to your needs.

20.2.4 Switching Profiles

To switch profiles, open the profile manager. The active profile is marked with an arrow. Select the profile to which to switch and click *Switch To*. SCPM checks for new or modified resources and adds them, if necessary.

If a resource has been modified, YaST opens the *Confirm Switch* dialog. *Modified Resource Groups of Active Profile* lists all resource groups of the active profile that have been modified but not yet saved to the active profile. *Save or Ignore* for the currently selected resource group determines whether changes to this resource group should be saved to the active profile or discarded. Alternatively, select each resource and click *Details* to analyze the changes in detail. This shows a list of all configuration files or executables belonging to this resource group that have been modified. To get a line-by-line comparison of the old and new version, click *Show Changes*. After analyzing the changes, decide what to do with them in *Action*:

Save Resource
Save this resource to the active profile, but leave all other profiles untouched.

Ignore Resource
Leave the active resource untouched. This change is discarded.

Save to All Profiles
Copy the entire configuration of this resource to all other profiles.

Patch All Profiles
Apply only the most recent changes to all profiles.

Save or Ignore All just saves or discards the changes of all resources shown in this dialog.

After confirming the changes to the active profile, leave the *Confirm Switch* dialog by clicking *OK*. SCPM then switches to the new profile. While switching, it executes the prestop and poststop scripts of the old profile and the prestart and poststart scripts for the new profile.

20.3 Configuring SCPM Using the Command Line

This section introduces the command-line configuration of SCPM. Learn how to start it, configure it, and work with profiles.

20.3.1 Starting SCPM and Defining Resource Groups

SCPM must be activated before use. Activate SCPM with `scpm enable`. When run for the first time, SCPM is initialized, which takes a few seconds. Deactivate SCPM with `scpm disable` at any time to prevent the unintentional switching of profiles. A subsequent reactivation simply resumes the initialization.

By default, SCPM handles network and printer settings as well as the X.Org configuration. To manage special services or configuration files, activate the respective resource groups. To list the predefined resource groups, use `scpm list_groups`. To see only the groups already activated, use `scpm list_groups -a`. Issue these commands as `root` on the command line.

```
scpm list_groups -a

nis                  Network Information Service client
mail                 Mail subsystem
ntpd                 Network Time Protocol daemon
xf86                 X Server settings
autofs               Automounter service
network              Basic network settings
printer              Printer settings
```

Activate or deactivate a group with `scpm activate_group NAME` or `scpm deactivate_group NAME`. Replace `NAME` with the relevant group name.

20.3.2 Creating and Managing Profiles

A profile named `default` already exists after SCPM has been activated. Get a list of all available profiles with `scpm list`. This one existing profile is also the active one, which can be verified with `scpm active`. The profile `default` is a basic configuration from which the other profiles are derived. For this reason, all settings that should be identical in all profiles should be made first. Then store these modifications in the active profile with `scpm reload`. The `default` profile can be copied and renamed as the basis for new profiles.

There are two ways to add a new profile. If the new profile (named `work` here) should be based on the profile `default`, create it with `scpm copy default work`. The command `scpm switch work` changes into the new profile, which can then be modified. You may want to modify the system configuration for special purposes and save the changes to a new profile. The command `scpm add work` creates a new profile by saving the current system configuration in the profile `work` and marking it as active. Running `scpm reload` then saves changes to the profile `work`.

Profiles can be renamed or deleted with the commands `scpm rename x y` and `scpm delete z`. For example, to rename `work` to `project`, enter `scpm rename work project`. To delete `project`, enter `scpm delete project`. The active profile cannot be deleted.

20.3.3 Switching Configuration Profiles

The command `scpm switch work` switches to another profile (the profile `work`, in this case). Switch to the active profile to include modified settings of the system configuration in the profile. This corresponds to the command `scpm reload`.

When switching profiles, SCPM first checks which resources of the active profile have been modified. It then queries whether the modification of each resource should be added to the active profile or dropped. If you prefer a separate listing of the resources (as in former versions of SCPM), use the switch command with the `-r` parameter: `scpm switch -r work`.

```
scpm switch -r work

Checking for modified resources
Checking for Resources to be started/shut down
Checking for dependencies
Restoring profile default
```

SCPM then compares the current system configuration with the profile to which to switch. In this phase, SCPM evaluates which system services need to be stopped or restarted due to mutual dependencies or to reflect the changes in configuration. This is like a partial system reboot that concerns only a small part of the system while the rest continues operating without change. It is only at this point that the system services are stopped, all modified resources, such as configuration files, are written, and the system services are restarted.

20.3.4 Advanced Profile Settings

You can enter a description for every profile that is displayed with `scpm list`. For the active profile, set it with `scpm set description "text"`. Provide the name of the profile for inactive profiles, for example, `scpm set description "text" work`. Sometimes it might be desirable to perform additional actions not provided by SCPM while switching profiles. Attach up to four executables for each profile. They are invoked at different stages of the switching process. These stages are referred to as:

prestop
: prior to stopping services when leaving the profile

poststop
after stopping services when leaving the profile

prestart
prior to starting services when activating the profile

poststart
after starting services when activating the profiles

Insert these actions with the command `set` by entering `scpm set prestop filename`, `scpm set poststop filename`, `scpm set prestart filename`, or `scpm set poststart filename`. The scripts must be executable and refer to the correct interpreter.

WARNING: Integrating a Custom Script

Additional scripts to be executed by SCPM must be made readable and executable for the superuser (`root`). The access to these files must be blocked for all other users. Enter the commands `chmod 700 filename` and `chown root:root filename` to give `root` exclusive permissions to the files.

Query all additional settings entered with `set` with `get`. The command `scpm get poststart`, for example, returns the name of the poststart call or simply nothing if nothing has been attached. Reset such settings by overwriting with `""`. The command `scpm set prestop ""` removes the attached prestop program.

All `set` and `get` commands can be applied to an arbitrary profile in the same manner as comments are added. For example, `scpm get prestop filename work` or `scpm get prestop work`.

20.4 Using the Profile Chooser Applet

The Profile Chooser applet in your GNOME or KDE desktop panel allows you to easily control your SCPM settings. Create, modify or delete profiles via YaST as described in Section 20.2, “Using the YaST Profile Manager” (page 248) and switch profiles. Switching profiles can be done as normal user provided the system administrator allows this to happen. Start Profile Chooser from your desktop menu using *System → Desktop Applet → Profile Chooser*.

Enable normal users to switch profiles by right-clicking the Profile Chooser icon in the desktop panel and choosing *Allow user switching* from the menu that opens. Provide the root password. Any authorized user on your system can switch profiles from now on.

All profiles configured in YaST, either directly via a YaST call or via the *Start YaST2 Profile Manager Module* are displayed after you click the Profile Chooser icon. Select the one to switch to using the cursor keys and SCPM changes to the new profile automatically.

20.5 Troubleshooting

This section covers frequent problems encountered with SCPM. Learn how they can arise and how you can solve these issues.

20.5.1 Termination During the Switch Process

Sometimes SCPM stops working during a switch procedure. This may be caused by some outside effect, such as a user abort, a power failure, or even an error in SCPM itself. If this happens, an error message stating SCPM is locked appears the next time you start SCPM. This is for system safety, because the data stored in its database may differ from the state of the system. To resolve this issue, run `scpm recover`. SCPM performs all missing operations of the previous run. You can also run `scpm recover -b`, which tries to undo all already performed operations of the previous run. If you are using the YaST profile manager, get a recover dialog on start-up that offers to perform the commands described above.

20.5.2 Changing the Resource Group Configuration

To modify the configuration of the resource group when SCPM is already initialized, enter `scpm rebuild` after adding or removing groups. In this way, new resources are added to all profiles and the removed resources are deleted permanently. If the deleted resources are configured differently in the various profiles, this configuration data is lost, except for the current version in your system, which SCPM does not touch. If you modify the configuration with YaST, the rebuild command does not need to be entered, because this is handled by YaST.

20.6 Selecting a Profile When Booting the System

To select a profile when booting the system, press F3 in the boot screen to access a list of available profiles. Use the arrow keys to select a profile and confirm your selection with Enter. The selected profile is then used as a boot option.

20.7 For More Information

The latest documentation is available in the SCPM info pages (info scpm). Information for developers is available in `/usr/share/doc/packages/scpm`.

21 Power Management

Power management is especially important on laptop computers, but is also useful on other systems. Two technologies are available: APM (advanced power management) and ACPI (advanced configuration and power interface). In addition to these, it is also possible to control CPU frequency scaling to save power or decrease noise. These options can be configured manually or using a special YaST module.

Unlike APM, which was previously used on laptops for power management only, the hardware information and configuration tool ACPI is available on all modern computers (laptops, desktops, and servers). All power management technologies require suitable hardware and BIOS routines. Most laptops and many modern desktops and servers meet these requirements.

APM had been used in many older computers. Because APM largely consists of a function set implemented in the BIOS, the level of APM support may vary depending on the hardware. This is even more true of ACPI, which is even more complex. For this reason, it is virtually impossible to recommend one over the other. Simply test the various procedures on your hardware then select the technology that is best supported.

IMPORTANT: Power Management for AMD64 Processors

AMD64 processors with a 64-bit kernel only support ACPI.

21.1 Power Saving Functions

Power saving functions are not only significant for the mobile use of laptops, but also for desktop systems. The main functions and their use in the power management systems APM and ACPI are:

Standby

This operating mode turns off the display. On some computers, the processor performance is throttled. This function is not available in all APM implementations. This function corresponds to the ACPI state S1 or S2.

Suspend (to memory)

This mode writes the entire system state to the RAM. Subsequently, the entire system except the RAM is put to sleep. In this state, the computer consumes very little power. The advantage of this state is the possibility of resuming work at the same point within a few seconds without having to boot and restart applications. Devices using APM can usually be suspended by closing the lid and activated by opening it. This function corresponds to the ACPI state S3. The support of this state is still under development and therefore largely depends on the hardware.

Hibernation (suspend to disk)

In this operating mode, the entire system state is written to the hard disk and the system is powered off. Reactivation from this state takes about 30 to 90 seconds. The state prior to the suspend is restored. Some manufacturers offer useful hybrid variants of this mode, such as RediSafe in IBM Thinkpads. The corresponding ACPI state is S4. In Linux, suspend to disk is performed by kernel routines that are independent from APM and ACPI.

Battery Monitor

ACPI and APM check the battery charge status and provide information about the charge status. Additionally, both systems coordinate actions to perform when a critical charge status is reached.

Automatic Power-Off

Following a shutdown, the computer is powered off. This is especially important when an automatic shutdown is performed shortly before the battery is empty.

Shutdown of System Components

Switching off the hard disk is the greatest single aspect of the power saving potential of the overall system. Depending on the reliability of the overall system, the hard disk can be put to sleep for some time. However, the risk of losing data increases with the duration of the sleep periods. Other components can be deactivated via ACPI (at least theoretically) or permanently in the BIOS setup.

Processor Speed Control
In connection with the CPU, energy can be saved in three different ways: frequency and voltage scaling (also known as PowerNow! or Speedstep), throttling, and putting the processor to sleep (C states). Depending on the operating mode of the computer, these methods can also be combined.

21.2 APM

Some of the power saving functions are performed by the APM BIOS itself. On many laptops, standby and suspend states can be activated with key combinations or by closing the lid without any special operating system function. However, to activate these modes with a command, certain actions must be triggered before the system is suspended. To view the battery charge level, you need special program packages and a suitable kernel.

SUSE Linux kernels have built-in APM support. However, APM is only activated if ACPI is not implemented in the BIOS and an APM BIOS is detected. To activate APM support, ACPI must be disabled with `acpi=off` at the boot prompt. Enter `cat /proc/apm` to check if APM is active. An output consisting of various numbers indicates that everything is OK. You should now be able to shut down the computer with the command `shutdown -h`.

BIOS implementations that are not fully standard-compliant can cause problems with APM. Some problems can be circumvented with special boot parameters. All parameters are entered at the boot prompt in the form `apm=parameter`:

on or off
Enable or disable APM support.

(no-)allow-ints
Allow interrupts during the execution of BIOS functions.

(no-)broken-psr
The “GetPowerStatus” function of the BIOS does not work properly.

(no-)realmode-power-off
Reset processor to real mode prior to shutdown.

(no-)debug
Log APM events in system log.

(no-)power-off
Power system off after shutdown.

bounce-interval=*n*
Time in hundredths of a second after a suspend event during which additional suspend events are ignored.

idle-threshold=*n*
System inactivity percentage from which the BIOS function `idle` is executed (0=always, 100=never).

idle-period=*n*
Time in hundredths of a second after which the system activity is measured.

The APM daemon (apmd) is no longer used. Its functionality is now handled by the new powersaved, which also supports ACPI and CPU frequency scaling.

21.3 ACPI

ACPI (advanced configuration and power interface) was designed to enable the operating system to set up and control the individual hardware components. ACPI supersedes both PnP and APM. It delivers information about the battery, AC adapter, temperature, fan, and system events, like “close lid” or “battery low.”

The BIOS provides tables containing information about the individual components and hardware access methods. The operating system uses this information for tasks like assigning interrupts or activating and deactivating components. Because the operating system executes commands stored in the BIOS, the functionality depends on the BIOS implementation. The tables ACPI can detect and load are reported in `/var/log/boot.msg`. See Section 21.3.4, “Troubleshooting” (page 267) for more information about troubleshooting ACPI problems.

21.3.1 ACPI in Action

If the kernel detects an ACPI BIOS when the system is booted, ACPI is activated automatically and APM is deactivated. The boot parameter `acpi=force` may be necessary for some older machines. The computer must support ACPI 2.0 or later. Check the kernel boot messages in `/var/log/boot.msg` to see if ACPI was activated.

Subsequently, a number of modules must be loaded. This is done by the start script of the powersave daemon. If any of these modules cause problems, the respective module can be excluded from loading or unloading in `/etc/sysconfig/powersave/common`. The system log (`/var/log/messages`) contains the messages of the modules, enabling you to see which components were detected.

`/proc/acpi` now contains a number of files that provide information about the system state or can be used to change some of the states. Some features do not work yet because they are still under development and the support of some functions largely depends on the implementation of the manufacturer.

All files (except `dsdt` and `fadt`) can be read with `cat`. In some files, settings can be modified with `echo`, for example, `echo X > file` to specify suitable values for X. Always use the command `powersave` to access this information and control options. The following describes the most important files:

`/proc/acpi/info`
General information about ACPI.

`/proc/acpi/alarm`
Here, specify when the system should wake from a sleep state. Currently, this feature is not fully supported.

`/proc/acpi/sleep`
Provides information about possible sleep states.

`/proc/acpi/event`
All events are reported here and processed by the Powersave daemon (`powersaved`). If no daemon accesses this file, events, such as a brief click on the power button or closing the lid, can be read with `cat /proc/acpi/event` (terminate with Ctrl + C).

`/proc/acpi/dsdt` and `/proc/acpi/fadt`
These files contain the ACPI tables DSDT (differentiated system description table) and FADT (fixed ACPI description table). They can be read with `acpidmp`, `acpidisasm`, and `dmdecode`. These programs and their documentation are located in the package `pmtools`. For example, `acpidmp DSDT | acpidisasm`.

`/proc/acpi/ac_adapter/AC/state`
Shows whether the AC adapter is connected.

`/proc/acpi/battery/BAT*/{alarm,info,state}`
Detailed information about the battery state. The charge level is read by comparing the `last full capacity` from `info` with the `remaining capacity` from `state`. A more comfortable way to do this is to use one of the special programs introduced in Section 21.3.3, "ACPI Tools" (page 267). The charge level at which a battery event is triggered can be specified in `alarm`.

`/proc/acpi/button`
This directory contains information about various switches.

`/proc/acpi/fan/FAN/state`
Shows if the fan is currently active. Activate or deactivate the fan manually by writing `0` (on) or `3` (off) into this file. However, both the ACPI code in the kernel and the hardware (or the BIOS) overwrite this setting when it gets too warm.

`/proc/acpi/processor/*`
A separate subdirectory is kept for each CPU included in your system.

`/proc/acpi/processor/*/info`
Information about the energy saving options of the processor.

`/proc/acpi/processor/*/power`
Information about the current processor state. An asterisk next to `C2` indicates that the processor is idle. This is the most frequent state, as can be seen from the `usage` value.

`/proc/acpi/processor/*/throttling`
Can be used to set the throttling of the processor clock. Usually, throttling is possible in eight levels. This is independent of the frequency control of the CPU.

/proc/acpi/processor/*/limit

If the performance (outdated) and the throttling are automatically controlled by a daemon, the maximum limits can be specified here. Some of the limits are determined by the system. Some can be adjusted by the user.

/proc/acpi/thermal_zone/

A separate subdirectory exists for every thermal zone. A thermal zone is an area with similar thermal properties whose number and names are designated by the hardware manufacturer. However, many of the possibilities offered by ACPI are rarely implemented. Instead, the temperature control is handled conventionally by the BIOS. The operating system is not given much opportunity to intervene, because the life span of the hardware is at stake. Therefore, some of the files only have a theoretical value.

/proc/acpi/thermal_zone/*/temperature

Current temperature of the thermal zone.

/proc/acpi/thermal_zone/*/state

The state indicates if everything is `ok` or if ACPI applies `active` or `passive` cooling. In the case of ACPI-independent fan control, this state is always `ok`.

/proc/acpi/thermal_zone/*/cooling_mode

Select the cooling method controlled by ACPI. Choose from passive (less performance, economical) or active cooling mode (full performance, fan noise).

/proc/acpi/thermal_zone/*/trip_points

Enables the determination of temperature limits for triggering specific actions, like passive or active cooling, suspension (`hot`), or a shutdown (`critical`). The possible actions are defined in the DSDT (device-dependent). The trip points determined in the ACPI specification are `critical`, `hot`, `passive`, `active1`, and `active2`. Even if not all of them are implemented, they must always be entered in this file in this order. For example, the entry `echo 90:0:70:0:0 > trip_points` sets the temperature for `critical` to `90` and the temperature for `passive` to `70` (all temperatures measured in degrees Celsius).

/proc/acpi/thermal_zone/*/polling_frequency

If the value in `temperature` is not updated automatically when the temperature changes, toggle the polling mode here. The command `echo X > /proc/acpi/thermal_zone/*/polling_frequency` causes the temperature to be queried every `X` seconds. Set `X=0` to disable polling.

None of these settings, information, and events need to be edited manually. This can be done with the Powersave daemon (powersaved) and various applications, like powersave, kpowersave, and wmpowersave. See Section 21.3.3, “ACPI Tools” (page 267). Because powersaved covers the functionalities of the older acpid, acpid is no longer needed.

21.3.2 Controlling the CPU Performance

The CPU can save energy in three ways. Depending on the operating mode of the computer, these methods can be combined. Saving energy also means that the system heats up less and the fans are activated less frequently.

Frequency and Voltage Scaling

PowerNow! and Speedstep are the designations AMD and Intel use for this technology. However, this technology is also applied in processors of other manufacturers. The clock frequency of the CPU and its core voltage are reduced at the same time, resulting in more than linear energy savings. This means that when the frequency is halved (half performance), far less than half of the energy is consumed. This technology is independent from APM or ACPI and requires a daemon that adapts the frequency and the current need for performance. The settings can be made in the directory `/sys/devices/system/cpu/cpu*/cpufreq/`.

Throttling the Clock Frequency

This technology omits a certain percentage of the clock signal impulses for the CPU. At 25% throttling, every fourth impulse is omitted. At 87.5%, only every eighth impulse reaches the processor. However, the energy savings are a little less than linear. Normally, throttling is only used if frequency scaling is not available or to maximize power savings. This technology, too, must be controlled by a special process. The system interface is `/proc/acpi/processor/*/throttling`.

Putting the Processor to Sleep

The operating system puts the processor to sleep whenever there is nothing to do. In this case, the operating system sends the CPU a `halt` command. There are three states: C1, C2, and C3. In the most economic state, C3, even the synchronization of the processor cache with the main memory is halted. Therefore, this state can only be applied if no other device modifies the contents of the main memory via bus master activity. Some drivers prevent the use of C3. The current state is displayed in `/proc/acpi/processor/*/power`.

Frequency scaling and throttling are only relevant if the processor is busy, because the most economic C state is applied anyway when the processor is idle. If the CPU is busy, frequency scaling is the recommended power saving method. Often the processor only works with a partial load. In this case, it can be run with a lower frequency. Usually, dynamic frequency scaling controlled by a daemon, such as powersaved, is the best approach. A static setting to a low frequency is useful for battery operation or if you want the computer to be cool or quiet.

Throttling should be used as the last resort, for example, to extend the battery operation time despite a high system load. However, some systems do not run smoothly when they are throttled too much. Moreover, CPU throttling does not make sense if the CPU has little to do.

In SUSE Linux these technologies are controlled by the powersave daemon. The configuration is explained in Section 21.5, “The powersave Package” (page 270).

21.3.3 ACPI Tools

The range of more or less comprehensive ACPI utilities includes tools that merely display information, like the battery charge level and the temperature (acpi, klaptopdaemon, wmacpimon, etc.), tools that facilitate the access to the structures in `/proc/acpi` or that assist in monitoring changes (akpi, acpiw, gtkacpiw), and tools for editing the ACPI tables in the BIOS (package `pmtools`).

21.3.4 Troubleshooting

There are two different types of problems. On one hand, the ACPI code of the kernel may contain bugs that were not detected in time. In this case, a solution will be made available for download. More often, however, the problems are caused by the BIOS. Sometimes, deviations from the ACPI specification are purposely integrated in the BIOS to circumvent errors in the ACPI implementation in other widespread operating systems. Hardware components that have serious errors in the ACPI implementation are recorded in a blacklist that prevents the Linux kernel from using ACPI for these components.

The first thing to do when problems are encountered is to update the BIOS. If the computer does not boot at all, one of the following boot parameters may be helpful:

pci=noacpi

Do not use ACPI for configuring the PCI devices.

acpi=oldboot

Only perform a simple resource configuration. Do not use ACPI for other purposes.

acpi=off

Disable ACPI.

WARNING: Problems Booting without ACPI

Some newer machines (especially SMP systems and AMD64 systems) need ACPI for configuring the hardware correctly. On these machines, disabling ACPI can cause problems.

Monitor the boot messages of the system with the command `dmesg | grep -2i acpi` (or all messages, because the problem may not be caused by ACPI) after booting. If an error occurs while parsing an ACPI table, the most important table—the DSDT—can be replaced with an improved version. In this case, the faulty DSDT of the BIOS is ignored. The procedure is described in Section 21.5.4, "Troubleshooting" (page 276).

In the kernel configuration, there is a switch for activating ACPI debug messages. If a kernel with ACPI debugging is compiled and installed, experts searching for an error can be supported with detailed information.

If you experience BIOS or hardware problems, it is always advisable to contact the manufacturers. Especially if they do not always provide assistance for Linux, they should be confronted with the problems. Manufacturers will only take the issue seriously if they realize that an adequate number of their customers use Linux.

For More Information

Additional documentation and help on ACPI:

- `http://www.cpqlinux.com/acpi-howto.html` (detailed ACPI HOWTO, contains DSDT patches)

- `http://www.intel.com/technology/iapc/acpi/faq.htm` (ACPI FAQ @Intel)

- `http://acpi.sourceforge.net/` (the ACPI4Linux project at Sourceforge)
- `http://www.poupinou.org/acpi/` (DSDT patches by Bruno Ducrot)

21.4 Rest for the Hard Disk

In Linux, the hard disk can be put to sleep entirely if it is not needed or it can be run in a more economic or quieter mode. On modern laptops, you do not need to switch off the hard disks manually, because they automatically enter an economic operating mode whenever they are not needed. However, if you want to maximize power savings, test some of the following methods. Most of the functions can be controlled with powersaved and the YaST power management module, which is discussed in further detail in Section 21.6, "The YaST Power Management Module" (page 278).

The hdparm application can be used to modify various hard disk settings. The option `-y` instantly switches the hard disk to the standby mode. `-Y` puts it to sleep. `hdparm -S x` causes the hard disk to be spun down after a certain period of inactivity. Replace *x* as follows: `0` disables this mechanism, causing the hard disk to run continuously. Values from `1` to `240` are multiplied by 5 seconds. Values from `241` to `251` correspond to 1 to 11 times 30 minutes.

Internal power saving options of the hard disk can be controlled with the option `-B`. Select a value from `0` to `255` for maximum saving to maximum throughput. The result depends on the hard disk used and is difficult to assess. To make a hard disk quieter, use the option `-M`. Select a value from `128` to `254` for quiet to fast.

Often, it is not so easy to put the hard disk to sleep. In Linux, numerous processes write to the hard disk, waking it up repeatedly. Therefore, it is important to understand how Linux handles data that needs to be written to the hard disk. First, all data is buffered in the RAM. This buffer is monitored by the kernel update daemon (kupdated). When the data reaches a certain age limit or when the buffer is filled to a certain degree, the buffer content is flushed to the hard disk. The buffer size is dynamic and depends on the size of the memory and the system load. By default, kupdated is set to short intervals to achieve maximum data integrity. It checks the buffer every 5 seconds and notifies the bdflush daemon when data is older than 30 seconds or the buffer reaches a fill level of 30%. The bdflush daemon then writes the data to the hard disk. It also writes independently from kupdated if, for instance, the buffer is full.

WARNING: Impairment of the Data Integrity

Changes to the kernel update daemon settings endanger the data integrity.

Apart from these processes, journaling file systems, like ReiserFS and Ext3, write their metadata independently from bdflush, which also prevents the hard disk from spinning down. To avoid this, a special kernel extension has been developed for mobile devices. See `/usr/src/linux/Documentation/laptop-mode.txt` for details.

Another important factor is the way active programs behave. For example, good editors regularly write hidden backups of the currently modified file to the hard disk, causing the disk to wake up. Features like this can be disabled at the expense of data integrity.

In this connection, the mail daemon postfix makes use of the variable `POSTFIX_LAPTOP`. If this variable is set to `yes`, postfix accesses the hard disk far less frequently. However, this is irrelevant if the interval for kupdated was increased.

21.5 The powersave Package

The `powersave` package is responsible for the power saving function in laptops during battery operation. Some of its features are also useful for normal workstations and servers, such as suspend, standby, ACPI button functionality, and putting IDE hard disks to sleep.

This package contains all power management features of your computer. It supports hardware using ACPI, APM, IDE hard disks, and PowerNow! or SpeedStep technologies. The functionalities from the packages `apmd`, `acpid`, `ospmd`, and `cpufreqd` (now `cpuspeed`) have been consolidated in the `powersave` package. Daemons from these packages should not be run concurrently with the powersave daemon.

Even if your system does not contain all the hardware elements listed above, use the powersave daemon for controlling the power saving function. Because ACPI and APM are mutually exclusive, you can only use one of these systems on your computer. The daemon automatically detects any changes in the hardware configuration.

21.5.1 Configuring the powersave Package

Normally, the configuration of powersave is distributed to several files:

/etc/sysconfig/powersave/common

This file contains general settings for the powersave daemon. For example, the amount of debug messages in /var/log/messages can be increased by increasing the value of the variable DEBUG.

/etc/sysconfig/powersave/events

The powersave daemon needs this file for processing system events. An event can be assigned external actions or actions performed by the daemon itself. For external actions, the daemon tries to run an executable file in /usr/lib/powersave/scripts/. Predefined internal actions:

- ignore
- throttle
- dethrottle
- suspend_to_disk
- suspend_to_ram
- standby
- do_suspend_to_disk
- do_suspend_to_ram
- do_standby

throttle slows down the processor by the value defined in MAX_THROTTLING. This value depends on the current scheme. dethrottle sets the processor to full performance. suspend_to_disk, suspend_to_ram, and standby trigger the system event for a sleep mode. These three actions are generally responsible for triggering the sleep mode, but they should always be associated with specific system events.

The directory /usr/lib/powersave/scripts contains scripts for processing events:

notify
Notification about an event by way of the console, X window, or acoustic signal.

screen_saver
Activates the screen saver.

switch_vt
Useful if the screen is displaced after a suspend or standby.

wm_logout
Saves the settings and logs out from GNOME, KDE, or other window managers.

wm_shutdown
Saves the GNOME or KDE settings and shuts down the system.

If, for example, the variable EVENT_GLOBAL_SUSPEND2DISK="prepare_suspend_to_disk do_suspend_to_disk" is set, the two scripts or actions are processed in the specified order as soon as the user gives powersaved the command for the sleep mode suspend to disk. The daemon runs the external script /usr/lib/powersave/scripts/prepare_suspend_to_disk. After this script has been processed successfully, the daemon runs the internal action do_suspend_to_disk and sets the computer to the sleep mode after the script has unloaded critical modules and stopped services.

The actions for the event of a sleep button could be modified as in EVENT_BUTTON_SLEEP="notify suspend_to_disk". In this case, the user is informed about the suspend by the external script notify. Subsequently, the event EVENT_GLOBAL_SUSPEND2DISK is generated, resulting in the execution of the mentioned actions and a secure system suspend mode. The script notify can be customized using the variable NOTIFY_METHOD in /etc/sysconfig/powersave/common.

/etc/sysconfig/powersave/cpufreq
Contains variables for optimizing the dynamic CPU frequency settings.

/etc/sysconfig/powersave/battery
Contains battery limits and other battery-specific settings.

`/etc/sysconfig/powersave/sleep`

In this file, activate the sleep modes and determine which critical modules should be unloaded and which services should be stopped prior to a suspend or standby event. When the system is resumed, these modules are reloaded and the services are restarted. You can even delay a triggered sleep mode, for example, to save files. The default settings mainly concern USB and PCMCIA modules. A failure of suspend or standby is usually caused by certain modules. See Section 21.5.4, "Troubleshooting" (page 276) for more information about identifying the error.

`/etc/sysconfig/powersave/thermal`

Activates cooling and thermal control. Details about this subject are available in the file `/usr/share/doc/packages/powersave/README.thermal`.

`/etc/sysconfig/powersave/scheme_*`

These are the various schemes that adapt the power consumption to certain deployment scenarios. A number of schemes are preconfigured and can be used as they are. Custom schemes can be saved here.

21.5.2 Configuring APM and ACPI

Suspend and Standby

By default, the sleep modes are inactive, because they still do not work on some computers. There are three basic ACPI sleep modes and two APM sleep modes:

Suspend to Disk (ACPI S4, APM suspend)

Saves the entire memory content to the hard disk. The computer is switched off completely and does not consume any power.

Suspend to RAM (ACPI S3, APM suspend)

Saves the states of all devices to the main memory. Only the main memory continues consuming power.

Standby (ACPI S1, APM standby)

Switches some devices off (manufacturer-dependent).

Make sure that the following default options are set in the file `/etc/sysconfig/powersave/events` for the correct processing of suspend, standby, and resume (default settings following the installation of SUSE Linux):

```
EVENT_GLOBAL_SUSPEND2DISK=
    "prepare_suspend_to_disk do_suspend_to_disk"
EVENT_GLOBAL_SUSPEND2RAM=
    "prepare_suspend_to_ram do_suspend_to_ram"
EVENT_GLOBAL_STANDBY=
    "prepare_standby do_standby"
EVENT_GLOBAL_RESUME_SUSPEND2DISK=
    "restore_after_suspend_to_disk"
EVENT_GLOBAL_RESUME_SUSPEND2RAM=
    "restore_after_suspend_to_ram"
EVENT_GLOBAL_RESUME_STANDBY=
    "restore_after_standby"
```

Custom Battery States

In the file /etc/sysconfig/powersave/battery, define three battery charge levels (in percent) that trigger system alerts or specific actions when they are reached.

```
BATTERY_WARNING=20
BATTERY_LOW=10
BATTERY_CRITICAL=5
```

The actions or scripts to execute when the charge levels drop under the specified limits are defined in the configuration file /etc/sysconfig/powersave/events. The standard actions for buttons can be modified as described in Section 21.5.1, “Configuring the powersave Package” (page 271).

```
EVENT_BATTERY_NORMAL="ignore"
EVENT_BATTERY_WARNING="notify"
EVENT_BATTERY_LOW="notify"
EVENT_BATTERY_CRITICAL="wm_shutdown"
```

Adapting Power Consumption to Various Conditions

The system behavior can be adapted to the type of power supply. The power consumption of the system should be reduced when the system is disconnected from the AC power supply and operated with the battery. Similarly, the performance should automatically increase as soon as the system is connected to the AC power supply. The CPU frequency, the power saving function of IDE, and a number of other parameters can be modified.

The actions to execute when the computer is disconnected from or connected to the AC power supply are defined in /etc/sysconfig/powersave/events. Select the schemes to use in /etc/sysconfig/powersave/common:

```
AC_SCHEME="performance"
BATTERY_SCHEME="powersave"
```

The schemes are stored in files in `/etc/sysconfig/powersave`. The filenames are in the format `scheme_name-of-the-scheme`. The example refers to two schemes: `scheme_performance` and `scheme_powersave`. `performance`, `powersave`, `presentation`, and `acoustic` are preconfigured. Existing schemes can be edited, created, deleted, or associated with different power supply states with the help of the YaST power management module described in Section 21.6, "The YaST Power Management Module" (page 278).

21.5.3 Additional ACPI Features

If you use ACPI, you can control the response of your system to *ACPI buttons* (power, sleep, lid open, and lid closed). Configure execution of the actions in `/etc/sysconfig/powersave/events`. Refer to this configuration file for an explanation of the individual options.

EVENT_BUTTON_POWER="wm_shutdown"
When the power button is pressed, the system responds by shutting down the respective window manager (KDE, GNOME, fvwm, etc.).

EVENT_BUTTON_SLEEP="suspend_to_disk"
When the sleep button is pressed, the system is set to the suspend-to-disk mode.

EVENT_BUTTON_LID_OPEN="ignore"
Nothing happens when the lid is opened.

EVENT_BUTTON_LID_CLOSED="screen_saver"
When the lid is closed, the screen saver is activated.

Further throttling of the CPU performance is possible if the CPU load does not exceed a specified limit for a specified time. Specify the load limit in `PROCESSOR_IDLE_LIMIT` and the time-out in `CPU_IDLE_TIMEOUT`. If the CPU load stays below the limit longer than the time-out, the event configured in `EVENT_PROCESSOR_IDLE` is activated. If the CPU is busy again, `EVENT_PROCESSOR_BUSY` is executed.

21.5.4 Troubleshooting

All error messages and alerts are logged in the file /var/log/messages. If you cannot find the needed information, increase the verbosity of the messages of powersave using DEBUG in the file /etc/sysconfig/powersave/common. Increase the value of the variable to 7 or even 15 and restart the daemon. The more detailed error messages in /var/log/messages should help you to find the error. The following sections cover the most common problems with powersave.

ACPI Activated with Hardware Support but Functions Do Not Work

If you experience problems with ACPI, use the command dmesg|grep -i acpi to search the output of dmesg for ACPI-specific messages. A BIOS update may be required to resolve the problem. Go to the home page of your laptop manufacturer, look for an updated BIOS version, and install it. Ask the manufacturer to comply with the latest ACPI specification. If the errors persist after the BIOS update, proceed as follows to replace the faulty DSDT table in your BIOS with an updated DSDT:

1. Download the DSDT for your system from http://acpi.sourceforge.net/dsdt/tables. Check if the file is decompressed and compiled as shown by the file extension .aml (ACPI machine language). If this is the case, continue with step 3.

2. If the file extension of the downloaded table is .asl (ACPI source language), compile it with iasl (package pmtools). Enter the command iasl -sa file.asl. The latest version of iasl (Intel ACPI compiler) is available at http://developer.intel.com/technology/iapc/acpi/downloads.htm.

3. Copy the file DSDT.aml to any location (/etc/DSDT.aml is recommended). Edit /etc/sysconfig/kernel and adapt the path to the DSDT file accordingly. Start mkinitrd (package mkinitrd). Whenever you install the kernel and use mkinitrd to create an initrd, the modified DSDT is integrated and loaded when the system is booted.

CPU Frequency Does Not Work

Refer to the kernel sources (`kernel-source`) to see if your processor is supported. You may need a special kernel module or module option to activate CPU frequency control. This information is available in `/usr/src/linux/Documentation/cpu-freq/*`. If a special module or module option is needed, configure it in the file `/etc/sysconfig/powersave/cpufreq` by means of the variables `CPUFREQD_MODULE` and `CPUFREQD_MODULE_OPTS`.

Suspend and Standby Do Not Work

There are several kernel-related problems that prevent the use of suspend and standby on ACPI systems:

- Currently, systems with more than 1 GB RAM do not support suspend.

- Currently, multiprocessor systems and systems with a P4 processor (with hyper-threading) do not support suspend.

The error may also be due to a faulty DSDT implementation (BIOS). If this is the case, install a new DSDT.

On ACPI and APM systems: When the system tries to unload faulty modules, the system is arrested or the suspend event is not triggered. The same can also happen if you do not unload modules or stop services that prevent a successful suspend. In both cases, try to identify the faulty module that prevented the sleep mode. The log files generated by the powersave daemon in `/var/log/suspend2ram.log` and `/var/log/suspend2disk.log` are very helpful in this regard. If the computer does not enter the sleep mode, the cause lies in the last module unloaded. Manipulate the following settings in `/etc/sysconfig/powersave/sleep` to unload problematic modules prior to a suspend or standby.

```
UNLOAD_MODULES_BEFORE_SUSPEND2DISK=""
UNLOAD_MODULES_BEFORE_SUSPEND2RAM=""
UNLOAD_MODULES_BEFORE_STANDBY=""
SUSPEND2DISK_RESTART_SERVICES=""
SUSPEND2RAM_RESTART_SERVICES=""
STANDBY_RESTART_SERVICES=""
```

If you use suspend or standby in changing network environments or in connection with remotely mounted file systems, such as Samba and NIS, use automounter to mount them or add the respective services, for example, smbfs or nfs, in the above-mentioned variable. If an application accesses the remotely mounted file system prior to a suspend or standby, the service cannot be stopped correctly and the file system cannot be unmounted properly. After resuming the system, the file system may be corrupt and must be remounted.

Using ACPI, Powersave Does Not Notice Battery Limits

With ACPI, the operating system can request the BIOS to send a message when the battery charge level drops under a certain limit. The advantage of this method is that the battery state does not need to be polled constantly, which would impair the performance of the computer. However, this notification may not take place when the charge level drops under the specified limit, even though the BIOS supposedly supports this feature. If this happens on your system, set the variable FORCE_BATTERY_POLLING in the file /etc/sysconfig/powersave/battery to yes to force battery polling.

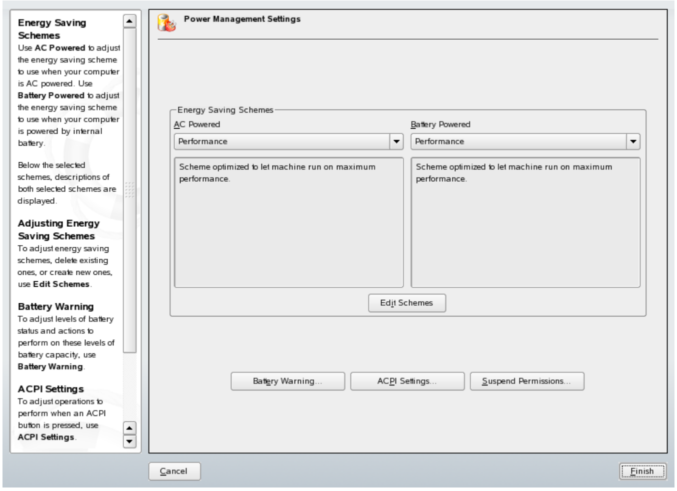
21.5.5 For More Information

Information about the powersave package is also available in /usr/share/doc/packages/powersave.

21.6 The YaST Power Management Module

The YaST power management module can configure all power management settings already described. When started from the YaST Control Center with *System* → *Power Management*, the first dialog of the module opens. It is shown in Figure 21.1, “Scheme Selection” (page 279).

Figure 21.1 *Scheme Selection*



Energy Saving Schemes
Use **AC Powered** to adjust the energy saving scheme to use when your computer is AC powered. Use **Battery Powered** to adjust the energy saving scheme to use when your computer is powered by internal battery.

Below the selected schemes, descriptions of both selected schemes are displayed.

Adjusting Energy Saving Schemes
To adjust energy saving schemes, delete existing ones, or create new ones, use **Edit Schemes**.

Battery Warning
To adjust levels of battery status and actions to perform on these levels of battery capacity, use **Battery Warning**.

ACPI Settings
To adjust operations to perform when an ACPI button is pressed, use **ACPI Settings**.

Power Management Settings

Energy Saving Schemes

AC Powered: Performance

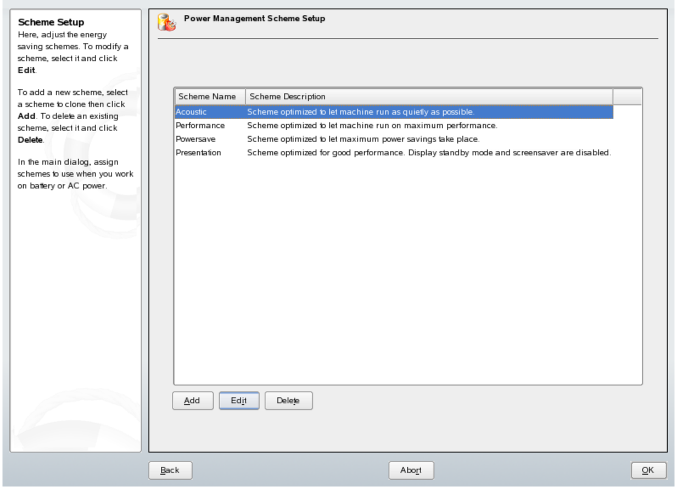
Scheme optimized to let machine run on maximum performance.

Battery Powered: Performance

Scheme optimized to let machine run on maximum performance.

In this dialog, select the schemes to use for battery operation and AC operation. To add or modify the schemes, click *Edit Schemes*, which opens an overview of the existing schemes like that shown in Figure 21.2, “Overview of Existing Schemes” (page 279).

Figure 21.2 *Overview of Existing Schemes*



Scheme Setup
Here, adjust the energy saving schemes. To modify a scheme, select it and click **Edit**.

To add a new scheme, select a scheme to clone then click **Add**. To delete an existing scheme, select it and click **Delete**.

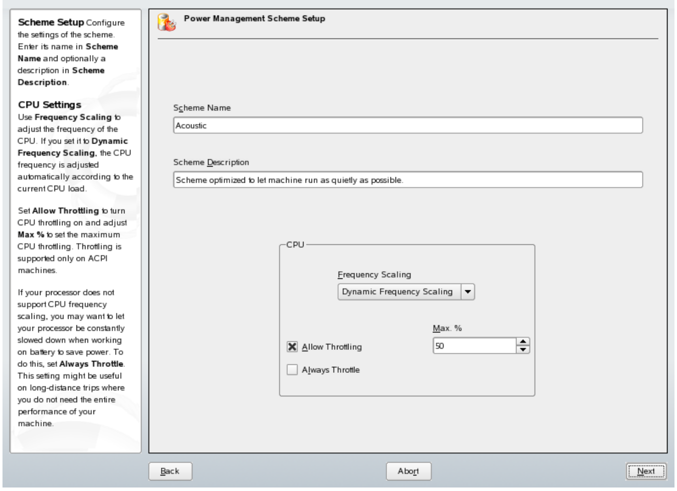
In the main dialog, assign schemes to use when you work on battery or AC power.

Power Management Scheme Setup

Scheme Name	Scheme Description
Acoustic	Scheme optimized to let machine run as quietly as possible.
Performance	Scheme optimized to let machine run on maximum performance.
Powersave	Scheme optimized to let maximum power savings take place.
Presentation	Scheme optimized for good performance. Display standby mode and screensaver are disabled.

In the scheme overview, select the scheme to modify then click *Edit*. To create a new scheme, click *Add*. The dialog that opens is the same in both cases and is shown in Figure 21.3, "Configuring a Scheme" (page 280).

Figure 21.3 *Configuring a Scheme*



Scheme Setup Configure the settings of the scheme. Enter its name in **Scheme Name** and optionally a description in **Scheme Description**.

CPU Settings
Use **Frequency Scaling** to adjust the frequency of the CPU. If you set it to **Dynamic Frequency Scaling**, the CPU frequency is adjusted automatically according to the current CPU load.

Set **Allow Throttling** to turn CPU throttling on and adjust **Max %** to set the maximum CPU throttling. Throttling is supported only on ACPI machines.

If your processor does not support CPU frequency scaling, you may want to let your processor be constantly slowed down when working on battery to save power. To do this, set **Always Throttle**. This setting might be useful on long-distance trips where you do not need the entire performance of your machine.

Power Management Scheme Setup

Scheme Name: Acoustic

Scheme Description: Scheme optimized to let machine run as quietly as possible.

CPU

Frequency Scaling: Dynamic Frequency Scaling

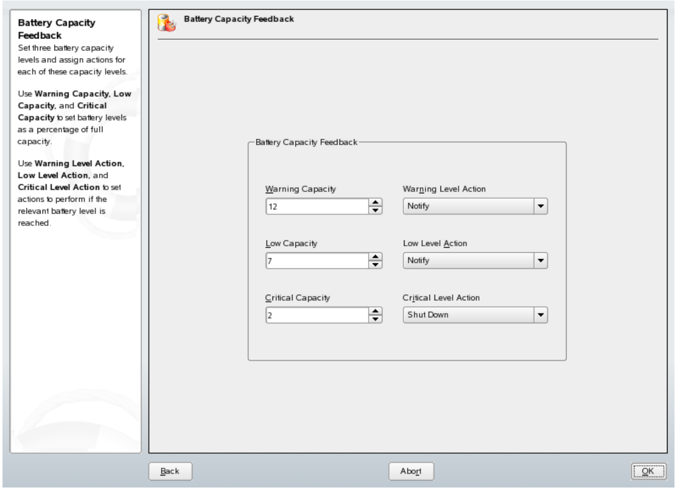
☒ Allow Throttling — Max. %: 50

☐ Always Throttle

First, enter a suitable name and description for the new or edited scheme. Determine if and how the CPU performance should be controlled for this scheme. Decide if and to what extent frequency scaling and throttling should be used. In the following dialog for the hard disk, define a *Standby Policy* for maximum performance or for energy saving. The *Acoustic Policy* controls the noise level of the hard disk (supported by few hard disks). The *Cooling Policy* determines the cooling method to use. Unfortunately, this type of thermal control is rarely supported by the BIOS. Read `/usr/share/doc/packages/powersave/README.thermal` to learn how you can use the fan and passive cooling methods.

Global power management settings can also be made from the initial dialog using *Battery Warnings*, *ACPI Settings*, or *Enable Suspend*. Click *Battery Warnings* to access the dialog for the battery charge level, shown in Figure 21.4, "Battery Charge Level" (page 281).

Figure 21.4 Battery Charge Level



Battery Capacity Feedback
Set three battery capacity levels and assign actions for each of these capacity levels.

Use **Warning Capacity**, **Low Capacity**, and **Critical Capacity** to set battery levels as a percentage of full capacity.

Use **Warning Level Action**, **Low Level Action**, and **Critical Level Action** to set actions to perform if the relevant battery level is reached.

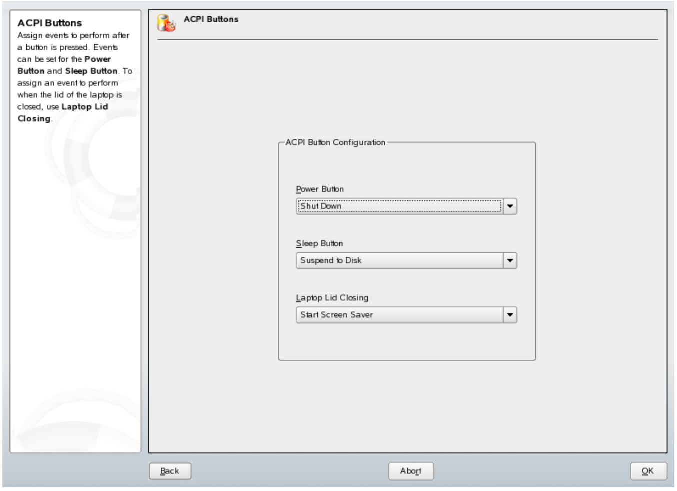
Battery Capacity Feedback

Battery Capacity Feedback

Warning Capacity: 12
Warning Level Action: Notify
Low Capacity: 7
Low Level Action: Notify
Critical Capacity: 2
Critical Level Action: Shut Down

The BIOS of your system notifies the operating system whenever the charge level drops under certain configurable limits. In this dialog, define three limits: *Warning Capacity*, *Low Capacity*, and *Critical Capacity*. Specific actions are triggered when the charge level drops under these limits. Usually, the first two states merely trigger a notification to the user. The third critical level triggers a shutdown, because the remaining energy is not sufficient for continued system operation. Select suitable charge levels and the desired actions then click *OK* to return to the start dialog.

Figure 21.5 *ACPI Settings*



ACPI Buttons
Assign events to perform after a button is pressed. Events can be set for the **Power Button** and **Sleep Button**. To assign an event to perform when the lid of the laptop is closed, use **Laptop Lid Closing**.

ACPI Buttons

ACPI Button Configuration

Power Button: Shut Down

Sleep Button: Suspend to Disk

Laptop Lid Closing: Start Screen Saver

Access the dialog for configuring the ACPI buttons using *ACPI Settings*. It is shown in Figure 21.5, "ACPI Settings" (page 282). The settings for the ACPI buttons determine how the system should respond to certain switches. Configure the system response to pressing the power button, pressing the sleep button, and closing the laptop lid. Click *OK* to complete the configuration and return to the start dialog.

Click *Enable Suspend* to enter a dialog in which to determine if and how users of this system may use the suspend or standby functionality. Click *OK* to return to the main dialog. Click *OK* again to exit the module and confirm your power management settings.

22 Wireless Communication

There are several possibilities for using your Linux system to communicate with other computers, cellular phones, or peripheral devices. WLAN (wireless LAN) can be used to network laptops. Bluetooth can be used to connect individual system components (mouse, keyboard), peripheral devices, cellular phones, PDAs, and individual computers with each other. IrDA is mostly used for communication with PDAs or cellular phones. This chapter introduces all three technologies and their configuration.

22.1 Wireless LAN

Wireless LANs have become an indispensable aspect of mobile computing. Today, most laptops have built-in WLAN cards. The 802.11 standard for the wireless communication of WLAN cards was prepared by the IEEE organization. Originally, this standard provided for a maximum transmission rate of 2 MBit/s. Meanwhile, several supplements have been added to increase the data rate. These supplements define details such as the modulation, transmission output, and transmission rates:

***Table 22.1** Overview of Various WLAN Standards*

Name	Band (GHz)	Maximum Transmission Rate (MBit/s)	Note
802.11	2.4	2	Outdated; virtually no end devices available

Name	Band (GHz)	Maximum Transmission Rate (MBit/s)	Note
802.11b	2.4	11	Widespread
802.11a	5	54	Less common
802.11g	2.4	54	Backward-compatible with 11b

Additionally, there are proprietary standards, like the 802.11b variation of Texas Instruments with a maximum transmission rate of 22 MBit/s (sometimes referred to as 802.11b+). However, the popularity of cards using this standard is limited.

22.1.1 Hardware

802.11 cards are not supported by SUSE Linux. Most cards using 802.11a, 802.11b, and 802.11g are supported. New cards usually comply with the 802.11g standard, but cards using 802.11b are still available. Normally, cards with the following chips are supported:

- Aironet 4500, 4800
- Atheros 5210, 5211, 5212
- Atmel at76c502, at76c503, at76c504, at76c506
- Intel PRO/Wireless 2100, 2200BG, 2915ABG
- Intersil Prism2/2.5/3
- Intersil PrismGT
- Lucent/Agere Hermes
- Ralink RT2400, RT2500
- Texas Instruments ACX100, ACX111

- ZyDAS zd1201

A number of older cards that are rarely used and no longer available are also supported. An extensive list of WLAN cards and the chips they use is available at the Web site of *AbsoluteValue Systems* at `http://www.linux-wlan.org/docs/wlan_adapters.html.gz`. `http://wiki.uni-konstanz.de/wiki/bin/view/Wireless/ListeChipsatz` provides an overview of the various WLAN chips.

Some cards need a firmware image that must be loaded into the card when the driver is initialized. This is the case with Intersil PrismGT, Atmel, and TI ACX100 and ACX111. The firmware can easily be installed with the YaST Online Update. The firmware for Intel PRO/Wireless cards ships with SUSE Linux and is automatically installed by YaST as soon as a card of this type is detected. More information about this subject is available in the installed system in `/usr/share/doc/packages/wireless-tools/README.firmware`.

Cards without native Linux support can be used by running the ndiswrapper application. ndiswrapper uses the Windows drivers that are shipped together with most WLAN cards. A description of ndiswrapper can be found under `/usr/share/doc/packages/ndiswrapper/README.SUSE` when the package `ndiswrapper` is installed. For in-depth information about ndiswrapper, refer to the project's Web site at `http://ndiswrapper.sourceforge.net/support.html`.

22.1.2 Function

In wireless networking, various techniques and configurations are used to ensure fast, high-quality, and secure connections. Different operating types suit different setups. It can be difficult to choose the right authentication method. The available encryption methods have different advantages and pitfalls.

Operating Mode

Basically, wireless networks can be classified as managed networks and ad-hoc networks. Managed networks have a managing element: the access point. In this mode (also referred to as infrastructure mode), all connections of the WLAN stations in the network run over the access point, which may also serve as a connection to an ethernet. Ad-hoc networks do not have an access point. The stations communicate directly with each other. The transmission range and number of participating stations are greatly limited in ad-hoc networks. Therefore, an access point is usually more efficient. It is even possible to use a WLAN card as an access point. Most cards support this functionality.

Because a wireless network is much easier to intercept and compromise than a wired network, the various standards include authentication and encryption methods. In the original version of the IEEE 802.11 standard, these are described under the term WEP. However, because WEP has proven to be insecure (see Section “Security” (page 292)), the WLAN industry (joined under the name *Wi-Fi Alliance*) has defined a new extension called WPA, which is supposed to eliminate the weaknesses of WEP. The later IEEE 802.11i standard (also referred to as WPA2, because WPA is based on a draft version 802.11i) includes WPA and some other authentication and encryption methods.

Authentication

To make sure that only authorized stations can connect, various authentication mechanisms are used in managed networks:

Open

An open system is a system that does not require authentication. Any station can join the network. Nevertheless, WEP encryption (see Section “Encryption” (page 287)) can be used.

Shared Key (according to IEEE 802.11)

In this procedure, the WEP key is used for the authentication. However, this procedure is not recommended, because it makes the WEP key more susceptible to attacks. All an attacker needs to do is to listen long enough to the communication between the station and the access point. During the authentication process, both sides exchange the same information, once in encrypted form and once in unencrypted form. This makes it possible for the key to be reconstructed with suitable tools. Because this method makes use of the WEP key for the authentication and for the encryption, it does not enhance the security of the network. A station that has the correct WEP key can authenticate, encrypt, and decrypt. A station that does not have the key cannot decrypt received packets. Accordingly, it cannot communicate, regardless of whether it had to authenticate itself.

WPA-PSK (according to IEEE 802.1x)

WPA-PSK (PSK stands for preshared key) works similarly to the Shared Key procedure. All participating stations as well as the access point need the same key. The key is 256 bits in length and is usually entered as a passphrase. This system does not need a complex key management like WPA-EAP and is more suitable for private use. Therefore, WPA-PSK is sometimes referred to as WPA “Home”.

WPA-EAP (according to IEEE 802.1x)

Actually, WPA-EAP is not an authentication system but a protocol for transporting authentication information. WPA-EAP is used to protect wireless networks in enterprises. In private networks, it is scarcely used. For this reason, WPA-EAP is sometimes referred to as WPA “Enterprise”.

WPA-EAP needs a Radius server to authenticate users. EAP offers three different methods for connecting and authenticating to the server: TLS (Transport Layer Security), TTLS (Tunneled Transport Layer Security), and PEAP (Protected Extensible Authentication Protocol). In a nutshell, these options work as follows:

EAP-TLS

TLS authentication relies on the mutual exchange of certificates both for server and client. First, the server presents its certificate to the client where it is evaluated. If the certificate is considered valid, the client in turn presents its certificate to the server. While TLS is secure, it requires a working certification management infrastructure in your network. This infrastructure is rarely found in private networks.

EAP-TTLS and PEAP

Both TTLS and PEAP are two-stage protocols. In the first stage, a secure is established and in the second one the client authentication data is exchanged. They require far less certification management overhead than TLS, if any.

Encryption

There are various encryption methods to ensure that no unauthorized person can read the data packets that are exchanged in a wireless network or gain access to the network:

WEP (defined in IEEE 802.11)

This standard makes use of the RC4 encryption algorithm, originally with a key length of 40 bits, later also with 104 bits. Often, the length is declared as 64 bits or 128 bits, depending on whether the 24 bits of the initialization vector are included. However, this standard has some weaknesses. Attacks against the keys generated by this system may be successful. Nevertheless, it is better to use WEP than not encrypt the network at all.

TKIP (defined in WPA/IEEE 802.11i)

This key management protocol defined in the WPA standard uses the same encryption algorithm as WEP, but eliminates its weakness. Because a new key is generated for every data packet, attacks against these keys are in vain. TKIP is used together with WPA-PSK.

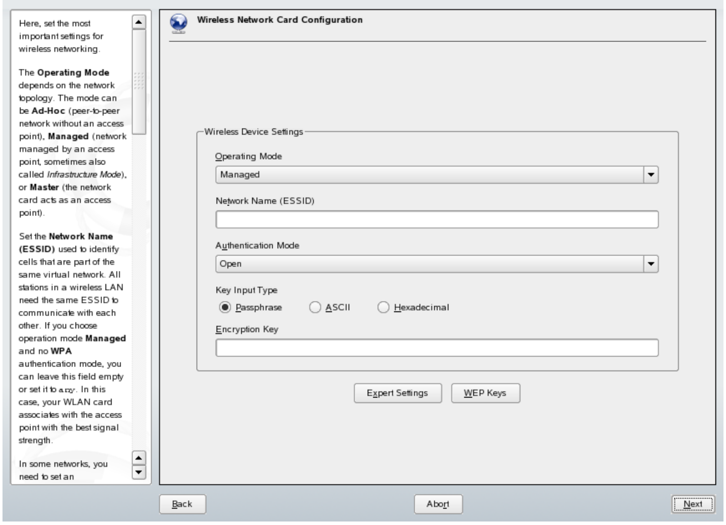
CCMP (defined in IEEE 802.11i)

CCMP describes the key management. Usually, it is used in connection with WPA-EAP, but it can also be used with WPA-PSK. The encryption takes place according to AES and is stronger than the RC4 encryption of the WEP standard.

22.1.3 Configuration with YaST

To configure your wireless network card, start the YaST *Network Card* module. In *Network Address Setup*, select the device type *Wireless* and click *Next*. In *Wireless Network Card Configuration*, shown in Figure 22.1, “YaST: Configuring the Wireless Network Card” (page 288), make the basic settings for the WLAN operation:

Figure 22.1 *YaST: Configuring the Wireless Network Card*



Here, set the most important settings for wireless networking.

The **Operating Mode** depends on the network topology. The mode can be **Ad-Hoc** (peer-to-peer network without an access point), **Managed** (network managed by an access point, sometimes also called *Infrastructure Mode*), or **Master** (the network card acts as an access point).

Set the **Network Name (ESSID)** used to identify cells that are part of the same virtual network. All stations in a wireless LAN need the same ESSID to communicate with each other. If you choose operation mode **Managed** and no **WPA** authentication mode, you can leave this field empty or set it to `any`. In this case, your WLAN card associates with the access point with the best signal strength.

In some networks, you need to set an

Wireless Network Card Configuration

Wireless Device Settings

Operating Mode: Managed

Network Name (ESSID):

Authentication Mode: Open

Key Input Type: ● Passphrase ○ ASCII ○ Hexadecimal

Encryption Key:

Operating Mode

A station can be integrated in a WLAN in three different modes. The suitable mode depends on the network in which to communicate: *Ad-hoc* (peer-to-peer network without access point), *Managed* (network is managed by an access point), or *Master* (your network card should be used as the access point). To use any of the WPA-PSK or WPA-EAP modes, the operating mode must be set to *managed*.

Network Name (ESSID)
All stations in a wireless network need the same ESSID for communicating with each other. If nothing is specified, the card automatically selects an access point, which may not be the one you intended to use.

Authentication Mode
Select a suitable authentication method for your network: *Open*, *Shared Key*, *WPA-PSK*, or *WPA-EAP*. If you select WPA authentication, a network name must be set.

Expert Settings
This button opens a dialog for the detailed configuration of your WLAN connection. A detailed description of this dialog is provided later.

After completing the basic settings, your station is ready for deployment in the WLAN.

IMPORTANT: Security in Wireless Networks

Be sure to use one of the supported authentication and encryption methods to protect your network traffic. Unencrypted WLAN connections allow third parties to intercept all network data. Even a weak encryption (WEP) is better than none at all. Refer to Section "Encryption" (page 287) and Section "Security" (page 292) for information.

Depending on the selected authentication method, YaST prompts you to fine-tune the settings in another dialog. For *Open*, there is nothing to configure, because this setting implements unencrypted operation without authentication.

WEP Keys
Set a key input type. Choose from *Passphrase*, *ASCII*, or *Hexadecimal*. You may keep up to four different keys to encrypt the transmitted data. Click *Multiple Keys* to enter the key configuration dialog. Set the length of the key: *128 bit* or *64 bit*. The default setting is *128 bit*. In the list area at the bottom of the dialog, up to four different keys can be specified for your station to use for the encryption. Press *Set as Default* to define one of them as the default key. Unless you change this, YaST uses the first entered key as the default key. If the standard key is deleted, one of the other keys must be marked manually as the default key. Click *Edit* to modify

existing list entries or create new keys. In this case, a pop-up window prompts you to select an input type (*Passphrase*, *ASCII*, or *Hexadecimal*). If you select *Passphrase*, enter a word or a character string from which a key is generated according to the length previously specified. *ASCII* requests an input of 5 characters for a 64-bit key and 13 characters for a 128-bit key. For *Hexadecimal*, enter 10 characters for a 64-bit key or 26 characters for a 128-bit key in hexadecimal notation.

WPA-PSK

To enter a key for WPA-PSK, select the input method *Passphrase* or *Hexadecimal*. In the *Passphrase* mode, the input must be 8 to 63 characters. In the *Hexadecimal* mode, enter 64 characters.

WPA-EAP

Enter the credentials you have been given by your network administrator. For TLS, provide the *Client Certificate* and *Server Certificate*. TTLS and PEAP require *Identity* and *Password*. *Server Certificate* is optional. YaST searches for any certificate under `/etc/cert`, so save the certificates given to you to this location and restrict access to these files to `0600` (owner read and write).

Click *Expert Settings* to leave the dialog for the basic configuration of the WLAN connection and enter the expert configuration. The following options are available in this dialog:

Channel

The specification of a channel on which the WLAN station should work is only needed in *Ad-hoc* and *Master* modes. In *Managed* mode, the card automatically searches the available channels for access points. In *Ad-hoc* mode, select one of the 12 offered channels for the communication of your station with the other stations. In *Master* mode, determine on which channel your card should offer access point functionality. The default setting for this option is *Auto*.

Bit Rate

Depending on the performance of your network, you may want to set a certain bit rate for the transmission from one point to another. In the default setting *Auto*, the system tries to use the highest possible data transmission rate. Some WLAN cards do not support the setting of bit rates.

Access Point

In an environment with several access points, one of them can be preselected by specifying the MAC address.

Use Power Management

When you are on the road, use power saving technologies to maximize the operating time of your battery. More information about power management is available in Chapter 21, *Power Management* (page 259).

22.1.4 Utilities

hostap (package `hostap`) is used to run a WLAN card as an access point. More information about this package is available at the project home page (`http://hostap.epitest.fi/`).

kismet (package `kismet`) is a network diagnosis tool with which to listen to the WLAN packet traffic. In this way, you can also detect any intrusion attempts in your network. More information is available at `http://www.kismetwireless.net/` and in the manual page.

22.1.5 Tips and Tricks for Setting Up a WLAN

These tips can help tweak speed and stability as well as security aspects of your WLAN.

Stability and Speed

The performance and reliability of a wireless network mainly depend on whether the participating stations receive a clean signal from the other stations. Obstructions like walls greatly weaken the signal. The more the signal strength sinks, the more the transmission slows down. During operation, check the signal strength with the iwconfig utility on the command line (`Link Quality` field) or with KInternet in KDE. If you have problems with the signal quality, try to set up the devices somewhere else or adjust the position of the antennas of your access points. Auxiliary antennas that substantially improve the reception are available for a number of PCMCIA WLAN cards. The rate specified by the manufacturer, such as 54 MBit/s, is a nominal value that represents the theoretical maximum. In practice, the maximum data throughput is no more than half this value.

Security

If you want to set up a wireless network, remember that anybody within the transmission range can easily access it if no security measures are implemented. Therefore, be sure to activate an encryption method. All WLAN cards and access points support WEP encryption. Although this is not entirely safe, it does present an obstacle for a potential attacker. WEP is usually adequate for private use. WPA-PSK would be even better, but it is not implemented in older access points or routers with WLAN functionality. On some devices, WPA can be implemented by means of a firmware update. Furthermore, Linux does not support WPA on all hardware components. When this documentation was prepared, WPA only worked with cards using Atheros, Intel PRO/Wireless, or Prism2/2.5/3 chips. On Prism2/2.5/3, WPA only works if the hostap driver is used (see Section “Problems with Prism2 Cards” (page 292)). If WPA is not available, WEP is better than no encryption. In enterprises with advanced security requirements, wireless networks should only be operated with WPA.

22.1.6 Troubleshooting

If your WLAN card fails to respond, check if you have downloaded the needed firmware. Refer to Section 22.1.1, “Hardware” (page 284). The following paragraphs cover some known problems.

Multiple Network Devices

Modern laptops usually have a network card and a WLAN card. If you configured both devices with DHCP (automatic address assignment), you may encounter problems with the name resolution and the default gateway. This is evident from the fact that you can ping the router but cannot surf the Internet. The Support Database at http://portal.suse.com features an article on this subject. To find the article, enter “DHCP” in the search dialog.

Problems with Prism2 Cards

Several drivers are available for devices with Prism2 chips. The various cards work more or less smoothly with the various drivers. With these cards, WPA is only possible with the hostap driver. If such a card does not work properly or not at all or you want to use WPA, read /usr/share/doc/packages/wireless-tools/README.prism2.

WPA

WPA support is quite new in SUSE Linux and still under development. Thus, YaST does not support the configuration of all WPA authentication methods. Not all wireless LAN cards and drivers support WPA. Some cards need a firmware update to enable WPA. If you want to use WPA, read /usr/share/doc/packages/wireless-tools/README.wpa.

22.1.7 For More Information

The Internet pages of Jean Tourrilhes, who developed the *Wireless Tools* for Linux, present a wealth of useful information about wireless networks. See http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Wireless.html.

22.2 Bluetooth

Bluetooth is a wireless technology for connecting various devices, such as cellular phones, PDAs, peripheral devices, laptops, or system components like the keyboard or mouse. The name is derived from the Danish king Harold Bluetooth, who united various warring factions in Scandinavia. The Bluetooth logo is based on the runes for “H” (resembles a star) and “B”.

A number of important aspects distinguish Bluetooth from IrDA. First, the individual devices do not need to “see” each other directly and, second, several devices can be connected in a network. However, the maximum data rate is 720 Kbps (in the current version 1.2). Theoretically, Bluetooth can even communicate through walls. In practice, however, this depends on the properties of the wall and the device class. There are three device classes with transmission ranges between ten and a hundred meters.

22.2.1 Basics

The following sections outline the basic principles of how Bluetooth works. Learn which software requirements need to be met, how Bluetooth interacts with your system, and how Bluetooth profiles work.

Software

To be able to use Bluetooth, you need a Bluetooth adapter (either a built-in adapter or an external device), drivers, and a Bluetooth protocol stack. The Linux kernel already contains the basic drivers for using Bluetooth. The Bluez system is used as protocol stack. To make sure that the applications work with Bluetooth, the base packages `bluez-libs` and `bluez-utils` must be installed. These packages provide a number of needed services and utilities. Additionally, some adapters, such as Broadcom or AVM BlueFritz!, require the `bluez-firmware` package to be installed. The `bluez-cups` package enables printing over Bluetooth connections.

General Interaction

A Bluetooth system consists of four interlocked layers that provide the desired functionality:

Hardware
The adapter and a suitable driver for support by the Linux kernel.

Configuration Files
Used for controlling the Bluetooth system.

Daemons
Services that are controlled by the configuration files and provide the functionality.

Applications
The applications allow the functionality provided by the daemons to be used and controlled by the user.

When inserting a Bluetooth adapter, its driver is loaded by the hotplug system. After the driver is loaded, the system checks the configuration files to see if Bluetooth should be started. If this is the case, it determines the services to start. Based on this information, the respective daemons are started. Bluetooth adapters are probed upon installation. If one or more are found, Bluetooth is enabled. Otherwise the Bluetooth system is deactivated. Any Bluetooth device added later must be enabled manually.

Profiles

In Bluetooth, services are defined by means of profiles, such as the file transfer profile, the basic printing profile, and the personal area network profile. To enable a device to use the services of another device, both must understand the same profile—a piece of information that is often missing in the device package and manual. Unfortunately, some manufacturers do not comply strictly with the definitions of the individual profiles. Despite this, communication between the devices usually works smoothly.

In the following text, local devices are those physically connected to the computer. All other devices that can only be accessed over wireless connections are referred to as remote devices.

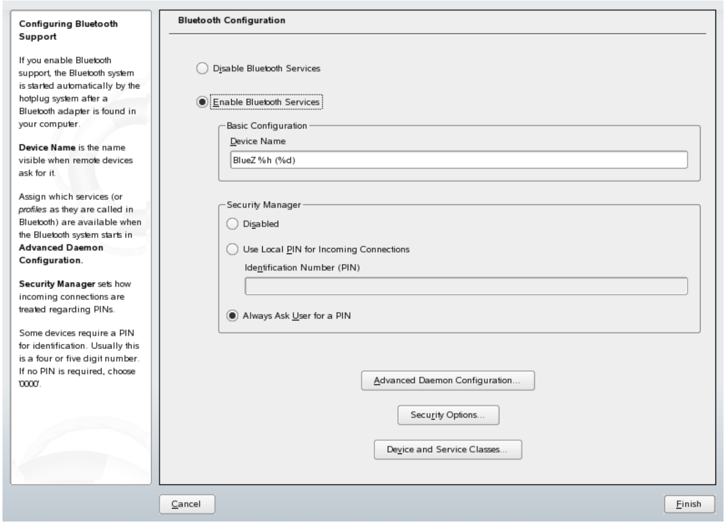
22.2.2 Configuration

This section introduces Bluetooth configuration. Learn which configuration files are involved, which tools are needed, and how to configure Bluetooth with YaST or manually.

Configuring Bluetooth with YaST

Use the YaST Bluetooth module, shown in Figure 22.2, “YaST Bluetooth Configuration” (page 296), to configure Bluetooth support on your system. As soon as hotplug detects a Bluetooth adapter on your system (for example, during booting or when you plug in an adapter), Bluetooth is automatically started with the settings configured in this module.

Figure 22.2 YaST Bluetooth Configuration



Configuring Bluetooth Support

If you enable Bluetooth support, the Bluetooth system is started automatically by the hotplug system after a Bluetooth adapter is found in your computer.

Device Name is the name visible when remote devices ask for it.

Assign which services (or *profiles* as they are called in Bluetooth) are available when the Bluetooth system starts in **Advanced Daemon Configuration.**

Security Manager sets how incoming connections are treated regarding PINs.

Some devices require a PIN for identification. Usually this is a four or five digit number. If no PIN is required, choose '0000'.

Bluetooth Configuration

○ Disable Bluetooth Services

● Enable Bluetooth Services

Basic Configuration

Device Name: BlueZ %h (%d)

Security Manager

○ Disabled

○ Use Local PIN for Incoming Connections

Identification Number (PIN):

● Always Ask User for a PIN

Advanced Daemon Configuration...
Security Options...
Device and Service Classes...
Cancel
Finish

In the first step of the configuration, determine whether Bluetooth services should be started on your system. If you have enabled the Bluetooth services, two things can be configured. First, the *Device Name*. This is the name other devices display when your computer has been discovered. There are two placeholders available—`%h` stands for the hostname of the system (useful, for example, if it is assigned dynamically by DHCP) and `%d` inserts the interface number (only useful if you have more than one Bluetooth adapter in your computer). For example, if you enter `Laptop %h` in the field and DHCP assigns the name `unit123` to your computer, other remote devices would know your computer as `Laptop unit123`.

The *Security Manager* parameter is related to the behavior of the local system when a remote device tries to connect. The difference is in the handling of the PIN number. Either allow any device to connect without a PIN or determine how the correct PIN is chosen if one is needed. You can enter a PIN (stored in a configuration file) in the appropriate input field. If a device tries to connect, it first uses this PIN. If it fails, it falls back to using no PIN. For maximum security, it is best to choose *Always Ask User for PIN*. This option allows you to use different PINs for different (remote) devices.

Click *Advanced Daemon Configuration* to enter the dialog for selecting and configuring the available services (called *profiles* in Bluetooth). All available services are displayed in a list and can be enabled or disabled by clicking *Activate* or *Deactivate*. Click *Edit* to open a dialog in which to specify additional arguments for the selected service (daemon). Do not change anything unless you are familiar with the service. After completing the configuration of the daemons, exit this dialog by clicking *OK*.

Back in the main dialog, click *Security Options* to enter the security dialog and specify encryption, authentication, and scan settings. Then exit the security dialog to return to the main dialog. After you close the main dialog with *Finish*, your Bluetooth system is ready for use.

From the main dialog, you can reach the *Device and Service Classes* dialog, too. Bluetooth devices are grouped into various device classes. In this dialog, choose the correct one for your computer, such as *Desktop* or *Laptop*. The device class is not very important, unlike the service class, also set here. Sometimes remote Bluetooth devices, like cell phones, only allow certain functions if they can detect the correct service class set on your system. This is often the case for cell phones that expect a class called *Object Transfer* before they allow the transfer of files from or to the computer. You can choose multiple classes. It is not useful to select all classes “just in case.” The default selection should be appropriate in most cases.

To use Bluetooth to set up a network, activate *PAND* in the *Advanced Daemon Configuration* dialog and set the mode of the daemon with *Edit*. For a functional Bluetooth network connection, one pand must operate in the *Listen* mode and the peer in the *Search* mode. By default, the *Listen* mode is preset. Adapt the behavior of your local pand. Additionally, configure the `bnepX` interface (`X` stands for the device number in the system) in the YaST *Network Card* module.

Configuring Bluetooth Manually

The configuration files for the individual components of the Bluez system are located in the directory `/etc/bluetooth`. The only exception is the file `/etc/sysconfig/bluetooth` for starting the components, which is modified by the YaST module.

The configuration files described below can only be modified by the user `root`. Currently, there is no graphical user interface to change all settings. The most important ones can be set using the YaST Bluetooth module, described in Section “Configuring Bluetooth with YaST” (page 295). All other settings are only of interest for experienced users with special cases. Usually, the default settings should be adequate.

A PIN number provides basic protection against unwanted connections. Mobile phones usually query the PIN when establishing the first contact (or when setting up a device contact on the phone). For two devices to be able to communicate, both must identify themselves with the same PIN. On the computer, the PIN is located in the file `/etc/bluetooth/pin`.

IMPORTANT: Security of Bluetooth Connections

Despite the PINs, the transmission between two devices may not be fully secure. By default, the authentication and encryption of Bluetooth connections is deactivated. Activating authentication and encryption may result in communication problems with some Bluetooth devices.

Various settings, such as the device names and the security mode, can be changed in the configuration file `/etc/bluetooth/hcid.conf`. Usually, the default settings should be adequate. The file contains comments describing the options for the various settings.

Two sections in the included file are designated as `options` and `device`. The first contains general information that hcid uses for starting. The latter contains settings for the individual local Bluetooth devices.

One of the most important settings of the `options` section is `security auto;`. If set to `auto`, hcid tries to use the local PIN for incoming connections. If it fails, it switches to `none` and establishes the connection anyway. For increased security, this default setting should be set to `user` to make sure that the user is requested to enter a PIN every time a connection is established.

Set the name under which the computer is displayed on the other side in the `device` section. The device class, such as `Desktop`, `Laptop`, or `Server`, is defined in this section. Authentication and encryption are also enabled or disabled here.

22.2.3 System Components and Utilities

The operability of Bluetooth depends on the interaction of various services. At least two background daemons are needed: hcid (host controller interface daemon), which serves as an interface for the Bluetooth device and controls it, and sdpd (service discovery protocol daemon), by means of which a device can find out which services the host makes available. If they are not activated automatically when the system is started, both hcid and sdpd can be activated with the command `rcbluetooth start`. This command must be executed as `root`.

The following paragraphs briefly describe the most important shell tools that can be used for working with Bluetooth. Although various graphical components are now available for controlling Bluetooth, it can be worthwhile to check these programs.

Some of the commands can only be executed as `root`. This includes the command `l2ping` *`device_address`* for testing the connection to a remote device.

hcitool

hcitool can be used to determine whether local and remote devices are detected. The command `hcitool dev` lists the local devices. The output generates a line in the form *`interface_name`* *`device_address`* for every detected local device.

Search for remote devices with the command `hcitool inq`. Three values are returned for every detected device: the device address, the clock offset, and the device class. The device address is important, because other commands use it for identifying the target device. The clock offset mainly serves a technical purpose. The class specifies the device type and the service type as a hexadecimal value.

The command `hcitool name` *`device-address`* can be used to determine the device name of a remote device. In the case of a remote computer, the class and the device name correspond to the information in its `/etc/bluetooth/hcid.conf`. Local device addresses generate an error output.

hciconfig

The command `/usr/sbin/hciconfig` delivers further information about the local device. If `hciconfig` is executed without any arguments, the output shows device information, such as the device name (`hciX`), the physical device address (a 12-digit number in the form `00:12:34:56:78`), and information about the amount of transmitted data.

`hciconfig hci0 name` displays the name that is returned by your computer when it receives requests from remote devices. As well as querying the settings of the local device, `hciconfig` can be used for modifying these settings. For example, `hciconfig hci0 name TEST` sets the name to `TEST`.

sdptool

The program sdptool can be used to check which services are made available by a specific device. The command `sdptool browse device_address` returns all services of a device. Use the command `sdptool search service_code` to search for a specific service. This command scans all accessible devices for the requested service. If one of the devices offers the service, the program prints the full service name returned by the device together with a brief description. View a list of all possible service codes by entering `sdptool` without any parameters.

22.2.4 Graphical Applications

In Konqueror, enter the URL `bluetooth:/` to list local and remote Bluetooth devices. Double-click a device for an overview of the services provided by the device. If you move across one of the specified services with the mouse, the browser's status bar shows which profile is used for the service. If you click a service, a dialog opens, asking what to do: save, use the service (an application must be started to do this), or cancel the action. Mark a check box if you do not want the dialog to be displayed again but always want the selected action to be performed. For some services, support is not yet available. For others, additional packages may need to be installed.

22.2.5 Examples

This section features two typical examples of possible Bluetooth scenarios. The first shows how a network connection between two hosts can be established via Bluetooth. The second features a connection between a computer and a mobile phone.

Network Connection between Two Hosts

In the first example, a network connection is established between the hosts *H1* and *H2*. These two hosts have the Bluetooth device addresses *baddr1* and *baddr2* (determined on both hosts with the command `hcitool dev` as described above). The hosts should be identified with the IP addresses `192.168.1.3` (*H1*) and `192.168.1.4` (*H2*).

The Bluetooth connection is established with the help of pand (personal area networking daemon). The following commands must be executed by the user `root`. The description focuses on the Bluetooth-specific actions and does not provide a detailed explanation of the network command `ip`.

Enter `pand -s` to start pand on the host *H1*. Subsequently, a connection can be established on the host *H2* with `pand -c` *`baddr1`*. If you enter `ip link show` on one of the hosts to list the available network interfaces, the output should contain an entry like the following:

```
bnep0: <BROADCAST,MULTICAST> mtu 1500 qdisc noop qlen 1000
 link/ether 00:12:34:56:89:90 brd ff:ff:ff:ff:ff:ff
```

Instead of `00:12:34:56:89:90`, the output should contain the local device address *baddr1* or *baddr2*. Now this interface must be assigned an IP address and activated. On *H1*, this can be done with the following two commands:

```
ip addr add 192.168.1.3/24 dev bnep0
ip link set bnep0 up
```

On *H2*:

```
ip addr add 192.168.1.4/24 dev bnep0
ip link set bnep0 up
```

Now *H1* can be accessed from *H2* under the IP `192.168.1.3`. Use the command `ssh 192.168.1.4` to access *H2* from *H1*, assuming *H2* runs an sshd, which is activated by default in SUSE Linux. The command `ssh 192.168.1.4` can also be run as a normal user.

Data Transfer from a Mobile Phone to the Computer

The second example shows how to transfer a photograph created with a mobile phone with a built-in digital camera to a computer (without incurring additional costs for the transmission of a multimedia message). Although the menu structure may differ on various mobile phones, the procedure is usually quite similar. Refer to the manual of your phone, if necessary. This example describes the transfer of a photograph from a Sony Ericsson mobile phone to a laptop. The service Obex-Push must be available on the computer and the computer must grant the mobile phone access. In the first step, the service is made available on the laptop. This is done by means of the opd daemon from the package `bluez-utils`. Start the daemon with the following command:

```
opd --mode OBEX --channel 10 --daemonize --path /tmp --sdp
```

Two important parameters are used: `--sdp` registers the service with sdpd and `--path /tmp` instructs the program where to save the received data—in this case to `/tmp`. You can also specify any other directory to which you have write access.

Now the mobile phone must get to know the computer. To do this, open the *Connect* menu on the phone and select *Bluetooth*. If necessary, click *Turn On* before selecting *My devices*. Select *New device* and let your phone search for the laptop. If a device is detected, its name appears in the display. Select the device associated with the laptop. If you encounter a PIN query, enter the PIN specified in `/etc/bluetooth/pin`. Now your phone recognizes the laptop and is able to exchange data with the laptop. Exit the current menu and go to the image menu. Select the image to transfer and press *More*. In the next menu, press *Send* to select a transmission mode. Select *Via Bluetooth*. The laptop should be listed as a target device. Select the laptop to start the transmission. The image is then saved to the directory specified with the `opd` command. Audio tracks can be transferred to the laptop in the same way.

22.2.6 Troubleshooting

If you have difficulties establishing a connection, proceed according to the following list. Remember that the error can be on either side of a connection or even on both sides. If possible, reconstruct the problem with another Bluetooth device to verify that the device is not defective.

Is the local device listed in the output of `hcitool dev`?

If the local device is not listed in this output, hcid is not started or the device is not recognized as a Bluetooth device. This can have various causes. The device may be defective or the correct driver may be missing. Laptops with built-in Bluetooth often have an on and off switch for wireless devices, like WLAN and Bluetooth. Check the manual of your laptop to see if your device has such a switch. Restart the Bluetooth system with the command `rcbluetooth restart` and check if any errors are reported in `/var/log/messages`.

Does your Bluetooth adapter need a firmware file?

If it does, install `bluez-bluefw` and restart the Bluetooth system with `rcbluetooth restart`.

Does the output of `hcitool inq` return other devices?

Test this command more than once. The connection may have interferences, because the frequency band of Bluetooth is also used by other devices.

Do the PINs match?

Check if the PIN number of the computer (in `/etc/bluetooth/pin`) matches that of the target device.

Can the remote device “see” your computer?

Try to establish the connection from the remote device. Check if this device sees the computer.

Can a network connection be established (see Section “Network Connection between Two Hosts” (page 300))?

The setup described in Section “Network Connection between Two Hosts” (page 300) may not work for several reasons. For example, one of the two computers may not support the ssh protocol. Try `ping 192.168.1.3` or `ping 192.168.1.4`. If this works, check if sshd is active. Another problem could be that one of the two devices already has network settings that conflict with the address `192.168.1.X` in the example. If this is the case, try different addresses, such as `10.123.1.2` and `10.123.1.3`.

Does the laptop appear as a target device (see Section “Data Transfer from a Mobile Phone to the Computer” (page 301))? Does the mobile device recognize the Obex-Push service on the laptop?

In *My devices*, select the respective device and view the list of *Services*. If Obex-Push is not displayed (even after the list is updated), the problem is caused by opd on the laptop. Is opd active? Do you have write access to the specified directory?

Does the scenario described in Section “Data Transfer from a Mobile Phone to the Computer” (page 301) work the other way around?

If the `obexftp` package is installed, the command `obexftp -b` *`device_address`* `-B 10 -p` *`image`* can be used on some devices. Several Siemens and Sony Ericsson models have been tested and found to be functional. Refer to the documentation in `/usr/share/doc/packages/obexftp`.

22.2.7 For More Information

An extensive overview of various instructions for the use and configuration of Bluetooth is available at `http://www.holtmann.org/linux/bluetooth/`. Other useful information and instructions:

- Official howto of the Bluetooth protocol stack integrated in the kernel: `http://bluez.sourceforge.net/howto/index.html`

- Connection to PalmOS PDA: `http://www.cs.ucl.ac.uk/staff/s.zachariadis/btpalmlinux.html`

22.3 Infrared Data Transmission

IrDA (Infrared Data Association) is an industry standard for wireless communication with infrared light. Many laptops sold today are equipped with an IrDA-compatible transceiver that enables communication with other devices, such as printers, modems, LANs, or other laptops. The transfer speed ranges from 2400 bps to 4 Mbps.

There are two IrDA operation modes. The standard mode, SIR, accesses the infrared port through a serial interface. This mode works on almost all systems and is sufficient for most requirements. The faster mode, FIR, requires a special driver for the IrDA chip. Not all chip types are supported in FIR mode because of a lack of appropriate drivers. Set the desired IrDA mode in the BIOS of your computer. The BIOS also shows which serial interface is used in SIR mode.

Information about IrDA can be found in the IrDA how-to by Werner Heuser at `http://tuxmobil.org/Infrared-HOWTO/Infrared-HOWTO.html`. Additionally refer to the Web site of the Linux IrDA Project at `http://irda.sourceforge.net/`.

22.3.1 Software

The necessary kernel modules are included in the kernel package. The package `irda` provides the necessary helper applications for supporting the infrared interface. The documentation can be found at `/usr/share/doc/packages/irda/README` after the installation of the package.

22.3.2 Configuration

The IrDA system service is not started automatically when the system is booted. Use the YaST IrDA module for the activation. Only one setting can be modified in this module: the serial interface of the infrared device. The test window shows two outputs. One is the output of irdadump, which logs all sent and received IrDA packets. This output should contain the name of the computer and the names of all infrared devices in transmission range. An example for these messages is shown in Section 22.3.4, “Troubleshooting” (page 306). All devices to which an IrDA connection exists are listed in the lower part of the window.

IrDA consumes a considerable amount of battery power, because a discovery packet is sent every few seconds to detect other peripheral devices. Therefore, IrDA should only be started when necessary if you depend on battery power. Enter the command rcirda start to activate it or rcirda stop to deactivate it. All needed kernel modules are loaded automatically when the interface is activated.

Manual configuration can be performed in the file /etc/sysconfig/irda. This file contains only one variable, IRDA_PORT, which determines the interface to use in SIR mode.

22.3.3 Usage

Data can be sent to the device file /dev/irlpt0 for printing. The device file /dev/irlpt0 acts just like the normal /dev/lp0 cabled interface, except the printing data is sent wirelessly with infrared light. For printing, make sure that the printer is in visual range of the computer's infrared interface and the infrared support is started.

A printer that is operated over the infrared interface can be configured with the YaST Printer module. Because it is not detected automatically, configure it manually by clicking *Other (not detected)*. In the following dialog, select *IrDA printer*. Usually, irlpt0 is the right connection. Details about operating printers in Linux are available in Chapter 31, *Printer Operation* (page 461).

Communication with other hosts and with mobile phones or other similar devices is conducted through the device file /dev/ircomm0. The Siemens S25 and Nokia 6210 mobile phones, for example, can dial and connect to the Internet with the wvdial application using the infrared interface. Synchronizing data with a Palm Pilot is also possible, provided the device setting of the corresponding application has been set to /dev/ircomm0.

If you want, you can address only devices that support the printer or IrCOMM protocols. Devices that support the IROBEX protocol, such as the 3Com Palm Pilot, can be accessed with special applications, like irobexpalm and irobexreceive. Refer to the *IR-HOWTO* (http://tldp.org/HOWTO/Infrared-HOWTO/) for information. The protocols supported by the device are listed in brackets after the name of the device in the output of irdadump. IrLAN protocol support is still a “work in progress.”

22.3.4 Troubleshooting

If devices connected to the infrared port do not respond, use the command irdadump (as root) to check if the other device is recognized by the computer. Something similar to Example 22.1, “Output of irdadump” (page 306) appears regularly when a Canon BJC-80 printer is in visible range of the computer:

Example 22.1 *Output of irdadump*

```
21:41:38.435239 xid:cmd 5b62bed5 > ffffffff S=6 s=0 (14)
21:41:38.525167 xid:cmd 5b62bed5 > ffffffff S=6 s=1 (14)
21:41:38.615159 xid:cmd 5b62bed5 > ffffffff S=6 s=2 (14)
21:41:38.705178 xid:cmd 5b62bed5 > ffffffff S=6 s=3 (14)
21:41:38.795198 xid:cmd 5b62bed5 > ffffffff S=6 s=4 (14)
21:41:38.885163 xid:cmd 5b62bed5 > ffffffff S=6 s=5 (14)
21:41:38.965133 xid:rsp 5b62bed5 < 6cac38dc S=6 s=5 BJC-80
                        hint=8804 [Printer IrCOMM ] (23)
21:41:38.975176 xid:cmd 5b62bed5 > ffffffff S=6 s=* earth
                        hint=0500 [ PnP Computer ] (21)
```

Check the configuration of the interface if there is no output or the other device does not reply. Verify that the correct interface is used. The infrared interface is sometimes located at /dev/ttyS2 or at /dev/ttyS3 and an interrupt other than IRQ 3 is sometimes used. These settings can be checked and modified in the BIOS setup menu of almost every laptop.

A simple video camera can also help in determining whether the infrared LED lights up at all. Most video cameras can see infrared light; the human eye cannot.

Part VII Administration

Security in Linux 23

Masquerading and a firewall ensure a controlled data flow and data exchange. SSH (secure shell) enables you to log in to remote hosts over an encrypted connection. The encryption of files or entire partitions protects your data in the event that third parties gain access to your system. Along with technical instructions, find information about security aspects of Linux networks.

23.1 Masquerading and Firewalls

Whenever Linux is used in a networked environment, you can use the kernel functions that allow the manipulation of network packets to maintain a separation between internal and external network areas. The Linux netfilter framework provides the means to establish an effective firewall that keeps different networks apart. With the help of iptables—a generic table structure for the definition of rule sets—precisely control the packets allowed to pass a network interface. Such a packet filter can be set up quite easily with the help of SuSEfirewall2 and the corresponding YaST module.

23.1.1 Packet Filtering with iptables

The components netfilter and iptables are responsible for the filtering and manipulation of network packets as well as for network address translation (NAT). The filtering criteria and any actions associated with them are stored in chains, which must be matched one after another by individual network packets as they arrive. The chains to match are stored in tables. The `iptables` command allows you to alter these tables and rule sets.

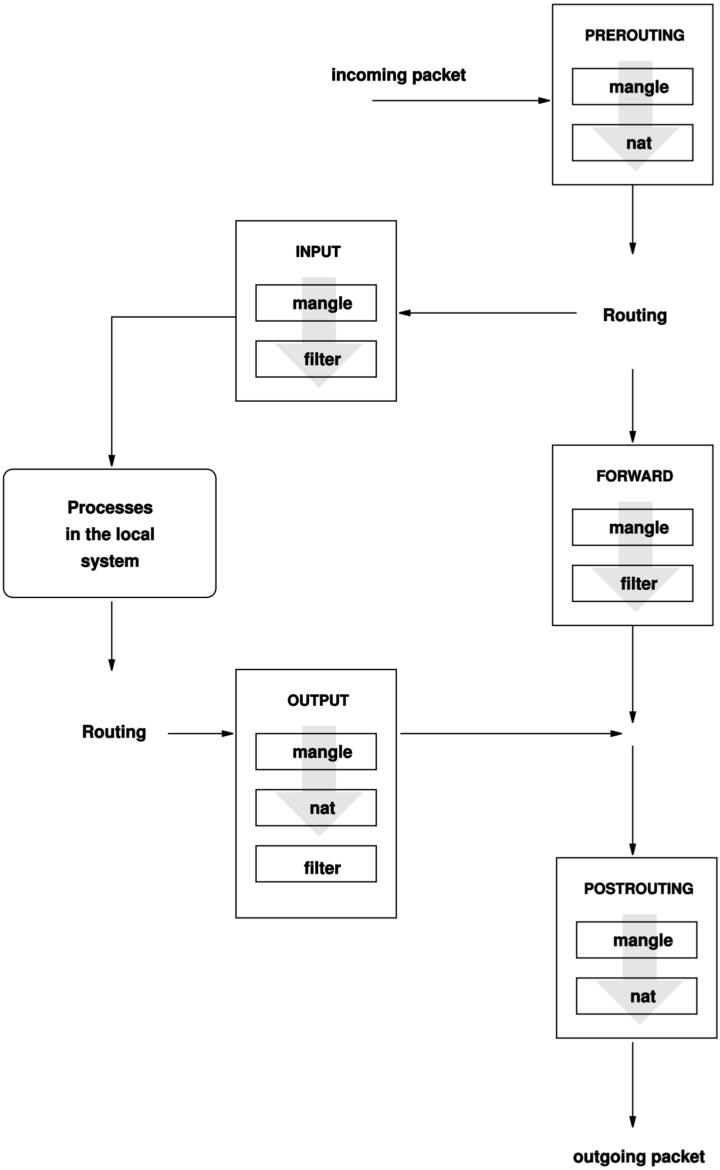
The Linux kernel maintains three tables, each for a particular category of functions of the packet filter:

filter
This table holds the bulk of the filter rules, because it implements the *packet filtering* mechanism in the stricter sense, which determines whether packets are let through (`ACCEPT`) or discarded (`DROP`), for example.

nat
This table defines any changes to the source and target addresses of packets. Using these functions also allows you to implement *masquerading*, which is a special case of NAT used to link a private network with the Internet.

mangle
The rules held in this table make it possible to manipulate values stored in IP headers (such as the type of service).

Figure 23.1 *iptables: A Packet's Possible Paths*



These tables contain several predefined chains to match packets:

PREROUTING
This chain is applied to incoming packets.

INPUT
This chain is applied to packets destined for the system's internal processes.

FORWARD
This chain is applied to packets that are only routed through the system.

OUTPUT
This chain is applied to packets originating from the system itself.

POSTROUTING
This chain is applied to all outgoing packets.

Figure 23.1, “iptables: A Packet's Possible Paths” (page 311) illustrates the paths along which a network packet may travel on a given system. For the sake of simplicity, the figure lists tables as parts of chains, but in reality these chains are held within the tables themselves.

In the simplest of all possible cases, an incoming packet destined for the system itself arrives at the `eth0` interface. The packet is first referred to the `PREROUTING` chain of the `mangle` table then to the `PREROUTING` chain of the `nat` table. The following step, concerning the routing of the packet, determines that the actual target of the packet is a process of the system itself. After passing the `INPUT` chains of the `mangle` and the `filter` table, the packet finally reaches its target, provided that the rules of the `filter` table are actually matched.

23.1.2 Masquerading Basics

Masquerading is the Linux-specific form of NAT (network address translation). It can be used to connect a small LAN (where hosts use IP addresses from the private range—see Section 38.1.2, “Netmasks and Routing” (page 551)) with the Internet (where official IP addresses are used). For the LAN hosts to be able to connect to the Internet, their private addresses are translated to an official one. This is done on the router, which acts as the gateway between the LAN and the Internet. The underlying principle is a simple one: The router has more than one network interface, typically a network card and a separate interface connecting with the Internet. While the latter links the router with the outside world, one or several others link it with the LAN hosts. With these hosts in the local network connected to the network card (such as `eth0`) of the router, they can send any packets not destined for the local network to their default gateway or router.

IMPORTANT: Using the Correct Network Mask

When configuring your network, make sure both the broadcast address and the netmask are the same for all local hosts. Failing to do so prevents packets from being routed properly.

As mentioned, whenever one of the LAN hosts sends a packet destined for an Internet address, it goes to the default router. However, the router must be configured before it can forward such packets. For security reasons, SUSE Linux does not enable this in a default installation. To enable it, set the variable `IP_FORWARD` in the file `/etc/sysconfig/sysctl` to `IP_FORWARD=yes`.

The target host of the connection can see your router, but knows nothing about the host in your internal network where the packets originated. This is why the technique is called masquerading. Because of the address translation, the router is the first destination of any reply packets. The router must identify these incoming packets and translate their target addresses, so packets can be forwarded to the correct host in the local network.

With the routing of inbound traffic depending on the masquerading table, there is no way to open a connection to an internal host from the outside. For such a connection, there would be no entry in the table. In addition, any connection already established has a status entry assigned to it in the table, so the entry cannot be used by another connection.

As a consequence of all this, you might experience some problems with a number of application protocols, such as ICQ, cucme, IRC (DCC, CTCP), and FTP (in PORT mode). Netscape, the standard FTP program, and many others use the PASV mode. This passive mode is much less problematic as far as packet filtering and masquerading are concerned.

23.1.3 Firewalling Basics

Firewall is probably the term most widely used to describe a mechanism that provides and manages a link between networks while also controlling the data flow between them. Strictly speaking, the mechanism described in this section is called a *packet filter*. A packet filter regulates the data flow according to certain criteria, such as protocols, ports, and IP addresses. This allows you to block packets that, according to their addresses, are not supposed to reach your network. To allow public access to your Web server, for example, explicitly open the corresponding port. However, a packet filter does not scan the contents of packets with legitimate addresses, such as those directed to your Web server. For example, if incoming packets were intended to compromise a CGI program on your Web server, the packet filter would still let them through.

A more effective but more complex mechanism is the combination of several types of systems, such as a packet filter interacting with an application gateway or proxy. In this case, the packet filter rejects any packets destined for disabled ports. Only packets directed to the application gateway are accepted. This gateway or proxy pretends to be the actual client of the server. In a sense, such a proxy could be considered a masquerading host on the protocol level used by the application. One example for such a proxy is Squid, an HTTP proxy server. To use Squid, the browser must be configured to communicate via the proxy. Any HTTP pages requested are served from the proxy cache and pages not found in the cache are fetched from the Internet by the proxy. As another example, the SUSE proxy-suite (`proxy-suite`) provides a proxy for the FTP protocol.

The following section focuses on the packet filter that comes with SUSE Linux. For further information about packet filtering and firewalling, read the Firewall HOWTO included in the `howto` package. If this package is installed, read the HOWTO with `less /usr/share/doc/howto/en/txt/Firewall-HOWTO.gz`.

23.1.4 SuSEfirewall2

SuSEfirewall2 is a script that reads the variables set in `/etc/sysconfig/SuSEfirewall2` to generate a set of iptables rules. It defines three security zones, although only the first and the second one are considered in the following sample configuration:

External Zone
Given that there is no way to control what is happening on the external network, the host needs to be protected from it. In most cases, the external network is the Internet, but it could be another insecure network, such as a WLAN.

Internal Zone

This refers to the private network, in most cases the LAN. If the hosts on this network use IP addresses from the private range (see Section 38.1.2, "Netmasks and Routing" (page 551)), enable network address translation (NAT), so hosts on the internal network can access the external one.

Demilitarized Zone (DMZ)

While hosts located in this zone can be reached both from the external and the internal network, they cannot access the internal network themselves. This setup can be used to put an additional line of defense in front of the internal network, because the DMZ systems are isolated from the internal network.

Any kind of network traffic not explicitly allowed by the filtering rule set is suppressed by iptables. Therefore, each of the interfaces with incoming traffic must be placed into one of the three zones. For each of the zones, define the services or protocols allowed. The rule set is only applied to packets originating from remote hosts. Locally generated packets are not captured by the firewall.

The configuration can be performed with YaST (see Section "Configuring with YaST" (page 315)). It can also be made manually in the file `/etc/sysconfig/SuSEfirewall2`, which is well commented. Additionally, a number of example scenarios are available in `/usr/share/doc/packages/SuSEfirewall2/EXAMPLES`.

Configuring with YaST

IMPORTANT: Automatic Firewall Configuration

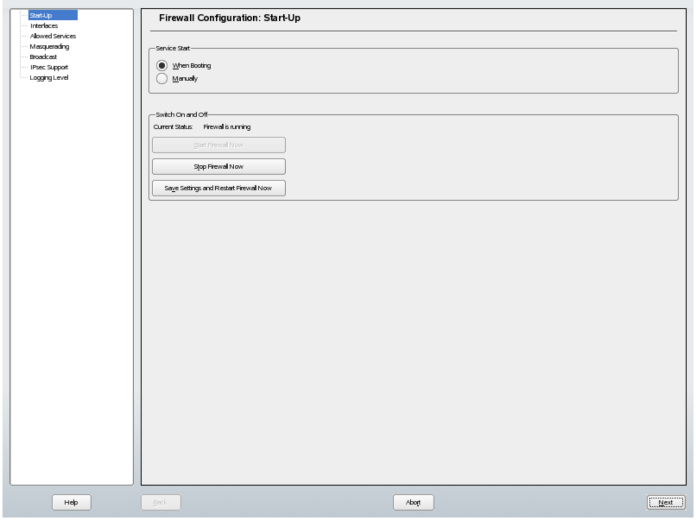
After the installation, YaST automatically starts a firewall on all configured interfaces. If a server is configured and activated on the system, YaST can modify the automatically-generated firewall configuration with the options *Open Ports on Selected Interface in Firewall* or *Open Ports on Firewall* in the server configuration modules. Some server module dialogs include a *Firewall Details* button for activating additional services and ports. The YaST firewall configuration module can be used to activate, deactivate, or reconfigure the firewall.

The YaST dialogs for the graphical configuration can be accessed from the YaST Control Center. Select *Security and Users* → *Firewall*. The configuration is divided into seven sections that can be accessed directly from the tree structure on the left side.

Start-Up

Set the start-up behavior in this dialog. In a default installation, SuSEfirewall2 is started automatically. You can also start and stop the firewall here. To implement your new settings in a running firewall, use *Save Settings and Restart Firewall Now*.

Figure 23.2 *The YaST Firewall Configuration*



Firewall Configuration: Start-Up

Service Start

● When Booting

○ Manually

Switch On and Off

Current Status: Firewall is running

Interfaces

All known network interfaces are listed here. To remove an interface from a zone, select the interface, press *Change*, and choose *No Zone Assigned*. To add an interface to a zone, select the interface, press *Change* and choose any of the available zones. You may also create a special interface with your own settings by using *Custom*.

Allowed Services

You need this option to offer services from your system to a zone from which it is protected. By default, the system is only protected from external zones. Explicitly allow the services that should be available to external hosts. Activate the services after selecting the desired zone in *Allowed Services for Selected Zone*.

Masquerading

Masquerading hides your internal network from external networks, such as the Internet, while enabling hosts in the internal network to access the external network transparently. Requests from the external network to the internal one are blocked and requests from the internal network seem to be issued by the masquerading server when seen externally. If special services of an internal machine need to be available to the external network, add special redirect rules for the service.

Broadcast

In this dialog, configure the UDP ports that allow broadcasts. Add the required port numbers or services to the appropriate zone, separated by spaces. See also the file `/etc/services`.

The logging of broadcasts that are not accepted can be enabled here. This may be problematic, because Windows hosts use broadcasts to know about each other and so generate many packets that are not accepted.

IPsec Support

Configure whether the IPsec service should be available to the external network in this dialog. Configure which packets are trusted under *Details*.

Logging Level

There are two rules for the logging: accepted and not accepted packets. Packets that are not accepted are DROPPED or REJECTED. Select from *Log All*, *Log Critical*, or *Do Not Log Any* for both of them.

When completed with the firewall configuration, exit this dialog with *Next*. A zone-oriented summary of your firewall configuration then opens. In it, check all settings. All services, ports, and protocols that have been allowed are listed in this summary. To modify the configuration, use *Back*. Press *Accept* to save your configuration.

Configuring Manually

The following paragraphs provide step-by-step instructions for a successful configuration. Each configuration item is marked as to whether it is relevant to firewalling or masquerading. Aspects related to the DMZ (demilitarized zone) as mentioned in the configuration file are not covered here. They are applicable only to a more complex network infrastructure found in larger organizations (corporate networks), which require extensive configuration and in-depth knowledge about the subject.

First, use the YaST module System Services (Runlevel) to enable SuSEfirewall2 in your runlevel (3 or 5 most likely). It sets the symlinks for the SuSEfirewall2_* scripts in the `/etc/init.d/rc?.d/` directories.

`FW_DEV_EXT` (firewall, masquerading)

The device linked to the Internet. For a modem connection, enter `ppp0`. For an ISDN link, use `ippp0`. DSL connections use `dsl0`. Specify `auto` to use the interface that corresponds to the default route.

`FW_DEV_INT` (firewall, masquerading)

The device linked to the internal, private network (such as `eth0`). Leave this blank if there is no internal network and the firewall protects only the host on which it runs.

`FW_ROUTE` (firewall, masquerading)

If you need the masquerading function, set this to `yes`. Your internal hosts will not be visible to the outside, because their private network addresses (e.g., `192.168.x.x`) are ignored by Internet routers.

For a firewall without masquerading, only set this to `yes` if you want to allow access to the internal network. Your internal hosts need to use officially registered IPs in this case. Normally, however, you should *not* allow access to your internal network from the outside.

`FW_MASQUERADE` (masquerading)

Set this to `yes` if you need the masquerading function. This provides a virtually direct connection to the Internet for the internal hosts. It is more secure to have a proxy server between the hosts of the internal network and the Internet. Masquerading is not needed for services a proxy server provides.

`FW_MASQ_NETS` (masquerading)

Specify the hosts or networks to masquerade, leaving a space between the individual entries. For example:

```
FW_MASQ_NETS="192.168.0.0/24 192.168.10.1"
```

`FW_PROTECT_FROM_INT` (firewall)

Set this to `yes` to protect your firewall host from attacks originating in your internal network. Services are only available to the internal network if explicitly enabled. Also see `FW_SERVICES_INT_TCP` and `FW_SERVICES_INT_UDP`.

`FW_SERVICES_EXT_TCP` (firewall)

Enter the TCP ports that should be made available. Leave this blank for a normal workstation at home that should not offer any services.

FW_SERVICES_EXT_UDP (firewall)
Leave this blank unless you run a UDP service and want to make it available to the outside. The services that use UDP include include DNS servers, IPSec, TFTP, DHCP and others. In that case, enter the UDP ports to use.

FW_SERVICES_INT_TCP (firewall)
With this variable, define the services available for the internal network. The notation is the same as for FW_SERVICES_EXT_TCP, but the settings are applied to the *internal* network. The variable only needs to be set if FW_PROTECT_FROM_INT is set to yes.

FW_SERVICES_INT_UDP (firewall)
See FW_SERVICES_INT_TCP.

After configuring the firewall, test your setup. The firewall rule sets are created by entering SuSEfirewall2 start as root. Then use telnet, for example, from an external host to see whether the connection is actually denied. After that, review /var/log/messages, where you should see something like this:

```
Mar 15 13:21:38 linux kernel: SFW2-INext-DROP-DEFLT IN=eth0
OUT= MAC=00:80:c8:94:c3:e7:00:a0:c9:4d:27:56:08:00 SRC=192.168.10.0
DST=192.168.10.1 LEN=60 TOS=0x10 PREC=0x00 TTL=64 ID=15330 DF PROTO=TCP
SPT=48091 DPT=23 WINDOW=5840 RES=0x00 SYN URGP=0
OPT (020405B40402080A061AFEBC0000000001030300)
```

Other packages to test your firewall setup are nmap or nessus. The documentation of nmap is found at /usr/share/doc/packages/nmap and the documentation of nessus resides in the directory /usr/share/doc/packages/nessus-core after installing the respective package.

23.1.5 For More Information

The most up-to-date information and other documentation about the SuSEfirewall2 package is found in /usr/share/doc/packages/SuSEfirewall2. The home page of the netfilter and iptables project, http://www.netfilter.org, provides a large collection of documents in many languages.

23.2 SSH: Secure Network Operations

With more and more computers installed in networked environments, it often becomes necessary to access hosts from a remote location. This normally means that a user sends login and password strings for authentication purposes. As long as these strings are transmitted as plain text, they could be intercepted and misused to gain access to that user account without the authorized user even knowing about it. Apart from the fact that this would open all the user's files to an attacker, the illegal account could be used to obtain administrator or `root` access or to penetrate other systems. In the past, remote connections were established with telnet, which offers no guards against eavesdropping in the form of encryption or other security mechanisms. There are other unprotected communication channels, like the traditional FTP protocol and some remote copying programs.

The SSH suite provides the necessary protection by encrypting the authentication strings (usually a login name and a password) and all the other data exchanged between the hosts. With SSH, the data flow could still be recorded by a third party, but the contents are encrypted and cannot be reverted to plain text unless the encryption key is known. So SSH enables secure communication over insecure networks, such as the Internet. The SSH flavor that comes with SUSE Linux is OpenSSH.

23.2.1 The OpenSSH Package

SUSE Linux installs the package OpenSSH by default. The programs ssh, scp, and sftp are then available as alternatives to telnet, rlogin, rsh, rcp, and ftp. In the default configuration, system access of a SUSE Linux system is only possible with the OpenSSH utilities and only if the firewall permits access.

23.2.2 The ssh Program

Using the ssh program, it is possible to log in to remote systems and work interactively. It replaces both telnet and rlogin. The slogin program is just a symbolic link pointing to ssh. For example, log in to the host sun with the command `ssh sun`. The host then prompts for the password on sun.

After successful authentication, you can work on the remote command line or use interactive applications, such as YaST. If the local username is different from the remote username, you can log in using a different login name with `ssh -l augustine sun` or `ssh augustine@sun`.

Furthermore, ssh offers the possibility to run commands on remote systems, as known from rsh. In the following example, run the command `uptime` on the host sun and create a directory with the name `tmp`. The program output is displayed on the local terminal of the host earth.

```
ssh otherplanet "uptime; mkdir tmp"
tux@otherplanet's password:
1:21pm  up  2:17,  9 users,  load average: 0.15, 0.04, 0.02
```

Quotation marks are necessary here to send both instructions with one command. It is only by doing this that the second command is executed on sun.

23.2.3 scp—Secure Copy

scp copies files to a remote machine. It is a secure and encrypted substitute for rcp. For example, `scp MyLetter.tex sun:` copies the file `MyLetter.tex` from the host earth to the host sun. If the username on earth is different than the username on sun, specify the latter using the `username@host` format. There is no `-l` option for this command.

After the correct password is entered, scp starts the data transfer and shows a growing row of asterisks to simulate a progress bar. In addition, the program displays the estimated time of arrival to the right of the progress bar. Suppress all output by giving the option `-q`.

scp also provides a recursive copying feature for entire directories. The command `scp -r src/ sun:backup/` copies the entire contents of the directory `src` including all subdirectories to the `backup` directory on the host sun. If this subdirectory does not exist yet, it is created automatically.

The option `-p` tells scp to leave the time stamp of files unchanged. `-C` compresses the data transfer. This minimizes the data volume to transfer, but creates a heavier burden on the processor.

23.2.4 sftp—Secure File Transfer

The sftp program can be used instead of scp for secure file transfer. During an sftp session, you can use many of the commands known from ftp. The sftp program may be a better choice than scp, especially when transferring data for which the filenames are unknown.

23.2.5 The SSH Daemon (sshd)—Server-Side

To work with the SSH client programs ssh and scp, a server, the SSH daemon, must be running in the background, listening for connections on `TCP/IP port 22`. The daemon generates three key pairs when starting for the first time. Each key pair consist of a private and a public key. Therefore, this procedure is referred to as public key–based. To guarantee the security of the communication via SSH, access to the private key files must be restricted to the system administrator. The file permissions are set accordingly by the default installation. The private keys are only required locally by the SSH daemon and must not be given to anyone else. The public key components (recognizable by the name extension `.pub`) are sent to the client requesting the connection. They are readable for all users.

A connection is initiated by the SSH client. The waiting SSH daemon and the requesting SSH client exchange identification data to compare the protocol and software versions and to prevent connections through the wrong port. Because a child process of the original SSH daemon replies to the request, several SSH connections can be made simultaneously.

For the communication between SSH server and SSH client, OpenSSH supports versions 1 and 2 of the SSH protocol. A newly installed SUSE Linux system defaults to version 2. To continue using version 1 after an update, follow the instructions in `/usr/share/doc/packages/openssh/README.SuSE`. This document also describes how an SSH 1 environment can be transformed into a working SSH 2 environment with just a few steps.

When using version 1 of SSH, the server sends its public host key and a server key, which is regenerated by the SSH daemon every hour. Both allow the SSH client to encrypt a freely chosen session key, which is sent to the SSH server. The SSH client also tells the server which encryption method (cipher) to use.

Version 2 of the SSH protocol does not require a server key. Both sides use an algorithm according to Diffie-Helman to exchange their keys.

The private host and server keys are absolutely required to decrypt the session key and cannot be derived from the public parts. Only the SSH daemon contacted can decrypt the session key using its private keys (see `man /usr/share/doc/packages/openssh/RFC.nroff`). This initial connection phase can be watched closely by turning on the verbose debugging option `-v` of the SSH client.

Version 2 of the SSH protocol is used by default. Override this to use version 1 of the protocol with the `-1` switch. The client stores all public host keys in `~/.ssh/known_hosts` after its first contact with a remote host. This prevents any man-in-the-middle attacks—attempts by foreign SSH servers to use spoofed names and IP addresses. Such attacks are detected either by a host key that is not included in `~/.ssh/known_hosts` or by the server's inability to decrypt the session key in the absence of an appropriate private counterpart.

It is recommended to back up the private and public keys stored in `/etc/ssh/` in a secure, external location. In this way, key modifications can be detected and the old ones can be used again after a reinstallation. This spares users any unsettling warnings. If it is verified that, despite the warning, it is indeed the correct SSH server, the existing entry for the system must be removed from `~/.ssh/known_hosts`.

23.2.6 SSH Authentication Mechanisms

Now the actual authentication takes place, which, in its simplest form, consists of entering a password as mentioned above. The goal of SSH was to introduce a secure software that is also easy to use. Because it is meant to replace rsh and rlogin, SSH must also be able to provide an authentication method appropriate for daily use. SSH accomplishes this by way of another key pair, which is generated by the user. The SSH package provides a helper program for this: ssh-keygen. After entering `ssh-keygen -t rsa` or `ssh-keygen -t dsa`, the key pair is generated and you are prompted for the base filename in which to store the keys.

Confirm the default setting and answer the request for a passphrase. Even if the software suggests an empty passphrase, a text from 10 to 30 characters is recommended for the procedure described here. Do not use short and simple words or phrases. Confirm by repeating the passphrase. Subsequently, you will see where the private and public keys are stored, in this example, the files `id_rsa` and `id_rsa.pub`.

Use `ssh-keygen -p -t rsa` or `ssh-keygen -p -t dsa` to change your old passphrase. Copy the public key component (`id_rsa.pub` in the example) to the remote machine and save it to `~/.ssh/authorized_keys`. You will be asked to authenticate yourself with your passphrase the next time you establish a connection. If this does not occur, verify the location and contents of these files.

In the long run, this procedure is more troublesome than giving your password each time. Therefore, the SSH package provides another tool, ssh-agent, which retains the private keys for the duration of an X session. The entire X session is started as a child process of ssh-agent. The easiest way to do this is to set the variable `usessh` at the beginning of the `.xsession` file to `yes` and log in via a display manager, such as KDM or XDM. Alternatively, enter `ssh-agent startx`.

Now you can use ssh or scp as usual. If you have distributed your public key as described above, you are no longer prompted for your password. Take care of terminating your X session or locking it with a password protection application, such as xlock.

All the relevant changes that resulted from the introduction of version 2 of the SSH protocol are also documented in the file `/usr/share/doc/packages/openssh/README.SuSE`.

23.2.7 X, Authentication, and Forwarding Mechanisms

Beyond the previously described security-related improvements, SSH also simplifies the use of remote X applications. If you run `ssh` with the option `-X`, the DISPLAY variable is automatically set on the remote machine and all X output is exported to the remote machine over the existing SSH connection. At the same time, X applications started remotely and locally viewed with this method cannot be intercepted by unauthorized individuals.

By adding the option `-A`, the ssh-agent authentication mechanism is carried over to the next machine. This way, you can work from different machines without having to enter a password, but only if you have distributed your public key to the destination hosts and properly saved it there.

Both mechanisms are deactivated in the default settings, but can be permanently activated at any time in the systemwide configuration file `/etc/ssh/sshd_config` or the user's `~/.ssh/config`.

ssh can also be used to redirect TCP/IP connections. In the examples below, SSH is told to redirect the SMTP and the POP3 port, respectively:

```
ssh -L 25:sun:25 earth
```

With this command, any connection directed to earth port 25 (SMTP) is redirected to the SMTP port on sun via an encrypted channel. This is especially useful for those using SMTP servers without SMTP-AUTH or POP-before-SMTP features. From any arbitrary location connected to a network, e-mail can be transferred to the “home” mail server for delivery. Similarly, all POP3 requests (port 110) on earth can be forwarded to the POP3 port of sun with this command:

```
ssh -L 110:sun:110 earth
```

Both commands must be executed as `root`, because the connection is made to privileged local ports. E-mail is sent and retrieved by normal users in an existing SSH connection. The SMTP and POP3 host must be set to `localhost` for this to work. Additional information can be found in the manual pages for each of the programs described above and also in the files under `/usr/share/doc/packages/openssh`.

23.3 Encrypting Partitions and Files

Every user has some confidential data that third parties should not be able to access. The more connected and mobile you are, the more carefully you should handle your data. The encryption of files or entire partitions is recommended if others have access over a network connection or direct physical access.

WARNING: Encrypted Media Is Limited Protection

Be aware that with the methods described in this section, you cannot protect your running system from being compromised. After the encrypted media is successfully mounted, everybody with appropriate permissions have access to it. Encrypted media makes sense if you lose your computer or it is stolen and unauthorized individuals want to read your confidental data.

The following list features a number of imaginable usage scenarios.

Laptops

If you travel with your laptop, it is a good idea to encrypt hard disk partitions containing confidential data. If you lose your laptop or if it is stolen, your data will be out of reach if it resides in an encrypted file system or a single encrypted file.

Removable Media

USB flash drives or external hard disks are as prone to being stolen as laptops. An encrypted file system provides protection against third-party access.

Workstations

In companies where almost everyone has access to your computer, it can makes sense to encrypt partition or single files.

23.3.1 Setting Up a Crypto File System with YaST

YaST offers the encryption of files or partitions during installation as well as in an already installed system. An encrypted file can be created at any time, because it fits nicely in an existing partition layout. To encrypt an entire partition, dedicate a partition for encryption in the partition layout. The standard partitioning proposal as suggested by YaST does not, by default, include an encrypted partition. Add it manually in the partitioning dialog.

Creating an Encrypted Partition during Installation

WARNING: Password Input

Observe the warnings about password security when setting the password for encrypted partitions and memorize it well. Without the password, the encrypted data cannot be accessed or restored.

The YaST expert dialog for partitioning, described in Section “Partitioner” (Chapter 3, *System Configuration with YaST*, ↑Start-Up), offers the options needed for creating an encrypted partition. Click *Create* like when creating a regular partition. In the dialog that opens, enter the partitioning parameters for the new partition, such as the desired formatting and the mount point. Complete the process by clicking *Encrypt File System*. In the following dialog, enter the password twice. The new encrypted partition is created after the partitioning dialog is closed by clicking *OK*. While booting, the operating system requests the password before mounting the partition.

If you do not want to mount the encrypted partition during start-up, click Enter when prompted for the password. Then decline the offer to enter the password again. In this case, the encrypted file system is not mounted and the operating system continues booting, blocking access to your data. The partition is available to all users once it has been mounted.

If the encrypted file system should only be mounted when necessary, enable *Do Not Mount During Booting* in the *fstab Options* dialog. The respective partition will not be mounted when the system is booted. To make it available afterwards, mount it manually with `mount name_of_partition mount_point`. Enter the password when prompted to do so. After finishing your work with the partition, unmount it with `umount name_of_partition` to protect it from access by other users.

Creating an Encrypted Partition on a Running System

WARNING: Activating Encryption in a Running System

It is also possible to create encrypted partitions on a running system like during installation. However, encrypting an existing partition destroys all data on it.

On a running system, select *System* → *Partitioning* in the YaST control center. Click *Yes* to proceed. Instead of selecting *Create* as mentioned above, click *Edit*. The rest of the procedure is the same.

Installing Encrypted Files

Instead of using a partition, it is possible to create encrypted file systems within single files for holding confidential data. These are created from the same YaST dialog. Select *Crypt File* and enter the path to the file to create along with its intended size. Accept the proposed formatting settings and the file system type. Then specify the mount point and decide whether the encrypted file system should be mounted when the system is booted.

The advantage of encrypted files is that they can be added without repartitioning the hard disk. They are mounted with the help of a loop device and behave just like normal partitions.

Using vi to Encrypt Files

The disadvantage of using encrypted partitions is that while the partition is mounted, at least `root` can access the data. To prevent this, vi can be used in encrypted mode.

Use `vi -x` *`filename`* to edit a new file. vi prompts you to set a password, after which it encrypts the content of the file. Whenever you access this file, vi requests the correct password.

For even more security, you can place the encrypted text file in an encrypted partition. This is recommended because the encryption used in vi is not very strong.

23.3.2 Encrypting the Content of Removable Media

YaST treats removable media like external hard disks or USB flash drives like any other hard disk. Files or partitions on such media can be encrypted as described above. However, do not select to mount these media when the system is booted, because they are usually only connected while the system is running.

23.4 Security and Confidentiality

One of the main characteristics of a Linux or UNIX system is its ability to handle several users at the same time (multiuser) and to allow these users to perform several tasks (multitasking) on the same computer simultaneously. Moreover, the operating system is network transparent. The users often do not know whether the data and applications they are using are provided locally from their machine or made available over the network.

With the multiuser capability, the data of different users must be stored separately. Security and privacy need to be guaranteed. Data security was already an important issue, even before computers could be linked through networks. Just like today, the most important concern was the ability to keep data available in spite of a lost or otherwise damaged data medium, a hard disk in most cases.

This section is primarily focused on confidentiality issues and on ways to protect the privacy of users, but it cannot be stressed enough that a comprehensive security concept should always include procedures to have a regularly updated, workable, and tested backup in place. Without this, you could have a very hard time getting your data back—not only in the case of some hardware defect, but also if the suspicion arises that someone has gained unauthorized access and tampered with files.

23.4.1 Local Security and Network Security

There are several ways of accessing data:

- personal communication with people who have the desired information or access to the data on a computer

- directly from the console of a computer (physical access)

- over a serial line

- using a network link

In all these cases, a user should be authenticated before accessing the resources or data in question. A Web server might be less restrictive in this respect, but you still would not want it to disclose all your personal data to any surfer.

In the list above, the first case is the one where the highest amount of human interaction is involved, such as when you are contacting a bank employee and are required to prove that you are the person owning that bank account. Then you are asked to provide a signature, a PIN, or a password to prove that you are the person you claim to be. In some cases, it might be possible to elicit some intelligence from an informed person just by mentioning known bits and pieces to win the confidence of that person by using clever rhetoric. The victim could be led to reveal gradually more information, maybe without even becoming aware of it. Among hackers, this is called *social engineering*. You can only guard against this by educating people and by dealing with language and information in a conscious way. Before breaking into computer systems, attackers often try to target receptionists, service people working with the company, or even family members. In many cases, such an attack based on social engineering is only discovered at a much later time.

A person wanting to obtain unauthorized access to your data could also use the traditional way and try to get at your hardware directly. Therefore, the machine should be protected against any tampering so that no one can remove, replace, or cripple its components. This also applies to backups and even any network cable or the power cord. Also secure the boot procedure, because there are some well-known key combinations that might provoke unusual behavior. Protect yourself against this by setting passwords for the BIOS and the boot loader.

Serial terminals connected to serial ports are still used in many places. Unlike network interfaces, they do not rely on a network protocol to communicate with the host. A simple cable or an infrared port is used to send plain characters back and forth between the devices. The cable itself is the weakest point of such a system: with an older printer connected to it, it is easy to record anything that runs over the wires. What can be achieved with a printer can also be accomplished in other ways, depending on the effort that goes into the attack.

Reading a file locally on a host requires other access rules than opening a network connection with a server on a different host. There is a distinction between local security and network security. The line is drawn where data must be put into packets to be sent somewhere else.

Local Security

Local security starts with the physical environment in the location where the computer is running. Set up your machine in a place where security is in line with your expectations and needs. The main goal of local security is to keep users separate from each other, so no user can assume the permissions or the identity of another. This is a general rule to be observed, but it is especially true for the user `root`, who holds the supreme power on the system. `root` can take on the identity of any other local user without being prompted for the password and read any locally stored file.

Passwords

On a Linux system, passwords are not stored as plain text and the text string entered is not simply matched with the saved pattern. If this were the case, all accounts on your system would be compromised as soon as someone got access to the corresponding file. Instead, the stored password is encrypted and, each time it is entered, is encrypted again and the two encrypted strings are compared. This only provides more security if the encrypted password cannot be reverse-computed into the original text string.

This is actually achieved by a special kind of algorithm, also called *trapdoor algorithm*, because it only works in one direction. An attacker who has obtained the encrypted string is not able to get your password by simply applying the same algorithm again.

Instead, it would be necessary to test all the possible character combinations until a combination is found that looks like your password when encrypted. With passwords eight characters long, there are quite a number of possible combinations to calculate.

In the seventies, it was argued that this method would be more secure than others due to the relative slowness of the algorithm used, which took a few seconds to encrypt just one password. In the meantime, however, PCs have become powerful enough to do several hundred thousand or even millions of encryptions per second. Because of this, encrypted passwords should not be visible to regular users (`/etc/shadow` cannot be read by normal users). It is even more important that passwords are not easy to guess, in case the password file becomes visible due to some error. Consequently, it is not really useful to “translate” a password like “tantalize” into “t@nt@1lz3”.

Replacing some letters of a word with similar looking numbers is not safe enough. Password cracking programs that use dictionaries to guess words also play with substitutions like that. A better way is to make up a word with no common meaning, something that only makes sense to you personally, like the first letters of the words of a sentence or the title of a book, such as “The Name of the Rose” by Umberto Eco. This would give the following safe password: “TNotRbUE9”. In contrast, passwords like “beerbuddy” or “jasmine76” are easily guessed even by someone who has only some casual knowledge about you.

The Boot Procedure

Configure your system so it cannot be booted from a floppy or from CD, either by removing the drives entirely or by setting a BIOS password and configuring the BIOS to allow booting from a hard disk only. Normally, a Linux system is started by a boot loader, allowing you to pass additional options to the booted kernel. Prevent others from using such parameters during boot by setting an additional password in `/boot/grub/menu.lst` (see Chapter 29, *The Boot Loader* (page 427)). This is crucial to your system's security. Not only does the kernel itself run with `root` permissions, but it is also the first authority to grant `root` permissions at system start-up.

File Permissions

As a general rule, always work with the most restrictive privileges possible for a given task. For example, it is definitely not necessary to be `root` to read or write e-mail. If the mail program has a bug, this bug could be exploited for an attack that acts with exactly the permissions of the program when it was started. By following the above rule, minimize the possible damage.

The permissions of the more than 200,000 files included in a SUSE distribution are carefully chosen. A system administrator who installs additional software or other files should take great care when doing so, especially when setting the permission bits. Experienced and security-conscious system administrators always use the `-l` option with the command `ls` to get an extensive file list, which allows them to detect any incorrect file permissions immediately. An incorrect file attribute does not only mean that files could be changed or deleted. These modified files could be executed by `root` or, in the case of configuration files, programs could use such files with the permissions of `root`. This significantly increases the possibilities of an attacker. Attacks like this are called cuckoo eggs, because the program (the egg) is executed (hatched) by a different user (bird), just like a cuckoo tricks other birds into hatching its eggs.

A SUSE Linux system includes the files `permissions`, `permissions.easy`, `permissions.secure`, and `permissions.paranoid`, all in the directory `/etc`. The purpose of these files is to define special permissions, such as world-writable directories or, for files, the setuser ID bit (programs with the setuser ID bit set do not run with the permissions of the user that has launched it, but with the permissions of the file owner, in most cases `root`). An administrator can use the file `/etc/permissions.local` to add his own settings.

To define which of the above files is used by SUSE's configuration programs to set permissions accordingly, select *Security* in YaST. To learn more about the topic, read the comments in `/etc/permissions` or consult the manual page of `chmod` (`man chmod`).

Buffer Overflows and Format String Bugs

Special care must be taken whenever a program is supposed to process data that can or could be changed by a user, but this is more of an issue for the programmer of an application than for regular users. The programmer must make sure that his application interprets data in the correct way, without writing it into memory areas that are too small to hold it. Also, the program should hand over data in a consistent manner, using the interfaces defined for that purpose.

A *buffer overflow* can happen if the actual size of a memory buffer is not taken into account when writing to that buffer. There are cases where this data (as generated by the user) uses up some more space than what is available in the buffer. As a result, data is written beyond the end of that buffer area, which, under certain circumstances, makes it possible for a program to execute program sequences influenced by the user (and not by the programmer), rather than just processing user data. A bug of this kind may have serious consequences, especially if the program is being executed with special privileges (see Section “File Permissions” (page 331)).

Format string bugs work in a slightly different way, but again it is the user input that could lead the program astray. In most cases, these programming errors are exploited with programs executed with special permissions—setuid and setgid programs—which also means that you can protect your data and your system from such bugs by removing the corresponding execution privileges from programs. Again, the best way is to apply a policy of using the lowest possible privileges (see Section “File Permissions” (page 331)).

Given that buffer overflows and format string bugs are bugs related to the handling of user data, they are not only exploitable if access has been given to a local account. Many of the bugs that have been reported can also be exploited over a network link. Accordingly, buffer overflows and format string bugs should be classified as being relevant for both local and network security.

Viruses

Contrary to what some people say, there are viruses that run on Linux. However, the viruses that are known were released by their authors as a *proof of concept* to prove that the technique works as intended. None of these viruses have been spotted *in the wild* so far.

Viruses cannot survive and spread without a host on which to live. In this case, the host would be a program or an important storage area of the system, such as the master boot record, which needs to be writable for the program code of the virus. Owing to its multiuser capability, Linux can restrict write access to certain files, especially important with system files. Therefore, if you did your normal work with `root` permissions, you would increase the chance of the system being infected by a virus. In contrast, if you follow the principle of using the lowest possible privileges as mentioned above, chances of getting a virus are slim.

Apart from that, you should never rush into executing a program from some Internet site that you do not really know. SUSE's RPM packages carry a cryptographic signature as a digital label that the necessary care was taken to build them. Viruses are a typical sign that the administrator or the user lacks the required security awareness, putting at risk even a system that should be highly secure by its very design.

Viruses should not be confused with worms, which belong to the world of networks entirely. Worms do not need a host to spread.

Network Security

Network security is important for protecting from an attack that is started outside. The typical login procedure requiring a username and a password for user authentication is still a local security issue. In the particular case of logging in over a network, differentiate between the two security aspects. What happens until the actual authentication is network security and anything that happens afterwards is local security.

X Window System and X Authentication

As mentioned at the beginning, network transparency is one of the central characteristics of a UNIX system. X, the windowing system of UNIX operating systems, can make use of this feature in an impressive way. With X, it is basically no problem to log in at a remote host and start a graphical program that is then sent over the network to be displayed on your computer.

When an X client should be displayed remotely using an X server, the latter should protect the resource managed by it (the display) from unauthorized access. In more concrete terms, certain permissions must be given to the client program. With the X Window System, there are two ways to do this, called host-based access control and cookie-based access control. The former relies on the IP address of the host where the client should run. The program to control this is xhost. xhost enters the IP address of a legitimate client into a tiny database belonging to the X server. However, relying on IP addresses for authentication is not very secure. For example, if there were a second user working on the host sending the client program, that user would have access to the X server as well—just like someone stealing the IP address. Because of these shortcomings, this authentication method is not described in more detail here, but you can learn about it with `man xhost`.

In the case of cookie-based access control, a character string is generated that is only known to the X server and to the legitimate user, just like an ID card of some kind. This cookie (the word goes back not to ordinary cookies, but to Chinese fortune cookies, which contain an epigram) is stored on login in the file `.Xauthority` in the user's home directory and is available to any X client wanting to use the X server to display a window. The file `.Xauthority` can be examined by the user with the tool xauth. If you were to rename `.Xauthority` or if you deleted the file from your home directory by accident, you would not be able to open any new windows or X clients. Read more about X Window System security mechanisms in the man page of Xsecurity (`man Xsecurity`).

SSH (secure shell) can be used to encrypt a network connection completely and forward it to an X server transparently without the encryption mechanism being perceived by the user. This is also called X forwarding. X forwarding is achieved by simulating an X server on the server side and setting a DISPLAY variable for the shell on the remote host. Further details about SSH can be found in Section 23.2, "SSH: Secure Network Operations" (page 320).

WARNING

If you do not consider the host where you log in to be a secure host, do not use X forwarding. With X forwarding enabled, an attacker could authenticate via your SSH connection to intrude on your X server and sniff your keyboard input, for instance.

Buffer Overflows and Format String Bugs

As discussed in Section "Buffer Overflows and Format String Bugs" (page 332), buffer overflows and format string bugs should be classified as issues concerning both local and network security. As with the local variants of such bugs, buffer overflows in network programs, when successfully exploited, are mostly used to obtain `root` permissions. Even if that is not the case, an attacker could use the bug to gain access to an unprivileged local account to exploit any other vulnerabilities that might exist on the system.

Buffer overflows and format string bugs exploitable over a network link are certainly the most frequent form of remote attacks in general. Exploits for these—programs to exploit these newly-found security holes—are often posted on the security mailing lists. They can be used to target the vulnerability without knowing the details of the code. Over the years, experience has shown that the availability of exploit codes has contributed to more secure operating systems, obviously due to the fact that operating system makers were forced to fix the problems in their software. With free software, anyone has access to the source code (SUSE Linux comes with all available source codes) and anyone who finds a vulnerability and its exploit code can submit a patch to fix the corresponding bug.

Denial of Service

The purpose of a denial of service (DoS) attack is to block a server program or even an entire system, something that could be achieved by various means: overloading the server, keeping it busy with garbage packets, or exploiting a remote buffer overflow. Often a DoS attack is made with the sole purpose of making the service disappear. However, once a given service has become unavailable, communications could become vulnerable to *man-in-the-middle attacks* (sniffing, TCP connection hijacking, spoofing) and DNS poisoning.

Man in the Middle: Sniffing, Hijacking, Spoofing

In general, any remote attack performed by an attacker who puts himself between the communicating hosts is called a *man-in-the-middle attack*. What almost all types of man-in-the-middle attacks have in common is that the victim is usually not aware that there is something happening. There are many possible variants, for example, the attacker could pick up a connection request and forward that to the target machine. Now the victim has unwittingly established a connection with the wrong host, because the other end is posing as the legitimate destination machine.

The simplest form of a man-in-the-middle attack is called *sniffer*—the attacker is "just" listening to the network traffic passing by. As a more complex attack, the "man in the middle" could try to take over an already established connection (hijacking). To do so, the attacker would need to analyze the packets for some time to be able to predict the TCP sequence numbers belonging to the connection. When the attacker finally seizes the role of the target host, the victims notice this, because they get an error message saying the connection was terminated due to a failure. The fact that there are protocols not secured against hijacking through encryption, which only perform a simple authentication procedure upon establishing the connection, makes it easier for attackers.

Spoofing is an attack where packets are modified to contain counterfeit source data, usually the IP address. Most active forms of attack rely on sending out such fake packets—something that, on a Linux machine, can only be done by the superuser (`root`).

Many of the attacks mentioned are carried out in combination with a DoS. If an attacker sees an opportunity to bring down a certain host abruptly, even if only for a short time, it makes it easier for him to push the active attack, because the host will not be able to interfere with the attack for some time.

DNS Poisoning

DNS poisoning means that the attacker corrupts the cache of a DNS server by replying to it with spoofed DNS reply packets, trying to get the server to send certain data to a victim who is requesting information from that server. Many servers maintain a trust relationship with other hosts, based on IP addresses or hostnames. The attacker needs a good understanding of the actual structure of the trust relationships among hosts to disguise itself as one of the trusted hosts. Usually, the attacker analyzes some packets received from the server to get the necessary information. The attacker often needs to target a well-timed DoS attack at the name server as well. Protect yourself by using encrypted connections that are able to verify the identity of the hosts to which to connect.

Worms

Worms are often confused with viruses, but there is a clear difference between the two. Unlike viruses, worms do not need to infect a host program to live. Instead, they are specialized to spread as quickly as possible on network structures. The worms that appeared in the past, such as Ramen, Lion, or Adore, make use of well-known security holes in server programs like bind8 or lprNG. Protection against worms is relatively easy. Given that some time elapses between the discovery of a security hole and the moment the worm hits your server, there is a good chance that an updated version of the affected program is available on time. That is only useful if the administrator actually installs the security updates on the systems in question.

23.4.2 Some General Security Tips and Tricks

To handle security competently, it is important to keep up with new developments and stay informed about the latest security issues. One very good way to protect your systems against problems of all kinds is to get and install the updated packages recommended by security announcements as quickly as possible. SUSE security announcements are published on a mailing list to which you can subscribe by following the link http://

www.novell.com/linux/security/securitysupport.html. The list suse-security-announce@suse.de is a first-hand source of information regarding updated packages and includes members of SUSE's security team among its active contributors.

The mailing list suse-security@suse.de is a good place to discuss any security issues of interest. Subscribe to it on the same Web page.

bugtraq@securityfocus.com is one of the best-known security mailing lists worldwide. Reading this list, which receives between 15 and 20 postings per day, is recommended. More information can be found at http://www.securityfocus.com.

The following is a list of rules you may find useful in dealing with basic security concerns:

- According to the rule of using the most restrictive set of permissions possible for every job, avoid doing your regular jobs as `root`. This reduces the risk of getting a cuckoo egg or a virus and protects you from your own mistakes.

- If possible, always try to use encrypted connections to work on a remote machine. Using `ssh` (secure shell) to replace `telnet`, `ftp`, `rsh`, and `rlogin` should be standard practice.

- Avoid using authentication methods based on IP addresses alone.

- Try to keep the most important network-related packages up-to-date and subscribe to the corresponding mailing lists to receive announcements on new versions of such programs (bind, sendmail, ssh, etc.). The same should apply to software relevant to local security.

- Change the `/etc/permissions` file to optimize the permissions of files crucial to your system's security. If you remove the setuid bit from a program, it might well be that it cannot do its job anymore in the intended way. On the other hand, consider that, in most cases, the program will also have ceased to be a potential security risk. You might take a similar approach with world-writable directories and files.

- Disable any network services you do not absolutely require for your server to work properly. This makes your system safer. Open ports, with the socket state LISTEN, can be found with the program netstat. As for the options, it is recommended to use `netstat -ap` or `netstat -anp`. The `-p` option allows you to see which process is occupying a port under which name.

Compare the netstat results with those of a thorough port scan done from outside your host. An excellent program for this job is nmap, which not only checks out the ports of your machine, but also draws some conclusions as to which services are waiting behind them. However, port scanning may be interpreted as an aggressive act, so do not do this on a host without the explicit approval of the administrator. Finally, remember that it is important not only to scan TCP ports, but also UDP ports (options `-sS` and `-sU`).

- To monitor the integrity of the files of your system in a reliable way, use the program AIDE (Advanced Intrusion Detection Environment), available on SUSE Linux. Encrypt the database created by AIDE to prevent someone from tampering with it. Furthermore, keep a backup of this database available outside your machine, stored on an external data medium not connected to it by a network link.

- Take proper care when installing any third-party software. There have been cases where a hacker had built a trojan horse into the tar archive of a security software package, which was fortunately discovered very quickly. If you install a binary package, have no doubts about the site from which you downloaded it.

  SUSE's RPM packages are gpg-signed. The key used by SUSE for signing is:

  ID:9C800ACA 2000-10-19 SUSE Package Signing Key <build@suse.de>

  Key fingerprint = 79C1 79B2 E1C8 20C1 890F 9994 A84E DAE8 9C80 0ACA

  The command `rpm --checksig package.rpm` shows whether the checksum and the signature of an uninstalled package are correct. Find the key on the first CD of the distribution and on most key servers worldwide.

- Check your backups of user and system files regularly. Consider that if you do not test whether the backup works, it might actually be worthless.

- Check your log files. Whenever possible, write a small script to search for suspicious entries. Admittedly, this is not exactly a trivial task. In the end, only you can know which entries are unusual and which are not.

- Use `tcp_wrapper` to restrict access to the individual services running on your machine, so you have explicit control over which IP addresses can connect to a service. For further information regarding `tcp_wrapper`, consult the manual pages of tcpd and hosts_access (`man 8 tcpd`, `man hosts_access`).

- Use SuSEfirewall to enhance the security provided by tcpd (tcp_wrapper).

- Design your security measures to be redundant: a message seen twice is much better than no message at all.

23.4.3 Using the Central Security Reporting Address

If you discover a security-related problem (please check the available update packages first), write an e-mail to security@suse.de. Please include a detailed description of the problem and the version number of the package concerned. SUSE will try to send a reply as soon as possible. You are encouraged to pgp encrypt your e-mail messages. SUSE's pgp key is:

```
ID:3D25D3D9 1999-03-06 SUSE Security Team <security@suse.de>
Key fingerprint = 73 5F 2E 99 DF DB 94 C4 8F 5A A3 AE AF 22 F2 D5
```

This key is also available for download from http://www.novell.com/linux/security/securitysupport.html.

Access Control Lists in Linux 24

This chapter provides a brief summary of the background and functions of POSIX ACLs (access control lists) for Linux file systems. ACLs can be used as an expansion of the traditional permission concept for file system objects. With ACLs, permissions can be defined more flexibly than the traditional permission concept allows.

The term *POSIX ACL* suggests that this is a true POSIX (*portable operating system interface*) standard. The respective draft standards POSIX 1003.1e and POSIX 1003.2c have been withdrawn for several reasons. Nevertheless, ACLs as found on many systems belonging to the UNIX family are based on these drafts and the implementation of file system ACLs as described in this chapter follows these two standards as well. They can be viewed at `http://wt.xpilot.org/publications/posix.1e/`.

24.1 Advantages of ACLs

Traditionally, three sets of permissions are defined for each file object on a Linux system. These sets include the read (`r`), write (`w`), and execute (`x`) permissions for each of three types of users—the file owner, the group, and other users. In addition to that, it is possible to set the *set user id*, the *set group id*, and the *sticky* bit. This lean concept is fully adequate for most practical cases. However, for more complex scenarios or advanced applications, system administrators formerly had to use a number of tricks to circumvent the limitations of the traditional permission concept.

ACLs can be used for situations that require an extension of the traditional file permission concept. They allow assignment of permissions to individual users or groups even if these do not correspond to the original owner or the owning group. Access control lists are a feature of the Linux kernel and are currently supported by ReiserFS, Ext2, Ext3, JFS, and XFS. Using ACLs, complex scenarios can be realized without implementing complex permission models on the application level.

The advantages of ACLs are clearly evident in a situation like replacement of a Windows server with a Linux server. Some of the connected workstations may continue to run under Windows even after the migration. The Linux system offers file and print services to the Windows clients with Samba. Given that Samba supports access control lists, user permissions can be configured both on the Linux server and in Windows with a graphical user interface (only Windows NT and later). With winbindd, it is even possible to assign permissions to users that only exist in the Windows domain without any account on the Linux server.

24.2 Definitions

user class

The conventional POSIX permission concept uses three *classes* of users for assigning permissions in the file system: the owner, the owning group, and other users. Three permission bits can be set for each user class, giving permission to read (`r`), write (`w`), and execute (`x`).

access ACL

The user and group access permissions for all kinds of file system objects (files and directories) are determined by means of access ACLs.

default ACL

Default ACLs can only be applied to directories. They determine the permissions a file system object inherits from its parent directory when it is created.

ACL entry

Each ACL consists of a set of ACL entries. An ACL entry contains a type (see Table 24.1, “ACL Entry Types” (page 343)), a qualifier for the user or group to which the entry refers, and a set of permissions. For some entry types, the qualifier for the group or users is undefined.

24.3 Handling ACLs

Table 24.1, “ACL Entry Types” (page 343) summarizes the six possible types of ACL entries, each defining permissions for a user or a group of users. The *owner* entry defines the permissions of the user owning the file or directory. The *owning group* entry defines the permissions of the file's owning group. The superuser can change the owner or owning group with `chown` or `chgrp`, in which case the owner and owning group entries refer to the new owner and owning group. Each *named user* entry defines the permissions of the user specified in the entry's qualifier field, which is the middle field in the text form shown in Table 24.1, “ACL Entry Types” (page 343). Each *named group* entry defines the permissions of the group specified in the entry's qualifier field. Only the named user and named group entries have a qualifier field that is not empty. The *other* entry defines the permissions of all other users.

The *mask* entry further limits the permissions granted by *named user*, *named group*, and *owning group* entries by defining which of the permissions in those entries are effective and which are masked. If permissions exist in one of the mentioned entries as well as in the mask, they are effective. Permissions contained only in the mask or only in the actual entry are not effective—meaning the permissions are not granted. All permissions defined in the *owner* and *owning group* entries are always effective. The example in Table 24.2, “Masking Access Permissions” (page 344) demonstrates this mechanism.

There are two basic classes of ACLs: A *minimum* ACL contains only the entries for the types *owner*, *owning group*, and *other*, which correspond to the conventional permission bits for files and directories. An *extended* ACL goes beyond this. It must contain a *mask* entry and may contain several entries of the *named user* and *named group* types.

Table 24.1 *ACL Entry Types*

Type	Text Form
owner	`user::rwx`
named user	`user:name:rwx`
owning group	`group::rwx`
named group	`group:name:rwx`

Type	Text Form
mask	`mask::rwx`
other	`other::rwx`

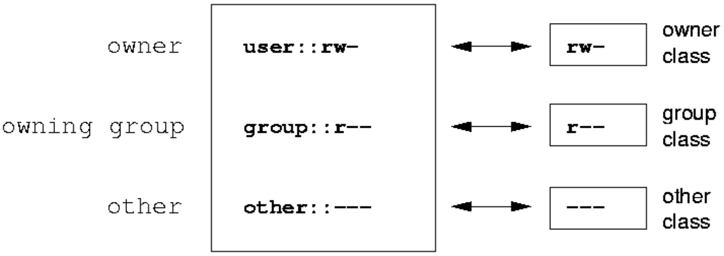
***Table 24.2** Masking Access Permissions*

Entry Type	Text Form	Permissions
named user	`user:geeko:r-x`	`r-x`
mask	`mask::rw-`	`rw-`
	effective permissions:	`r--`

24.3.1 ACL Entries and File Mode Permission Bits

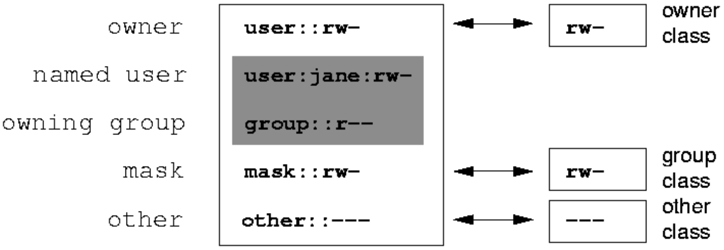
Figure 24.1, "Minimum ACL: ACL Entries Compared to Permission Bits" (page 345) and Figure 24.2, "Extended ACL: ACL Entries Compared to Permission Bits" (page 345) illustrate the two cases of a minimum ACL and an extended ACL. The figures are structured in three blocks—the left block shows the type specifications of the ACL entries, the center block displays an example ACL, and the right block shows the respective permission bits according to the conventional permission concept, for example, as displayed by `ls -l`. In both cases, the *owner class* permissions are mapped to the ACL entry *owner*. *Other class* permissions are mapped to the respective ACL entry. However, the mapping of the *group class* permissions is different in the two cases.

Figure 24.1 Minimum ACL: ACL Entries Compared to Permission Bits



In the case of a minimum ACL—without *mask*—the *group class* permissions are mapped to the ACL entry *owning group*. This is shown in Figure 24.1, “Minimum ACL: ACL Entries Compared to Permission Bits” (page 345). In the case of an extended ACL—with *mask*—the *group class* permissions are mapped to the *mask* entry. This is shown in Figure 24.2, “Extended ACL: ACL Entries Compared to Permission Bits” (page 345).

Figure 24.2 Extended ACL: ACL Entries Compared to Permission Bits



This mapping approach ensures the smooth interaction of applications, regardless of whether they have ACL support. The access permissions that were assigned by means of the permission bits represent the upper limit for all other “fine adjustments” made with an ACL. Changes made to the permission bits are reflected by the ACL and vice versa.

24.3.2 A Directory with an Access ACL

The handling of access ACLs is demonstrated in the following example:

Before you create the directory, use the `umask` command to define which access permissions should be masked each time a file object is created. The command `umask 027` sets the default permissions by giving the owner the full range of permissions (`0`), denying the group write access (2), and giving other users no permissions at all (7). umask actually masks the corresponding permission bits or turns them off. For details, consult the corresponding man page (man umask).

mkdir mydir should create the mydir directory with the default permissions as set by umask. Use ls -dl mydir to check if all permissions were assigned correctly. The output for this example is:

```
drwxr-x--- ... tux project3 ... mydir
```

With getfacl mydir, check the initial state of the ACL. This gives information like:

```
# file: mydir
# owner: tux
# group: project3
user::rwx
group::r-x
other::---
```

The output of getfacl precisely reflects the mapping of permission bits and ACL entries as described in Section 24.3.1, "ACL Entries and File Mode Permission Bits" (page 344). The first three output lines display the name, owner, and owning group of the directory. The next three lines contain the three ACL entries *owner*, *owning group*, and *other*. In fact, in the case of this minimum ACL, the getfacl command does not produce any information you could not have obtained with ls.

Modify the ACL to assign read, write, and execute permissions to an additional user geeko and an additional group mascots with:

```
setfacl -m user:geeko:rwx,group:mascots:rwx mydir
```

The option -m prompts setfacl to modify the existing ACL. The following argument indicates the ACL entries to modify (multiple entries are separated by commas). The final part specifies the name of the directory to which these modifications should be applied. Use the getfacl command to take a look at the resulting ACL.

```
# file: mydir
# owner: tux
# group: project3
user::rwx
user:geeko:rwx
group::r-x
group:mascots:rwx
mask::rwx
other::---
```

In addition to the entries initiated for the user geeko and the group mascots, a *mask* entry has been generated. This *mask* entry is set automatically so that all permissions are effective. setfacl automatically adapts existing *mask* entries to the settings modified, unless you deactivate this feature with -n. *mask* defines the maximum effective access permissions for all entries in the *group class*. This includes *named user*, *named group*, and *owning group*. The *group class* permission bits displayed by ls -dl mydir now correspond to the mask entry.

```
drwxrwx---+ ... tux project3 ... mydir
```

The first column of the output now contains an additional + to indicate that there is an *extended* ACL for this item.

According to the output of the ls command, the permissions for the *mask* entry include write access. Traditionally, such permission bits would mean that the *owning group* (here project3) also has write access to the directory mydir. However, the effective access permissions for the *owning group* correspond to the overlapping portion of the permissions defined for the *owning group* and for the *mask*—which is r-x in our example (see Table 24.2, “Masking Access Permissions” (page 344)). As far as the effective permissions of the *owning group* in this example are concerned, nothing has changed even after the addition of the ACL entries.

Edit the *mask* entry with setfacl or chmod. For example, use chmod g-w mydir. ls -dl mydir then shows:

```
drwxr-x---+ ... tux project3 ... mydir
```

getfacl mydir provides the following output:

```
# file: mydir
# owner: tux
# group: project3
user::rwx
user:geeko:rwx          # effective: r-x
group::r-x
group:mascots:rwx      # effective: r-x
mask::r-x
other::---
```

After executing the chmod command to remove the write permission from the *group class* bits, the output of the ls command is sufficient to see that the *mask* bits must have changed accordingly: write permission is again limited to the owner of mydir. The output of the getfacl confirms this. This output includes a comment for all those entries in which the effective permission bits do not correspond to the original permissions, because they are filtered according to the *mask* entry. The original permissions can be restored at any time with `chmod g+w mydir`.

24.3.3 A Directory with a Default ACL

Directories can have a default ACL, which is a special kind of ACL defining the access permissions that objects in the directory inherit when they are created. A default ACL affects both subdirectories and files.

Effects of a Default ACL

There are two different ways in which the permissions of a directory's default ACL are passed to the files and subdirectories in it:

- A subdirectory inherits the default ACL of the parent directory both as its default ACL and as an access ACL.

- A file inherits the default ACL as its access ACL.

All system calls that create file system objects use a `mode` parameter that defines the access permissions for the newly created file system object. If the parent directory does not have a default ACL, the permission bits as defined by the `umask` are subtracted from the permissions as passed by the `mode` parameter, with the result being assigned to the new object. If a default ACL exists for the parent directory, the permission bits assigned to the new object correspond to the overlapping portion of the permissions of the `mode` parameter and those that are defined in the default ACL. The `umask` is disregarded in this case.

Application of Default ACLs

The following three examples show the main operations for directories and default ACLs:

1. Add a default ACL to the existing directory `mydir` with:

   ```
   setfacl -d -m group:mascots:r-x mydir
   ```

   The option `-d` of the `setfacl` command prompts `setfacl` to perform the following modifications (option `-m`) in the default ACL.

Take a closer look at the result of this command:

```
getfacl mydir

# file: mydir
# owner: tux
# group: project3
user::rwx
user:geeko:rwx
group::r-x
group:mascots:rwx
mask::rwx
other::---
default:user::rwx
default:group::r-x
default:group:mascots:r-x
default:mask::r-x
default:other::---
```

`getfacl` returns both the access ACL and the default ACL. The default ACL is formed by all lines that start with `default`. Although you merely executed the `setfacl` command with an entry for the `mascots` group for the default ACL, `setfacl` automatically copied all other entries from the access ACL to create a valid default ACL. Default ACLs do not have an immediate effect on access permissions. They only come into play when file system objects are created. These new objects inherit permissions only from the default ACL of their parent directory.

2. In the next example, use `mkdir` to create a subdirectory in `mydir`, which inherits the default ACL.

```
mkdir mydir/mysubdir

getfacl mydir/mysubdir

# file: mydir/mysubdir
# owner: tux
# group: project3
user::rwx
group::r-x
group:mascots:r-x
mask::r-x
other::---
default:user::rwx
default:group::r-x
default:group:mascots:r-x
default:mask::r-x
default:other::---
```

As expected, the newly-created subdirectory mysubdir has the permissions from the default ACL of the parent directory. The access ACL of mysubdir is an exact reflection of the default ACL of mydir. The default ACL that this directory will hand down to its subordinate objects is also the same.

3. Use touch to create a file in the mydir directory, for example, touch mydir/myfile. ls -l mydir/myfile then shows:

   ```
   -rw-r-----+ ... tux project3 ... mydir/myfile
   ```

   The output of getfacl mydir/myfile is:

   ```
   # file: mydir/myfile
   # owner: tux
   # group: project3
   user::rw-
   group::r-x          # effective:r--
   group:mascots:r-x   # effective:r--
   mask::r--
   other::---
   ```

   touch uses a mode with the value 0666 when creating new files, which means that the files are created with read and write permissions for all user classes, provided no other restrictions exist in umask or in the default ACL (see Section "Effects of a Default ACL" (page 348)). In effect, this means that all access permissions not contained in the mode value are removed from the respective ACL entries. Although no permissions were removed from the ACL entry of the *group class*, the *mask* entry was modified to mask permissions not set in mode.

   This approach ensures the smooth interaction of applications, such as compilers, with ACLs. You can create files with restricted access permissions and subsequently mark them as executable. The mask mechanism guarantees that the right users and groups can execute them as desired.

24.3.4 The ACL Check Algorithm

A check algorithm is applied before any process or application is granted access to an ACL-protected file system object. As a basic rule, the ACL entries are examined in the following sequence: *owner*, *named user*, *owning group* or *named group*, and *other*. The access is handled in accordance with the entry that best suits the process. Permissions do not accumulate.

Things are more complicated if a process belongs to more than one group and would potentially suit several *group* entries. An entry is randomly selected from the suitable entries with the required permissions. It is irrelevant which of the entries triggers the final result “access granted”. Likewise, if none of the suitable *group* entries contains the required permissions, a randomly selected entry triggers the final result “access denied”.

24.4 ACL Support in Applications

ACLs can be used to implement very complex permission scenarios that meet the requirements of modern applications. The traditional permission concept and ACLs can be combined in a smart manner. The basic file commands (`cp`, `mv`, `ls`, etc.) support ACLs, as does Samba.

Unfortunately, many editors and file managers still lack ACL support. When copying files with Konqueror, for instance, the ACLs of these files are lost. When modifying files with an editor, the ACLs of files are sometimes preserved and sometimes not, depending on the backup mode of the editor used. If the editor writes the changes to the original file, the access ACL is preserved. If the editor saves the updated contents to a new file that is subsequently renamed to the old filename, the ACLs may be lost, unless the editor supports ACLs. Except for the star archiver, there are currently no backup applications that preserve ACLs.

24.5 For More Information

Detailed information about ACLs is available at `http://acl.bestbits.at/`. Also see the man pages for `getfacl(1)`, `acl(5)`, and `setfacl(1)`.

25 System Monitoring Utilities

A number of programs and mechanisms, some of which are presented here, can be used to examine the status of your system. Also described are some utilities that are useful for routine work, along with their most important parameters.

For each of the commands introduced, examples of the relevant outputs are presented. In these examples, the first line is the command itself (after the dollar sign prompt). Omissions are indicated with square brackets (`[...]`) and long lines are wrapped where necessary. Line breaks for long lines are indicated by a backslash (\).

```
$ command -x -y
output line 1
output line 2
output line 3 is annoyingly long, so long that \
    we have to break it
output line 3
[...]
output line 98
output line 99
```

The descriptions have been kept short to allow as many utilities as possible to be mentioned. Further information for all the commands can be found in the man pages. Most of the commands also understand the parameter `--help`, which produces a brief list of the possible parameters.

25.1 List of Open Files: lsof

To view a list of all the files open for the process with process ID *`PID`*, use `-p`. For example, to view all the files used by the current shell, enter:

```
$ lsof -p $$
COMMAND  PID USER   FD   TYPE DEVICE    SIZE     NODE NAME
zsh     4694   jj  cwd    DIR   0,18     144 25487368 /suse/jj/t
(totan:/real-home/jj)
zsh     4694   jj  rtd    DIR    3,2     608        2 /
zsh     4694   jj  txt    REG    3,2  441296    20414 /bin/zsh
zsh     4694   jj  mem    REG    3,2  104484    10882 /lib/ld-2.3.3.so
zsh     4694   jj  mem    REG    3,2   11648    20610
/usr/lib/zsh/4.2.0/zsh/rlimits.so
[...]
zsh     4694   jj  mem    REG    3,2   13647    10891 /lib/libdl.so.2
zsh     4694   jj  mem    REG    3,2   88036    10894 /lib/libnsl.so.1
zsh     4694   jj  mem    REG    3,2  316410   147725 /lib/libncurses.so.5.4
zsh     4694   jj  mem    REG    3,2  170563    10909 /lib/tls/libm.so.6
zsh     4694   jj  mem    REG    3,2 1349081    10908 /lib/tls/libc.so.6
zsh     4694   jj  mem    REG    3,2      56    12410
/usr/lib/locale/de_DE.utf8/LC_TELEPHONE
[...]
zsh     4694   jj  mem    REG    3,2      59    14393
/usr/lib/locale/en_US/LC_NUMERIC
zsh     4694   jj  mem    REG    3,2  178476    14565
/usr/lib/locale/en_US/LC_CTYPE
zsh     4694   jj  mem    REG    3,2   56444    20598
/usr/lib/zsh/4.2.0/zsh/computil.so
zsh     4694   jj    0u   CHR 136,48              50 /dev/pts/48
zsh     4694   jj    1u   CHR 136,48              50 /dev/pts/48
zsh     4694   jj    2u   CHR 136,48              50 /dev/pts/48
zsh     4694   jj   10u   CHR 136,48              50 /dev/pts/48
```

The special shell variable `$$`, whose value is the process ID of the shell, has been used.

The command `lsof` lists all the files currently open when used without any parameters. Because there are often thousands of open files, listing all of them is rarely useful. However, the list of all files can be combined with search functions to generate useful lists. For example, list all used character devices:

```
$ lsof | grep CHR
sshd       4685    root  mem    CHR    1,5            45833 /dev/zero
sshd       4685    root  mem    CHR    1,5            45833 /dev/zero
sshd       4693      jj  mem    CHR    1,5            45833 /dev/zero
sshd       4693      jj  mem    CHR    1,5            45833 /dev/zero
zsh        4694      jj    0u   CHR 136,48               50 /dev/pts/48
zsh        4694      jj    1u   CHR 136,48               50 /dev/pts/48
zsh        4694      jj    2u   CHR 136,48               50 /dev/pts/48
zsh        4694      jj   10u   CHR 136,48               50 /dev/pts/48
X          6476    root  mem    CHR    1,1            38042 /dev/mem
lsof      13478      jj    0u   CHR 136,48               50 /dev/pts/48
lsof      13478      jj    2u   CHR 136,48               50 /dev/pts/48
grep      13480      jj    1u   CHR 136,48               50 /dev/pts/48
grep      13480      jj    2u   CHR 136,48               50 /dev/pts/48
```

25.2 User Accessing Files: fuser

It can be useful to determine what processes or users are currently accessing certain files. Suppose, for example, you want to unmount a file system mounted at /mnt. umount returns "device is busy." The command fuser can then be used to determine what processes are accessing the device:

```
$ fuser -v /mnt/*

                     USER        PID ACCESS COMMAND
/mnt/notes.txt
                     jj        26597 f....  less
```

Following termination of the less process, which was running on another terminal, the file system can successfully be unmounted.

25.3 File Properties: stat

The command stat displays file properties:

```
$ stat xml-doc.txt
  File: `xml-doc.txt'
  Size: 632             Blocks: 8          IO Block: 4096   regular file
Device: eh/14d  Inode: 5938009     Links: 1
Access: (0644/-rw-r--r--)  Uid: (11994/      jj)   Gid: (   50/    suse)
Access: 2004-04-27 20:08:58.000000000 +0200
Modify: 2003-06-03 15:29:34.000000000 +0200
Change: 2003-07-23 17:48:27.000000000 +0200
```

The parameter --filesystem produces details of the properties of the file system in which the specified file is located:

```
$ stat  . --filesystem
  File: "."
    ID: 0        Namelen: 255     Type: ext2/ext3
Blocks: Total: 19347388   Free: 17831731   Available: 16848938   Size: 4096
Inodes: Total: 9830400    Free: 9663967
```

If you use the z shell (zsh), you must enter /usr/bin/stat, because the z shell has a shell built-in stat with different options and a different output format:

```
% type stat
stat is a shell builtin
% stat .
```

```
device  769
inode   4554808
mode    16877
nlink   12
uid     11994
gid     50
rdev    0
size    4096
atime   1091536882
mtime   1091535740
ctime   1091535740
blksize 4096
blocks  8
link
```

25.4 USB Devices: lsusb

The command `lsusb` lists all USB devices. With the option `-v`, print a more detailed list. The detailed information is read from the directory `/proc/bus/usb/`. The following is the output of `lsusb` after a USB memory stick was attached. The last lines indicate the presence of the new device.

```
Bus 004 Device 001: ID 0000:0000
Bus 003 Device 001: ID 0000:0000
Bus 002 Device 001: ID 0000:0000
Bus 001 Device 001: ID 0000:0000
Bus 001 Device 018: ID 0402:5634 ALi Corp.
```

25.5 Information about a SCSI Device: scsiinfo

The command `scsiinfo` lists information about a SCSI device. With the option `-l`, list all SCSI devices known to the system (similar information is obtained via the command `lsscsi`). The following is the output of `scsiinfo -i /dev/sda`, which gives information about a hard disk. The option `-a` gives even more information.

```
Inquiry command
---------------
Relative Address                   0
Wide bus 32                        0
Wide bus 16                        1
Synchronous neg.                   1
Linked Commands                    1
```

```
Command Queueing                   1
SftRe                              0
Device Type                        0
Peripheral Qualifier               0
Removable?                         0
Device Type Modifier               0
ISO Version                        0
ECMA Version                       0
ANSI Version                       3
AENC                               0
TrmIOP                             0
Response Data Format               2
Vendor:                    FUJITSU
Product:                   MAS3367NP
Revision level:            0104A0K7P43002BE
```

There is a defects list with two tables of bad blocks of a hard disk: first the one supplied by the vendor (manufacturer table) and second the list of bad blocks that appeared in operation (grown table). If the number of entries in the grown table increases, it might be a good idea to replace the hard disk.

25.6 Processes: top

The command `top`, which stands for "table of processes," displays a list of processes that is refreshed every two seconds. To terminate the program, press Q. The parameter `-n 1` terminates the program after a single display of the process list. The following is an example output of the command `top -n 1`:

```
top - 14:19:53 up 62 days,  3:35, 14 users,  load average: 0.01, 0.02, 0.00
Tasks: 102 total,   7 running,  93 sleeping,   0 stopped,   2 zombie
Cpu(s):   0.3% user,   0.1% system,   0.0% nice,  99.6% idle
Mem:    514736k total,   497232k used,    17504k free,    56024k buffers
Swap:  1794736k total,   104544k used,  1690192k free,   235872k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  Command
 1426 root      15   0  116m  41m  18m S  1.0  8.2  82:30.34 X
20836 jj        15   0   820  820  612 R  1.0  0.2   0:00.03 top
    1 root      15   0   100   96   72 S  0.0  0.0   0:08.43 init
    2 root      15   0     0    0    0 S  0.0  0.0   0:04.96 keventd
    3 root      34  19     0    0    0 S  0.0  0.0   0:00.99 ksoftirqd_CPU0
    4 root      15   0     0    0    0 S  0.0  0.0   0:33.63 kswapd
    5 root      15   0     0    0    0 S  0.0  0.0   0:00.71 bdflush
        [...]
 1362 root      15   0   488  452  404 S  0.0  0.1   0:00.02 nscd
 1363 root      15   0   488  452  404 S  0.0  0.1   0:00.04 nscd
 1377 root      17   0    56    4    4 S  0.0  0.0   0:00.00 mingetty
 1379 root       18   0    56    4    4 S  0.0  0.0   0:00.01 mingetty
 1380 root       18   0    56    4    4 S  0.0  0.0   0:00.01 mingetty
```

If you press F while `top` is running, a menu opens with which to make extensive changes to the format of the output.

The parameter `-U UID` monitors only the processes associated with a particular user. Replace *`UID`* with the user ID of the user. `top -U $(id -u username)` returns the UID of the user on the basis of the username and displays his processes.

25.7 Process List: ps

The command `ps` produces a list of processes. If the parameter `r` is added, only processes currently using computing time are shown:

```
$ ps r
  PID TTY      STAT   TIME COMMAND
22163 pts/7    R      0:01 -zsh
 3396 pts/3    R      0:03 emacs new-makedoc.txt
20027 pts/7    R      0:25 emacs xml/common/utilities.xml
20974 pts/7    R      0:01 emacs jj.xml
27454 pts/7    R      0:00 ps r
```

This parameter must be written without a minus sign. The various parameters are written sometimes with and sometimes without the minus sign. The man page could easily frighten off potential users, but fortunately the `ps --help` command produces a brief page of help.

To check how many `emacs` processes are running, use:

```
$ ps x | grep emacs
 1288 ?        S      0:07 emacs
 3396 pts/3    S      0:04 emacs new-makedoc.txt
 3475 ?        S      0:03 emacs .Xresources
20027 pts/7    S      0:40 emacs xml/common/utilities.xml
20974 pts/7    S      0:02 emacs jj.xml

$ pidof emacs
20974 20027 3475 3396 1288
```

The parameter `-p` selects processes via the process ID:

```
$ ps www -p $(pidof xterm)
  PID TTY      STAT   TIME COMMAND
 9025 ?        S      0:01 xterm -g 100x45+0+200
 9176 ?        S      0:00 xterm -g 100x45+0+200
```

```
29854 ?        S      0:21 xterm -g 100x75+20+0 -fn \
  -B&H-LucidaTypewriter-Medium-R-Normal-Sans-12-120-75-75-M-70-iso10646-1
 4378 ?        S      0:01 xterm -bg MistyRose1 -T root -n root -e su -l
25543 ?        S      0:02 xterm -g 100x45+0+200
22161 ?        R      0:14 xterm -g 100x45+0+200
16832 ?        S      0:01 xterm -bg MistyRose1 -T root -n root -e su -l
16912 ?        S      0:00 xterm -g 100x45+0+200
17861 ?        S      0:00 xterm -bg DarkSeaGreen1 -g 120x45+40+300
19930 ?        S      0:13 xterm -bg LightCyan
21686 ?        S      0:04 xterm -g 100x45+0+200 -fn \
lucidasanstypewriter-12
23104 ?        S      0:00 xterm -g 100x45+0+200
26547 ?        S      0:00 xterm -g 100x45+0+200
```

The process list can be formatted according to your needs. The option `-L` returns a list of all keywords. Enter the following command to issue a list of all processes sorted by memory usage:

```
$ ps ax --format pid,rss,cmd --sort rss
  PID  RSS CMD
    2    0 [ksoftirqd/0]
    3    0 [events/0]
   17    0 [kblockd/0]
[...]
10164 5260 xterm
31110 5300 xterm
17010 5356 xterm
 3896 29292 /usr/X11R6/bin/X -nolisten tcp -br vt7 -auth
/var/lib/xdm/authdir/au
```

25.8 Process Tree: pstree

The command `pstree` produces a list of processes in the form of a tree:

```
$ pstree
init-+-atd
     |-3*[automount]
     |-bdflush
     |-cron
  [...]
     |-usb-storage-1
     |-usb-storage-2
     |-10*[xterm---zsh]
     |-xterm---zsh---mutt
     |-2*[xterm---su---zsh]
     |-xterm---zsh---ssh
     |-xterm---zsh---pstree
     |-ypbind---ypbind---2*[ypbind]
     `-zsh---startx---xinit4-+-X
```

```
                             `-ctwm-+-xclock
                                    |-xload
                                    `-xosview.bin
```

The parameter -p adds the process ID to a given name. To have the command lines displayed as well, use the -a parameter:

```
$ pstree -pa
init,1
  |-atd,1255
[...]
  `-zsh,1404
      `-startx,1407 /usr/X11R6/bin/startx
          `-xinit4,1419 /suse/jj/.xinitrc [...]
              |-X,1426 :0 -auth /suse/jj/.Xauthority
              `-ctwm,1440
                  |-xclock,1449 -d -geometry -0+0 -bg grey
                  |-xload,1450 -scale 2
                  `-xosview.bin,1451 +net -bat +net
```

25.9 Who Is Doing What: w

With the command w, find out who is logged onto the system and what each user is doing. For example:

```
$ w
 15:17:26 up 62 days,  4:33, 14 users,  load average: 0.00, 0.04, 0.01
USER     TTY        LOGIN@   IDLE   JCPU   PCPU WHAT
jj       pts/0     30Mar04  4days  0.50s  0.54s xterm -e su -l
jj       pts/1     23Mar04  5days  0.20s  0.20s -zsh
jj       pts/2     23Mar04  5days  1.28s  1.28s -zsh
jj       pts/3     23Mar04  3:28m  3.21s  0.50s -zsh
[...]
jj       pts/7     07Apr04  0.00s  9.02s  0.01s w
jj       pts/9     25Mar04  3:24m  7.70s  7.38s mutt
[...]
jj       pts/14    12:49   37:34   0.20s  0.13s ssh totan
```

The last line shows that the user jj has established a secure shell (ssh) connection to the computer totan. If any users of other systems have logged in remotely, the parameter -f shows the computers from which they have established the connection.

25.10 Memory Usage: free

The utility `free` examines RAM usage. Details of both free and used memory (and swap areas) are shown:

```
$ free
             total       used       free     shared    buffers     cached
Mem:        514736     273964     240772          0      35920      42328
-/+ buffers/cache:     195716     319020
Swap:      1794736     104096    1690640
```

With `−m`, all sizes are expressed in megabytes:

```
$ free -m
             total       used       free     shared    buffers     cached
Mem:           502        267        235          0         35         41
-/+ buffers/cache:        191        311
Swap:         1752        101       1651
```

The really interesting information is contained in the following line:

```
-/+ buffers/cache:        191        311
```

This calculates the amount of memory taken up with buffers and caches. The parameter `−d delay` ensures that the display is refreshed every *`delay`* seconds. For example, `free −d 1.5` produces an update every 1.5 seconds.

25.11 Kernel Ring Buffer: dmesg

The Linux kernel keeps certain messages in a ring buffer. To view these messages, enter the command `dmesg`:

```
$ dmesg
[...]
sdc : READ CAPACITY failed.
sdc : status = 1, message = 00, host = 0, driver = 08
Info fld=0xa00 (nonstd), Current sd00:00: sense key Not Ready
sdc : block size assumed to be 512 bytes, disk size 1GB.
sdc: test WP failed, assume Write Enabled
 sdc: I/O error: dev 08:20, sector 0
 I/O error: dev 08:20, sector 0
 I/O error: dev 08:20, sector 2097144
 I/O error: dev 08:20, sector 2097144
 I/O error: dev 08:20, sector 0
 I/O error: dev 08:20, sector 0
 unable to read partition table
 I/O error: dev 08:20, sector 0
nfs: server totan not responding, still trying
nfs: server totan OK
```

The last line indicates that there is a temporary problem in the NFS server `totan`. The lines up to that point are triggered by the insertion of a USB flash drive. Older events are logged in the files `/var/log/messages` and `/var/log/warn`.

25.12 File Systems and Their Usage: mount, df, and du

The command `mount` shows which file system (device and type) is mounted at which mount point:

```
$ mount
/dev/hdb2 on / type ext2 (rw)
proc on /proc type proc (rw)
devpts on /dev/pts type devpts (rw,mode=0620,gid=5)
/dev/hda1 on /data type ext2 (rw)
shmfs on /dev/shm type shm (rw)
usbdevfs on /proc/bus/usb type usbdevfs (rw)
automount(pid1012) on /suse type autofs \
    (rw,fd=5,pgrp=1012,minproto=2,maxproto=3)
totan:/real-home/jj on /suse/jj type nfs \
    (rw,nosuid,rsize=8192,wsize=8192,hard,intr,nolock,addr=10.10.0.1)
```

Obtain information about total usage of the file systems with the command `df`. The parameter `-h` (or `--human-readable`) transforms the output into a form understandable for common users.

```
$ df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/hdb2             7.4G  5.1G  2.0G  73% /
/dev/hda1              74G  5.8G   65G   9% /data
shmfs                 252M     0  252M   0% /dev/shm
totan:/real-home/jj   350G  324G   27G  93% /suse/jj
```

Users of the NFS file server `totan` should clear their home directory without delay. Display the total size of all the files in a given directory and its subdirectories with the command `du`. The parameter `-s` suppresses the output of detailed information. `-h` again transforms the data into a form that ordinary people can understand. With this command:

```
$ du -sh ~
361M    /suse/jj
```

see how much space your own home directory occupies.

25.13 The /proc File System

The `/proc` file system is a pseudo file system in which the kernel reserves important information in the form of virtual files. For example, display the CPU type with this command:

```
$ cat /proc/cpuinfo
processor       : 0
vendor_id       : AuthenticAMD
cpu family      : 6
model           : 8
model name      : AMD Athlon(tm) XP 2400+
stepping        : 1
cpu MHz         : 2009.343
cache size      : 256 KB
fdiv_bug        : no
[...]
```

The allocation and use of interrupts can be queried with the following command:

```
$ cat /proc/interrupts
           CPU0
  0:  537544462          XT-PIC  timer
  1:     820082          XT-PIC  keyboard
  2:          0          XT-PIC  cascade
  8:          2          XT-PIC  rtc
  9:          0          XT-PIC  acpi
 10:      13970          XT-PIC  usb-uhci, usb-uhci
 11:  146467509          XT-PIC  ehci_hcd, usb-uhci, eth0
 12:    8061393          XT-PIC  PS/2 Mouse
 14:    2465743          XT-PIC  ide0
 15:       1355          XT-PIC  ide1
NMI:          0
LOC:          0
ERR:          0
MIS:          0
```

Some of the important files and their contents are:

`/proc/devices`
available devices

/proc/modules
kernel modules loaded

/proc/cmdline
kernel command line

/proc/meminfo
detailed information about memory usage

/proc/config.gz
gzip-compressed configuration file of the kernel currently running

Further information is available in the text file /usr/src/linux/Documentation/filesystems/proc.txt. Information about processes currently running can be found in the /proc/*NNN* directories, where *NNN* is the process ID (PID) of the relevant process. Every process can find its own characteristics in /proc/self/:

```
$ ls -l /proc/self
lrwxrwxrwx  1 root root 64 Apr 29 13:52 /proc/self -> 27585

$ ls -l /proc/self/
total 0
dr-xr-xr-x  2 jj suse 0 Apr 29 13:52 attr
-r--------  1 jj suse 0 Apr 29 13:52 auxv
-r--r--r--  1 jj suse 0 Apr 29 13:52 cmdline
lrwxrwxrwx  1 jj suse 0 Apr 29 13:52 cwd -> /suse/jj/t
-r--r--r--  1 jj suse 0 Apr 29 13:52 delay
-r--------  1 jj suse 0 Apr 29 13:52 environ
lrwxrwxrwx  1 jj suse 0 Apr 29 13:52 exe -> /bin/ls
dr-x------  2 jj suse 0 Apr 29 13:52 fd
-rw-------  1 jj suse 0 Apr 29 13:52 mapped_base
-r--r--r--  1 jj suse 0 Apr 29 13:52 maps
-rw-------  1 jj suse 0 Apr 29 13:52 mem
-r--r--r--  1 jj suse 0 Apr 29 13:52 mounts
lrwxrwxrwx  1 jj suse 0 Apr 29 13:52 root -> /
-r--r--r--  1 jj suse 0 Apr 29 13:52 stat
-r--r--r--  1 jj suse 0 Apr 29 13:52 statm
-r--r--r--  1 jj suse 0 Apr 29 13:52 status
dr-xr-xr-x  3 jj suse 0 Apr 29 13:52 task
-r--r--r--  1 jj suse 0 Apr 29 13:52 wchan
```

The address assignment of executables and libraries is contained in the maps file:

```
$ cat /proc/self/maps
08048000-0804c000 r-xp 00000000 03:02 22890      /bin/cat
0804c000-0804d000 rw-p 00003000 03:02 22890      /bin/cat
0804d000-0806e000 rwxp 0804d000 00:00 0
```

```
40000000-40016000 r-xp 00000000 03:02 10882      /lib/ld-2.3.3.so
40016000-40017000 rw-p 00015000 03:02 10882      /lib/ld-2.3.3.so
40017000-40018000 rw-p 40017000 00:00 0
4002b000-40135000 r-xp 00000000 03:02 10908      /lib/tls/libc.so.6
40135000-4013d000 rw-p 0010a000 03:02 10908      /lib/tls/libc.so.6
4013d000-40141000 rw-p 4013d000 00:00 0
bfffe000-c0000000 rw-p bfffe000 00:00 0
ffffe000-fffff000 ---p 00000000 00:00 0
```

25.14 vmstat, iostat, and mpstat

The utility vmstat reports virtual memory statistics. It reads the files /proc/meminfo, /proc/stat, and /proc/*/stat. It is useful to identify bottlenecks of the system performance. The command iostat reports statistics about the CPU and input and output for devices and partitions. The displayed information is taken from the files /proc/stat and /proc/partitions. The output can be used to better balance the input and output load between hard disks. The command mpstat reports CPU-related statistics.

25.15 procinfo

Important information from the /proc file system is summarized by the command procinfo:

```
$ procinfo
Linux 2.6.4-54.5-default (geeko@buildhost) (gcc 3.3.3 ) #1 1CPU [roth.suse.de]


Memory:      Total        Used        Free      Shared     Buffers
Mem:        516696      513200        3496           0       43284
Swap:       530136        1352      528784

Bootup: Wed Jul  7 14:29:08 2004    Load average: 0.07 0.04 0.01 1/126 5302

user  :       2:42:28.08   1.3%  page in :        0
nice  :       0:31:57.13   0.2%  page out:        0
system:       0:38:32.23   0.3%  swap in :        0
idle  :   3d 19:26:05.93  97.7%  swap out:        0
uptime:   4d  0:22:25.84         context :207939498

irq  0: 776561217 timer                 irq  8:         2 rtc

irq  1:    276048 i8042                 irq  9:     24300 VIA8233
irq  2:         0 cascade [4]          irq 11:  38610118 acpi, eth0, uhci_hcd
irq  3:          3                       irq 12:   3435071 i8042
irq  4:          3                       irq 14:   2236471 ide0
irq  6:          2                       irq 15:       251 ide1
```

To see all the information, use the parameter `-a`. The parameter `-nN` produces updates of the information every *N* seconds. In this case, terminate the program by pressing `Q`.

By default, the cumulative values are displayed. The parameter `-d` produces the differential values. `procinfo -dn5` displays the values that have changed in the last five seconds:

```
Memory:      Total        Used        Free      Shared     Buffers      Cached
Mem:             0           2          -2           0           0           0
Swap:            0           0           0

Bootup: Wed Feb 25 09:44:17 2004    Load average: 0.00 0.00 0.00 1/106 31902

user  :       0:00:00.02   0.4%  page in :        0  disk 1:        0r       0w
nice  :       0:00:00.00   0.0%  page out:        0  disk 2:        0r       0w
system:       0:00:00.00   0.0%  swap in :        0  disk 3:        0r       0w
idle  :       0:00:04.99  99.6%  swap out:        0  disk 4:        0r       0w
uptime:  64d  3:59:12.62         context :     1087

irq  0:       501 timer                irq 10:         0 usb-uhci, usb-uhci
irq  1:         1 keyboard             irq 11:        32 ehci_hcd, usb-uhci,
irq  2:         0 cascade [4]          irq 12:       132 PS/2 Mouse
irq  6:         0                      irq 14:         0 ide0
irq  8:         0 rtc                  irq 15:         0 ide1
irq  9:         0 acpi
```

25.16 PCI Resources: lspci

The command `lspci` lists the PCI resources:

```
$ lspci
00:00.0 Host bridge: VIA Technologies, Inc. \
    VT8366/A/7 [Apollo KT266/A/333]
00:01.0 PCI bridge: VIA Technologies, Inc. \
    VT8366/A/7 [Apollo KT266/A/333 AGP]
00:0b.0 Ethernet controller: Digital Equipment Corporation \
    DECchip 21140 [FasterNet] (rev 22)
00:10.0 USB Controller: VIA Technologies, Inc. USB (rev 80)
00:10.1 USB Controller: VIA Technologies, Inc. USB (rev 80)
00:10.2 USB Controller: VIA Technologies, Inc. USB (rev 80)
00:10.3 USB Controller: VIA Technologies, Inc. USB 2.0 (rev 82)
00:11.0 ISA bridge: VIA Technologies, Inc. VT8235 ISA Bridge
```

```
00:11.1 IDE interface: VIA Technologies, Inc. VT82C586/B/686A/B \
    PIPC Bus Master IDE (rev 06)
00:11.5 Multimedia audio controller: VIA Technologies, Inc. \
    VT8233 AC97 Audio Controller (rev 50)
01:00.0 VGA compatible controller: Matrox Graphics, Inc. \
    MGA G550 AGP (rev 01)
```

Using -v results in a more detailed listing:

```
$ lspci -v
[...]
01:00.0 \
 VGA compatible controller: Matrox Graphics, Inc. MGA G550 AGP (rev 01) \
    (prog-if 00 [VGA])
 Subsystem: Matrox Graphics, Inc. Millennium G550 Dual Head DDR 32Mb
 Flags: bus master, medium devsel, latency 32, IRQ 10
 Memory at d8000000 (32-bit, prefetchable) [size=32M]
 Memory at da000000 (32-bit, non-prefetchable) [size=16K]
 Memory at db000000 (32-bit, non-prefetchable) [size=8M]
 Expansion ROM at <unassigned> [disabled] [size=128K]
 Capabilities: <available only to root>
```

Information about device name resolution is obtained from file /usr/share/pci.ids. PCI IDs not listed in this file are marked “Unknown device”.

The parameter -vv produces all the information that could be queried by the program. To view the pure numeric values, you should use the parameter -n.

25.17 System Calls of a Program Run: strace

The utility strace enables you to trace all the system calls of a process currently running. Enter the command in the normal way, adding strace at the beginning of the line:

```
$ strace ls

execve("/bin/ls", ["ls"], [/* 88 vars */]) = 0
uname({sys="Linux", node="edison", ...}) = 0
brk(0)                                   = 0x805b000
old_mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) \
          = 0x40017000
open("/etc/ld.so.preload", O_RDONLY)     = -1 ENOENT (No such file or directory)
open("/etc/ld.so.cache", O_RDONLY)       = 3
fstat64(3, {st_mode=S_IFREG|0644, st_size=76333, ...}) = 0
old_mmap(NULL, 76333, PROT_READ, MAP_PRIVATE, 3, 0) = 0x40018000
[...]
```

```
ioctl(1, SNDCTL_TMR_TIMEBASE or TCGETS, {B38400 opost isig icanon echo ...}) = 0
ioctl(1, TIOCGWINSZ, {ws_row=53, ws_col=110, ws_xpixel=897, ws_ypixel=693}) = 0
open(".", O_RDONLY|O_NONBLOCK|O_LARGEFILE|O_DIRECTORY) = 3
fstat64(3, {st_mode=S_IFDIR|0755, st_size=144, ...}) = 0
fcntl64(3, F_SETFD, FD_CLOEXEC)         = 0
getdents64(3, /* 5 entries */, 4096)    = 160
getdents64(3, /* 0 entries */, 4096)    = 0
close(3)                                = 0
fstat64(1, {st_mode=S_IFCHR|0620, st_rdev=makedev(136, 48), ...}) = 0
mmap2(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) \
         = 0x40018000
write(1, "ltrace-ls.txt  myfile.txt  strac"..., 41) = 41
munmap(0x40018000, 4096)                = 0
exit_group(0)                           = ?
```

For example, to trace all attempts to open a particular file, use the following:

```
$ strace -e open ls myfile.txt

open("/etc/ld.so.preload", O_RDONLY)    = -1 ENOENT (No such file or directory)
open("/etc/ld.so.cache", O_RDONLY)      = 3
open("/lib/tls/librt.so.1", O_RDONLY)   = 3
open("/lib/libacl.so.1", O_RDONLY)      = 3
open("/lib/libselinux.so.1", O_RDONLY)  = 3
open("/lib/tls/libc.so.6", O_RDONLY)    = 3
open("/lib/tls/libpthread.so.0", O_RDONLY) = 3
open("/lib/libattr.so.1", O_RDONLY)     = 3
open("/proc/mounts", O_RDONLY)          = 3
[...]
open("/proc/filesystems", O_RDONLY)     = 3
open("/proc/self/attr/current", O_RDONLY) = 4
```

To trace all the child processes, use the parameter `-f`. The behavior and output format of strace can be largely controlled. For information, see `man strace`.

25.18 Library Calls of a Program Run: ltrace

The command `ltrace` enables you to trace the library calls of a process. This command is used in a similar fashion to `strace`. The parameter `-c` outputs the number and duration of the library calls that have occurred:

```
$ ltrace -c find /usr/share/doc
% time     seconds  usecs/call     calls    errors syscall
------ ----------- ----------- --------- --------- ----------------
 86.27    1.071814          30     35327           write
 10.15    0.126092          38      3297           getdents64
  2.33    0.028931           3     10208           lstat64
  0.55    0.006861           2      3122         1 chdir
  0.39    0.004890           3      1567         2 open
[...]
  0.00    0.000003           3         1           uname
  0.00    0.000001           1         1           time
------ ----------- ----------- --------- --------- ----------------
100.00    1.242403                 58269         3 total
```

25.19 Specifying the Required Library: ldd

The command `ldd` can be used to find out which libraries would load the dynamic executable specified as argument:

```
$ ldd /bin/ls
linux-gate.so.1 =>  (0xffffe000)
librt.so.1 => /lib/tls/librt.so.1 (0x4002b000)
libacl.so.1 => /lib/libacl.so.1 (0x40033000)
libselinux.so.1 => /lib/libselinux.so.1 (0x40039000)
libc.so.6 => /lib/tls/libc.so.6 (0x40048000)
libpthread.so.0 => /lib/tls/libpthread.so.0 (0x4015d000)
/lib/ld-linux.so.2 => /lib/ld-linux.so.2 (0x40000000)
libattr.so.1 => /lib/libattr.so.1 (0x4016d000)
```

Static binaries do not need any dynamic libraries:

```
$ ldd /bin/sash
        not a dynamic executable
$ file /bin/sash
/bin/sash: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), \
for GNU/Linux 2.2.5, statically linked, stripped
```

25.20 Additional Information about ELF Binaries

The content of binaries can be read with the `readelf` utility. This even works with ELF files that were built for other hardware architectures:

```
$ readelf --file-header /bin/ls
ELF Header:
  Magic:   7f 45 4c 46 01 01 01 00 00 00 00 00 00 00 00 00
  Class:                             ELF32
  Data:                              2's complement, little endian
  Version:                           1 (current)
  OS/ABI:                            UNIX - System V
  ABI Version:                       0
  Type:                              EXEC (Executable file)
  Machine:                           Intel 80386
  Version:                           0x1
  Entry point address:               0x8049b40
  Start of program headers:          52 (bytes into file)
  Start of section headers:          76192 (bytes into file)
  Flags:                             0x0
  Size of this header:               52 (bytes)
  Size of program headers:           32 (bytes)
  Number of program headers:         9
  Size of section headers:           40 (bytes)
  Number of section headers:         29
  Section header string table index: 26
```

25.21 Interprocess Communication: ipcs

The command `ipcs` produces a list of the IPC resources currently in use:

```
$ ipcs
------ Shared Memory Segments --------
key        shmid      owner      perms      bytes      nattch     status
0x000027d9 5734403    toms      660        64528      2
0x00000000 5767172    toms      666        37044      2
0x00000000 5799941    toms      666        37044      2

------ Semaphore Arrays --------
key        semid      owner      perms      nsems
0x000027d9 0          toms      660        1

------ Message Queues --------
key        msqid      owner      perms      used-bytes   messages
```

25.22 Time Measurement with time

The time spent by commands can be determined with the `time` utility. This utility is available in two versions: as a shell built-in and as a program (`/usr/bin/time`).

```
$ time find . > /dev/null

real    0m4.051s
user    0m0.042s
sys     0m0.205s
```

Part VIII System

32-Bit and 64-Bit Applications in a 64-Bit System Environment 26

SUSE Linux is available for several 64-bit platforms. This does not necessarily mean that all the applications included have already been ported to 64-bit platforms. SUSE Linux supports the use of 32-bit applications in a 64-bit system environment. This chapter offers a brief overview of how this support is implemented on 64-bit SUSE Linux platforms. It explains how 32-bit applications are executed (runtime support) and how 32-bit applications should be compiled to enable them to run both in 32-bit and 64-bit system environments. Additionally, find information about the Kernel API and an explanation of how 32-bit applications can run under a 64-bit kernel.

SUSE Linux for the 64-bit platforms AMD64 and EM64T is designed so that existing 32-bit applications run in the 64-bit environment "out-of-the-box." This support means that you can continue to use your preferred 32-bit applications without waiting for a corresponding 64-bit port to become available.

26.1 Runtime Support

IMPORTANT: Conflicts between Application Versions

If an application is available both for 32-bit and 64-bit environments, the parallel installation of both versions is bound to lead to problems. In such cases, decide on one of the two versions and install and use this.

To be executed correctly, every application requires a range of libraries. Unfortunately, the names for the 32-bit and 64-bit versions of these libraries are identical. They must be differentiated from each other in another way.

To retain compatibility with the 32-bit version, the libraries are stored at the same place in the system as in the 32-bit environment. The 32-bit version of `libc.so.6` is located under `/lib/libc.so.6` in both the 32-bit and 64-bit environments.

All 64-bit libraries and object files are located in directories called `lib64`. The 64-bit object files you would normally expect to find under `/lib`, `/usr/lib`, and `/usr/X11R6/lib` are now found under `/lib64`, `/usr/lib64`, and `/usr/X11R6/lib64`. This means that there is space for the 32-bit libraries under `/lib`, `/usr/lib` and `/usr/X11R6/lib`, so the filename for both versions can remain unchanged.

No subdirectories of the object directories whose data content does not depend on the word size are moved. For example, the X11 fonts are still found in the usual location under `/usr/X11R6/lib/X11/fonts`. This scheme conforms to the LSB (Linux Standards Base) and the FHS (File System Hierarchy Standard).

26.2 Software Development

A biarch development toolchain allows generatation of 32-bit and 64-bit objects. The default is to compile 64-bit objects. It is possible to generate 32-bit objects by using special flags. For GCC, this special flag is `-m32`.

All header files must be written in an architecture-independent form. The installed 32-bit and 64-bit libraries must have an API (application programming interface) that matches the installed header files. The normal SUSE environment is designed according to this principle. In the case of manually updated libraries, resolve these issues yourself.

26.3 Software Compilation on Biarch Platforms

To develop binaries for the other architecture on a biarch architecture, the respective libraries for the second architecture must additionally be installed. These packages are called `rpmname-32bit` . You also need the respective headers and libraries from the `rpmname-devel` packages and the development libraries for the second architecture from `rpmname-devel-32bit`.

Most Open Source programs use an `autoconf`-based program configuration. To use `autoconf` for configuring a program for the second architecture, overwrite the normal compiler and linker settings of `autoconf` by running the `configure` script with additional environment variables.

The following example refers to an AMD64 or EM64T system with x86 as the second architecture:

1. Set `autoconf` to use the 32-bit compiler:

   ```
   CC="gcc -m32"
   ```

2. Instruct the linker to process 32-bit objects:

   ```
   LD="ld -m elf64_i386"
   ```

3. Set the assembler to generate 32-bit objects:

   ```
   AS="gcc -c -m32"
   ```

4. Determine that the libraries for `libtool` and so on come from `/usr/lib`:

   ```
   LDFLAGS="-L/usr/lib"
   ```

5. Determine that the libraries are stored in the `lib` subdirectory:

   ```
   --libdir=/usr/lib
   ```

6. Determine that the 32-bit X libraries are used:

   ```
   --x-libraries=/usr/X11R6/lib/
   ```

Not all of these variables are needed for every program. Adapt them to the respective program.

```
CC="gcc -m64"            \
LDFLAGS="-L/usr/lib64;"  \
        .configure       \
          --prefix=/usr  \
          --libdir=/usr/lib64
make
make install
```

26.4 Kernel Specifications

The 64-bit kernels for AMD64 and EM64T offer both a 64-bit and a 32-bit kernel ABI (application binary interface). The latter is identical with the ABI for the corresponding 32-bit kernel. This means that the 32-bit application can communicate with the 64-bit kernel in the same way as with the 32-bit kernel.

The 32-bit emulation of system calls for a 64-bit kernel does not support a number of APIs used by system programs. This depends on the platform. For this reason, a small number of applications, like `lspci` or the LVM administration programs, must be compiled as 64-bit programs to function properly.

A 64-bit kernel can only load 64-bit kernel modules that have been specially compiled for this kernel. It is not possible to use 32-bit kernel modules.

TIP

Some applications require separate kernel-loadable modules. If you intend to use such a 32-bit application in a 64-bit system environment, contact the provider of this application and SUSE to make sure that the 64-bit version of the kernel-loadable module and the 32-bit compiled version of the kernel API are available for this module.

Working with the Shell 27

Graphical user interfaces are becoming increasingly important for Linux, but using the mouse is not always the best way to perform daily tasks. The command line provides high flexibility and efficiency. Text-based applications are especially important for controlling computers over slow network links or if you want to perform tasks as `root` on the command line in an xterm. The Bash shell is the default command line interpreter on SUSE Linux.

Linux is a multiuser system and access to files is controlled by user permissions. Whether using the command line or a GUI, it is useful to understand the permission concept. When using the command line, a number of commands are important. The vi text editor is often used when configuring a system from the command line. It is also popular with many system administrators and developers.

27.1 Using of Bash on the Command Line

In the KDE taskbar, there is an icon depicting a monitor with a seashell. When you click this icon, a terminal window opens in which to enter commands. Konsole, the terminal program, normally runs Bash (Bourne again shell), a program developed as part of the GNU project. On the GNOME desktop, click an icon with a computer monitor in the upper panel to start a terminal that normally runs Bash.

Once you have opened the shell, see the prompt on the first line. The prompt usually consists of the username, hostname, and current path, but it can be customized. When the cursor is after this prompt, you can send commands directly to your computer system.

27.1.1 Entering Commands

A command consists of several elements. The first element is always the actual command, followed by parameters or options. Commands are executed when you press Enter. Before doing so, easily edit the command line, add options, or correct typing errors. One of the most frequently used commands is `ls`, which can be used with or without arguments. Entering the plain `ls` command shows the contents of the current directory.

Options are prefixed with a hyphen. The command `ls -l`, for example, shows the contents of the same directory in full detail (long listing format). Next to each filename, see the date when the file was created, the file size in bytes, and further details, which are covered later. One important option that exists for many commands is `--help`. By entering `ls --help`, display all the options for the `ls` command.

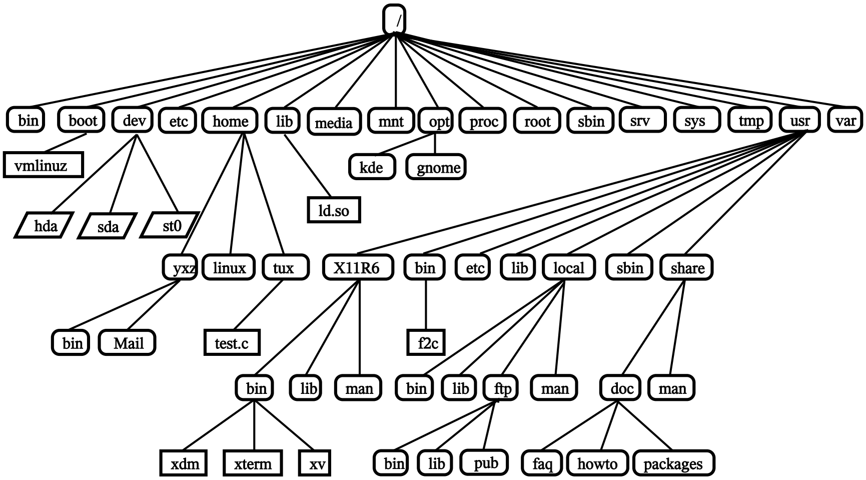
It is important to get the “quoting” right. If a filename contains a space, either escape the space using a back slash (`\`) or enclose the filename in single or double quotes. Otherwise Bash interprets a filename like `My Documents` as the names of two files or directories. The difference between single and double quotes is that variable expansion takes place within double quotes. Single quotes ensure that the shell sees the quoted string literally.

27.1.2 Files and Directories

To use the shell efficiently, it is really useful to have some knowledge of the file and directory structures of a Linux system. You can think of directories as electronic folders in which files, programs, and subdirectories are stored. The top level directory in the hierarchy is the root directory, referred to as `/`. This is the place from which all other directories can be accessed.

The `/home` directory contains the directories in which the individual users can store their personal files. Figure 27.1, “Excerpt from a Standard Directory Tree” (page 379) shows the standard directory tree in Linux, with the home directories of the example users `xyz`, `linux`, and `tux`. The directory tree of a Linux system has a functional structure that follows the *Filesystem Hierarchy Standard* (FHS). The following list provides a brief description of the standard directories in Linux.

Figure 27.1 *Excerpt from a Standard Directory Tree*



`/`
Root directory, starting point of the directory tree

`/home`
Personal directories of users

`/dev`
Device files that represent hardware components

`/etc`
Important files for system configuration

`/etc/init.d`
Boot scripts

`/usr/bin`
Generally accessible programs

`/bin`
Programs needed early in the boot process

/usr/sbin
Programs reserved for the system administrator

/sbin
Programs reserved for the system administrator and needed for booting

/usr/include
Header files for the C compiler

/usr/include/g++
Header files for the C++ compiler

/usr/share/doc
Various documentation files

/usr/share/man
System manual pages (man pages)

/usr/src
Source code of system software

/usr/src/linux
Kernel source code

/tmp, **/var/tmp**
Temporary files

/usr
All application programs

/var
Configuration files (such as those linked from `/usr`)

/var/log
System log files

/var/adm
System administration data

/lib
Shared libraries (for dynamically linked programs)

`/proc`
Process file system

`/sys`
System file system where all device information for the kernel is gathered

`/usr/local`
Local, distribution-independent extensions

`/opt`
Optional software, larger add-on program packages (such as KDE, GNOME, Netscape)

27.1.3 Bash Features

There are two important features of the shell that can make your work a lot easier:

History
To repeat a command that has been entered before, press ↑ until the previous command appears at the prompt. Move forward through the list of previously entered commands by pressing ↓. To edit the command line, just move the cursor to the desired position using the arrow keys and start typing. Use Ctrl + R to search in the history.

Completion
Complete a filename to its full length after typing its first letters until it can be uniquely identified. To do so, type the first letters then hit Tab. If there are several filenames starting with the same letters, obtain a list of them by hitting Tab twice.

First Example: Managing Files

Now that you know what a command looks like, which directories exist in SUSE Linux, and how to speed up things when using Bash, put this knowledge into practice with a small exercise.

1. Open a console from the KDE or GNOME desktop by clicking the shell icon.

2. Enter the `ls` command to see the contents of your home directory.

3. Use the command `mkdir` (which stands for *make directory*) to create a new subdirectory called `test` by entering `mkdir test`.

4. Now launch an editor by pressing Alt + F2 and entering `kate` Kate in for KDE `gedit` for Gedit in GNOME. Type a few letters in the editor then save the file as `Testfile` in your home directory. Linux distinguishes between uppercase and lowercase. For this example, use an uppercase T.

5. View the contents of your home directory again. Instead of typing `ls` again, just press ↑ twice and the `ls` command should reappear at the prompt. To execute the command, hit Enter. The newly created directory `test` should appear in blue letters and `Testfile` in black. This is how directories and files can be distinguished in a console.

6. Move `Testfile` into the subdirectory `test` with the command `mv`. To speed this up, use the expansion function: just enter `mv T` and press Tab. As long as there is no other file beginning with this letter in the directory, the shell expands the filename and adds the string *estfile*. Otherwise, add a letter or two yourself and test Tab each time to see whether the shell can now expand the name. Finally, type a space then `test` after the expanded filename and press Enter to execute the command.

7. At this point, `Testfile` should no longer be in the directory. Check this by entering `ls` again.

8. To see whether the file has been successfully moved, change into the directory `test` with the command `cd test`. Now enter `ls` again. You should see `Testfile` in the listing. Change back to your home directory at any point by entering only `cd`.

9. To make a copy of a file, use `cp`. For instance, enter `cp Testfile Testbackup` to copy `Testfile` to `Testbackup`. Once again, the command `ls` can be used to see whether both files are in the directory.

27.1.4 Specifying Paths

When working with files or directories, it is important specify the correct path. However, you do not need to enter the entire (absolute) path from the root directory to the respective file. You can start from the current directory. Address your home directory directly with ~. This means that there are two ways to list the file `Testfile` in the directory `test`: by entering the relative path with `ls test` or by specifying the absolute path with `ls ~/test`.

To list the contents of home directories of other users, enter `ls ~username`. In the example directory tree, one of the sample users is `tux`. In this case, `ls ~tux` would list the contents of the home directory of `tux`.

Refer to the current directory with a dot (`.`). The next higher level in the tree is represented by two dots (`..`). By entering `ls ..`, see the contents of the parent directory of the current directory. The command `ls ../..` shows the contents of the directory two levels higher in the hierarchy.

Second Example: Working with Paths

Here is another example to illustrate how to move around in the directories of your SUSE Linux system.

1. Change into your home directory with the command `cd`. Then create a directory in it with the name `test2` by entering `mkdir test2`.

2. Change into the new directory with `cd test2` and create a subdirectory in it with the name `subdirectory`. To change into it, use the expansion function: enter `cd su` then press Tab. The shell expands the rest of the directory name.

3. Now try to move the previously created file `Testbackup` into the current directory (`subdirectory`) without changing the directory again. To achieve this, specify the relative path to that file: `mv ../../test/Testbackup .` (note the dot at the end). The dot at the end of this command is required to tell the shell that the current directory is the destination to which to move the file. `../../`, in this example, refers to your home directory.

27.1.5 Wild Cards

Another convenience offered by the shell is wild cards for pathname expansion. There are three different types of these in Bash:

?

Matches exactly one arbitrary character

*****

Matches any number of characters

[set]

Matches one of the characters from the group specified inside the square brackets, which is represented here by the string *set*. As part of *set* you can also specify character classes using the syntax *[:`class`:]*, where a class is one of `alnum`, `alpha`, `ascii`, etc.

Using `!` or `^` at the beginning of the group (*[!set]*) matches one character other than those identified by *set*.

Assuming that your `test` directory contains the files `Testfile`, `Testfile1`, `Testfile2`, and `datafile`, the command `ls Testfile?` lists the files `Testfile1` and `Testfile2`. With `ls Test*`, the list also includes `Testfile`. `ls *fil*` shows all the sample files. Finally, you can use the `set` wild card to address all sample files whose last character is a number: `ls Testfile[1-9]` or, using classes, `ls Testfile[[:digit:]]`.

Of the four types of wild cards, the most inclusive one is the asterisk. It could be used to copy all files contained in one directory to another one or to delete all files with one command. The command `rm *fil*`, for instance, would delete all files in the current directory whose name includes the string *fil*.

27.1.6 Less and More

Linux includes two small programs for viewing text files directly in the shell. Rather than starting an editor to read a file like `Readme.txt`, simply enter `less Readme.txt` to display the text in the console window. Use Space to scroll down one page. Use Page Up and Page Down to move forward or backward in the text. To exit less, press Q.

Instead of less, you can also use the older program more. However, it is less convenient because it does not allow you to scroll backwards.

The program less got its name from the the precept that *less is more* and can also be used to view the output of commands in a convenient way. To see how this works, read Section 27.1.7, “Pipes and Redirection” (page 385).

27.1.7 Pipes and Redirection

Normally, the standard output in the shell is your screen or the console window and the standard input is the keyboard. To forward the output of a command to an application like less, use a *pipeline*.

To view the files in the `test` directory, enter the command `ls test | less`. The contents of the `test` directory are then displayed with less. This only makes sense if the normal output with `ls` would be too lengthy. For instance, if you view the contents of the `dev` directory with `ls /dev`, you only see a small portion in the window. View the entire list with `ls /dev | less`.

It is also possible to save the output of commands to a file. For example, `echo "test one" > Content` generates a new file called `Content` that contains the words `test one`. View the file with `less Content`.

You can also use a file as the input for a command. For example, with `tr` replace characters from standard input that redirected from the file `Content` and write the result to standard output: replace `t` with `x` by calling `tr t x < Content`. The output of `tr` is sent to the screen.

If you need a new file containing the output, pipe the output of `tr` to a file. To test this, change into `test` and enter the command `tr t x < ../Content > new`. Finally, view `new` with `less new`.

Just like the standard output, the standard error output is sent to the console. However, to redirect the standard error output to a file named `errors`, append `2> errors` to the corresponding command. Both standard output and standard error are saved to one file named `alloutput` if you append `>& alloutput`. Finally, to append the output of a command to an already existing file, the command must be followed by `>>` instead of `>`.

27.1.8 Archives and Data Compression

Now that you have already created a number of files and directories, consider the subject of archives and data compression. Suppose you want to have the entire `test` directory packed in one file that you can save on a USB stick as a backup copy or send by e-mail.

To do so, use the command `tar` (for *tape archiver*). With `tar --help`, view all the options for the `tar` command. The most important of these options are explained here:

-c
: (for create) Create a new archive.

-t
: (for table) Display the contents of an archive.

-x
: (for extract) Unpack the archive.

-v
: (for verbose) Show all files on screen while creating the archive.

-f
: (for file) Choose a filename for the archive file. When creating an archive, this option must always be given as the last one.

To pack the `test` directory with all its files and subdirectories into an archive named `testarchive.tar`, use the options `-c` and `-f`. For testing purposes, also add `-v` to follow the progress of the archiving, although this option is not mandatory. After using `cd` to change to your home directory where the `test` directory is located, enter `tar -cvf testarchive.tar test`. After that, view the contents of the archive file with `tar -tf testarchive.tar`. The `test` directory with all its files and directories has remained unchanged on your hard disk. To unpack the archive, enter `tar -xvf testarchive.tar`, but do not try this yet.

For file compression, the obvious choice is `gzip` or, for a even better compression ratio, `bzip2`. Just enter `gzip testarchive.tar` (or `bzip2 testarchive.tar`, but `gzip` is used in this example). With `ls`, now see that the file `testarchive.tar` is no longer there and that the file `testarchive.tar.gz` has been created instead. This file is much smaller and therefore much better suited for transfer via e-mail or storage on a USB stick.

Now, unpack this file in the `test2` directory created earlier. To do so, enter `cp testarchive.tar.gz test2` to copy the file to that directory. Change to the directory with `cd test2`. A compressed archive with the `.tar.gz` extension can be unzipped with the `gunzip` command. Enter `gunzip testarchive.tar.gz`, which results in the file `testarchive.tar`, which then needs to be extracted or *untarred* with `tar -xvf testarchive.tar`. You can also unzip and extract a compressed archive in one step with `tar -xvf testarchive.tar.gz` (adding the `-z` option is no longer required). With `ls`, you can see that a new `test` directory has been created with the same contents as your `test` directory in your home directory.

27.1.9 mtools

`mtools` are a set of commands for working with MS-DOS file systems. The commands included in `mtools` allow you to address the first floppy drive as `a:`, just like under MS-DOS, and the commands are like MS-DOS commands except they are prefixed with an `m`.

`mdir a:`
Displays the contents of the floppy disk in drive `a:`

`mcopy Testfile a:`
Copies the file `Testfile` to the floppy disk

`mdel a:Testfile`
Deletes `Testfile` in `a:`

`mformat a:`
Formats the floppy disk in MS-DOS format (using the `fdformat` command)

`mcd a:`
Makes `a:` your current directory

`mmd a:test`
Creates the subdirectory `test` on the floppy disk

`mrd a:test`
Deletes the subdirectory `test` from the floppy disk

27.1.10 Cleaning Up

After this crash course, you should be familiar with the basics of the Linux shell or command line. You may want to clean up your home directory by deleting the various test files and directories using the `rm` and `rmdir` commands. In Section 27.3, “Important

Linux Commands" (page 393), find a list of the most important commands and a brief description of their functions.

27.2 Users and Access Permissions

Since its inception in the early 1990s, Linux has been developed as a multiuser system. Any number of users can work on it simultaneously. Users need to log in to the system before starting a session at their workstations. Each user has a username with a corresponding password. This differentiation of users guarantees that unauthorized users cannot see files for which they do not have permission. Larger changes to the system, such as installing new programs, are also usually impossible or restricted for normal users. Only the root user, or *super user*, has the unrestricted capacity to make changes to the system and has unlimited access to all files. Those who use this concept wisely, only logging in with full `root` access when necessary, can cut back the risk of unintentional loss of data. Because under normal circumstances only root can delete system files or format hard disks, the threat from the *Trojan horse effect* or from accidentally entering destructive commands can be significantly reduced.

27.2.1 File System Permissions

Basically, every file in a Linux file system belongs to a user and a group. Both of these proprietary groups and all others can be authorized to write, read, or execute these files.

A group, in this case, can be defined as a set of connected users with certain collective rights. For example, call a group working on a certain project `project3`. Every user in a Linux system is a member of at least one proprietary group, normally `users`. There can be as many groups in a system as needed, but only `root` is able to add groups. Every user can find out, with the command `groups`, of which groups he is a member.

File Access

The organization of permissions in the file system differs for files and directories. File permission information can be displayed with the command `ls -l`. The output could appear as in Example 27.1, "Sample Output Showing File Permissions" (page 389).

Example 27.1 *Sample Output Showing File Permissions*

```
-rw-r----- 1 tux project3 14197 Jun 21  15:03 Roadmap
```

As shown in the third column, this file belongs to user `tux`. It is assigned to the group `project3`. To discover the user permissions of the `Roadmap` file, the first column must be examined more closely.

-	rw-	r--	---
Type	Users Permissions	Group Permissions	Permissions for Other Users

This column consists of one leading character followed by nine characters grouped in threes. The first of the ten letters stands for the type of file system component. The hyphen (`-`) shows that this is a file. A directory (`d`), a link (`l`), a block device (`b`), or a character device could also be indicated.

The next three blocks follow a standard pattern. The first three characters refer to whether the file is readable (`r`) or not (`-`). A `w` in the middle portion symbolizes that the corresponding object can be edited and a hyphen (`-`) means it is not possible to write to the file. An `x` in the third position denotes that the object can be executed. Because the file in this example is a text file and not one that is executable, executable access for this particular file is not needed.

In this example, `tux` has, as owner of the file `Roadmap`, read (`r`) and write access (`w`) to it, but cannot execute it (`x`). The members of the group `project3` can read the file, but they cannot modify it or execute it. Other users do not have any access to this file. Other permissions can be assigned by means of ACLs (access control lists). See Section 27.2.6, “Access Control Lists” (page 392) for background information.

Directory Permissions

Access permissions for directories have the type `d`. For directories, the individual permissions have a slightly different meaning.

Example 27.2 *Sample Output Showing Directory Permissions*

```
drwxrwxr-x 1 tux project3 35 Jun 21 15:15  ProjectData
```

In Example 27.2, “Sample Output Showing Directory Permissions” (page 389), the owner (tux) and the owning group (project3) of the directory ProjectData are easy to recognize. In contrast to the file access permissions from File Access (page 388), the set reading permission (r) means that the contents of the directory can be shown. The write permission (w) means that new files can be created. The executable permission (x) means that the user can change to this directory. In the above example, the user tux as well as the members of the group project3 can change to the ProjectData directory (x), view the contents (r), and add or delete files (w). The rest of the users, on the other hand, are given less access. They may enter the directory (x) and browse through it (r), but not insert any new files (w).

27.2.2 Modifying File Permissions

Changing Access Permissions

The access permissions of a file or directory can be changed by the owner and, of course, by root with the command chmod followed by the parameters changing the permissions and one or more filenames. The parameters form different categories:

1. users concerned

- u (*user*)—owner of the file

- g (*group*)—group that owns the file

- o (*others*)—additional users (if no parameter is given, the changes apply to all categories)

2. a character for deletion (–), setting (=), or insertion (+)

3. the abbreviations

- r—*read*

- w—*write*

- x—*execute*

4. filename or filenames separated by spaces

If, for example, the user `tux` in Example 27.2, “Sample Output Showing Directory Permissions” (page 389) also wants to grant other users write (`w`) access to the directory `ProjectData`, he can do this using the command `chmod o+w ProjectData`.

If, however, he wants to deny all users other than himself write permissions, he can do this by entering the command `chmod go-w ProjectData`. To prohibit all users from adding a new file to the folder `ProjectData`, enter `chmod -w ProjectData`. Now, not even the owner can write to the file without first reestablishing write permissions.

Changing Ownership Permissions

Other important commands to control the ownership and permissions of the file system components are `chown` (change owner) and `chgrp` (change group). The command `chown` can be used to transfer ownership of a file to another user. However, only `root` is permitted to perform this change.

Suppose the file `Roadmap` from Example 27.2, “Sample Output Showing Directory Permissions” (page 389) should no longer belong to `tux`, but to the user `geeko`. `root` should then enter `chown geeko Roadmap`.

`chgrp` changes the group ownership of the file. However, the owner of the file must be a member of the new group. In this way, the user `tux` from Example 27.1, “Sample Output Showing File Permissions” (page 389) can switch the group owning the file `ProjectData` to `project4` with the command `chgrp project4 ProjectData`, as long as he is a member of this new group.

27.2.3 The setuid Bit

In certain situations, the access permissions may be too restrictive. Therefore, Linux has additional settings that enable the temporary change of the current user and group identity for a specific action. For example, the `passwd` program normally requires root permissions to access `/etc/passwd`. This file contains some important information, like the home directories of users and user and group IDs. Thus, a normal user would not be able to change `passwd`, because it would be too dangerous to grant all users direct access to this file. A possible solution to this problem is the *setuid* mechanism. setuid (set user ID) is a special file attribute that instructs the system to execute programs marked accordingly under a specific user ID. Consider the `passwd` command:

```
-rwsr-xr-x  1 root shadow 80036 2004-10-02 11:08 /usr/bin/passwd
```

You can see the `s` that denotes that the setuid bit is set for the user permission. By means of the setuid bit, all users starting the `passwd` command execute it as `root`.

27.2.4 The setgid Bit

The setuid bit applies to users. However, there is also an equivalent property for groups: the *setgid* bit. A program for which this bit was set runs under the group ID under which it was saved, no matter which user starts it. Therefore, in a directory with the setgid bit, all newly created files and subdirectories are assigned to the group to which the directory belongs. Consider the following example directory:

```
drwxrws--- 2 tux archive 48 Nov 19 17:12  backup
```

You can see the `s` that denotes that the setgid bit is set for the group permission. The owner of the directory and members of the group `archive` may access this directory. Users that are not members of this group are “mapped” to the respective group. The effective group ID of all written files will be `archive`. For example, a backup program that runs with the group ID `archive` is able to access this directory even without root privileges.

27.2.5 The Sticky Bit

There is also the *sticky bit*. It makes a difference whether it belongs to an executable program or a directory. If it belongs to a program, a file marked in this way is loaded to RAM to avoid needing to get it from the hard disk each time it is used. This attribute is used rarely, because modern hard disks are fast enough. If this bit is assigned to a directory, it prevents users from deleting each other's files. Typical examples include the `/tmp` and `/var/tmp` directories:

```
drwxrwxrwt   2 root  root   1160 2002-11-19 17:15 /tmp
```

27.2.6 Access Control Lists

The traditional permission concept for Linux file system objects, such as files or directories, can be expanded by means of ACLs (access control lists). They allow the assignment of permissions to individual users or groups other than the original owner or owning group of a file system object.

Files or directories bearing extended access permissions can be detected with a simple `ls -l` command:

```
-rw-r--r--+ 1 tux project3 14197 Jun 21  15:03 Roadmap
```

`Roadmap` is owned by `tux` who belongs to the group `project3`. `tux` holds both write and read access to this file. The group as well as all other users have read access. The only difference that distinguishes this file from a file without an ACL is the additional + in the column holding the permission bits.

Get details about the ACL by executing `getfacl Roadmap`:

```
# file: Roadmap
# owner: tux
# group: project3
user::rw-
user:jane:rw-       effective: r--
group::r--
group:djungle:rw-   effective: r--
mask::r--
other::---
```

The first three lines of the output do not hold any information not available with `ls -l`. These lines only state filename, owner, and owning group. Lines 4 to 9 hold the ACL entries. Conventional access permissions represent a subset of those possible when using ACLs. The example ACL grants read and write access to the owner of the file as well as to user `jane` (lines 4 and 5). The conventional concept has been expanded allowing access to an extra user. The same applies to the handling of group access. The owning group holds read permissions (line 6) and the group `djungle` holds read and write permissions. The `mask` entry in line 8 reduces the effective permissions for the user `jane` and the group `djungle` to read access. Other users and groups do not get any kind of access to the file (line 9).

Only very basic information has been provided here. Find more detailed information about ACLs in Chapter 24, *Access Control Lists in Linux* (page 341).

27.3 Important Linux Commands

This section gives insight into the most important commands of your SUSE Linux system. There are many more commands than listed in this chapter. Along with the individual commands, parameters are listed and, where appropriate, a typical sample application is introduced. To learn more about the various commands, use the manual pages, accessed with `man` followed by the name of the command, for example, `man ls`.

In the man pages, move up and down with PgUp and PgDn. Move between the beginning and the end of a document with Home and End. End this viewing mode by pressing Q. Learn more about the `man` command itself with `man man`.

In the following overview, the individual command elements are written in different typefaces. The actual command and its mandatory options are always printed as `command option`. Specifications or parameters that are not required are placed in `[square brackets]`.

Adjust the settings to your needs. It makes no sense to write `ls file`, if no file named `file` actually exists. You can usually combine several parameters, for example, by writing `ls -la` instead of `ls -l -a`.

27.3.1 File Commands

The following section lists the most important commands for file management. It covers anything from general file administration to manipulation of file system ACLs.

File Administration

`ls [options] [files]`

If you run `ls` without any additional parameters, the program lists the contents of the current directory in short form.

`-l`
Detailed list

`-a`
Displays hidden files

`cp [options] source target`

Copies `source` to `target`.

-i

Waits for confirmation, if necessary, before an existing `target` is overwritten

-r

Copies recursively (includes subdirectories)

`mv [options] source target`

Copies `source` to `target` then deletes the original `source`.

-b

Creates a backup copy of the `source` before moving

-i

Waits for confirmation, if necessary, before an existing `targetfile` is overwritten

`rm [options] files`

Removes the specified files from the file system. Directories are not removed by `rm` unless the option `-r` is used.

`-r`

Deletes any existing subdirectories

`-i`

Waits for confirmation before deleting each file.

`ln [options] source target`

Creates an internal link from `source` to `target`. Normally, such a link points directly to `source` on the same file system. However, if `ln` is executed with the `-s` option, it creates a symbolic link that only points to the directory in which `source` is located, enabling linking across file systems.

-s

Creates a symbolic link

cd [options] [directory]

Changes the current directory. `cd` without any parameters changes to the user's home directory.

mkdir [options] directory

Creates a new directory.

rmdir [options] directory

Deletes the specified directory, if it is already empty.

chown [options] username[:[group]] files

Transfers ownership of a file to the user with the specified username.

-R
Changes files and directories in all subdirectories

chgrp [options] groupname files

Transfers the group ownership of a given `file` to the group with the specified group name. The file owner can only change group ownership if a member of both the current and the new group.

chmod [options] mode files

Changes the access permissions.

The `mode` parameter has three parts: `group`, `access`, and `access type`. `group` accepts the following characters:

u
user

g
group

o
others

For `access`, grant access with + and deny it with `–`.

The `access type` is controlled by the following options:

r
: read

w
: write

x
: execute—executing files or changing to the directory

s
: Setuid bit—the application or program is started as if it were started by the owner of the file

As an alternative, a numeric code can be used. The four digits of this code are composed of the sum of the values 4, 2, and 1—the decimal result of a binary mask. The first digit sets the set user ID (SUID) (4), the set group ID (2), and the sticky (1) bits. The second digit defines the permissions of the owner of the file. The third digit defines the permissions of the group members and the last digit sets the permissions for all other users. The read permission is set with 4, the write permission with 2, and the permission for executing a file is set with 1. The owner of a file would usually receive a 6 or a 7 for executable files.

`gzip [parameters] files`

This program compresses the contents of files using complex mathematical algorithms. Files compressed in this way are given the extension `.gz` and need to be uncompressed before they can be used. To compress several files or even entire directories, use the `tar` command.

-d
: Decompresses the packed gzip files so they return to their original size and can be processed normally (like the command `gunzip`)

`tar options archive files`

`tar` puts one or more files into an archive. Compression is optional. `tar` is a quite complex command with a number of options available. The most frequently used options are:

`-f`
: Writes the output to a file and not to the screen as is usually the case

`-c`
: Creates a new tar archive

`-r`
: Adds files to an existing archive

`-t`
: Outputs the contents of an archive

`-u`
: Adds files, but only if they are newer than the files already contained in the archive

`-x`
: Unpacks files from an archive (*extraction*)

`-z`
: Packs the resulting archive with `gzip`

`-j`
: Compresses the resulting archive with `bzip2`

`-v`
: Lists files processed

The archive files created by `tar` end with `.tar`. If the tar archive was also compressed using `gzip`, the ending is `.tgz` or `.tar.gz`. If it was compressed using `bzip2`, the ending is `.tar.bz2`. Application examples can be found in Section 27.1.8, “Archives and Data Compression” (page 385).

`locate patterns`

This command is only available if you have installed the `findutils-locate` package. The `locate` command can find in which directory a specified file is located. If desired, use wild cards to specify filenames. The program is very speedy, because it uses a database specifically created for the purpose (rather than searching through the entire file system). This very fact, however, also results in a major drawback: locate is unable to find any files created after the latest update of its database. The database can be generated by `root` with `updatedb`.

`updatedb [options]`

This command performs an update of the database used by `locate`. To include files in all existing directories, run the program as `root`. It also makes sense to place it in the background by appending an ampersand (`&`), so you can immediately continue working on the same command line (`updatedb &`). This command usually runs as a daily cron job (see `cron.daily`).

`find [options]`

With `find`, search for a file in a given directory. The first argument specifies the directory in which to start the search. The option `-name` must be followed by a search string, which may also include wild cards. Unlike `locate`, which uses a database, `find` scans the actual directory.

Commands to Access File Contents

`cat [options] files`

The `cat` command displays the contents of a file, printing the entire contents to the screen without interruption.

-n
Numbers the output on the left margin

`less [options] files`

This command can be used to browse the contents of the specified file. Scroll half a screen page up or down with PgUp and PgDn or a full screen page down with

Space. Jump to the beginning or end of a file using Home and End. Press Q to exit the program.

`grep [options] searchstring files`

The grep command finds a specific search string in the specified files. If the search string is found, the command displays the line in which `searchstring` was found along with the filename.

`-i`
Ignores case

`-H`
Only displays the names of the respective files, but not the text lines

`-n`
Additionally displays the numbers of the lines in which it found a hit

`-l`
Only lists the files in which `searchstring` does not occur

`diff [options] file1 file2`

The `diff` command compares the contents of any two files. The output produced by the program lists all lines that do not match. This is frequently used by programmers who need only send their program alterations and not the entire source code.

`-q`
Only reports whether the two files differ

`-u`
Produces a “unified” diff, which makes the output more readable

File Systems

`mount [options] [device] mountpoint`

This command can be used to mount any data media, such as hard disks, CD-ROM drives, and other drives, to a directory of the Linux file system.

-r

mount read-only

-t filesystem

Specifies the file system, commonly `ext2` for Linux hard disks, `msdos` for MS-DOS media, `vfat` for the Windows file system, and `iso9660` for CDs

For hard disks not defined in the file `/etc/fstab`, the device type must also be specified. In this case, only `root` can mount it. If the file system should also be mounted by other users, enter the option `user` in the appropriate line in the `/etc/fstab` file (separated by commas) and save this change. Further information is available in the `mount(1)` man page.

umount [options] mountpoint

This command unmounts a mounted drive from the file system. To prevent data loss, run this command before taking a removable data medium from its drive. Normally, only `root` is allowed to run the commands `mount` and `umount`. To enable other users to run these commands, edit the `/etc/fstab` file to specify the option `user` for the respective drive.

27.3.2 System Commands

The following section lists a few of the most important commands needed for retrieving system information and process and network control.

System Information

df [options] [directory]

The `df` (disk free) command, when used without any options, displays information about the total disk space, the disk space currently in use, and the free space on all the mounted drives. If a directory is specified, the information is limited to the drive on which that directory is located.

-h

Shows the number of occupied blocks in gigabytes, megabytes, or kilobytes—in human-readable format

-T
Type of file system (ext2, nfs, etc.)

du [options] [path]

This command, when executed without any parameters, shows the total disk space occupied by files and subdirectories in the current directory.

-a
Displays the size of each individual file

-h
Output in human-readable form

-s
Displays only the calculated total size

free [options]

The command `free` displays information about RAM and swap space usage, showing the total and the used amount in both categories. See Section 30.1.6, “The free Command” (page 454) for more information.

-b
Output in bytes

-k
Output in kilobytes

-m
Output in megabytes

date [options]

This simple program displays the current system time. If run as `root`, it can also be used to change the system time. Details about the program are available in the date(1) man page.

Processes

`top [options]`

`top` provides a quick overview of the currently running processes. Press `H` to access a page that briefly explains the main options for customizing the program.

`ps [options] [process ID]`

If run without any options, this command displays a table of all your own programs or processes—those you started. The options for this command are not preceded by hyphen.

aux

Displays a detailed list of all processes, independent of the owner

`kill [options] process ID`

Unfortunately, sometimes a program cannot be terminated in the normal way. In most cases, you should still be able to stop such a runaway program by executing the `kill` command, specifying the respective process ID (see `top` and `ps`). `kill` sends a *TERM* signal that instructs the program to shut itself down. If this does not help, the following parameter can be used:

-9

Sends a *KILL* signal instead of a *TERM* signal, bringing the specified process to an end in almost all cases

`killall [options] processname`

This command is similar to `kill`, but uses the process name (instead of the process ID) as an argument, killing all processes with that name.

Network

`ping [options] hostname or IP address`

The `ping` command is the standard tool for testing the basic functionality of TCP/IP networks. It sends a small data packet to the destination host, requesting an immediate reply. If this works, `ping` displays a message to that effect, which indicates that the network link is basically functioning.

-c *number*

Determines the total number of packages to send and ends after they have been dispatched (by default, there is no limitation set)

-f

flood ping: sends as many data packages as possible; a popular means, reserved for `root`, to test networks

-i *value*

Specifies the interval between two data packages in seconds (default: one second)

nslookup

The domain name system resolves domain names to IP addresses. With this tool, send queries to name servers (DNS servers).

telnet [options] hostname or IP address [port]

Telnet is actually an Internet protocol that enables you to work on remote hosts across a network. telnet is also the name of a Linux program that uses this protocol to enable operations on remote computers.

WARNING

Do not use telnet over a network on which third parties can "eavesdrop". Particularly on the Internet, use encrypted transfer methods, such as `ssh`, to avoid the risk of malicious misuse of a password (see the man page for `ssh`).

Miscellaneous

passwd [options] [username]

Users may change their own passwords at any time using this command. The administrator `root` can use the command to change the password of any user on the system.

su [options] [username]

The `su` command makes it possible to log in under a different username from a running session. Specify a username and the corresponding password. The password is not required from `root`, because `root` is authorized to assume the identity of any user. When using the command without specifying a username, you are prompted for the `root` password and change to the superuser (`root`).

-

Use `su -` to start a login shell for the different user.

`halt [options]`

To avoid loss of data, you should always use this program to shut down your system.

`reboot [options]`

Does the same as `halt` except the system performs an immediate reboot.

`clear`

This command cleans up the visible area of the console. It has no options.

27.3.3 For More Information

There are many more commands than listed in this chapter. For information about other commands or more detailed information, the O'Reilly publication *Linux in a Nutshell* is recommended.

27.4 The vi Editor

Text editors are still used for many system administration tasks as well as for programming. In the world of Unix, vi stands out as an editor that offers comfortable editing functions and is more ergonomic than many editors with mouse support.

27.4.1 Operating Modes

Basically, vi makes use of three operating modes: *insert* mode, *command* mode, and *extended* mode. The keys have different functions depending on the mode. On start-up, vi is normally set to the *command* mode. The first thing to learn is how to switch between the modes:

Command Mode to Insert Mode

There are many possibilities, including A for append, I for insert, or O for a new line under the current line.

Insert Mode to Command Mode

Press Esc to exit the *insert* mode. vi cannot be terminated in *insert* mode, so it is important to get used to pressing Esc.

Command Mode to Extended Mode

The *extended* mode of vi can be activated by entering a colon (:). The *extended* or *ex* mode is similar to an independent line-oriented editor that can be used for various simple and more complex tasks.

Extended Mode to Command Mode

After executing a command in *extended* mode, the editor automatically returns to *command* mode. If you decide not to execute any command in *extended* mode, delete the colon with <—. The editor returns to *command* mode.

It is not possible to switch directly from *insert* mode to *extended* mode without first switching to *command* mode.

vi, like other editors, has its own procedure for terminating the program. You cannot terminate vi while in *insert* mode. First, exit *insert* mode by pressing Esc. Subsequently, you have two options:

1. *Exit without saving:* To terminate the editor without saving the changes, enter : – Q – ! in *command* mode. The exclamation mark (!) causes vi to ignore any changes.

2. *Save and exit:* There are several possibilities to save your changes and terminate the editor. In *command* mode, use Shift + Z + Z. To exit the program saving all changes using the *extended* mode, enter : – W – Q. In *extended* mode, `w` stands for writ and `q` for quit.

27.4.2 vi in Action

vi can be used as a normal editor. In *insert* mode, enter text and delete text with the <— and Del keys. Use the arrow keys to move the cursor.

However, these control keys often cause problems, because there are many terminal types that use special key codes. This is where the *command* mode comes into play. Press Esc to switch from *insert* mode to *command* mode. In *command* mode, move the cursor with H, J, K, and L. The keys have the following functions:

H
: move one character to the left

J
: move one line down

K
: move one line up

L
: move one character to the right

The commands in *command* mode allow diverse variations. To execute a command several times, simply enter the number of repetitions before entering the actual command. For example, enter 5 L to move the cursor five characters to the right.

A selection of important commands is shown in Table 27.1, "Simple Commands of the vi Editor" (page 407) This list is far from complete. More complete lists are available in the documentation found in Section 27.4.3, "For More Information" (page 408)

Table 27.1 *Simple Commands of the vi Editor*

Esc	Change to command mode
I	Change to insert mode (characters appear at the current cursor position)
A	Change to insert mode (characters are inserted after the current cursor position)
Shift + A	Changes to insert mode (characters are added at the end of the line)
Shift + R	Change to replace mode (overwrite the old text)

R	Replace character under the cursor
O	Change to insert mode (a new line is inserted after the current one)
Shift + O	Change to insert mode (a new line is inserted before the current one)
X	Delete the current character
D – D	Delete the current line
D – W	Delete up to the end of the current word
C – W	Change to insert mode (the rest of the current word is overwritten by the next entries you make)
U	Undo the last command
Ctrl + R	Redo change that was undone
Shift + J	Join the following line with the current one
.	Repeat the last command

27.4.3 For More Information

vi supports a wide range of commands. It enables the use of macros, shortcuts, named buffers, and many other useful features. A detailed description of the various options would exceed the scope of this manual. SUSE Linux comes with vim (vi improved), an improved version of vi. There are numerous information sources for this application:

- vimtutor is an interactive tutor for vim.
- In vim, enter the command `:help` to get help for many subjects.
- A book about vim is available online at `http://www.truth.sk/vim/vimbook-OPL.pdf`.

- The Web pages of the vim project at `http://www.vim.org` feature all kinds of news, mailing lists, and other documentation.
- A number of vim sources are available on the Internet: `http://www.selflinux.org/selflinux/html/vim.html`, `http://www.linuxgazette.com/node/view/9039`, and `http://www.apmaths.uwo.ca/~xli/vim/vim_tutorial.html`. See `http://linux-universe.com/HOWTO/Vim-HOWTO/vim-tutorial.html` for further links to tutorials.

IMPORTANT: The VIM License

vim is "charityware," which means that the authors do not charge any money for the software but encourage you to support a nonprofit project with a monetary contribution. This project solicits help for poor children in Uganda. More information is available online at `http://iccf-holland.org/index.html`, `http://www.vim.org/iccf/`, and `http://www.iccf.nl/`.

Booting and Configuring a Linux System 28

Booting a Linux system involves various different components. This chapter outlines the underlying principles and highlights the components involved. The concept of runlevels and SUSE's system configuration with `sysconfig` are also discussed in this chapter.

28.1 The Linux Boot Process

The Linux boot process consists of several stages each represented by another component. The following list briefly summarizes the boot process and features all the major components involved.

1. **BIOS** After the computer has been turned on, the BIOS initializes the screen and keyboard and tests the main memory. Up to this stage, the machine does not access any mass storage media. Subsequently, the information about the current date, time, and the most important peripherals are loaded from the CMOS values. When the first hard disk and its geometry are recognized, the system control passes from the BIOS to the boot loader.

2. **Boot Loader** The first physical 512-byte data sector of the first hard disk is loaded into the main memory and the *boot loader* that resides at the beginning of this sector takes over. The commands executed by the boot loader determine the remaining part of the boot process. Therefore, the first 512 bytes on the first hard disk are referred to as the *Master Boot Record* (MBR). The boat loader then passes control to the actual operating system, in this case, the Linux kernel. More

information about GRUB, the Linux boot loader, can be found in Chapter 29, *The Boot Loader* (page 427).

3. **Kernel and initramfs** To pass system control, the boot loader loads both the kernel and an initial RAM-based file system (initramfs) into memory. The contents of the initial ramfs can be used by the kernel directly. The init ramfs contains a small executable called init that handles the mounting of the real root file system. In former versions of SUSE Linux, these tasks were handled by initrd and linuxrc, respectively. For more information about initramfs, refer to Section 28.1.1, "initramfs" (page 412).

4. **init on initramfs** This program performs all actions needed to mount the proper root file system, like providing kernel functionality for the needed file system and device drivers for mass storage controllers. After the root file system has been found, it is checked for errors and mounted. If this has been successful, the initramfs is cleaned and the init program on the root file system is executed. For more information about init, refer to Section 28.1.2, "init on initramfs" (page 413).

5. **init** init handles the actual booting of the system through several different levels providing different functionality. init is described in Section 28.2, "The init Process" (page 414).

28.1.1 initramfs

initramfs is a small file system that the kernel can load to a RAM disk. It provides a minimal Linux environment that enables the execution of programs before the actual root file system is mounted. This minimal Linux environment is loaded into memory by BIOS routines and does not have specific hardware requirements other than sufficient memory. initramfs must always provide an executable named init that should execute the actual init program on the root file system for the boot process to proceed.

Before the actual root file system can be mounted and the actual operating system can be started, the kernel needs the corresponding drivers to access the device on which the root file system is located. These drivers may include special drivers for certain kinds of hard drives or even network drivers to access a network file system. The needed modules for the root file system may be loaded by init on initramfs. initramfs is available during the entire boot process. This makes it possible to handle all hotplug events generated during boot.

If you need to change hardware (hard disks) in an installed system and this hardware requires different drivers to be present in the kernel at boot time, you must update initramfs. This is done in the same way as with initramfs' predecessor, initrd, by calling `mkinitrd`. Calling `mkinitrd` without any argument creates an initramfs. Calling `mkinitrd -R` creates an initrd. In SUSE Linux, the modules to load are specified by the variable `INITRD_MODULES` in `/etc/sysconfig/kernel`. After installation, this variable is automatically set to the correct value. The modules are loaded in exactly the order in which they appear in `INITRD_MODULES`. This is especially important if several SCSI drivers are used, because otherwise the names of the hard disks would change. Strictly speaking, it would be sufficient just to load those drivers needed to access the root file system. However, all SCSI drivers needed for installation are loaded by means of initramfs or initrd because later loading could be problematic.

IMPORTANT: Updating initramfs or initrd

The boot loader loads initramfs or initrd in the same way as the kernel. It is not necessary to reinstall GRUB after updating initramfs or initrd, because GRUB searches the directory for the right file when booting.

28.1.2 init on initramfs

The main purpose of init on initramfs is to prepare the mounting of and access to the real root file system. Depending on your actual system configuration, init is responsible for the following tasks.

Loading Kernel Modules

Depending on your hardware configuration, special drivers may be needed to access the hardware components of your computer (the most important component being your hard drive). To access the final root file system, the kernel needs to load the proper file system drivers.

Managing RAID and LVM Setups

If you configured your system to hold the root file system under RAID or LVM, init sets up LVM or RAID to enable access to the root file sytem later. Information about RAID can be found in Section 2.3, “Soft RAID Configuration” (page 65). Information about LVM can be found in Section 2.2, “LVM Configuration” (page 58).

Managing Network Configuration

If you configured your system to use a network-mounted root file system (mounted via NFS), init must make sure that the proper network drivers are loaded and that they are set up to allow access to the root file system.

When init is called during the initial boot as part of the installation process, its tasks differ from those mentioned earlier:

Finding the Installation Medium

As you start the installation process, your machine loads an installation kernel and a special initrd with the YaST installer from the installation medium. The YaST installer, which is run in a RAM file system, needs to have information about the actual location of the installation medium to access it and install the operating system.

Initiating Hardware Recognition and Loading Appropriate Kernel Modules

As mentioned in Section 28.1.1, “initramfs” (page 412), the boot process starts with a minimum set of drivers that can be used with most hardware configurations. init starts an initial hardware scanning process that determines the set of drivers suitable for your hardware configuration. These values are later written to `INITRD_MODULES` in `/etc/sysconfig/kernel` to enable any subsequent boot process to use a custom initrd. During the installation process, init loads this set of modules.

Loading the Installation System or Rescue System

As soon as the hardware has been properly recognized and the appropriate drivers have been loaded, init starts the installation system, which contains the actual YaST installer, or the rescue system.

Starting YaST

Finally, init starts YaST, which starts package installation and system configuration.

28.2 The init Process

The program init is the process with process number 1. It is responsible for initializing the system in the required way. init takes a special role. It is started directly by the kernel and resists signal 9, which normally kills processes. All other programs are either started directly by init or by one of its child processes.

init is centrally configured in the `/etc/inittab` file where the *runlevels* are defined (see Section 28.2.1, "Runlevels" (page 415)). The file also specifies which services and daemons are available in each of the levels. Depending on the entries in `/etc/inittab`, several scripts are run by init. For reasons of clarity, these scripts, called *init scripts*, all reside in the directory `/etc/init.d` (see Section 28.2.2, "Init Scripts" (page 418)).

The entire process of starting the system and shutting it down is maintained by init. From this point of view, the kernel can be considered a background process whose task is to maintain all other processes and adjust CPU time and hardware access according to requests from other programs.

28.2.1 Runlevels

In Linux, *runlevels* define how the system is started and what services are available in the running system. After booting, the system starts as defined in `/etc/inittab` in the line `initdefault`. Usually this is `3` or `5`. See Table 28.1, "Available Runlevels" (page 415). As an alternative, the runlevel can be specified at boot time (at the boot prompt, for instance). Any parameters that are not directly evaluated by the kernel itself are passed to init.

Table 28.1 *Available Runlevels*

Runlevel	Description
0	System halt
S	Single user mode; from the boot prompt, only with US keyboard mapping
1	Single user mode
2	Local multiuser mode without remote network (NFS, etc.)
3	Full multiuser mode with network
4	Not used

Runlevel	Description
5	Full multiuser mode with network and X display manager—KDM, GDM, or XDM
6	System reboot

IMPORTANT: Avoid Runlevel 2 with a /usr Partition Mounted via NFS

You should not use runlevel 2 if your system mounts the `/usr` partition via NFS. The `/usr` directory holds important programs essential for the proper functioning of the system. Because the NFS service is not available in runlevel 2 (local multiuser mode without remote network), the system would be seriously restricted in many aspects.

To change runlevels while the system is running, enter `init` and the corresponding number as an argument. Only the system administrator is allowed to do this. The following list summarizes the most important commands in the runlevel area.

`init 1` **or** **`shutdown now`**
The system changes to *single user mode*. This mode is used for system maintenance and administration tasks.

`init 3`
All essential programs and services (including network) are started and regular users are allowed to log in and work with the system without a graphical environment.

`init 5`
The graphical environment is enabled. This can be one of the desktops (GNOME or KDE) or any window manager.

`init 0` **or** **`shutdown -h now`**
The system halts.

`init 6` **or** **`shutdown -r now`**
The system halts then reboots.

Runlevel `5` is the default runlevel in all SUSE Linux standard installations. Users are prompted for login with a graphical interface. If the default runlevel is `3`, the X Window

System must be configured properly, as described in Chapter 35, *The X Window System* (page 509), before the runlevel can be switched to `5`. If this is done, check whether the system works in the desired way by entering `init 5`. If everything turns out as expected, you can use YaST to set the default runlevel to `5`.

WARNING: Errors in /etc/inittab May Result in a Faulty System Boot

If `/etc/inittab` is damaged, the system might not boot properly. Therefore, be extremely careful while editing `/etc/inittab` and always keep a backup of an intact version. To repair damage, try entering `init=/bin/sh` after the kernel name at the boot prompt to boot directly into a shell. After that, make your root file system writable with the command `mount -o remount,rw /` and replace `/etc/inittab` with your backup version using `cp`. To prevent file system errors, change your root file system to read-only before you reboot with `mount -o remount,ro /`.

Generally, two things happen when you change runlevels. First, stop scripts of the current runlevel are launched, closing down some programs essential for the current runlevel. Then start scripts of the new runlevel are started. Here, in most cases, a number of programs are started. For example, the following occurs when changing from runlevel 3 to 5:

1. The administrator (`root`) requests init to change to a different runlevel by entering `init 5`.

2. init consults its configuration file (`/etc/inittab`) and determines it should start `/etc/init.d/rc` with the new runlevel as a parameter.

3. Now `rc` calls all the stop scripts of the current runlevel, but only those for which there is no start script in the new runlevel. In this example, these are all the scripts that reside in `/etc/init.d/rc3.d` (old runlevel was 3) and start with a `K`. The number following `K` specifies the order to start, because there are some dependencies to consider.

4. The last things to start are the start scripts of the new runlevel. These are, in this example, in `/etc/init.d/rc5.d` and begin with an `S`. The same procedure regarding the order in which they are started is applied here.

When changing into the same runlevel as the current runlevel, init only checks /etc/inittab for changes and starts the appropriate steps, for example, for starting a getty on another interface.

28.2.2 Init Scripts

There are two types of scripts in /etc/init.d:

Scripts Executed Directly by init
This is the case only during the boot process or if an immediate system shutdown is initiated (power failure or a user pressing Ctrl + Alt + Del). The execution of these scripts is defined in /etc/inittab.

Scripts Executed Indirectly by init
These are run when changing the runlevel and always call the master script /etc/init.d/rc, which guarantees the correct order of the relevant scripts.

All scripts are located in /etc/init.d. Scripts for changing the runlevel are also found there, but are called through symbolic links from one of the subdirectories (/etc/init.d/rc0.d to /etc/init.d/rc6.d). This is just for clarity reasons and avoids duplicate scripts if they are used in several runlevels. Because every script can be executed as both a start and a stop script, these scripts must understand the parameters start and stop. The scripts also understand the restart, reload, force-reload, and status options. These different options are explained in Table 28.2, “Possible init Script Options” (page 418). Scripts that are run directly by init do not have these links. They are run independently from the runlevel when needed.

Table 28.2 *Possible init Script Options*

Option	Description
start	Start service.
stop	Stop service.
restart	If the service is running, stop it then restart it. If it is not running, start it.

Option	Description
`reload`	Reload the configuration without stopping and restarting the service.
`force-reload`	Reload the configuration if the service supports this. Otherwise, do the same as if `restart` had been given.
`status`	Show the current status of service.

Links in each runlevel-specific subdirectory make it possible to associate scripts with different runlevels. When installing or uninstalling packages, these links are added and removed with the help of the program insserv (or using `/usr/lib/lsb/install_initd`, which is a script calling this program). See the insserv(8) man page for details.

A short introduction to the boot and stop scripts launched first or last, respectively, follows as well as an explanation of the maintaining script.

boot

Executed while starting the system directly using init. It is independent of the chosen runlevel and is only executed once. Here, the `proc` and `pts` file systems are mounted and blogd (boot logging daemon) is activated. If the system is booted for the first time after an update or an installation, the initial system configuration is started.

The blogd daemon is a service started by boot and rc before any other one. It is stopped after the actions triggered by the above scripts (running a number of subscripts, for example) are completed. blogd writes any screen output to the log file `/var/log/boot.msg`, but only if and when `/var` is mounted read-write. Otherwise, blogd buffers all screen data until `/var` becomes available. Get further information about blogd on the blogd(8) man page.

The script `boot` is also responsible for starting all the scripts in `/etc/init.d/boot.d` with a name that starts with `S`. There, the file systems are checked and loop devices are configured if needed. The system time is also set. If an error occurs while automatically checking and repairing the file system, the system administrator can intervene after first entering the root password. Last executed is the script `boot.local`.

boot.local

Here, enter additional commands to execute at boot before changing into a runlevel. It can be compared to AUTOEXEC.BAT on DOS systems.

boot.setup

This script is executed when changing from single user mode to any other runlevel and is responsible for a number of basic settings, such as the keyboard layout and initialization of the virtual consoles.

halt

This script is only executed while changing into runlevel 0 or 6. Here, it is executed either as halt or as reboot. Whether the system shuts down or reboots depends on how halt is called.

rc

This script calls the appropriate stop scripts of the current runlevel and the start scripts of the newly selected runlevel.

You can create your own scripts and easily integrate them into the scheme described above. For instructions about formatting, naming, and organizing custom scripts, refer to the specifications of the LSB and to the man pages of init, init.d, and insserv. Additionally consult the man pages of startproc and killproc.

WARNING: Faulty init Scripts May Halt Your System

Faulty init scripts may hang your machine. Edit such scripts with great care and, if possible, subject them to heavy testing in the multiuser environment. Some useful information about init scripts can be found in Section 28.2.1, "Runlevels" (page 415).

To create a custom init script for a given program or service, use the file /etc/init.d/skeleton as a template. Save a copy of this file under the new name and edit the relevant program and filenames, paths, and other details as needed. You may also need to enhance the script with your own parts, so the correct actions are triggered by the init procedure.

The INIT INFO block at the top is a required part of the script and should be edited. See Example 28.1, "A Minimal INIT INFO Block" (page 421).

Example 28.1 *A Minimal INIT INFO Block*

```
### BEGIN INIT INFO
# Provides:          FOO
# Required-Start:    $syslog $remote_fs
# Required-Stop:     $syslog $remote_fs
# Default-Start:     3 5
# Default-Stop:      0 1 2 6
# Description:       Start FOO to allow XY and provide YZ
### END INIT INFO
```

In the first line of the `INFO` block, after `Provides:`, specify the name of the program or service controlled by this init script. In the `Required-Start:` and `Required-Stop:` lines, specify all services that need to be started or stopped before the service itself is started or stopped. This information is used later to generate the numbering of script names, as found in the runlevel directories. After `Default-Start:` and `Default-Stop:`, specify the runlevels in which the service should automatically be started or stopped. Finally, for `Description:`, provide a short description of the service in question.

To create the links from the runlevel directories (`/etc/init.d/rc?.d/`) to the corresponding scripts in `/etc/init.d/`, enter the command `insserv` *`new-script-name`*. The insserv program evaluates the `INIT INFO` header to create the necessary links for start and stop scripts in the runlevel directories (`/etc/init.d/rc?.d/`). The program also takes care of the correct start and stop order for each runlevel by including the necessary numbers in the names of these links. If you prefer a graphical tool to create such links, use the runlevel editor provided by YaST, as described in Section 28.2.3, "Configuring System Services (Runlevel) with YaST" (page 422).

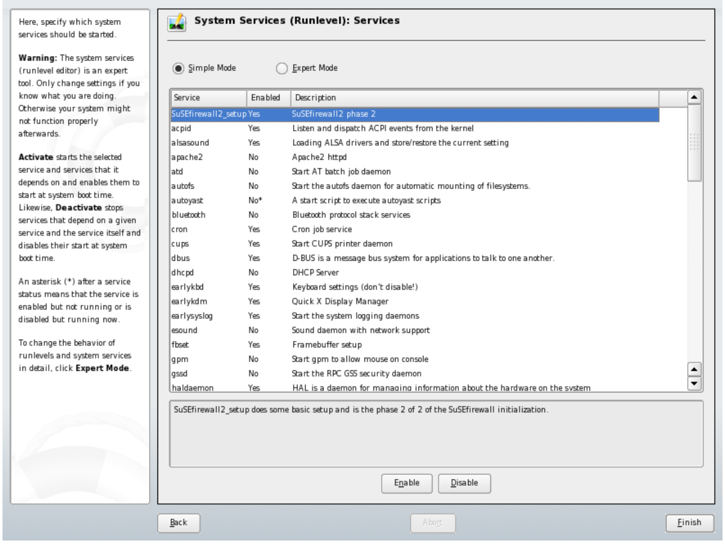
If a script already present in `/etc/init.d/` should be integrated into the existing runlevel scheme, create the links in the runlevel directories right away with insserv or by enabling the corresponding service in the runlevel editor of YaST. Your changes are applied during the next reboot—the new service is started automatically.

Do not set these links manually. If something is wrong in the `INFO` Block, problems will arise when `insserv` is run later for some other service.

28.2.3 Configuring System Services (Runlevel) with YaST

After starting this YaST module with *YaST* → *System* → *System Services (Runlevel)*, it displays an overview listing all the available services and the current status of each service (disabled or enabled). Decide whether to use the module in *Simple Mode* or in *Expert Mode*. The default *Simple Mode* should be sufficient for most purposes. The left column shows the name of the service, the center column indicates its current status, and the right column gives a short description. For the selected service, a more detailed description is provided in the lower part of the window. To enable a service, select it in the table then select *Enable*. The same steps apply to disable a service.

Figure 28.1 *System Services (Runlevel)*



Here, specify which system services should be started.

Warning: The system services (runlevel editor) is an expert tool. Only change settings if you know what you are doing. Otherwise your system might not function properly afterwards.

Activate starts the selected service and services that it depends on and enables them to start at system boot time. Likewise, **Deactivate** stops services that depend on a given service and the service itself and disables their start at system boot time.

An asterisk (*) after a service status means that the service is enabled but not running or is disabled but running now.

To change the behavior of runlevels and system services in detail, click **Expert Mode**.

System Services (Runlevel): Services

● Simple Mode ○ Expert Mode

Service	Enabled	Description
SuSEfirewall2_setup	Yes	SuSEfirewall2 phase 2
acpid	Yes	Listen and dispatch ACPI events from the kernel
alsasound	Yes	Loading ALSA drivers and store/restore the current setting
apache2	No	Apache2 httpd
atd	No	Start AT batch job daemon
autofs	No	Start the autofs daemon for automatic mounting of filesystems.
autoyast	No*	A start script to execute autoyast scripts
bluetooth	No	Bluetooth protocol stack services
cron	Yes	Cron job service
cups	Yes	Start CUPS printer daemon
dbus	Yes	D-BUS is a message bus system for applications to talk to one another.
dhcpd	No	DHCP Server
earlykbd	Yes	Keyboard settings (don't disable!)
earlykdm	Yes	Quick X Display Manager
earlysyslog	Yes	Start the system logging daemons
esound	No	Sound daemon with network support
fbset	Yes	Framebuffer setup
gpm	No	Start gpm to allow mouse on console
gssd	No	Start the RPC GSS security daemon
haldaemon	Yes	HAL is a daemon for managing information about the hardware on the system

SuSEfirewall2_setup does some basic setup and is the phase 2 of 2 of the SuSEfirewall initialization.

For detailed control over the runlevels in which a service is started or stopped or to change the default runlevel, first select *Expert Mode*. The current default runlevel or “initdefault” (the runlevel into which the system boots by default) is displayed at the top. Normally, the default runlevel of a SUSE Linux system is runlevel 5 (full multiuser mode with network and X). A suitable alternative might be runlevel 3 (full multiuser mode with network).

This YaST dialog allows the selection of one of the runlevels (as listed in Table 28.1, "Available Runlevels" (page 415)) as the new default. Additionally use the table in this window to enable or disable individual services and daemons. The table lists the services and daemons available, shows whether they are currently enabled on your system, and, if so, for which runlevels. After selecting one of the rows with the mouse, click the check boxes representing the runlevels (*B*, *0*, *1*, *2*, *3*, *5*, *6*, and *S*) to define the runlevels in which the selected service or daemon should be running. Runlevel 4 is initially undefined to allow creation of a custom runlevel. A brief description of the currently selected service or daemon is provided below the table overview.

With *Start, Stop, or Refresh*, decide whether a service should be activated. *Refresh status* checks the current status. *Set or Reset* lets you select whether to apply your changes to the system or to restore the settings that existed before starting the runlevel editor. Selecting *Finish* saves the changed settings to disk.

WARNING: Faulty Runlevel Settings May Damage Your System

Faulty runlevel settings may render a system unusable. Before applying your changes, make absolutely sure that you know their consequences.

28.3 System Configuration via /etc/sysconfig

The main configuration of SUSE Linux is controlled by the configuration files in `/etc/sysconfig`. The individual files in `/etc/sysconfig` are only read by the scripts to which they are relevant. This ensures that network settings, for example, only need to be parsed by network-related scripts. Many other system configuration files are generated according to the settings in `/etc/sysconfig`. This task is performed by SuSEconfig. For example, if you change the network configuration, SuSEconfig might make changes to the file `/etc/host.conf` as well, because this is one of the files relevant for the network configuration. This concept enables you to make basic changes to your configuration without needing to reboot the system.

There are two ways to edit the system configuration. Either use the YaST sysconfig Editor or edit the configuration files manually.

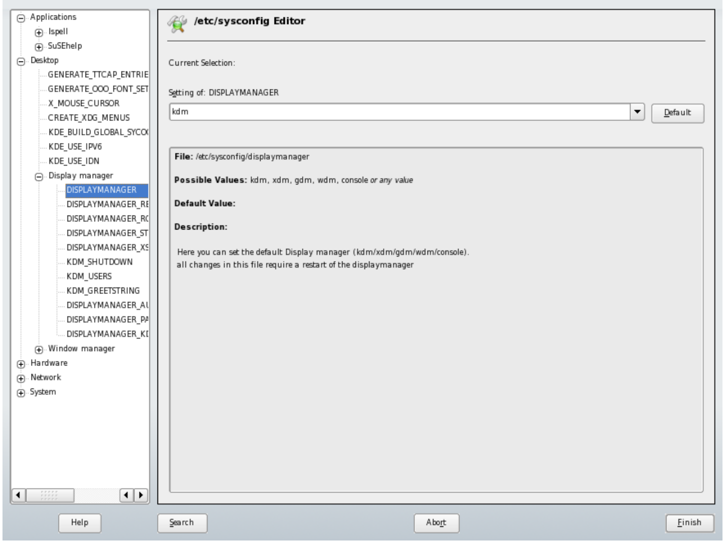
28.3.1 Changing the System Configuration Using the YaST sysconfig Editor

The YaST sysconfig editor provides an easy-to-use front-end to system configuration. Without any knowledge of the actual location of the configuration variable you need to change, you can just use the built-in search function of this module, change the value of the configuration variable as needed, and let YaST take care of applying these changes, updating configurations that depend on the values set in `sysconfig` and restarting services.

WARNING: Modifying /etc/sysconfig/* Files Can Damage Your Installation

Do not modify the `/etc/sysconfig` files if you lack previous experience and knowledge. It could do considerable damage to your system. The files in `/etc/sysconfig` include a short comment for each variable to explain what effect they actually have.

Figure 28.2 *System Configuration Using the sysconfig Editor*



/etc/sysconfig Editor

Current Selection:

Setting of: DISPLAYMANAGER

kdm

File: /etc/sysconfig/displaymanager

Possible Values: kdm, xdm, gdm, wdm, console *or any value*

Default Value:

Description:

Here you can set the default Display manager (kdm/xdm/gdm/wdm/console).
all changes in this file require a restart of the displaymanager

The YaST sysconfig dialog is split into three parts. The left part of the dialog shows a tree view of all configurable variables. When you select a variable, the right part displays both the current selection and the current setting of this variable. Below, a third window displays a short description of the variable's purpose, possible values, the default value, and the actual configuration file from which this variable originates. The dialog also provides information about which configuration script is executed after changing the variable and which new service is started as a result of the change. YaST prompts you to confirm your changes and informs you which scripts will be executed after you leave the dialog by selecting *Finish*. Also select the services and scripts to skip for now, so they are started later. YaST applies all changes automatically and restarts any services involved for your changes to take an effect.

28.3.2 Changing the System Configuration Manually

To manually change the system configuration, proceed as follows

1. Become `root`.

2. Bring the system into single user mode (runlevel 1) with `init 1`.

3. Change the configuration files as needed with an editor of your choice.

   If you do not use YaST to change the configuration files in `/etc/sysconfig`, make sure that empty variable values are represented by two quotation marks (`KEYTABLE=""`) and that values with blanks in them are enclosed in quotation marks. Values consisting of one word only do not need to be quoted.

4. Execute `SuSEconfig` to make sure that the changes take effect.

5. Bring your system back to the previous runlevel with a command like `init` *`default_runlevel`*. Replace *`default_runlevel`* with the default runlevel of the system. Choose `5` if you want to return to full multiuser with network and X or choose `3` if you prefer to work in full multiuser with network.

This procedure is mainly relevant when changing systemwide settings, such as the network configuration. Small changes should not require going into single user mode, but you could still do so to make absolutely sure that all the programs concerned are correctly restarted.

TIP: Configuring Automated System Configuration

To disable the automated system configuration by SuSEconfig, set the variable `ENABLE_SUSECONFIG` in `/etc/sysconfig/suseconfig` to `no`. Do not disable SuSEconfig if you want to use the SUSE installation support. It is also possible to disable the autoconfiguration partially.

29 The Boot Loader

This chapter describes how to configure GRUB, the boot loader used in SUSE Linux. A special YaST module is available for performing all settings. If you are not familiar with the subject of booting in Linux, read the following sections to acquire some background information. This chapter also describes some of the problems frequently encountered when booting with GRUB and their solutions.

This chapter focuses on boot management and the configuration of the boot loader GRUB. The boot procedure as a whole is outlined in Chapter 28, *Booting and Configuring a Linux System* (page 411). A boot loader represents the interface between machine (BIOS) and the operating system (SUSE Linux). The configuration of the boot loader directly impacts the start of the operating system.

The following terms appear frequently in this chapter and might need some explanation:

Master Boot Record

> The structure of the MBR is defined by an operating system–independent convention. The first 446 bytes are reserved for the program code. They typically hold the boot loader program, in this case, GRUB. The next 64 bytes provide space for a partition table of up to four entries (see Section “Partition Types” (Chapter 1, *Installation with YaST*, ↑Start-Up)). The partition table contains information about the partitioning of the hard disk and the file system type. The operating system needs this table for handling the hard disk. The last two bytes of the MBR must contain a static “magic number” (AA55). An MBR containing a different value is regarded as invalid by the BIOS and all PC operating systems.

Boot Sectors

Boot sectors are the first sectors of hard disk partitions with the exception of the extended partition, which merely serves as a “container” for other partitions. These boot sectors have 512 bytes of space for code used to boot an operating system installed in the respective partition. This applies to boot sectors of formatted DOS, Windows, and OS/2 partitions, which also contain some important basic data of the file system. In contrast, the boot sectors of Linux partitions are initally empty after setting up a file system. Therefore, a Linux partition is not bootable by itself, even if it contains a kernel and a valid root file system. A boot sector with valid code for booting the system has the same magic number as the MBR in its last two bytes (`AA55`).

29.1 Boot Management

In the simplest case—if only one operating system is installed on a computer—the boot loader automaticly takes place in the MBR. If several operating systems are installed on a computer, the following options are available:

Booting Additional Systems from External Media

One of the operating systems is booted from the hard disk. The other operating systems are booted by means of a boot manager installed on an external medium (floppy disk, CD, USB storage medium).

Installing a Boot Manager in the MBR

A boot manager enables concurrent installation and alternate use of several systems on one computer. Users can select the system to boot during the boot process. To change to another system, the computer must be rebooted. This is only possible if the selected boot manager is compatible with the installed operating systems.

29.2 Selecting a Boot Loader

By default, the boot loader GRUB is used in SUSE Linux. However, in some cases and for special hardware and software constellations, LILO may be more suitable. If you update from an older SUSE Linux version that uses LILO, LILO is installed.

Information about the installation and configuration of LILO is available in the Support Database under the keyword LILO and in `/usr/share/doc/packages/lilo`.

29.3 Booting with GRUB

GRUB (Grand Unified Bootloader) comprises two stages. stage1 consists of 512 bytes and is written to the MBR or the boot sector of a hard disk partition or floppy disk. Subsequently, stage2 is loaded. This stage contains the actual program code. The only task of the first stage is to load the second stage of the boot loader.

stage2 is able to access file systems. Currently, Ext2, Ext3, ReiserFS, Minix, and the DOS FAT file system used by Windows are supported. To a certain extent, JFS, XFS, and UFS and FFS used by BSD systems are also supported. Since version 0.95, GRUB is also able to boot from a CD or DVD containing an ISO 9660 standard file system pursuant to the "El Torito" specification. Even before the system is booted, GRUB can access file systems of supported BIOS disk devices (floppy disks or hard disks, CD drives, and DVD drives detected by the BIOS). Therefore, changes to the GRUB configuration file (`menu.lst`) do not require a reinstallation of the boot manager. When the system is booted, GRUB reloads the menu file with the valid paths and partition data of the kernel or the initial RAM disk (`initrd`) and locates these files.

The actual configuration of GRUB is based on three files that are described below:

`/boot/grub/menu.lst`

This file contains all information about partitions or operating systems that can be booted with GRUB. Without this information, the system control cannot be passed to the operating system.

`/boot/grub/device.map`

This file translates device names from the GRUB and BIOS notation to Linux device names.

`/etc/grub.conf`

This file contains the parameters and options the GRUB shell needs for installing the boot loader correctly.

GRUB can be controlled in various ways. Boot entries from an existing configuration can be selected from the graphical menu (splash screen). The configuration is loaded from the file `menu.lst`.

In GRUB, all boot parameters can be changed prior to booting. For example, errors made when editing the menu file can be corrected in this way. Boot commands can also be entered interactively at a kind of input prompt (see Section "Editing Menu Entries during the Boot Procedure" (page 434)). GRUB offers the possibility of determining the location of the kernel and the `initrd` prior to booting. In this way, you can even boot an installed operating system for which no entry exists in the boot loader configuration.

The *GRUB shell* provides an emulation of GRUB in the installed system. It can be used to install GRUB or test new settings before applying them. See Section 29.3.4, "The GRUB Shell" (page 437).

29.3.1 The GRUB Boot Menu

The graphical splash screen with the boot menu is based on the GRUB configuration file `/boot/grub/menu.lst`, which contains all information about all partitions or operating systems that can be booted by the menu.

Every time the system is booted, GRUB loads the menu file from the file system. For this reason, GRUB does not need to be reinstalled after every change to the file. Use the YaST boot loader to modify the GRUB configuration as described in Section 29.4, "Configuring the Boot Loader with YaST" (page 439).

The menu file contains commands. The syntax is very simple. Every line contains a command followed by optional parameters separated by spaces like in the shell. For historical reasons, some commands permit an = in front of the first parameter. Comments are introduced by a hash (`#`).

To identify the menu items in the menu overview, set a `title` for every entry. The text (including any spaces) following the keyword `title` is displayed as a selectable option in the menu. All commands up to the next `title` are executed when this menu item is selected.

The simplest case is the redirection to boot loaders of other operating systems. The command is chainloader and the argument is usually the boot block of another partition in GRUB block notation. For example:

```
chainloader (hd0,3)+1
```

The device names in GRUB are explained in Section "Naming Conventions for Hard Disks and Partitions" (page 431). This example specifies the first block of the fourth partition of the first hard disk.

Use the command kernel to specify a kernel image. The first argument is the path to the kernel image in a partition. The other arguments are passed to the kernel on the command line.

If the kernel does not have built-in drivers for access to the root partition, initrd must be specified with a separate GRUB command whose only argument is the path to the initrd file. Because the loading address of the initrd is written to the loaded kernel image, the command initrd must follow immediately after the kernel command.

The command root simplifies the specification of kernel and initrd files. The only argument of root is a GRUB device or a partition on a GRUB device. This device is used for all kernel, initrd, or other file paths for which no device is explicitly specified until the next root command. This command is not used in the menu.lst file generated during the installation. It merely facilitates manual editing.

The boot command is implied at the end of every menu entry, so it does not need to be written into the menu file. However, if you use GRUB interactively for booting, you must enter the boot command at the end. The command itself has no arguments. It merely boots the loaded kernel image or the specified chain loader.

After writing all menu entries, define one of them as the default entry. Otherwise, the first one (entry 0) is used. You can also specify a time-out in seconds after which the default entry should boot. timeout and default usually precede the menu entries. An example file is described in Section "An Example Menu File" (page 432).

Naming Conventions for Hard Disks and Partitions

The naming conventions GRUB uses for hard disks and partitions differ from those used for normal Linux devices. In GRUB, the numbering of the partitions starts with zero. This means that (hd0, 0) is the first partition of the first hard disk. On a common desktop machine with a hard disk connected as primary master, the corresponding Linux device name is /dev/hda1.

The four possible primary partitions are assigned the partition numbers 0 to 3. The logical partitions are numbered from 4:

```
(hd0,0)   first primary partition of the first hard disk
(hd0,1)   second primary partition
(hd0,2)   third primary partition
(hd0,3)   fourth primary partition (usually an extended partition)
(hd0,4)   first logical partition
(hd0,5)   second logical partition
```

GRUB does not distinguish between IDE, SCSI, and RAID devices. All hard disks recognized by the BIOS or other controllers are numbered according to the boot sequence preset in the BIOS.

Unfortunately, GRUB is not able to map the Linux device names to BIOS device names exactly. It generates this mapping with the help of an algorithm and saves it to the file device.map, which can be edited if necessary. Information about the file device.map is available in Section 29.3.2, “The File device.map” (page 436).

A complete GRUB path consists of a device name written in parentheses and the path to the file in the file system in the specified partition. The path begins with a slash. For example, the bootable kernel could be specified as follows on a system with a single IDE hard disk containing Linux in its first partition:

```
(hd0,0)/boot/vmlinuz
```

An Example Menu File

The following example shows the structure of a GRUB menu file. The example installation has a Linux boot partition under /dev/hda5, a root partition under /dev/hda7, and a Windows installation under /dev/hda1.

```
gfxmenu (hd0,4)/message
color white/blue black/light-gray
default 0
timeout 8

title linux
   kernel (hd0,4)/vmlinuz root=/dev/hda7 vga=791
   initrd (hd0,4)/initrd
```

```
title windows
   chainloader(hd0,0)+1

title floppy
   chainloader(fd0)+1

title failsafe
   kernel (hd0,4)/vmlinuz.shipped root=/dev/hda7 ide=nodma \
   apm=off acpi=off vga=normal nosmp maxcpus=0 3
   initrd (hd0,4)/initrd.shipped
```

The first block defines the configuration of the splash screen:

gfxmenu (hd0,4)/message
The background image `message` is located in `/dev/hda5`.

color white/blue black/light-gray
Color scheme: white (foreground), blue (background), black (selection), and light gray (background of the selection). The color scheme has no effect on the splash screen, only on the customizable GRUB menu that you can access by exiting the splash screen with Esc.

default 0
The first menu entry `title linux` is the one to boot by default.

timeout 8
After eight seconds without any user input, GRUB automatically boots the default entry. To deactivate automatic boot, delete the `timeout` line. If you set `timeout 0`, GRUB boots the default entry immediately.

The second and largest block lists the various bootable operating systems. The sections for the individual operating systems are introduced by `title`.

- The first entry (`title linux`) is responsible for booting SUSE Linux. The kernel (`vmlinuz`) is located in the first logical partition (the boot partition) of the first hard disk. Kernel parameters, such as the root partition and VGA mode, are appended here. The root partition is specified according to the Linux naming convention (`/dev/hda7/`), because this information is read by the kernel and has nothing to do with GRUB. The `initrd` is also located in the first logical partition of the first hard disk.

- The second entry is responsible for loading Windows. Windows is booted from the first partition of the first hard disk (hd0,0). The command chainloader +1 causes GRUB to read and execute the first sector of the specified partition.
- The next entry enables booting from floppy disk without modifying the BIOS settings.
- The boot option failsafe starts Linux with a selection of kernel parameters that enables Linux to boot even on problematic systems.

The menu file can be changed whenever necessary. GRUB then uses the modified settings during the next boot. Edit the file permanently using YaST or an editor of your choice. Alternatively, make temporary changes interactively using the edit function of GRUB. See Section “Editing Menu Entries during the Boot Procedure” (page 434).

Editing Menu Entries during the Boot Procedure

In the graphical GRUB boot menu, select the operating system to boot with the arrow keys. If you select a Linux system, you can enter additional boot parameters at the boot prompt. To edit individual menu entries directly, press Esc to exit the splash screen then press E. Changes made in this way only apply to the current boot and are not adopted permanently.

IMPORTANT: Keyboard Layout during the Boot Procedure

The US keyboard layout is the only one available when booting.

After activating the editing mode, use the arrow keys to select the menu entry of the configuration to edit. To make the configuration editable, press E again. In this way, edit incorrect partitions or path specifications before they have a negative effect on the boot process. Press Enter to exit the editing mode and return to the menu. Then press B to boot this entry. Further possible actions are displayed in the help text at the bottom.

To enter changed boot options permanently and pass them to the kernel, open the file menu.lst as the user root and append the respective kernel parameters to the existing line, separated by spaces:

```
title linux
   kernel (hd0,0)/vmlinuz root=/dev/hda3 additional parameter
   initrd (hd0,0)/initrd
```

GRUB automatically adopts the new parameters the next time the system is booted. Alternatively, this change can also be made with the YaST boot loader module. Append the new parameters to the existing line, separated by spaces.

Using Wild Cards to Select the Boot Kernel

Especially when developing or using custom kernels, you must either change the entries in `menu.lst` or edit the command line to reflect the current kernel and initrd filenames. To simplify this procedure, use *wild cards* to update the kernel list of GRUB dynamically. All kernel images that match a specific pattern are then automatically added to the list of bootable images. Note that there is no support for this feature.

Activate the wild card option by entering an additional menu entry in `menu.lst`. To be useful, all kernel and initrd images must have a common base name and an identifier that matches the kernel with its associated initrd. Consider the following setup:

```
initrd-default
initrd-test
vmlinuz-default
vmlinuz-test
```

In this case, you may add both boot images in one GRUB configuration. To get the menu entries `linux-default` and `linux-test`, the following entry in `menu.lst` would be needed:

```
title linux-*
   wildcard (hd0,4)/vmlinuz-*
   kernel (hd0,4)/vmlinuz-* root=/dev/hda7 vga=791
   initrd (hd0,4)/initrd-*
```

In this example, GRUB searches the partition (hd0,4) for entries matching the wild card. These entries are used to generate new GRUB menu entries. In the previous example, GRUB behaves as if the following entries exist in `menu.lst`:

```
title linux-default
   wildcard (hd0,4)/vmlinuz-default
   kernel (hd0,4)/vmlinuz-default root=/dev/hda7 vga=791
   initrd (hd0,4)/initrd-default
title linux-test
   wildcard (hd0,4)/vmlinuz-test
   kernel (hd0,4)/vmlinuz-test root=/dev/hda7 vga=791
   initrd (hd0,4)/initrd-test
```

Problems with this configuration can be expected if filenames are not used consistently or if one of the expanded files, such as an initrd image, is missing.

29.3.2 The File device.map

The file device.map maps GRUB device names to Linux device names. In a mixed system containing IDE and SCSI hard disks, GRUB must try to determine the boot sequence by a special procedure, because GRUB does not have access to the BIOS information on the boot sequence. GRUB saves the result of this analysis in the file /boot/grub/device.map. For a system on which the boot sequence in the BIOS is set to IDE before SCSI, the file device.map could appear as follows:

```
(fd0)  /dev/fd0
(hd0)  /dev/hda
(hd1)  /dev/sda
```

Because the order of IDE, SCSI, and other hard disks depends on various factors and Linux is not able to identify the mapping, the sequence in the file device.map can be set manually. If you encounter problems when booting, check if the sequence in this file corresponds to the sequence in the BIOS and use the GRUB shell, described in Section 29.3.4, "The GRUB Shell" (page 437), to modify it temporarily if necessary. After the Linux system has booted, the file device.map can be edited permanently with the YaST boot loader module or an editor of your choice.

After manually changing device.map, execute the following command to reinstall GRUB. This command causes the file device.map to be reloaded and the commands listed in grub.conf to be executed:

```
grub --batch < /etc/grub.conf
```

29.3.3 The File /etc/grub.conf

The third important GRUB configuration file apart from menu.lst and device.map is /etc/grub.conf. This file contains the parameters and options the command grub needs for installing the boot loader correctly:

```
root (hd0,4)
   install /grub/stage1 d (hd0) /grub/stage2 0x8000 (hd0,4)/grub/menu.lst
   quit
```

Meaning of the individual entries:

root (hd0,4)
This command tells GRUB to apply the following commands to the first logical partition of the first hard disk (the location of the boot files).

install `parameter`

The command `grub` should be run with the parameter `install`. `stage1` of the boot loader should be installed in the MBR of the first hard disk (`/grub/stage1 d (hd0)`). `stage2` should be loaded to the memory address 0x8000 (`/grub/stage2 0x8000`). The last entry (`(hd0,4)/grub/menu.lst`) tells GRUB where to look for the menu file.

29.3.4 The GRUB Shell

GRUB actually exists in two versions: as a boot loader and as a normal Linux program in `/usr/sbin/grub`. This program is referred to as the *GRUB shell*. The functionality to install GRUB as boot loader on a hard disk or floppy disk is integrated in GRUB in the form of the commands `install` and `setup`. This is available in the GRUB shell when Linux is loaded.

However, the commands `setup` and `install` are also available during the boot procedure before Linux is started. This facilitates the repair of a defective system that can no longer be booted, because the faulty configuration file of the boot loader can be circumvented by manually entering parameters. Manually entering parameters during the boot procedure is also useful for testing new settings without impairing the native system. Simply enter the experimental configuration file with a syntax similar to that in `menu.lst`. Then test the functionality of this entry without changing the existing configuration file. For example, to test a new kernel, enter the command `kernel` and the path to the new kernel. If the boot procedure fails, you can continue using the intact `menu.lst` the next time you boot. Similarly, the command line interface can also be used to boot a system despite a faulty `menu.lst` file by entering the corrected parameters. In the running system, then enter the correct parameters in `menu.lst` to make the system permanently bootable.

The mapping of GRUB devices to Linux device names is only relevant when running the GRUB shell as a Linux program (by entering `grub` as described in Section 29.3.2, “The File device.map” (page 436)). For this purpose, the program reads the file `device.map`. For more information, see Section 29.3.2, “The File device.map” (page 436).

29.3.5 Setting a Boot Password

Even before the operating system is booted, GRUB enables access to file systems. Users without root permissions can access files in your Linux system to which they have no access once the system is booted. To block this kind of access or prevent users from booting certain operating systems, set a boot password.

IMPORTANT: Boot Password and Splash Screen

If you use a boot password for GRUB, the usual splash screen is not displayed.

As the user `root`, proceed as follows to set a boot password:

1. At the root prompt, enter `grub`.

2. Encrypt the password in the GRUB shell:

   ```
   grub> md5crypt
   Password: ****
   Encrypted: $1$lS2dv/$JOYcdxIn7CJk9xShzzJVw/
   ```

3. Paste the encrypted string into the global section of the file `menu.lst`:

   ```
   gfxmenu (hd0,4)/message
   color white/blue black/light-gray
   default 0
   timeout 8
   password --md5 $1$lS2dv/$JOYcdxIn7CJk9xShzzJVw/
   ```

   Now GRUB commands can only be executed at the boot prompt after pressing P and entering the password. However, users can still boot all operating systems from the boot menu.

4. To prevent one or several operating systems from being booted from the boot menu, add the entry `lock` to every section in `menu.lst` that should not be bootable without entering a password. For example:

   ```
   title linux
   kernel (hd0,4)/vmlinuz root=/dev/hda7 vga=791
   initrd (hd0,4)/initrd
   lock
   ```

   After rebooting the system and selecting the Linux entry from the boot menu, the following error message is displayed:

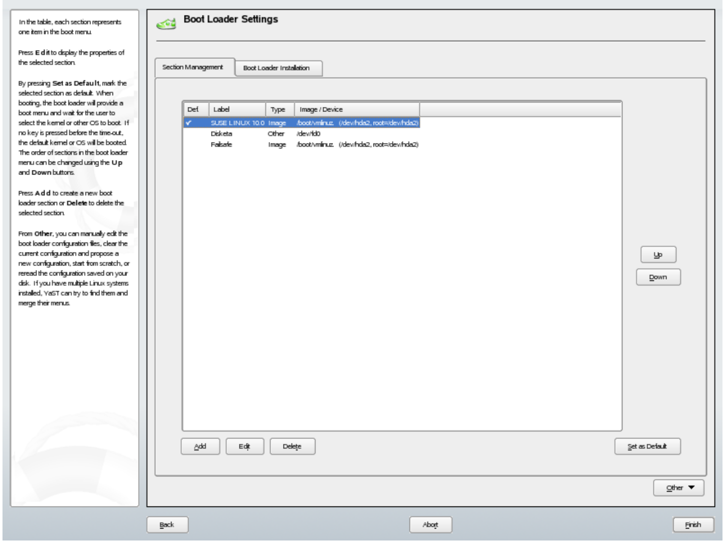
```
Error 32: Must be authenticated
```

Press Enter to enter the menu. Then press P to get a password prompt. After entering the password and pressing Enter, the selected operating system (Linux in this case) should boot.

29.4 Configuring the Boot Loader with YaST

The easiest way to configure the boot loader in your SUSE Linux system is to use the YaST module. In the YaST Control Center, select *System* → *Boot Loader Configuration*. As in Figure 29.1, "Configuring the Boot Loader with YaST" (page 439), this shows the current boot loader configuration of your system and allows you to make changes.

Figure 29.1 *Configuring the Boot Loader with YaST*



In the table, each section represents one item in the boot menu.

Press **Edit** to display the properties of the selected section.

By pressing **Set as Default**, mark the selected section as default. When booting, the boot loader will provide a boot menu and wait for the user to select the kernel or other OS to boot. If no key is pressed before the time-out, the default kernel or OS will be booted. The order of sections in the boot loader menu can be changed using the **Up** and **Down** buttons.

Press **Add** to create a new boot loader section or **Delete** to delete the selected section.

From **Other**, you can manually edit the boot loader configuration files, clear the current configuration and propose a new configuration, start from scratch, or reread the configuration saved on your disk. If you have multiple Linux systems installed, YaST can try to find them and merge their menus.

Boot Loader Settings

Section Management | Boot Loader Installation

Def	Label	Type	Image / Device
✓	SUSE LINUX 10.0	Image	/boot/vmlinuz (/dev/hda2, root=/dev/hda2)
	Disketa	Other	/dev/fd0
	Failsafe	Image	/boot/vmlinuz (/dev/hda2, root=/dev/hda2)

Use the *Section Management* tab to edit, change, and delete boot loader sections for the individual operating systems. To add an option, click *Add*. To change the value of an existing option, select it with the mouse and click *Edit*. If you do not want to use an existing option at all, select it and click *Delete*. If you are not familiar with boot loader options, read Section 29.3, "Booting with GRUB" (page 429) first.

Use the *Boot Loader Installation* tab to view and change settings related to type, location, and advanced loader settings.

29.4.1 Boot Loader Type

Set the boot loader type in *Boot Loader Installation*. The default boot loader in SUSE Linux is GRUB. To use LILO, proceed as follows:

Procedure 29.2 *Changing the Boot Loader Type*

1. Select the *Boot Loader Installation* tab.

2. For *Boot Loader*, select *LILO*.

3. In the dialog box that opens, select one of the following actions:

   Propose New Configuration
   Have YaST propose a new configuration.

   Convert Current Configuration
   Have YaST convert the current configuration. When converting the configuration, some settings may be lost.

   Start New Configuration from Scratch
   Write a custom configuration. This action is not available during the installation of SUSE Linux.

   Read Configuration Saved on Disk
   Load your own `/etc/lilo.conf`. This action is not available during the installation of SUSE Linux.

4. Click *OK* to save the changes

5. Click *Finish* in the main dialog to apply the changes.

During the conversion, the old GRUB configuration is saved to disk. To use it, simply change the boot loader type back to GRUB and choose *Restore Configuration Saved before Conversion*. This action is available only on an installed system.

NOTE: Custom Boot Loader

If you want use a boot loader other than GRUB or LILO, select *Do Not Install Any Boot Loader*. Read the documentation of your boot loader carefully before choosing this option.

29.4.2 Boot Loader Location

To change the location of the boot loader, follow these steps:

Procedure 29.3 *Changing the Boot Loader Location*

1 Select the *Boot Loader Installation* tab then select one of the following options for *Boot Loader Location*:

Master Boot Record of /dev/hdX
This installs the boot loader in the MBR of a disk. This is recommended whenever YaST determines the system can be booted this way. X identifies the hard disk, for example, a, b, c, or d:

```
hda => ide0 master
 hdb => ide0 slave
 hdc => ide1 master
 hdd => ide1 slave
```

Boot Sector of Boot Partition /dev/hdXY
The boot sector of the `/boot` partition. This option is the default if you have several operating systems installed on your hard drive. The Y stands for the partition (1, 2, 3, 4, 5, etc.) as in:

```
/dev/hda1
```

Boot Sector of Root Partition /dev/hdXY
The boot sector of the `/` (root) partition. Also use this option if you have several operating systems installed on your hard drive and want to continue using your old boot manager.

Other
Use this option to specify the location of the boot loader manually.

2 Click *Finish* to apply your changes.

29.4.3 Default System

To change the system that is booted by default, proceed as follows:

Procedure 29.4 *Setting the Default System*

1. Open the *Section Management* tab.

2. Select the desired system from the list.

3. Click *Set as Default*.

4. Click *Finish* to activate these changes.

29.4.4 Boot Loader Time-Out

The boot loader does not boot the default system immediately. During the time-out, you can select the system to boot or write some kernel parameters. To set the boot loader time-out, proceed as follows:

Procedure 29.5 *Changing the Boot Loader Time-Out*

1. Open the *Boot Loader Installation* tab.

2. Click *Boot Loader Options*.

3. Check *Show Boot Menu*.

4. In *Boot Menu*, change the value of *Boot Menu Time-Out* by typing in a new value, clicking the appropriate arrow key with your mouse, or by using the arrow keys on the keyboard.

5. Click *OK*.

6. Click *Finish* to save the changes.

Set for the boot menu should be displayed permanently without timing out by disabling *Continue Booting after a Time-Out*.

29.4.5 Security Settings

Using this YaST module, you can also set a password to protect booting. This gives you an additional level of security.

Procedure 29.6 *Setting a Boot Loader Password*

1. Open the *Boot Loader Installation* tab.

2. Click *Boot Loader Options*.

3. In *Password Protection*, check *Protect Boot Loader with Password* and set your password.

4. Click *OK*.

5. Click *Finish* to save the changes.

29.4.6 Disk Order

If your computer has more than one hard disk, you can specify the boot sequence of the disks to match the BIOS setup of the machine (see Section 29.3.2, “The File device.map” (page 436)). To do so, proceed as follows:

Procedure 29.7 *Setting the Disk Order*

1. Open the *Boot Loader Installation* tab.

2. Click *Boot Loader Installation Details*.

3. If more than one disk is listed, select a disk and click *Up* or *Down* to reorder the displayed disks.

4. Click *OK* to save the changes.

5. Click *Finish* to save the changes.

Using this module, you can also replace the master boot record with generic code, which boots the active partition. Click *Replace MBR with Gerneric Code* in *Disk System Area*

Update. Enable *Activate Boot Loader Partition* to activate the partition that contains the boot loader. Click *Finish* to save the changes.

29.5 Uninstalling the Linux Boot Loader

YaST can be used to uninstall the Linux boot loader and restore the MBR to the state it had prior to the installation of Linux. During the installation, YaST automatically creates a backup copy of the original MBR and restores it on request, overwriting GRUB.

To uninstall GRUB, start the YaST boot loader module (*System* → *Boot Loader Configuration*). In the first dialog, select *Reset* → *Restore MBR of Hard Disk* and exit the dialog with *Finish*. In the MBR, GRUB is overwritten with the data of the original MBR.

29.6 Creating Boot CDs

If problems occur booting your system using a boot manager or if the boot manager cannot be installed on the MBR of your hard disk or a floppy disk, it is also possible to create a bootable CD with all the necessary start-up files for Linux. This requires a CD writer installed in your system.

Creating a bootable CD-ROM with GRUB merely requires a special form of *stage2* called `stage2_eltorito` and, optionally, a customized `menu.lst`. The classic files `stage1` and `stage2` are not required.

Create a directory in which to create the ISO image, for example, with `cd /tmp` and `mkdir iso`. Also create a subdirectory for GRUB with `mkdir -p iso/boot/grub`. Copy the file `stage2_eltorito` into the directory `grub`:

```
cp /usr/lib/grub/stage2_eltorito iso/boot/grub
```

Also copy the kernel (`/boot/vmlinuz`), the `initrd` (`/boot/initrd`), and the file `/boot/message` to `iso/boot/`:

```
cp /boot/vmlinuz iso/boot/
cp /boot/initrd iso/boot/
cp /boot/message iso/boot/
```

To make them available to GRUB, copy the file `menu.lst` to `iso/boot/grub` and adjust the path entries to make them point to a CD-ROM device. Do this by replacing the device name of the hard disks, listed in the format `(hd*)`, in the pathnames with the device name of the CD-ROM drive, which is `(cd)`:

```
gfxmenu (cd)/boot/message
timeout 8
default 0

title Linux
    kernel (cd)/boot/vmlinuz root=/dev/hda5 vga=794 resume=/dev/hda1
splash=verbose showopts
    initrd (cd)/boot/initrd
```

Finally, create the ISO image with the following command:

```
mkisofs -R -b boot/grub/stage2_eltorito -no-emul-boot \
-boot-load-size 4 -boot-info-table -o grub.iso iso
```

Then write the resulting file `grub.iso` to a CD using your preferred utility.

29.7 The Graphical SUSE Screen

Since SUSE Linux 7.2, the graphical SUSE screen is displayed on the first console if the option “vga=<value>” is used as a kernel parameter. If you install using YaST, this option is automatically activated in accordance with the selected resolution and the graphics card. There are three ways to disable the SUSE screen, if desired:

Disabling the SUSE Screen When Necessary
Enter the command `echo 0 >/proc/splash` on the command line to disable the graphical screen. To activate it again, enter `echo 1 >/proc/splash`.

Disabling the SUSE screen by default.
Add the kernel parameter `splash=0` to your boot loader configuration. Chapter 29, *The Boot Loader* (page 427) provides more information about this. However, if you prefer the text mode, which was the default in earlier versions, set `vga=normal`.

Completely Disabling the SUSE Screen

Compile a new kernel and disable the option *Use splash screen instead of boot logo* in *framebuffer support*.

TIP

Disabling framebuffer support in the kernel automatically disables the splash screen as well. SUSE cannot provide any support for your system if you run it with a custom kernel.

29.8 Troubleshooting

This section lists some of the problems frequently encountered when booting with GRUB and a short description of possible solutions. Some of the problems are covered in articles in the Support Database at `http://portal.suse.de/sdb/en/index.html`. If your specific problem is not included in this list, use the search dialog of the Support Database at `https://portal.suse.com/PM/page/search.pm` to search for keywords like *GRUB*, *boot*, and *boot loader*.

GRUB and XFS

XFS leaves no room for `stage1` in the partition boot block. Therefore, do not specify an XFS partition as the location of the boot loader. This problem can be solved by creating a separate boot partition that is not formatted with XFS.

GRUB and JFS

Although technically possible, the combination of GRUB with JFS is problematic. In this case, create a separate boot partition (`/boot`) and format it with Ext2. Install GRUB in this partition.

GRUB Reports GRUB Geom Error

GRUB checks the geometry of connected hard disks when the system is booted. Sometimes, the BIOS returns inconsistent information and GRUB reports a GRUB Geom Error. If this is the case, use LILO or update the BIOS. Detailed information about the installation, configuration, and maintenance of LILO is available in the Support Database under the keyword LILO.

GRUB also returns this error message if Linux was installed on an additional hard disk that is not registered in the BIOS. *stage1* of the boot loader is found and loaded correctly, but *stage2* is not found. This problem can be remedied by registering the new hard disk in the BIOS.

System Containing IDE and SCSI Hard Disks Does Not Boot

During the installation, YaST may have incorrectly determined the boot sequence of the hard disks. For example, GRUB may regard `/dev/hda` as `hd0` and `/dev/sda` as `hd1`, although the boot sequence in the BIOS is reversed (SCSI *before* IDE).

In this case, correct the hard disks during the boot process with the help of the GRUB command line. After the system has booted, edit `device.map` to apply the new mapping permanently. Then check the GRUB device names in the files `/boot/grub/menu.lst` and `/boot/grub/device.map` and reinstall the boot loader with the following command:

```
grub --batch < /etc/grub.conf
```

Booting Windows from the Second Hard Disk

Some operating systems, such as Windows, can only boot from the first hard disk. If such an operating system is installed on a hard disk other than the first hard disk, you can effect a logical change for the respective menu entry.

```
...
title windows
map (hd0) (hd1)
map (hd1) (hd0)
chainloader(hd1,0)+1
...
```

In this example, Windows is started from the second hard disk. For this purpose, the logical order of the hard disks is changed with `map`. This change does not affect the logic within the GRUB menu file. Therefore, the second hard disk must be specified for `chainloader`.

29.9 For More Information

Extensive information about GRUB is available at `http://www.gnu.org/software/grub/`. Also refer to the `grub` info page. You can also search for the keyword “GRUB” in the Support Database at `http://portal.suse.de/sdb/en/index.html` to get information about special issues.

Special Features of SUSE Linux 30

This chapter starts with information about various software packages, the virtual consoles, and the keyboard layout. We talk about software components like `bash`, `cron`, and `logrotate`, because they were changed or enhanced during the last release cycles. Even if they are small or considered of minor importance, users may want to change their default behavior, because these components are often closely coupled with the system. The chapter is finished by a section about language and country-specific settings (I18N and L10N).

30.1 Information about Special Software Packages

The programs `bash`, `cron`, `logrotate`, `locate`, `ulimit`, and `free`, and the file `resolv.conf` are very important for system administrators and many users. Man pages and info pages are two useful sources of information about commands, but both are not always available. GNU Emacs is a popular and very configurable text editor.

30.1.1 The Package bash and /etc/profile

Bash is the default shell in SUSE Linux. When used as a login shell, it reads several initialization files. Bash processes them in the order they appear in this list.

1. `/etc/profile`

2. `~/.profile`

3. `/etc/bash.bashrc`

4. `~/.bashrc`

Custom settings can be made in `~/.profile` or in `~/.bashrc`. To ensure the correct processing of these files, it is necessary to copy the basic settings from `/etc/skel/.profile` or `/etc/skel/.bashrc` into the home directory of the user. It is recommended to copy the settings from `/etc/skel` following an update. Execute the following shell commands to prevent the loss of personal adjustments:

```
mv ~/.bashrc ~/.bashrc.old
cp /etc/skel/.bashrc ~/.bashrc
mv ~/.profile ~/.profile.old
cp /etc/skel/.profile ~/.profile
```

Then copy personal adjustments back from the `*.old` files.

30.1.2 The cron Package

If you want to run commands regularly and automatically in the background at predefined times, cron is the traditional tool to use. cron is driven by specially formatted time tables. Some of of them come with the system and users can write their own tables if needed.

The cron tables are located in `/var/spool/cron/tabs`. `/etc/crontab` serves as a systemwide cron table. Enter the name of the user who should run the command directly after the time table. In Example 30.1, "Entry in /etc/crontab" (page 450), `root` is entered. Package-specific tables, located in `/etc/cron.d`, have the same format. See the `cron` man page (`man cron`).

***Example 30.1** Entry in /etc/crontab*

```
1-59/5 * * * *   root   test -x /usr/sbin/atrun && /usr/sbin/atrun
```

You cannot edit `/etc/crontab` by calling the command `crontab -e`. This file must be loaded directly into an editor, modified, then saved.

A number of packages install shell scripts to the directories `/etc/cron.hourly`, `/etc/cron.daily`, `/etc/cron.weekly`, and `/etc/cron.monthly`, whose instructions are controlled by `/usr/lib/cron/run-crons`. `/usr/lib/cron/`

run-crons is run every 15 minutes from the main table (/etc/crontab). This guarantees that processes that may have been neglected can be run at the proper time.

To run the hourly, daily, or other periodic maintenance scipts at custom times, remove the time stamp files regulary using of /etc/crontab entries (see Example 30.2, "/etc/crontab: Remove Time Stamp Files" (page 451), which removes the hourly one before every full hour, the daily one once a day at 2:14 a.m., etc.).

Example 30.2 */etc/crontab: Remove Time Stamp Files*

```
59 *  * * *     root  rm -f /var/spool/cron/lastrun/cron.hourly
14 2  * * *     root  rm -f /var/spool/cron/lastrun/cron.daily
29 2  * * 6     root  rm -f /var/spool/cron/lastrun/cron.weekly
44 2  1 * *     root  rm -f /var/spool/cron/lastrun/cron.monthly
```

The daily system maintenance jobs have been distributed to various scripts for reasons of clarity. They are contained in the package aaa_base. /etc/cron.daily contains, for example, the components suse.de-backup-rpmdb, suse.de-clean-tmp, or suse.de-cron-local.

30.1.3 Log Files: Package logrotate

There are a number of system services (*daemons*) that, along with the kernel itself, regularly record the system status and specific events to log files. This way, the administrator can regularly check the status of the system at a certain point in time, recognize errors or faulty functions, and troubleshoot them with pinpoint precision. These log files are normally stored in /var/log as specified by FHS and grow on a daily basis. The logrotate package helps control the growth of these files.

Configuration

Configure logrotate with the file /etc/logrotate.conf. In particular, the include specification primarily configures the additional files to read. SUSE Linux ensures that programs that produce log files install individual configation files in /etc/logrotate.d. For example, such programs come with the packages apache2 (/etc/logrotate.d/apache2) and syslogd (/etc/logrotate.d/syslog).

Example 30.3 *Example for /etc/logrotate.conf*

```
# see "man logrotate" for details
# rotate log files weekly
weekly

# keep 4 weeks worth of backlogs
rotate 4

# create new (empty) log files after rotating old ones
create

# uncomment this if you want your log files compressed
#compress

# RPM packages drop log rotation information into this directory
include /etc/logrotate.d

# no packages own lastlog or wtmp - we'll rotate them here
#/var/log/wtmp {
#    monthly
#    create 0664 root utmp
#    rotate 1
#}

# system-specific logs may be also be configured here.
```

logrotate is controlled through cron and is called daily by `/etc/cron.daily/logrotate`.

IMPORTANT

The `create` option reads all settings made by the administrator in `/etc/permissions*`. Ensure that no conflicts arise from any personal modifications.

30.1.4 The Command locate

locate, a command for quickly finding files, is not included in the standard scope of installed software. If desired, install the package `find-locate`. The updatedb process is started automatically every night or about 15 minutes after booting the system.

30.1.5 The Command ulimit

With the `ulimit` (*user limits*) command, it is possible to set limits for the use of system resources and to have these displayed. `ulimit` is especially useful for limiting the memory available for applications. With this, an application can be prevented from using too much memory on its own, which could bring the system to a standstill.

`ulimit` can be used with various options. To limit memory usage, use the options listed in Table 30.1, "`ulimit`: Setting Resources for the User" (page 453).

Table 30.1 *ulimit: Setting Resources for the User*

`-m`	Maximum size of physical memory
`-v`	Maximum size of virtual memory
`-s`	Maximum size of the stack
`-c`	Maximum size of the core files
`-a`	Display of limits set

Systemwide settings can be made in `/etc/profile`. There, enable creation of core files, needed by programmers for *debugging*. A normal user cannot increase the values specified in `/etc/profile` by the system administrator, but can make special entries in `~/.bashrc`.

Example 30.4 *ulimit: Settings in ~/.bashrc*

```
# Limits of physical memory:
ulimit -m 98304

# Limits of virtual memory:
ulimit -v 98304
```

Memory amounts must be specified in KB. For more detailed information, see `man bash`.

IMPORTANT

Not all shells support `ulimit` directives. PAM (for instance, `pam_limits`) offers comprehensive adjustment possibilities if you depend on encompassing settings for these restrictions.

30.1.6 The free Command

The `free` command is somewhat misleading if your goal is to find out how much RAM is currently being used. That information can be found in `/proc/meminfo`. These days, users with access to a modern operating system, such as Linux, should not really need to worry much about memory. The concept of *available RAM* dates back to before the days of unified memory management. The slogan *free memory is bad memory* applies well to Linux. As a result, Linux has always made the effort to balance out caches without actually allowing free or unused memory.

Basically, the kernel does not have direct knowledge of any applications or user data. Instead, it manages applications and user data in a *page cache*. If memory runs short, parts of it are written to the swap partition or to files, from which they can initially be read with the help of the `mmap` command (see `man mmap`).

The kernel also contains other caches, such as the *slab cache*, where the caches used for network access are stored. This may explain differences between the counters in `/proc/meminfo`. Most, but not all of them, can be accessed via `/proc/slabinfo`.

30.1.7 The File /etc/resolv.conf

Domain name resolution is handled through the file `/etc/resolv.conf`. Refer to Chapter 40, *The Domain Name System* (page 593).

This file is updated by the script `/sbin/modify_resolvconf` exclusively, with no other program having permission to modify `/etc/resolv.conf` directly. Enforcing this rule is the only way to guarantee that the system's network configuration and the relevant files are kept in a consistent state.

30.1.8 Man Pages and Info Pages

For some GNU applications (such as tar), the man pages are no longer maintained. For these commands, use the `--help` option to get a quick overview of the info pages, which provide more in-depth instructions. info is GNU's hypertext system. Read an introduction to this system by entering `info info`. Info pages can be viewed with Emacs by entering `emacs -f info` or directly in a console with `info`. You can also use tkinfo, xinfo, or the SUSE help system to view info pages.

30.1.9 Settings for GNU Emacs

GNU Emacs is a complex work environment. The following sections cover the configuration files processed when GNU Emacs is started. More information is available at `http://www.gnu.org/software/emacs/`.

On start-up, Emacs reads several files containing the settings of the user, system administrator, and distributor for customization or preconfiguration. The initialization file `~/.emacs` is installed to the home directories of the individual users from `/etc/skel`. `.emacs`, in turn, reads the file `/etc/skel/.gnu-emacs`. To customize the program, copy `.gnu-emacs` to the home directory (with `cp /etc/skel/.gnu-emacs ~/.gnu-emacs`) and make the desired settings there.

`.gnu-emacs` defines the file `~/.gnu-emacs-custom` as `custom-file`. If users make settings with the `customize` options in Emacs, the settings are saved to `~/.gnu-emacs-custom`.

With SUSE Linux, the `emacs` package installs the file `site-start.el` in the directory `/usr/share/emacs/site-lisp`. The file `site-start.el` is loaded before the initialization file `~/.emacs`. Among other things, `site-start.el` ensures that special configuration files distributed with Emacs add-on packages, such as `psgml`, are loaded automatically. Configuration files of this type are located in `/usr/share/emacs/site-lisp`, too, and always begin with `suse-start-`. The local system administrator can specify systemwide settings in `default.el`.

More information about these files is available in the Emacs info file under *Init File*: `info:/emacs/InitFile`. Information about how to disable loading these files (if necessary) is also provided at this location.

The components of Emacs are divided into several packages:

- The base package emacs.
- emacs-x11 (usually installed): the program *with* X11 support.
- emacs-nox: the program *without* X11 support.
- emacs-info: online documentation in info format.
- emacs-el: the uncompiled library files in Emacs Lisp. These are not required at runtime.
- Numerous add-on packages can be installed if needed: emacs-auctex (for LaTeX), psgml (for SGML and XML), gnuserv (for client and server operation), and others.

30.2 Virtual Consoles

Linux is a multiuser and multitasking system. The advantages of these features can be appreciated even on a stand-alone PC system. In text mode, there are six virtual consoles available. Switch between them using Alt + F1 to Alt + F6. The seventh console is reserved for X and the tenth console shows kernel messages. More or fewer consoles can be assigned by modifying the file /etc/inittab.

To switch to a console from X without shutting it down, use Ctrl + Alt + F1 to Ctrl + Alt + F6. To return to X, press Alt + F7.

30.3 Keyboard Mapping

To standardize the keyboard mapping of programs, changes were made to the following files:

```
/etc/inputrc
/usr/X11R6/lib/X11/Xmodmap
/etc/skel/.Xmodmap
/etc/skel/.exrc
/etc/skel/.less
/etc/skel/.lesskey
/etc/csh.cshrc
```

```
/etc/termcap
/usr/lib/terminfo/x/xterm
/usr/X11R6/lib/X11/app-defaults/XTerm
/usr/share/emacs/<VERSION>/site-lisp/term/*.el
```

These changes only affect applications that use `terminfo` entries or whose configuration files are changed directly (`vi`, `less`, etc.). Applications not shipped with SUSE Linux should be adapted to these defaults.

Under X, the compose key (multikey) can be accessed using Ctrl + Shift (right). Also see the corresponding entry in `/usr/X11R6/lib/X11/Xmodmap`.

Further settings are possible using the X Keyboard Extension (XKB). This extension is also used by the desktop environments GNOME (gswitchit) and KDE (kxkb).

TIP: For More Information

Information about XKB is available in `/etc/X11/xkb/README` and the documents listed there.

Detailed information about the input of Chinese, Japanese, and Korean (CJK) is available at Mike Fabian's page: `http://www.suse.de/~mfabian/suse-cjk/input.html`.

30.4 Language and Country-Specific Settings

SUSE Linux is, to a very large extent, internationalized and can be modified for local needs in a flexible manner. In other words, internationalization (*I18N*) allows specific localizations (*L10N*). The abbreviations I18N and L10N are derived from the first and last letters of the words and, in between, the number of letters omitted.

Settings are made with `LC_` variables defined in the file `/etc/sysconfig/language`. This refers not only to *native language support*, but also to the categories *Messages* (Language), *Character Set*, *Sort Order*, *Time and Date*, *Numbers*, and *Money*. Each of these categories can be defined directly with its own variable or indirectly with a master variable in the file `language` (see the `locale` man page).

RC_LC_MESSAGES, RC_LC_CTYPE, RC_LC_COLLATE, RC_LC_TIME, RC_LC_NUMERIC, RC_LC_MONETARY
These variables are passed to the shell without the `RC_` prefix and represent the listed categories. The shell profiles concerned are listed below. The current setting can be shown with the command `locale`.

RC_LC_ALL
This variable, if set, overwrites the values of the variables already mentioned.

RC_LANG
If none of the previous variables are set, this is the fallback. By default, SUSE Linux only sets `RC_LANG`. This makes it easier for users to enter their own values.

ROOT_USES_LANG
A `yes` or `no` variable. If it is set to `no`, `root` always works in the POSIX environment.

The other variables can be set via the YaST sysconfig editor (see Section 28.3.1, "Changing the System Configuration Using the YaST sysconfig Editor" (page 424)). The value of such a variable contains the language code, country code, encoding, and modifier. The individual components are connected by special characters:

```
LANG=<language>[[_<COUNTRY>].<Encoding>[@<Modifier>]]
```

30.4.1 Some Examples

You should always set the language and country codes together. Language settings follow the standard ISO 639 available at `http://www.evertype.com/standards/iso639/iso639-en.html` and `http://www.loc.gov/standards/iso639-2/`. Country codes are listed in ISO 3166 available at `http://www.din.de/gremien/nas/nabd/iso3166ma/codlstp1/en_listp1.html`.

It only makes sense to set values for which usable description files can be found in `/usr/lib/locale`. Additional description files can be created from the files in `/usr/share/i18n` using the command `localedef`. The description files are part of the `glibc-i18ndata` package. A description file for `en_US.UTF-8` (for English and United States) can be created with:

```
localedef -i en_US -f UTF-8 en_US.UTF-8
```

LANG=en_US.UTF-8
This is the default setting if American English is selected during installation. If you selected another language, that language is enabled but still with UTF-8 as the character encoding.

LANG=en_US.ISO-8859-1
This sets the language to English, country to United States, and the character set to ISO-8859-1. This character set does not support the Euro sign, but it can be useful sometimes for programs that have not been updated to support UTF-8. The string defining the charset (ISO-8859-1 in this case) is then evaluated by programs like Emacs.

LANG=en_IE@euro
The above example explicitly includes the Euro sign in a language setting. Strictly speaking, this setting is obsolete now, because UTF-8 also covers the Euro symbol. It is only useful if an application does not support UTF-8, but ISO-8859-15.

SuSEconfig reads the variables in /etc/sysconfig/language and writes the necessary changes to /etc/SuSEconfig/profile and /etc/SuSEconfig/csh.cshrc. /etc/SuSEconfig/profile is read or *sourced* by /etc/profile. /etc/SuSEconfig/csh.cshrc is sourced by /etc/csh.cshrc. This makes the settings available systemwide.

Users can override the system defaults by editing their ~/.bashrc accordingly. For instance, if you do not want to use the systemwide en_US for program messages, include LC_MESSAGES=es_ES so messages are displayed in Spanish instead.

30.4.2 Settings for Language Support

Files in the category *Messages* are, as a rule, only stored in the corresponding language directory (like en) to have a fallback. If you set LANG to en_US and the message file in /usr/share/locale/en_US/LC_MESSAGES does not exist, it falls back to /usr/share/locale/en/LC_MESSAGES.

A fallback chain can also be defined, for example, for Breton to French or for Galician to Spanish to Portuguese:

```
LANGUAGE="br_FR:fr_FR"
```

```
LANGUAGE="gl_ES:es_ES:pt_PT"
```

If desired, use the Norwegian variants Nynorsk and Bokmål instead (with additional fallback to `no`):

```
LANG="nn_NO"
```

```
LANGUAGE="nn_NO:nb_NO:no"
```

or

```
LANG="nb_NO"
```

```
LANGUAGE="nb_NO:nn_NO:no"
```

Note that in Norwegian, `LC_TIME` is also treated differently.

One problem that can arise is a separator used to delimit groups of digits not being recognized properly. This occurs if `LANG` is set to only a two-letter language code like `de`, but the definition file glibc uses is located in `/usr/share/lib/de_DE/LC_NUMERIC`. Thus `LC_NUMERIC` must be set to `de_DE` to make the separator definition visible to the system.

30.4.3 For More Information

- *The GNU C Library Reference Manual*, Chapter "Locales and Internationalization". It is included in `glibc-info`.
- Markus Kuhn, *UTF-8 and Unicode FAQ for Unix/Linux*, currently at `http://www.cl.cam.ac.uk/~mgk25/unicode.html`.
- *Unicode-Howto*, by Bruno Haible: `/usr/share/doc/howto/en/html/Unicode-HOWTO.html`.

Printer Operation 31

CUPS is the standard print system in SUSE Linux. CUPS is highly user-oriented. In many cases, it is compatible with LPRng or can be adapted with relatively little effort. LPRng is included in SUSE Linux only for reasons of compatibility.

Printers can be distinguished by interface, such as USB or network, and printer language. When buying a printer, make sure that the printer has an interface that is supported by the hardware and a suitable printer language. Printers can be categorized on the basis of the following three classes of printer languages:

PostScript Printers

PostScript is the printer language in which most print jobs in Linux and Unix are generated and processed by the internal print system. This language is already quite old and very efficient. If PostScript documents can be processed directly by the printer and do not need to be converted in additional stages in the print system, the number of potential error sources is reduced. Because PostScript printers are subject to substantial license costs, these printers usually cost more than printers without a PostScript interpreter.

Standard Printer (languages like PCL and ESC/P)

Although these printer languages are quite old, they are still undergoing expansion to address new features in printers. In the case of known printer languages, the print system can convert PostScript jobs to the respective printer language with the help of Ghostscript. This processing stage is referred to as interpreting. The best-known languages are PCL, which is mostly used by HP printers and their clones, and ESC/P, which is used by Epson printers. These printer languages are usually supported by Linux and produce a decent print result. Linux may not be able to address some functions of extremely new and fancy printers, because the open source developers may still be working on these features. Except for the `hpijs` drivers developed by HP, there are currently no printer manufacturers who develop Linux drivers and make them available to Linux distributors under an open source license. Most of these printers are in the medium price range.

Proprietary Printers (usually GDI printers)

Usually only one or several Windows drivers are available for proprietary printers. These printers do not support any of the common printer languages and the printer languages they use are subject to change when a new edition of a model is released. See Section 31.7.1, “Printers without Standard Printer Language Support” (page 477) for more information.

Before you buy a new printer, refer to the following sources to check how well the printer you intend to buy is supported:

- `http://cdb.suse.de/`—the SUSE Linux printer database
- `http://www.linuxprinting.org/`—the LinuxPrinting.org printer database
- `http://www.cs.wisc.edu/~ghost/`—the Ghostscript Web page
- `/usr/share/doc/packages/ghostscript/catalog.devices`—list of included drivers

The online databases always show the latest Linux support status. However, a Linux distribution can only integrate the drivers available at production time. Accordingly, a printer currently rated as “perfectly supported” may not have had this status when the latest SUSE Linux version was released. Thus, the databases may not necessarily indicate the correct status, but only provide an approximation.

31.1 Workflow of the Printing System

The user creates a print job. The print job consists of the data to print plus information for the spooler, such as the name of the printer or the name of the printer queue, and, optionally, the information for the filter, such as printer-specific options.

A dedicated printer queue exists for every printer. The spooler holds the print job in the queue until the desired printer is ready to receive data. When the printer is ready, the spooler sends the data through the filter and back-end to the printer.

The filter converts the data the user wants to print (ASCII, PostScript, PDF, JPEG, etc.) into printer-specific data (PostScript, PCL, ESC/P, etc.). The features of the printer are described in the PPD files. A PPD file contains printer-specific options with the parameters needed to enable them on the printer. The filter system makes sure that options selected by the user are enabled.

If you use a PostScript printer, the filter system converts the data into printer-specific PostScript. This does not require a printer driver. If you use a non-PostScript printer, the filter system converts the data into printer-specific data using Ghostscript. This requires a Ghostscript printer driver suitable for your printer. The back-end receives the printer-specific data from the filter passes it to the printer.

31.2 Methods and Protocols for Connecting Printers

There are various possibilities for connecting a printer to the system. The configuration of the CUPS print system does not distinguish between a local printer and a printer connected to the system over the network. In Linux, local printers must be connected as described in the manual of the printer manufacturer. CUPS supports serial, USB, parallel, and SCSI connections. For more information about the printer connection, read the article *CUPS in a Nutshell* in the Support Database at `http://portal.suse.com`. Find the article by entering *cups* in the search dialog.

WARNING: Cable Connection to the Machine

When connecting the printer to the machine, do not forget that only USB devices can be plugged in or unplugged during operation. The system should be shut down before changing other kinds of connections.

31.3 Installing the Software

PPD (PostScript printer description) is the computer language that describes the properties, like resolution, and options, such as the availability of a duplex unit. These descriptions are required for using various printer options in CUPS. Without a PPD file, the print data would be forwarded to the printer in a "raw" state, which is usually not desired. During the installation of SUSE Linux, many PPD files are preinstalled to enable even printers without PostScript support to be used.

To configure a PostScript printer, the best approach is to get a suitable PPD file. Many PPD files are available in the package `manufacturer-PPDs`, which is automatically installed within the scope of the standard installation. See Section 31.6.3, "PPD Files in Various Packages" (page 474) and Section 31.7.2, "No Suitable PPD File Available for a PostScript Printer" (page 477).

New PPD files can be stored in the directory `/usr/share/cups/model/` or added to the print system with YaST (see Section "Manual Configuration" (page 465)). Subsequently, the PPD file can be selected during the installation.

Be careful if a printer manufacturer wants you to install entire software packages in addition to modifying configuration files. First, this kind of installation would result in the loss of the support provided by SUSE Linux and, second, print commands may work differently and the system may no longer be able to address devices of other manufacturers. For this reason, the installation of manufacturer software is not recommended.

31.4 Configuring the Printer

After connecting the printer to the computer and installing the software, install the printer in the system. This should be done with the tools delivered with SUSE Linux. Because SUSE Linux puts great emphasis on security, third-party tools often have difficulties with the security restrictions and cause more complications than benefits. See Section 31.6.1, "CUPS Server and Firewall" (page 471) and Section 31.6.2, "Changes in the CUPS Print Service" (page 472) for more information about troubleshooting.

31.4.1 Local Printers

If an unconfigured local printer is detected when you log in, YaST starts for configuring it. This uses the same dialogs as the following description of configuration.

To configure the printer, select *Hardware* → *Printer* in the YaST control center. This opens the main printer configuration window, where the detected devices are listed in the upper part. The lower part lists any queues configured so far. If your printer was not detected, configure it manually.

IMPORTANT

If the *Printer* entry is not available in the YaST control center, the `yast2-printer` package probably is not installed. To solve this problem, install the `yast2-printer` package and restart YaST.

Automatic Configuration

YaST is able to configure the printer automatically if the parallel or USB port can be set up automatically and the connected printer can be detected. The printer database must also contain the ID string of the printer that YaST retrieves during the automatic hardware detection. If the hardware ID differs from the model designation, select the model manually.

To make sure that everything works properly, each configuration should be checked with the print test function of YaST. The test page also provides important information about the configuration tested.

Manual Configuration

If the requirements for automatic configuration are not met or if you want a custom setup, configure the printer manually. Depending on how successful the autodetection is and how much information about the printer model is found in the database, YaST may be able to determine the right settings automatically or at least make a reasonable preselection.

The following parameters must be configured:

Hardware Connection (Port)

The configuration of the hardware connection depends on whether YaST has been able to find the printer during hardware autodetection. If YaST is able to detect the printer model automatically, it can be assumed that the printer connection works on the hardware level and no settings need to be changed in this respect. If YaST is unable to autodetect the printer model, there may be some problem with the connection on the hardware level. In this case, some manual intervention is required to configure the connection.

In the *Printer Configuration* dialog, press *Add* to start the manual configuration workflow. Here, select your *Printer Type* (for example `USB printer`) and, with *Next*, enter the *Printer Connection* and select the device.

Name of the Queue

The queue name is used when issuing print commands. The name should be relatively short and consist of lowercase letters and numbers only. Enter the *Name for printing* in the next dialog (*Queue name*).

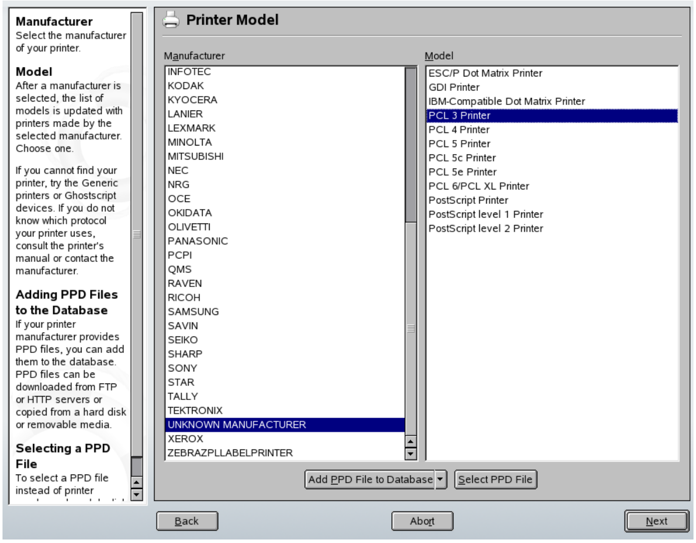
Printer Model and PPD File

All printer-specific parameters, such as the Ghostscript driver to use and the printer filter parameters for the driver, are stored in a PPD (PostScript Printer Description) file. See Section 31.3, “Installing the Software” (page 464) for more information about PPD files.

For many printer models, several PPD files are available, for example, if several Ghostscript drivers work with the given model. When you select a manufacturer and a model in the next dialog (*Printer model*), YaST selects the PPD file that corresponds to the printer. If several PPD files are available for the model, YaST defaults to one of them (normally the one marked `recommended`). You can change the chosen PPD file in the next dialog with *Edit*.

For non-PostScript models, all printer-specific data is produced by the Ghostscript driver. For this reason, the driver configuration is the single most important factor determining the output quality. The printout is affected both by the kind of Ghostscript driver (PPD file) selected and the options specified for it. If necessary, change additional options (as made available by the PPD file) after selecting *Edit*.

Figure 31.1 Selecting the Printer Model



Manufacturer
Select the manufacturer of your printer.

Model
After a manufacturer is selected, the list of models is updated with printers made by the selected manufacturer. Choose one.

If you cannot find your printer, try the Generic printers or Ghostscript devices. If you do not know which protocol your printer uses, consult the printer's manual or contact the manufacturer.

Adding PPD Files to the Database
If your printer manufacturer provides PPD files, you can add them to the database. PPD files can be downloaded from FTP or HTTP servers or copied from a hard disk or removable media.

Selecting a PPD File
To select a PPD file instead of printer

Printer Model

Manufacturer

INFOTEC
KODAK
KYOCERA
LANIER
LEXMARK
MINOLTA
MITSUBISHI
NEC
NRG
OCE
OKIDATA
OLIVETTI
PANASONIC
PCPI
QMS
RAVEN
RICOH
SAMSUNG
SAVIN
SEIKO
SHARP
SONY
STAR
TALLY
TEKTRONIX
UNKNOWN MANUFACTURER
XEROX
ZEBRAZPLLABELPRINTER

Model

ESC/P Dot Matrix Printer
GDI Printer
IBM-Compatible Dot Matrix Printer
PCL 3 Printer
PCL 4 Printer
PCL 5 Printer
PCL 5c Printer
PCL 5e Printer
PCL 6/PCL XL Printer
PostScript Printer
PostScript level 1 Printer
PostScript level 2 Printer

Always check whether your settings work as expected by printing the test page. If the output is garbled, for example, with several pages almost empty, you should be able to stop the printer by first removing all paper then stopping the test from YaST.

If the printer database does not include an entry for your model, you can either add a new PPD file by selecting *Add PPD File to Database*, or use a collection of generic PPD files to make the printer work with one of the standard printer languages. To do so, select *UNKNOWN MANUFACTURER* as your printer manufacturer.

Advanced Settings
Normally, you do not need to change any of these settings.

31.4.2 Network Printers

A network printer can support various protocols, some of them even concurrently. Although most of the supported protocols are standardized, some manufacturers expand (modify) the standard because they test systems that have not implemented the standard correctly or because they want to provide certain functions that are not available in the standard. Manufacturers then provide drivers for only a few operating systems, eliminating difficulties with those systems. Unfortunately, Linux drivers are rarely provided.

The current situation is such that you cannot act on the assumption that every protocol works smoothly in Linux. Therefore, you may have to experiment with various options to achieve a functional configuration.

CUPS supports the `socket`, `LPD`, `IPP`, and `smb` protocols. Here is some detailed information about these protocols:

socket
Socket refers to a connection in which the data is sent to an Internet socket without first performing a data handshake. Some of the socket port numbers that are commonly used are `9100` or `35`. An example device URI is `socket://host-printer:9100/`.

LPD (line printer daemon)
The proven LPD protocol is described in RFC 1179. Under this protocol, some job-related data, such as the ID of the printer queue, is sent before the actual print data is sent. Therefore, a printer queue must be specified when configuring the LPD protocol for the data transmission. The implementations of diverse printer manufacturers are flexible enough to accept any name as the printer queue. If necessary, the printer manual should indicate what name to use. LPT, LPT1, LP1, or similar names are often used. An LPD queue can also be configured on a different Linux or Unix host in the CUPS system. The port number for an LPD service is `515`. An example device URI is `lpd://host-printer/LPT1`.

IPP (Internet printing protocol)
IPP is a relatively new (1999) protocol based on the HTTP protocol. With IPP, more job-related data is transmitted than with the other protocols. CUPS uses IPP for internal data transmission. This is the preferred protocol for a forwarding queue between two CUPS servers. The name of the print queue is necessary to configure IPP correctly. The port number for IPP is `631`. Example device URIs are `ipp://host-printer/ps` and `ipp://host-cupsserver/printers/ps`.

SMB (Windows share)
CUPS also supports printing on printers connected to Windows shares. The protocol used for this purpose is SMB. SMB uses the port numbers `137`, `138`, and `139`. Example device URIs are `smb://user:password@workgroup/server/printer`, `smb://user:password@host/printer`, and `smb://server/printer`.

The protocol supported by the printer must be determined before configuration. If the manufacturer does not provide the needed information, the command nmap, which comes with the nmap package, can be used to guess the protocol. nmap checks a host for open ports. For example:

```
nmap -p 35,137-139,515,631,9100-10000 printerIP
```

Configuring CUPS in the Network Using YaST

Network printers should be configured with YaST. YaST facilitates the configuration and is best equipped to handle the security restrictions in CUPS (see Section 31.6.2, "Changes in the CUPS Print Service" (page 472)). For guidelines for installation of CUPS in the network, read the article *CUPS in a Nutshell* in the Support Database at http://portal.suse.com.

Start the printer configuration then click *Add*. If not told otherwise by the network adminstrator try the option *Print Directly to a Network Printer* and proceed according to your local requirements.

Configuring with Command Line Tools

Alternatively, CUPS can be configured with command-line tools like lpadmin and lpoptions. You need a device URI (uniform resource identifier) consisting of a back-end, such as usb, and parameters, like /dev/usb/lp0. For example, the full URI could be parallel:/dev/lp0 (printer connected to the first parallel port) or usb:/dev/usb/lp0 (first detected printer connected to the USB port).

With lpadmin, the CUPS server administrator can add, remove, or manage class and print queues. To add a printer queue use the following syntax:

```
lpadmin -p queue -v device-URI \
-P PPD-file -E
```

Then the device (-v) will be available as *queue* (-p), using the specified PPD file (-P). This means that you must know the PPD file and the name of the device if you want to configure the printer manually.

Do not use -E as the first option. For all CUPS commands, -E as the first argument sets use of an encrypted connection. To enable the printer, -E must be used as shown in the following example:

```
lpadmin -p ps -v parallel:/dev/lp0 -P \
/usr/share/cups/model/Postscript.ppd.gz -E
```

The following example configures a network printer:

```
lpadmin -p ps -v socket://192.168.1.0:9100/ -P \
/usr/share/cups/model/Postscript-level1.ppd.gz -E
```

For more options of `lpadmin`, see the `lpadmin(1)` man page.

During system installation, certain options are set as default. These options can be modified for every print job (depending on the print tool used). Changing these default options with YaST is also possible. Using command line tools, set default options as follows:

1. First, list all options:

   ```
   lpoptions -p queue -l
   ```

   Example:

   ```
   Resolution/Output Resolution: 150dpi *300dpi 600dpi
   ```

   The activated default option is evident from the preceding asterisk (`*`).

2. Change the option with `lpadmin`:

   ```
   lpadmin -p queue -o Resolution=600dpi
   ```

3. Check the new setting:

   ```
   lpoptions -p queue -l
   ```

   ```
   Resolution/Output Resolution: 150dpi 300dpi *600dpi
   ```

When a normal user runs `lpoptions`, settings are written to `~/.lpoptions.root` settings are written to `/etc/cups/lpoptions`.

31.5 Configuration for Applications

Applications rely on the existing printer queues in the same way as command line tools do. There is usually no need to reconfigure the printer for a particular application, because you should be able to print from applications using the available queues.

To print from the command line, enter `lp -d queuename filename`, substituting the corresponding names for *`queuename`* and *`filename`*.

Some applications rely on the `lp` command for printing. In this case, enter the correct command in the application's print dialog, usually without specifying *`filename`*, for example, `lp -d queuename`. To make this work with KDE programs, enable *Print through an external program*. Otherwise you cannot enter the print command.

Tools such as xpp and the KDE program kprinter provide a graphical interface for choosing among queues and setting both CUPS standard options and printer-specific options made available through the PPD file. You can use kprinter as the standard printing interface of non-KDE applications by specifying `kprinter` or `kprinter --stdin` as the print command in the print dialogs of these applications. The behavior of the application itself determines which of these two commands to choose. If set up correctly, the application should call the kprinter dialog whenever a print job is issued from it, so you can use the dialog to select a queue and set other printing options. This requires that the application's own print setup does not conflict with that of kprinter and that printing options are only changed through kprinter after it has been enabled.

31.6 Special Features in SUSE Linux

A number of CUPS features have been adapted for SUSE Linux. Some of the most important changes are covered here.

31.6.1 CUPS Server and Firewall

There are several ways to configure CUPS as the client of a network server.

1. For every queue on the network server, you can configure a local queue through which to forward all jobs to the corresponding network server. Usually, this approach is not recommended, because all client machines must be reconfigured whenever the configuration of the network server changes.

2. Print jobs can also be forwarded directly to one network server. For this type of configuration, do not run a local CUPS daemon. `lp` or corresponding library

calls of other programs can send jobs directly to the network server. However, this configuration does not work if you also want to print on a local printer.

3. The CUPS daemon can listen to IPP broadcast packets that other network servers send to announce available queues. To use this method, port 631/UDP must be open for incoming packets.

   This is the best CUPS configuration for printing over remote CUPS servers. However, there is a risk that an attacker sends IPP broadcasts with queues and the local daemon accesses a counterfeit queue. If it then displays the queue with the same name as another queue on the local server, the owner of the job may believe the job is sent to a local server, while in reality it is sent to the attacker's server.

YaST can find CUPS servers by scanning all network hosts to see if they offer this service and by listening to IPP broadcasts. The second method is used during the system installation to find CUPS servers for the proposal. It requires that port 631/UDP be open for incoming packets. Opening a port to configure access to remote queues using the second method can be a security risk because an attacker could broadcast a server that might be accepted by users.

The default setting of the firewall shown in the proposal dialog is to reject IPP broadcasts on any interface. Accordingly, the second method for detecting remote queues and the third method for accessing remote queues cannot work. Therefore, the firewall configuration must be modified by marking one of the interfaces as `internal`, which opens the port by default, or by explicitly opening the port of an `external` interface. For security reasons, no ports are open by default.

The proposed firewall configuration must be modified to enable CUPS to detect remote queues during installation and access remote servers from the local system during normal operation. Alternatively, the user can detect CUPS servers by actively scanning the local network hosts or configure all queues manually. However, because of the reasons mentioned in the beginning of this section, this method is not recommended.

31.6.2 Changes in the CUPS Print Service

These changes were initially applied for SUSE Linux 9.1.

cupsd Runs as the User lp

On start-up, cupsd changes from the user root to the user lp. This provides a much higher level of security, because the CUPS print service does not run with unrestricted permissions, only with the permissions needed for the print service.

However, the authentication (the password check) cannot be performed via /etc/shadow, because lp has no access to /etc/shadow. Instead, the CUPS-specific authentication via /etc/cups/passwd.md5 must be used. For this purpose, a CUPS administrator with the CUPS administration group sys and a CUPS password must be entered in /etc/cups/passwd.md5. To do this, enter the following as root:

```
lppasswd -g sys -a CUPS-admin-name
```

This setting is also essential if you want to use the administration Web front-end (CUPS) or the printer administration tool (KDE).

When cupsd runs as lp, /etc/printcap cannot be generated, because lp is not permitted to create files in /etc/. Therefore, cupsd generates /etc/cups/printcap. To ensure that applications that can only read queue names from /etc/printcap continue to work properly, /etc/printcap is a symbolic link pointing to /etc/cups/printcap.

When cupsd runs as lp, port 631 cannot be opened. Therefore, cupsd cannot be reloaded with rccups reload. Use rccups restart instead.

Generalized Functionality for BrowseAllow and BrowseDeny

The access permissions set for BrowseAllow and BrowseDeny apply to all kinds of packages sent to cupsd. The default settings in /etc/cups/cupsd.conf are as follows:

```
BrowseAllow @LOCAL
BrowseDeny All
```

and

```
<Location />
  Order Deny,Allow
  Deny From All
  Allow From 127.0.0.1
  Allow From 127.0.0.2
  Allow From @LOCAL
</Location>
```

In this way, only `LOCAL` hosts can access `cupsd` on a CUPS server. `LOCAL` hosts are hosts whose IP addresses belong to a non-PPP interface (interfaces whose `IFF_POINTOPOINT` flags are not set) and whose IP addresses belong to the same network as the CUPS server. Packets from all other hosts are rejected immediately.

`cupsd` Activated by Default

In a standard installation, `cupsd` is activated automatically, enabling comfortable access to the queues of CUPS network servers without any additional manual actions. The items in Section "cupsd Runs as the User lp" (page 473) and Section "Generalized Functionality for `BrowseAllow` and `BrowseDeny`" (page 473) are vital preconditions for this feature, because otherwise the security would not be sufficient for an automatic activation of `cupsd`.

31.6.3 PPD Files in Various Packages

The YaST printer configuration sets up the queues for CUPS using only the PPD files installed in `/usr/share/cups/model/` on the system. To find the suitable PPD files for the printer model, YaST compares the vendor and model determined during hardware detection with the vendors and models in all PPD files available in `/usr/share/cups/model/` on the system. For this purpose, the YaST printer configuration generates a database from the vendor and model information extracted from the PPD files. When you select a printer from the list of vendors and models, receive the PPD files matching the vendor and model.

The configuration using only PPD files and no other information sources has the advantage that the PPD files in `/usr/share/cups/model/` can be modified freely. The YaST printer configuration recognizes changes and regenerates the vendor and model database. For example, if you only have PostScript printers, normally you do not need the Foomatic PPD files in the `cups-drivers` package or the Gimp-Print PPD files in the `cups-drivers-stp` package. Instead, the PPD files for your PostScript printers can be copied directly to `/usr/share/cups/model/` (if they do not already exist in the `manufacturer-PPDs` package) to achieve an optimum configuration for your printers.

CUPS PPD Files in the cups Package

The generic PPD files in the cups package have been complemented with adapted Foomatic PPD files for PostScript level 1 and level 2 printers:

- /usr/share/cups/model/Postscript-level1.ppd.gz
- /usr/share/cups/model/Postscript-level2.ppd.gz

PPD Files in the cups-drivers Package

Normally, the Foomatic printer filter foomatic-rip is used together with Ghostscript for non-PostScript printers. Suitable Foomatic PPD files have the entries *NickName: ... Foomatic/Ghostscript driver and *cupsFilter: ... foomatic-rip. These PPD files are located in the cups-drivers package.

YaST prefers a Foomatic PPD file if a Foomatic PPD file with the entry *NickName: ... Foomatic ... (recommended) matches the printer model and the manufacturer-PPDs package does not contain a more suitable PPD file.

Gimp-Print PPD Files in the cups-drivers-stp Package

Instead of foomatic-rip, the CUPS filter rastertoprinter from Gimp-Print can be used for many non-PostScript printers. This filter and suitable Gimp-Print PPD files are available in the cups-drivers-stp package. The Gimp-Print PPD files are located in /usr/share/cups/model/stp/ and have the entries *NickName: ... CUPS+Gimp-Print and *cupsFilter: ... rastertoprinter.

PPD Files from Printer Manufacturers in the manufacturer-PPDs Package

The manufacturer-PPDs package contains PPD files from printer manufacturers that are released under a sufficiently liberal license. PostScript printers should be configured with the suitable PPD file of the printer manufacturer, because this file enables the use of all functions of the PostScript printer. YaST prefers a PPD file from the manufacturer-PPDs package if the following conditions are met:

- The vendor and model determined during the hardware detection match the vendor and model in a PPD file from the manufacturer-PPDs package.

- The PPD file from the manufacturer-PPDs package is the only suitable PPD file for the printer model or a there is a Foomatic PPD file with a *NickName: ... Foomatic/Postscript (recommended) entry that also matches the printer model.

Accordingly, YaST does not use any PPD file from the manufacturer-PPDs package in the following cases:

- The PPD file from the the manufacturer-PPDs package does not match the vendor and model. This may happen if the manufacturer-PPDs package contains only one PPD file for similar models, for example, if there is no separate PPD file for the individual models of a model series, but the model name is specified in a form like Funprinter 1000 series in the PPD file.

- The Foomatic PostScript PPD file is not recommended. This may be because the printer model does not operate efficiently enough in PostScript mode, for example, the printer may be unreliable in this mode because it has too little memory or the printer is too slow because its processor is too weak. Furthermore, the printer may not support PostScript by default, for example, because PostScript support is only available as an optional module.

If a PPD file from the manufacturer-PPDs package is suitable for a PostScript printer, but YaST cannot configure it for these reasons, select the respective printer model manually in YaST.

31.7 Troubleshooting

The following sections cover some of the most frequently encountered printer hardware and software problems and ways to solve or circumvent these problems.

31.7.1 Printers without Standard Printer Language Support

Printers that do not support any common printer language and can only be addressed with special control sequences are called *GDI printers*. These printers only work with the operating system versions for which the manufacturer delivers a driver. *GDI* is a programming interface developed by Microsoft for graphics devices. The actual problem is not the programming interface, but the fact that GDI printers can only be addressed with the proprietary printer language of the respective printer model.

Some printers can be switched to operate either in GDI mode or one of the standard printer languages. Some manufacturers provide proprietary drivers for their GDI printers. The disadvantage of proprietary printer drivers is that there is no guarantee that these work with the installed print system and that they are suitable for the various hardware platforms. In contrast, printers that support a standard printer language do not depend on a special print system version or a special hardware platform.

Instead of spending time trying to make a proprietary Linux driver work, it may be more cost-effective to purchase a supported printer. This would solve the driver problem once and for all, eliminating the need to install and configure special driver software and obtain driver updates that may be required due to new developments in the print system.

31.7.2 No Suitable PPD File Available for a PostScript Printer

If the `manufacturer-PPDs` package does not contain any suitable PPD file for a PostScript printer, it should be possible to use the PPD file from the driver CD of the printer manufacturer or download a suitable PPD file from the Web page of the printer manufacturer.

If the PPD file is provided as a zip archive (.zip) or a self-extracting zip archive (`.exe`), unpack it with `unzip`. First, review the license terms of the PPD file. Then use the `cupstestppd` utility to check if the PPD file complies with “Adobe PostScript Printer Description File Format Specification, version 4.3.” If the utility returns “FAIL,” the errors in the PPD files are serious and are likely to cause major problems. The problem spots reported by cupstestppd should be eliminated. If necessary, ask the printer manufacturer for a suitable PPD file.

31.7.3 Parallel Ports

The safest approach is to connect the printer directly to the first parallel port and to select the following parallel port settings in the BIOS:

- I/O address: 378 (hexadecimal)
- Interrupt: irrelevant
- Mode: Normal, SPP, or Output Only
- DMA: disabled

If the printer cannot be addressed on the parallel port despite these settings, enter the I/O address explicitly in accordance with the setting in the BIOS in the form 0x378 in /etc/modprobe.conf. If there are two parallel ports that are set to the I/O addresses 378 and 278 (hexadecimal), enter these in the form 0x378,0x278.

If interrupt 7 is free, it can be activated with the entry shown in Example 31.1, "/etc/modprobe.conf: Interrupt Mode for the First Parallel Port" (page 478). Before activating the interrupt mode, check the file /proc/interrupts to see which interrupts are already in use. Only the interrupts currently being used are displayed. This may change depending on which hardware components are active. The interrupt for the parallel port must not be used by any other device. If you are not sure, use the polling mode with irq=none.

Example 31.1 */etc/modprobe.conf: Interrupt Mode for the First Parallel Port*

```
alias parport_lowlevel parport_pc
options parport_pc io=0x378 irq=7
```

31.7.4 Network Printer Connections

Identifying Network Problems

Connect the printer directly to the computer. For test purposes, configure the printer as a local printer. If this works, the problems are related to the network.

Checking the TCP/IP Network

The TCP/IP network and name resolution must be functional.

Checking a Remote `lpd`

Use the following command to test if a TCP connection can be established to `lpd` (port `515`) on *`host`*:

```
netcat -z host 515 && echo ok || echo failed
```

If the connection to `lpd` cannot be established, `lpd` may not be active or there may be basic network problems.

As the user `root`, use the following command to query a (possibly very long) status report for *`queue`* on remote *`host`*, provided the respective `lpd` is active and the host accepts queries:

```
echo -e "\004queue" \
  | netcat -w 2 -p 722 host 515
```

If `lpd` does not respond, it may not be active or there may be basic network problems. If `lpd` responds, the response should show why printing is not possible on the `queue` on `host`. If you receive a response like that in Example 31.2, "Error Message from the `lpd`" (page 479), the problem is caused by the remote `lpd`.

Example 31.2 *Error Message from the lpd*

```
lpd: your host does not have line printer access
lpd: queue does not exist
printer: spooling disabled
printer: printing disabled
```

Checking a Remote `cupsd`

By default, the CUPS network server should broadcast its queues every 30 seconds on UDP port `631`. Accordingly, the following command can be used to test whether there is a CUPS network server in the network.

```
netcat -u -l -p 631 & PID=$! ; sleep 40 ; kill $PID
```

If a broadcasting CUPS network server exists, the output appears as shown in Example 31.3, "Broadcast from the CUPS Network Server" (page 479).

Example 31.3 *Broadcast from the CUPS Network Server*

```
ipp://host.domain:631/printers/queue
```

The following command can be used to test if a TCP connection can be established to `cupsd` (port `631`) on *`host`*:

```
netcat -z host 631 && echo ok || echo failed
```

If the connection to `cupsd` cannot be established, `cupsd` may not be active or there may be basic network problems. `lpstat -h host -l -t` returns a (possibly very long) status report for all queues on *`host`*, provided the respective `cupsd` is active and the host accepts queries.

The next command can be used to test if the *`queue`* on *`host`* accepts a print job consisting of a single carriage-return character. Nothing should be printed. Possibly, a blank page may be ejected.

```
echo -en "\r" \
  | lp -d queue -h host
```

Troubleshooting a Network Printer or Print Server Box

Spoolers running in a print server box sometimes cause problems when they have to deal with a lot of print jobs. Because this is caused by the spooler in the print server box, there is nothing you can do about it. As a work-around, circumvent the spooler in the print server box by addressing the printer connected to the print server box directly via TCP socket. See Section 31.4.2, “Network Printers” (page 467).

In this way, the print server box is reduced to a converter between the various forms of data transfer (TCP/IP network and local printer connection). To use this method, you need to know the TCP port on the print server box. If the printer is connected to the print server box and powered on, this TCP port can usually be determined with the `nmap` utility from the `nmap` package some time after the print server box is powered on. For example, `nmap` *`IP-address`* may deliver the following output for a print server box:

```
Port        State       Service
23/tcp      open        telnet
80/tcp      open        http
515/tcp     open        printer
631/tcp     open        cups
9100/tcp    open        jetdirect
```

This output indicates that the printer connected to the print server box can be addressed via TCP socket on port `9100`. By default, `nmap` only checks a number of commonly known ports listed in `/usr/share/nmap/nmap-services`. To check all possible ports, use the command `nmap`

`-p` *`from_port-to_port IP-address`*. This may take some time. For further information, refer to the `nmap` man page.

Enter a command like

```
echo -en "\rHello\r\f" | netcat -w 1 IP-address port
cat file | netcat -w 1 IP-address port
```

to send character strings or files directly to the respective port to test if the printer can be addressed on this port.

31.7.5 Defective Printouts without Error Message

For the print system, the print job is completed when the CUPS back-end completes the data transfer to the recipient (printer). If the further processing on the recipient fails, for example, if the printer is not able to print the printer-specific data, the print system does not notice this. If the printer is not able to print the printer-specific data, select a different PPD file that is more suitable for the printer.

31.7.6 Disabled Queues

If the data transfer to the recipient fails entirely after several attempts, the CUPS back-end, such as `usb` or `socket`, reports an error to the print system (to `cupsd`). The back-end decides whether and how many attempts make sense until the data transfer is reported as impossible. Because further attempts would be in vain, `cupsd` disables printing for the respective queue. After eliminating the cause of the problem, the system administrator must reenable printing with the command `/usr/bin/enable`.

31.7.7 CUPS Browsing: Deleting Print Jobs

If a CUPS network server broadcasts its queues to the client hosts via browsing and a suitable local `cupsd` is active on the client hosts, the client `cupsd` accepts print jobs from applications and forwards them to the `cupsd` on the server. When `cupsd` accepts a print job, it is assigned a new job number. Therefore, the job number on the client host is different from the job number on the server. Because a print job is usually forwarded immediately, it cannot be deleted with the job number on the client host, because the client `cupsd` regards the print job as completed as soon as it has been forwarded to the server `cupsd`.

To delete the print job on the server, use a command such as `lpstat -h print-server -o` to determine the job number on the server, provided the server has not already completed the print job (that is, sent it to the printer). Using this job number, the print job on the server can be deleted:

```
cancel -h print-server queue-jobnnumber
```

31.7.8 Defective Print Jobs and Data Transfer Errors

Print jobs remain in the queues and printing resumes if you switch the printer off and on or shut down and reboot the computer during the printing process. Defective print jobs must be removed from the queue with `cancel`.

If a print job is defective or an error occurs in the communication between the host and the printer, the printer prints numerous sheets of paper with unintelligible characters, because it is unable to process the data correctly. To deal with this, follow these steps:

1. To stop printing, remove all paper from ink jet printers or open the paper trays of laser printers. High-quality printers have a button for canceling the current printout.

2. The print job may still be in the queue, because jobs are only removed after they are sent completely to the printer. Use `lpstat -o` or `lpstat -h` *`print-server`* `-o` to check which queue is currently printing. Delete the print job with `cancel` *`queue-jobnumber`* or `cancel -h` *`print-server queue-jobnumber`*.

3. Some data may still be transferred to the printer even though the print job has been deleted from the queue. Check if a CUPS back-end process is still running for the respective queue and terminate it. For example, for a printer connected to the parallel port, the command `fuser -k /dev/lp0` can be used to terminate all processes that are still accessing the printer (more precisely: the parallel port).

4. Reset the printer completely by switching it off for some time. Then insert the paper and turn on the printer.

31.7.9 Debugging the CUPS Print System

Use the following generic procedure to locate problems in the CUPS print system:

1. Set `LogLevel debug` in `/etc/cups/cupsd.conf`.

2. Stop `cupsd`.

3. Remove `/var/log/cups/error_log*` to avoid having to search through very large log files.

4. Start `cupsd`.

5. Repeat the action that led to the problem.

6. Check the messages in `/var/log/cups/error_log*` to identify the cause of the problem.

31.7.10 For More Information

Solutions to many specific problems are presented in the Support Database. If you experience problems with printers, refer to the Support Database articles *Installing a Printer* and *Printer Configuration from SUSE Linux 9.2*, which you can find by searching for the keyword *printer*.

32 The Hotplug System

The hotplug system controls the initialization of most devices in a computer. It is not only used for devices that can be inserted and removed during operation, but for all devices that are detected while the system is booting. It works closely together with the `sysfs` file system and `udev`, which are described in Chapter 33, *Dynamic Device Nodes with `udev`* (page 491).

Until the kernel has been booted, only devices that are absolutely necessary, like the bus system, boot disks, and keyboard, are initialized. The kernel triggers hotplug events for all devices that were detected. The `udevd` daemon listens to these events and runs `udev` to create the device node and configure the device. For devices that cannot be detected automatically, like old ISA cards, a static configuration is used.

Apart from a few historic exceptions, most devices are initialized immediately as soon as they are accessible, either during system boot or when devices are hotplugged. During initialization, interfaces are registered with the kernel. This registration triggers further hotplug events that cause an automatic configuration of the respective interface.

In former versions of SUSE Linux, a static set of configuration data was used as the basis for initializing devices. Any hotplug events were handled by separate scripts, called agents. With this release of SUSE Linux the hotplug subsystem is integrated into udev, with udev rules provide the functionality of the former hotplug agents.

The general settings for the hotplug subsystem can be found in `/etc/sysconfig/hotplug`. All variables are commented. General device configuration is made depending on matching rules found in `/etc/udev/rules.d` (see Chapter 33, *Dynamic Device Nodes with `udev`* (page 491)). Configuration files for specific devices are located in `/etc/sysconfig/hardware`. The hotplug event callback used in former version of SUSE Linux, /proc/sys/kernel/hotplug, is usually empty because udevd receives hotplug messages via a netlink socket.

32.1 Devices and Interfaces

The hotplug system configures not only devices but also interfaces. A device is commonly connected to a bus and provides the functionality required for an interface. An interface represents the user-visible abstraction of either the entire or a certain subset of a device. A device usually requires a device driver in the form of kernel modules to function properly. Additionally, some higher-level driver might be needed to provide the interface to the user. Interfaces are mostly represented by device nodes created by udev. The distinction of devices and interfaces is important for understanding the overall concept.

Devices entered in the sysfs file system are found under /sys/devices. Interfaces are located under /sys/class or /sys/block. All interfaces in sysfs should have a link to their devices. However, there are still some drivers that do not automatically add this link. Without that link, it is unknown to which device this interface belongs and a suitable configuration cannot be found.

Devices are addressed by means of a device description. This may be the device path in sysfs (/sys/devices/pci0000:00/0000:00:1e.0/0000:02:00.0), a description of the connection point (bus-pci-0000:02:00.0), an individual ID (id-32311AE03FB82538), or something similar. In the past, interfaces were addressed by means of their names. These names represented a simple numbering of the existing devices and might have changed when devices were added or removed.

Interfaces can also be addressed by means of a description of the associated device. Usually, the context indicates whether the description refers to the device itself or to its interface. Typical examples of devices, interfaces, and descriptions include:

PCI Network Card

A device that is connected to the PCI bus (/sys/devices/pci0000:00/0000:00:1e.0/0000:02:00.0 or bus-pci-0000:02:00.0) and has a network interface (eth0, id-00:0d:60:7f:0b:22 or bus-pci-0000:02:00.0). The network interface is used by network services or connected to a virtual network device, such as a tunnel or VLAN, which in turn has an interface.

PCI SCSI Controller

A device (/sys/devices/pci0000:20/0000:20:01.1/host1/1:0:0:0 or bus-scsi-1:0:0:0) that makes several physical interfaces available in the form of a bus (/sys/class/scsi_host/host1).

SCSI Hard Disk

A device (/sys/devices/pci0000:20/0000:20:01.1/host1/1:0:0:0 or bus-scsi-1:0:0:0) with several interfaces (/sys/block/sda*).

32.2 Hotplug Events

Every device and every interface has an associated *hotplug event*, which is processed by udev. Hotplug events are triggered by the kernel when a link to a device is established or removed or when a driver registers or deletes an interface. Since SUSE Linux 9.3, udevd receives and processes hotplug events. Either udevd listens directly to netlink messages from the kernel or /sbin/udevsend must be specified in /proc/sys/kernel/hotplug. udevd configures the device according to a set of rules (see Chapter 33, *Dynamic Device Nodes with udev* (page 491)).

32.3 Hotplug Device Configuration

Hotplug agents have been deprecated as of SUSE Linux 10.0. All device configuration should now be done via udev rules. udev provides a compability rule to call existing custom agents. However, converting custom agents into udev rules should be considered.

A hotplug agent is an executable program that performs suitable actions for an event. The agents for device events are located in /etc/hotplug.d/*event name* and /etc/hotplug.d/default. All programs in these directories that have the suffix .hotplug are executed in alphabetical order.

To facilitate device configuration it is usually sufficient to load a kernel module. In some cases, additional commands need to be called for a proper device configuration. In SUSE Linux, this is handled generally by udev rules. However, if a custom device configuration is required, the device configuration is done by /sbin/hwup or /sbin/hwdown. These programs search for a configuration suitable for the device in the directory /etc/sysconfig/hardware and apply it. For example, to prevent a specific device from being initialized, create a configuration file with an appropriate name and set the start mode to `manual` or `off`. If `/sbin/hwup` does not find any configuration, it looks for the environment variable `MODALIAS`. If it exists, `modprobe` automatically loads the corresponding module. The `MODALIAS` variable is automatically generated by kernel hotplug events for devices that require a module to be loaded. For more information, see Section 32.4, "Automatic Module Loading" (page 489). More information about `/sbin/hwup` is available in the file `/usr/share/doc/packages/sysconfig/README` and in the manual page `man hwup`.

Before interface agents are called, `udev` usually generates a device node the system can access. `udev` enables the assignment of persistent names to interfaces. See Chapter 33, *Dynamic Device Nodes with `udev`* (page 491) for details. The interfaces itself are then set up according to the respective udev rules. The procedures for some interfaces are described below.

32.3.1 Activating Network Interfaces

Network interfaces are initialized with `/sbin/ifup` and deactivated with `/sbin/ifdown`. Details are provided in the file `/usr/share/doc/packages/sysconfig/README` and in the `ifup` man page.

If a computer has several network devices with different drivers, the designations of the interface can change if another driver is loaded faster while the system is booting. SUSE Linux tries to keep the numbering persistent—the devices retain the interface name they have been assigned during configuration. This assigment is done via udev rules. To change the assignment later, the udev rules must be changed.

The best solution, however, is to use persistent interface designations. You can specify the names of the individual interfaces in the configuration files. Details about this method are available in the file `/usr/share/doc/packages/sysconfig/README`. Since SUSE Linux 9.3, udev also deals with network interfaces, although these are not device nodes. This allows use of persistent interface names in a more standardized manner.

32.3.2 Activating Storage Devices

Interfaces to storage devices must be mounted to be able to access them. This can be fully automated or preconfigured. Additionally, SUSE Linux distinguishes between system and user devices. System devices can only be automatically mounted by creating an entry in /etc/fstab. User devices are handled via hal by default. If a different configuration for user devices is required, these devices can be entered into /etc/fstab. Alternatively, the handling of this device in hal can be modified. For more information about hal, refer to /usr/share/doc/packages/hal/hal-spec.html.

The use of persistent device names is recommended, because traditional device names may change depending on the initialization sequence. Details about persistent device names is available in Chapter 33, *Dynamic Device Nodes with udev* (page 491).

32.4 Automatic Module Loading

If /sbin/hwup fails to detect a configuration file, modprobe searches for a corresponding module based on the contents of the environment variable MODALIAS. This environment variable is generated by the kernel for the corresponding hotplug event. To use a driver other than the standard driver for the kernel, an appropriate hardware configuration file in /etc/sysconfig/hardware should be created.

32.5 The Boot Script Coldplug

boot.coldplug is responsible for initializing all devices that have not been configured during boot. It calls hwup for every static device configuration designated as /etc/sysconfig/hardware/hwcfg-static-*. After this, it replays all events stored in /lib/klibc/events to initialize all devices.

32.6 Error Analysis

32.6.1 Log Files

Unless otherwise specified, hotplug only sends a few important messages to syslog. To obtain more information, set the variable HOTPLUG_DEBUG in the file /etc/sysconfig/hotplug to yes. If you set this variable to the value max, every shell command is logged for all hotplug scripts. This means that /var/log/messages in which syslog stores all the messages becomes much larger. Because syslog is launched during the boot process after hotplug and coldplug, it is possible, however, for the first messages not to be logged. If these messages are important to you, specify a different log file via the variable HOTPLUG_SYSLOG. Information about this topic is available in /etc/sysconfig/hotplug.

32.6.2 Boot Problems

If a computer hangs during the boot process, disable hotplug or coldplug by entering NOHOTPLUG=yes or NOCOLDPLUG=yes at the boot prompt. Due to the deactivation of hotplug, the kernel does not issue any hotplug events. In the running system, you can activate hotplug by entering the command /etc/init.d/boot.hotplug start. All events generated up to that time are then issued and processed. To reject the queued events, first enter /bin/true in /proc/sys/kernel/hotplug and reset the entry to /sbin/hotplug after some time. Because of the deactivation of coldplug, static configurations are not applied. To apply the static configurations, later enter /etc/init.d/boot.coldplug start.

To find out whether a particular module loaded by hotplug is responsible for the problem, enter HOTPLUG_TRACE=<N> at the boot prompt. The names of all the modules to load are then listed on the screen before they are actually loaded after *N* seconds. You cannot intervene while this is going on.

32.6.3 The Event Recorder

The script /sbin/hotplugeventrecorder is executed for every event by a udev rule. If a directory /events exists, all hotplug events are stored as individual files in this directory. Thus, events can be regenerated for test purposes. If this directory does not exist, nothing is recorded.

Dynamic Device Nodes with udev 33

Linux kernel 2.6 introduces a new user space solution for a dynamic device directory /dev with persistent device designations: udev. It provides only the files for devices that are actually present. It creates or removes device node files usually located in the /dev directory and is able to rename network interfaces. The previous implementation of a dynamic /dev with devfs has been replaced by udev.

Traditionally, device nodes were stored in the /dev directory on Linux systems. There was a node for every possible type of device, regardless of whether it actually existed in the system. As a result, this directory contained thousands of unused files. Before a newly added subsystem or kernel device was usable, the corresponding nodes needed to be created with an special application. The devfs file system brought a significant improvement, because only devices that actually existed and were known to the kernel were given a device node in /dev.

udev introduces a new way of creating device nodes. The kernel exports its internal state in sysfs and, every time a device is recognized by the kernel, it updates the information in sysfs and sends an event to user space. With the information made available by sysfs udev matches a simple rule syntax with the provided device attributes and creates or removes the corresponding device nodes.

The user is not required to create any udev rule for new devices. If a device is connected, the appropriate device node is created automatically. However, the rules introduce the possibility of defining a policy for device naming. This also offers the convenience of replacing a cryptic device name with a name that is easy to remember and also of having persistent device names where two devices of the same type have been connected at the same time.

Assume you have two printers, a high-quality color laser printer and a black-and-white ink jet printer, both connected via USB. They appear as /dev/usb/lpX, where X is a number depending on the order in which they have been connected. Using udev, create custom udev rules naming one printer /dev/colorlaser and the other /dev/inkprinter. Because these device nodes are created by udev based on the characteristics of the device, they always point to the correct device, regardless of the connection order or status.

33.1 Creating Rules

Before udev creates device nodes under /dev, it reads all files in /etc/udev/rules.d with the suffix .rules in alphabetical order. The first rule that fits a device is used, even if other rules would also apply. Comments are introduced with a hash sign (#). Rules take the following form:

```
key, [key,...] NAME [, SYMLINK]
```

At least one key must be specified, because rules are assigned to devices on the basis of these keys. It is also essential to specify a name. The device node that is created in /dev bears this name. The optional symlink parameter allows nodes to be created in other places. A rule for a printer could take the following form:

```
BUS=="usb", SYSFS{serial}=="12345", NAME="lp_hp", SYMLINK+="printers/hp"
```

In this example, there are two keys, BUS and SYSFS{serial}. udev compares the serial number to the serial number of the device that is connected to the USB bus. To assign the name lp_hp to the device in the /dev directory, all the keys must be identical. In addition, a symbolic link /dev/printers/hp, which refers to the device node, is created. At the same time, the printers directory is automatically created. Print jobs can then be sent to /dev/printers/hp or /dev/lp_hp.

33.2 Placeholder Substitution

The parameters NAME and SYMLINK allow the use of placeholders to substitute special values. A simple example illustrates the procedure:

```
BUS=="usb", SYSFS{vendor}=="abc", SYSFS{model}=="xyz", NAME="camera%n"
```

The operator %n in the name is replaced by the number of the camera device, such as `camera0` or `camera1`. Another useful operator is `%k`, which is replaced by the standard device name of the kernel, for example, `hda1`. You may also call an external program in udev rules and use the string that is returned in the `NAME` and `SYMLINK` values. The complete list of possible placeholders is described in the `udev` man page.

33.3 Pattern Matching in Keys

In the keys of udev rules, you may use shell-style pattern matching, known as wild cards. For example, the character `*` can be used as a placeholder for any characters or `?` can be used for precisely one arbitrary character.

```
KERNEL="ts*", NAME="input/%k"
```

This rule assigns the standard kernel name in the standard directory to a device whose designation begins with the letters “ts”. Find detailed information about the use of pattern matching in udev rules in the `udev` man page.

33.4 Key Selection

To identify a device uniquely and distinguish multiple devices from each other, a unique property is essential for a working udev rule. Here are some examples of standard keys:

SUBSYSTEM
Subsystem of which the device is part

BUS
Device bus type

KERNEL
Device name the kernel uses

ID
Device number on the bus (for example, PCI bus ID)

SYSFS{...}
sysfs device attributes, like label, vendor, or serial number

The keys SUBSYSTEM and ID can be useful, but usually the keys BUS, KERNEL, and SYSFS{...} are used. The udev configuration also provides keys that call external scripts and evaluate their results. Find details about this in the udev man page.

The file system sysfs exposes information about the hardware in a directory tree. Each file generally only contains one item of information, such as the device name, the vendor, or the serial number. Each of these files can be used to match with a key. To use several SYSFS keys in one rule, however, you can only use files in the same directory as key values. The tool udevinfo can help finding useful and unique key values.

You must find one subdirectory of /sys that refers to the relevant device and contains a file dev. These directories are all located under /sys/block or /sys/class. If a device node already exists for the device, udevinfo can find the right subdirectory for you. The command udevinfo -q path -n /dev/sda outputs /block/sda. This means that the desired directory is /sys/block/sda. Now call udevinfo with the command udevinfo -a -p /sys/block/sda. The two commands can also be combined, as in udevinfo -a -p `udevinfo -q path -n /dev/sda`. The following is an extract from the output:

```
BUS=="scsi"
ID=="0:0:0:0"
SYSFS{detach_state}=="0"
SYSFS{type}=="0"
SYSFS{max_sectors}=="240"
SYSFS{device_blocked}=="0"
SYSFS{queue_depth}=="1"
SYSFS{scsi_level}=="3"
SYSFS{vendor}==" "
SYSFS{model}=="USB 2.0M DSC"
SYSFS{rev}=="1.00"
SYSFS{online}=="1"
```

From the output information, look for suitable keys that do not change. Remember that you cannot use keys from different directories in one rule.

33.5 Persistent Names for Mass Storage Devices

SUSE Linux comes with predefined rules that allow you always to assign the same designations to hard disks and other storage devices, no matter in which order they are initialized. Unique device attributes, like hardware serial numbers, UUIDs or file system labels, can be read with small helper programs that come with udev. The helper programs make specific device information available to the udev rule processing. As a simplified example, the first rule imports the values gathered from the SCSI device in the udev environment. The second rule uses the imported values to create a persistent symlink.

```
KERNEL="sd*[!0-9]", IMPORT="/sbin/scsi_id -g -x -s $p -d %N"
KERNEL="sd*[!0-9]", SYMLINK+="$env{ID_TYPE}/by-id/$env{ID_BUS}-$env{ID_SERIAL}"
```

As soon as a driver for a mass storage device has been loaded, it registers all the available hard disks with the kernel. Each of them triggers a hotplug block event that calls `udev`. Then `udev` reads the rules to determine whether a symlink needs to be created.

If the driver is loaded via `initrd`, the hotplug events are lost. However, all the information is stored in `sysfs`. The `udevstart` utility finds all the device files under `/sys/block` and `/sys/class` and starts `udev`.

There is also a start script `boot.udev`, which recreates all the device nodes during the boot process. However, the start script must be activated through the YaST runlevel editor or with the command `insserv boot.udev`.

File Systems in Linux

34

Linux supports a number of different file systems. This chapter presents a brief overview of the most popular Linux file systems, elaborating on their design concept, advantages, and fields of application. Some additional information about LFS (large file support) in Linux is also provided.

34.1 Terminology

metadata

A file system–internal data structure that assures all the data on disk is properly organized and accessible. Essentially, it is “data about the data.” Almost every file system has its own structure of metadata, which is part of why the file systems show different performance characteristics. It is extremely important to maintain metadata intact, because otherwise all data on the file system could become inaccessible.

inode

Inodes contain various information about a file, including size, number of links, date and time of creation, modification, and access, and pointers to the disk blocks where the file contents are actually stored.

journal

In the context of a file system, a journal is an on-disk structure containing a kind of log in which the file system stores what it is about to change in the file system's metadata. Journaling greatly reduces the recovery time of a Linux system because it obsoletes the lengthy search process that checks the entire file system at system start-up. Instead, only the journal is replayed.

34.2 Major File Systems in Linux

Unlike two or three years ago, choosing a file system for a Linux system is no longer a matter of a few seconds (Ext2 or ReiserFS?). Kernels starting from 2.4 offer a variety of file systems from which to choose. The following is an overview of how these file systems basically work and which advantages they offer.

It is very important to bear in mind that there may be no file system that best suits all kinds of applications. Each file system has its particular strengths and weaknesses, which must be taken into account. Even the most sophisticated file system cannot substitute for a reasonable backup strategy, however.

The terms *data integrity* and *data consistency*, when used in this chapter, do not refer to the consistency of the user space data (the data your application writes to its files). Whether this data is consistent must be controlled by the application itself.

IMPORTANT: Setting Up File Systems

Unless stated otherwise in this chapter, all the steps required to set up or change partitions and file systems can be performed using the YaST module.

34.2.1 ReiserFS

Officially one of the key features of the 2.4 kernel release, ReiserFS has been available as a kernel patch for 2.2.x SUSE kernels since SUSE Linux version 6.4. ReiserFS was designed by Hans Reiser and the Namesys development team. It has proven itself to be a powerful alternative to the old Ext2. Its key assets are better disk space utilization, better disk access performance, and faster crash recovery.

ReiserFS's strengths, in more detail, are:

Better Disk Space Utilization

In ReiserFS, all data is organized in a structure called B^*-balanced tree. The tree structure contributes to better disk space utilization because small files can be stored directly in the B^* tree leaf nodes instead of being stored elsewhere and just maintaining a pointer to the actual disk location. In addition to that, storage is not allocated in chunks of 1 or 4 kB, but in portions of the exact size needed. Another benefit lies in the dynamic allocation of inodes. This keeps the file system more flexible than traditional file systems, like Ext2, where the inode density must be specified at file system creation time.

Better Disk Access Performance

For small files, file data and “stat_data” (inode) information are often stored next to each other. They can be read with a single disk I/O operation, meaning that only one access to disk is required to retrieve all the information needed.

Fast Crash Recovery

Using a journal to keep track of recent metadata changes makes a file system check a matter of seconds, even for huge file systems.

Reliability through Data Journaling

ReiserFS also supports data journaling and ordered data modes similar to the concepts outlined in the Ext3 section, Section 34.2.3, “Ext3” (page 500). The default mode is `data=ordered`, which ensures both data and metadata integrity, but uses journaling only for metadata.

34.2.2 Ext2

The origins of Ext2 go back to the early days of Linux history. Its predecessor, the Extended File System, was implemented in April 1992 and integrated in Linux 0.96c. The Extended File System underwent a number of modifications and, as Ext2, became the most popular Linux file system for years. With the creation of journaling file systems and their astonishingly short recovery times, Ext2 became less important.

A brief summary of Ext2's strengths might help understand why it was—and in some areas still is—the favorite Linux file system of many Linux users.

Solidity

Being quite an “old-timer,” Ext2 underwent many improvements and was heavily tested. This may be the reason why people often refer to it as rock-solid. After a system outage when the file system could not be cleanly unmounted, e2fsck starts to analyze the file system data. Metadata is brought into a consistent state and pending files or data blocks are written to a designated directory (called `lost+found`). In contrast to journaling file systems, e2fsck analyzes the entire file system and not just the recently modified bits of metadata. This takes significantly longer than checking the log data of a journaling file system. Depending on file system size, this procedure can take half an hour or more. Therefore, it is not desirable to choose Ext2 for any server that needs high availability. However, because Ext2 does not maintain a journal and uses significantly less memory, it is sometimes faster than other file systems.

Easy Upgradability

The code for Ext2 is the strong foundation on which Ext3 could become a highly-acclaimed next-generation file system. Its reliability and solidity were elegantly combined with the advantages of a journaling file system.

34.2.3 Ext3

Ext3 was designed by Stephen Tweedie. Unlike all other next-generation file systems, Ext3 does not follow a completely new design principle. It is based on Ext2. These two file systems are very closely related to each other. An Ext3 file system can be easily built on top of an Ext2 file system. The most important difference between Ext2 and Ext3 is that Ext3 supports journaling. In summary, Ext3 has three major advantages to offer:

Easy and Highly Reliable Upgrades from Ext2

Because Ext3 is based on the Ext2 code and shares its on-disk format as well as its metadata format, upgrades from Ext2 to Ext3 are incredibly easy. Unlike transitions to other journaling file systems, such as ReiserFS, JFS, or XFS, which can be quite tedious (making backups of the entire file system and recreating it from scratch), a transition to Ext3 is a matter of minutes. It is also very safe, because recreating an entire file system from scratch might not work flawlessly. Considering the number of existing Ext2 systems that await an upgrade to a journaling file system, you can easily figure out why Ext3 might be of some importance to many system administrators. Downgrading from Ext3 to Ext2 is as easy as the upgrade. Just perform a clean unmount of the Ext3 file system and remount it as an Ext2 file system.

Reliability and Performance

Some other journaling file systems follow the “metadata-only” journaling approach. This means your metadata is always kept in a consistent state, but the same cannot be automatically guaranteed for the file system data itself. Ext3 is designed to take care of both metadata and data. The degree of “care” can be customized. Enabling Ext3 in the `data=journal` mode offers maximum security (data integrity), but can slow down the system because both metadata and data are journaled. A relatively new approach is to use the `data=ordered` mode, which ensures both data and metadata integrity, but uses journaling only for metadata. The file system driver collects all data blocks that correspond to one metadata update. These data blocks are written to disk before the metadata is updated. As a result, consistency is achieved for metadata and data without sacrificing performance. A third option to use is `data=writeback`, which allows data to be written into the main file system after its metadata has been committed to the journal. This option is often considered the best in performance. It can, however, allow old data to reappear in files after crash and recovery while internal file system integrity is maintained. Unless you specify something else, Ext3 is run with the `data=ordered` default.

34.2.4 Converting an Ext2 File System into Ext3

Converting from Ext2 to Ext3 involves two separate steps:

Creating the Journal
Log in as `root` and run `tune2fs -j`. This creates an Ext3 journal with the default parameters. To decide yourself how large the journal should be and on which device it should reside, run `tune2fs -J` instead together with the desired journal options `size=` and `device=`. More information about the tune2fs program is available in its manual page (tune2fs(8)).

Specifying the File System Type in /etc/fstab
To ensure that the Ext3 file system is recognized as such, edit the file `/etc/fstab`, changing the file system type specified for the corresponding partition from `ext2` to `ext3`. The change takes effect after the next reboot.

Using Ext3 for the Root Directory
To boot a root file system set up as an Ext3 partition, include the modules `ext3` and `jbd` in the `initrd`. To do so, edit the file `/etc/sysconfig/kernel` to include the two modules under `INITRD_MODULES` then execute the command `mkinitrd`.

34.2.5 Reiser4

Right after kernel 2.6 had been released, the family of journaling file systems was joined by another member: Reiser4. Reiser4 is fundamentally different from its predecessor

ReiserFS (version 3.6). It introduces the concept of plug-ins to tweak the file system functionality and a finer grained security concept.

Fine Grained Security Concept

In designing Reiser4, its developers put an emphasis on the implementation of security-relevant features. Reiser4 therefore comes with a set of dedicated security plug-ins. The most important one introduces the concept of file "items." Currently, file access controls are defined per file. If there is a large file containing information relevant to several users, groups, or applications, the access rights had be fairly imprecise to include all parties involved. In Reiser4, you can split those files into smaller portions (the "items"). Access rights can then be set for each item and each user separately, allowing a much more precise file security management. A perfect example would be `/etc/passwd`. To date, only `root` can read and edit the file while non-`root` users only get read access to this file. Using the item concept of Reiser4, you could split this file in a set of items (one item per user) and allow users or applications to modify their own data but not access other users' data. This concept adds both to security and flexibility.

Extensibility through Plug-Ins

Many file system functions and external functions normally used by a file system are implemented as plug-ins in Reiser4. These plug-ins can easily be added to the base system. You no longer need to recompile the kernel or reformat the hard disk to add new functionalities to your file system.

Better File System Layout through Delayed Allocation

Like XFS, Reiser4 supports delayed allocation. See Section 34.2.7, "XFS" (page 503). Using delayed allocation even for metadata can result in better overall layout.

34.2.6 JFS

JFS, the *Journaling File System*, was developed by IBM. The first beta version of the JFS Linux port reached the Linux community in the summer of 2000. Version 1.0.0 was released in 2001. JFS is tailored to suit the needs of high throughput server environments where performance is the ultimate goal. Being a full 64-bit file system, JFS supports both large files and partitions, which is another reason for its use in server environments.

A closer look at JFS shows why this file system might prove a good choice for your Linux server:

Efficient Journaling

JFS follows a "metadata-only" approach. Instead of an extensive check, only metadata changes generated by recent file system activity are checked, which saves a great amount of time in recovery. Concurrent operations requiring multiple concurrent log entries can be combined into one group commit, greatly reducing performance loss of the file system through multiple write operations.

Efficient Directory Organization

JFS holds two different directory organizations. For small directories, it allows the directory's content to be stored directly into its inode. For larger directories, it uses B^+trees, which greatly facilitate directory management.

Better Space Usage through Dynamic inode Allocation

For Ext2, you must define the inode density in advance (the space occupied by management information), which restricts the maximum number of files or directories of your file system. JFS spares you these considerations—it dynamically allocates inode space and frees it when it is no longer needed.

34.2.7 XFS

Originally intended as the file system for their IRIX OS, SGI started XFS development in the early 1990s. The idea behind XFS was to create a high-performance 64-bit journaling file system to meet the extreme computing challenges of today. XFS is very good at manipulating large files and performs well on high-end hardware. However, even XFS has a drawback. Like ReiserFS, XFS takes great care of metadata integrity, but less of data integrity.

A quick review of XFS's key features explains why it may prove a strong competitor for other journaling file systems in high-end computing.

High Scalability through the Use of Allocation Groups

At the creation time of an XFS file system, the block device underlying the file system is divided into eight or more linear regions of equal size. Those are referred to as *allocation groups*. Each allocation group manages its own inodes and free disk space. Practically, allocation groups can be seen as file systems in a file system. Because allocation groups are rather independent of each other, more than one of them can be addressed by the kernel simultaneously. This feature is the key to XFS's great scalability. Naturally, the concept of independent allocation groups suits the needs of multiprocessor systems.

High Performance through Efficient Management of Disk Space

Free space and inodes are handled by B^+ trees inside the allocation groups. The use of B^+ trees greatly contributes to XFS's performance and scalability. XFS uses *delayed allocation*. It handles allocation by breaking the process into two pieces. A pending transaction is stored in RAM and the appropriate amount of space is reserved. XFS still does not decide where exactly (speaking of file system blocks) the data should be stored. This decision is delayed until the last possible moment. Some short-lived temporary data may never make its way to disk, because it may be obsolete by the time XFS decides where actually to save it. Thus XFS increases write performance and reduces file system fragmentation. Because delayed allocation results in less frequent write events than in other file systems, it is likely that data loss after a crash during a write is more severe.

Preallocation to Avoid File System Fragmentation

Before writing the data to the file system, XFS *reserves* (preallocates) the free space needed for a file. Thus, file system fragmentation is greatly reduced. Performance is increased because the contents of a file are not distributed all over the file system.

34.3 Some Other Supported File Systems

Table 34.1, "File System Types in Linux" (page 504) summarizes some other file systems supported by Linux. They are supported mainly to ensure compatibility and interchange of data with different kinds of media or foreign operating systems.

Table 34.1 *File System Types in Linux*

`cramfs`	*Compressed ROM file system*: A compressed read-only file system for ROMs.
`hpfs`	*High Performance File System*: The IBM OS/2 standard file system—only supported in read-only mode.
`iso9660`	Standard file system on CD-ROMs.

`minix` This file system originated from academic projects on operating systems and was the first file system used in Linux. Today, it is used as a file system for floppy disks.

`msdos` *fat*, the file system originally used by DOS, is today used by various operating systems.

`ncpfs` File system for mounting Novell volumes over networks.

`nfs` *Network File System*: Here, data can be stored on any machine in a network and access may be granted via a network.

`smbfs` *Server Message Block* is used by products such as Windows to enable file access over a network.

`sysv` Used on SCO UNIX, Xenix, and Coherent (commercial UNIX systems for PCs).

`ufs` Used by BSD, SunOS, and NeXTstep. Only supported in read-only mode.

`umsdos` *UNIX on MSDOS*: Applied on top of a normal `fat` file system, achieves UNIX functionality (permissions, links, long filenames) by creating special files.

`vfat` *Virtual FAT*: Extension of the `fat` file system (supports long filenames).

`ntfs` *Windows NT file system*, read-only.

34.4 Large File Support in Linux

Originally, Linux supported a maximum file size of 2 GB. This was enough before the explosion of multimedia and as long as no one tried to manipulate huge databases on Linux. Becoming more and more important for server computing, the kernel and C library were modified to support file sizes larger than 2 GB when using a new set of interfaces that applications must use. Today, almost all major file systems offer LFS support, allowing you to perform high-end computing. Table 34.2, "Maximum Sizes of File Systems (On-Disk Format)" (page 506) offers an overview of the current limitations of Linux files and file systems.

Table 34.2 *Maximum Sizes of File Systems (On-Disk Format)*

File System	File Size (Bytes)	File System Size (Bytes)
Ext2 or Ext3 (1 kB block size)	2^{34} (16 GB)	2^{41} (2 TB)
Ext2 or Ext3 (2 kB block size)	2^{38} (256 GB)	2^{43} (8 TB)
Ext2 or Ext3 (4 kB block size)	2^{41} (2 TB)	2^{44} (16 TB)
Ext2 or Ext3 (8 kB block size) (systems with 8 kB pages, like Alpha)	2^{46} (64 TB)	2^{45} (32 TB)
ReiserFS v3	2^{46} (64 GB)	2^{45} (32 TB)
XFS	2^{63} (8 EB)	2^{63} (8 EB)
JFS (512 byte block size)	2^{63} (8 EB)	2^{49} (512 TB)
JFS (4 kB block size)	2^{63} (8 EB)	2^{52} (4 PB)
NFSv2 (client side)	2^{31} (2 GB)	2^{63} (8 EB)
NFSv3 (client side)	2^{63} (8 EB)	2^{63} (8 EB)

IMPORTANT: Linux Kernel Limits

Table 34.2, "Maximum Sizes of File Systems (On-Disk Format)" (page 506) describes the limitations regarding the on-disk format. The 2.6 kernel imposes its own limits on the size of files and file systems handled by it. These are as follows:

File Size
On 32-bit systems, files may not exceed the size of 2 TB (2^{41} bytes).

File System Size

File systems may be up to 2^{73} bytes large. However, this limit is still out of reach for the currently available hardware.

34.5 For More Information

Each of the file system projects described above maintains its own home page on which to find mailing list information, further documentation, and FAQs.

- http://e2fsprogs.sourceforge.net/
- http://www.zipworld.com.au/~akpm/linux/ext3/
- http://www.namesys.com/
- http://oss.software.ibm.com/developerworks/opensource/jfs/
- http://oss.sgi.com/projects/xfs/

A comprehensive multipart tutorial about Linux file systems can be found at *IBM developerWorks*: http://www-106.ibm.com/developerworks/library/l-fs.html. For a comparison of the different journaling file systems in Linux, look at Juan I. Santos Florido's article at *Linuxgazette*: http://www.linuxgazette.com/issue55/florido.html. Those interested in an in-depth analysis of LFS in Linux should try Andreas Jaeger's LFS site: http://www.suse.de/~aj/linux_lfs.html.

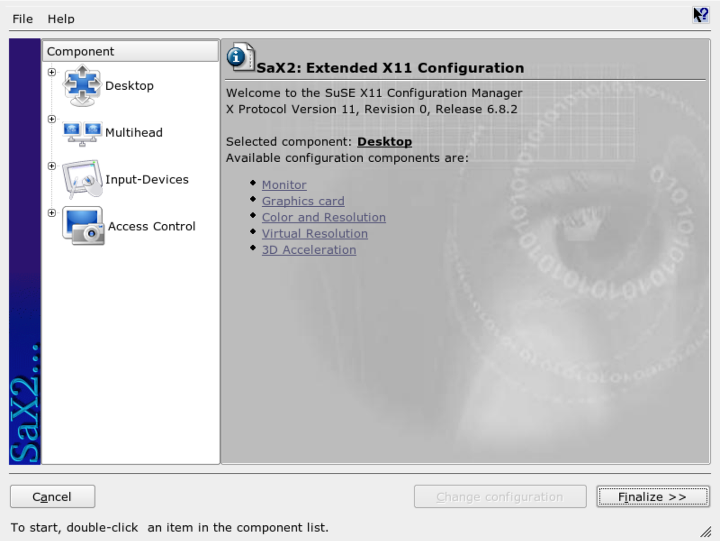
35 The X Window System

The X Window System (X11) is the de facto standard for graphical user interfaces in UNIX. X is network-based, enabling applications started on one host to be displayed on another host connected over any kind of network (LAN or Internet). This chapter describes the setup and optimization of the X Window System environment, provides background information about the use of fonts in SUSE Linux, and explains the configuration of OpenGL and 3D.

35.1 X11 Setup with SaX2

The graphical user interface, or X server, handles the communication between hardware and software. Desktops, like KDE and GNOME, and the wide variety of window managers, use the X server for interaction with the user. The graphical user interface is initially configured during installation. To change the settings afterwards, use the respective module from the YaST control center or run SaX2 manually from the command line with the command `sax2`. The SaX2 main window provides a common umbrella for the individual modules from the YaST control center.

Figure 35.1 *The Main Window of SaX2*



SaX2: Extended X11 Configuration

Welcome to the SuSE X11 Configuration Manager
X Protocol Version 11, Revision 0, Release 6.8.2

Selected component: **Desktop**
Available configuration components are:

- Monitor
- Graphics card
- Color and Resolution
- Virtual Resolution
- 3D Acceleration

To start, double-click an item in the component list.

In the left navigation bar, there are six items, each of them showing the respective configuration dialog from the YaST control center. Find the sections mentioned below in Chapter *System Configuration with YaST* (↑Start-Up).

Monitor

For a description of the monitor and graphics card configuration, see Section “Card and Monitor Properties” (Chapter 3, *System Configuration with YaST*, ↑Start-Up).

Mouse

For a description of the mouse configuration in the graphical environment, see Section “Mouse Properties” (Chapter 3, *System Configuration with YaST*, ↑Start-Up).

Keyboard

For a description of the keyboard configuration in the graphical environment, see Section “Keyboard Properties” (Chapter 3, *System Configuration with YaST*, ↑Start-Up).

Tablet

For a description of the graphics tablet configuration, see Section “Tablet Properties” (Chapter 3, *System Configuration with YaST*, ↑Start-Up).

Touchscreen

For a description of the touchscreen configuration, see Section “Touchscreen Properties” (Chapter 3, *System Configuration with YaST*, ↑Start-Up).

VNC

For a description of the VNC configuration, see Section “Remote Access Properties” (Chapter 3, *System Configuration with YaST*, ↑Start-Up).

35.2 Optimizing the X Configuration

X.Org is an Open Source implementation of the X Window System. It is further developed by the X.Org Foundation, which is also responsible for the development of new technologies and standards of the X Window System.

To use the available hardware, including mouse, graphics card, monitor, and keyboard, in the best way possible, the configuration can be optimized manually. Some aspects of this optimization are explained below. For detailed information about configuring the X Window System, review the various files in the directory `/usr/share/doc/packages/Xorg` and `man xorg.conf`.

WARNING

Be very careful when configuring your X Window System. Never start the X Window System until the configuration is finished. A wrongly configured system can cause irreparable damage to your hardware (this applies especially to fixed-frequency monitors). The authors of this book and SUSE Linux cannot be held responsible for damage. This information has been carefully researched, but this does not guarantee that all methods presented here are correct and will not damage your hardware.

The programs SaX2 and xorgconfig create the file `xorg.conf`, by default in `/etc/X11`. This is the primary configuration file for the X Window System. Find all the settings here concerning your graphics card, mouse, and monitor.

The following paragraphs describe the structure of the configuration file `/etc/X11/xorg.conf`. It consists of several sections, each one dealing with a certain aspect of the configuration. Each section starts with the keyword `Section <designation>` and ends with `EndSection`. The sections have the form:

```
Section designation
  entry 1
  entry 2
  entry n
EndSection
```

The available section types are listed in Table 35.1, "Sections in /etc/X11/xorg.conf" (page 512).

Table 35.1 Sections in /etc/X11/xorg.conf

Type	Meaning
`Files`	This section describes the paths used for fonts and the RGB color table.
`ServerFlags`	General switches are set here.
`InputDevice`	Input devices, like keyboards and special input devices (touchpads, joysticks, etc.), are configured in this section. Important parameters in this section are `Driver` and the options defining the `Protocol` and `Device`.
`Monitor`	Describes the monitor used. The individual elements of this section are the name, which is referred to later in the `Screen` definition, the `bandwidth`, and the synchronization frequency limits (`HorizSync` and `VertRefresh`). Settings are given in MHz, kHz, and Hz. Normally, the server refuses any modeline that does not correspond with the specification of the monitor. This prevents too high frequencies from being sent to the monitor by accident.
`Modes`	The modeline parameters are stored here for the specific screen resolutions. These parameters can be calculated by SaX2 on the basis of the values given by the user and normally do not need to be changed. Intervene manually at this point, if, for example, you want to connect a fixed frequency monitor. Find details of the meaning of individual number values in the HOWTO file `/usr/share/doc/howto/en/XFree86-Video-Timings-HOWTO.gz`.

Type	Meaning
Device	This section defines a specific graphics card. It is referenced by its descriptive name.
Screen	This section puts together a Monitor and a Device to form all the necessary settings for X.Org. In the Display subsection, specify the size of the virtual screen (Virtual), the ViewPort, and the Modes used with this screen.
ServerLayout	This section defines the layout of a single or multihead configuration. This section binds the input devices InputDevice and the display devices Screen.

Monitor, Device, and Screen are explained in more detail below. Further information about the other sections can be found in the manual pages of X.Org and xorg.conf.

There can be several different Monitor and Device sections in xorg.conf. Even multiple Screen sections are possible. The following ServerLayout section determines which one is used.

35.2.1 Screen Section

First, take a closer look at the screen section, which combines a monitor with a device section and determines the resolution and color depth to use. A screen section might resemble Example 35.1, "Screen Section of the File /etc/X11/xorg.conf" (page 514).

Example 35.1 *Screen Section of the File /etc/X11/xorg.conf*

```
Section "Screen"
  DefaultDepth  16
  SubSection "Display"
    Depth       16
    Modes       "1152x864" "1024x768" "800x600"
    Virtual     1152x864
  EndSubSection
  SubSection "Display"
    Depth       24
    Modes       "1280x1024"
  EndSubSection
  SubSection "Display"
    Depth       32
    Modes "640x480"
  EndSubSection
  SubSection "Display"
    Depth        8
    Modes       "1280x1024"
  EndSubSection
  Device        "Device[0]"
  Identifier    "Screen[0]"
  Monitor       "Monitor[0]"
EndSection
```

The line `Identifier` (here `Screen[0]`) gives this section a defined name with which it can be uniquely referenced in the following `ServerLayout` section. The lines `Device` and `Monitor` specify the graphics card and the monitor that belong to this definition. These are just links to the `Device` and `Monitor` sections with their corresponding names or *identifiers*. These sections are discussed in detail below.

Use the `DefaultDepth` setting to select the color depth the server should use unless it is started with a specific color depth. There is a `Display` subsection for each color depth. The keyword `Depth` assigns the color depth valid for this subsection. Possible values for `Depth` are 8, 15, 16, and 24. Not all X server modules support all these values.

After the color depth, a list of resolutions is set in the `Modes` section. This list is checked by the X server from left to right. For each resolution, the X server searches for a suitable `Modeline` in the `Modes` section. The `Modeline` depends on the capability of both the monitor and the graphics card. The `Monitor` settings determine the resulting `Modeline`.

The first resolution found is the `Default mode`. With Ctrl + Alt + + (on the number pad), switch to the next resolution in the list to the right. With Ctrl + Alt + – (on the number pad), switch to the left. This enables you to vary the resolution while X is running.

The last line of the Display subsection with Depth 16 refers to the size of the virtual screen. The maximum possible size of a virtual screen depends on the amount of memory installed on the graphics card and the desired color depth, not on the maximum resolution of the monitor. Because modern graphics cards have a large amount of video memory, you can create very large virtual desktops. However, you may no longer be able to use 3D functionality if you fill most of the video memory with a virtual desktop. If the card has 16 MB video RAM, for example, the virtual screen can be up to 4096x4096 pixels in size at 8-bit color depth. Especially for accelerated cards, however, it is not recommended to use all your memory for the virtual screen, because this memory on the card is also used for several font and graphics caches.

35.2.2 Device Section

A device section describes a specific graphics card. You can have as many device entries in xorg.conf as you like, as long as their names are differentiated, using the keyword Identifier. As a rule—if you have more than one graphics card installed—the sections are simply numbered in order. The first one is called Device[0], the second one Device[1], and so on. The following file shows an excerpt from the Device section of a computer with a Matrox Millennium PCI graphics card:

```
Section "Device"
  BoardName     "MGA2064W"
  BusID         "0:19:0"
  Driver        "mga"
  Identifier    "Device[0]"
  VendorName    "Matrox"
  Option        "sw_cursor"
EndSection
```

If you use SaX2 for configuring, the device section should look something like the above example. Both the Driver and BusID are dependent on the hardware installed in your computer and are detected by SaX2 automatically. The BusID defines the PCI or AGP slot in which the graphics card is installed. This matches the ID displayed by the command lspci. The X server needs details in decimal form, but lspci displays these in hexadecimal form.

Via the Driver parameter, specify the driver to use for this graphics card. If the card is a Matrox Millennium, the driver module is called mga. The X server then searches through the ModulePath defined in the Files section in the drivers subdirectory. In a standard installation, this is the directory /usr/X11R6/lib/modules/drivers. _drv.o is added to the name, so, in the case of the mga driver, the driver file mga_drv.o is loaded.

The behavior of the X server or of the driver can also be influenced through additional options. An example of this is the option sw_cursor, which is set in the device section. This deactivates the hardware mouse cursor and depicts the mouse cursor using software. Depending on the driver module, there are various options available, which can be found in the description files of the driver modules in the directory /usr/X11R6/lib/X11/doc. Generally valid options can also be found in the manual pages (man xorg.conf and man X.Org).

35.2.3 Monitor and Modes Section

Like the Device sections, the Monitor and Modes sections describe one monitor each. The configuration file /etc/X11/xorg.conf can contain as many Monitor sections as desired. The server layout section specifies which Monitor section is relevant.

Monitor definitions should only be set by experienced users. The modelines constitute an important part of the Monitor sections. Modelines set horizontal and vertical timings for the respective resolution. The monitor properties, especially the allowed frequencies, are stored in the Monitor section.

WARNING

Unless you have an in-depth knowledge of monitor and graphics card functions, nothing should be changed in the modelines, because this could cause severe damage to your monitor.

Those who try to develop their own monitor descriptions should be very familiar with the documentation in /usr/X11/lib/X11/doc. The section covering the video modes deserves a special mention. It describes, in detail, how the hardware functions and how to create modelines.

Manual specification of modelines is rarely required today. If you are using a modern multisync monitor, the allowed frequencies and optimal resolutions can, as a rule, be read directly from the monitor by the X server via DDC, as described in the SaX2 configuration section. If this is not possible for some reason, use one of the VESA modes included in the X server. This will function with practically all graphics card and monitor combinations.

35.3 Installing and Configuring Fonts

The installation of additional fonts in SUSE Linux is very easy. Simply copy the fonts to any directory located in the X11 font path (see Section 35.3.2, "X11 Core Fonts" (page 521)). To enable use of the fonts, the installation directory should be a subdirectory of the directories configured in `/etc/fonts/fonts.conf` (see Section 35.3.1, "Xft" (page 517)).

The font files can be copied manually (as `root`) to a suitable directory, such as `/usr/X11R6/lib/X11/fonts/truetype`. Alternatively, the task can be performed with the KDE font installer in the KDE Control Center. The result is the same.

Instead of copying the actual fonts, you can also create symbolic links. For example, you may want to do this if you have licensed fonts on a mounted Windows partition and want to use them. Subsequently, run `SuSEconfig --module fonts`.

`SuSEconfig --module fonts` executes the script `/usr/sbin/fonts-config`, which handles the configuration of the fonts. To see what this script does, refer to the manual page of the script (`man fonts-config`).

The procedure is the same for bitmap fonts, TrueType and OpenType fonts, and Type1 (PostScript) fonts. All these font types can be installed in any directory. Only CID-keyed fonts require a slightly different procedure. For this, see Section 35.3.3, "CID-Keyed Fonts" (page 522).

X.Org contains two completely different font systems: the old *X11 core font system* and the newly designed *Xft and fontconfig* system. The following sections briefly describe these two systems.

35.3.1 Xft

From the outset, the programmers of Xft made sure that scalable fonts including antialiasing are supported well. If Xft is used, the fonts are rendered by the application using the fonts, not by the X server as in the X11 core font system. In this way, the respective application has access to the actual font files and full control of how the glyphs are rendered. This constitutes the basis for the correct display of text in a number of languages. Direct access to the font files is very useful for embedding fonts for printing to make sure that the printout looks the same as the screen output.

In SUSE Linux, the two desktop environments KDE and GNOME, Mozilla, and many other applications already use Xft by default. Xft is already used by more applications than the old X11 core font system.

Xft uses the fontconfig library for finding fonts and influencing how they are rendered. The properties of fontconfig are controlled by the global configuration file `/etc/fonts/fonts.conf` and the user-specific configuration file `~/.fonts.conf`. Each of these fontconfig configuration files must begin with

```
<?xml version="1.0"?>
<!DOCTYPE fontconfig SYSTEM "fonts.dtd">
<fontconfig>
```

and end with

```
</fontconfig>
```

To add directories to search for fonts, append lines such as the following:

```
<dir>/usr/local/share/fonts/</dir>
```

However, this is usually not necessary. By default, the user-specific directory `~/.fonts` is already entered in `/etc/fonts/fonts.conf`. Accordingly, all you need to do to install additional fonts is to copy them to `~/.fonts`.

You can also insert rules that influence the appearance of the fonts. For example, enter

```
<match target="font">
 <edit name="antialias" mode="assign">
  <bool>false</bool>
 </edit>
</match>
```

to disable antialiasing for all fonts or

```
<match target="font">
 <test name="family">
  <string>Luxi Mono</string>
  <string>Luxi Sans</string>
 </test>
 <edit name="antialias" mode="assign">
 <bool>false</bool>
 </edit>
</match>
```

to disable antialiasing for specific fonts.

By default, most applications use the font names `sans-serif` (or the equivalent `sans`), `serif`, or `monospace`. These are not real fonts but only aliases that are resolved to a suitable font, depending on the language setting.

Users can easily add rules to `~/.fonts.conf` to resolve these aliases to their favorite fonts:

```
<alias>
 <family>sans-serif</family>
 <prefer>
  <family>FreeSans</family>
 </prefer>
</alias>
<alias>
 <family>serif</family>
 <prefer>
  <family>FreeSerif</family>
 </prefer>
</alias>
<alias>
 <family>monospace</family>
 <prefer>
  <family>FreeMono</family>
 </prefer>
</alias>
```

Because nearly all applications use these aliases by default, this affects almost the entire system. Thus, you can easily use your favorite fonts almost everywhere without having to modify the font settings in the individual applications.

Use the command `fc-list` to find out which fonts are installed and available for use. For instance, the command `fc-list` returns a list of all fonts. To find out which of the available scalable fonts (`:outline=true`) contain all glyphs required for Hebrew (`:lang=he`), their font names (`family`), their style (`style`), their weight (`weight`), and the name of the files containing the fonts, enter the following command:

```
fc-list ":lang=he:outline=true" family style weight
```

The output of this command could appear as follows:

```
FreeSansBold.ttf: FreeSans:style=Bold:weight=200
FreeMonoBoldOblique.ttf: FreeMono:style=BoldOblique:weight=200
```

```
FreeSerif.ttf: FreeSerif:style=Medium:weight=80
FreeSerifBoldItalic.ttf: FreeSerif:style=BoldItalic:weight=200
FreeSansOblique.ttf: FreeSans:style=Oblique:weight=80
FreeSerifItalic.ttf: FreeSerif:style=Italic:weight=80
FreeMonoOblique.ttf: FreeMono:style=Oblique:weight=80
FreeMono.ttf: FreeMono:style=Medium:weight=80
FreeSans.ttf: FreeSans:style=Medium:weight=80
FreeSerifBold.ttf: FreeSerif:style=Bold:weight=200
FreeSansBoldOblique.ttf: FreeSans:style=BoldOblique:weight=200
FreeMonoBold.ttf: FreeMono:style=Bold:weight=200
```

Important parameters that can be queried with `fc-list`:

Table 35.2 *Parameters of fc-list*

Parameter	**Meaning and Possible Values**
`family`	Name of the font family, for example, `FreeSans`.
`foundry`	The manufacturer of the font, for example, `urw`.
`style`	The font style, such as `Medium`, `Regular`, `Bold`, `Italic`, `Heavy`.
`lang`	The language that the font supports, for example, `de` for German, `ja` for Japanese, `zh-TW` for traditional Chinese, or `zh-CN` for simplified Chinese.
`weight`	The font weight, such as `80` for regular, `200` for bold.
`slant`	The slant, usually `0` for none and `100` for italic.
`file`	The name of the file containing the font.
`outline`	`true` for outline fonts, `false` for other fonts.
`scalable`	`true` for scalable fonts, `false` for other fonts.
`bitmap`	`true` for bitmap fonts, `false` for other fonts.
`pixelsize`	Font size in pixels. In connection with fc-list, this option only makes sense for bitmap fonts.

35.3.2 X11 Core Fonts

Today, the X11 core font system supports not only bitmap fonts but also scalable fonts, like Type1 fonts, TrueType and OpenType fonts, and CID-keyed fonts. Unicode fonts have also been supported for quite some time. In 1987, the X11 core font system was originally developed for X11R1 for the purpose of processing monochrome bitmap fonts. All extensions mentioned above were added later.

Scalable fonts are only supported without antialiasing and subpixel rendering and the loading of large scalable fonts with glyphs for many languages may take a long time. The use of Unicode fonts may also be slow and requires more memory.

The X11 core font system has a few inherent weaknesses. It is outdated and can no longer be extended in a meaningful fashion. Although it must be retained for reasons of backward compatibility, the more modern Xft and fontconfig system should be used if at all possible.

For its operation, the X server needs to know what fonts it has available and where in the system it can find them. This is handled by a FontPath variable, which contains the path to all valid system font directories. In each of these directories, a file named `fonts.dir` lists the available fonts in this directory. The FontPath is generated by the X server at start-up. It searches for a valid `fonts.dir` file in each of the `FontPath` entries in the configuration file `/etc/X11/xorg.conf`. These entries are found in the `Files` section. Display the actual FontPath with `xset q`. This path may also be changed at runtime with xset. To add an additional path, use `xset +fp <path>`. To remove an unwanted path, use `xset -fp <path>`.

If the X server is already active, newly installed fonts in mounted directories can be made available with the command `xset fp rehash`. This command is executed by `SuSEconfig --module fonts`. Because the command `xset` needs access to the running X server, this only works if `SuSEconfig --module fonts` is started from a shell that has access to the running X server. The easiest way to achieve this is to assume `root` permissions by entering `su` and the root password. `su` transfers the access permissions of the user who started the X server to the root shell. To check if the fonts were installed correctly and are available by way of the X11 core font system, use the command `xlsfonts` to list all available fonts.

By default, SUSE Linux uses UTF-8 locales. Therefore, Unicode fonts should be preferred (font names ending with `iso10646-1` in `xlsfonts` output). All available

Unicode fonts can be listed with `xlsfonts | grep iso10646-1`. Nearly all Unicode fonts available in SUSE Linux contain at least the glyphs needed for European languages (formerly encoded as `iso-8859-*`).

35.3.3 CID-Keyed Fonts

In contrast to the other font types, you cannot simply install CID-keyed fonts in just any directory. CID-keyed fonts must be installed in `/usr/share/ghostscript/Resource/CIDFont`. This is not relevant for Xft and fontconfig, but it is necessary for Ghostscript and the X11 core font system.

TIP

See `http://www.xfree86.org/current/fonts.html` for more information about fonts under X11.

35.4 OpenGL—3D Configuration

35.4.1 Hardware Support

SUSE Linux includes several OpenGL drivers for 3D hardware support. Table 35.3, "Supported 3D Hardware" (page 522) provides an overview.

Table 35.3 Supported 3D Hardware

OpenGL Driver	Supported Hardware
nVidia	nVidia Chips: all except Riva 128(ZX)
DRI	3Dfx Voodoo Banshee, 3Dfx Voodoo-3/4/5, Intel i810/i815/i830M, Intel 845G/852GM/855GM/865G/915,

OpenGL Driver	Supported Hardware
	Matrox G200/G400/G450/G550, ATI Rage 128(Pro)/Radeon (up to 9250)

If you are installing with YaST for the first time, 3D acceleration can be activated during installation, provided YaST detects 3D support. For nVidia graphics chips, the nVidia driver must be installed first. To do this, select the nVidia driver patch in YOU (YaST Online Update). Due to license restrictions, the nVidia driver is not included in the distribution.

If an update is carried out instead of a new installation or a 3Dfx add-on graphics adapter (Voodoo Graphics or Voodoo-2) needs to be set up, the procedure for configuring 3D hardware support is different. This depends on which OpenGL driver is used. Further details are provided in the following section.

35.4.2 OpenGL Drivers

The OpenGL drivers nVidia and DRI can be configured easily with SaX2. For nVidia adapters, the nVidia driver must be installed first. Enter the command `3Ddiag` to check if the configuration for nVidia or DRI is correct.

For security reasons, only users belonging to the group `video` are permitted to access the 3D hardware. Therefore, make sure that all local users are members of this group. Otherwise, the slow *software rendering fallback* of the OpenGL driver is used for OpenGL applications. Use the command `id` to check whether the current user belongs to the group `video`. If this is not the case, use YaST to add the user to the group.

35.4.3 The Diagnosis Tool 3Ddiag

The diagnosis tool 3Ddiag allows verification of the 3D configuration in SUSE Linux. This is a command line tool that must be started in a terminal. Enter `3Ddiag -h` to list possible options for 3Ddiag.

To verify the X.Org configuration, the tool checks if the packages needed for 3D support are installed and if the correct OpenGL library and GLX extension are used. Follow the instructions of 3Ddiag if you receive failed messages. If everything is correct, you only see done messages on the screen.

35.4.4 OpenGL Test Utilities

For testing OpenGL, the program `glxgears` and games like `tuxracer` and `armagetron` (packages have the same names) can be useful. If 3D support has been activated, it should be possible to play these smoothly on a fairly new computer. Without 3D support, these games would run very slowly (slideshow effect). Use the `glxinfo` command to verify that 3D is active, in which case the output contains a line with `direct rendering: Yes`.

35.4.5 Troubleshooting

If the OpenGL 3D test results are negative (the games cannot be smoothly played), use 3Ddiag to make sure no errors exist in the configuration (failed messages). If correcting these does not help or if failed messages have not appeared, take a look at the X.Org log files.

Often, you will find the line `DRI is disabled` in the X.Org file `/var/log/Xorg.0.log`. The exact cause can only be discovered by closely examining the log file—a task requiring some experience.

In such cases, no configuration error exists, because this would have already been detected by 3Ddiag. Consequently, at this point, the only choice is to use the software rendering fallback of the DRI driver, which does not provide 3D hardware support. You should also go without 3D support if you get OpenGL representation errors or instability. Use SaX2 to disable 3D support completely.

35.4.6 Installation Support

Apart from the `software rendering fallback` of the DRI driver, all OpenGL drivers in Linux are in developmental phases and are therefore considered experimental. The drivers are included in the distribution because of the high demand for 3D hardware acceleration in Linux. Considering the experimental status of OpenGL drivers, SUSE cannot offer any installation support for configuring 3D hardware acceleration or provide any further assistance with related problems. The basic configuration of the graphical user interface (X Window System) does not include 3D hardware acceleration configuration. If you experience problems with 3D hardware acceleration, it is recommended to disable 3D support completely.

35.4.7 Additional Online Documentation

For information about DRI, refer to `/usr/X11R6/lib/X11/doc/README.DRI` (`xorg-x11-doc`). More information about nvidia driver installation is found at `http://ftp.suse.com/pub/suse/i386/supplementary/X/nvidia-installer-HOWTO.html`.

Authentication with PAM 36

Linux uses PAM (Pluggable Authentication Modules) in the authentication process as a layer that mediates between user and application. PAM modules are available on a systemwide basis, so they can be requested by any application. This chapter describes how the modular authentication mechanism works and how it is configured.

System administrators and programmers often want to restrict access to certain parts of the system or to limit the use of certain functions of an application. Without PAM, applications must be adapted every time a new authentication mechanism, such as LDAP or SAMBA, is introduced. This process, however, is rather time-consuming and error-prone. One way to avoid these drawbacks is to separate applications from the authentication mechanism and to delegate the latter to centrally managed modules. Whenever a newly required authentication scheme is needed, it is sufficient to adapt or write a suitable PAM module for use by the program in question.

Every program that relies on the PAM mechanism has its own configuration file in the directory `/etc/pam.d/`*`programname`*. These files define the PAM modules used for authentication. In addition, there are global configuration files for most PAM modules under `/etc/security`, which define the exact behavior of these modules (examples include `pam_env.conf`, `pam_pwcheck.conf`, `pam_unix2.conf`, and `time.conf`). Every application that uses a PAM module actually calls a set of PAM functions, which then process the information in the various configuration files and return the result to the calling application.

36.1 Structure of a PAM Configuration File

Each line in a PAM configuration file contains a maximum of four columns:

```
<Type of module> <Control flag> <Module path> <Options>
```

PAM modules are processed as stacks. Different types of modules have different purposes, for example, one module checks the password, another one verifies the location from which the system is accessed, and yet another one reads user-specific settings. PAM knows about four different types of modules:

`auth`
The purpose of this type of module is to check the user's authenticity. This is traditionally done by querying a password, but it can also be achieved with the help of a chip card or through biometrics (fingerprints or iris scan).

`account`
Modules of this type check whether the user has general permission to use the requested service. As an example, such a check should be performed to ensure that no one can log in under the username of an expired account.

`password`
The purpose of this type of module is to enable the change of an authentication token. In most cases, this is a password.

`session`
Modules of this type are responsible for managing and configuring user sessions. They are started before and after authentication to register login attempts in system logs and to configure the user's specific environment (mail accounts, home directory, system limits, etc.).

The second column contains control flags to influence the behavior of the modules started:

`required`
A module with this flag must be successfully processed before the authentication may proceed. After the failure of a module with the `required` flag, all other

modules with the same flag are processed before the user receives a message about the failure of the authentication attempt.

requisite
Modules having this flag must also be processed successfully, in much the same way as a module with the `required` flag. However, in case of failure a module with this flag gives immediate feedback to the user and no further modules are processed. In case of success, other modules are subsequently processed, just like any modules with the `required` flag. The `requisite` flag can be used as a basic filter checking for the existence of certain conditions that are essential for a correct authentication.

sufficient
After a module with this flag has been successfully processed, the calling application receives an immediate message about the success and no further modules are processed, provided there was no preceding failure of a module with the `required` flag. The failure of a module with the `sufficient` flag has no direct consequences, in the sense that any subsequent modules are processed in their respective order.

optional
The failure or success of a module with this flag does not have any direct consequences. This can be useful for modules that are only intended to display a message (for example, to tell the user that mail has arrived) without taking any further action.

include
If this flag is given, the file specified as argument is inserted at this place.

The module path does not need to be specified explicitly, as long as the module is located in the default directory `/lib/security` (for all 64-bit platforms supported by SUSE Linux, the directory is `/lib64/security`). The fourth column may contain an option for the given module, such as `debug` (enables debugging) or `nullok` (allows the use of empty passwords).

36.2 The PAM Configuration of sshd

To show how the theory behind PAM works, consider the PAM configuration of sshd as a practical example:

Example 36.1 *PAM Configuration for sshd*

```
#%PAM-1.0
auth     include        common-auth
auth     required       pam_nologin.so
account  include        common-account
password include        common-password
session  include        common-session
# Enable the following line to get resmgr support for
# ssh sessions (see /usr/share/doc/packages/resmgr/README.SuSE)
#session  optional      pam_resmgr.so fake_ttyname
```

The typical PAM configuration of an application (sshd, in this case) contains four include statements referring to the configuration files of four module types: common-auth, common-account, common-password, and common-session. These four files hold the default configuration for each module type. By including them instead of calling each module separately for each PAM application, automatically get an updated PAM configuration if the administrator changes the defaults. In former times, you had to adjust all configuration files manually for all applications when changes to PAM occured or a new application was installed. Now the PAM configuration is made with central configuration files and all changes are automatically inherited by the PAM configuration of each service.

The first include file (common-auth) calls two modules of the auth type: pam_env and pam_unix2. See Example 36.2, "Default Configuration for the auth Section" (page 530).

Example 36.2 *Default Configuration for the auth Section*

```
auth    required        pam_env.so
auth    required        pam_unix2.so
```

The first one, pam_env, loads the file /etc/security/pam_env.conf to set the environment variables as specified in this file. This can be used to set the DISPLAY variable to the correct value, because the pam_env module knows about the location from which the login is taking place. The second one, pam_unix2, checks the user's login and password against /etc/passwd and /etc/shadow.

After the modules specified in common-auth have been successfully called, a third module called pam_nologin checks whether the file /etc/nologin exists. If it does, no user other than root may log in. The whole stack of auth modules is processed before sshd gets any feedback about whether the login has succeeded. Given that all modules of the stack have the required control flag, they must all be processed successfully before sshd receives a message about the positive result. If one of the modules is not successful, the entire module stack is still processed and only then is sshd notified about the negative result.

As soon as all modules of the auth type have been successfully processed, another include statement is processed, in this case, that in Example 36.3, “Default Configuration for the account Section” (page 531). common-account contains just one module, pam_unix2. If pam_unix2 returns the result that the user exists, sshd receives a message announcing this success and the next stack of modules (password) is processed, shown in Example 36.4, “Default Configuration for the password Section” (page 531).

Example 36.3 *Default Configuration for the account Section*

```
account required        pam_unix2.so
```

Example 36.4 *Default Configuration for the password Section*

```
password required       pam_pwcheck.so  nullok
password required       pam_unix2.so    nullok use_first_pass use_authtok
#password required      pam_make.so     /var/yp
```

Again, the PAM configuration of sshd involves just an include statement referring to the default configuration for password modules located in common-password. These modules must successfully be completed (control flag required) whenever the application requests the change of an authentication token. Changing a password or another authentication token requires a security check. This is achieved with the pam_pwcheck module. The pam_unix2 module used afterwards carries over any old and new passwords from pam_pwcheck, so the user does not need to authenticate again. This also makes it impossible to circumvent the checks carried out by pam_pwcheck. The modules of the password type should be used wherever the preceding modules of the account or the auth type are configured to complain about an expired password.

Example 36.5 *Default Configuration for the session Section*

```
session required        pam_limits.so
session required        pam_unix2.so
```

As the final step, the modules of the session type, bundled in the common-session file are called to configure the session according to the settings for the user in question. Although pam_unix2 is processed again, it has no practical consequences due to its none option specified in the respective configuration file of this module, pam_unix2.conf. The pam_limits module loads the file /etc/security/limits.conf, which may define limits on the use of certain system resources. The session modules are called a second time when user logs out.

36.3 Configuration of PAM Modules

Some of the PAM modules are configurable. The corresponding configuration files are located in /etc/security. This section briefly describes the configuration files relevant to the sshd example—pam_unix2.conf, pam_env.conf, pam_pwcheck.conf, and limits.conf.

36.3.1 pam_unix2.conf

The traditional password-based authentication method is controlled by the PAM module pam_unix2. It can read the necessary data from /etc/passwd, /etc/shadow, NIS maps, NIS+ tables, or from an LDAP database. The behavior of this module can be influenced by configuring the PAM options of the individual application itself or globally by editing /etc/security/pam_unix2.conf. A very basic configuration file for the module is shown in Example 36.6, "pam_unix2.conf" (page 532).

Example 36.6 *pam_unix2.conf*

```
auth:   nullok
account:
password:       nullok
session:        none
```

The nullok option for module types auth and password specifies that empty passwords are permitted for the corresponding type of account. Users are also allowed to change passwords for their accounts. The none option for the module type session specifies that no messages are logged on its behalf (this is the default). Learn about additional configuration options from the comments in the file itself and from the manual page pam_unix2(8).

36.3.2 pam_env.conf

This file can be used to define a standardized environment for users that is set whenever the pam_env module is called. With it, preset environment variables using the following syntax:

```
VARIABLE   [DEFAULT=[value]]   [OVERRIDE=[value]]
```

VARIABLE
Name of the environment variable to set.

[DEFAULT=[value]]
Default value the administrator wants set.

[OVERRIDE=[value]]
Values that may be queried and set by pam_env, overriding the default value.

A typical example of how pam_env can be used is the adaptation of the DISPLAY variable, which is changed whenever a remote login takes place. This is shown in Example 36.7, “pam_env.conf” (page 533).

Example 36.7 *pam_env.conf*

```
REMOTEHOST     DEFAULT=localhost OVERRIDE=@{PAM_RHOST}
DISPLAY        DEFAULT=${REMOTEHOST}:0.0 OVERRIDE=${DISPLAY}
```

The first line sets the value of the REMOTEHOST variable to localhost, which is used whenever pam_env cannot determine any other value. The DISPLAY variable in turn contains the value of REMOTEHOST. Find more information in the comments in the file /etc/security/pam_env.conf.

36.3.3 pam_pwcheck.conf

This configuration file is for the pam_pwcheck module, which reads options from it for all password type modules. Settings stored in this file take precedence over the PAM settings of an individual application. If application-specific settings have not been defined, the application uses the global settings. Example 36.8, “pam_pwcheck.conf” (page 533) tells pam_pwcheck to allow empty passwords and modification of passwords. More options for the module are mentioned in the file /etc/security/pam_pwcheck.conf.

Example 36.8 *pam_pwcheck.conf*

```
password:    nullok
```

36.3.4 limits.conf

System limits can be set on a user or group basis in the file `limits.conf`, which is read by the `pam_limits` module. The file allows you to set hard limits, which may not be exceeded at all, and soft limits, which may be exceeded temporarily. To learn about the syntax and the available options, read the comments included in the file.

36.4 For More Information

In the directory `/usr/share/doc/packages/pam` of your installed system, find the following additional documentation:

READMEs
In the top level of this directory, there are some general README files. The subdirectory `modules` holds README files about the available PAM modules.

The Linux-PAM System Administrators' Guide
This document includes everything that a system administrator should know about PAM. It discusses a range of topics, from the syntax of configuration files to the security aspects of PAM. The document is available as a PDF file, in HTML format, and as plain text.

The Linux-PAM Module Writers' Manual
This document summarizes the topic from the developer's point of view, with information about how to write standard-compliant PAM modules. It is available as a PDF file, in HTML format, and as plain text.

The Linux-PAM Application Developers' Guide
This document includes everything needed by an application developer who wants to use the PAM libraries. It is available as a PDF file, in HTML format, and as plain text.

Thorsten Kukuk has developed a number of PAM modules for SUSE Linux and made some information available about them at `http://www.suse.de/~kukuk/pam/` .

37 Virtualization with Xen

Xen makes it possible to run several Linux systems on one physical machine. The hardware for the different systems is provided virtually. This chapter gives an overview of the possibilities and limitations of this technology. Sections about installing, configuring, and running Xen complete this introduction.

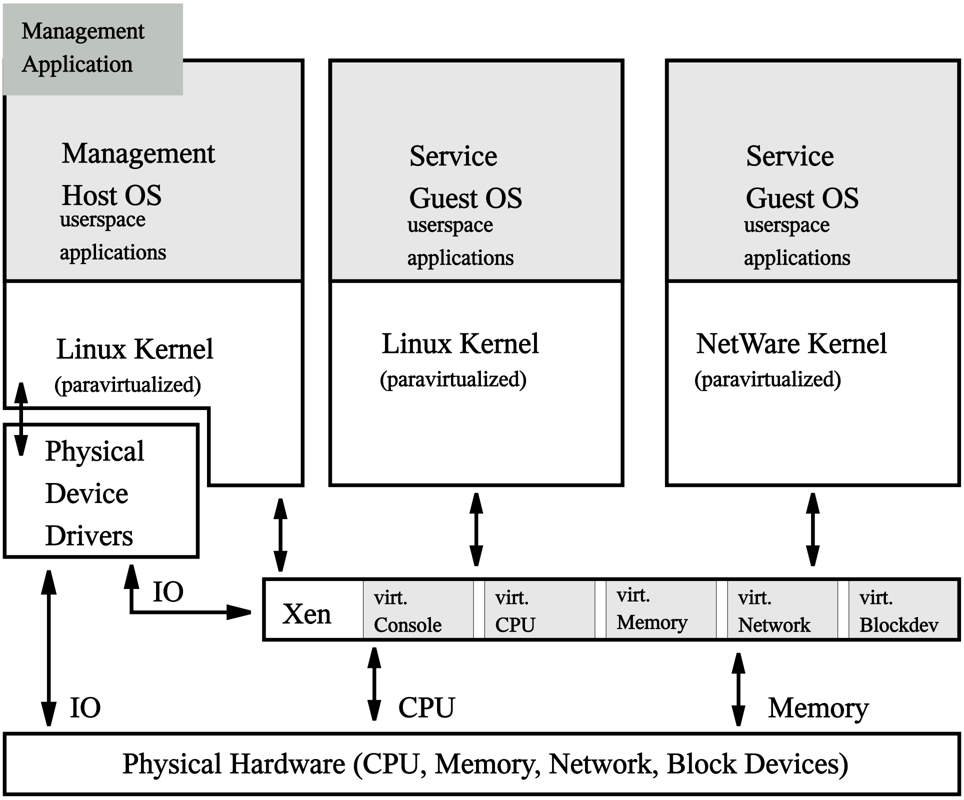
Virtual machines commonly need to emulate the hardware a system needs. The disadvantage is that the emulated hardware is much slower than the real silicon. Xen has a different approach. It restricts emulation to as few parts as possible. To achieve this, Xen uses *paravirtualization*. This is a technique that presents virtual machines similarly, but not identically to the underlying hardware. Therefore, host and guest operating systems are adapted on kernel level. The user space remains unchanged. Xen controls the hardware with a hypervisor and a controlling guest, also called domain-0. These provide all needed virtualized block and network devices. The guest systems use these virtual block and network devices to run the system and connect to other guests or the local network. When several physical machines running Xen are configured in a way that the virtual block and network devices are available, it is also possible to migrate a guest system from one piece of hardware to another while running. Originally, Xen was developed to run up to 100 guest systems on one computer, but this number depends strongly on the system requirements of the running guest systems, especially the memory consumption.

To limit the CPU utilization, the Xen hypervisor offers three different schedulers. The scheduler also may be changed while running the guest system, making it is possible to change the priority of the running guest system. On a higher level, migrating a guest may also be used to adjust the available CPU power.

The Xen virtualization system also has some drawbacks regarding the supported hardware:

- Several closed source drivers, such as those from Nvidia or ATI, do not work as expected. In these cases, you must use the open source drivers if available, even if they do not support the full capabilities of the chips. Also several WLAN chips and cardbus bridges are not supported when using Xen.
- In version 2, Xen does not support PAE (physical address extension), which means that it does not support more than 4 GB of memory.
- There is no support for ACPI. Power management and other modes that depend on ACPI do not work.

Figure 37.1 *Xen Overview*



37.1 Xen Installation

The installation procedure of Xen involves the setup of a domain-0 domain and the installation of Xen clients. First, make sure that the needed packages are installed. These are `python`, `bridge-utils`, `xen`, and a `kernel-xen` package. When using SUSE packages, Xen is added to the GRUB configuration. For other cases, make an entry in `boot/grub/menu.lst`. This entry should be similar to the following:

```
title Xen2
    kernel (hd0,0)/boot/xen.gz dom0_mem=458752
    module (hd0,0)/boot/vmlinuz-xen <parameters>
    module (hd0,0)/boot/initrd-xen
```

Replace (hd0,0) with the partition that holds your `/boot` directory. See also Chapter 29, *The Boot Loader* (page 427). Alter the amount of dom0_mem to match your system. The maximum value is your system memory in kB minus 65536. Replace <parameters> with the parameters normally used to boot a Linux kernel. Then reboot into Xen mode. This boots the Xen hypervisor and a slightly changed Linux kernel as Domain-0 that runs most of the hardware. Apart from the exceptions already mentioned, everything should work as normal.

37.2 Domain Installation

The installation and setup of a guest domain involves several procedures. In the following, a first guest domain is installed and all the different tasks to create a first network connection are completed.

To install a guest system, you must provide a root file system in a block device or in a file system image, which needs to be set up. To access this system later, use an emulated console or set up the network connection for this guest. The installation of SUSE Linux into a directory is supported by YaST. The hardware requirements of such a guest are similar to a normal Linux installation.

Domains can share file systems that are mounted read-only from all domains, such as `/usr` or `/opt`. Never share a file system that is mounted read-write. For sharing writable data among several guest domains, use NFS or other networked or cluster file systems.

WARNING: Starting a Guest Domain

When you start a guest domain, make sure that the file systems of the guest are not mounted anymore by an installer or by the controlling domain-0.

The first thing to do is to create a file system image in which the Linux for the guest is installed:

1. To create an empty image named `guest1` in the directory `/var/tmp/` that is 4 GB size, use the following command:

   ```
   dd if=/dev/zero of=/var/tmp/guest1 seek=1M bs=4096 count=1
   ```

2. The image is just a big empty file without any information in it. To be able to write files into it, a file system is needed:

   ```
   mkreiserfs -f /var/tmp/guest1
   ```

   The command `mkreiserfs` informs you that this is not a block special device and asks for a confirmation. Enter Y then Enter to continue.

3. The actual installation is made in a directory. Therefore the file system image `/var/tmp/guest1` must be mounted to a directory:

   ```
   mkdir -p /var/tmp/dirinstall
   mount -o loop /var/tmp/guest1 /var/tmp/dirinstall
   ```

IMPORTANT

When you are finished with the installation, unmount this file system image again. YaST also mounts the `/proc` file system when installing, which must be unmounted as well:

```
umount /var/tmp/dirinstall/proc
umount /var/tmp/dirinstall
```

37.2.1 Using YaST to Install a Guest Domain

To install a guest domain with YaST, you need the previously prepared the file system image for the new guest. Start YaST and select *Software → Installation into Directory for XEN*.

The YaST module for directory installation has several options that should be set according your needs:

- Target Directory: `/var/tmp/dirinstall`

  Set this option to the mount point of the file system image to use. The default is usually acceptable.

- Run YaST and SuSEconfig at First Boot: Yes

  Set this option to *Yes*. You will be asked for a root password and a first user when starting the guest for the first time.

- Create Image: No

  The image this creates is just a tar archive of the installation directory. This is not useful here.

- Software

  Select the type of installation to use. Any of the defaults should be a good start.

Click *Next* to start the installation. Depending on the number of packages, the installation takes a while. After the installation has finished, the tls libraries must be moved away:

```
mv /var/tmp/dirinstall/lib/tls /var/tmp/dirinstall/lib/tls.disabled
```

Xen uses one of the kernels that are installed in domain-0 to start the guest domain. To be able to use networking in the guest, the modules of this kernel must be available for the guest as well.

```
cp -a /lib/modules/$(rpm -qf --qf %{VERSION}-%{RELEASE}-xen \
    /boot/vmlinuz-xen) /var/tmp/dirinstall/lib/modules
```

To prevent file system errors, the file system image must to be unmounted after the installation:

```
umount /var/tmp/dirinstall/proc
umount /var/tmp/dirinstall/
```

It would be possible to build specialized kernels for domain-0 on one hand and for the guest systems on the other hand. The main difference are the hardware drivers that are unneeded in guest systems. Because these drivers are modular and not used in the guest systems, SUSE delivers only one kernel for both tasks.

37.2.2 Setting Up a Rescue System to Work as a Guest Domain

The easiest way to get a running system quickly is to reuse an existing root file system, such as the rescue system of SUSE Linux. Basically, exchange the kernel image and the device drivers of the virtual block and network devices in this image. To make this task easier, the script mk-xen-rescue-img.sh is available in /usr/share/doc/packages/xen/.

The disadvantage of using the rescue method of constructing a root file system is that the result does not have an RPM database, so you cannot easily add packages using RPM. On the positive side, the result is relatively small but has most of what is needed to get started with networking.

To run the script mk-xen-rescue-img.sh, you need at least the directory with the rescue image and a destination location for the resulting image. By default, the directory resides on the boot DVD in the directory /boot.

```
cd /usr/share/doc/packages/xen
./mk-xen-rescue-img.sh /media/dvd/boot /usr/local/xen 64
```

The first parameter of the script is the directory of the rescue image. The second parameter is the destination of the image file. Optional parameters are the disk space requirements of the newly generated guest domain and the kernel version to use.

The script then copies the image to the new location, replaces the kernel and several kernel modules, and disables the tls directory in the system. As a last step, it generates a configuration file for the new image in /etc/xen/.

37.3 Configuring a Xen Guest Domain

The documentation about how to configure a guest domain is not very exhaustive. The most information about how to configure such a domain can be found in the example configuration file /etc/xen/config. The needed options are explained together with a default value or at least an example configuration. For the installation described in Section 37.2.1, “Using YaST to Install a Guest Domain” (page 538), create a file /etc/xen/guest1 with the following content:

```
kernel = "/boot/vmlinuz-xen"    ❶
ramdisk = "/boot/initrd-xen"    ❷
memory = 128                    ❸
name = "guest1"                 ❹
nics = "1"                      ❺
vif = [ 'mac=aa:cc:00:00:00:ab, bridge=xen-br0' ] ❻
disk = [ 'file:/var/tmp/guest1,hda1,w' ] ❼
root = "/dev/hda1 ro"           ❽
extra = "3"                     ❾
```

❶ Enter the path to the Xen kernel in domain-0. This kernel will run in the guest system later.

❷ Select the appropriate initial RAM disk that contains the device drivers for the Xen kernel. Without this, the kernel typically panics because it is unable to mount its root file system.

❸ Define how much memory the guest domain should be given. This fails if the system does not have enough memory available for its guests.

❹ The name for this guest.

❺ The number of virtual network interfaces for the guest domain.

❻ The configuration of the virtual network interface, including its MAC address and the bridge to which it is connected.

❼ Set the available virtual block devices for the Xen guest. To use real block devices, create entries like `['phy:sdb1,hda1,w', 'phy:system/swap1,hda2,w']`.

❽ Sets the root device for the kernel. This must be the virtual device as seen by the guest.

❾ Add extra kernel parameters here. The example `3` means that the guest is started in runlevel 3.

37.4 Starting and Controlling Xen Domains

Before the guest domain may be started, the Xen hypervisor must have enough free memory for the new guest. First, check the amount of memory used:

```
xm list
Name                  Id  Mem(MB)  CPU  State  Time(s)  Console
Domain-0               0      458    0  r----    181.8
```

If this is a computer with 512 MB, the Xen hypervisor takes away 64 MB and Domain-0 occupies the rest. To free some of the memory for the new guest, the command `xm balloon` is used. To set the size of Domain-0 to 330 MB, enter the following as `root`:

```
xm balloon 0 330
```

In the next `xm list`, the memory usage of Domain-0 should have dropped to 330 MB. Now there is enough memory available to start a guest with 128 MB. The command `xm start guest1 -c` starts the guest and links the console of the starting guest to the current terminal. If this is the first time that this guest starts, finish the installation with YaST.

It is always possible to detach this console or reattach it from another terminal. To detach, use Ctrl +]. To reattach, first check the ID of the needed guest with `xm list` and attach to that ID with `xm console` *`ID`*.

The xm tool of Xen has many possible parameters. View a list with a short explanation by entering `xm help`. Table 37.1, "xm Commands" (page 542) provides some of the most important commands as a starting point.

***Table 37.1** xm Commands*

`xm help`	Print a list of commands that are available for the xm tool.
`xm console` *`ID`*	Connect to the first console (tty1) of the guest with ID *`ID`*.
`xm balloon` *`ID Mem`*	Set the memory size of the domain with ID *`ID`* to *`Mem`* in MB.

`xm create` *`domname`* `[-c]`	Start the domain with configuration file *`domname`*. The optional `-c` links the current terminal to the first tty of the new guest.
`xm shutdown` *`ID`*	Do a normal shutdown of the guest with ID *`ID`*.
`xm destroy` *`ID`*	Terminate the guest with ID *`ID`* immediately.
`xm list`	Print a list of all running domains with their respective ID, memory, and CPU time values.
`xm info`	Display information about the Xen host, including CPU and memory information.

37.5 For More Information

More information about Xen can be found on the following Web sites:

- `file:/usr/share/doc/packages/xen/user/html/index.html`—Official information for Xen users. It requires the package `xen-doc-html`.
- `file:/usr/share/doc/packages/xen/interface/html/index.html`—Some more technical interface documentation. It also requires the package `xen-doc-html`.
- `http://www.cl.cam.ac.uk/Research/SRG/netos/xen/index.html`—Xen home page with many different documentation links.
- `http://lists.xensource.com/`—Several mailing lists about Xen.

Part IX Services

38 Basic Networking

Linux offers the necessary networking tools and features for integration into all types of network structures. The customary Linux protocol, TCP/IP, has various services and special features, which are discussed here. Network access using a network card, modem, or other device can be configured with YaST. Manual configuration is also possible. Only the fundamental mechanisms and the relevant network configuration files are discussed in this chapter.

Linux and other Unix operating systems use the TCP/IP protocol. It is not a single network protocol, but a family of network protocols that offer various services. The protocols listed in Table 38.1, “Several Protocols in the TCP/IP Protocol Family” (page 547) are provided for the purpose of exchanging data between two machines via TCP/IP. Networks combined by TCP/IP, comprising a worldwide network are also referred to, in their entirety, as “the Internet.”

RFC stands for *Request for Comments*. RFCs are documents that describe various Internet protocols and implementation procedures for the operating system and its applications. The RFC documents describe the setup of Internet protocols. To expand your knowledge about any of the protocols, refer to the appropriate RFC documents. They are available online at `http://www.ietf.org/rfc.html`.

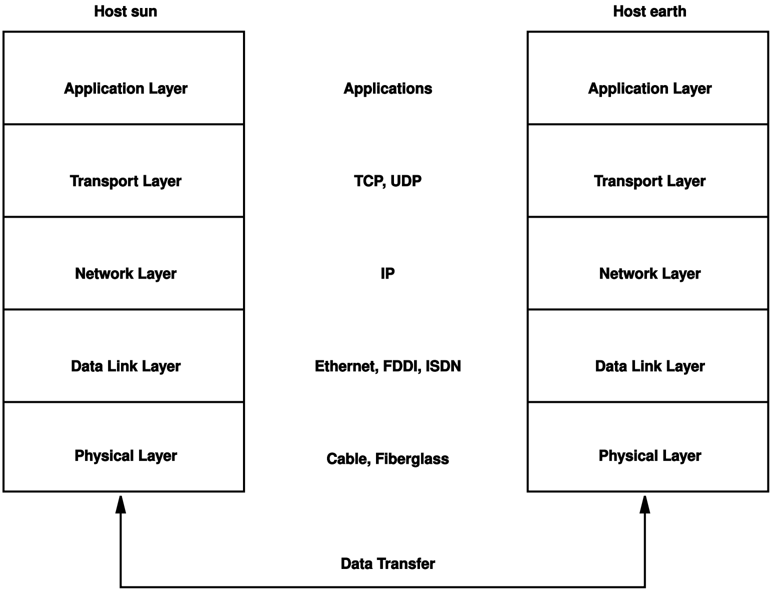
Table 38.1 *Several Protocols in the TCP/IP Protocol Family*

Protocol	Description
TCP	Transmission Control Protocol: A connection-oriented secure protocol. The data to transmit is first sent by the application as a stream of data

Protocol	Description
	then converted by the operating system to the appropriate format. The data arrives at the respective application on the destination host in the original data stream format in which it was initially sent. TCP determines whether any data has been lost during the transmission and that there is no mix-up. TCP is implemented wherever the data sequence matters.
UDP	User Datagram Protocol: A connectionless, insecure protocol. The data to transmit is sent in the form of packets generated by the application. The order in which the data arrives at the recipient is not guaranteed and data loss is a possibility. UDP is suitable for record-oriented applications. It features a smaller latency period than TCP.
ICMP	Internet Control Message Protocol: Essentially, this is not a protocol for the end user, but a special control protocol that issues error reports and can control the behavior of machines participating in TCP/IP data transfer. In addition, it provides a special echo mode that can be viewed using the program ping.
IGMP	Internet Group Management Protocol: This protocol controls machine behavior when implementing IP multicast.

As shown in Figure 38.1, “Simplified Layer Model for TCP/IP” (page 549), data exchange takes place in different layers. The actual network layer is the insecure data transfer via IP (Internet protocol). On top of IP, TCP (transmission control protocol) guarantees, to a certain extent, security of the data transfer. The IP layer is supported by the underlying hardware-dependent protocol, such as ethernet.

Figure 38.1 Simplified Layer Model for TCP/IP

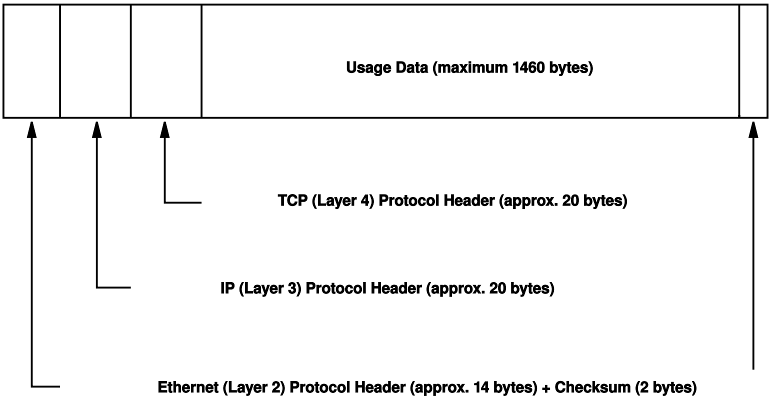


The diagram provides one or two examples for each layer. The layers are ordered according to *abstraction levels*. The lowest layer is very close to the hardware. The uppermost layer, however, is almost a complete abstraction from the hardware. Every layer has its own special function. The special functions of each layer are mostly implicit in their description. The data link and physical layers represent the physical network used, such as ethernet.

Almost all hardware protocols work on a packet-oriented basis. The data to transmit is packaged in *packets*, because it cannot be sent all at once. The maximum size of a TCP/IP packet is approximately 64 KB. Packets are normally quite a bit smaller, because the network hardware can be a limiting factor. The maximum size of a data packet on an ethernet is about fifteen hundred bytes. The size of a TCP/IP packet is limited to this amount when the data is sent over an ethernet. If more data is transferred, more data packets need to be sent by the operating system.

For the layers to serve their designated functions, additional information regarding each layer must be saved in the data packet. This takes place in the *header* of the packet. Every layer attaches a small block of data, called the protocol header, to the front of each emerging packet. A sample TCP/IP data packet traveling over an ethernet cable is illustrated in Figure 38.2, "TCP/IP Ethernet Packet" (page 550). The proof sum is located at the end of the packet, not at the beginning. This simplifies things for the network hardware.

Figure 38.2 *TCP/IP Ethernet Packet*


Usage Data (maximum 1460 bytes)
TCP (Layer 4) Protocol Header (approx. 20 bytes)
IP (Layer 3) Protocol Header (approx. 20 bytes)
Ethernet (Layer 2) Protocol Header (approx. 14 bytes) + Checksum (2 bytes)

When an application sends data over the network, the data passes through each layer, all implemented in the Linux kernel except the physical layer. Each layer is responsible for preparing the data so it can be passed to the next layer. The lowest layer is ultimately responsible for sending the data. The entire procedure is reversed when data is received. Like the layers of an onion, in each layer the protocol headers are removed from the transported data. Finally, the transport layer is responsible for making the data available for use by the applications at the destination. In this manner, one layer only communicates with the layer directly above or below it. For applications, it is irrelevant whether data is transmitted via a 100 MBit/s FDDI network or via a 56-kbit/s modem line. Likewise, it is irrelevant for the data line which kind of data is transmitted, as long as packets are in the correct format.

38.1 IP Addresses and Routing

The discussion in this section is limited to IPv4 networks. For information about IPv6 protocol, the successor to IPv4, refer to Section 38.2, “IPv6—The Next Generation Internet” (page 553).

38.1.1 IP Addresses

Every computer on the Internet has a unique 32-bit address. These 32 bits (or 4 bytes) are normally written as illustrated in the second row in Example 38.1, "Writing IP Addresses" (page 551).

Example 38.1 *Writing IP Addresses*

```
IP Address (binary):  11000000 10101000 00000000 00010100
IP Address (decimal):      192.     168.       0.      20
```

In decimal form, the four bytes are written in the decimal number system, separated by periods. The IP address is assigned to a host or a network interface. It cannot be used anywhere else in the world. There are exceptions to this rule, but these are not relevant in the following passages.

The points in IP addresses indicate the hierarchical system. Until the 1990s, IP addresses were strictly categorized in classes. However, this system has proven too inflexible and was discontinued. Now, *classless routing* (CIDR, classless interdomain routing) is used.

38.1.2 Netmasks and Routing

Netmasks are used to define the address range of a subnetwork. If two hosts are in the same subnetwork, they can reach each other directly, if they are not in the same subnetwork, they need the address of a gateway that handles all the traffic between the subnetwork and the rest of the world. To check if two IP addresses are in the same subnet, simply "AND" both addresses with the netmask. If the result is identical, both IP addresses are in the same local network. If there are differences, the remote IP address, and thus the remote interface, can only be reached over a gateway.

To understand how the netmask works, look at Example 38.2, "Linking IP Addresses to the Netmask" (page 552). The netmask consists of 32 bits that identify how much of an IP address belongs to the network. All those bits that are `1` mark the corresponding bit in the IP address as belonging to the network. All bits that are `0` mark bits inside the subnetwork. This means that the more bits are `1`, the smaller the subnetwork is. Because the netmask always consists of several successive `1` bits, it is also possible to just count the number of bits in the netmask. In Example 38.2, "Linking IP Addresses to the Netmask" (page 552) the first net with 24 bits could also be written as `192.168.0.0/24`.

Example 38.2 *Linking IP Addresses to the Netmask*

```
IP address (192.168.0.20):  11000000 10101000 00000000 00010100
Netmask   (255.255.255.0):  11111111 11111111 11111111 00000000
---------------------------------------------------------------
Result of the link:         11000000 10101000 00000000 00000000
In the decimal system:           192.     168.       0.       0

IP address (213.95.15.200): 11010101 10111111 00001111 11001000
Netmask    (255.255.255.0): 11111111 11111111 11111111 00000000
---------------------------------------------------------------
Result of the link:         11010101 10111111 00001111 00000000
In the decimal system:           213.      95.      15.       0
```

To give another example: all machines connected with the same ethernet cable are usually located in the same subnetwork and are directly accessible. Even when the subnet is physically divided by switches or bridges, these hosts can still be reached directly.

IP addresses outside the local subnet can only be reached if a gateway is configured for the target network. In the most common case, there is only one gateway that handles all traffic that is external. However, it is also possible to configure several gateways for different subnets.

If a gateway has been configured, all external IP packets are sent to the appropriate gateway. This gateway then attempts to forward the packets in the same manner—from host to host—until it reaches the destination host or the packet's TTL (time to live) expires.

Table 38.2 *Specific Addresses*

Address Type	Description
Base Network Address	This is the netmask AND any address in the network, as shown in Example 38.2, “Linking IP Addresses to the Netmask” (page 552) under `Result`. This address cannot be assigned to any hosts.
Broadcast Address	This basically says, “Access all hosts in this subnetwork.” To generate this, the netmask is inverted in binary form and linked to the base network address with a logical OR. The above example therefore results in 192.168.0.255. This address cannot be assigned to any hosts.

Address Type	Description
Local Host	The address `127.0.0.1` is assigned to the “loopback device” on each host. A connection can be set up to your own machine with this address.

Because IP addresses must be unique all over the world, you cannot just select random addresses. There are three address domains to use if you want to set up a private IP-based network. These cannot get any connection from the rest of the Internet, because they cannot be transmitted over the Internet. These address domains are specified in RFC 1597 and listed in Table 38.3, “Private IP Address Domains” (page 553).

Table 38.3 *Private IP Address Domains*

Network/Netmask	Domain
`10.0.0.0/255.0.0.0`	`10.x.x.x`
`172.16.0.0/255.240.0.0`	`172.16.x.x – 172.31.x.x`
`192.168.0.0/255.255.0.0`	`192.168.x.x`

38.2 IPv6—The Next Generation Internet

Due to the emergence of the WWW (World Wide Web), the Internet has experienced explosive growth with an increasing number of computers communicating via TCP/IP in the past fifteen years. Since Tim Berners-Lee at CERN (`http://public.web.cern.ch`) invented the WWW in 1990, the number of Internet hosts has grown from a few thousand to about a hundred million.

As mentioned, an IPv4 address consists of only 32 bits. Also, quite a few IP addresses are lost—they cannot be used due to the way in which networks are organized. The number of addresses available in your subnet is two to the power of the number of bits, minus two. A subnetwork has, for example, 2, 6, or 14 addresses available. To connect 128 hosts to the Internet, for example, you need a subnetwork with 256 IP addresses, from which only 254 are usable, because two IP addresses are needed for the structure of the subnetwork itself: the broadcast and the base network address.

Under the current IPv4 protocol, DHCP or NAT (network address translation) are the typical mechanisms used to circumvent the potential address shortage. Combined with the convention to keep private and public address spaces separate, these methods can certainly mitigate the shortage. The problem with them lies in their configuration, which is a chore to set up and a burden to maintain. To set up a host in an IPv4 network, you need a number of address items, such as the host's own IP address, the subnetmask, the gateway address, and maybe a name server address. All these items need to be known and cannot be derived from somewhere else.

With IPv6, both the address shortage and the complicated configuration should be a thing of the past. The following sections tell more about the improvements and benefits brought by IPv6 and about the transition from the old protocol to the new one.

38.2.1 Advantages

The most important and most visible improvement brought by the new protocol is the enormous expansion of the available address space. An IPv6 address is made up of 128 bit values instead of the traditional 32 bits. This provides for as many as several quadrillion IP addresses.

However, IPv6 addresses are not only different from their predecessors with regard to their length. They also have a different internal structure that may contain more specific information about the systems and the networks to which they belong. More details about this are found in Section 38.2.2, “Address Types and Structure” (page 555).

The following is a list of some other advantages of the new protocol:

Autoconfiguration

IPv6 makes the network “plug and play” capable, which means that a newly set up system integrates into the (local) network without any manual configuration. The new host uses its automatic configuration mechanism to derive its own address from the information made available by the neighboring routers, relying on a protocol called the *neighbor discovery* (ND) protocol. This method does not require any intervention on the administrator's part and there is no need to maintain a central server for address allocation—an additional advantage over IPv4, where automatic address allocation requires a DHCP server.

Mobility

IPv6 makes it possible to assign several addresses to one network interface at the same time. This allows users to access several networks easily, something that could be compared with the international roaming services offered by mobile phone companies: when you take your mobile phone abroad, the phone automatically logs in to a foreign service as soon as it enters the corresponding area, so you can be reached under the same number everywhere and are able to place an outgoing call just like in your home area.

Secure Communication

With IPv4, network security is an add-on function. IPv6 includes IPSec as one of its core features, allowing systems to communicate over a secure tunnel to avoid eavesdropping by outsiders on the Internet.

Backward Compatibility

Realistically, it would be impossible to switch the entire Internet from IPv4 to IPv6 at one time. Therefore, it is crucial that both protocols are able to coexist not only on the Internet, but also on one system. This is ensured by compatible addresses (IPv4 addresses can easily be translated into IPv6 addresses) and through the use of a number of tunnels. See Section 38.2.3, “Coexistence of IPv4 and IPv6” (page 560). Also, systems can rely on a *dual stack IP* technique to support both protocols at the same time, meaning that they have two network stacks that are completely separate, such that there is no interference between the two protocol versions.

Custom Tailored Services through Multicasting

With IPv4, some services, such as SMB, need to broadcast their packets to all hosts in the local network. IPv6 allows a much more fine-grained approach by enabling servers to address hosts through *multicasting*—by addressing a number of hosts as parts of a group (which is different from addressing all hosts through *broadcasting* or each host individually through *unicasting*). Which hosts are addressed as a group may depend on the concrete application. There are some predefined groups to address all name servers (the *all name servers multicast group*), for example, or all routers (the *all routers multicast group*).

38.2.2 Address Types and Structure

As mentioned, the current IP protocol is lacking in two important aspects: there is an increasing shortage of IP addresses and configuring the network and maintaining the routing tables is becoming a more complex and burdensome task. IPv6 solves the first problem by expanding the address space to 128 bits. The second one is countered by introducing a hierarchical address structure, combined with sophisticated techniques to allocate network addresses, as well as *multihoming* (the ability to assign several addresses to one device, giving access to several networks).

When dealing with IPv6, it is useful to know about three different types of addresses:

Unicast
: Addresses of this type are associated with exactly one network interface. Packets with such an address are delivered to only one destination. Accordingly, unicast addresses are used to transfer packets to individual hosts on the local network or the Internet.

Multicast
: Addresses of this type relate to a group of network interfaces. Packets with such an address are delivered to all destinations that belong to the group. Multicast addresses are mainly used by certain network services to communicate with certain groups of hosts in a well-directed manner.

Anycast
: Addresses of this type are related to a group of interfaces. Packets with such an address are delivered to the member of the group that is closest to the sender, according to the principles of the underlying routing protocol. Anycast addresses are used to make it easier for hosts to find out about servers offering certain services in the given network area. All servers of the same type have the same anycast address. Whenever a host requests a service, it receives a reply from the server with the closest location, as determined by the routing protocol. If this server should fail for some reason, the protocol automatically selects the second closest server, then the third one, and so forth.

An IPv6 address is made up of eight four-digit fields, each representing 16 bits, written in hexadecimal notation. They are also separated by colons (`:`). Any leading zero bytes within a given field may be dropped, but zeros within the field or at its end may not. Another convention is that more than four consecutive zero bytes may be collapsed into a double colon. However, only one such `::` is allowed per address. This kind of shorthand notation is shown in Example 38.3, “Sample IPv6 Address” (page 557), where all three lines represent the same address.

Example 38.3 *Sample IPv6 Address*

```
fe80 : 0000 : 0000 : 0000 : 0000 : 10 : 1000 : 1a4
fe80 :    0 :    0 :    0 :    0 : 10 : 1000 : 1a4
fe80 :                           : 10 : 1000 : 1a4
```

Each part of an IPv6 address has a defined function. The first bytes form the prefix and specify the type of address. The center part is the network portion of the address, but it may be unused. The end of the address forms the host part. With IPv6, the netmask is defined by indicating the length of the prefix after a slash at the end of the address. An address, as shown in Example 38.4, "IPv6 Address Specifying the Prefix Length" (page 557), contains the information that the first 64 bits form the network part of the address and the last 64 form its host part. In other words, the `64` means that the netmask is filled with 64 1-bit values from the left. Just like with IPv4, the IP address is combined with AND with the values from the netmask to determine whether the host is located in the same subnetwork or in another one.

Example 38.4 *IPv6 Address Specifying the Prefix Length*

```
fe80::10:1000:1a4/64
```

IPv6 knows about several predefined types of prefixes. Some of these are shown in Table 38.4, "Various IPv6 Prefixes" (page 557).

Table 38.4 *Various IPv6 Prefixes*

Prefix (hex)	Definition
`00`	IPv4 addresses and IPv4 over IPv6 compatibility addresses. These are used to maintain compatibility with IPv4. Their use still requires a router able to translate IPv6 packets into IPv4 packets. Several special addresses, such as the one for the loopback device, have this prefix as well.
`2` or `3` as the first digit	Aggregatable global unicast addresses. As is the case with IPv4, an interface can be assigned to form part of a certain subnetwork. Currently, there are the following address spaces: `2001::/16` (production quality address space) and `2002::/16` (6to4 address space).

Prefix (hex)	Definition
`fe80::/10`	Link-local addresses. Addresses with this prefix should not be routed and should therefore only be reachable from within the same subnetwork.
`fec0::/10`	Site-local addresses. These may be routed, but only within the network of the organization to which they belong. In effect, they are the IPv6 equivalent of the current private network address space, such as `10.x.x.x`.
`ff`	These are multicast addresses.

A unicast address consists of three basic components:

Public Topology
The first part (which also contains one of the prefixes mentioned above) is used to route packets through the public Internet. It includes information about the company or institution that provides the Internet access.

Site Topology
The second part contains routing information about the subnetwork to which to deliver the packet.

Interface ID
The third part identifies the interface to which to deliver the packet. This also allows for the MAC to form part of the address. Given that the MAC is a globally unique, fixed identifier coded into the device by the hardware maker, the configuration procedure is substantially simplified. In fact, the first 64 address bits are consolidated to form the `EUI-64` token, with the last 48 bits taken from the MAC, and the remaining 24 bits containing special information about the token type. This also makes it possible to assign an `EUI-64` token to interfaces that do not have a MAC, such as those based on PPP or ISDN.

On top of this basic structure, IPv6 distinguishes between five different types of unicast addresses:

`::` (unspecified)
This address is used by the host as its source address when the interface is initialized for the first time—when the address cannot yet be determined by other means.

`::1` (loopback)

The address of the loopback device.

IPv4 Compatible Addresses

The IPv6 address is formed by the IPv4 address and a prefix consisting of 96 zero bits. This type of compatibility address is used for tunneling (see Section 38.2.3, “Coexistence of IPv4 and IPv6” (page 560)) to allow IPv4 and IPv6 hosts to communicate with others operating in a pure IPv4 environment.

IPv4 Addresses Mapped to IPv6

This type of address specifies a pure IPv4 address in IPv6 notation.

Local Addresses

There are two address types for local use:

link-local

This type of address can only be used in the local subnetwork. Packets with a source or target address of this type should not be routed to the Internet or other subnetworks. These addresses contain a special prefix (`fe80::/10`) and the interface ID of the network card, with the middle part consisting of zero bytes. Addresses of this type are used during automatic configuration to communicate with other hosts belonging to the same subnetwork.

site-local

Packets with this type of address may be routed to other subnetworks, but not to the wider Internet—they must remain inside the organization's own network. Such addresses are used for intranets and are an equivalent of the private address space defined by IPv4. They contain a special prefix (`fec0::/10`), the interface ID, and a 16 bit field specifying the subnetwork ID. Again, the rest is filled with zero bytes.

As a completely new feature introduced with IPv6, each network interface normally gets several IP addresses, with the advantage that several networks can be accessed through the same interface. One of these networks can be configured completely automatically using the MAC and a known prefix with the result that all hosts on the local network can be reached as soon as IPv6 is enabled (using the link-local address). With the MAC forming part of it, any IP address used in the world is unique. The only variable parts of the address are those specifying the *site topology* and the *public topology*, depending on the actual network in which the host is currently operating.

For a host to go back and forth between different networks, it needs at least two addresses. One of them, the *home address*, not only contains the interface ID but also an identifier of the home network to which it normally belongs (and the corresponding prefix). The home address is a static address and, as such, it does not normally change. Still, all packets destined to the mobile host can be delivered to it, regardless of whether it operates in the home network or somewhere outside. This is made possible by the completely new features introduced with IPv6, such as *stateless autoconfiguration* and *neighbor discovery*. In addition to its home address, a mobile host gets one or more additional addresses that belong to the foreign networks where it is roaming. These are called *care-of* addresses. The home network has a facility that forwards any packets destined to the host when it is roaming outside. In an IPv6 environment, this task is performed by the *home agent*, which takes all packets destined to the home address and relays them through a tunnel. On the other hand, those packets destined to the care-of address are directly transferred to the mobile host without any special detours.

38.2.3 Coexistence of IPv4 and IPv6

The migration of all hosts connected to the Internet from IPv4 to IPv6 is a gradual process. Both protocols will coexist for some time to come. The coexistence on one system is guaranteed where there is a *dual stack* implementation of both protocols. That still leaves the question of how an IPv6 enabled host should communicate with an IPv4 host and how IPv6 packets should be transported by the current networks, which are predominantly IPv4 based. The best solutions offer tunneling and compatibility addresses (see Section 38.2.2, "Address Types and Structure" (page 555)).

IPv6 hosts that are more or less isolated in the (worldwide) IPv4 network can communicate through tunnels: IPv6 packets are encapsulated as IPv4 packets to move them across an IPv4 network. Such a connection between two IPv4 hosts is called a *tunnel*. To achieve this, packets must include the IPv6 destination address (or the corresponding prefix) as well as the IPv4 address of the remote host at the receiving end of the tunnel. A basic tunnel can be configured manually according to an agreement between the hosts' administrators. This is also called *static tunneling*.

However, the configuration and maintenance of static tunnels is often too labor-intensive to use them for daily communication needs. Therefore, IPv6 provides for three different methods of *dynamic tunneling*:

6over4

IPv6 packets are automatically encapsulated as IPv4 packets and sent over an IPv4 network capable of multicasting. IPv6 is tricked into seeing the whole network (Internet) as a huge local area network (LAN). This makes it possible to determine the receiving end of the IPv4 tunnel automatically. However, this method does not scale very well and is also hampered by the fact that IP multicasting is far from widespread on the Internet. Therefore, it only provides a solution for smaller corporate or institutional networks where multicasting can be enabled. The specifications for this method are laid down in RFC 2529.

6to4

With this method, IPv4 addresses are automatically generated from IPv6 addresses, enabling isolated IPv6 hosts to communicate over an IPv4 network. However, a number of problems have been reported regarding the communication between those isolated IPv6 hosts and the Internet. The method is described in RFC 3056.

IPv6 Tunnel Broker

This method relies on special servers that provide dedicated tunnels for IPv6 hosts. It is described in RFC 3053.

IMPORTANT: The 6bone Initiative

In the heart of the “old-time” Internet, there is already a globally distributed network of IPv6 subnets that are connected through tunnels. This is the *6bone* network (`http://www.6bone.net`), an IPv6 test environment that may be used by programmers and Internet providers who want to develop and offer IPv6-based services to gain the experience necessary to implement the new protocol. More information can be found on the project's Internet site.

38.2.4 Configuring IPv6

To configure IPv6, you do not normally need to make any changes on the individual workstations. However, IPv6 support must be loaded. To do this, enter `modprobe ipv6` as `root`.

Because of the autoconfiguration concept of IPv6, the network card is assigned an address in the *link-local* network. Normally, no routing table management takes place on a workstation. The network routers can be queried by the workstation, using the *router advertisement protocol*, for what prefix and gateways should be implemented. The radvd program can be used to set up an IPv6 router. This program informs the workstations which prefix to use for the IPv6 addresses and which routers. Alternatively, use zebra for automatic configuration of both addresses and routing.

Consult the ifup(8) man page to get information about how to set up various types of tunnels using the `/etc/sysconfig/network` files.

38.2.5 For More Information

The above overview does not cover the topic of IPv6 comprehensively. For a more in-depth look at the new protocol, refer to the following online documentation and books:

http://www.ngnet.it/e/cosa-ipv6.php
An article series providing a well-written introduction to the basics of IPv6. A good primer on the topic.

http://www.bieringer.de/linux/IPv6/
Here, find the Linux IPv6-HOWTO and many links related to the topic.

http://www.6bone.net/
Visit this site if you want to join a tunneled IPv6 network.

http://www.ipv6.org/
The starting point for everything about IPv6.

RFC 2640
The fundamental RFC about IPv6.

IPv6 Essentials
A book describing all the important aspects of the topic is *IPv6 Essentials* by Silvia Hagen (ISBN 0-596-00125-8).

38.3 Name Resolution

DNS assists in assigning an IP address to one or more names and assigning a name to an IP address. In Linux, this conversion is usually carried out by a special type of software known as bind. The machine that takes care of this conversion is called a *name server*. The names make up a hierarchical system in which each name component is separated by dots. The name hierarchy is, however, independent of the IP address hierarchy described above.

Consider a complete name, such as `earth.example.com`, written in the format `hostname.domain`. A full name, referred to as a *fully qualified domain name* (FQDN), consists of a hostname and a domain name (`example.com`). The latter also includes the *top level domain* or TLD (`com`).

TLD assignment has become quite confusing for historical reasons. Traditionally, three-letter domain names are used in the USA. In the rest of the world, the two-letter ISO national codes are the standard. In addition to that, longer TLDs were introduced in 2000 that represent certain spheres of activity (for example, `.info`, `.name`, `.museum`).

In the early days of the Internet (before 1990), the file `/etc/hosts` was used to store the names of all the machines represented over the Internet. This quickly proved to be impractical in the face of the rapidly growing number of computers connected to the Internet. For this reason, a decentralized database was developed to store the hostnames in a widely distributed manner. This database, similar to the name server, does not have the data pertaining to all hosts in the Internet readily available, but can dispatch requests to other name servers.

The top of the hierarchy is occupied by *root name servers*. These root name servers manage the top level domains and are run by the Network Information Center (NIC). Each root name server knows about the name servers responsible for a given top level domain. Information about top level domain NICs is available at `http://www.internic.net`.

DNS can do more than just resolve hostnames. The name server also knows which host is receiving e-mails for an entire domain—the *mail exchanger (MX)*.

For your machine to resolve an IP address, it must know about at least one name server and its IP address. Easily specify such a name server with the help of YaST. If you have a modem dial-up connection, you may not need to configure a name server manually at all. The dial-up protocol provides the name server address as the connection is made. The configuration of name server access with SUSE Linux is described in Chapter 40, *The Domain Name System* (page 593).

The protocol `whois` is closely related to DNS. With this program, quickly find out who is responsible for any given domain.

38.4 Configuring a Network Connection with YaST

There are many supported networking types on Linux. Most of them use different device names and the configuration files are spread over several locations in the file system. For a detailed overview of the aspects of manual network configuration, see Section 38.5, “Configuring a Network Connection Manually” (page 574).

During installation, YaST can be used to configure automatically all interfaces that have been detected. Additional hardware can be configured any time after installation in the installed system. The following sections describe the network configuration for all types of network connections supported by SUSE Linux.

38.4.1 Configuring the Network Card with YaST

After starting the module, YaST displays a general network configuration dialog. The upper part shows a list with all the network cards yet to be configured. Any card properly detected is listed with its name. Devices that could not be detected may be configured using *Add* as described in Section “Manual Configuration of an Undetected Network Card” (page 564). Configure a new network card or change an existing configuration.

Manual Configuration of an Undetected Network Card

Configuring a network card that was not detected includes the following items:

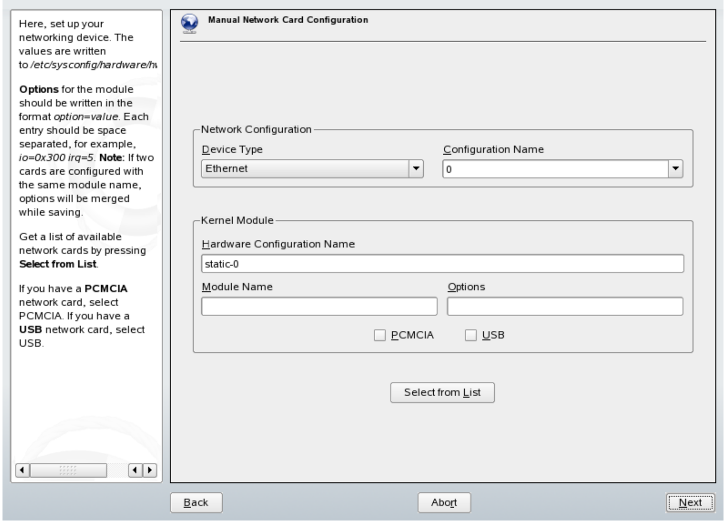
Network Configuration
: Set the device type of the interface from the available options and the configuration name. Information about the naming conventions for configuration names is available in the `getcfg(8)` man page.

Kernel Module
: *Hardware Configuration Name* specifies the name of the `/etc/sysconfig/hardware/hwcfg-*` file containing the hardware settings of your network card. This contains the name of the suitable kernel module as well as the needed options

to initialize the hardware. Usually, YaST proposes useful names for PCMCIA and USB hardware. For other hardware, `hwcfg-static-0` usually only makes sense if the card is configured with the configuration name `0`.

If the network card is a PCMCIA or USB device, activate the respective check boxes and exit this dialog with *Next*. Otherwise, select your network card model from *Select from List*. YaST then automatically selects the suitable kernel module for the card. Exit this dialog with *Next*.

Figure 38.3 *Configuration of the Network Card*



Here, set up your networking device. The values are written to */etc/sysconfig/hardware/hw*

Options for the module should be written in the format *option=value*. Each entry should be space separated, for example, *io=0x300 irq=5*. **Note**: If two cards are configured with the same module name, options will be merged while saving.

Get a list of available network cards by pressing **Select from List**.

If you have a **PCMCIA** network card, select PCMCIA. If you have a **USB** network card, select USB.

Manual Network Card Configuration

Network Configuration

Device Type: Ethernet

Configuration Name: 0

Kernel Module

Hardware Configuration Name: static-0

Module Name:

Options:

□ PCMCIA □ USB

Setting the Network Address

Set the device type of the interface and the configuration name. Select the device type from those provided. Specify a configuration name according to your needs. Usually, the default settings are useful and can be accepted. Information about the naming conventions for configuration names is available in the `getcfg(8)` man page .

If you selected *Wireless* as the device type of the interface, configure the operating mode, the network name (ESSID), and the encryption in the next dialog, *Wireless Network Card Configuration*. Click *OK* to complete the configuration of your card. A detailed description of the configuration of WLAN cards is provided in Section 22.1.3, "Configuration with YaST" (page 288). For all other interface types, proceed with the network address setup:

Automatic Address Setup (via DHCP)

If your network includes a DHCP server, you can rely on it to set up your network address automatically. The option should also be used if you are using a DSL line but with no static IP assigned by the ISP. If you decide to use DHCP, configure the details after selecting *DHCP Client Options*. Specify whether the DHCP server should always honor broadcast requests and any identifier to use. By default, DHCP servers use the card's hardware address to identify an interface. If you have a virtual host setup where different hosts communicate through the same interface, an identifier is necessary to distinguish them.

Static Address Setup

If you have a static address, enable that option. Then enter the address and subnet mask for your network. The preset subnet mask should match the requirements of a typical home network.

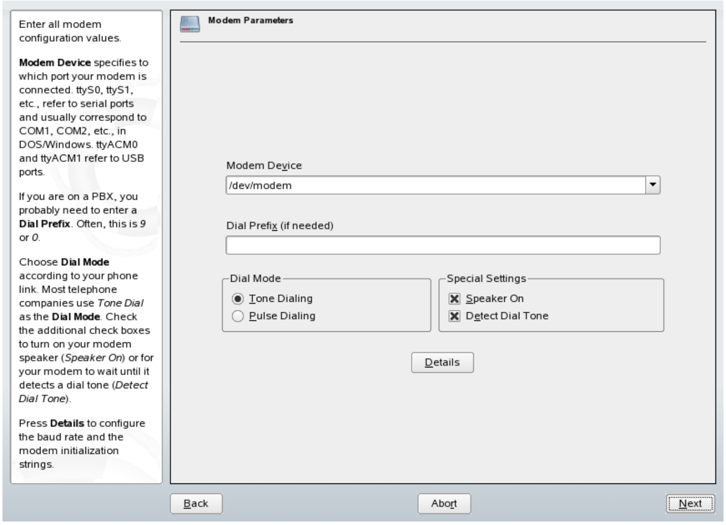
Leave this dialog by selecting *Next* or proceed to configure the hostname, name server, and routing details (see the sections on DNS Server (↑Start-Up) and Routing (↑Start-Up)).

Advanced enables you to specify more complex settings. Under *Detailed Settings*, use *User Controlled* to delegate the control over the network card from the administrator (`root`) to the normal user. For mobile operation, this allows the user to adapt changing network connections in a more flexible way, because he can control the activation or deactivation of the interface. The MTU (maximum transmission unit) and the type of *Device Activation* can also be set in this dialog.

38.4.2 Modem

In the YaST Control Center, access the modem configuration under *Network Devices*. If your modem was not automatically detected, open the dialog for manual configuration. In the dialog that opens, enter the interface to which the modem is connected under *Modem*.

Figure 38.4 Modem Configuration



Enter all modem configuration values.

Modem Device specifies to which port your modem is connected. ttyS0, ttyS1, etc., refer to serial ports and usually correspond to COM1, COM2, etc., in DOS/Windows. ttyACM0 and ttyACM1 refer to USB ports.

If you are on a PBX, you probably need to enter a **Dial Prefix**. Often, this is *9* or *0*.

Choose **Dial Mode** according to your phone link. Most telephone companies use *Tone Dial* as the **Dial Mode**. Check the additional check boxes to turn on your modem speaker (*Speaker On*) or for your modem to wait until it detects a dial tone (*Detect Dial Tone*).

Press **Details** to configure the baud rate and the modem initialization strings.

Modem Parameters

Modem Device: /dev/modem

Dial Prefix (if needed):

Dial Mode
- ● Tone Dialing
- ○ Pulse Dialing

Special Settings
- ☒ Speaker On
- ☒ Detect Dial Tone

If you are behind a private branch exchange (PBX), you may need to enter a dial prefix. This is often a zero. Consult the instructions that came with the PBX to find out. Also select whether to use tone or pulse dialing, whether the speaker should be on, and whether the modem should wait until it detects a dial tone. The last option should not be enabled if the modem is connected to an exchange.

Under *Details*, set the baud rate and the modem initialization strings. Only change these settings if your modem was not autodetected or if it requires special settings for data transmission to work. This is mainly the case with ISDN terminal adapters. Leave this dialog by clicking *OK*. To delegate control over the modem to the normal user without root permissions, activate *User Controlled*. In this way, a user without administrator permissions can activate or deactivate an interface. Under *Dial Prefix Regular Expression*, specify a regular expression. The *Dial Prefix* in KInternet, which can be modified by the normal user, must match this regular expression. If this field is left empty, the user cannot set a different *Dial Prefix* without administrator permissions.

In the next dialog, select the ISP (Internet service provider). To choose from a predefined list of ISPs operating in your country, select *Country*. Alternatively, click *New* to open a dialog in which to provide the data for your ISP. This includes a name for the dial-up connection and ISP as well as the login and password provided by your ISP. Enable *Always Ask for Password* to be prompted for the password each time you connect.

In the last dialog, specify additional connection options:

Dial on Demand
If you enable dial on demand, set at least one name server.

Modify DNS when Connected
This option is enabled by default, with the effect that the name server address is updated each time you connect to the Internet.

Automatically Retrieve DNS
If the provider does not transmit its domain name server after connecting, disable this option and enter the DNS data manually.

Stupid Mode
This option is enabled by default. With it, input prompts sent by the ISP's server are ignored to prevent them from interfering with the connection process.

External Firewall Interface* and *Restart Firewall
Selecting these options enables the SUSEfirewall2, which protects you from outside attacks for the duration of your Internet connection.

Idle Time-Out (seconds)
With this option, specify a period of network inactivity after which the modem disconnects automatically.

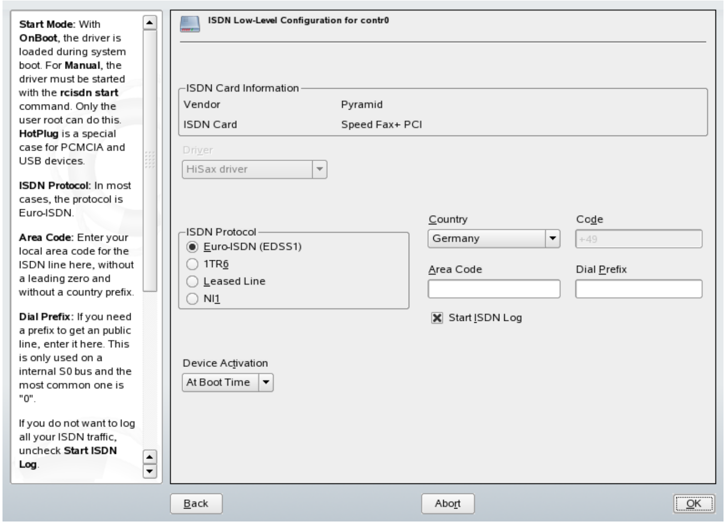
IP Details
This opens the address configuration dialog. If your ISP does not assign a dynamic IP address to your host, disable *Dynamic IP Address* then enter your host's local IP address and the remote IP address. Ask your ISP for this information. Leave *Default Route* enabled and close the dialog by selecting *OK*.

Selecting *Next* returns to the original dialog, which displays a summary of the modem configuration. Close this dialog with *Finish*.

38.4.3 ISDN

Use this module to configure one or several ISDN cards for your system. If YaST did not detect your ISDN card, manually select it. Multiple interfaces are possible, but several ISPs can be configured for one interface. In the subsequent dialogs, set the ISDN options necessary for the proper functioning of the card.

Figure 38.5 *ISDN Configuration*



Start Mode: With **OnBoot**, the driver is loaded during system boot. For **Manual**, the driver must be started with the **rcisdn start** command. Only the user root can do this. **HotPlug** is a special case for PCMCIA and USB devices.

ISDN Protocol: In most cases, the protocol is Euro-ISDN.

Area Code: Enter your local area code for the ISDN line here, without a leading zero and without a country prefix.

Dial Prefix: If you need a prefix to get an public line, enter it here. This is only used on a internal S0 bus and the most common one is "0".

If you do not want to log all your ISDN traffic, uncheck **Start ISDN Log**.

ISDN Low-Level Configuration for contr0

ISDN Card Information

Vendor	Pyramid
ISDN Card	Speed Fax+ PCI

Driver: HiSax driver

ISDN Protocol
- ● Euro-ISDN (EDSS1)
- ○ 1TR6
- ○ Leased Line
- ○ NI1

Country: Germany

Code: +49

Area Code:

Dial Prefix:

☒ Start ISDN Log

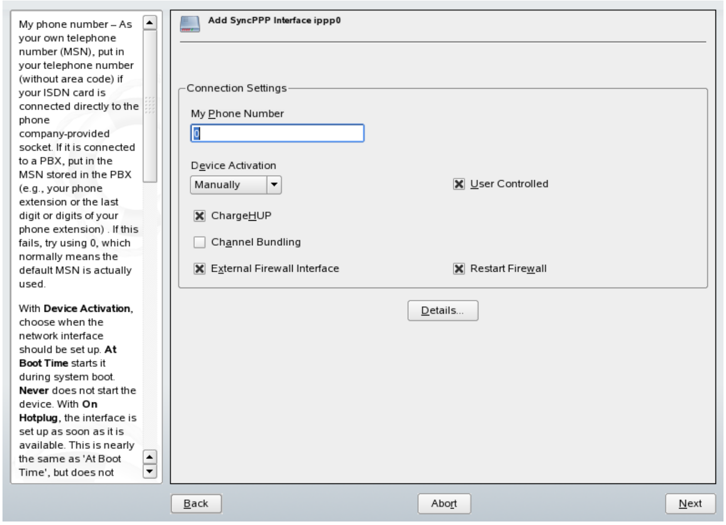
Device Activation: At Boot Time

In the next dialog, shown in Figure 38.5, "ISDN Configuration" (page 569), select the protocol to use. The default is *Euro-ISDN (EDSS1)*, but for older or larger exchanges, select *1TR6*. If you are in the US, select *NI1*. Select your country in the relevant field. The corresponding country code then appears in the field next to it. Finally, provide your *Area Code* and the *Dial Prefix* if necessary.

Start Mode defines how the ISDN interface should be started: *At Boot Time* causes the ISDN driver to be initialized each time the system boots. *Manually* requires you to load the ISDN driver as `root` with the command `rcisdn start`. *On Hotplug*, used for PCMCIA or USB devices, loads the driver after the device is plugged in. When finished with these settings, select *OK*.

In the next dialog, specify the interface type for your ISDN card and add ISPs to an existing interface. Interfaces may be either the `SyncPPP` or the `RawIP` type, but most ISPs operate in the `SyncPPP` mode, which is described below.

Figure 38.6 ISDN Interface Configuration



My phone number – As your own telephone number (MSN), put in your telephone number (without area code) if your ISDN card is connected directly to the phone company-provided socket. If it is connected to a PBX, put in the MSN stored in the PBX (e.g., your phone extension or the last digit or digits of your phone extension) . If this fails, try using 0, which normally means the default MSN is actually used.

With **Device Activation**, choose when the network interface should be set up. **At Boot Time** starts it during system boot. **Never** does not start the device. With **On Hotplug**, the interface is set up as soon as it is available. This is nearly the same as 'At Boot Time', but does not

Add SyncPPP Interface ippp0

Connection Settings

My Phone Number: 0

Device Activation: Manually

☒ User Controlled

☒ ChargeHUP

☐ Channel Bundling

☒ External Firewall Interface

☒ Restart Firewall

The number to enter for *My Phone Number* depends on your particular setup:

ISDN Card Directly Connected to Phone Outlet
A standard ISDN line provides three phone numbers (called multiple subscriber numbers, or MSNs). If the subscriber asked for more, there may be up to 10. One of these MSNs must be entered here, but without your area code. If you enter the wrong number, your phone operator automatically falls back to the first MSN assigned to your ISDN line.

ISDN Card Connected to a Private Branch Exchange
Again, the configuration may vary depending on the equipment installed:

1. Smaller private branch exchanges (PBX) built for home purposes mostly use the Euro-ISDN (EDSS1) protocol for internal calls. These exchanges have an internal S0 bus and use internal numbers for the equipment connected to them.

   Use one of the internal numbers as your MSN. You should be able to use at least one of the exchange's MSNs that have been enabled for direct outward dialing. If this does not work, try a single zero. For further information, consult the documentation that came with your phone exchange.

2. Larger phone exchanges designed for businesses normally use the 1TR6 protocol for internal calls. Their MSN is called EAZ and usually corresponds to the direct-dial number. For the configuration under Linux, it should be sufficient to enter the last digit of the EAZ. As a last resort, try each of the digits from 1 to 9.

For the connection to be terminated just before the next charge unit is due, enable *ChargeHUP*. However, remember that may not work with every ISP. You can also enable channel bundling (multilink PPP) by selecting the corresponding option. Finally, you can enable SuSEfirewall2 for your link by selecting *External Firewall Interface* and *Restart Firewall*. To enable the normal user without administrator permissions to activate or deactivate the interface, select the *User Controlled*.

Details opens a dialog in which to implement more complex connection schemes, which are not relevant for normal home users. Leave the *Details* dialog by selecting *OK*.

In the next dialog, make IP address settings. If you have not been given a static IP by your provider, select *Dynamic IP Address*. Otherwise, use the fields provided to enter your host's local IP address and the remote IP address according to the specifications of your ISP. If the interface should be the default route to the Internet, select *Default Route*. Each host can only have one interface configured as the default route. Leave this dialog by selecting *Next*.

The following dialog allows you to set your country and select an ISP. The ISPs included in the list are call-by-call providers only. If your ISP is not in the list, select *New*. This opens the *Provider Parameters* dialog in which to enter all the details for your ISP. When entering the phone number, do not include any blanks or commas among the digits. Finally, enter your login and the password as provided by the ISP. When finished, select *Next*.

To use *Dial on Demand* on a stand-alone workstation, also specify the name server (DNS server). Most ISPs support dynamic DNS, which means the IP address of a name server is sent by the ISP each time you connect. For a single workstation, however, you still need to provide a placeholder address like `192.168.22.99`. If your ISP does not support dynamic DNS, specify the name server IP addresses of the ISP. If desired, specify a time-out for the connection—the period of network inactivity (in seconds) after which the connection should be automatically terminated. Confirm your settings with *Next*. YaST displays a summary of the configured interfaces. To make all these settings active, select *Finish*.

38.4.4 Cable Modem

In some countries, such as Austria and the US, it is quite common to access the Internet through the TV cable network. The TV cable subscriber usually gets a modem that is connected to the TV cable outlet on one side and to a computer network card on the other (using a 10Base-TG twisted pair cable). The cable modem then provides a dedicated Internet connection with a fixed IP address.

Depending on the instructions provided by your ISP, when configuring the network card either select *Automatic Address Setup (via DHCP)* or *Static Address Setup*. Most providers today use DHCP. A static IP address often comes as part of a special business account.

38.4.5 DSL

To configure your DSL device, select the *DSL* module from the YaST *Network Devices* section. This YaST module consists of several dialogs in which to set the parameters of DSL links based on one of the following protocols:

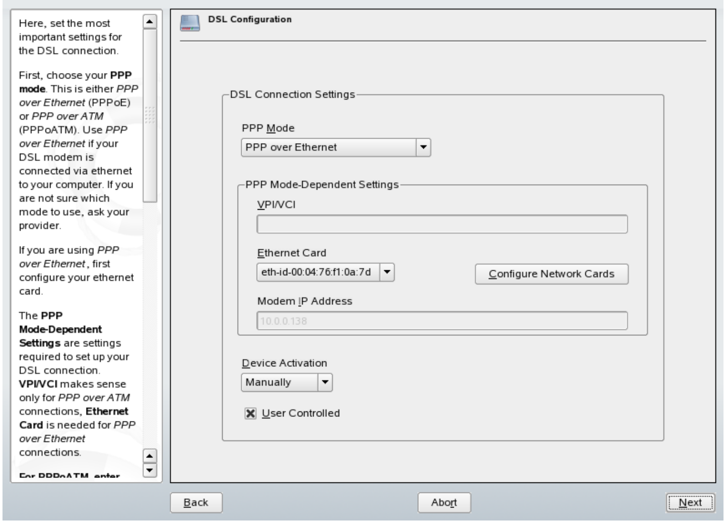
- PPP over Ethernet (PPPoE)
- PPP over ATM (PPPoATM)
- CAPI for ADSL (Fritz Cards)
- Point-to-Point Tunneling Protocol (PPTP)—Austria

The configuration of a DSL connection based on PPPoE or PPTP requires that the corresponding network card has already been set up in the correct way. If you have not done so yet, first configure the card by selecting *Configure Network Cards* (see Section 38.4.1, “Configuring the Network Card with YaST” (page 564)). In the case of a DSL link, addresses may be assigned automatically but not via DHCP, which is why you should not enable the option *Automatic address setup (via DHCP)*. Instead, enter a static dummy address for the interface, such as `192.168.22.1`. In *Subnet Mask*, enter `255.255.255.0`. If you are configuring a stand-alone workstation, leave *Default Gateway* empty.

TIP

Values in *IP Address* and *Subnet Mask* are only placeholders. They are only needed to initialize the network card and do not represent the DSL link as such.

Figure 38.7 *DSL Configuration*



Here, set the most important settings for the DSL connection.

First, choose your **PPP mode**. This is either *PPP over Ethernet* (PPPoE) or *PPP over ATM* (PPPoATM). Use *PPP over Ethernet* if your DSL modem is connected via ethernet to your computer. If you are not sure which mode to use, ask your provider.

If you are using *PPP over Ethernet*, first configure your ethernet card.

The **PPP Mode-Dependent Settings** are settings required to set up your DSL connection. **VPI/VCI** makes sense only for *PPP over ATM* connections, **Ethernet Card** is needed for *PPP over Ethernet* connections.

For PPPoATM, enter

DSL Configuration

DSL Connection Settings

PPP Mode: PPP over Ethernet

PPP Mode-Dependent Settings

VPI/VCI:

Ethernet Card: eth-id-00:04:76:f1:0a:7d

Modem IP Address: 10.0.0.138

Device Activation: Manually

☒ User Controlled

To begin the DSL configuration (see Figure 38.7, “DSL Configuration” (page 573)), first select the PPP mode and the ethernet card to which the DSL modem is connected (in most cases, this is `eth0`). Then use *Device Activation* to specify whether the DSL link should be established during the boot process. Click *User Controlled* to authorize the normal user without root permissions to activate or deactivate the interface with KInternet. The dialog also lets you select your country and choose from a number of ISPs operating in it. The details of any subsequent dialogs of the DSL configuration depend on the options set so far, which is why they are only briefly mentioned in the following paragraphs. For details on the available options, read the detailed help available from the dialogs.

To use *Dial on Demand* on a stand-alone workstation, also specify the name server (DNS server). Most ISPs support dynamic DNS—the IP address of a name server is sent by the ISP each time you connect. For a single workstation, however, provide a placeholder address like `192.168.22.99`. If your ISP does not support dynamic DNS, enter the name server IP address provided by your ISP.

Idle Time-Out (seconds) defines a period of network inactivity after which to terminate the connection automatically. A reasonable time-out value is between 60 and 300 seconds. If *Dial on Demand* is disabled, it may be useful to set the time-out to zero to prevent automatic hang-up.

The configuration of T-DSL is very similar to the DSL setup. Just select *T-Online* as your provider and YaST opens the T-DSL configuration dialog. In this dialog, provide some additional information required for T-DSL—the line ID, the T-Online number, the user code, and your password. All of these should be included in the information you received after subscribing to T-DSL.

38.5 Configuring a Network Connection Manually

Manual configuration of the network software should always be the last alternative. Using YaST is recommended. However, this background information about the network configuration can also assist your work with YaST.

All built-in network cards and hotplug network cards (PCMCIA, USB, some PCI cards) are detected and configured via hotplug. The system sees a network card in two different ways: first as a physical device and second as an interface. The insertion or detection of a device triggers a hotplug event. This hotplug event triggers the initialization of the device with the script `hwup`. When the network card is initialized as a new network interface, the kernel generates another hotplug event that triggers the setup of the interface with `ifup`.

The kernel numbers interface names according to the temporal order of their registration. The initialization sequence is decisive for the assignment of names. If one of several network card fails, the numbering of all subsequently initialized cards is shifted. For real hotpluggable cards, the order in which the devices are connected is what matters.

To achieve a flexible configuration, the configuration of the device (hardware) and the interface has been separated and the mapping of configurations to devices and interfaces is no longer managed on the basis of the interface names. The device configurations are located in `/etc/sysconfig/hardware/hwcfg-*`. The interface configurations are located in `/etc/sysconfig/network/ifcfg-*`. The names of the configurations are assigned in such a way that they describe the devices and interfaces with which they are associated. Because the former mapping of drivers to interface name required static interface names, this mapping can no longer take place in /etc/modprobe.conf. In the new concept, alias entries in this file would cause undesirable side effects.

The configuration names—everything after hwcfg- or ifcfg-—can describe the devices by means of the slot, a device-specific ID, or the interface name. For example, the configuration name for a PCI card could be bus-pci-0000:02:01.0 (PCI slot) or vpid-0x8086-0x1014-0x0549 (vendor and product ID). The name of the associated interface could be bus-pci-0000:02:01.0 or wlan-id-00:05:4e:42:31:7a (MAC address).

To assign a certain network configuration to any card of a certain type (of which only one is inserted at a time) instead of a certain card, select less specific configuration names. For example, bus-pcmcia would be used for all PCMCIA cards. On the other hand, the names can be limited by a preceding interface type. For example, wlan-bus-usb would be assigned to WLAN cards connected to a USB port.

The system always uses the configuration that best describes an interface or the device providing the interface. The search for the most suitable configuration is handled by getcfg. The output of getcfg delivers all information that can be used for describing a device. Details regarding the specification of configuration names are available in the manual page of getcfg.

With the described method, a network interface is configured with the correct configuration even if the network devices are not always initialized in the same order. However, the name of the interface still depends on the initialization sequence. There are two ways to ensure reliable access to the interface of a certain network card:

- getcfg-interface *configuration name* returns the name of the associated network interface. Therefore, the configuration name, such as firewall, dhcpd, routing, or various virtual network interfaces (tunnels), can be entered in some configuration files instead of the interface name, which is not persistent.

- Persistent interface names are assigned to each interface automatically. You may adjust them to suit your needs. When creating interface names, proceed as outlined in /etc/udev/rules.d/30-net_persistent_names.rules. However, the persistent name *pname* should not be the same as the name that would automatically be assigned by the kernel. Therefore, eth*, tr*, wlan*, and so on are not permitted. Instead, use net* or descriptive names like external, internal, or dmz. Make sure that the same interface name is not used twice. Allowed characters in interface names are restricted to [a-zA-Z0-9]. A persistent name can only be assigned to an interface immediately after its registration, which means that the driver of the network card must be reloaded or hwup *device description* must be executed. The command rcnetwork restart is not sufficient for this purpose.

IMPORTANT: Using Persistent Interface Names

The use of persistent interface names has not been tested in all areas. Therefore, some applications may not be able to handle freely selected interface names.

ifup requires an existing interface, because it does not initialize the hardware. The initialization of the hardware is handled by the command hwup (executed by hotplug or coldplug). When a device is initialized, ifup is automatically executed for the new interface via hotplug and the interface is set up if the start mode is onboot, hotplug, or auto and the network service was started. Formerly, the command ifup *interfacename* triggered the hardware initialization. Now the procedure has been reversed. First, a hardware component is initialized then all other actions follow. In this way, a varying number of devices can always be configured in the best way possible with an existing set of configurations.

Table 38.5, "Manual Network Configuration Scripts" (page 576) summarizes the most important scripts involved in the network configuration. Where possible, the scripts are distinguished by hardware and interface.

Table 38.5 *Manual Network Configuration Scripts*

Configuration Stage	Command	Function
Hardware	hw{up,down,status}	The hw* scripts are executed by the hotplug subsystem to initialize a device, undo the initialization, or query the status of a device. More information is available in the manual page of hwup.
Interface	getcfg	getcfg can be used to query the interface name associated with a configuration

Configuration Stage	Command	Function
		name or a hardware description. More information is available in the manual page of getcfg.
Interface	if{up,down,status}	The if* scripts start existing network interfaces or return the status of the specified interface. More information is available in the manual page of ifup.

More information about hotplug and persistent device names is available in Chapter 32, *The Hotplug System* (page 485) and Chapter 33, *Dynamic Device Nodes with udev* (page 491).

38.5.1 Configuration Files

This section provides an overview of the network configuration files and explains their purpose and the format used.

/etc/syconfig/hardware/hwcfg-*

These files contain the hardware configurations of network cards and other devices. They contain the needed parameters, such as the kernel module, start mode, and script associations. Refer to the manual page of hwup for details. Regardless of the existing hardware, the hwcfg-static-* configurations are applied when coldplug is started.

/etc/sysconfig/network/ifcfg-*

These files contain the configurations for network interface. They include information such as the start mode and the IP address. Possible parameters are described in the manual page of ifup. Additionally, all variables from the files dhcp, wireless, and config can be used in the ifcfg-* files if a general setting should be used for only one interface.

/etc/sysconfig/network/config, dhcp, wireless

The file config contains general settings for the behavior of ifup, ifdown, and ifstatus. dhcp contains settings for DHCP and wireless for wireless LAN cards. The variables in all three configuration files are commented and can also be used in ifcfg-* files, where they are treated with higher priority.

/etc/sysconfig/network/routes,ifroute-*

The static routing of TCP/IP packets is determined here. All the static routes required by the various system tasks can be entered in the /etc/sysconfig/network/routes file: routes to a host, routes to a host via a gateway, and routes to a network. For each interface that needs individual routing, define an additional configuration file: /etc/sysconfig/network/ifroute-*. Replace * with the name of the interface. The entries in the routing configuration files look like this:

```
# Destination     Dummy/Gateway     Netmask            Device
#
127.0.0.0         0.0.0.0           255.255.255.0      lo
204.127.235.0     0.0.0.0           255.255.255.0      eth0
default           204.127.235.41    0.0.0.0            eth0
207.68.156.51     207.68.145.45     255.255.255.255    eth1
192.168.0.0       207.68.156.51     255.255.0.0        eth1
```

The route's destination is in the first column. This column may contain the IP address of a network or host or, in the case of *reachable* name servers, the fully qualified network or hostname.

The second column contains the default gateway or a gateway through which a host or network can be accessed. The third column contains the netmask for networks or hosts behind a gateway. For example, the mask is 255.255.255.255 for a host behind a gateway.

The fourth column is only relevant for networks connected to the local host such as loopback, Ethernet, ISDN, PPP, and dummy device. The device name must be entered here.

An (optional) fifth column can be used to specify the type of a route. Columns that are not needed should contain a minus sign - to ensure that the parser correctly interprets the command. For details, refer to the routes(5) man page.

/etc/resolv.conf

The domain to which the host belongs is specified in this file (keyword search). Also listed is the status of the name server address to access (keyword nameserver). Multiple domain names can be specified. When resolving a name that is not fully qualified, an attempt is made to generate one by attaching the individual search entries. Use multiple name servers by entering several lines, each beginning with nameserver. Precede comments with # signs. YaST enters the specified name server in this file. Example 38.5, "/etc/resolv.conf" (page 579) shows what /etc/resolv.conf could look like.

Example 38.5 */etc/resolv.conf*

```
# Our domain
search example.com
#
# We use sun (192.168.0.20) as nameserver
nameserver 192.168.0.20
```

Some services, like pppd (wvdial), ipppd (isdn), dhcp (dhcpcd and dhclient), pcmcia, and hotplug, modify the file /etc/resolv.conf by means of the script modify_resolvconf. If the file /etc/resolv.conf has been temporarily modified by this script, it contains a predefined comment giving information about the service that modified it, the location where the original file has been backed up, and how to turn off the automatic modification mechanism. If /etc/resolv.conf is modified several times, the file includes modifications in a nested form. These can be reverted in a clean way even if this reversal takes place in an order different from the order in which modifications were introduced. Services that may need this flexibility include isdn, pcmcia, and hotplug.

If a service was not terminated in a normal, clean way, modify_resolvconf can be used to restore the original file. Also, on system boot, a check is performed to see whether there is an uncleaned, modified resolv.conf, for example, after a system crash, in which case the original (unmodified) resolv.conf is restored.

YaST uses the command modify_resolvconf check to find out whether resolv.conf has been modified and subsequently warns the user that changes will be lost after restoring the file. Apart from this, YaST does not rely on modify_resolvconf, which means that the impact of changing resolv.conf through YaST is the same as that of any manual change. In both cases, changes have a permanent effect. Modifications requested by the mentioned services are only temporary.

/etc/hosts

In this file, shown in Example 38.6, “/etc/hosts” (page 580), IP addresses are assigned to hostnames. If no name server is implemented, all hosts to which an IP connection will be set up must be listed here. For each host, enter a line consisting of the IP address, the fully qualified hostname, and the hostname into the file. The IP address must be at the beginning of the line and the entries separated by blanks and tabs. Comments are always preceded by the # sign.

***Example 38.6** /etc/hosts*

```
127.0.0.1 localhost
192.168.0.20 sun.example.com sun
192.168.0.0 earth.example.com earth
```

/etc/networks

Here, network names are converted to network addresses. The format is similar to that of the hosts file, except the network names precede the addresses. See Example 38.7, “/etc/networks” (page 580).

***Example 38.7** /etc/networks*

```
loopback     127.0.0.0
localnet     192.168.0.0
```

/etc/host.conf

Name resolution—the translation of host and network names via the *resolver* library—is controlled by this file. This file is only used for programs linked to libc4 or libc5. For current glibc programs, refer to the settings in /etc/nsswitch.conf. A parameter must always stand alone in its own line. Comments are preceded by a # sign. Table 38.6, “Parameters for /etc/host.conf” (page 580) shows the parameters available. A sample /etc/host.conf is shown in Example 38.8, “ /etc/host.conf ” (page 581).

***Table 38.6** Parameters for /etc/host.conf*

order *hosts*, *bind*	Specifies in which order the services are accessed for the name resolution. Available arguments are (separated by blank spaces or commas):

hosts: Searches the `/etc/hosts` file

bind: Accesses a name server

nis: Uses NIS

multi *on*/*off*

Defines if a host entered in `/etc/hosts` can have multiple IP addresses.

nospoof *on*
spoofalert *on*/*off*

These parameters influence the name server *spoofing*, but, apart from that, do not exert any influence on the network configuration.

trim *domainname*

The specified domain name is separated from the hostname after hostname resolution (as long as the hostname includes the domain name). This option is useful if only names from the local domain are in the `/etc/hosts` file, but should still be recognized with the attached domain names.

Example 38.8 *`/etc/host.conf`*

```
# We have named running
order hosts bind
# Allow multiple addrs
multi on
```

`/etc/nsswitch.conf`

The introduction of the GNU C Library 2.0 was accompanied by the introduction of the *Name Service Switch* (NSS). Refer to the `nsswitch.conf(5)` man page and *The GNU C Library Reference Manual* for details.

The order for queries is defined in the file `/etc/nsswitch.conf`. A sample `nsswitch.conf` is shown in Example 38.9, "`/etc/nsswitch.conf`" (page 582). Comments are introduced by # signs. In this example, the entry under the `hosts` database means that a request is sent to `/etc/hosts` (`files`) via DNS (see Chapter 40, *The Domain Name System* (page 593)).

Example 38.9 */etc/nsswitch.conf*

```
passwd:     compat
group:      compat

hosts:      files dns
networks:   files dns

services:   db files
protocols:  db files

netgroup:   files
automount:  files nis
```

The “databases” available over NSS are listed in Table 38.7, “Databases Available via /etc/nsswitch.conf” (page 582). In addition, automount, bootparams, netmasks, and publickey are expected in the near future. The configuration options for NSS databases are listed in Table 38.8, “Configuration Options for NSS “Databases”” (page 583).

Table 38.7 *Databases Available via /etc/nsswitch.conf*

aliases	Mail aliases implemented by sendmail; see man 5 aliases.
ethers	Ethernet addresses.
group	For user groups, used by getgrent. See also the man page for group.
hosts	For hostnames and IP addresses, used by gethostbyname and similar functions.
netgroup	Valid host and user lists in the network for the purpose of controlling access permissions; see the netgroup(5) man page.
networks	Network names and addresses, used by getnetent.
passwd	User passwords, used by getpwent; see the passwd(5) man page.

`protocols`	Network protocols, used by `getprotoent`; see the `protocols(5)` man page.
`rpc`	Remote procedure call names and addresses, used by `getrpcbyname` and similar functions.
`services`	Network services, used by `getservent`.
`shadow`	Shadow passwords of users, used by `getspnam`; see the `shadow(5)` man page.

Table 38.8 *Configuration Options for NSS "Databases"*

`files`	directly access files, for example, `/etc/aliases`
`db`	access via a database
`nis`, `nisplus`	NIS, see also Chapter 41, *Using NIS* (page 615)
`dns`	can only be used as an extension for `hosts` and `networks`
`compat`	can only be used as an extension for `passwd`, `shadow`, and `group`

/etc/nscd.conf

This file is used to configure nscd (name service cache daemon). See the `nscd(8)` and `nscd.conf(5)` man pages. By default, the system entries of `passwd` and `groups` are cached by nscd. This is important for the performance of directory services, like NIS and LDAP, because otherwise the network connection needs to be used for every access to names or groups. `hosts` is not cached by default, because the mechanism in nscd to cache hosts makes the local system unable to trust forward and reverse lookup checks. Instead of asking nscd to cache names, set up a caching DNS server.

If the caching for `passwd` is activated, it usually takes about fifteen seconds until a newly added local user is recognized. Reduce this waiting time by restarting nscd with the command `rcnscd restart`.

/etc/HOSTNAME

This contains the hostname without the domain name attached. This file is read by several scripts while the machine is booting. It may only contain one line in which the hostname is set.

38.5.2 Start-Up Scripts

Apart from the configuration files described above, there are also various scripts that load the network programs while the machine is booting. These are started as soon as the system is switched to one of the *multiuser runlevels*. Some of these scripts are described in Table 38.9, "Some Start-Up Scripts for Network Programs" (page 584).

Table 38.9 *Some Start-Up Scripts for Network Programs*

/etc/init.d/network	This script handles the configuration of the network interfaces. The hardware must already have been initialized by /etc/init.d/coldplug (via hotplug). If the network service was not started, no network interfaces are implemented when they are inserted via hotplug.
/etc/init.d/inetd	Starts xinetd. xinetd can be used to make server services available on the system. For example, it can start vsftpd whenever an FTP connection is initiated.
/etc/init.d/portmap	Starts the portmapper needed for the RPC server, such as an NFS server.
/etc/init.d/nfsserver	Starts the NFS server.
/etc/init.d/sendmail	Controls the sendmail process.
/etc/init.d/ypserv	Starts the NIS server.
/etc/init.d/ypbind	Starts the NIS client.

38.6 smpppd as Dial-up Assistant

Most home users do not have a dedicated line connecting them to the Internet. Instead, they use dial-up connections. Depending on the dial-up method (ISDN or DSL), the connection is controlled by ipppd or pppd. Basically, all that needs to be done to go online is to start these programs correctly.

If you have a flat-rate connection that does not generate any additional costs for the dial-up connection, simply start the respective daemon. Control the dial-up connection with a KDE applet or a command-line interface. If the Internet gateway is not the host you are using, you might want to control the dial-up connection by way of a network host.

This is where smpppd is involved. It provides a uniform interface for auxiliary programs and acts in two directions. First, it programs the required pppd or ipppd and controls its dial-up properties. Second, it makes various providers available to the user programs and transmits information about the current status of the connection. As smpppd can also be controlled by way of the network, it is suitable for controlling dial-up connections to the Internet from a workstation in a private subnetwork.

38.6.1 Configuring smpppd

The connections provided by smpppd are automatically configured by YaST. The actual dial-up programs KInternet and cinternet are also preconfigured. Manual settings are only required to configure additional features of smpppd, such as remote control.

The configuration file of smpppd is `/etc/smpppd.conf`. By default, it does not enable remote control. The most important options of this configuration file are:

open-inet-socket = *`yes|no`*

To control smpppd via the network, this option must be set to `yes`. The port on which smpppd listens is `3185`. If this parameter is set to `yes`, the parameters `bind-address`, `host-range`, and `password` should also be set accordingly.

bind-address = *`ip`*

If a host has several IP addresses, use this parameter to determine at which IP address smpppd should accept connections.

host-range = ***`min ip`*****`max ip`***

The parameter `host-range` defines a network range. Hosts whose IP addresses are within this range are granted access to smpppd. All hosts not within this range are denied access.

password = ***`password`***

By assigning a password, limit the clients to authorized hosts. As this is a plain-text password, you should not overrate the security it provides. If no password is assigned, all clients are permitted to access smpppd.

slp-register = ***`yes|no`***

With this parameter, the smpppd service can be announced in the network via SLP.

More information about smpppd is available in the `smpppd(8)` and `smpppd.conf(5)` man pages.

38.6.2 Configuring KInternet, cinternet, and qinternet for Remote Use

KInternet, cinternet, and qinternet can be used to control a local or remote smpppd. cinternet is the command-line counterpart of the graphical KInternet. qinternet is basicallly the same as KInternet, but does not use the KDE libraries, so it can be used without KDE and must be installed separately. To prepare these utilities for use with a remote smpppd, edit the configuration file `/etc/smpppd-c.conf` manually or using KInternet. This file only uses three options:

sites = ***`list of sites`***

Here, tell the front-ends where to search for smpppd. The front-ends test the options in the order specified here. The `local` option orders the establishment of a connection to the local smpppd. `gateway` points to an smpppd on the gateway. The connection should be established as specified under `server` in `config-file`. `slp` orders the front-ends to connect to an smpppd found via SLP.

server = ***`server`***

Here, specify the host on which smpppd runs.

password = ***`password`***

Insert the password selected for smpppd.

If smpppd is active, you can now try to access it, for example, with `cinternet --verbose --interface-list`. If you experience difficulties at this point, refer to the `smpppd-c.conf(5)` and `cinternet(8)` man pages.

39 SLP Services in the Network

The *service location protocol* (SLP) was developed to simplify the configuration of networked clients within a local network. To configure a network client, including all required services, the administrator traditionally needs detailed knowledge of the servers available in the network. SLP makes the availability of selected services known to all clients in the local network. Applications that support SLP can use the information distributed and be configured automatically.

SUSE Linux supports installation using installation sources provided with SLP and contains many system services with integrated support for SLP. YaST and Konqueror both have appropriate front-ends for SLP. You can use SLP to provide networked clients with central functions, such as an installation server, YOU server, file server, or print server on your SUSE Linux.

39.1 Registering Your Own Services

Many applications under SUSE Linux already have integrated SLP support through the use of the `libslp` library. If a service has not been compiled with SLP support, use one of the following methods to make it available with SLP:

Static Registration with `/etc/slp.reg.d`

Create a separate registration file for each new service. The following is an example of a file for registering a scanner service:

```
## Register a saned service on this system
## en means english language
## 65535 disables the timeout, so the service registration does
## not need refreshes
service:scanner.sane://$HOSTNAME:6566,en,65535
watch-port-tcp=6566
description=SANE scanner daemon
```

The most important line in this file is the *service URL*, which begins with `service:`. This contains the service type (`scanner.sane`) and the address under which the service is available on the server. *`$HOSTNAME`* is automatically replaced with the full hostname. The name of the TCP port on which the relevant service can be found follows, separated by a colon. Then enter the language in which the service should appear and the duration of registration in seconds. These should be separated from the service URL by commas. Set the value for the duration of registration between `0` and `65535`. `0` prevents registration. `65535` removes all restrictions.

The registration file also contains the two variables `watch-tcp-port` and `description`. `watch-tcp-port` links the SLP service announcement to whether the relevant service is active by having slpd check the status of the service. The second variable contains a more precise description of the service that is displayed in suitable browsers.

Static Registration with `/etc/slp.reg`

The only difference from the procedure with `/etc/slp.reg.d` is the grouping of all services within a central file.

Dynamic Registration with slptool

If a service should be registered for SLP from proprietary scripts, use the slptool command line front-end.

39.2 SLP Front-Ends in SUSE Linux

SUSE Linux contains several front-ends that enable SLP information to be checked and used by means of a network:

slptool

slptool is a simple command line program that can be used to announce SLP inquiries in the network or announce proprietary services. `slptool --help` lists all available options and functions. slptool can also be called from scripts that process SLP information.

YaST SLP Browser
YaST contains a separate SLP browser that lists all services in the local network announced by SLP in a tree diagram under *Network Services* → *SLP Browser*.

Konqueror
When used as a network browser, Konqueror can display all SLP services available in the local network at `slp:/`. Click the icons in the main window to obtain more detailed information about the relevant service. If you use Konqueror with `service:/`, click the relevant icon once in the browser window to set up a connection with the selected service.

39.3 Activating SLP

slpd must run on your system if you want to offer services. It is not necessary to start this daemon simply to make service inquiries. Like most system services in SUSE Linux, the slpd daemon is controlled by means of a separate init script. The daemon is inactive by default. To activate it for the duration of a session, run `rcslpd start` as `root` to start it and `rcslpd stop` to stop it. Perform a restart or status check with `restart` or `status`. If slpd should be active by default, run the `insserv slpd` command once as `root`. This automatically includes slpd in the set of services to start when a system boots.

39.4 For More Information

The following sources provide further information about SLP:

RFC 2608, 2609, 2610
RFC 2608 generally deals with the definition of SLP. RFC 2609 deals with the syntax of the service URLs used in greater detail and RFC 2610 deals with DHCP via SLP.

http://www.openslp.com
The home page of the OpenSLP project.

/usr/share/doc/packages/openslp
This directory contains all available documentation for SLP, including a `README.SuSE` containing the SUSE Linux details, the RFCs mentioned above, and two

introductory HTML documents. Programmers who want to use the SLP functions should install the `openslp-devel` package to consult its supplied *Programmers Guide*.

The Domain Name System 40

DNS (domain name system) is needed to resolve the domain names and hostnames into IP addresses. In this way, the IP address 192.168.0.0 is assigned to the hostname `earth`, for example. Before setting up your own name server, read the general information about DNS in Section 38.3, "Name Resolution" (page 562). The following configuration examples refer to BIND.

40.1 DNS Basics

40.2 Configuration with YaST

You can use the DNS module of YaST to configure a DNS server for your local network. When starting the module for the first time, a wizard starts, prompting you to make just a few basic decisions concerning administration of the server. Completing this initial setup produces a very basic server configuration that should be functioning in its essential aspects. The expert mode can be used to deal with more advanced configuration tasks.

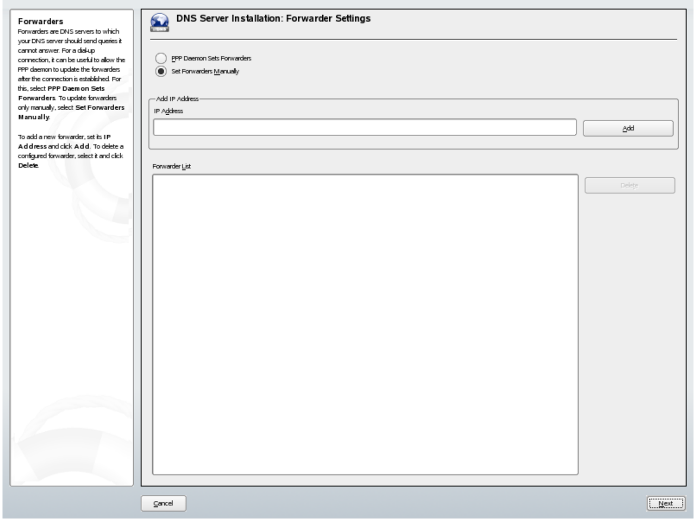
40.2.1 Wizard Configuration

The wizard consists of three steps or dialogs. At the appropriate places in the dialogs, you are given the opportunity to enter the expert configuration mode.

Forwarder Settings

When starting the module for the first time, see the dialog shown in Figure 40.1, "DNS Server Installation: Forwarder Settings" (page 594). In it, decide whether the PPP daemon should provide a list of forwarders on dial-up via DSL or ISDN (*PPP Daemon Sets Forwarders*) or whether you want to supply your own list (*Set Forwarders Manually*).

Figure 40.1 *DNS Server Installation: Forwarder Settings*



Forwarders

Forwarders are DNS servers to which your DNS server should send queries it cannot answer. For a dial-up connection, it can be useful to allow the PPP daemon to update the forwarders after the connection is established. For this, select **PPP Daemon Sets Forwarders**. To update forwarders only manually, select **Set Forwarders Manually**.

To add a new forwarder, set its **IP Address** and click **Add**. To delete a configured forwarder, select it and click **Delete**.

DNS Server Installation: Forwarder Settings

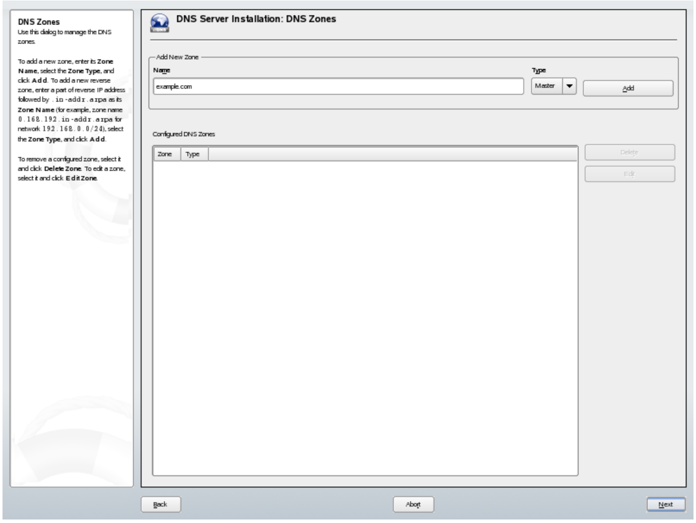
○ PPP Daemon Sets Forwarders

● Set Forwarders Manually

DNS Zones

This dialog consists of several parts and is responsible for the management of zone files, described in Section 40.5, "Zone Files" (page 607). For a new zone, provide a name for it in *Zone Name*. To add a reverse zone, the name must end in `.in-addr.arpa`. Finally, select the *Zone Type* (master or slave). See Figure 40.2, "DNS Server Installation: DNS Zones" (page 595). Click *Edit Zone* to configure other settings of an existing zone. To remove a zone, click *Delete Zone*.

Figure 40.2 *DNS Server Installation: DNS Zones*



DNS Zones

Use this dialog to manage the DNS zones.

To add a new zone, enter its **Zone Name**, select the **Zone Type**, and click **Add**. To add a new reverse zone, enter a part of reverse IP address followed by `.in-addr.arpa` as its **Zone Name** (for example, zone name `0.168.192.in-addr.arpa` for network `192.168.0.0/24`), select the **Zone Type**, and click **Add**.

To remove a configured zone, select it and click **Delete Zone**. To edit a zone, select it and click **Edit Zone**.

DNS Server Installation: DNS Zones

Add New Zone

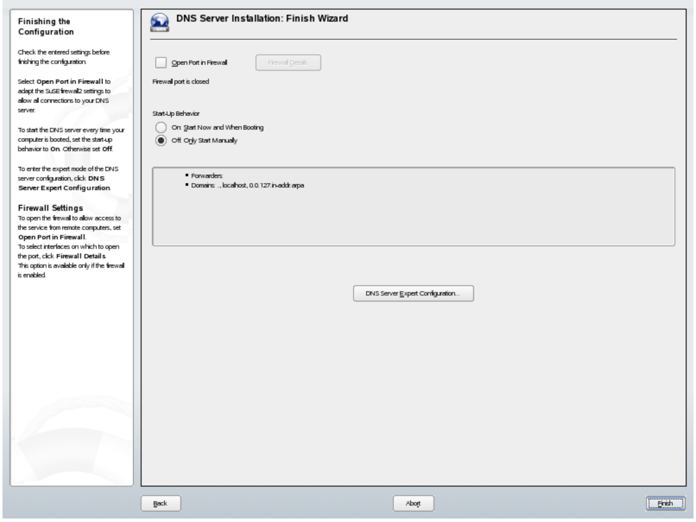
Name: example.com

Type: Master

Finish Wizard

In the final dialog, you can open the ports for the DNS service in the firewall that is activated during the installation and decide whether DNS should be started. The expert configuration can also be accessed from this dialog. See Figure 40.3, "DNS Server Installation: Finish Wizard" (page 596).

Figure 40.3 *DNS Server Installation: Finish Wizard*



Finishing the Configuration

Check the entered settings before finishing the configuration.

Select **Open Port in Firewall** to adapt the SuSE firewall2 settings to allow all connections to your DNS server.

To start the DNS server every time your computer is booted, set the start-up behavior to **On**. Otherwise set **Off**.

To enter the expert mode of the DNS server configuration, click **DNS Server Expert Configuration**.

Firewall Settings

To open the firewall to allow access to the service from remote computers, set **Open Port in Firewall**.
To select interfaces on which to open the port, click **Firewall Details**.
This option is available only if the firewall is enabled.

DNS Server Installation: Finish Wizard

□ Open Port in Firewall

Firewall port is closed

Start-Up Behavior

○ On: Start Now and When Booting
● Off: Only Start Manually

- Forwarders:
- Domains: ., localhost, 0.0.127.in-addr.arpa

40.2.2 Expert Configuration

After starting the module, YaST opens a window displaying several configuration options. Completing it results in a DNS server configuration with the basic functions in place:

Start-Up

Under *Booting*, define whether the DNS server should be started when the system boots (during booting the system) or manually. To start the DNS server immediately, select *Start DNS Server Now*. To stop the DNS server, select *Stop DNS Server Now*. To save the current settings, select *Save Settings and Restart DNS Server Now*. You can open the DNS port in the firewall with *Open Port in Firewall* and modify the firewall settings with *Firewall Details*.

Forwarders

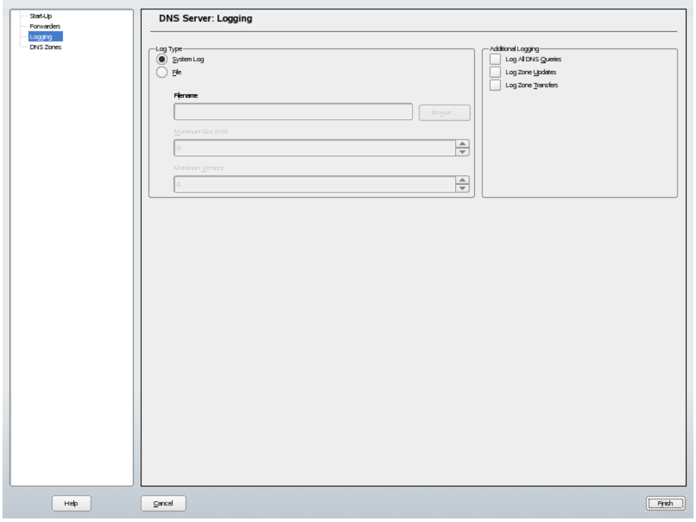
This is the same dialog as the one opened after starting the wizard configuration (see Forwarder Settings (page 594)).

Logging

This section allows you to set what the DNS server should log and how. Under *Log Type*, specify where the DNS server should write the log data. Use the systemwide log file `/var/log/messages` by selecting *Log to System Log* or specify a different file by selecting *Log to File*. In the latter case, additionally specify the maximum file size in megabytes and the number of log files to store.

Further options are available under *Additional Logging*. Enabling *Log All DNS Queries* causes *every* query to be logged, in which case the log file could grow extremely large. For this reason, it is not a good idea to enable this option for other than debugging purposes. To log the data traffic during zone updates between DHCP and DNS server, enable *Log Zone Updates*. To log the data traffic during a zone transfer from master to slave, enable *Log Zone Transfer*. See Figure 40.4, "DNS Server: Logging" (page 597).

Figure 40.4 *DNS Server: Logging*



DNS Server: Logging

Log Type
● System Log
○ File
Filename
Maximum Size (MB): 0
Maximum Versions: 0

Additional Logging
□ Log All DNS Queries
□ Log Zone Updates
□ Log Zone Transfers

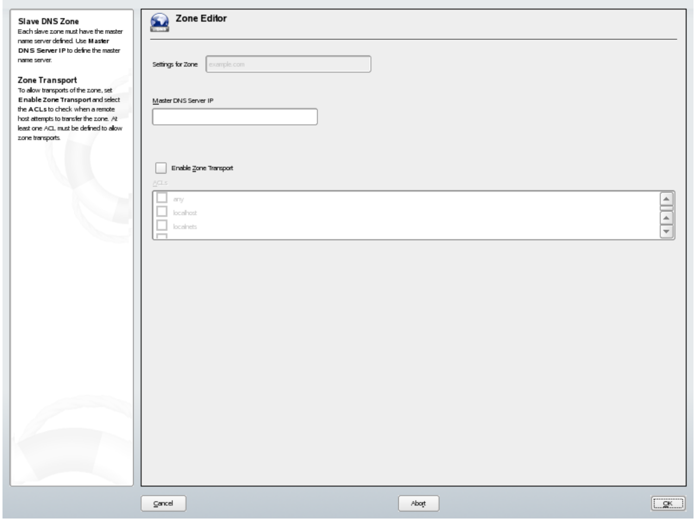
DNS Zones

This dialog is explained for the wizard configuration. See Section 40.2.1, "Wizard Configuration" (page 593).

Slave Zone Editor

This dialog opens if you selected the zone type *Slave* in the step described in DNS Zones (page 597). Under *Master DNS Server*, specify the master from which the slave should fetch its data. To limit access to the server, select one of the ACLs from the list. See Figure 40.5, "DNS Server: Slave Zone Editor" (page 598).

Figure 40.5 *DNS Server: Slave Zone Editor*



Slave DNS Zone

Each slave zone must have the master name server defined. Use **Master DNS Server IP** to define the master name server.

Zone Transport

To allow transports of the zone, set **Enable Zone Transport** and select the **ACLs** to check when a remote host attempts to transfer the zone. At least one ACL must be defined to allow zone transports.

Settings for Zone: example.com

Master DNS Server IP:

□ Enable Zone Transport

ACLs

□ any
□ localhost
□ localnets

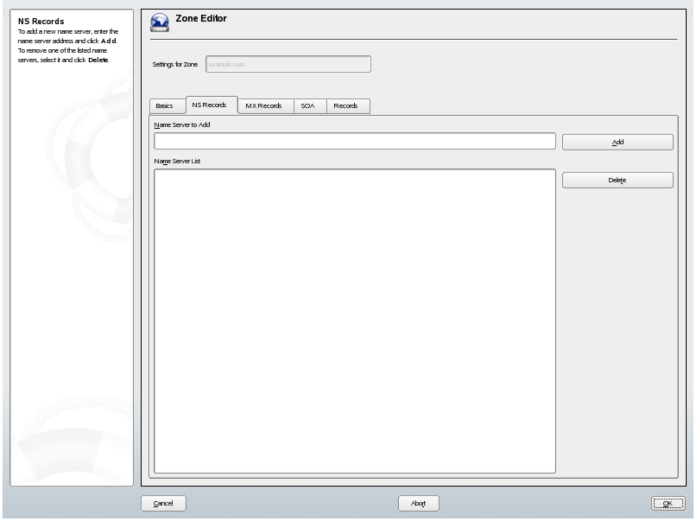
Master Zone Editor

This dialog opens if you selected the zone type *Master* in the step described in DNS Zones (page 597). The dialog comprises several pages: *Basic* (the one opened first), *NS Records*, *MX Records*, *SOA*, and *Records*.

Zone Editor (NS Records)

This dialog allows you to define alternative name servers for the zones specified. Make sure that your own name server is included in the list. To add a record, enter its name under *Name Server to Add* then confirm with *Add*. See Figure 40.6, “DNS Server: Zone Editor (NS Records)” (page 599).

Figure 40.6 DNS Server: Zone Editor (NS Records)

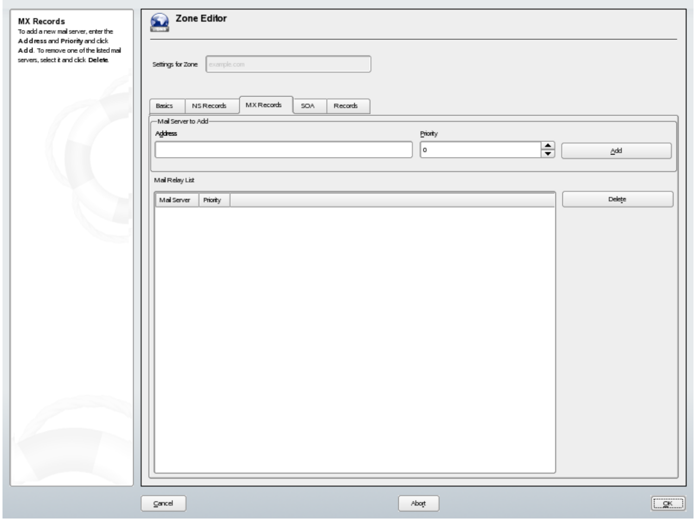

NS Records
To add a new name server, enter the name server address and click **Add**. To remove one of the listed name servers, select it and click **Delete**.

Settings for Zone: example.com

Zone Editor (MX Records)

To add a mail server for the current zone to the existing list, enter the corresponding address and priority value. After doing so, confirm by selecting *Add*. See Figure 40.7, “DNS Server: Zone Editor (MX Records)” (page 600).

Figure 40.7 DNS Server: Zone Editor (MX Records)



MX Records

To add a new mail server, enter the **Address** and **Priority** and click **Add**. To remove one of the listed mail servers, select it and click **Delete**.

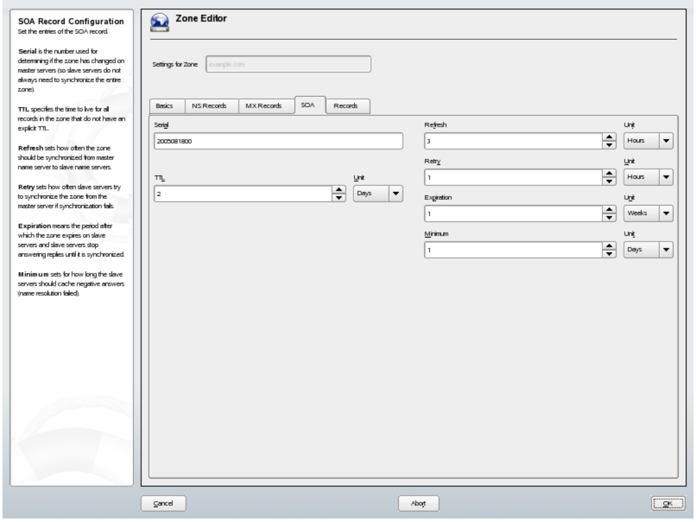
Settings for Zone: example.com

Priority: 0

Zone Editor (SOA)

This page allows you to create SOA (start of authority) records. For an explanation of the individual options, refer to Example 40.6, “File /var/lib/named/world.zone” (page 608).

Figure 40.8 DNS Server: Zone Editor (SOA)



SOA Record Configuration
Set the entries of the SOA record.

Serial is the number used for determining if the zone has changed on master servers (so slave servers do not always need to synchronize the entire zone).

TTL specifies the time to live for all records in the zone that do not have an explicit TTL.

Refresh sets how often the zone should be synchronized from master name server to slave name servers.

Retry sets how often slave servers try to synchronize the zone from the master server if synchronization fails.

Expiration means the period after which the zone expires on slave servers and slave servers stop answering replies until it is synchronized.

Minimum sets for how long the slave servers should cache negative answers (name resolution failed).

Zone Editor

Settings for Zone: example.com

Serial: 2005081800

TTL: 2 Unit: Days

Refresh: 3 Unit: Hours

Retry: 1 Unit: Hours

Expiration: 1 Unit: Weeks

Minimum: 1 Unit: Days

Zone Editor (Records)

This dialog manages name resolution. In *Record Key*, enter the hostname then select its type. *A-Record* represents the main entry. The value for this should be an IP address. *CNAME* is an alias. Use the types *NS* and *MX* for detailed or partial records that expand on the information provided in the *NS Records* and *MX Records* tabs. These three types resolve to an existing A record. *PTR* is for reverse zones. It is the opposite of an A record.

40.3 Starting the Name Server BIND

On a SUSE Linux system, the name server BIND (*Berkeley Internet name domain*) comes preconfigured so it can be started right after installation without any problem. If you already have a functioning Internet connection and have entered `127.0.0.1` as the name server address for `localhost` in `/etc/resolv.conf`, you normally already have a working name resolution without needing to know the DNS of the provider. BIND carries out name resolution via the root name server, a notably slower process. Normally, the DNS of the provider should be entered with its IP address in the configuration file `/etc/named.conf` under `forwarders` to ensure effective and secure name resolution. If this works so far, the name server runs as a pure *caching-only* name server. Only when you configure its own zones will it become a proper DNS.

A simple example of this is included in the documentation in /usr/share/doc/packages/bind/sample-config.

TIP: Automatic Adaptation of the Name Server Information

Depending on the type of Internet connection or the network connection, the name server information can automatically be adapted to the current conditions. To do this, set the variable MODIFY_NAMED_CONF_DYNAMICALLY in the file /etc/sysconfig/network/config to yes.

However, do not set up any official domains until assigned one by the responsible institution. Even if you have your own domain and it is managed by the provider, you are better off not using it, because BIND would otherwise not forward requests for this domain. The Web server at the provider, for example, would not be accessible for this domain.

To start the name server, enter the command rcnamed start as root. If “done” appears to the right in green, named, as the name server process is called, has been started successfully. Test the name server immediately on the local system with the host or dig programs, which should return localhost as the default server with the address 127.0.0.1. If this is not the case, /etc/resolv.conf probably contains an incorrect name server entry or the file does not exist at all. For the first test, enter host 127.0.0.1, which should always work. If you get an error message, use rcnamed status to see whether the server is actually running. If the name server does not start or behaves unexpectedly, you can usually find the cause in the log file /var/log/messages.

To use the name server of the provider or one already running on your network as the forwarder, enter the corresponding IP address or addresses in the options section under forwarders. The addresses included in Example 40.1, “Forwarding Options in named.conf” (page 602) are just examples. Adjust these entries to your own setup.

***Example 40.1** Forwarding Options in named.conf*

```
options {
        directory "/var/lib/named";
        forwarders { 10.11.12.13; 10.11.12.14; };
        listen-on { 127.0.0.1; 192.168.0.99; };
        allow-query { 127/8; 192.168.0/24; };
        notify no;
        };
```

The options entry is followed by entries for the zone, localhost, and 0.0.127.in-addr.arpa. The type hint entry under "." should always be present. The corresponding files do not need to be modified and should work as they are. Also make sure that each entry is closed with a ";" and that the curly braces are in the correct places. After changing the configuration file /etc/named.conf or the zone files, tell BIND to reread them with rcnamed reload. Achieve the same by stopping and restarting the name server with rcnamed restart. Stop the server at any time by entering rcnamed stop.

40.4 The Configuration File /etc/named.conf

All the settings for the BIND name server itself are stored in the file /etc/named.conf. However, the zone data for the domains to handle, consisting of the hostnames, IP addresses, and so on, are stored in separate files in the /var/lib/named directory. The details of this are described later.

/etc/named.conf is roughly divided into two areas. One is the options section for general settings and the other consists of zone entries for the individual domains. A logging section and acl (access control list) entries are optional. Comment lines begin with a # sign or //. A minimal /etc/named.conf is shown in Example 40.2, "A Basic /etc/named.conf" (page 604).

***Example 40.2** A Basic /etc/named.conf*

```
options {
        directory "/var/lib/named";
        forwarders { 10.0.0.1; };
        notify no;
};

zone "localhost" in {
       type master;
       file "localhost.zone";
};

zone "0.0.127.in-addr.arpa" in {
        type master;
        file "127.0.0.zone";
};

zone "." in {
        type hint;
        file "root.hint";
};
```

40.4.1 Important Configuration Options

directory "*filename*";

Specifies the directory in which BIND can find the files containing the zone data. Usually, this is `/var/lib/named`.

forwarders { *ip-address*; };

Specifies the name servers (mostly of the provider) to which DNS requests should be forwarded if they cannot be resolved directly. Replace *`ip-address`* with an IP address like `10.0.0.1`.

forward first;

Causes DNS requests to be forwarded before an attempt is made to resolve them via the root name servers. Instead of `forward first`, `forward only` can be written to have all requests forwarded and none sent to the root name servers. This makes sense for firewall configurations.

listen-on port 53 { 127.0.0.1; *ip-address*; };

Tells BIND on which network interfaces and port to accept client queries. `port 53` does not need to be specified explicitly, because `53` is the default port. Enter

`127.0.0.1` to permit requests from the local host. If you omit this entry entirely, all interfaces are used by default.

listen-on-v6 port 53 {any; };

Tells BIND on which port it should listen for IPv6 client requests. The only alternative to `any` is `none`. As far as IPv6 is concerned, the server only accepts a wild card address.

query-source address * port 53;

This entry is necessary if a firewall is blocking outgoing DNS requests. This tells BIND to post requests externally from port 53 and not from any of the high ports above 1024.

query-source-v6 address * port 53;

Tells BIND which port to use for IPv6 queries.

allow-query { 127.0.0.1; *net*; };

Defines the networks from which clients can post DNS requests. Replace *net* with address information like `192.168.1/24`. The `/24` at the end is an abbreviated expression for the netmask, in this case, `255.255.255.0`.

allow-transfer ! *;;

Controls which hosts can request zone transfers. In the example, such requests are completely denied with `! *`. Without this entry, zone transfers can be requested from anywhere without restrictions.

statistics-interval 0;

In the absence of this entry, BIND generates several lines of statistical information per hour in `/var/log/messages`. Set it to 0 to suppress these statistics completely or set an interval in minutes.

cleaning-interval 720;

This option defines at which time intervals BIND clears its cache. This triggers an entry in `/var/log/messages` each time it occurs. The time specification is in minutes. The default is 60 minutes.

interface-interval 0;

BIND regularly searches the network interfaces for new or nonexisting interfaces. If this value is set to `0`, this is not done and BIND only listens at the interfaces detected at start-up. Otherwise, the interval can be defined in minutes. The default is sixty minutes.

notify no;

no prevents other name servers from being informed when changes are made to the zone data or when the name server is restarted.

40.4.2 Logging

What, how, and where logging takes place can be extensively configured in BIND. Normally, the default settings should be sufficient. Example 40.3, “Entry to Disable Logging” (page 606) shows the simplest form of such an entry and completely suppresses any logging.

***Example 40.3** Entry to Disable Logging*

```
logging {
        category default { null; };
};
```

40.4.3 Zone Entries

***Example 40.4** Zone Entry for my-domain.de*

```
zone "my-domain.de" in {
      type master;
      file "my-domain.zone";
      notify no;
};
```

After zone, specify the name of the domain to administer (my-domain.de) followed by in and a block of relevant options enclosed in curly braces, as shown in Example 40.4, “Zone Entry for my-domain.de” (page 606). To define a *slave zone*, switch the type to slave and specify a name server that administers this zone as master (which, in turn, may be a slave of another master), as shown in Example 40.5, “Zone Entry for other-domain.de” (page 606).

***Example 40.5** Zone Entry for other-domain.de*

```
zone "other-domain.de" in {
      type slave;
      file "slave/other-domain.zone";
      masters { 10.0.0.1; };
};
```

The zone options:

type master;

By specifying `master`, tell BIND that the zone is handled by the local name server. This assumes that a zone file has been created in the correct format.

type slave;

This zone is transferred from another name server. It must be used together with `masters`.

type hint;

The zone `.` of the `hint` type is used to set the root name servers. This zone definition can be left as is.

file `my-domain.zone` or file "slave/other-domain.zone";

This entry specifies the file where zone data for the domain is located. This file is not required for a slave, because this data is fetched from another name server. To differentiate master and slave files, use the directory `slave` for the slave files.

masters { *`server-ip-address`*; };

This entry is only needed for slave zones. It specifies from which name server the zone file should be transferred.

allow-update {! *; };

This option controls external write access, which would allow clients to make a DNS entry—something not normally desirable for security reasons. Without this entry, zone updates are not allowed at all. The above entry achieves the same because `! *` effectively bans any such activity.

40.5 Zone Files

Two types of zone files are needed. One assigns IP addresses to hostnames and the other does the reverse: it supplies a hostname for an IP address.

TIP: Using the Dot in Zone Files

The `.` has an important meaning in the zone files. If hostnames are given without a final `.`, the zone is appended. Complete hostnames specified with a full domain name must end with a `.` to avoid having the domain added to it again. A missing or wrongly placed dot is probably the most frequent cause of name server configuration errors.

The first case to consider is the zone file `world.zone`, responsible for the domain `world.cosmos`, shown in Example 40.6, “File /var/lib/named/world.zone” (page 608).

***Example 40.6** File /var/lib/named/world.zone*

```
$TTL 2D
world.cosmos. IN SOA      gateway  root.world.cosmos. (
            2003072441  ; serial
            1D          ; refresh
            2H          ; retry
            1W          ; expiry
            2D )        ; minimum

            IN NS       gateway
            IN MX       10 sun

gateway     IN A        192.168.0.1
            IN A        192.168.1.1
sun         IN A        192.168.0.2
moon        IN A        192.168.0.3
earth       IN A        192.168.1.2
mars        IN A        192.168.1.3
www         IN CNAME    moon
```

Line 1:
`$TTL` defines the default time to live that should apply to all the entries in this file. In this example, entries are valid for a period of two days (`2 D`).

Line 2:
This is where the SOA (start of authority) control record begins:

- The name of the domain to administer is `world.cosmos` in the first position. This ends with a `.`, because otherwise the zone would be appended a second time. Alternatively, `@` can be entered here, in which case the zone would be extracted from the corresponding entry in `/etc/named.conf`.

- After `IN SOA` is the name of the name server in charge as master for this zone. The name is expanded from `gateway` to `gateway.world.cosmos`, because it does not end with a `.`.

- An e-mail address of the person in charge of this name server follows. Because the `@` sign already has a special meaning, `.` is entered here instead. For `root@world.cosmos` the entry must read `root.world.cosmos.`. The `.` must be included at the end to prevent the zone from being added.

- The `(` includes all lines up to `)` into the SOA record.

Line 3:
The `serial number` is an arbitrary number that is increased each time this file is changed. It is needed to inform the secondary name servers (slave servers) of changes. For this, a 10 digit number of the date and run number, written as YYYYMMDDNN, has become the customary format.

Line 4:
The `refresh rate` specifies the time interval at which the secondary name servers verify the zone `serial number`. In this case, one day.

Line 5:
The `retry rate` specifies the time interval at which a secondary name server, in case of error, attempts to contact the primary server again. Here, two hours.

Line 6:
The `expiration time` specifies the time frame after which a secondary name server discards the cached data if it has not regained contact to the primary server. Here, it is a week.

Line 7:
The last entry in the SOA record specifies the `negative caching TTL`—the time for which results of unresolved DNS queries from other servers may be cached.

Line 9:
The `IN NS` specifies the name server responsible for this domain. `gateway` is extended to `gateway.world.cosmos` because it does not end with a `.`. There can be several lines like this—one for the primary and one for each secondary name server. If `notify` is not set to `no` in `/etc/named.conf`, all the name servers listed here are informed of the changes made to the zone data.

Line 10:
The MX record specifies the mail server that accepts, processes, and forwards e-mails for the domain `world.cosmos`. In this example, this is the host `sun.world.cosmos`. The number in front of the hostname is the preference value. If there are multiple MX entries, the mail server with the smallest value is

taken first and, if mail delivery to this server fails, an attempt is made with the next higher value.

Lines 12–17:

These are the actual address records where one or more IP addresses are assigned to hostnames. The names are listed here without a `.` because they do not include their domain, so `world.cosmos` is added to all of them. Two IP addresses are assigned to the host `gateway`, because it has two network cards. Wherever the host address is a traditional one (IPv4), the record is marked with `A`. If the address is an IPv6 address, the entry is marked with `A6`. The previous token for IPv6 addresses was `AAAA`, which is now obsolete.

Line 18:

The alias `www` can be used to address `mond` (`CNAME` means *canonical name*).

The pseudodomain `in-addr.arpa` is used for the reverse lookup of IP addresses into hostnames. It is appended to the network part of the address in reverse notation. So `192.168.1` is resolved into `1.168.192.in-addr.arpa`. See Example 40.7, "Reverse Lookup" (page 610).

Example 40.7 *Reverse Lookup*

```
$TTL 2D
1.168.192.in-addr.arpa. IN SOA gateway.world.cosmos. root.world.cosmos. (
                        2003072441      ; serial
                        1D              ; refresh
                        2H              ; retry
                        1W              ; expiry
                        2D )            ; minimum

                        IN NS           gateway.world.cosmos.

1                       IN PTR          gateway.world.cosmos.
2                       IN PTR          earth.world.cosmos.
3                       IN PTR          mars.world.cosmos.
```

Line 1:

$TTL defines the standard TTL that applies to all entries here.

Line 2:

The configuration file should activate reverse lookup for the network `192.168.1.0`. Given that the zone is called `1.168.192.in-addr.arpa`, should not be added to the hostnames. Therefore, all hostnames are entered in their complete form—with their domain and with a . at the end. The remaining entries correspond to those described for the previous world.cosmos example.

Lines 3–7:
See the previous example for world.cosmos.

Line 9:
Again this line specifies the name server responsible for this zone. This time, however, the name is entered in its complete form with the domain and a . at the end.

Lines 11–13:
These are the pointer records hinting at the IP addresses on the respective hosts. Only the last part of the IP address is entered at the beginning of the line, without the . at the end. Appending the zone to this (without the .in-addr.arpa) results in the complete IP address in reverse order.

Normally, zone transfers between different versions of BIND should be possible without any problem.

40.6 Dynamic Update of Zone Data

The term *dynamic update* refers to operations by which entries in the zone files of a master server are added, changed, or deleted. This mechanism is described in RFC 2136. Dynamic update is configured individually for each zone entry by adding an optional allow-update or update-policy rule. Zones to update dynamically should not be edited by hand.

Transmit the entries to update to the server with the command nsupdate. For the exact syntax of this command, check the manual page for nsupdate (man 8 nsupdate). For security reasons, any such update should be performed using TSIG keys as described in Section 40.7, “Secure Transactions” (page 611).

40.7 Secure Transactions

Secure transactions can be made with the help of transaction signatures (TSIGs) based on shared secret keys (also called TSIG keys). This section describes how to generate and use such keys.

Secure transactions are needed for communication between different servers and for the dynamic update of zone data. Making the access control dependent on keys is much more secure than merely relying on IP addresses.

Generate a TSIG key with the following command (for details, see `man dnssec-keygen`):

```
dnssec-keygen -a hmac-md5 -b 128 -n HOST host1-host2
```

This creates two files with names similar to these:

```
Khost1-host2.+157+34265.private Khost1-host2.+157+34265.key
```

The key itself (a string like `ejIkuCyyGJwwuN3xAteKgg==`) is found in both files. To use it for transactions, the second file (`Khost1-host2.+157+34265.key`) must be transferred to the remote host, preferably in a secure way (using scp, for example). On the remote server, the key must be included in the file `/etc/named.conf` to enable a secure communication between `host1` and `host2`:

```
key host1-host2. {
 algorithm hmac-md5;
 secret ";ejIkuCyyGJwwuN3xAteKgg==;
};
```

WARNING: File Permissions of `/etc/named.conf`

Make sure that the permissions of `/etc/named.conf` are properly restricted. The default for this file is `0640`, with the owner being `root` and the group `named`. As an alternative, move the keys to an extra file with specially limited permissions, which is then included from `/etc/named.conf`.

To enable the server `host1` to use the key for `host2` (which has the address `192.168.2.3` in this example), the server's `/etc/named.conf` must include the following rule:

```
server 192.168.2.3 {
  keys { host1-host2. ;};
};
```

Analogous entries must be included in the configuration files of `host2`.

Add TSIG keys for any ACLs (access control lists, not to be confused with file system ACLs) that are defined for IP addresses and address ranges to enable transaction security. The corresponding entry could look like this:

```
allow-update { key host1-host2. ;};
```

This topic is discussed in more detail in the *BIND Administrator Reference Manual* under `update-policy`.

40.8 DNS Security

DNSSEC, or DNS security, is described in RFC 2535. The tools available for DNSSEC are discussed in the BIND Manual.

A zone considered secure must have one or several zone keys associated with it. These are generated with `dnssec-keygen`, just like the host keys. The DSA encryption algorithm is currently used to generate these keys. The public keys generated should be included in the corresponding zone file with an `$INCLUDE` rule.

With the command `dnssec-makekeyset`, all keys generated are packaged into one set, which must then be transferred to the parent zone in a secure manner. On the parent, the set is signed with `dnssec-signkey`. The files generated by this command are then used to sign the zones with `dnssec-signzone`, which in turn generates the files to include for each zone in `/etc/named.conf`.

40.9 For More Information

For additional information, refer to the *BIND Administrator Reference Manual*, which is installed under `/usr/share/doc/packages/bind/`. Consider additionally consulting the RFCs referenced by the manual and the manual pages included with BIND. `/usr/share/doc/packages/bind/README.SuSE` contains up-to-date information about BIND in SUSE Linux.

41 Using NIS

As soon as multiple UNIX systems in a network want to access common resources, it becomes important that all user and group identities are the same for all machines in that network. The network should be transparent to users: whatever machines they use, they always find themselves in exactly the same environment. This is made possible by means of NIS and NFS services. NFS distributes file systems over a network and is discussed in Chapter 42, *Sharing File Systems with NFS* (page 623).

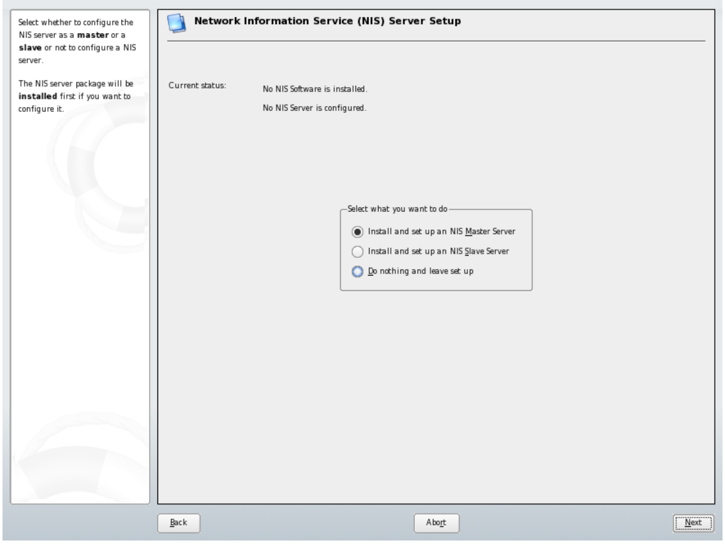
NIS (Network Information Service) can be described as a database-like service that provides access to the contents of `/etc/passwd`, `/etc/shadow`, and `/etc/group` across networks. NIS can also be used for other purposes (making the contents of files like `/etc/hosts` or `/etc/services` available, for example), but this is beyond the scope of this introduction. People often refer to NIS as *YP*, because it works like the network's “yellow pages.”

41.1 Configuring NIS Servers Using YaST

For configuration, select *Network Services* → *NIS Server* from the YaST control center. If there is no NIS server yet in your network, activate *Install and set up a NIS Master Server* in the next screen. YaST immediately installs the required packages.

If you have already installed NIS software, click *Create NIS Master Server*. If you already have an NIS server (a *master*), you can add a NIS slave server (for example, if you want to configure a new subnetwork). First, the configuration of the master server is described. Clicking *Do nothing and leave setup* takes you back to the YaST Control Center with no saved changes.

Figure 41.1 *NIS Server Setup*



Select whether to configure the NIS server as a **master** or a **slave** or not to configure a NIS server.

The NIS server package will be **installed** first if you want to configure it.

Network Information Service (NIS) Server Setup

Current status: No NIS Software is installed.

No NIS Server is configured.

Select what you want to do

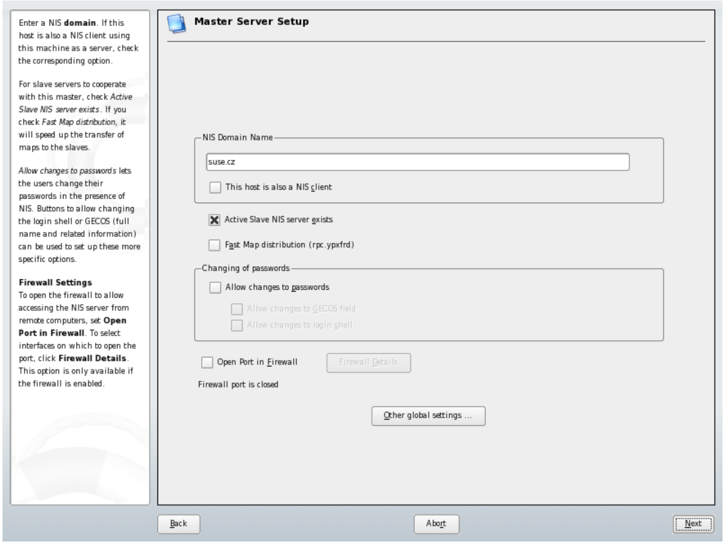
● Install and set up an NIS Master Server

○ Install and set up an NIS Slave Server

○ Do nothing and leave set up

After all packages have been installed, enter the NIS domain name at the top of the configuration dialog, which is shown in Figure 41.2, “Master Server Setup” (page 617). With the check box, define whether the host should also be a NIS client, enabling users to log in and access data from the NIS server. Check the boxes to apply, including the *Changing of paswords* option. Further options can be set by clicking *Other global settings*. Here, access a screen in which you can change the source directory, merge passwords, and set minimum user and group IDs. Click *OK* to return to the main dialog. Click *Next* to continue with configuration.

Figure 41.2 *Master Server Setup*



Enter a NIS **domain**. If this host is also a NIS client using this machine as a server, check the corresponding option.

For slave servers to cooperate with this master, check *Active Slave NIS server exists*. If you check *Fast Map distribution*, it will speed up the transfer of maps to the slaves.

Allow changes to passwords lets the users change their passwords in the presence of NIS. Buttons to allow changing the login shell or GECOS (full name and related information) can be used to set up these more specific options.

Firewall Settings
To open the firewall to allow accessing the NIS server from remote computers, set **Open Port in Firewall**. To select interfaces on which to open the port, click **Firewall Details**. This option is only available if the firewall is enabled.

Master Server Setup

NIS Domain Name: suse.cz

□ This host is also a NIS client

☒ Active Slave NIS server exists

□ Fast Map distribution (rpc.ypxfrd)

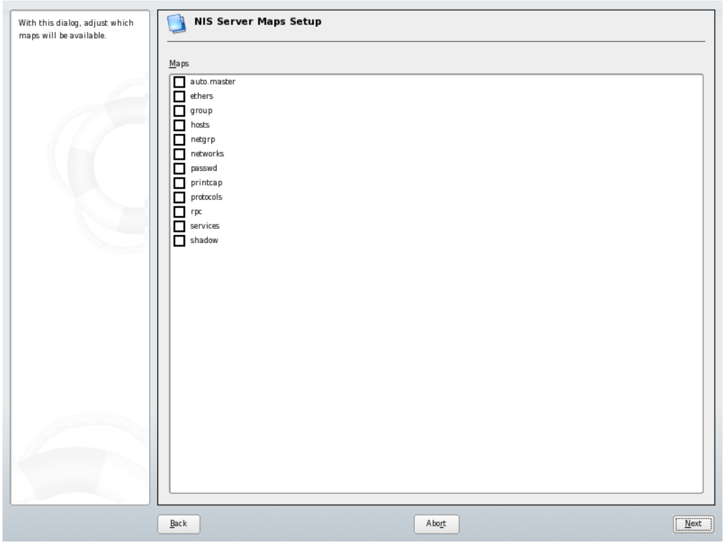
Changing of passwords
□ Allow changes to passwords
□ Allow changes to GECOS field
□ Allow changes to login shell

□ Open Port in Firewall

Firewall port is closed

In the next screen, specify which maps should be made available. Clicking *Next* takes you to the following screen in which you determine which hosts are allowed to query the NIS server. You can add, delete, and edit hosts. Click *Finish* to save changes and exit the configuration dialog.

Figure 41.3 *NIS Server Maps Setup*



With this dialog, adjust which maps will be available.

NIS Server Maps Setup

Maps

- □ auto.master
- □ ethers
- □ group
- □ hosts
- □ netgrp
- □ networks
- □ passwd
- □ printcap
- □ protocols
- □ rpc
- □ services
- □ shadow

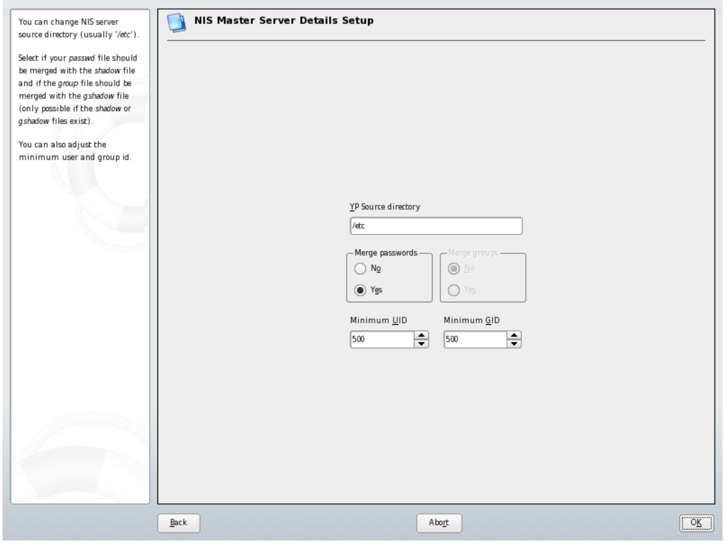
To configure additional NIS *slave servers* in your network, activate *Install and set up a NIS Slave Server* now. If NIS software has already been installed, click *Create NIS Slave Server* and click *Next* to continue. In the next screen, enter the NIS domain name and check the boxes that apply.

To allow users in your network (both local users and those managed through the NIS server) to change their passwords on the NIS server (with the command `yppasswd`), activate the corresponding option. This makes the options *Allow Changes to GECOS Field* and *Allow Changes to Login Shell* available. "GECOS" means that the users can also change their names and address settings with the command `ypchfn`. "SHELL" allows users to change their default shell with the command `ypchsh`, for example, to switch from bash to sh.

Further options can be set by clicking *Other global settings*. Here, access a screen, shown in Figure 41.4, "Changing the Directory and Synchronizing Files for a NIS Server" (page 619), in which to change the source directory of the NIS server (`/etc` by default). In addition, passwords and groups can be merged here. The setting should be *Yes* so the files (`/etc/passwd`, `/etc/shadow`, and `/etc/group`) can be synchronized. Also determine the smallest user and group ID. Click *OK* to confirm your settings and return to the previous screen.

After your settings have been made, advance to the next screen by clicking *Next*. In the next dialog, check which maps should be available then click *Next* to continue. In the final screen, enter which hosts are allowed to query the NIS server. You can add, edit, or delete hosts by clicking the appropriate buttons. Click *Finish* to save changes and exit setup. Then click *Next*.

Figure 41.4 *Changing the Directory and Synchronizing Files for a NIS Server*



You can change NIS server source directory (usually '*/etc*').

Select if your *passwd* file should be merged with the *shadow* file and if the *group* file should be merged with the *gshadow* file (only possible if the *shadow* or *gshadow* files exist).

You can also adjust the minimum user and group id.

NIS Master Server Details Setup

YP Source directory: /etc

Merge passwords: ○ No ● Yes

Merge groups: ● No ○ Yes

Minimum UID: 500

Minimum GID: 500

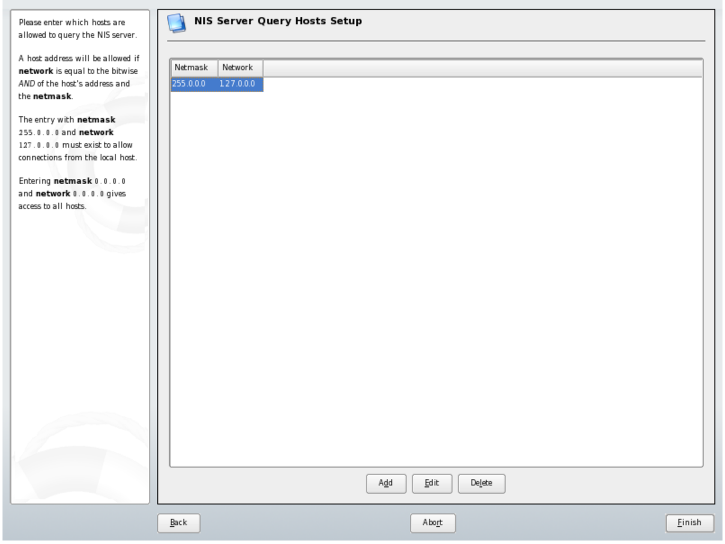
If you previously enabled *Active Slave NIS Server Exists*, enter the hostnames used as slaves and click *Next*. If you do not use slave servers, the slave configuration is skipped and you continue directly to the dialog for the database configuration. Here, specify the *maps*, the partial databases to transfer from the NIS server to the client. The default settings are usually adequate.

Next continues to the last dialog, shown in Figure 41.5, “Setting Request Permissions for a NIS Server” (page 620). Specify from which networks requests can be sent to the NIS server. Normally, this is your internal network. In this case, there should be the following two entries:

```
255.0.0.0     127.0.0.0
0.0.0.0       0.0.0.0
```

The first entry enables connections from your own host, which is the NIS server. The second one allows all hosts with access to the same network to send requests to the server.

Figure 41.5 *Setting Request Permissions for a NIS Server*



Please enter which hosts are allowed to query the NIS server.

A host address will be allowed if **network** is equal to the bitwise *AND* of the host's address and the **netmask**.

The entry with **netmask** `255.0.0.0` and **network** `127.0.0.0` must exist to allow connections from the local host.

Entering **netmask** `0.0.0.0` and **network** `0.0.0.0` gives access to all hosts.

NIS Server Query Hosts Setup

Netmask	Network
255.0.0.0	127.0.0.0

IMPORTANT: Automatic Firewall Configuration

If a firewall is active on your system (SuSEfirewall2), YaST adapts its configuration for the NIS server by enabling the `portmap` service when *Open Ports in Firewall* is selected.

41.2 Configuring NIS Clients

Use this module to configure a NIS client. After you choose to use NIS and, depending on the circumstances, the automounter, this dialog opens. Select whether the host has a static IP address or receives one issued by DHCP. DHCP also provides the NIS domain and the NIS server. For information about DHCP, see Chapter 43, *DHCP* (page 629). If a static IP address is used, specify the NIS domain and the NIS server manually. See Figure 41.6, “Setting Domain and Address of a NIS Server” (page 621). *Find* makes

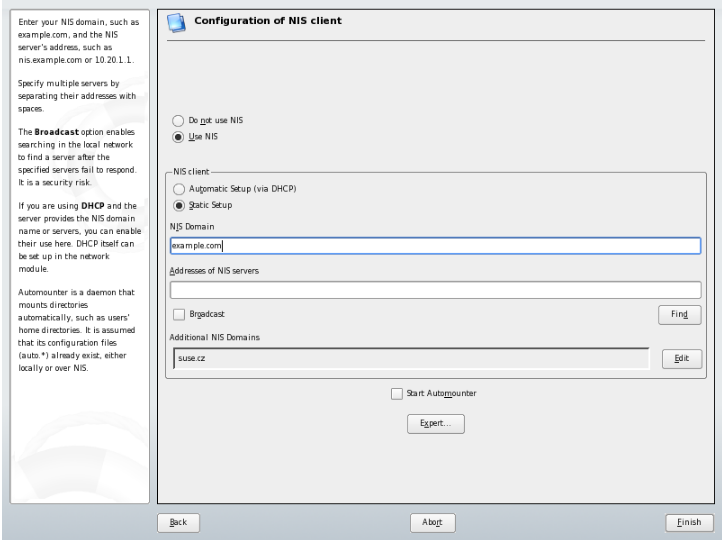
YaST search for an active NIS server in your network. *Broadcast* enables searching in the local network to find a server after the specified servers fail to respond.

You can also specify multiple servers by entering their addresses in *Addresses of NIS servers* and separating them by spaces.

In the expert settings, disable *Answer Remote Hosts* if you do not want other hosts to be able to query which server your client is using. By checking *Broken Server*, the client is enabled to receive replies from a server communicating through an unprivileged port. For further information, see `man ypbind`.

After you have made your settings, click *Finish* to save them and return to the YaST control center.

Figure 41.6 *Setting Domain and Address of a NIS Server*



Enter your NIS domain, such as example.com, and the NIS server's address, such as nis.example.com or 10.20.1.1.

Specify multiple servers by separating their addresses with spaces.

The **Broadcast** option enables searching in the local network to find a server after the specified servers fail to respond. It is a security risk.

If you are using **DHCP** and the server provides the NIS domain name or servers, you can enable their use here. DHCP itself can be set up in the network module.

Automounter is a daemon that mounts directories automatically, such as users' home directories. It is assumed that its configuration files (auto.*) already exist, either locally or over NIS.

Configuration of NIS client

○ Do not use NIS
● Use NIS

NIS client
○ Automatic Setup (via DHCP)
● Static Setup

NIS Domain: example.com

Addresses of NIS servers:

□ Broadcast

Additional NIS Domains: suse.cz

□ Start Automounter

Sharing File Systems with NFS 42

As mentioned in Chapter 41, *Using NIS* (page 615), NFS works with NIS to make a network transparent to the user. With NFS, it is possible to distribute file systems over the network. It does not matter at which terminal users are logged in. They always find themselves in the same environment.

Like NIS, NFS is a client/server system. A machine can be both—it can supply file systems over the network (export) and mount file systems from other hosts (import).

IMPORTANT: Need for DNS

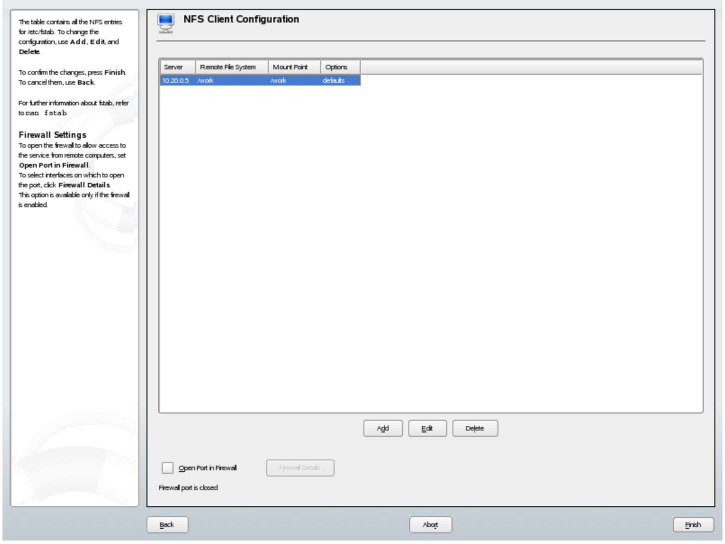
In principle, all exports can be made using IP addresses only. To avoid timeouts, however, you should have a working DNS system. This is necessary at least for logging purposes, because the mountd daemon does reverse lookups.

42.1 Importing File Systems with YaST

Users authorized to do so can mount NFS directories from an NFS server into their own file trees. This can be achieved most easily using the YaST module *NFS Client*. Just enter the hostname of the NFS server, the directory to import, and the mount point at which to mount this directory locally. All this is done after *Add* is clicked in the first dialog. Click *Open Port in Firewall* to open the firewall to allow access to the service from remote computers. The firewall status is displayed next to the check box. Clicking

OK saves your changes. See Figure 42.1, “NFS Client Configuration with YaST” (page 624).

Figure 42.1 *NFS Client Configuration with YaST*



The table contains all the NFS entries for /etc/fstab. To change the configuration, use **Add**, **Edit**, and **Delete**.

To confirm the changes, press **Finish**. To cancel them, use **Back**.

For further information about fstab, refer to `man fstab`.

Firewall Settings
To open the firewall to allow access to the service from remote computers, set **Open Port in Firewall**.
To select interfaces on which to open the port, click **Firewall Details**.
This option is available only if the firewall is enabled.

NFS Client Configuration

Server	Remote File System	Mount Point	Options
10.20.0.5	/work	/work	defaults

□ Open Port in Firewall

Firewall port is closed

42.2 Importing File Systems Manually

File systems can easily be imported manually from an NFS server. The only prerequisite is a running RPC port mapper, which can be started by entering the command `rcportmap start` as `root`. Once this prerequisite is met, remote file systems exported on the respective machines can be mounted in the file system just like local hard disks using the command `mount` with the following syntax:

```
mount host:remote-path local-path
```

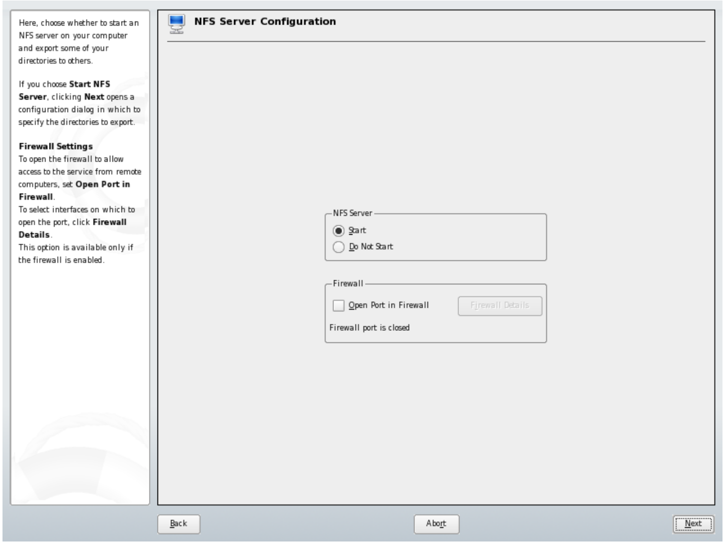
If user directories from the machine sun, for example, should be imported, use the following command:

```
mount sun:/home /home
```

42.3 Exporting File Systems with YaST

With YaST, turn a host in your network into an NFS server—a server that exports directories and files to all hosts granted access to it. This could be done to provide applications to all members of a group without installing them locally on each and every host. To install such a server, start YaST and select *Network Services* → *NFS Server*. A dialog like that in Figure 42.2, “NFS Server Configuration Tool” (page 625) opens.

Figure 42.2 *NFS Server Configuration Tool*



Here, choose whether to start an NFS server on your computer and export some of your directories to others.

If you choose **Start NFS Server**, clicking **Next** opens a configuration dialog in which to specify the directories to export.

Firewall Settings
To open the firewall to allow access to the service from remote computers, set **Open Port in Firewall**.
To select interfaces on which to open the port, click **Firewall Details**.
This option is available only if the firewall is enabled.

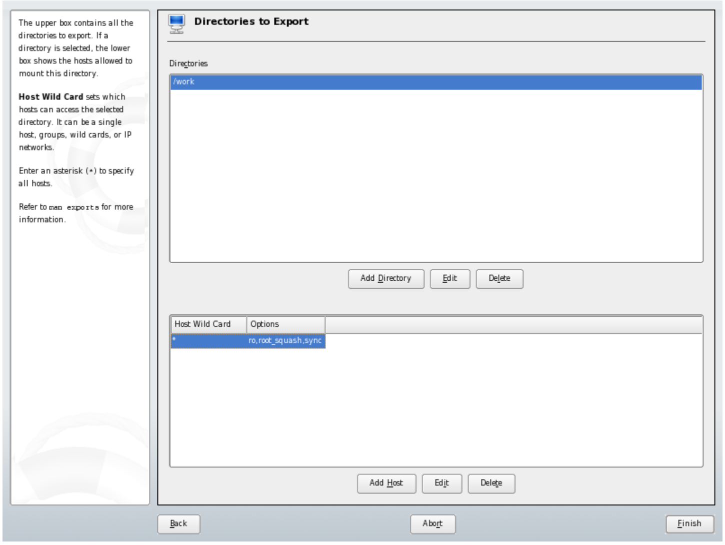
NFS Server Configuration

NFS Server
● Start
○ Do Not Start

Firewall
□ Open Port in Firewall
Firewall port is closed

Next, activate *Start NFS Server* and click *Next*. In the upper text field, enter the directories to export. Below, enter the hosts that should have access to them. This dialog is shown in Figure 42.3, “Configuring an NFS Server with YaST” (page 626). There are four options that can be set for each host: `single host`, `netgroups`, `wildcards`, and `IP networks`. A more thorough explanation of these options is provided by `man exports`. *Exit* completes the configuration.

Figure 42.3 *Configuring an NFS Server with YaST*



The upper box contains all the directories to export. If a directory is selected, the lower box shows the hosts allowed to mount this directory.

Host Wild Card sets which hosts can access the selected directory. It can be a single host, groups, wild cards, or IP networks.

Enter an asterisk (*) to specify all hosts.

Refer to `man exports` for more information.

Directories to Export

Directories

/work

Host Wild Card	Options
*	ro,root_squash,sync

IMPORTANT: Automatic Firewall Configuration

If a firewall is active on your system (SuSEfirewall2), YaST adapts its configuration for the NFS server by enabling the `nfs` service when *Open Ports in Firewall* is selected.

42.4 Exporting File Systems Manually

If you do not want to use YaST, make sure the following systems run on the NFS server:

- RPC portmapper (portmap)
- RPC mount daemon (rpc.mountd)
- RPC NFS daemon (rpc.nfsd)

For these services to be started by the scripts /etc/init.d/portmap and /etc/init.d/nfsserver when the system is booted, enter the commands insserv /etc/init.d/nfsserver and insserv /etc/init.d/portmap. Also define which file systems should be exported to which host in the configuration file /etc/exports.

For each directory to export, one line is needed to set which machines may access that directory with what permissions. All subdirectories of this directory are automatically exported as well. Authorized machines are usually specified with their full names (including domain name), but it is possible to use wild cards like * or ? (which expand the same way as in the Bash shell). If no machine is specified here, any machine is allowed to import this file system with the given permissions.

Set permissions for the file system to export in brackets after the machine name. The most important options are shown in Table 42.1, "Permissions for Exported File System" (page 627).

Table 42.1 *Permissions for Exported File System*

option	meaning
ro	The file system is exported with read-only permission (default).
rw	The file system is exported with read-write permission.
root_squash	This ensures that the user root of an importing machine does not have root permissions on this file system. This is achieved by assigning user ID 65534 to users with user ID 0 (root). This user ID should be set to nobody (which is the default).
no_root_squash	Does not assign user ID 0 to user ID 65534, keeping the root permissions valid.
link_relative	Converts absolute links (those beginning with /) to a sequence of ../. This is only useful if the entire file system of a machine is mounted (default).

option	meaning
link_absolute	Symbolic links remain untouched.
map_identity	User IDs are exactly the same on both client and server (default).
map_daemon	Client and server do not have matching user IDs. This tells nfsd to create a conversion table for user IDs. The ugidd daemon is required for this to work.

Your exports file might look like Example 42.1, “/etc/exports” (page 628). /etc/exports is read by mountd and nfsd. If you change anything in this file, restart mountd and nfsd for your changes to take effect. This can easily be done with rcnfsserver restart.

Example 42.1 */etc/exports*

```
#
# /etc/exports
#
/home              sun(rw)    venus(rw)
/usr/X11           sun(ro)    venus(ro)
/usr/lib/texmf     sun(ro)    venus(rw)
/                  earth(ro,root_squash)
/home/ftp          (ro)
# End of exports
```

DHCP 43

The purpose of the *dynamic host configuration protocol* (DHCP) is to assign network settings centrally from a server rather than configuring them locally on each and every workstation. A host configured to use DHCP does not have control over its own static address. It is enabled to configure itself completely and automatically according to directions from the server.

One way to use DHCP is to identify each client using the hardware address of its network card (which is fixed in most cases), then supply that client with identical settings each time it connects to the server. DHCP can also be configured so the server assigns addresses to each interested client dynamically from an address pool set up for that purpose. In the latter case, the DHCP server tries to assign the same address to the client each time it receives a request from it, even over longer periods. This, of course, only works as long as the network does not have more clients than addresses.

With these possibilities, DHCP can make life easier for system administrators in two ways. Any changes, even bigger ones, related to addresses and the network configuration in general can be implemented centrally by editing the server's configuration file. This is much more convenient than reconfiguring numerous workstations. Also it is much easier to integrate machines, particularly new machines, into the network, because they can be given an IP address from the pool. Retrieving the appropriate network settings from a DHCP server can be especially useful in the case of laptops regularly used in different networks.

A DHCP server supplies not only the IP address and the netmask, but also the hostname, domain name, gateway, and name server addresses for the client to use. In addition to that, DHCP allows for a number of other parameters to be configured in a centralized way, for example, a time server from which clients may poll the current time or even a print server.

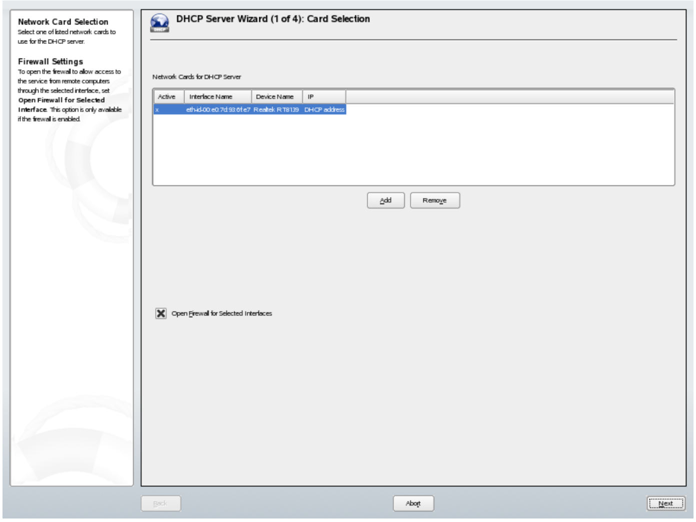
43.1 Configuring a DHCP Server with YaST

When the module is started for the first time, a wizard starts, prompting you to make a few basic decision concerning server administration. Completing this initial setup produces a very basic server configuration that should function in essential aspects. The expert mode can be used to deal with more advanced configuration tasks.

Card Selection

In the first step, YaST looks for the network interfaces available on your system then displays them in a list. From the list, select the interface on which the DHCP server should listen and click *Add* then select *Open Firewall for Selected Interface* to open the firewall for this interface. See Figure 43.1, "DHCP Server: Card Selection" (page 630).

Figure 43.1 *DHCP Server: Card Selection*



Network Card Selection
Select one of listed network cards to use for the DHCP server.

Firewall Settings
To open the firewall to allow access to the service from remote computers through the selected interface, set **Open Firewall for Selected Interface**. This option is only available if the firewall is enabled.

DHCP Server Wizard (1 of 4): Card Selection

Network Cards for DHCP Server

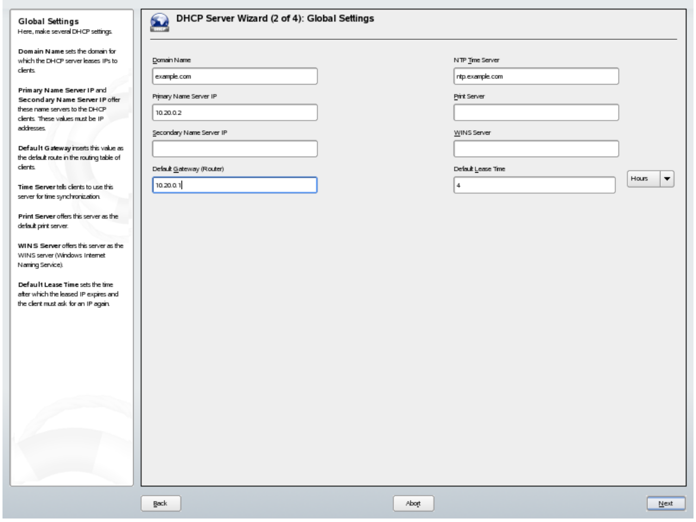
Active	Interface Name	Device Name	IP
x	eth-id-00:e0:7d:93:6f:e7	Realtek RT8139	DHCP address

☒ Open Firewall for Selected Interfaces

Global Settings

In the entry fields, provide the network specifics for all clients the DHCP server should manage. These specifics are the domain name, address of a time server, addresses of the primary and secondary name server, addresses of a print and a WINS server (for a mixed network with both Windows and Linux clients), gateway address, and lease time. See Figure 43.2, "DHCP Server: Global Settings" (page 631).

Figure 43.2 DHCP Server: Global Settings



Global Settings
Here, make several DHCP settings.

Domain Name sets the domain for which the DHCP server leases IPs to clients.

Primary Name Server IP and **Secondary Name Server IP** offer these name servers to the DHCP clients. These values must be IP addresses.

Default Gateway inserts this value as the default route in the routing table of clients.

Time Server tells clients to use this server for time synchronization.

Print Server offers this server as the default print server.

WINS Server offers this server as the WINS server (Windows Internet Naming Service).

Default Lease Time sets the time after which the leased IP expires and the client must ask for an IP again.

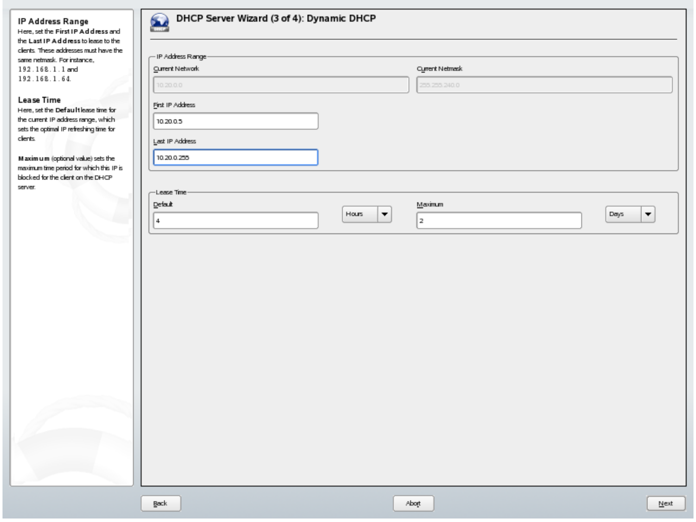
DHCP Server Wizard (2 of 4): Global Settings

Domain Name: example.com
Primary Name Server IP: 10.20.0.2
Secondary Name Server IP:
Default Gateway (Router): 10.20.0.1
NTP Time Server: ntp.example.com
Print Server:
WINS Server:
Default Lease Time: 4 Hours

Dynamic DHCP

In this step, configure how dynamic IP addresses should be assigned to clients. To do so, specify an IP range from which the server can assign addresses to DHCP clients. All these addresses must be covered by the same netmask. Also specify the lease time during which a client may keep its IP address without needing to request an extension of the lease. Optionally, specify the maximum lease time—the period during which the server reserves an IP address for a particular client. See Figure 43.3, "DHCP Server: Dynamic DHCP" (page 632).

Figure 43.3 *DHCP Server: Dynamic DHCP*



IP Address Range
Here, set the **First IP Address** and the **Last IP Address** to lease to the clients. These addresses must have the same netmask. For instance, `192.168.1.1` and `192.168.1.64`.

Lease Time
Here, set the **Default** lease time for the current IP address range, which sets the optimal IP refreshing time for clients.

Maximum (optional value) sets the maximum time period for which this IP is blocked for the client on the DHCP server.

DHCP Server Wizard (3 of 4): Dynamic DHCP

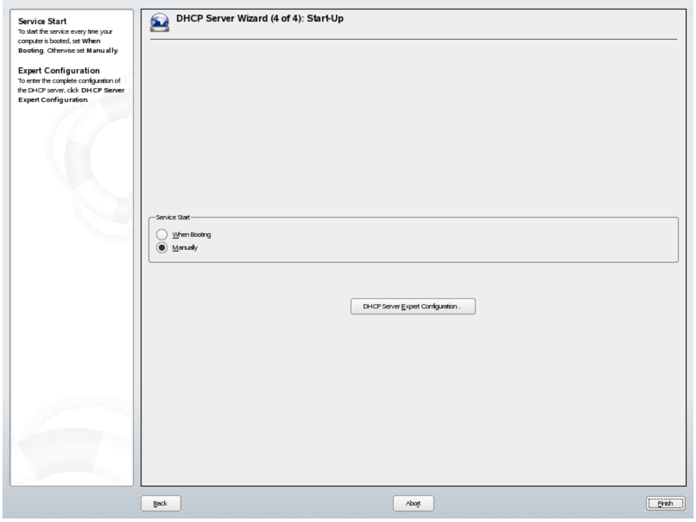
IP Address Range
Current Network: 10.20.0.0
Current Netmask: 255.255.240.0
First IP Address: 10.20.0.5
Last IP Address: 10.20.0.255

Lease Time
Default: 4 Hours
Maximum: 2 Days

Finishing the Configuration and Setting the Start Mode

After the third part of the configuration wizard, a last dialog is shown in which you can define how the DHCP server should be started. Here, specify whether to start the DHCP server automatically when the system is booted or manually when needed (for example, for test purposes). Click *Finish* to complete the configuration of the server. See Figure 43.4, “DHCP Server: Start-Up” (page 633).

Figure 43.4 *DHCP Server: Start-Up*


DHCP Server Wizard (4 of 4): Start-Up
DHCP Server Expert Configuration...
Back
Abort
Finish

Service Start
To start the service every time your computer is booted, set **When Booting**. Otherwise set **Manually**.

Expert Configuration
To enter the complete configuration of the DHCP server, click **DHCP Server Expert Configuration**.

Service Start
○ When Booting
● Manually

43.2 DHCP Software Packages

Both a DHCP server and DHCP clients are available for SUSE Linux. The DHCP server available is dhcpd (published by the Internet Software Consortium). On the client side, choose between two different DHCP client programs: dhclient (also from ISC) and the DHCP client daemon in the `dhcpcd` package.

SUSE Linux installs dhcpcd by default. The program is very easy to handle and is launched automatically on each system boot to watch for a DHCP server. It does not need a configuration file to do its job and works out of the box in most standard setups. For more complex situations, use the ISC dhclient, which is controlled by means of the configuration file `/etc/dhclient.conf`.

43.3 The DHCP Server dhcpd

The core of any DHCP system is the dynamic host configuration protocol daemon. This server *leases* addresses and watches how they are used, according to the settings defined in the configuration file `/etc/dhcpd.conf`. By changing the parameters and values in this file, a system administrator can influence the program's behavior in numerous ways. Look at the basic sample /etc/dhcpd.conf file in Example 43.1, “The Configuration File /etc/dhcpd.conf” (page 634).

Example 43.1 *The Configuration File /etc/dhcpd.conf*

```
default-lease-time 600;         # 10 minutes
max-lease-time 7200;            # 2  hours

option domain-name "cosmos.all";
option domain-name-servers 192.168.1.1, 192.168.1.2;
option broadcast-address 192.168.1.255;
option routers 192.168.1.254;
option subnet-mask 255.255.255.0;

subnet 192.168.1.0 netmask 255.255.255.0
 {
  range 192.168.1.10 192.168.1.20;
  range 192.168.1.100 192.168.1.200;
 }
```

This simple configuration file should be sufficient to get the DHCP server to assign IP addresses in the network. Make sure that a semicolon is inserted at the end of each line, because otherwise dhcpd is not started.

The sample file can be divided into three sections. The first one defines how many seconds an IP address is leased to a requesting client by default (default-lease-time) before it should apply for renewal. The section also includes a statement of the maximum period for which a machine may keep an IP address assigned by the DHCP server without applying for renewal (max-lease-time).

In the second part, some basic network parameters are defined on a global level:

- The line option domain-name defines the default domain of your network.

- With the entry option domain-name-servers, specify up to three values for the DNS servers used to resolve IP addresses into hostnames and vice versa. Ideally, configure a name server on your machine or somewhere else in your network before setting up DHCP. That name server should also define a hostname for each dynamic address and vice versa. To learn how to configure your own name server, read Chapter 40, *The Domain Name System* (page 593).

- The line option broadcast-address defines the broadcast address the requesting client should use.

- With `option routers`, set where the server should send data packets that cannot be delivered to a host on the local network (according to the source and target host address and the subnet mask provided). In most cases, especially in smaller networks, this router is identical to the Internet gateway.

- With `option subnet-mask`, specify the netmask assigned to clients.

The last section of the file defines a network, including a subnet mask. To finish, specify the address range that the DHCP daemon should use to assign IP addresses to interested clients. In this example, clients may be given any address between `192.168.1.10` and `192.168.1.20` as well as `192.168.1.100` and `192.168.1.200`.

After editing these few lines, you should be able to activate the DHCP daemon with the command `rcdhcpd start`. It will be ready for use immediately. Use the command `rcdhcpd check-syntax` to perform a brief syntax check. If you encounter any unexpected problems with your configuration—the server aborts with an error or does not return `done` on start—you should be able to find out what has gone wrong by looking for information either in the main system log `/var/log/messages` or on console 10 (Ctrl + Alt + F10).

On a default SUSE Linux system, the DHCP daemon is started in a chroot environment for security reasons. The configuration files must be copied to the chroot environment so the daemon can find them. Normally, there is no need to worry about this because the command `rcdhcpd start` automatically copies the files.

43.3.1 Clients with Fixed IP Addresses

DHCP can also be used to assign a predefined, static address to a specific client for each request. Addresses assigned explicitly always take priority over dynamic addresses from the pool. A static address never expires in the way a dynamic address would, for example, if there were not enough addresses available so the server needed to redistribute them among clients.

To identify a client configured with a static address, dhcpd uses the hardware address, which is a globally unique, fixed numerical code consisting of six octet pairs for the identification of all network devices (for example, `00:00:45:12:EE:F4`). If the respective lines, like the ones in Example 43.2, “Additions to the Configuration File” (page 636), are added to the configuration file of Example 43.1, “The Configuration

File /etc/dhcpd.conf” (page 634), the DHCP daemon always assigns the same set of data to the corresponding client.

Example 43.2 *Additions to the Configuration File*

```
host earth {
hardware ethernet 00:00:45:12:EE:F4;
fixed-address 192.168.1.21;
}
```

The name of the respective client (`host` *`hostname`*, here `earth`) is entered in the first line and the MAC address in the second line. On Linux hosts, find this address with the command `ifstatus` followed by the network device (for example, `eth0`). If necessary, activate the network card first with `ifup eth0`. The output should contain something like

```
link/ether 00:00:45:12:EE:F4
```

In the preceding example, a client with a network card having the MAC address `00:00:45:12:EE:F4` is assigned the IP address `192.168.1.21` and the hostname earth automatically. The type of hardware to enter is `ethernet` in nearly all cases, although `token-ring`, which is often found on IBM systems, is also supported.

43.3.2 The SUSE Linux Version

To improve security, the SUSE version of the ISC's DHCP server comes with the non-root/chroot patch by Ari Edelkind applied. This enables dhcpd to run with the user ID `nobody` and run in a chroot environment (`/var/lib/dhcp`). To make this possible, the configuration file `dhcpd.conf` must be located in `/var/lib/dhcp/etc`. The init script automatically copies the file to this directory when starting.

Control the server's behavior regarding this feature by means of entries in the file `/etc/sysconfig/dhcpd`. To run dhcpd without the chroot environment, set the variable `DHCPD_RUN_CHROOTED` in `/etc/sysconfig/dhcpd` to “no”.

To enable dhcpd to resolve hostnames even from within the chroot environment, some other configuration files must be copied as well:

- `/etc/localtime`

- `/etc/host.conf`

- /etc/hosts

- /etc/resolv.conf

These files are copied to /var/lib/dhcp/etc/ when starting the init script. Take these copies into account for any changes that they require if they are dynamically modified by scripts like /etc/ppp/ip-up. However, there should be no need to worry about this if the configuration file only specifies IP addresses (instead of hostnames).

If your configuration includes additional files that should be copied into the chroot environment, set these under the variable DHCPD_CONF_INCLUDE_FILES in the file /etc/sysconfig/dhcpd. To ensure that the DHCP logging facility keeps working even after a restart of the syslog daemon, it is necessary to add the option "-a /var/lib/dhcp/dev/log" under SYSLOGD_PARAMS in the file /etc/sysconfig/syslog.

43.4 For More Information

More information about DHCP is available at the Web site of the *Internet Software Consortium* (http://www.isc.org/products/DHCP/). Information is also available in the dhcpd, dhcpd.conf, dhcpd.leases, and dhcp-options man pages.

Time Synchronization with xntp 44

The NTP (network time protocol) mechanism is a protocol for synchronizing the system time over the network. First, a machine can obtain the time from a server that is a reliable time source. Second, a machine can itself act as a time source for other computers in the network. The goal is twofold—maintaining the absolute time and synchronizing the system time of all machines within a network.

Maintaining an exact system time is important in many situations. The built-in hardware (BIOS) clock does often not meet the requirements of applications like databases. Manual correction of the system time would lead to severe problems because, for example, a backward leap can cause malfunction of critical applications. Within a network, it is usually necessary to synchronize the system time of all machines, but manual time adjustment is a bad approach. xntp provides an mechanism to solve these problems. It continuously adjusts the system time with the help of reliable time servers in the network. It further enables the management of local reference clocks, such as radio-controlled clocks.

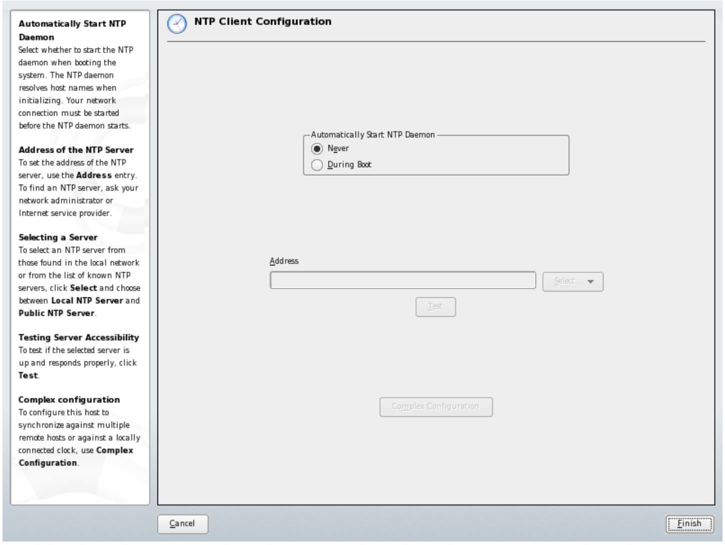
44.1 Configuring an NTP Client with YaST

xntp is preset to use the local computer clock as a time reference. Using the (BIOS) clock, however, only serves as a fallback for the case that no time source of greater precision is available. SUSE Linux facilitates the configuration of an NTP client with YaST. Use the quick or complex configuration for clients that do no run the SuSEfirewall because they are part of a protected intranet. Both are described in the following.

44.1.1 Quick NTP Client Configuration

The easy NTP client configuration (*Network Services → NTP Client*) consists of two dialogs. Set the start mode of xntpd and the server to query in the first dialog. To start xntpd automatically when the system is booted, click *During Boot*. Then click *Select* to access a second dialog in which to select a suitable time server for your network.

Figure 44.1 *YaST: Configuring an NTP Client*



Automatically Start NTP Daemon
Select whether to start the NTP daemon when booting the system. The NTP daemon resolves host names when initializing. Your network connection must be started before the NTP daemon starts.

Address of the NTP Server
To set the address of the NTP server, use the **Address** entry. To find an NTP server, ask your network administrator or Internet service provider.

Selecting a Server
To select an NTP server from those found in the local network or from the list of known NTP servers, click **Select** and choose between **Local NTP Server** and **Public NTP Server**.

Testing Server Accessibility
To test if the selected server is up and responds properly, click **Test**.

Complex configuration
To configure this host to synchronize against multiple remote hosts or against a locally connected clock, use **Complex Configuration**.

NTP Client Configuration

Automatically Start NTP Daemon
● Never
○ During Boot

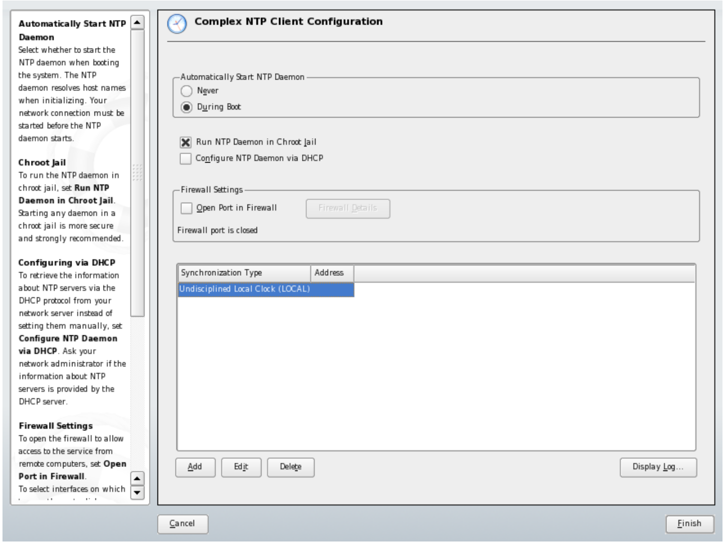
Address

In the detailed server selection dialog, determine whether to implement time synchronization using a time server from your local network (*Local NTP Server*) or an Internet-based time server that takes care of your time zone (*Public NTP Server*). For a local time server, click *Lookup* to start an SLP query for available time servers in your network. Select the most suitable time server from the list of search results and exit the dialog with *OK*. For a public time server, select your country (time zone) and a suitable server from the list under *Public NTP Server* then exit the dialog with *OK*. In the main dialog, test the availability of the selected server with *Test* and quit the dialog with *Finish*.

44.1.2 Complex NTP Client Configuration

The complex configuration of an NTP client can be accessed under *Complex Configuration* from the main dialog of the *NTP Client* module, shown in Figure 44.1, "YaST: Configuring an NTP Client" (page 640), after selecting the start-up mode as described in the quick configuration.

Figure 44.2 *YaST: Complex NTP Client Configuration*



Automatically Start NTP Daemon
Select whether to start the NTP daemon when booting the system. The NTP daemon resolves host names when initializing. Your network connection must be started before the NTP daemon starts.

Chroot Jail
To run the NTP daemon in chroot jail, set **Run NTP Daemon in Chroot Jail**. Starting any daemon in a chroot jail is more secure and strongly recommended.

Configuring via DHCP
To retrieve the information about NTP servers via the DHCP protocol from your network server instead of setting them manually, set **Configure NTP Daemon via DHCP**. Ask your network administrator if the information about NTP servers is provided by the DHCP server.

Firewall Settings
To open the firewall to allow access to the service from remote computers, set **Open Port in Firewall**.
To select interfaces on which

Complex NTP Client Configuration

Automatically Start NTP Daemon
○ Never
● During Boot

☒ Run NTP Daemon in Chroot Jail
☐ Configure NTP Daemon via DHCP

Firewall Settings
☐ Open Port in Firewall
Firewall port is closed

Synchronization Type	Address
Undisciplined Local Clock (LOCAL)	

In *Complex NTP Client Configuration*, determine whether xntpd should be started in a chroot jail. This increases the security in the event of an attack over xntpd, because it prevents the attacker from compromising the entire system. *Configure NTP Daemon via DHCP* sets up the NTP client to get a list of the NTP servers available in your network via DHCP.

The servers and other time sources for the client to query are listed in the lower part. Modify this list as needed with *Add*, *Edit*, and *Delete*. *Display Log* provides the possibility to view the log files of your client.

Click *Add* to add a new source of time information. In the following dialog, select the type of source with which the time synchronization should be made. The following options are available:

Server

Another dialog enables you to select an NTP server (as described in Section 44.1.1, “Quick NTP Client Configuration” (page 640)). Activate *Use for Initial Synchronization* to trigger the synchronization of the time information between the server and the client when the system is booted. An input field allows you to specify additional options for xntpd. Refer to `/usr/share/doc/packages/xntp-doc` (part of the `xntp-doc` package) for detailed information.

Peer

A peer is a machine to which a symmetric relationship is established: it acts both as a time server and as a client. To use a peer in the same network instead of a server, enter the address of the system. The rest of the dialog is identical to the *Server* dialog.

Radio Clock

To use a radio clock in your system for the time synchronization, enter the clock type, unit number, device name, and other options in this dialog. Click *Driver Calibration* to fine-tune the driver. Detailed information about the operation of a local radio clock is available in `/usr/share/doc/packages/xntp-doc/html/refclock.htm`.

Outgoing Broadcast

Time information and queries can also be transmitted by broadcast in the network. In this dialog, enter the address to which such broadcasts should be sent. Do not activate broadcasting unless you have a reliable time source like a radio controlled clock.

Incoming Broadcast

If you want your client to receive its information via broadcast, enter the address from which the respective packets should be accepted in this fields.

44.2 Configuring xntp in the Network

The easiest way to use a time server in the network is to set server parameters. For example, if a time server called `ntp.example.com` is reachable from the network, add its name to the file `/etc/ntp.conf` by adding the line `server` ntp.example.com. To add more time servers, insert additional lines with the keyword server. After initializing xntpd with the command rcxntpd start, it takes about one hour until the time is stabilized and the drift file for correcting the local computer clock is created. With the drift file, the systematic error of the hardware clock can be computed as soon as the computer is powered on. The correction is used immediately, resulting in a higher stability of the system time.

There are two possible ways to use the NTP mechanism as a client: First, the client can query the time from a known server in regular intervals. With many clients, this approach can cause a high load on the server. Second, the client can wait for NTP broadcasts sent out by broadcast time servers in the network. This approach has the disadvantage that the quality of the server is unknown and a server sending out wrong information can cause severe problems.

If the time is obtained via broadcast, you do not need the server name. In this case, enter the line broadcastclient in the configuration file /etc/ntp.conf. To use one or more known time servers exclusively, enter their names in the line starting with servers.

44.3 Setting Up a Local Reference Clock

The software package xntp contains drivers for connecting local reference clocks. A list of supported clocks is available in the xntp-doc package in the file /usr/share/doc/packages/xntp-doc/html/refclock.htm. Every driver is associated with a number. In xntp, the actual configuration takes place by means of pseudo IPs. The clocks are entered in the file /etc/ntp.conf as though they existed in the network. For this purpose, they are assigned special IP addresses in the form 127.127.t.u. Here, t stands for the type of the clock and determines which driver is used and u for unit, which determines the interface used.

Normally, the individual drivers have special parameters that describe configuration details. The file /usr/share/doc/packages/xntp-doc/html/driverNN.htm (where NN is the number of the driver) provides information about the particular type of clock. For example, the “type 8” clock (radio clock over serial interface) requires an additional mode that specifies the clock more precisely. The Conrad DCF77 receiver module, for example, has mode 5. To use this clock as a preferred reference, specify the keyword `prefer`. The complete `server` line for a Conrad DCF77 receiver module would be:

```
server 127.127.8.0 mode 5 prefer
```

Other clocks follow the same pattern. Following the installation of the `xntp-doc` package, the documentation for xntp is available in the directory `/usr/share/doc/packages/xntp-doc/html`. The file `/usr/share/doc/packages/xntp-doc/html/refclock.htm` provides links to the driver pages describing the driver parameters.

45 LDAP—A Directory Service

The Lightweight Directory Access Protocol (LDAP) is a set of protocols designed to access and maintain information directories. LDAP can be used for numerous purposes, like user and group management, system configuration management, or address management. This chapter provides a basic understanding of how OpenLDAP works and how to manage LDAP data with YaST. While there are several implementations of the LDAP protocol, this chapter focuses entirely on the OpenLDAP implementation.

It is crucial within a networked environment to keep important information structured and quickly available. This can be done with a directory service that, like the common yellow pages, keeps information available in a well-structured, quickly searchable form.

In the ideal case, a central server keeps the data in a directory and distributes it to all clients using a certain protocol. The data is structured in a way that allows a wide range of applications to access it. That way, it is not necessary for every single calendar tool and e-mail client to keep its own database—a central repository can be accessed instead. This notably reduces the administration effort for the information. The use of an open and standardized protocol like LDAP ensures that as many different client applications as possible can access such information.

A directory in this context is a type of database optimized for quick and effective reading and searching:

- To make numerous (concurrent) reading accesses possible, write access is limited to a small number of updates by the administrator. Conventional databases are optimized for accepting the largest possible data volume in a short time.

- Because write accesses can only be executed in a restricted fashion, a directory service is employed for administering mostly unchanging, static information. Data in a conventional database typically changes very often (*dynamic* data). Phone numbers in a company directory do not change nearly as often as, for example, the figures administered in accounting.

- When static data is administered, updates of the existing data sets are very rare. When working with dynamic data, especially when data sets like bank accounts or accounting are concerned, the consistency of the data is of primary importance. If an amount should be subtracted from one place to be added to another, both operations must happen concurrently, within a *transaction*, to ensure balance over the data stock. Databases support such transactions. Directories do not. Short-term inconsistencies of the data are quite acceptable in directories.

The design of a directory service like LDAP is not laid out to support complex update or query mechanisms. All applications accessing this service should gain access quickly and easily.

Many directory services have previously existed and still exist both in Unix and outside it. Novell NDS, Microsoft ADS, Banyan's Street Talk, and the OSI standard X.500 are just a few examples. LDAP was originally planned as a lean flavor of DAP, the directory access protocol, which was developed for accessing X.500. The X.500 standard regulates the hierarchical organization of directory entries.

LDAP is a trimmed down version of DAP. Without losing the X.500 entry hierarchy, profit from LDAP's cross-platform capabilities and save resources. The use of TCP/IP makes it substantially easier to establish interfaces between a docking application and the LDAP service.

LDAP, meanwhile, has evolved and is increasingly employed as a stand-alone solution without X.500 support. LDAP supports *referrals* with LDAPv3 (the protocol version in package `openldap2`), making it possible to have distributed databases. The usage of SASL (simple authentication and security layer) is also new.

LDAP is not limited to querying data from X.500 servers, as it was originally planned. There is an open source server slapd, which can store object information in a local database. There is also an extension called slurpd, which is responsible for replicating multiple LDAP servers.

The `openldap2` package consists of:

slapd

A stand-alone LDAPv3 server that administers object information in a BerkeleyDB-based database.

slurpd

This program enables the replication of modifications to data on the local LDAP server to other LDAP servers installed on the network.

additional tools for system maintenance

`slapcat`, `slapadd`, `slapindex`

45.1 LDAP versus NIS

The Unix system administrator traditionally uses the NIS service for name resolution and data distribution in a network. The configuration data contained in the files in `/etc` and the directories `group`, `hosts`, `mail`, `netgroup`, `networks`, `passwd`, `printcap`, `protocols`, `rpc`, and `services` are distributed by clients all over the network. These files can be maintained without major effort because they are simple text files. The handling of larger amounts of data, however, becomes increasingly difficult due to nonexistent structuring. NIS is only designed for Unix platforms. So, it isn't suitable as a centralized data administration tool in heterogenous networks.

Unlike NIS, the LDAP service is not restricted to pure Unix networks. Windows servers (from 2000) support LDAP as a directory service. Novell also offers an LDAP service. Application tasks mentioned above are additionally supported in non-Unix systems.

The LDAP principle can be applied to any data structure that should be centrally administered. A few application examples are:

- Employment as a replacement for the NIS service
- Mail routing (postfix, sendmail)
- Address books for mail clients, like Mozilla, Evolution, and Outlook
- Administration of zone descriptions for a BIND9 name server
- User authentication with Samba in heterogeneous networks

This list can be extended because LDAP is extensible, unlike NIS. The clearly-defined hierarchical structure of the data eases the administration of large amounts of data, because it can be searched better.

45.2 Structure of an LDAP Directory Tree

An LDAP directory has a tree structure. All entries (called objects) of the directory have a defined position within this hierarchy. This hierarchy is called the *directory information tree* (DIT). The complete path to the desired entry, which unambiguously identifies it, is called *distinguished name* or DN. A single node along the path to this entry is called *relative distinguished name* or RDN. Objects can generally be assigned to one of two possible types:

container

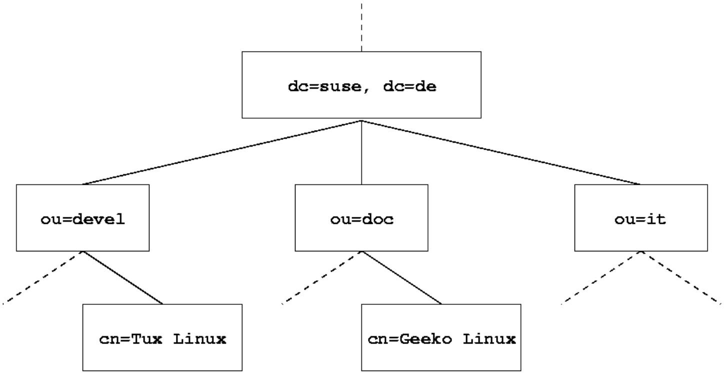
These objects can themselves contain other objects. Such object classes are `root` (the root element of the directory tree, which does not really exist), `c` (country), `ou` (organizational unit), and `dc` (domain component). This model is comparable to the directories (folders) in a file system.

leaf

These objects sit at the end of a branch and have no subordinate objects. Examples are `person`, `InetOrgPerson`, or `groupofNames`.

The top of the directory hierarchy has a root element `root`. This can contain `c` (country), `dc` (domain component), or `o` (organization) as subordinate elements. The relations within an LDAP directory tree become more evident in the following example, shown in Figure 45.1, “Structure of an LDAP Directory” (page 649).

Figure 45.1 *Structure of an LDAP Directory*



The complete diagram comprises a fictional directory information tree. The entries on three levels are depicted. Each entry corresponds to one box in the picture. The complete, valid *distinguished name* for the fictional SUSE employee `Geeko Linux`, in this case, is `cn=Geeko Linux,ou=doc,dc=suse,dc=de`. It is composed by adding the RDN `cn=Geeko Linux` to the DN of the preceding entry `ou=doc,dc=suse,dc=de`.

The global determination of which types of objects should be stored in the DIT is done following a *scheme*. The type of an object is determined by the *object class*. The object class determines what attributes the concerned object must or can be assigned. A scheme, therefore, must contain definitions of all object classes and attributes used in the desired application scenario. There are a few common schemes (see RFC 2252 and 2256). It is, however, possible to create custom schemes or to use multiple schemes complementing each other if this is required by the environment in which the LDAP server should operate.

Table 45.1, “Commonly Used Object Classes and Attributes” (page 650) offers a small overview of the object classes from `core.schema` and `inetorgperson.schema` used in the example, including required attributes and valid attribute values.

Table 45.1 Commonly Used Object Classes and Attributes

Object Class	Meaning	Example Entry	Compulsory Attributes
dcObject	*domainComponent* (name components of the domain)	suse	dc
organizationalUnit	*organizationalUnit* (organizational unit)	doc	ou
inetOrgPerson	*inetOrgPerson* (person-related data for the intranet or Internet)	Geeko Linux	sn and cn

Example 45.1, “Excerpt from schema.core ” (page 650) shows an excerpt from a scheme directive with explanations (line numbering for explanatory reasons).

Example 45.1 *Excerpt from schema.core*

```
#1 attributetype (2.5.4.11 NAME ( 'ou' 'organizationalUnitName')
#2        DESC 'RFC2256: organizational unit this object belongs to'
#3        SUP name )

...
#4 objectclass ( 2.5.6.5 NAME 'organizationalUnit'
#5        DESC 'RFC2256: an organizational unit'
#6        SUP top STRUCTURAL
#7        MUST ou
#8 MAY (userPassword $ searchGuide $ seeAlso $ businessCategory
    $ x121Address $ registeredAddress $ destinationIndicator
    $ preferredDeliveryMethod $ telexNumber
    $ teletexTerminalIdentifier $ telephoneNumber
    $ internationaliSDNNumber $ facsimileTelephoneNumber
    $ street $ postOfficeBox $ postalCode $ postalAddress
    $ physicalDeliveryOfficeName
    $ st $ l $ description) )
...
```

The attribute type `organizationalUnitName` and the corresponding object class `organizationalUnit` serve as an example here. Line 1 features the name of the attribute, its unique OID (*object identifier*) (numerical), and the abbreviation of the attribute.

Line 2 gives brief description of the attribute with DESC. The corresponding RFC on which the definition is based is also mentioned here. SUP in line 3 indicates a superordinate attribute type to which this attribute belongs.

The definition of the object class organizationalUnit begins in line 4, like in the definition of the attribute, with an OID and the name of the object class. Line 5 features a brief description of the object class. Line 6, with its entry SUP top, indicates that this object class is not subordinate to another object class. Line 7, starting with MUST, lists all attribute types that *must* be used in conjunction with an object of the type organizationalUnit. Line 8, starting with MAY, lists all attribute types that are permitted in conjunction with this object class.

A very good introduction to the use of schemes can be found in the documentation of OpenLDAP. When installed, find it in /usr/share/doc/packages/openldap2/admin-guide/index.html.

45.3 Server Configuration with slapd.conf

Your installed system contains a complete configuration file for your LDAP server at /etc/openldap/slapd.conf. The single entries are briefly described here and necessary adjustments are explained. Entries prefixed with a hash (#) are inactive. This comment character must be removed to activate them.

45.3.1 Global Directives in slapd.conf

Example 45.2 *slapd.conf: Include Directive for Schemes*

```
include         /etc/openldap/schema/core.schema
include         /etc/openldap/schema/cosine.schema
include         /etc/openldap/schema/inetorgperson.schema
include         /etc/openldap/schema/rfc2307bis.schema
include         /etc/openldap/schema/yast.schema
```

This first directive in slapd.conf, shown in Example 45.2, “slapd.conf: Include Directive for Schemes” (page 651), specifies the scheme by which the LDAP directory is organized. The entry core.schema is compulsory. Additionally required schemes are appended to this directive. Information can be found in the included OpenLDAP documentation.

Example 45.3 slapd.conf: pidfile and argsfile

```
pidfile /var/run/slapd/slapd.pid
argsfile /var/run/slapd/slapd.args
```

These two files contain the PID (process ID) and some of the arguments with which the `slapd` process is started. There is no need for modifications here.

Example 45.4 slapd.conf: Access Control

```
# Sample Access Control
#       Allow read access of root DSE
# Allow self write access
#       Allow authenticated users read access
#       Allow anonymous users to authenticate
# access to dn="" by * read
  access to * by self write
              by users read
              by anonymous auth
#
# if no access controls are present, the default is:
#       Allow read by all
#
# rootdn can always write!
```

Example 45.4, "slapd.conf: Access Control" (page 652) is the excerpt from `slapd.conf` that regulates the access permissions for the LDAP directory on the server. The settings made here in the global section of `slapd.conf` are valid as long as no custom access rules are declared in the database-specific section. These would overwrite the global declarations. As presented here, all users have read access to the directory, but only the administrator (`rootdn`) can write to this directory. Access control regulation in LDAP is a highly complex process. The following tips can help:

- Every access rule has the following structure:

  ```
  access to <what> by <who> <access>
  ```

- *`what`* is a placeholder for the object or attribute to which access is granted. Individual directory branches can be protected explicitly with separate rules. It is also possible to process regions of the directory tree with one rule by using regular expressions. `slapd` evaluates all rules in the order in which they are listed in the configuration file. More general rules should be listed after more specific ones—the first rule `slapd` regards as valid is evaluated and all following entries are ignored.

- *who* determines who should be granted access to the areas determined with *what*. Regular expressions may be used. slapd again aborts the evaluation of who after the first match, so more specific rules should be listed before the more general ones. The entries shown in Table 45.2, "User Groups and Their Access Grants" (page 653) are possible.

Table 45.2 *User Groups and Their Access Grants*

Tag	Scope
*	All users without exception
anonymous	Not authenticated ("anonymous") users
users	Authenticated users
self	Users connected with the target object
dn.regex=<regex>	All users matching the regular expression

- *access* specifies the type of access. Use the options listed in Table 45.3, "Types of Access" (page 653).

Table 45.3 *Types of Access*

Tag	Scope of Access
none	No access
auth	For contacting the server
compare	To objects for comparison access
search	For the employment of search filters
read	Read access
write	Write access

slapd compares the access right requested by the client with those granted in slapd.conf. The client is granted access if the rules allow a higher or equal right than the requested one. If the client requests higher rights than those declared in the rules, it is denied access.

Example 45.5, “slapd.conf: Example for Access Control” (page 654) shows an example of a simple access control that can be arbitrarily developed using regular expressions.

Example 45.5 *slapd.conf: Example for Access Control*

```
access to  dn.regex="ou=([^,]+),dc=suse,dc=de"
by dn.regex="cn=administrator,ou=$1,dc=suse,dc=de" write
by user read
by * none
```

This rule declares that only its respective administrator has write access to an individual ou entry. All other authenticated users have read access and the rest of the world has no access.

TIP: Establishing Access Rules

If there is no access to rule or no matching by directive, access is denied. Only explicitly declared access rights are granted. If no rules are declared at all, the default principle is write access for the administrator and read access for the rest of the world.

Find detailed information and an example configuration for LDAP access rights in the online documentation of the installed openldap2 package.

Apart from the possibility to administer access permissions with the central server configuration file (slapd.conf), there is access control information (ACI). ACI allows storage of the access information for individual objects within the LDAP tree. This type of access control is not yet common and is still considered experimental by the developers. Refer to http://www.openldap.org/faq/data/cache/758.html for information.

45.3.2 Database-Specific Directives in slapd.conf

Example 45.6 *slapd.conf: Database-Specific Directives*

```
database bdb
checkpoint      1024    5
cachesize       10000
suffix "dc=suse,dc=de"
rootdn "cn=admin,dc=suse,dc=de"
# Cleartext passwords, especially for the rootdn, should
# be avoided.  See slappasswd(8) and slapd.conf(5) for details.
# Use of strong authentication encouraged.
rootpw secret
# The database directory MUST exist prior to running slapd AND
# should only be accessible by the slapd/tools. Mode 700 recommended.
directory /var/lib/ldap
# Indices to maintain
index   objectClass     eq
```

The type of database, a Berkeley database in this case, is determined in the first line of this section (see Example 45.6, “slapd.conf: Database-Specific Directives” (page 655)). `checkpoint` determines the amount of data (in kb) that is kept in the transaction log before it is written to the actual database and the time (in minutes) between two write actions. `cachesize` sets the number of objects kept in the database's cache. `suffix` determines for which portion of the LDAP tree this server should be responsible. The following `rootdn` determines who owns administrator rights to this server. The user declared here does not need to have an LDAP entry or exist as regular user. The administrator password is set with `rootpw`. Instead of using `secret` here, it is possible to enter the hash of the administrator password created by `slappasswd`. The `directory` directive indicates the directory (in the file system) where the database directories are stored on the server. The last directive, `index objectClass eq`, results in the maintenance of an index of all object classes. Attributes for which users search most often can be added here according to experience. Custom `Access` rules defined here for the database are used instead of the global `Access` rules.

45.3.3 Starting and Stopping the Servers

Once the LDAP server is fully configured and all desired entries have been made according to the pattern described in Section 45.4, “Data Handling in the LDAP Directory” (page 656), start the LDAP server as `root` by entering `rcldap start`. To stop the server manually, enter the command `rcldap stop`. Request the status of the running LDAP server with `rcldap status`.

The YaST runlevel editor, described in Section 28.2.3, “Configuring System Services (Runlevel) with YaST” (page 422), can be used to have the server started and stopped automatically on boot and halt of the system. It is also possible to create the corresponding links to the start and stop scripts with the `insserv` command from a command prompt as described in Section 28.2.2, “Init Scripts” (page 418).

45.4 Data Handling in the LDAP Directory

OpenLDAP offers a series of tools for the administration of data in the LDAP directory. The four most important tools for adding to, deleting from, searching through, and modifying the data stock are briefly explained below.

45.4.1 Inserting Data into an LDAP Directory

Once the configuration of your LDAP server in `/etc/openldap/lsapd.conf` is correct and ready to go (it features appropriate entries for `suffix`, `directory`, `rootdn`, `rootpw`, and `index`), proceed to entering records. OpenLDAP offers the `ldapadd` command for this task. If possible, add the objects to the database in bundles for practical reasons. LDAP is able to process the LDIF format (LDAP data interchange format) for this. An LDIF file is a simple text file that can contain an arbitrary number of pairs of attribute and value. Refer to the schema files declared in `slapd.conf` for the available object classes and attributes. The LDIF file for creating a rough framework for the example in Figure 45.1, “Structure of an LDAP Directory” (page 649) would look like that in Example 45.7, “Example for an LDIF File” (page 657).

Example 45.7 *Example for an LDIF File*

```
# The SUSE Organization
dn: dc=suse,dc=de
objectClass: dcObject
objectClass: organization
o: SUSE AG dc: suse

# The organizational unit development (devel)
dn: ou=devel,dc=suse,dc=de
objectClass: organizationalUnit
ou: devel

# The organizational unit documentation (doc)
dn: ou=doc,dc=suse,dc=de
objectClass: organizationalUnit
ou: doc

# The organizational unit internal IT (it)
dn: ou=it,dc=suse,dc=de
objectClass: organizationalUnit
ou: it
```

IMPORTANT: Encoding of LDIF Files

LDAP works with UTF-8 (Unicode). Umlauts must be encoded correctly. Use an editor that supports UTF-8, such as Kate or recent versions of Emacs. Otherwise, avoid umlauts and other special characters or use `recode` to recode the input to UTF-8.

Save the file with the `.ldif` suffix then pass it to the server with the following command:

```
ldapadd -x -D <dn of the administrator> -W -f <file>.ldif
```

`-x` switches off the authentication with SASL in this case. `-D` declares the user that calls the operation. The valid DN of the administrator is entered here just like it has been configured in `slapd.conf`. In the current example, this is `cn=admin,dc=suse,dc=de`. `-W` circumvents entering the password on the command line (in clear text) and activates a separate password prompt. This password was previously determined in `slapd.conf` with `rootpw`. `-f` passes the filename. See the details of running `ldapadd` in Example 45.8, “ldapadd with example.ldif” (page 658).

Example 45.8 *ldapadd with example.ldif*

```
ldapadd -x -D cn=admin,dc=suse,dc=de -W -f example.ldif

Enter LDAP password:
adding new entry "dc=suse,dc=de"
adding new entry "ou=devel,dc=suse,dc=de"
adding new entry "ou=doc,dc=suse,dc=de"
adding new entry "ou=it,dc=suse,dc=de"
```

The user data of individuals can be prepared in separate LDIF files. Example 45.9, “LDIF Data for Tux” (page 658) adds `Tux` to the new LDAP directory.

Example 45.9 *LDIF Data for Tux*

```
# coworker Tux
dn: cn=Tux Linux,ou=devel,dc=suse,dc=de
objectClass: inetOrgPerson
cn: Tux Linux
givenName: Tux
sn: Linux
mail: tux@suse.de
uid: tux
telephoneNumber: +49 1234 567-8
```

An LDIF file can contain an arbitrary number of objects. It is possible to pass entire directory branches to the server at once or only parts of it as shown in the example of individual objects. If it is necessary to modify some data relatively often, a fine subdivision of single objects is recommended.

45.4.2 Modifying Data in the LDAP Directory

The tool `ldapmodify` is provided for modifying the data stock. The easiest way to do this is to modify the corresponding LDIF file then pass this modified file to the LDAP server. To change the telephone number of colleague Tux from `+49 1234 567-8` to `+49 1234 567-10`, edit the LDIF file like in Example 45.10, “Modified LDIF File tux.ldif” (page 659).

Example 45.10 *Modified LDIF File tux.ldif*

```
# coworker Tux
dn: cn=Tux Linux,ou=devel,dc=suse,dc=de
changetype: modify
replace: telephoneNumber
telephoneNumber: +49 1234 567-10
```

Import the modified file into the LDAP directory with the following command:

```
ldapmodify -x -D cn=admin,dc=suse,dc=de -W -f tux.ldif
```

Alternatively, pass the attributes to change directly to `ldapmodify`. The procedure for this is described below:

1. Start `ldapmodify` and enter your password:

   ```
   ldapmodify -x -D cn=admin,dc=suse,dc=de -W
   Enter LDAP password:
   ```

2. Enter the changes while carefully complying with the syntax in the order presented below:

   ```
   dn: cn=Tux Linux,ou=devel,dc=suse,dc=de
   changetype: modify
   replace: telephoneNumber
   telephoneNumber: +49 1234 567-10
   ```

Find detailed information about `ldapmodify` and its syntax in the ldapmodify(1) man page.

45.4.3 Searching or Reading Data from an LDAP Directory

OpenLDAP provides, with `ldapsearch`, a command line tool for searching data within an LDAP directory and reading data from it. A simple query would have the following syntax:

```
ldapsearch -x -b dc=suse,dc=de "(objectClass=*)"
```

The option `-b` determines the search base—the section of the tree within which the search should be performed. In the current case, this is `dc=suse,dc=de`. To perform a more finely-grained search in specific subsections of the LDAP directory (for example, only within the devel department), pass this section to ldapsearch with -b. -x requests activation of simple authentication. (objectClass=*) declares that all objects contained in the directory should be read. This command option can be used after the creation of a new directory tree to verify that all entries have been recorded correctly and the server responds as desired. More information about the use of ldapsearch can be found in the corresponding man page (ldapsearch(1)).

45.4.4 Deleting Data from an LDAP Directory

Delete unwanted entries with ldapdelete. The syntax is similar to that of the commands described above. To delete, for example, the complete entry for Tux Linux, issue the following command:

```
ldapdelete -x -D cn=admin,dc=suse,dc=de -W cn=Tux \
Linux,ou=devel,dc=suse,dc=de
```

45.5 The YaST LDAP Client

YaST includes a module to set up LDAP-based user management. If you did not enable this feature during the installation, start the module by selecting *Network Services → LDAP Client*. YaST automatically enables any PAM and NSS related changes as required by LDAP (described below) and installs the necessary files.

45.5.1 Standard Procedure

Background knowledge of the processes acting in the background of a client machine helps you understand how the YaST LDAP client module works. If LDAP is activated for network authentication or the YaST module is called, the packages pam_ldap and nss_ldap are installed and the two corresponding configuration files are adapted. pam_ldap is the PAM module responsible for negotiation between login processes and the LDAP directory as the source of authentication data. The dedicated module pam_ldap.so is installed and the PAM configuration is adapted (see Example 45.11, “pam_unix2.conf Adapted to LDAP” (page 661)).

Example 45.11 *pam_unix2.conf Adapted to LDAP*

```
auth:        use_ldap
account:     use_ldap
password:    use_ldap
session:     none
```

When manually configuring additional services to use LDAP, include the PAM LDAP module in the PAM configuration file corresponding to the service in `/etc/pam.d`. Configuration files already adapted to individual services can be found in `/usr/share/doc/packages/pam_ldap/pam.d/`. Copy appropriate files to `/etc/pam.d`.

`glibc` name resolution through the `nsswitch` mechanism is adapted to the employment of LDAP with `nss_ldap`. A new, adapted file `nsswitch.conf` is created in `/etc/` with the installation of this package. More about the workings of `nsswitch.conf` can be found in Section 38.5.1, "Configuration Files" (page 577). The following lines must be present in `nsswitch.conf` for user administration and authentication with LDAP. See Example 45.12, "Adaptations in nsswitch.conf" (page 661).

Example 45.12 *Adaptations in nsswitch.conf*

```
passwd: compat
group: compat

passwd_compat: ldap
group_compat: ldap
```

These lines order the resolver library of `glibc` first to evaluate the corresponding files in `/etc` and additionally access the LDAP server as sources for authentication and user data. Test this mechanism, for example, by reading the content of the user database with the command `getent passwd`. The returned set should contain a survey of the local users of your system as well as all users stored on the LDAP server.

To prevent regular users managed through LDAP from logging in to the server with `ssh` or `login`, the files `/etc/passwd` and `/etc/group` each need to include an additional line. This is the line `+::::::/sbin/nologin` in `/etc/passwd` and `+:::` in `/etc/group`.

45.5.2 Configuration of the LDAP Client

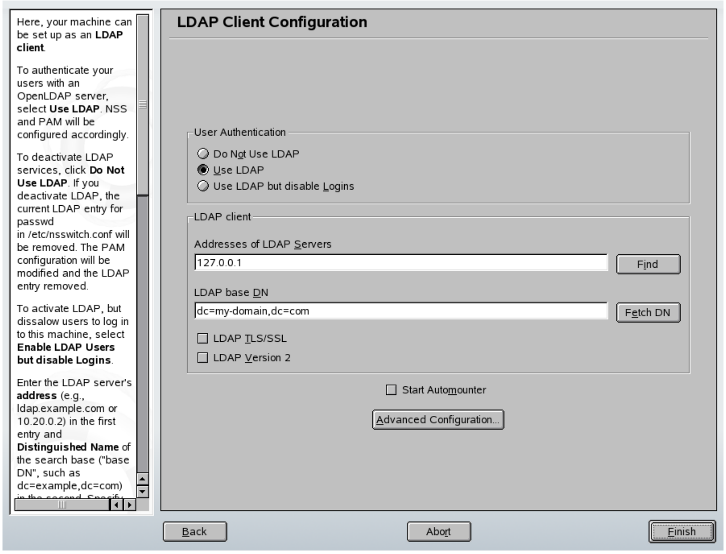
After the initial adjustments of `nss_ldap`, `pam_ldap`, `/etc/passwd`, and `/etc/group` have been taken care of by YaST, you can simply connect your client to the server and let YaST do user management via LDAP. This basic setup is described in Section "Basic Configuration" (page 662).

Use the YaST LDAP client to further configure the YaST group and user configuration modules. This includes manipulating the default settings for new users and groups and the number and nature of the attributes assigned to a user or a group. LDAP user management allows you to assign far more and different attributes to users and groups than traditional user or group management solutions. This is described in Section "Configuring the YaST Group and User Administration Modules" (page 665).

Basic Configuration

The basic LDAP client configuration dialog (Figure 45.2, "YaST: Configuration of the LDAP Client" (page 663)) opens during installation if you choose LDAP user management or when you select *Network Services* → *LDAP Client* in the YaST Control Center in the installed system.

Figure 45.2 *YaST: Configuration of the LDAP Client*



Here, your machine can be set up as an **LDAP client**.

To authenticate your users with an OpenLDAP server, select **Use LDAP**. NSS and PAM will be configured accordingly.

To deactivate LDAP services, click **Do Not Use LDAP**. If you deactivate LDAP, the current LDAP entry for passwd in /etc/nsswitch.conf will be removed. The PAM configuration will be modified and the LDAP entry removed.

To activate LDAP, but dissalow users to log in to this machine, select **Enable LDAP Users but disable Logins**.

Enter the LDAP server's **address** (e.g., ldap.example.com or 10.20.0.2) in the first entry and **Distinguished Name** of the search base ("base DN", such as dc=example,dc=com) in the second. Specify

LDAP Client Configuration

User Authentication

○ Do Not Use LDAP
● Use LDAP
○ Use LDAP but disable Logins

LDAP client

Addresses of LDAP Servers: 127.0.0.1

LDAP base DN: dc=my-domain,dc=com

□ LDAP TLS/SSL
□ LDAP Version 2

□ Start Automounter

To authenticate users of your machine against an OpenLDAP server and enable user management via OpenLDAP, proceed as follows:

1. Click *Use LDAP* to enable the use of LDAP. Select *Use LDAP but Disable Logins* instead if you want to use LDAP for authentication, but do not want other users to log in to this client.

2. Enter the IP address of the LDAP server to use.

3. Enter the *LDAP base DN* to select the search base on the LDAP server.

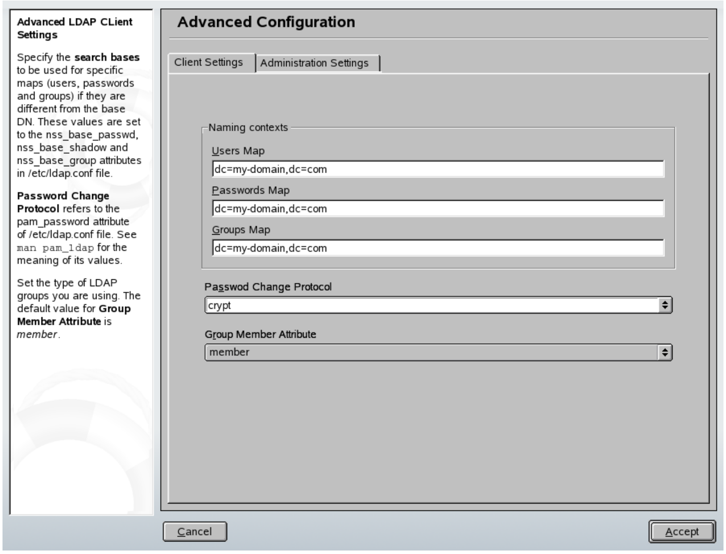
4. If TLS or SSL protected communication with the server is required, select *LDAP TLS/SSL*.

5. If the LDAP server still uses LDAPv2, explicitly enable the use of this protocol version by selecting *LDAP Version 2*.

6. Select *Start Automounter* to mount remote directories on your client, such as a remotely managed `/home`.

7. Click *Finish* to apply your settings.

Figure 45.3 *YaST: Advanced Configuration*



Advanced LDAP CLient Settings

Specify the **search bases** to be used for specific maps (users, passwords and groups) if they are different from the base DN. These values are set to the nss_base_passwd, nss_base_shadow and nss_base_group attributes in /etc/ldap.conf file.

Password Change Protocol refers to the pam_password attribute of /etc/ldap.conf file. See `man pam_ldap` for the meaning of its values.

Set the type of LDAP groups you are using. The default value for **Group Member Attribute** is *member*.

Naming contexts

Users Map: dc=my-domain,dc=com

Passwords Map: dc=my-domain,dc=com

Groups Map: dc=my-domain,dc=com

Paswod Change Protocol: crypt

Group Member Attribute: member

To modify data on the server as administrator, click *Advanced Configuration*. The following dialog is split in two tabs. See Figure 45.3, "YaST: Advanced Configuration" (page 664):

1 In the *Client Settings* tab, adjust the following settings to your needs:

a If the search base for users, passwords, and groups differs from the global search base specified the *LDAP base DN*, enter these different naming contexts in *User Map*, *Password Map*, and *Group Map*.

b Specify the password change protocol. The standard method to use whenever a password is changed is `crypt`, meaning that password hashes generated by `crypt` are used. For details on this and other options, refer to the `pam_ldap` man page.

c Specify the LDAP group to use with *Group Member Attribute*. The default value for this is `member`.

2 In *Administration Settings*, adjust the following settings:

a Set the base for storing your user management data via *Configuration Base DN*.

b Enter the appropriate value for *Administrator DN*. This DN must be identical with the `rootdn` value specified in `/etc/openldap/slapd.conf` to enable this particular user to manipulate data stored on the LDAP server.

c Check *Create Default Configuration Objects* to create the basic configuration objects on the server to enable user management via LDAP.

d If your client machine should act as a file server for home directories across your network, check *Home Directories on This Machine*.

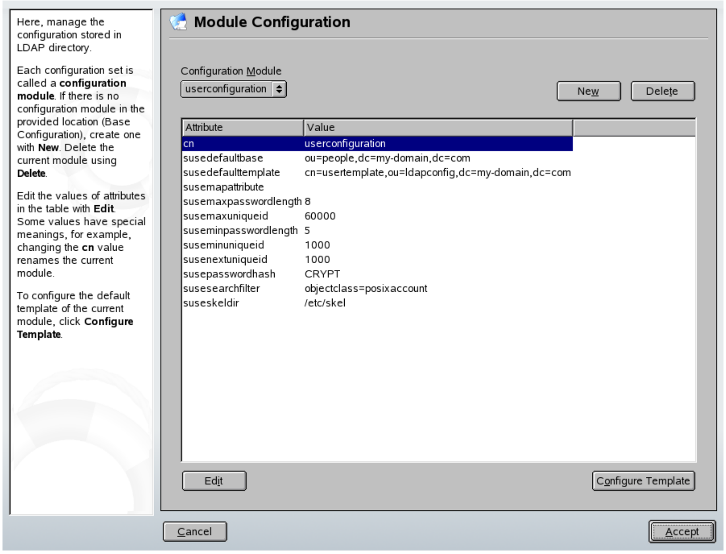
e Click *Accept* to leave the *Advanced Configuration* then *Finish* to apply your settings.

Use *Configure User Management Settings* to edit entries on the LDAP server. Access to the configuration modules on the server is then granted according to the ACLs and ACIs stored on the server. Follow the procedures outlined in Section "Configuring the YaST Group and User Administration Modules" (page 665).

Configuring the YaST Group and User Administration Modules

Use the YaST LDAP client to adapt the YaST modules for user and group administration and to extend them as needed. Define templates with default values for the individual attributes to simplify the data registration. The presets created here are stored as LDAP objects in the LDAP directory. The registration of user data is still done with the regular YaST modules for user and group management. The registered data is stored as LDAP objects on the server.

Figure 45.4 *YaST: Module Configuration*



Here, manage the configuration stored in LDAP directory.

Each configuration set is called a **configuration module**. If there is no configuration module in the provided location (Base Configuration), create one with **New**. Delete the current module using **Delete**.

Edit the values of attributes in the table with **Edit**. Some values have special meanings, for example, changing the **cn** value renames the current module.

To configure the default template of the current module, click **Configure Template**.

Module Configuration

Configuration Module: userconfiguration

Attribute	Value
cn	userconfiguration
susedefaultbase	ou=people,dc=my-domain,dc=com
susedefaulttemplate	cn=usertemplate,ou=ldapconfig,dc=my-domain,dc=com
susemapattribute	
susemaxpasswordlength	8
susemaxuniqueid	60000
suseminpasswordlength	5
suseminuniqueid	1000
susenextuniqueid	1000
susepasswordhash	CRYPT
susesearchfilter	objectclass=posixaccount
suseskeldir	/etc/skel

The dialog for module configuration (Figure 45.4, "YaST: Module Configuration" (page 666)) allows the creation of new modules, selection and modification of existing configuration modules, and design and modification of templates for such modules.

To create a new configuration module, proceed as follows:

1. Click *New* and select the type of module to create. For a user configuration module, select `suseuserconfiguration` and for a group configuration choose `susegroupconfiguration`.

2. Choose a name for the new template.

   The content view then features a table listing all attributes allowed in this module with their assigned values. Apart from all set attributes, the list also contains all other attributes allowed by the current schema but currently not used.

3. Accept the preset values or adjust the defaults to use in group and user configuration by selecting the respective attribute, pressing *Edit*, and entering the new value. Rename a module by simply changing the `cn` attribute of the module. Clicking *Delete* deletes the currently selected module.

4 After you click *OK*, the new module is added to the selection menu.

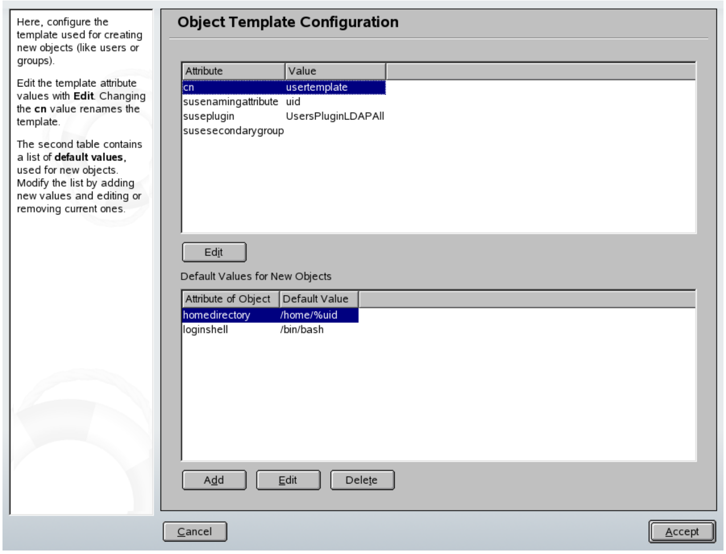
The YaST modules for group and user administration embed templates with sensible standard values. To edit a template associated with a configuration module, proceed as follows:

1 In the *Module Configuration* dialog, click *Configure Template*.

2 Determine the values of the general attributes assigned to this template according to your needs or leave some of them empty. Empty attributes are deleted on the LDAP server.

3 Modify, delete, or add new default values for new objects (user or group configuration objects in the LDAP tree).

Figure 45.5 *YaST: Configuration of an Object Template*



Here, configure the template used for creating new objects (like users or groups).

Edit the template attribute values with **Edit**. Changing the **cn** value renames the template.

The second table contains a list of **default values**, used for new objects. Modify the list by adding new values and editing or removing current ones.

Object Template Configuration

Attribute	Value
cn	usertemplate
susenamingattribute	uid
suseplugin	UsersPluginLDAPAll
susesecondarygroup	

Default Values for New Objects

Attribute of Object	Default Value
homedirectory	/home/%uid
loginshell	/bin/bash

Connect the template to its module by setting the `susedefaulttemplate` attribute value of the module to the DN of the adapted template.

TIP

The default values for an attribute can be created from other attributes by using a variable instead of an absolute value. For example, when creating a new user, `cn=%sn %givenName` is created automatically from the attribute values for `sn` and `givenName`.

Once all modules and templates are configured correctly and ready to run, new groups and users can be registered in the usual way with YaST.

45.6 Configuring LDAP Users and Groups in YaST

The actual registration of user and group data differs only slightly from the procedure when not using LDAP. The following brief instructions relate to the administration of users. The procedure for administering groups is analogous.

1 Access the YaST user administration with *Security & Users* → *User Administration*.

2 Use *Set Filter* to limit the view of users to the LDAP users and enter the password for Root DN.

3 Click *Add* and enter the configuration of a new user. A dialog with four tabs opens:

   a Specify username, login, and password in the *User Data* tab.

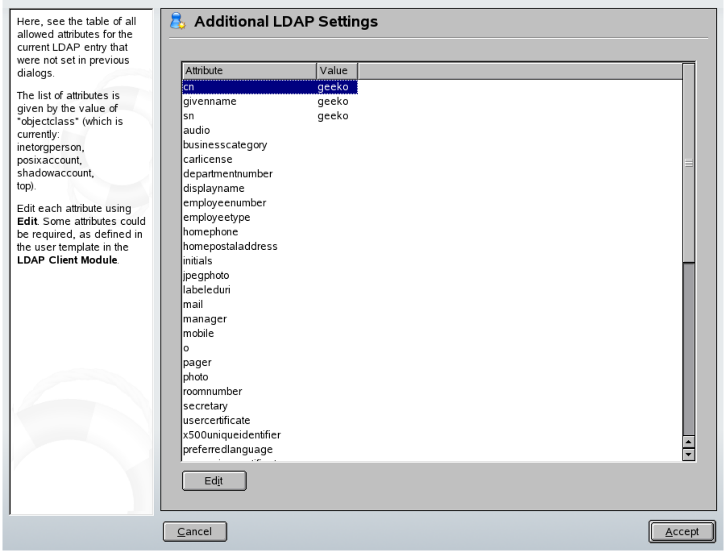
   b Check the *Details* tab for the group membership, login shell, and home directory of the new user. If necessary, change the default to values that better suit your needs. The default values as well as those of the password settings can be defined with the procedure described in Section “Configuring the YaST Group and User Administration Modules” (page 665).

   c Modify or accept the default *Password Settings*.

d Enter the *Plug-Ins* tab, select the LDAP plug-in, and click *Launch* to configure additional LDAP attributes assigned to the new user (see Figure 45.6, “YaST: Additional LDAP Settings” (page 669)).

4 Click *Accept* to apply your settings and leave the user configuration.

Figure 45.6 *YaST: Additional LDAP Settings*



Here, see the table of all allowed attributes for the current LDAP entry that were not set in previous dialogs.

The list of attributes is given by the value of "objectclass" (which is currently: inetorgperson, posixaccount, shadowaccount, top).

Edit each attribute using **Edit**. Some attributes could be required, as defined in the user template in the **LDAP Client Module**.

Attribute	Value
cn	geeko
givenname	geeko
sn	geeko
audio	
businesscategory	
carlicense	
departmentnumber	
displayname	
employeenumber	
employeetype	
homephone	
homepostaladdress	
initials	
jpegphoto	
labeleduri	
mail	
manager	
mobile	
o	
pager	
photo	
roomnumber	
secretary	
usercertificate	
x500uniqueidentifier	
preferredlanguage	

The initial input form of user administration offers *LDAP Options*. This gives the possibility to apply LDAP search filters to the set of available users or go to the module for the configuration of LDAP users and groups by selecting *LDAP User and Group Configuration*.

45.7 For More Information

More complex subjects, like SASL configuration or establishment of a replicating LDAP server that distributes the workload among multiple slaves, were intentionally not included in this chapter. Detailed information about both subjects can be found in the *OpenLDAP 2.2 Administrator's Guide* (references follow).

The Web site of the OpenLDAP project offers exhaustive documentation for beginning and advanced LDAP users:

OpenLDAP Faq-O-Matic

A very rich question and answer collection concerning installation, configuration, and use of OpenLDAP. Find it at `http://www.openldap.org/faq/data/cache/1.html`.

Quick Start Guide

Brief step-by-step instructions for installing your first LDAP server. Find it at `http://www.openldap.org/doc/admin22/quickstart.html` or on an installed system in `/usr/share/doc/packages/openldap2/admin-guide/quickstart.html`.

OpenLDAP 2.2 Administrator's Guide

A detailed introduction to all important aspects of LDAP configuration, including access controls and encryption. See `http://www.openldap.org/doc/admin22/` or, on an installed system, `/usr/share/doc/packages/openldap2/admin-guide/index.html`.

Understanding LDAP

A detailed general introduction to the basic principles of LDAP: `http://www.redbooks.ibm.com/redbooks/pdfs/sg244986.pdf`.

Printed literature about LDAP:

- *LDAP System Administration* by Gerald Carter (ISBN 1-56592-491-6)
- *Understanding and Deploying LDAP Directory Services* by Howes, Smith, and Good (ISBN 0-672-32316-8)

The ultimate reference material for the subject of LDAP is the corresponding RFCs (request for comments), 2251 to 2256.

46 The Apache Web Server

With a share of more than 60%, Apache is the world's most widely-used Web server according to http://www.netcraft.com. For Web applications, Apache is often used on Linux, with the database MySQL, and the programming languages PHP and Perl. This combination is commonly referred to as LAMP.

This chapter introduces the Web and application server software Apache in version 2.x. Installation and configuration of Apache are explained here, along with the usage of some of the available modules.

46.1 Preface and Terminology

This section provides definitions of frequently used terms, both Web-related and particular to Apache.

IMPORTANT: Terminology

In this document, the term *Apache* refers to Apache in version 2.x. For documentation on Apache 1.x, see http://httpd.apache.org/docs/.

46.1.1 Web Server

A Web server delivers Web pages requested by a client. The client can be a Web browser, such as Konqueror, or any other device that can connect to the World Wide

Web. These pages can be stored as a whole on disk (static pages) or generated in response to a query (dynamic pages) of an external entity, such as a database or a Web service.

46.1.2 HTTP

Communication between the client and the Web server takes place using the hypertext transfer protocol (HTTP). The current version, HTTP 1.1, is documented in RFC 2068 and in the update RFC 2616. These RFCs are available at `http://www.w3.org`.

46.1.3 URLs

URL stands for universal resource locator. Clients use URLs, such as `http://www.example.com/index.html`, to request pages from the server. A URL consists of:

Protocol

Frequently-used protocols:

http://

The HTTP protocol

https://

Secure, encrypted version of HTTP

ftp://

File transfer protocol for downloading and uploading files

Domain

In this example, the domain is `www.example.com`. The domain is the name that corresponds to an IP address. Thus, `www.example.com` maps uniquely to an IP address like 123.456.789.1. In turn, the number uniquely identifies the computer running a Web server. The mapping of a domain name to an IP address is commonly referred to as *name resolution*. The domain can be subdivided into several parts, here: `www`, `example`, and `com`. The last part of the domain name is the top level domain (TLD). In this example, `com` is the TLD. TLD represents the top level of the name resolution process. TLDs can be generic (gTLDs, such as `com`, `org`, and net) or country-specific (ccTLDs, such as de for Germany). All parts of a domain together are referred to as the fully qualified domain name (FQDN).

Resource

In this example, the resource is index.html. This part specifies the full path to the resource. The resource can be a file, as in this example. However, it can also be a CGI script, a JavaServer page, or some other resource.

The responsible Internet mechanism, such as the domain name system (DNS) forwards the query to the domain www.example.com to one or several computers holding the resource. Apache then delivers the actual resource, in this example, the page index.html, to the client. In this case, the file is located in the top level directory. However, resources can also be located in subdirectories, as in http://www.example.com/linux/novell/suse.

46.1.4 Directive

For configuring Apache, the term *directive* is often used as a synoynm for “configuration option.” Directive is the technical term pertaining to the Apache Web server software.

46.2 Installation

Apache on SUSE Linux runs "out of the box" with a standard, predefined configuration. By following the instructions in this chapter, can have the Apache Web server up and running in little time. You must be root to install and configure Apache.

46.2.1 Installing Apache with YaST

The SUSE Linux apache2 package differs slightly in its file system and application layout from the standard software package available on the Apache Web site (http://httpd.apache.org). The following section describes the installation of the SUSE Linux apache2 package in detail and denotes the variations where applicable.

To install a simple Web server, proceed as follows:

Procedure 46.1 *Quick Installation*

1 Start YaST in GUI or command line mode.

2 Select *Network Services* → *HTTP Server*.

3 Click *Continue* to confirm the installation of packages `apache2` and `apache2-prefork`.

4 When the installation has finished, the *Apache Configuration Wizard* appears and you can start configuring the Web server.

The disadvantage in preceeding as instructed above is that there is an absence of PHP and database support. To install a Web server with PHP and database support, proceed as follows:

Procedure 46.2 *Installation of Simple Web Server*

1 Start YaST in GUI or command line mode.

2 Select *Software* → *Software Management.*

3 Select *Selections* in *Filter* then check *Simple Web Server with Apache2.*

4 Press *OK*.

5 Confirm the installation of the dependent packages to finish the SUSE Linux Apache2 installation process.

For advanced users, SUSE Linux offers custom package selection. To perform a custom installation of a Web server, proceed as follows:

Procedure 46.3 *Installation of the Default Apache RPM with YaST*

1 Start YaST in GUI or command line mode. Select *Software* → *Software Management.*

2 Select *Search* in *Filter* then enter `apache2` in the *Search* field.

3 Select `apache2` for installation.

4. Use step 2 and 3 for module selection. See Section 46.5, “Apache Modules” (page 700).

5. After selection press *Accept*.

6. You are then prompted to choose one of the dependencies for the necessary `apache2-MPM` package: `apache2-prefork` or `apache2-worker`. Refer to the Section 46.2.2, “Multiprocessing Modules” (page 675) for an explanation of the differences between the two. If you are not sure, select the `apache2-prefork` package, which is the default for Unix-based operating systems then press *OK*.

7. Confirm the installation of the dependent packages to finish the SUSE Linux Apache2 installation process.

NOTE: Starting a Web Server

Installing Apache does not start the Web server automatically. Refer to Section 46.3.3, “Activating, Starting, and Stopping Apache” (page 694) for information about controlling Apache start-up and shutdown.

46.2.2 Multiprocessing Modules

As mentioned in Installation of the Default Apache RPM with YaST (page 674), SUSE Linux provides two different multiprocessing modules (MPMs) for use with Apache. MPMs are responsible for accepting and handling requests to the Web server, representing the core of the Web server software.

Prefork MPM

The prefork MPM implements a nonthreaded, preforking Web server. It makes the Web server behave similarly to Apache version 1.x in that it isolates each request and handles it by forking a separate child process. Thus problematic requests cannot affect others, avoiding a lockup of the Web server.

While providing stability with this process-based approach, the prefork MPM consumes more system resources than its counterpart, the worker MPM. The prefork MPM is considered the default MPM for Unix-based operating systems.

IMPORTANT: MPMs in This Document

This document assumes Apache is used with the prefork MPM.

Worker MPM

The worker MPM provides a multithreaded web server. A thread is a “lighter” form of a process. The advantage of a thread over a process is its lower resource consumption. Instead of only forking child processes, the worker MPM serves requests by using threads with server processes. The preforked child processes are multithreaded.

This approach makes Apache perform better by consuming fewer system resources than the prefork MPM. One major disadvantage is the stability of the worker MPM: if a thread becomes corrupt, all threads of a process can be affected. In the worst case, this may result in a server crash. Especially when using CGI (described in Section “Common Gateway Interface: `mod_cgi` ” (page 702)) with Apache under heavy load, internal server errors might occur due to threads unable to communicate with system resources.

Another argument against using the worker MPM with Apache is that not all available Apache modules (see Section 46.5, “Apache Modules” (page 700)) are thread-safe and thus cannot be used in conjunction with the worker MPM.

WARNING: PHP as an Apache Module (`mod_php`)

Not all available PHP modules are thread-safe. Using the worker MPM with `mod_php` is strongly discouraged.

46.2.3 Default File System and Application Layout

SUSE Linux places files of the Apache package into default locations. The locations of the most important files are listed here.

Binaries

Most of the executable files in SUSE Linux Apache have a `2` appended to them. This simplifies differentiation of binary files for a parallel installation of Apache 1.x and Apache 2.x.

`/usr/sbin/httpd2`
Symbolic link pointing to the chosen multiprocessing module as described in Section 46.2.2, “Multiprocessing Modules” (page 675). The default is `httpd2-prefork`. The symlink is maintained by the start script according to the system configuration setting of the MPM.

`/usr/sbin/httpd2-prefork`
The actual Apache2 executable.

`/usr/sbin/apache2ctl`
Control script to start and stop the Web server, provided by the Apache HTTPD project. See Section 46.3.3, “Activating, Starting, and Stopping Apache” (page 694) for more information or run `/usr/sbin/apache2ctl help`.

`/etc/init.d/apache2`
Start and stop script providing full integration in the SUSE Linux installation and starting Apache at boot time. It checks for a valid configuration before starting and stopping the server and overrides the location of the configuration. It allows easy inclusion of further configuration files, loading of modules, or even start of a seperate instance of the server without modification of the script.

`/usr/sbin/rcapache2`
A convenient symlink for `/etc/init.d/apache2`, because `/etc/init.d/` is not in the path by default. Simply use `rcapache2 start` to start Apache.

`/usr/sbin/htpasswd2`
Utility to generate encrypted passwords for `.htaccess`-based authentication. Refer to the `htpasswd2(1)` man page for details on how to use the tool.

Configuration files

Most of the configuration files reside under `/etc/apache2`. For information about how to change configuration settings, refer to Section 46.3, “Configuration” (page 680).

`/etc/apache2/httpd.conf`
: Top-level configuration file. If possible, avoid changes to this file. It mainly includes other configuration files and declares global settings.

`/etc/apache2/*.conf`
: Some external Apache modules put their configuration files in the directory `/etc/apache2/`, usually prefixed with the module name itself (`mod_*.conf`).

`/etc/apache2/conf.d/*`
: Directory holding various other configuration files that come with certain packages. For an example, see Section "Serving PHP: `mod_php4`, `mod_php5` " (page 708).

`/etc/apache2/vhosts.d/*`
: Directory holding the optional configuration files for virtual hosts. See Section 46.4, "Virtual Hosts" (page 696) for details.

`/etc/sysconfig/apache2`
: SUSE Linux configuration file relating to Apache2. It holds all relevant configuration parameters for controlling the Apache Web server. `/etc/sysconfig/apache2` is used by YaST for configuring Apache as described in Section 46.3.1, "Configuring Apache with YaST" (page 680). It can also be edited manually as described in Section 46.3.2, "Configuring Apache Manually" (page 688).

Log Files

By default, Apache provides various information about its runtime status in the following files:

`/var/log/apache2/error_log`
: Apache logs start-up and shutdown notices and all runtime errors into this file.

`/var/log/apache2/access_log`
: All requests to the Web server are logged into this file. The default format of the entries is combined format, showing information about the host and user agent sending the request and the referring URI.

Document Root

The physical directory /srv/www/htdocs is the default location from which Apache serves Web pages. It acts as “root directory” for a client request. To publish Web pages with Apache, store the files hierarchically in or under that directory.

A URL like http://www.example.com/index.html refers to /srv/www/htdocs/index.html in the default Apache configuration in SUSE Linux for a domain named example.com.

46.2.4 Building Modules Manually

Apache is built with a modular approach, meaning that modules provide the capabilities of the Web server software itself. Consequently, Apache can be extended by advanced users by writing custom modules. Refer to the man pages mentioned in the following for more detailed information.

apache2-devel

To be able to develop modules for Apache or compile third-party modules, the package apache2-devel is required along with the corresponding development tools. apache2-devel also contains the apxs2 tools, which are necessary for compiling additional modules for Apache.

apxs2

The apxs2 binaries are located under /usr/sbin:

- /usr/sbin/apxs2—suitable for building an extension module that works with any MPM. The installation location is /usr/lib/apache2.

- /usr/sbin/apxs2-prefork—suitable for prefork MPM modules. The installation location is /usr/lib/apache2-prefork.

- /usr/sbin/apxs2-worker—suitable for worker MPM modules.

apxs2 installs modules so they can be used for all MPMs. The other two programs install modules so they can only be used for the respective MPMs. apxs2 installs modules in /usr/lib/apache2 and apxs2-prefork installs modules in /usr/lib/apache2-prefork.

apxs2 enables the compilation and installation of modules from source code (including the required changes to the configuration files), which creates *dynamic shared objects* (DSOs) that can be loaded into Apache at runtime. Install a module from source code with the commands `cd /path/to/module/source; apxs2 -c -i *mod_foo*.c`. Other options of apxs2 are described in the apxs2(1) man page. The modules should then be activated in /etc/sysconfig/apache2 with the entry APACHE_MODULES as described in Section 46.3.2, “Configuring Apache Manually” (page 688).

46.3 Configuration

Apache in SUSE Linux can be configured in two different ways: with YaST or manually. Manual configuration offers a higher level of detail, but lacks the convenience of the YaST GUI.

IMPORTANT: Configuration Changes

Changes to some configuration values for Apache only take effect after Apache is restarted. This happens automatically when finishing the configuration using YaST with *Enabled* checked for the *HTTP Service*. Manual restart is described in Section 46.3.3, “Activating, Starting, and Stopping Apache” (page 694). Most configuration changes only require a reload with `rcapache2 reload`.

46.3.1 Configuring Apache with YaST

With YaST, you can turn a host in your network into a Web server. To configure such a server, start YaST and select *Network Services* → *HTTP Server*. When starting the module for the first time, the HTTP Server Wizard starts, prompting you to make just a few basic decisions concerning administration of the server.

HTTP Server Wizard

The HTTP Server Wizard consists of five steps or dialogs. In the last step of the dialog, you are given the opportunity to enter the expert configuration mode to make even more specific settings.

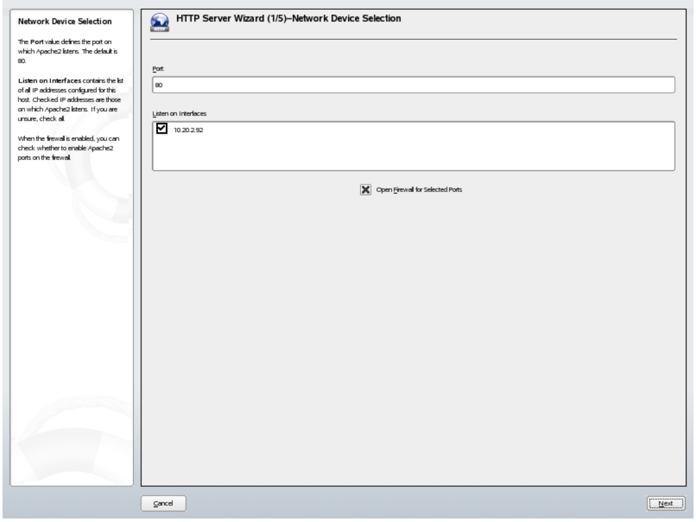
Network Device Selection

Here, specify the network interfaces and ports Apache uses to listen for incoming requests. You can select any combination of existing network interfaces and their respective IP addresses. Ports from all three ranges (well-known ports, registered ports, and dynamic or private ports) that are not reserved by other services can be used.

The default setting is to listen on all network interfaces (IP addresses) on port 80. When the firewall is enabled, you can check whether to enable Apache ports on the firewell.

Check *Open Firewall for Selected Ports* to open the ports in the firewall that the Web server listens on. This is necessary to make the Web server available on the network, which can be a LAN, WAN, or the public Internet. Keeping the `Listen` port closed is useful in test situations where no external access to the Web server is necessary. If you are satisfied with the default settings or if you have made any changes, click *Next* to continue with configuration.

Figure 46.1 *HTTP Server Wizard: Network Device Selection*



Network Device Selection

The **Port** value defines the port on which Apache2 listens. The default is 80.

Listen on Interfaces contains the list of all IP addresses configured for this host. Checked IP addresses are those on which Apache2 listens. If you are unsure, check all.

When the firewall is enabled, you can check whether to enable Apache2 ports on the firewall.

HTTP Server Wizard (1/5)–Network Device Selection

Port: 80

Listen on Interfaces

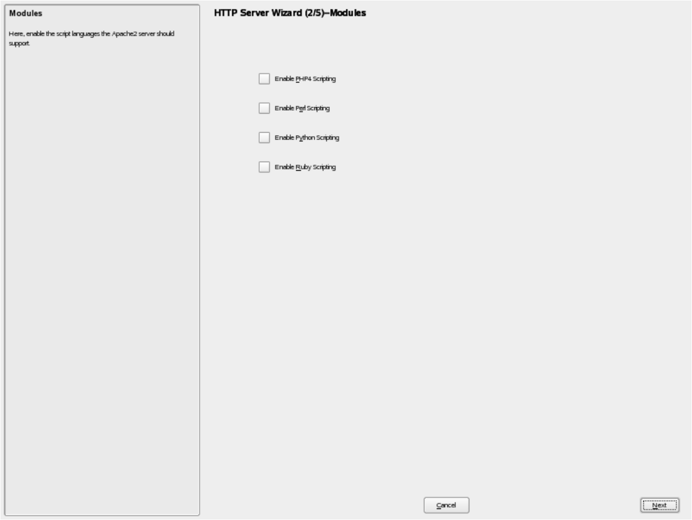
☑ 10.20.2.92

☒ Open Firewall for Selected Ports

Modules

The SUSE Linux Apache package comes with a wide variety of Apache modules. Modules extend Apache's functionality and are available for a wide range of tasks. The *Modules* configuration option allows for the loading and unloading of various Apache modules at when the server is started. For a more detailed explanation of modules, refer to Section 46.5, “Apache Modules” (page 700). Click *Next* to continue.

Figure 46.2 *HTTP Server Wizard: Modules*



Here, enable the script languages the Apache2 server should support.

□ Enable PHP4 Scripting

□ Enable Perl Scripting

□ Enable Python Scripting

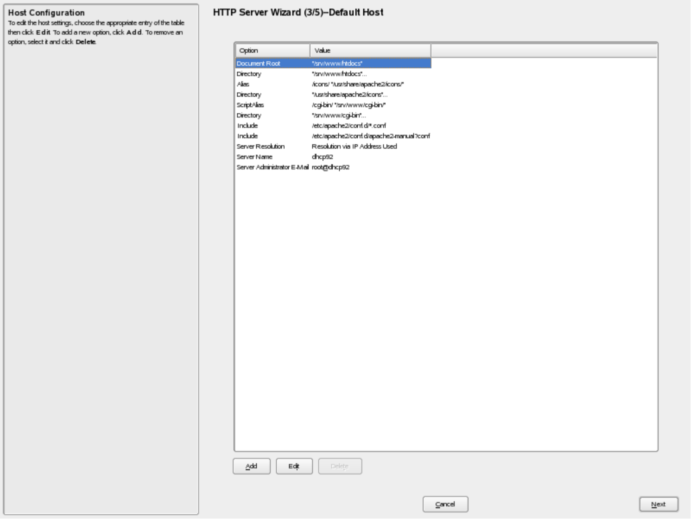
□ Enable Ruby Scripting

Default Host

This option pertains to the default Web server. As explained in Section 46.4, “Virtual Hosts” (page 696), Apache can serve multiple domains from a single physical machine. The first declared domain (or `VirtualHost`) in the configuraton file is commonly referred to as the *Default Host*. To edit the host settings, choose the appropriate entry in the table then click *Edit*. To add a new host, click *Add*. To delete a host, select it and click *Delete*.

In this step, you can decide to add an SSL (secure sockets layer) option and value to the host settings. You can read more about this in Section “Adding SSL Support” (page 688).

Figure 46.3 *HTTP Server Wizard: Default Host*



Host Configuration

To edit the host settings, choose the appropriate entry of the table then click **Edit**. To add a new option, click **Add**. To remove an option, select it and click **Delete**.

HTTP Server Wizard (3/5)–Default Host

Option	Value
Document Root	"/srv/www/htdocs"
Directory	"/srv/www/htdocs"...
Alias	/icons/ "/usr/share/apache2/icons/"
Directory	"/usr/share/apache2/icons"...
ScriptAlias	/cgi-bin/ "/srv/www/cgi-bin/"
Directory	"/srv/www/cgi-bin"...
Include	/etc/apache2/conf.d/*.conf
Include	/etc/apache2/conf.d/apache2-manual?conf
Server Resolution	Resolution via IP Address Used
Server Name	dhcp92
Server Administrator E-Mail	root@dhcp92

Here is list of the default settings of the server:

Document Root

As described in Section "Document Root" (page 679), `/srv/www/htdocs` is the default location from which Apache serves Web pages.

Directory

`/srv/www/htdocs` is the location of the Web pages.

Alias

With the help of `Alias` directives, URLs can be mapped to physical file system locations. This means that a certain path even *outside* the `Document Root` in the file system can be accessed via a URL aliasing that path.

The default SUSE Linux `Alias /icons` points to `/usr/share/apache2/icons` for the Apache icons displayed in the directory index view.

Directory

`/usr/shareapache2/icons` is the location of the `Alias` directory.

Script Alias

Similar to the `Alias` directive, the `ScriptAlias` directive maps a URL to a file system location. The difference is that `ScriptAlias` designates the target directory as a CGI location, meaning that CGI scripts should be executed in that location.

Directory

`/srv/www/cgi-bin` is the location of the `ScriptAlias` directory.

Include

`/etc/apache2/conf.d/*.conf` is the directory containing the configuration files that come with certain packages. `/etc/apache2/conf.d/apache2-manual?conf` is the directory containing all `apache2-manual` configuration files.

Server Resolution

This option refers to Section 46.4, "Virtual Hosts" (page 696).

Determine Request Server by HTTP Headers lets a `VirtualHost` answer on a request to its server name (see Section 46.4.1, "Name-Based Virtual Hosts" (page 696)).

Determine Request Server by Server IP Address makes Apache select the requested host by the HTTP header information the client sends. See Section 46.4.2, "IP-Based Virtual Hosts" (page 698) for more details on IP-based virtual hosts.

Server Name

This specifies the default URL used by clients to contact the Web server. Usa a FQDN (see Domain (page 672)) to reach the Web server at `http://`*`FQDN`* or its IP address.

Server Administrator E-Mail

Provide the Web server administrator's e-mail address for *Server Administrator E-Mail*.

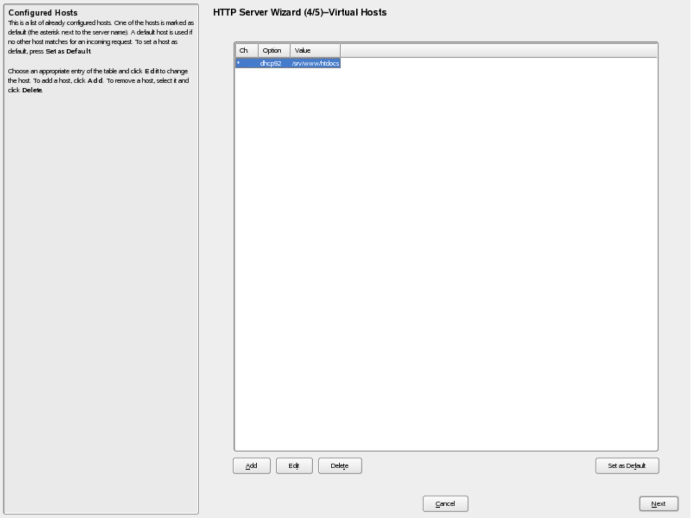
After finishing with the *Default Host* step, click *Next* to continue with the configuration dialog.

Virtual Hosts

In this step, the wizard displays a list of already configured virtual hosts (see Section 46.4, "Virtual Hosts" (page 696)). One of the hosts is marked as default (with an asterisk next to the server name). To set a default host, select the server and click *Set as Default*.

To add a host, click *Add* and a dialog appears in which to enter basic information about the host. *Server Indentification* includes the server name, server contents root, and administrator e-mail. The help text in the left frame of the window explains each of these items in detail. *Server Resolution* is used to determine how a host is identified. You can specify whether to determine a request server from HTTP headers or by server IP address by selecting the respective option. The other possibility is to determine the virtual host by the IP address used by the client when connecting to the server. You can also choose to enable SSL support by checking that option. The certificate file path can also be specified. By clicking *Browse*, the default directory `/etc/apache2/ssl.crt` is displayed. After all information has been entered, click *Next* to continue to the final step of configuration.

Figure 46.4 *HTTP Server Wizard: Virtual Hosts*



Configured Hosts

This is a list of already configured hosts. One of the hosts is marked as default (the asterisk next to the server name). A default host is used if no other host matches for an incoming request. To set a host as default, press **Set as Default**

Choose an appropriate entry of the table and click **Edit** to change the host. To add a host, click **Add**. To remove a host, select it and click **Delete**.

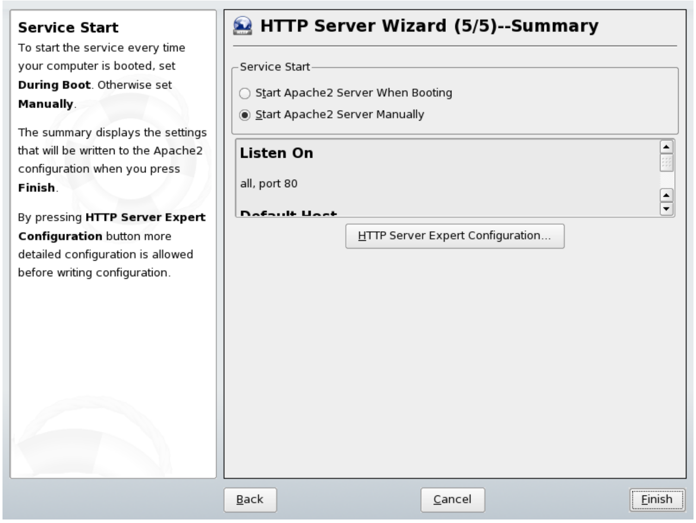
HTTP Server Wizard (4/5)–Virtual Hosts

Ch.	Option	Value
*	dhcp92	/srv/www/htdocs

Summary

This is the final step of the wizard. Here, determine how and when the Apache server is started: when booting or manually. The port selected earlier is also displayed along with the default and virtual hosts. If you are satisfied with your settings, click *Finish* to complete configuration.

Figure 46.5 *HTTP Server Wizard: Summary*



Service Start

To start the service every time your computer is booted, set **During Boot**. Otherwise set **Manually**.

The summary displays the settings that will be written to the Apache2 configuration when you press **Finish**.

By pressing **HTTP Server Expert Configuration** button more detailed configuration is allowed before writing configuration.

HTTP Server Wizard (5/5)--Summary

Service Start

○ Start Apache2 Server When Booting

● Start Apache2 Server Manually

Listen On

all, port 80

Default Host

HTTP Server Expert Configuration

The HTTP Server module also lets you make even more adjustments to the configuration. Click *HTTP Server Expert Configuration* to see more configuration options. The following changes can then be made:

Listen On
Selecting the *Listen on* setting and clicking *Edit* opens a new window in which you can add, delete, or edit entries.

Modules
By selecting the *Modules* settings and clicking *Edit*, you can change the status of Apache2 modules by clicking *Toggle Status*. Click *Add Module* to add a new module.

Default Host
Selecting *Default Host* and clicking *Edit* lets you edit host settings. You can also add, edit, or delete options.

Hosts
By selecting *Hosts* and clicking *Edit*, you can add, delete, edit, or select a host as the default.

In all of the preceding dialogs, you can click *Log Files* to view the error log and access log. Click *OK* to complete configuration and return to the YaST Control Center.

Adding SSL Support

To add an SSL option to the host, click *Add* from step three (default host) of the HTTP Server Wizard. If your server has already been set up and you no longer have access to the wizard, you can set up an SSL option by selecting *Default Hosts* from the HTTP Server Configuration dialog or clicking *Edit*, and *Add*. In both cases, a pop-up window appears in which you scroll to an *SSL* option and confirm with *OK*. You are then asked to enter a value for the option selected. This may be as simple as setting the value to *on* or *off*, however, the dialog may require that you enter an appropriate value. If uncertain, refer to documentation for value parameters when configuring SSL. After you click *OK*, the option and value appear in the host configuration list. Clicking *Next* takes you to the next step in the configuration dialog.

If *SSL* appears in the host configuration list, click *Edit* to open the SSL configuration dialog. If it is not displayed, click *Add*, select*SSL*, and *OK* and the dialog opens automatically. Here, add, delete, or edit SSL options. Click *OK* to return to the HTTP Server Wizard.

46.3.2 Configuring Apache Manually

Configuring Apache manually involves editing the plain text configuration files as the user `root`.

IMPORTANT: No SuSEconfig Module for Apache2

The SuSEconfig module for Apache2 has been removed from SUSE Linux. It is no longer necessary to run `SuSEconfig` after changing `/etc/sysconfig/apache2`.

/etc/sysconfig/apache2

`/etc/sysconfig/apache2` controls some global settings of Apache, like modules to load, additional configuration files to include, flags with which the server should be started, and flags that should be added to the command line. Every configuration option in this file is extensively documented and therefore not mentioned here. For a general-purpose Web server, `/etc/sysconfig/apache2` should be sufficient for any configuration needs. If a specific configuration is needed, refer to Section “Apache Directives in /etc/apache2/httpd.conf: Global Environment ” (page 689).

IMPORTANT: Files Created Automatically on Server Start

`/etc/sysconfig/apache2` creates or edits the following files automatically when the Web server is started or restarted.

- `/etc/apache2/sysconfig.d/loadmodule.conf`—modules that are loaded at runtime

- `/etc/apache2/sysconfig.d/global.conf`—serverwide general settings

- `/etc/apache2/sysconfig.d/include.conf`—list of included configuration files

Do not edit these files manually. Instead, edit the corresponding settings in `/etc/sysconfig/apache2`.

For fine-grained configuration tweaks, look at the files in `/etc/apache2/*`, specifically for changes on manual configuration of virtual hosts, the global environment, or the main server.

Apache Directives in /etc/apache2/httpd.conf: Global Environment

SUSE Linux uses `/etc/apache2/httpd.conf` as a central point of reference for other configuration files. Edit the file only to enable features that are not available in `/etc/sysconfig/apache2`. The directives in the *Global Environment* section of `httpd.conf` affect the overall operation of Apache.

The following sections describe some of the directives that are not available in YaST. Core directives like `Document Root` (Document Root (page 684)) are essential and required both in `Global Environment` and for `VirtualHost`.

The following parameters and directives are ordered by logical affiliation and configuration scope. All of these should be set in `/etc/apache2/httpd.conf`.

`LoadModule` ***`module_identifier /path/to/module`***

The `LoadModule` directive specifies an Apache module to load at runtime. *`module_identifier`* is the name of the module according to its documentation. *`/path/to/module`* can be an absolute or relative path pointing to the file.

Example 46.1 *LoadModule Directive*

```
LoadModule rewrite_module /usr/lib/apache2-prefork/mod_rewrite.so
```

On SUSE Linux, it is not necessary to use `LoadModule` statements directly. Instead, `APACHE_MODULE` is used in `/etc/sysconfig/apache2`.

`MaxClients` ***`number`***

The maximum number of clients Apache can handle concurrently. MaxClients must be large enough to handle as many simultaneous requests as the Web site expects to receive, but small enough to assure that there is enough physical RAM for all processes.

`Timeout` ***`seconds`***

Specifies the time period Apache waits before reporting a time-out for a request.

Apache Directives in /etc/apache2/httpd.conf: Main Server

The directives in the `Main Server` section apply when client requests are not handled by any `VirtualHost` and therefore need to be processed by a default or main server. Additionally, the parameters defined in this context are the defaults for all configured virtual hosts. As a consequence, all of the directives in the `Main Server` can also be set in the `VirtualHost` context, overwriting the defaults.

`DirectoryIndex` ***`filenames`***

Set which files Apache should search to complete a URL lacking a file specification. The default setting is `index.html`. For example, if the client requests the URL http://www.example.com/foo/ and the directory foo contains a file called index.html, Apache delivers this page to the client. Declare multiple files by separating them with spaces.

Example 46.2 *DirectoryIndex Directive*

```
DirectoryIndex index.html index.shtml start.php begin.pl
```

AllowOverride All | None | *option*

This directive can *only* be used inside a <Directory></Directory> declaration. See Directory (page 684).

AllowOverride specifies what access and display options a .htaccess file (or other files specified by AccessFileName as described in Section "AccessFileName *filenames* " (page 692)) can override.

Possible values are:

All
All options can be overridden by a .htaccess file.

None
No option can be overridden by a .htaccess file.

AuthConfig
Directories can be password protected with the help of a .htaccess file.

FileInfo
Allows the use of directives controlling document types within a .htaccess file. A typical example for this is to configure custom error pages with ErrorDocument (see http://httpd.apache.org/docs-2.0/mod/core.html#errordocument).

Indexes
In the event that no DirectoryIndex document is found, this parameter allows Apache to control the display of directory contents.

Limit
Controls access to a directory or to certain files for clients. The directives Allow, Deny, and Order are used within a .htaccess file for this purpose. For usage

of these directives, see the access module documentation (http://httpd.apache.org/docs-2.0/mod/mod_access.html).

`Options`

Allow the usage of the `Options` and `XBitHack` directives within a `.htaccess` file. The `Options` directive (http://httpd.apache.org/docs-2.0/mod/core.html#options) controls which server features are available in a particular directory. The `XBitHack` directive (http://httpd.apache.org/docs-2.0/mod/mod_include.html#xbithack) allows files with the execute bit set to be parsed as SSI (see Section "Server-Side Includes with `mod_include` " (page 701)).

IMPORTANT

These settings are applied recursively to the current directory and its subdirectories. These options, except `All` and `None`, can be combined, separated by spaces.

Example 46.3 *AllowOverride Directive*

```
<Directory /srv/www/htdocs>
    AllowOverride None
</Directory>
<Directory /srv/www/htdocs/project>
    AllowOverride All
</Directory>
<Directory /srv/www/htdocs/project/webapp>
    AllowOverride Indexes Limit AuthConfig
</Directory>
```

`AccessFileName` ***`filenames`***

`AccessFileName` sets the name for the files that can override the global access permissions and other settings for directories (see Directory (page 684)).

The default setting is `.htaccess`. Declare multiple files by separating them with spaces.

Example 46.4 *AccessFileName Directive*

```
AccessFileName .htaccess .acl permission.txt
```

ErrorLog ***file | "|command"***

Specifies the name of the file to which Apache logs error messages. Alternatively, Apache can also log to a command or script. The default setting is `/var/log/apache2/error_log`.

Example 46.5 *ErrorLog Directive*

```
ErrorLog /var/log/apache2/error_log
ErrorLog "|/path/to/script"
```

LogLevel ***level***

This sets the verbosity of the log messages to record. In ascending order of level of verbosity (and descending severity of messages), *`level`* can be

- `emerg`
- `alert`
- `crit`
- `error`
- `warn`
- `notice`
- `info`
- `debug`

The default setting is `warn`, which is recommended for everyday operation. For debugging purposes, `info` and `debug` provide helpful information.

Example 46.6 *LogLevel Directive*

```
LogLevel debug
```

Apache Directives in /etc/apache2/httpd.conf: Virtual Hosts Section

To maintain multiple domains or hostnames on one physical machine, `VirtualHost` containers are needed. They are declared in `Virtual Hosts` sections of the configuration. For more details on the syntax for and functionality of virtual hosts, refer to Section 46.4, "Virtual Hosts" (page 696).

46.3.3 Activating, Starting, and Stopping Apache

To activate the Apache Web server at boot time, use YaST's runlevel editor. To start it, select *System* → *System Services (Runlevel)* in YaST. Then navigate to the *apache2* entry. Choose *Enable* to have Apache start automatically when the machine is booted. Experienced users may want to use the `chkconfig` tool to achieve the same on the command line: `/sbin/chkconfig -a apache2`.

To start or stop Apache, use the `/usr/sbin/rcapache2` script as the `root` user. `/usr/sbin/rcapache2` takes the following parameters for starting and stopping the Apache Web server:

`start`
Starts the Apache Web Server.

`startssl`
Starts the Apache Web Server with SSL support. For information about configuring Apache with SSL, refer to Section "Adding SSL Support" (page 688) and Section "Secure Sockets Layer and Apache: `mod_ssl`" (page 705).

`stop`
Stops the Apache Web server.

`configtest`
Tests the Apache configuration without actually stopping, starting, or restarting the Web server. Because this test is forced everytime the server is started, reloaded, or restarted, it is usually not necessary to run the test explicitly.

`restart`

First stops then starts the Web server again.

`try-restart`

Restarts the Web server if it is running.

`restart-hup`

Restarts the Apache Web server by sending it a `SIGHUP` signal. This is normally not used.

`graceful` and `reload`

Stops the Web server by advising all forked Apache processes to first finish their request before shutting down. As each process dies, it is replaced by a newly started one, resulting in complete "restart" of Apache.

TIP

`rcapache2 reload` is the preferred method of restarting Apache in production environments, because it allows all clients to be served without causing connection break-offs.

`status`

Checks the runtime status of the Apache Web server.

***Example 46.7** Example Output When Starting and Stopping Apache*

```
tux@sun # rcapache2 status
Checking for httpd2:                                  unused

tux@sun # rcapache2 configtest
Syntax OK

tux@sun # rcapache2 start
Starting httpd2 (prefork)                             done

tux@sun # rcapache2 status
Checking for httpd2:                                  running

tux@sun # rcapache2 graceful
Reload httpd2 (graceful restart)                      done

tux@sun # rcapache2 status
Checking for httpd2:                                  running
```

A malformed configuration file can result in Apache not starting correctly or not starting at all. When not starting at all, there might not even be any message displayed. Always check the main error log for every start and restart.

46.4 Virtual Hosts

The term *virtual host* refers to Apache's ability to serve multiple URIs (universal resource identifiers) from the same physical machine. This means that several domains, such as www.example.com and www.example.net, are run by a single Web server on one physical machine.

It is common practice to use virtual hosts to save administrative effort (only a single Web server needs to be maintained) and hardware expenses (each domain does not require a dedicated server). Virtual hosts can be name based, IP based, or port based.

Virtual hosts can be configured via YaST (see Default Host (page 683)) or by manually editing the `Virtual Host` section of `httpd.conf` (see Section 46.3.2, “Configuring Apache Manually” (page 688)).

By default, Apache in SUSE Linux is prepared for one configuration file per virtual host in `/etc/apache2/vhosts.d/`. A basic template for a virtual host is provided in this directory (`vhost.template`). Virtual host configuration can also be added elsewhere, for example, in one file that is then included in the configuration.

IMPORTANT

It is very helpful to check the virtual host setup with the command `httpd2 -S`. It outputs the virtual host settings as understood by Apache and can help you make sure that you get the expected results. If you use Apache with flags like -DSSL, it is necessary to use the same flags when testing, for example, use `httpd2 -S -DSSL`.

46.4.1 Name-Based Virtual Hosts

With name-based virtual hosts, more than one Web site is served per IP address. Apache uses the host field in the HTTP header sent by the client to connect the request to a matching ServerName entry of one of the virtual host declarations. If no matching ServerName is found, the first specified VirtualHost is used as a default.

NameVirtualHost starts the Virtual Host section in an Apache configuration.

NameVirtualHost

NameVirtualHost tells the Apache Web server on which IP address and, optionally, which port to listen for requests by clients containing the domain name in the HTTP header.

The first argument can be a fully qualified domain name, but it is recommended to use the IP address. The second argument is the port and is optional. By default, port 80 is used and is configured via the Listen directive (Network Device Selection (page 681)).

The wild card * can be used for both the IP address and the port number to receive requests on all interfaces. IPv6 addresses must be enclosed in square brackets.

Example 46.8 *Variations of Name-Based VirtualHost Entries*

```
# NameVirtualHost IP-address[:Port]
NameVirtualHost 192.168.1.100:80
NameVirtualHost 192.168.1.100
NameVirtualHost *:80
NameVirtualHost *
NameVirtualHost [2002:c0a8:164::]:80
```

<VirtualHost></VirtualHost> in the Name-Based Context

The <VirtualHost></VirtualHost> blocks holds the information that applies to a particular domain. When Apache receives a client request for a defined VirtualHost, it uses the directives enclosed in this section. Any Apache directive that is allowed in the VirtualHost context can be used here. The opening VirtualHost tag takes the following arguments in a name-based virtual host configuration:

- IP address (or fully qualified domain name) previously declared with the NameVirtualHost directive.

- Optional port number previously declared with the NameVirtualHost directive.

The wild card * is also allowed as a substitute for the IP address. This syntax is only valid in combination with the wild card usage in `NameVirtualHost *`. When using IPv6 addresses, the address must be included in square brackets.

Example 46.9 *Name-Based VirtualHost Directives*

```
<VirtualHost 192.168.1.100:80>
    ServerName www.example.com
    DocumentRoot /srv/www/htdocs/example.com
    ServerAdmin webmaster@example.com
    ErrorLog /var/log/apache2/www.example.com-error_log
    CustomLog /var/log/apache2/www.example.com-access_log common
</VirtualHost>

<VirtualHost 192.168.1.100:80>
    ServerName www.example.net
    DocumentRoot /srv/www/htdocs/example.net
    ServerAdmin webmaster@example.net
    ErrorLog /var/log/apache2/www.example.net-error_log
    CustomLog /var/log/apache2/www.example.net-access_log common
</VirtualHost>

<VirtualHost [2002:c0a8:164::]>
    # 2002:c0a8:164:: is the IPv6 equivalent to 192.168.1.100
    ServerName www.example.org
    DocumentRoot /srv/www/htdocs/example.org
    ServerAdmin webmaster@example.org
    ErrorLog /var/log/apache2/www.example.org-error_log
    CustomLog /var/log/apache2/www.example.org-access_log common
</VirtualHost>
```

In this example, both the domain www.example.com and www.example.net are hosted on the machine with the IP address `192.168.1.100`. The first `VirtualHost` is the default for all incoming requests to the Web server.

The directives `ErrorLog` (described in Section “`ErrorLog` *`file | "|command"`*” (page 693)) and `CustomLog` (see `http://httpd.apache.org/docs-2.0/mod/mod_log_config.html#customlog`) do not need to contain the domain name. Here, use a name of your choice.

46.4.2 IP-Based Virtual Hosts

This alternative virtual host configuration requires the setup of multiple IPs for a machine. One instance of Apache hosts several domains, each of which is assigned a different IP.

IMPORTANT: IP Addresses and IP-Based Virtual Hosts

The physical server must have one IP address for each IP-based virtual host. If the machine does not have multiple network cards, virtual network interfaces (IP aliasing) can also be used.

Configuring IP Aliasing

For Apache to host multiple IPs, the physical machine must accept requests for multiple IPs. This is called multi-IP hosting. Additionally, IP aliasing must be activated in the kernel. This is the default setting in SUSE Linux.

Once the kernel has been configured for IP aliasing, the commands `ifconfig` and `route` can be used to set up additional IPs on the host. These commands must be executed as `root`.

For the following example, it is assumed that the host already has the IP `192.168.0.10` assigned for the network device `eth0`. Enter the command `ifconfig` to view the IP of the host. Further IP addresses can be added with the following commands:

```
ip addr add 192.168.0.20/24 dev eth0
ip addr add 192.168.0.30/24 dev eth0
```

All these IP addresses are assigned to the same physical network device (`eth0`).

`<VirtualHost></VirtualHost>` in the IP-Based Context

Once IP aliasing has been set up on the system (or the host has been equipped with several network cards), Apache can be configured. A separate `VirtualHost` block is needed for every virtual server.

The following example shows Apache running on a machine with the IP `192.168.1.10`, hosting two domains on the additional IPs `192.168.0.20` and `192.168.0.30`. This particular example only works on a private network, because IPs ranging from `192.168.0.0` to `192.168.0.255` are not routed to the public Internet.

Example 46.10 *IP-Based VirtualHost Directives*

```
<VirtualHost 192.168.0.20>
    ServerName www.example.com
    DocumentRoot /srv/www/htdocs/example.com
    ServerAdmin webmaster@example.com
    ErrorLog /var/log/apache2/www.example.com-error_log
    CustomLog /var/log/apache2/www.example.com-access_log common
</VirtualHost>

<VirtualHost 192.168.0.30>
    ServerName www.example.net
    DocumentRoot /srv/www/htdocs/example.net
    ServerAdmin tux@example.net
    ErrorLog /var/log/apache2/www.example.net-error_log
    CustomLog /var/log/apache2/www.example.net-access_log common
</VirtualHost>
```

Here, `VirtualHost` directives are only specified for interfaces other than `192.168.0.10`. When a `Listen` directive (described in Network Device Selection (page 681)) is also configured for `192.168.0.10`, a separate IP-based virtual host must be created to answer HTTP requests to that interface or the directives found in the `Main Server` section of `/etc/apache2/httpd.conf` (see Section "Apache Directives in /etc/apache2/httpd.conf: Main Server " (page 690)) are applied.

46.5 Apache Modules

The Apache software is built in a modular fashion: all functionality except some core tasks is handled by modules. This has progressed so far that even HTTP is processed by a module (http_core).

Apache modules can be compiled into the Apache binary at buildtime or dynamically loaded at runtime. For the runtime loading, refer to Section "`LoadModule` *`module_identifier /path/to/module`* " (page 690) for loading modules manually and to Modules (page 682) for using YaST.

Apache in SUSE Linux comes with the following modules readily available in the `apache2` RPM (prefix "mod_" omitted here): access, actions, alias, asis, auth, auth_anon, auth_dbm, auth_digest, auth_ldap, autoindex, cache, case_filter, case_filter_in, cern_meta, cgi, charset_lite, dav, dav_fs, deflate, dir, disk_cache, dumpio, echo, env, expires, ext_filter, file_cache, headers, imap, include, info, ldap, log_config, log_forensic, logio, mem_cache, mime, mime_magic, negotiation, proxy, proxy_connect, proxy_ftp, proxy_http, rewrite, setenvif, speling, ssl, status, suexec, unique_id, userdir, usertrack, and vhost_alias. Additionally, SUSE Linux provides the following Apache modules as RPM packages that need to be installed separately: apache2-mod_auth_mysql, apache2-mod_fastcgi, apache2-mod_macro, apache2-mod_murka, apache2-mod_perl, apache2-mod_php4, apache2-mod_php5, apache2-mod_python, and apache2-mod_ruby.

Some of these modules are documented in more detail in this section. For a description of other modules in the base distribution, see the Apache Modules Web site at http://httpd.apache.org/docs-2.0/mod/. For third-party modules, refer to http://modules.apache.org/.

Apache modules can be divided into three different categories: base modules, extension modules, and external modules.

46.5.1 Base Modules

Base modules are compiled into Apache by default. They are available unless explicitly left out at buildtime. Apache in SUSE Linux has only the minimum base modules compiled in, but all of them are available as *shared objects*: rather than being included in the /usr/sbin/httpd2 binary itself, they can be included at runtime by configuring APACHE_MODULES in /etc/sysconfig/apache2.

Server-Side Includes with mod_include

mod_include provides a means of file processing before data is sent to the client. Typically, mod_include is used to include files in a document that are in turn parsed as HTML before they reach the client. This is why it is called server-side includes (SSI).

With SSIs, special commands are executed on the server side, triggered by formatted SGML comments. These SGML commands have the syntax:

```
<!--#element attribute=value -->
```

For a list of *element* and *attribute* values, see the mod_include documentation at http://httpd.apache.org/docs-2.0/mod/mod_include.html.

To use mod_include in SUSE Linux, add include to APACHE_MODULES in /etc/sysconfig/apache2 or use YaST as described in Modules (page 682).

TIP

Use the XBitHack directive (http://httpd.apache.org/docs-2.0/mod/mod_include.html#xbithack) to instruct Apache to parse files with the execute bit set for SSI directives.

This means that, rather than having to change the extension of a file to mark it as holding SSI elements (.shtml in the example above), you can use a regular .html file and run chmod +x *myfile*.html.

Common Gateway Interface: mod_cgi

mod_cgi enables Apache to deliver content created by external CGI ("Common Gateway Interface") programs or scripts. It acts as an instance between a programming language available on the physical machine and the Apache Web server. Theoretically, CGI scripts can be written in any programming language. Usually, languages such as Perl or C are used. mod_cgi is the most common way to include dynamic content on a Web site.

CGI programming differs from "regular" programming in that the CGI programs and scripts must be able to generate a Content-type: text/html MIME type to produce HTML output.

***Example 46.11** A Simple CGI Script in Perl*

```
#!/path/to/perl
print "Content-type: text/html\n\n";
print "Hello, World.";
```

The difference between modules specifically bound to a programming language (such as mod_php5) and mod_cgi lies in the possibility of combining mod_cgi with mod_suexec (see Section “Running CGIs as a Different User with mod_suexec ” (page 704)). This combination allows CGI scripts to be executed with a specified user ID. Usually, scripts using mod_cgi alone or mod_php5 are executed with the user ID of the Apache user (default in SUSE Linux: wwwrun). Modules designed for a programming language (such as mod_php5 or mod_ruby) embed a persistent interpreter in Apache to execute scripts under the Apache user ID.

As a consequence, CGIs with mod_suexec aid in administrative clarity as the CGI processes can be assigned to individual users instead of the Web server itself. Also, better file system security is accounted for with this combination: the script inherits only the user's file system rights. In the contrary case of modules, the script is granted the Web server user's file permissions, which can lead to unintended visibility of data in the file system.

CGIs are terminated when the request of a client to the Web server ends. This means that CGIs are not persistent and release all occupied resources after termination. This is an advantage, especially in case of erroneous programming. With modules, the effects of programming errors can accumulate, because the interpreter is persistent. This may result in a failure to release resources, such as database connections, and can require an Apache restart.

To use mod_cgi in SUSE Linux, either add cgi to APACHE_MODULES in /etc/sysconfig/apache2 or use YaST as described in Modules (page 682). The default directory for CGIs in SUSE Linux is /srv/www/cgi-bin/.

If manually editing the Apache configuration file, use this example as a guideline for configuring mod_cgi.

Example 46.12 *Manual Activaton of mod_cgi*

```
# Global Environment
LoadModule cgi_module /path/to/mod_cgi.so

# Main Server and/or Virtual Host and/or
# Directory and/or .htaccess context
AddHandler cgi-script .cgi .pl

# Main Server and/or Virtual Host context
ScriptAlias /cgi-bin/ /srv/www/cgi-bin/

# Alternatively, explicitly allow CGI scripts in a directory
# Main Server and/or Virtual Host context
<Directory /srv/www/some/dir>
    Options +ExecCGI
<Directory>
```

46.5.2 Extension Modules

In general, modules labeled as extensions are included in the Apache software package, but are usually not compiled into the server statically. In SUSE Linux they are available as shared objects that can be loaded into Apache at runtime.

Running CGIs as a Different User with `mod_suexec`

In combination with `mod_cgi` (Section "Common Gateway Interface: `mod_cgi` " (page 702)), `mod_suexec` allows CGI scripts to run as a specified user and group. The suEXEC program at `/usr/sbin/suexec2` is used for that purpose. It is a wrapper called by Apache every time a CGI script or program is executed. Both wrapper and program then get the configured user and group ID assigned. This results in it being run as the configured user or group.

While this approach considerably reduces the security risk involved with user-generated CGI scripts, it also has some important considerations:

Considerations for suEXEC Usage

- `suEXEC docroot`—All execution of scripts is limited to this base directory. This means that running scripts with suexec outside of the `docroot` is not possible and results in an error. `docroot` is set at `suEXEC` compile time and cannot be changed at runtime. The default in SUSE Linux is `/srv/www`.

- `uidmin`—This represents the minimum ID a user must have to be used to execute scripts with `suEXEC`. This prevents scripts from being excuted as system users such as `root`. Do not create users with an ID lower than `uidmin` if they should be used with `mod_suexec`. The default `uidmin` in SUSE Linux is 96.

- `gidmin`—This is the same concept as `uidmin`, but for the group ID. The default `gidmin` in SUSE Linux is 96.

- Directory and File Permissions—The script in question must be owned by the same user and belong to the same group as specified as the `suEXEC` user and group. Additionally, the file must not be writable by any except the owner. The directory the script resides in must also be only writable by the owner.

- suEXEC safepath—All programs used in a script (such as Perl) must reside in the paths labeled as safe for suexec. safepath is set at suEXEC compile time and cannot be changed at runtime. The default safepath in SUSE Linux is /usr/local/bin:/usr/bin:/bin.

In case of errors caused by mod_suexec, consult the suexec log file at /var/log/apache2/suexec.log.

To use mod_suexec in SUSE Linux, either add suexec to APACHE_MODULES in /etc/sysconfig/apache2 or use YaST as described in Modules (page 682). Keep in mind that mod_cgi is needed to run suexec.

mod_suexec is most useful when applied in a virtual host environment, described in Section 46.4, “Virtual Hosts” (page 696). To specify a certain user and group as which to run CGI scripts, use the following syntax in the file holding the virtual host declarations (default in SUSE Linux is /etc/apache2/vhosts.d/*):

Example 46.13 *mod_suexec Configuration*

```
<VirtualHost 192.168.0>
# ...
ScriptAlias /cgi-bin/ /srv/www/vhosts/www.example.com/cgi-bin/
SuexecUserGroup tux users
# ...
</VirtualHost>
```

The SuexecUserGroup *username group* syntax in this example assigns all scripts residing in /srv/www/vhosts/www.example.com/cgi-bin/ the user ID of tux and group ID of users.

Secure Sockets Layer and Apache: mod_ssl

mod_ssl provides strong encryption using the secure sockets layer (SSL) and transport layer security (TLS) protocols for HTTP communication between a client and the Web server. For this purpose, the server sends an SSL certificate that holds information proving the server's valid identity before any request to a URL is answered. In turn, this guarantees that the server is the uniquley correct end point for the communication. Additionally, the certificate generates an encrypted connection between client and server that can transport information without the risk of exposing sensitive, plain-text content. The most visible effect of using mod_ssl with Apache is that URLs are prefixed with https:// instead of http://.

The default port for SSL and TLS requests on the Web server side is 443. There is no conflict between a “regular” Apache listening on port 80 and an SSL/TLS-enabled Apache listening on port 443. In fact, HTTP and HTTPS can be run with the same Apache instance. Usually one virtual host (see Section 46.4, “Virtual Hosts” (page 696)) is used to dispatch requests to port 80 and port 443 to separate virtual servers.

IMPORTANT: Name-Based Virtual Hosts and SSL

It is not possible to run multiple SSL-enabled virtual hosts on a server with only one IP address. Users connecting to such a setup will receive a warning message stating that the certificate does not match the server name every time they visit the URL. A separate IP address or port is necessary for every SSL-enabled domain to achieve communication based on a valid SSL certificate.

Despite the warning message, you still get the same level of encryption that you would have on any valid SSL site. This means that as long as the warning message is acceptable, communication between Web server and client is still secure. The concept of uniquely knowing the server's identity, which is guaranteed by a valid SSL certificate, is forfeited.

To activate `mod_ssl` in SUSE Linux, either add `ssl` to `APACHE_MODULES` in `/etc/sysconfig/apache2` or use YaST as described in Modules (page 682). Additionally, the Web server must be configured to listen on the standard HTTPS port 443. This can be done manually in `/etc/apache2/listen.conf` or in YaST via the *Listen* menu entry (see Network Device Selection (page 681)).

A test SSL certificate can be created by entering `cd /usr/share/doc/packages/apache2; ./certificate.sh` as `root`. Follow the on-screen instructions to build the SSL certificate. The resulting certificate files reside in the directories `/etc/apache2/ssl*`.

A “real” certificate with global validity can be obtained from vendors such as Thawte (`http://www.thawte.com/` or Verisign (`www.verisign.com`).

If manually editing the Apache configuration file, use this example as a guideline for configuring `mod_ssl`.

Example 46.14 *Manual Configuration of mod_ssl*

```
# Global Environment
# listen on the standard SSL port
Listen 443
# load module only if rcapache2 start-ssl was issued
<IfDefine SSL>
LoadModule ssl_module /path/to/mod_ssl.so
</IfDefine>

# Main Server context
# include global (server-wide) SSL configuration
# that is not specific to any virtual host
# only if ssl_module was loaded
<IfModule mod_ssl.c>
Include /etc/apache2/ssl-global.conf
</IfModule>
```

TIP

Do not forget to open the firewall for SSL-enabled Apache on port 443. This can be done via YaST by going to *Security and Users* → *Firewall* → *Allowed Services*. Then add *HTTPS Server* to the list of *Allowed Services*.

46.5.3 External Modules

Officially, modules labeled external are not included in the Apache distribution. However, SUSE Linux provides several of them readily available for usage. This chapter briefly explains some external modules and their functionality.

Using Perl to Manage Apache: `mod_perl`

`mod_perl` embeds a persistent Perl interpreter in Apache. This avoids the overhead caused by a `mod_cgi` that calls an external executable on every request to a CGI. `mod_perl` additionally allows controlling many aspects of Apache functionality with the help of the Perl programming language.

To use `mod_perl` in SUSE Linux, install the `apache2-mod_perl` RPM and activate the module either via YaST (Modules (page 682)) or manually in `/etc/sysconfig/apache2`. After installation and activation, a separate configuration file, `mod_perl.conf`, is placed in `/etc/apache2/conf.d/`. Additionally, the `mod_perl` startup script is installed as `mod_perl-startup.pl`. For more information about how to use the module, consult the documentation available on the mod_perl Web site (http://perl.apache.org/).

Serving PHP: mod_php4, mod_php5

PHP is a popular programming language originally geared towards usage on the Web. It exists in two versions, PHP4 and PHP5. While PHP4 represents the classic concept of and approach to PHP, PHP5 has introduced new object-oriented programming possibilities along with many other advanced features. Both mod_php4 and mod_php5 are available in SUSE Linux. They embed the PHP interpreter into Apache as a persistent module.

To use mod_php4 or mod_php5 in SUSE Linux, install the respective RPM (apache2-mod_php4, apache2-mod_php5) and activate the module either via YaST (Modules (page 682)) or manually in /etc/sysconfig/apache2.

After installation and activation, a separate configuration file for the respective module (either php4.conf or php5.conf) is placed in /etc/apache2/conf.d/. The PHP Web site (http://www.php.net) is an excellent resource for using Apache together with PHP.

Python and Apache: mod_python

mod_python embeds the Python interpreter into Apache. Python is an object-oriented programming language with a very clear and legible syntax. An unusual but convenient feature is that the program structure depends on the source code indentation rather than regular demarcation elements such as begin and end.

To use mod_python in SUSE Linux, install the apache2-mod_python RPM and activate the module either via YaST (Modules (page 682)) or manually in /etc/sysconfig/apache2. For more information about how to use the module, consult the documentation available on the mod_python Web site (http://www.modpython.org/).

Ruby Interpreter in Apache: mod_ruby

mod_ruby embeds the Ruby interpreter in the Apache Web server, allowing Ruby CGI scripts to be executed natively. Ruby is a relatively new, object-oriented high-level programming language that resembles certain aspects of Perl and Python. Like Python, it has a clean, transparent syntax. On the other hand, Ruby has adopted abbreviations (such as `$.r` for the number of the last line read in the input file) that are appreciated by some programmers and disliked by others. The basic concept of Ruby closely resembles that of Smalltalk.

To use `mod_ruby` in SUSE Linux, install the `apache2-mod_ruby` RPM and activate the module either via YaST (Modules (page 682)) or manually in `/etc/sysconfig/apache2`. For more information about how to use the module, consult the documentation available on the mod_ruby Web site (`http://www.modruby.net/en/index.rbx`).

Native File System Access: `mod_dav`

`mod_dav` provides WebDAV (Web-Based Distributed Authoring and Versioning) functionality for Apache. WebDAV is an extension of the HTTP protocol that allows users to collaboratively edit and manage files on remote servers. WebDAV's capabilities are similar to those of FTP with the major difference that HTTP is used as the underlying protocol for server access. In effect, `mod_dav` makes an Apache Web server an advanced remote file system.

It is good practice, if not required, to limit access to the directories available via WebDAV. The minimum precautions to take are to set up HTTP basic authentication for the WebDAV resource, along with Limit clauses inside a `Location` directive.

To access a WebDAV resource, WebDAV-capable software needs to be present on the client side. SUSE Linux already comes with WebDAV capabilities: `Konqueror` with the prefix `webdav://` or `webdavs://` (for WebDAV over SSL connections) can be used to connect to an Apache WebDAV file system.

`mod_dav` requires the module `mod_dav_fs`, which provides the actual file system access for WebDAV. To use `mod_dav` in SUSE Linux, activate the module either via YaST (Modules (page 682)) or manually in `/etc/sysconfig/apache2`. Do the same for `mod_dav_fs`. For more information about how to use the module, consult the documentation available on the mod_dav Web site (`http://httpd.apache.org/docs-2.0/mod/mod_dav.html`).

Offering User Home Pages: mod_userdir

mod_userdir in SUSE Linux defaults to offering the contents of each user's ~/public_html folder as public Web pages. The URL to access those pages then is http://www.example.com/~*username*/.

TIP

mod_userdir in SUSE Linux forbids access to any directories of the root user's home directory for security reasons. Additionally you can specifically allow only certain users to have public home pages by using:

```
# Main server context
UserDir disabled
UserDir enabled tux wilber
```

To use mod_userdir in SUSE Linux, activate the module either via YaST (Modules (page 682)) or manually in /etc/sysconfig/apache2. For more information about how to use the module, consult the documentation available on the mod_userdir Web site (http://httpd.apache.org/docs-2.0/mod/mod_userdir.html).

Changing URL Layout: mod_rewrite

mod_rewrite is often referred to as “the Swiss army knife of URL manipulation.” It rewrites requested URLs on the fly based on a specified rule set. The result typically looks similar to http://www.example.com/2/1/de for http://www.example.com/display.php?cat=2&article=1&lang=de.

The URL Rewriting Guide explains the advantages and disadvantages of the powerful but complex module:

“With mod_rewrite you either shoot yourself in the foot the first time and never use it again or love it for the rest of your life because of its power.”

RewriteRule sets can be set in all configuration contexts: for the main server, for virtual hosts, for directories, and for .htaccess files. A good starting point for URL rewriting with mod_rewrite is URL Rewriting Guide at http://httpd.apache.org/docs-2.0/misc/rewriteguide.html.

To use mod_rewrite in SUSE Linux, activate the module either via YaST (Modules (page 682)) or manually in /etc/sysconfig/apache2.

46.6 Security

A Web server exposed to the public Internet requires an ongoing administrative effort. It is inevitable that security issues appear, both related to the software and to accidental misconfiguration. Here are some tips for how to deal with them.

Staying Up to Date

In case there are vulnerabilities found in the Apache software, a security advisory will be issued by SUSE. It contains instructions for fixing the vulnerabilities, which in turn should be applied soon as possible. The SUSE security announcement mailing list is available at http://www.suse.com/us/private/support/online_help/mailinglists/. The latest information about security issues for the SUSE Linux packages are also available online at http://www.novell.com/linux/security/securitysupport.html.

Additionally, you should subscribe to the Apache announcement mailing list (http://httpd.apache.org/lists.html#http-announce where new releases and bug fixes are posted.

DocumentRoot Permissions

By default in SUSE Linux, the DocumentRoot directory /srv/www/htdocs and the CGI directory /srv/www/cgi-bin belong to the user root. You should not change these permissions. If the directories were writable for all, any user could place files into them. These files might then be executed by Apache with the permissions of wwwrun which may give the user unintended access to file system resources. Use subdirectories of /srv/www/htdocs and /srv/www/cgi-bin to organize user or domain-specific data in combination with the Directory directive (see Directory (page 684)).

CGI and SSI Directories

Interactive scripts in Perl, PHP, SSI or any other programming language can essentially run arbitrary commands. Limiting the execution of CGIs and SSIs (see Section “Common Gateway Interface: mod_cgi ” (page 702), Script Alias (page 685), and Section “Server-Side Includes with mod_include ” (page 701)) to specific directories instead of globally allowing them is one option for decreasing the risk.

Another possibility is to work with mod_suexec (see Section "Running CGIs as a Different User with mod_suexec " (page 704)) for CGIs in general. For Apache modules, a security-concious configuration for the interpreters, such as in Section "Serving PHP: mod_php4, mod_php5 " (page 708), helps to keep the Web environment safe.

Access Permissions

Often times, especially in test environments, access permissions to a Web server are handled casually because of the nature of testing a setup. This may result in accidentally revealing sensitive information or even exposing an entire server to the wrong audience. Use the Order directive (http://httpd.apache.org/docs-2.0/mod/mod_access.html#order) in combination with .htaccess files (see Section "AccessFileName *filenames* " (page 692)) to allow access to certain Web sites only for specific users or clients.

Additionally, you could use the "security by obfuscation" approach: a typical example for this is to run Apache on a nonstandard port (see Network Device Selection (page 681)). This results in URLs with the port appended, such as http://www.example.com:8765, which is acceptable in test environments.

46.7 Troubleshooting

If Apache does not start, the Web page is not accessible, or users cannot connect to the Web server, it is important to find the cause of the problem. Here are some typical places to look for error explanations and important things to check.

First, rcapache2 (described in Section 46.3.3, "Activating, Starting, and Stopping Apache" (page 694)) is verbose about errors, so can be quite helpful if it is actually used for operating Apache. Sometimes it is tempting to use the binary /usr/sbin/httpd2 for starting or stopping the Web server. Avoid doing this and use the rcapache2 script instead. rcapache2 even provides tips and hints for solving configuration errors.

Second, the importance of log files (see Section "Log Files" (page 678)) cannot be overemphasized. In case of both fatal and nonfatal errors, the Apache log files are the places to look for causes. Additionally, you can control the verbosity of the logged messages with the LogLevel directive (see Section "LogLevel *level* " (page 693)) if more detail is needed in the log files.

TIP

Watch the Apache log messages with the command `tail -F /var/log/apache2/*_log &`. Then run `rcapache2 restart`. Now, try to connect with a browser and check the output.

A common mistake is not to open the ports for Apache in the firewall configuration of the server. If you configure Apache with YaST, there is a separate option available to take care of this specific issue (see Network Device Selection (page 681)).

If the error cannot be tracked down with the help of any these, check the online Apache bug database at `http://httpd.apache.org/bug_report.html`. Additionally, the Apache user community can be reached via a mailing list available at `http://httpd.apache.org/userslist.html`. A recommended newsgroup is `comp.infosystems.www.servers.unix`.

46.8 For More Information

Apache is a widely used Web server. As a consequence, there are many Web sites offering support and help for it in different levels of quality. In any case, the starting point for any research about Apache and its possibilities should be `http://httpd.apache.org/docs-2.0/`.

Additionally, the RPM package `apache2-doc` contains the Apache manual for local installation and reference. For some SUSE-specific configuration hints, the file `/usr/share/doc/packages/apache2` contains a quick reference.

The RPM package `apache2-example-pages` holds some example pages for Apache that show information about the Web server.

46.8.1 Apache Modules

More information about external Apache modules from Section 46.5.3, “External Modules” (page 707) is available at the following locations:

- `http://httpd.apache.org/docs-2.0/mod/`

- http://www.php.net/manual/en/install.unix.apache2.php
- http://www.modpython.org/
- http://www.modruby.net/
- http://perl.apache.org/

46.8.2 CGI

More information about using mod_cgi (see Section "Common Gateway Interface: mod_cgi " (page 702)) and CGI programming is available at the following locations:

- http://www.modperl.com/
- http://www.modperlcookbook.org/
- http://www.fastcgi.com/
- http://www.boutell.com/cgic/

46.8.3 Miscellaneous Sources

If you experience difficulties specific to Apache in SUSE Linux, take a look at the SUSE Support Database at http://portal.suse.com/sdb/en/index.html.

The history of Apache is provided at http://httpd.apache.org/ABOUT_APACHE.html. This page also explains why the server is called Apache.

Information about upgrading from version 1.3 to 2.0 is available at http://httpd.apache.org/docs-2.0/en/upgrading.html.

File Synchronization 47

Today, many people use several computers—one computer at home, one or several computers at the workplace, and possibly a laptop or PDA on the road. Many files are needed on all these computers. You may want to be able to work with all computers and modify the files and subsequently have the latest version of the data available on all computers.

47.1 Available Data Synchronization Software

Data synchronization is no problem for computers that are permanently linked by means of a fast network. In this case, use a network file system, like NFS, and store the files on a server, enabling all hosts to access the same data via the network. This approach is impossible if the network connection is poor or not permanent. When you are on the road with a laptop, copies of all needed files must be on the local hard disk. However, it is then necessary to synchronize modified files. When you modify a file on one computer, make sure a copy of the file is updated on all other computers. For occasional copies, this can be done manually with scp or rsync. However, if many files are involved, the procedure can be complicated and requires great care to avoid errors, such as overwriting a new file with an old file.

WARNING: Risk of Data Loss

Before you start managing your data with a synchronization system, you should be well acquainted with the program used and test its functionality. A backup is indispensable for important files.

The time-consuming and error-prone task of manually synchronizing data can be avoided by using one of the programs that use various methods to automate this job. The following summaries are merely intended to convey a general understanding of how these programs work and how they can be used. If you plan to use them, read the program documentation.

47.1.1 Unison

Unison is not a network file system. Instead, the files are simply saved and edited locally. The program Unison can be executed manually to synchronize files. When the synchronization is performed for the first time, a database is created on the two hosts, containing checksums, time stamps, and permissions of the selected files. The next time it is executed, Unison can recognize which files were changed and propose transmission from or to the other host. Usually all suggestions can be accepted.

47.1.2 CVS

CVS, which is mostly used for managing program source versions, offers the possibility to keep copies of the files on multiple computers. Accordingly, it is also suitable for data synchronization. CVS maintains a central repository on the server in which the files and changes to files are saved. Changes that are performed locally are committed to the repository and can be retrieved from other computers by means of an update. Both procedures must be initiated by the user.

CVS is very resilient to errors when changes occur on several computers. The changes are merged and, if changes took place in the same lines, a conflict is reported. When a conflict occurs, the database remains in a consistent state. The conflict is only visible for resolution on the client host.

47.1.3 subversion

In contrast to CVS, which “evolved,” subversion is a consistently designed project. subversion was developed as a technically improved successor to CVS.

subversion has been improved in many respects to its predecessor. Due to its history, CVS only maintains files and is oblivious of directories. Directories also have a version history in subversion and can be copied and renamed just like files. It is also possible to add metadata to every file and to every directory. This metadata can be fully maintained with versioning. As opposed to CVS, subversion supports transparent network access over dedicated protocols, like WebDAV (Web-based Distributed Authoring and Versioning). WebDAV extends the functionality of the HTTP protocol to allow collaborative write access to files on remote Web servers.

subversion was largely assembled on the basis of existing software packages. Therefore, the Apache Web server and the WebDAV extension always run in conjunction with subversion.

47.1.4 mailsync

Unlike the synchronization tools covered in the previous sections, mailsync only synchronizes e-mails between mailboxes. The procedure can be applied to local mailbox files as well as to mailboxes on an IMAP server.

Based on the message ID contained in the e-mail header, the individual messages are either synchronized or deleted. Synchronization is possible between individual mailboxes and between mailbox hierarchies.

47.1.5 rsync

When no version control is needed but large directory structures need to be synchronized over slow network connections, the tool rsync offers well-developed mechanisms for transmitting only changes within files. This not only concerns text files, but also binary files. To detect the differences between files, rsync subdivides the files into blocks and computes checksums over them.

The effort put into the detection of the changes comes at a price. The systems to synchronize should be scaled generously for the usage of rsync. RAM is especially important.

47.2 Determining Factors for Selecting a Program

47.2.1 Client-Server versus Peer-to-Peer

Two different models are commonly used for distributing data. In the first model, all clients synchronize their files with a central server. The server must be accessible by all clients at least occasionally. This model is used by subversion, CVS, and WebDAV.

The other possibility is to let all networked hosts synchronize their data between each other as peers. This is the concept followed by unison. rsync actually works in client mode, but any client can also act as a server.

47.2.2 Portability

subversion, CVS, and unison are also available for many other operating systems, including various Unix and Windows systems.

47.2.3 Interactive versus Automatic

In subversion, CVS, WebDAV, and Unison, the data synchronization is started manually by the user. This allows fine control over the data to synchronize and easy conflict handling. However, if the synchronization intervals are too long, conflicts are more likely to occur.

47.2.4 Conflicts: Incidence and Solution

Conflicts only rarely occur in subversion or CVS, even when several people work on one large program project. This is because the documents are merged on the basis of individual lines. When a conflict occurs, only one client is affected. Usually conflicts in subversion or CVS can easily be resolved.

Unison reports conflicts, allowing the affected files to be excluded from the synchronization. However, changes cannot be merged as easily as in subversion or CVS.

As opposed to subversion or CVS, where it is possible to partially accept changes in cases of conflict, WebDAV only performs a check-in when the complete modification is considered successful.

There is no conflict handling in rsync. The user is responsible for not accidentally overwriting files and manually resolving all possible conflicts. To be on safe side, a versioning system like RCS can be additionally employed.

47.2.5 Selecting and Adding Files

In its standard configuration, Unison synchronizes an entire directory tree. New files appearing in the tree are automatically included in the synchronization.

In subversion or CVS, new directories and files must be added explicitly using the command `svn add` or `cvs add`, respectively. This results in greater user control over the files to synchronize. On the other hand, new files are often overlooked, especially when the question marks in the output of `svn update` and `svn status` or `cvs update` are ignored due to the large number of files.

47.2.6 History

An additional feature of subversion or CVS is that old file versions can be reconstructed. A brief editing remark can be inserted for each change and the development of the files can easily be traced later based on the content and the remarks. This is a valuable aid for theses and program texts.

47.2.7 Data Volume and Hard Disk Requirements

A sufficient amount of free space for all distributed data is required on the hard disks of all involved hosts. subversion and CVS require additional space for the repository database on the server. The file history is also stored on the server, requiring even more space. When files in text format are changed, only the modified lines need to be saved. Binary files require additional space amounting to the size of the file every time the file is changed.

47.2.8 GUI

Unison offers a graphical user interface that displays the synchronization procedures Unison wants to perform. Accept the proposal or exclude individual files from the synchronization. In text mode, interactively confirm the individual procedures.

Experienced users normally run subversion or CVS from the command line. However, graphical user interfaces are available for Linux, such as cervisia, and for other operating systems, like wincvs. Many development tools, such as kdevelop, and text editors, such as emacs, provide support for CVS or subversion. The resolution of conflicts is often much easier to perform with these front-ends.

47.2.9 User Friendliness

Unison and rsync are rather easy to use and are also suitable for newcomers. CVS and subversion are somewhat more difficult to operate. Users should understand the interaction between the repository and local data. Changes to the data should first be merged locally with the repository. This is done with the command `cvs update` or `svn update`. Then the data must be sent back to the repository with the command `cvs commit` or `svn commit`. Once this procedure has been understood, newcomers are also able to use CVS or subversion with ease.

47.2.10 Security against Attacks

During transmission, the data should ideally be protected against interception and manipulation. Unison, CVS, rsync, and subversion can easily be used via ssh (secure shell), providing security against attacks of this kind. Running CVS or Unison via rsh (remote shell) should be avoided. Accessing CVS with the *pserver* mechanism in insecure networks is likewise not advisable. subversion already provides the necessary security measures by running with Apache.

47.2.11 Protection against Data Loss

CVS has been used by developers for a long time to manage program projects and is extremely stable. Because the development history is saved, CVS even provides protection against certain user errors, such as unintentional deletion of a file. Despite subversion not being as common as CVS, it is already being employed in productive environments, for example, by the subversion project itself.

Unison is still relatively new, but boasts a high level of stability. However, it is more sensitive to user errors. Once the synchronization of the deletion of a file has been confirmed, there is no way to restore the file.

Table 47.1 *Features of the File Synchronization Tools: -- = very poor, - = poor or not available, o = medium, + = good, ++ = excellent, x = available*

	unison	**CVS/subv.**	**rsync**	**mailsync**
Client/Server	equal	C-S/C-S	C-S	equal
Portability	Lin,Un*x,Win	Lin,Un*x,Win	Lin,Un*x,Win	Lin,Un*x
Interactivity	x	x/x	x	-
Speed	-	o/+	+	+
Conflicts	o	++/++	o	+
File Sel.	Dir.	Sel./file, dir.	Dir.	Mailbox

	unison	CVS/subv.	rsync	mailsync
History	-	x/x	-	-
Hard Disk Space	o	--	o	+
GUI	+	o/o	-	-
Difficulty	+	o/o	+	o
Attacks	+(ssh)	+/+(ssh)	+(ssh)	+(SSL)
Data Loss	+	++/++	+	+

47.3 Introduction to Unison

Unison is an excellent solution for synchronizing and transferring entire directory trees. The synchronization is performed in both directions and can be controlled by means of an intuitive graphical front-end. A console version can also be used. The synchronization can be automated so interaction with the user is not required, but experience is necessary.

47.3.1 Requirements

Unison must be installed on the client as well as on the server. In this context, the term *server* refers to a second, remote host (unlike CVS, explained in Section 47.1.2, "CVS" (page 716)).

In the following section, Unison is used together with ssh. In this case, an SSH client must be installed on the client and an SSH server must be installed on the server.

47.3.2 Using Unison

The approach used by Unison is the association of two directories (*roots*) with each other. This association is symbolic—it is not an online connection. In this example, the directory layout is as follows:

Client:	`/home/tux/dir1`
Server:	`/home/geeko/dir2`

You want to synchronize these two directories. The user is known as tux on the client and as `geeko` on the server. The first thing to do is to test if the client-server communication works:

```
unison -testserver /home/tux/dir1 ssh://geeko@server//homes/geeko/dir2
```

The most frequently encountered problems are:

- The Unison versions used on the client and server are not compatible.
- The server does not allow SSH connections.
- Neither of the two specified paths exists.

If everything works, omit the option `-testserver`. During the first synchronization, Unison does not yet know the relationship between the two directories and submits suggestions for the transfer direction of the individual files and directories. The arrows in the *Action* column indicate the transfer direction. A question mark means that Unison is not able to make a suggestion regarding the transfer direction because both versions were changed or are new.

The arrow keys can be used to set the transfer direction for the individual entries. If the transfer directions are correct for all displayed entries, simply click *Go*.

The characteristics of Unison (for example, whether to perform the synchronization automatically in clear cases) can be controlled by means of command-line parameters specified when the program is started. View the complete list of all parameters with `unison --help`.

Example 47.1 *The file ~/.unison/example.prefs*

```
root=/home/tux/dir1
root=ssh://wilber@server//homes/wilber/dir2
batch=true
```

For each pair, a synchronization log is maintained in the user directory `~/.unison`. Configuration sets, such as `~/.unison/example.prefs`, can also be stored in this directory. To start the synchronization, specify this file as the command-line parameter as in `unison example.prefs`.

47.3.3 For More Information

The official documentation of Unison is extremely useful. For this reason, this section merely provides a brief introduction. The complete manual is available at `http://www.cis.upenn.edu/~bcpierce/unison/` and in the SUSE package `unison`.

47.4 Introduction to CVS

CVS is suitable for synchronization purposes if individual files are edited frequently and are stored in a file format, such as ASCII text or program source text. The use of CVS for synchronizing data in other formats, such as JPEG files, is possible, but leads to large amounts of data, because all variants of a file are stored permanently on the CVS server. In such cases, most of the capabilities of CVS cannot be used. The use of CVS for synchronizing files is only possible if all workstations can access the same server.

47.4.1 Configuring a CVS Server

The *server* is the host on which all valid files are located, including the latest versions of all files. Any stationary workstation can be used as a server. If possible, the data of the CVS repository should be included in regular backups.

When configuring a CVS server, it might be a good idea to grant users access to the server via SSH. If the user is known to the server as `tux` and the CVS software is installed on the server as well as on the client, the following environment variables must be set on the client side:

```
CVS_RSH=ssh CVS_ROOT=tux@server:/serverdir
```

The command `cvs init` can be used to initialize the CVS server from the client side. This needs to be done only once.

Finally, the synchronization must be assigned a name. Select or create a directory on the client exclusively to contain files to manage with CVS (the directory can also be empty). The name of the directory is also the name of the synchronization. In this example, the directory is called `synchome`. Change to this directory and enter the following command to set the synchronization name to `synchome`:

```
cvs import synchome tux wilber
```

Many CVS commands require a comment. For this purpose, CVS starts an editor (the editor defined in the environment variable `$EDITOR` or vi if no editor was defined). The editor call can be circumvented by entering the comment in advance on the command line, such as in the following example:

```
cvs import -m 'this is a test' synchome tux wilber
```

47.4.2 Using CVS

The synchronization repository can now be checked out from all hosts with `cvs co synchome`. This creates a new subdirectory `synchome` on the client. To commit your changes to the server, change to the directory `synchome` (or one of its subdirectories) and enter `cvs commit`.

By default, all files (including subdirectories) are committed to the server. To commit only individual files or directories, specify them as in `cvs commit file1 directory1`. New files and directories must be added to the repository with a command like `cvs add file1 directory1` before they are committed to the server. Subsequently, commit the newly added files and directories with `cvs commit file1 directory1`.

If you change to another workstation, check out the synchronization repository, if this has not been done during an earlier session at the same workstation (see above).

Start the synchronization with the server with `cvs update`. Update individual files or directories as in `cvs update file1 directory1`. To see the difference between the current files and the versions stored on the server, use the command `cvs diff` or `cvs diff file1 directory1`. Use `cvs -nq update` to see which files would be affected by an update.

Here are some of the status symbols displayed during an update:

U
: The local version was updated. This affects all files that are provided by the server and missing on the local system.

M
: The local version was modified. If there were changes on the server, it was possible to merge the differences in the local copy.

P
: The local version was patched with the version on the server.

C
: The local file conflicts with current version in the repository.

?
: This file does not exist in CVS.

The status `M` indicates a locally modified file. Either commit the local copy to the server or remove the local file and run the update again. In this case, the missing file is retrieved from the server. If you commit a locally modified file and the file was changed in the same line and committed, you might get a conflict, indicated with `C`.

In this case, look at the conflict marks (»> and «<) in the file and decide between the two versions. As this can be a rather unpleasant job, you might decide to abandon your changes, delete the local file, and enter `cvs up` to retrieve the current version from the server.

47.4.3 For More Information

This section merely offers a brief introduction to the many possibilities of CVS. Extensive documentation is available at the following URLs:

```
http://www.cvshome.org/
http://www.gnu.org/manual/
```

47.5 Introduction to Subversion

Subversion is a free open source versioning control system and is widely regarded as the successor to CVS, meaning that features already introduced for CVS are normally also in subversion. It is especially recommended when the advantages of CVS are sought without having to put up with its disadvantages. Many of these features have already been briefly introduced in Section 47.1.3, “subversion” (page 717).

47.5.1 Installing a Subversion Server

The installation of a repository database on a server is a relatively simple procedure. Subversion provides a dedicated administration tool for this purpose. The command to enter for creating a new repository is:

```
svnadmin create /path/to/repository
```

Other options can be listed with `svnadmin help`. As opposed to CVS, subversion is not based on RCS, but rather on the Berkeley Database. Make sure not to install a repository on remote file systems, like NFS, AFS, or Windows SMB. The database requires POSIX locking mechanisms, which these file systems do not support.

The command `svnlook` provides information about an existing repository.

```
svnlook info /path/to/repository
```

A server must be configured to allow different users to access the repository. Either use the Apache Web server with WebDAV to do this or use svnserve, the server packaged with subversion. Once svnserve is up and running, the repository can be accessed with a URL with `svn://` or `svn+ssh://`. Users that should authenticate themselves when calling `svn` can be set in `/etc/svnserve.conf`.

A decision for Apache or for svnserve depends on many factors. It is recommended to browse the subversion book. More information about it can be found in Section 47.5.3, “For More Information” (page 730).

47.5.2 Usage and Operation

Use the command `svn` (similar to `cvs`) to access a subversion repository. The content provided by a correctly configured server fitted with a corresponding repository can be accessed by any client with one of the following commands:

```
svn list http://svn.example.com/path/to/project
```

or

```
svn list svn://svn.example.com/path/to/project
```

Save an existing project in the current directory (check it out) with the command `svn checkout`:

```
svn checkout http://svn.example.com/path/to/project nameofproject
```

Checking out creates a new subdirectory `nameofproject` on the client. Operations (adding, copying, renaming, deleting) can then be performed on it:

```
svn add file
svn copy oldfile newfile
svn move oldfile newfile
svn delete file
```

These commands can also be used on directories. subversion can additionally record properties of a file or directory:

```
svn propset license GPL foo.txt
```

The preceding example sets the value `GPL` for the property `license`. Display properties with `svn proplist`:

```
svn proplist --verbose foo.txt
 Properties on 'foo.txt':
 license : GPL
```

Save the changes to the server with `svn commit` Another user can incorporate your changes in his working directory by synchronizing with the server using `svn update`.

Unlike CVS, the status of a working directory in subversion can be displayed *without* accessing the repository with `svn status`. Local changes are displayed in five columns, with the first one being the most important one:

''

    No changes.

'A'

Object is marked for addition.

'D'

Object is marked for deletion.

'M'

Object was modified.

'C'

Object is in conflict.

'I'

Object was ignored.

'?'

Object is not being maintained by versioning control.

'!'

Object is reported missing. This flag appears when the object was deleted or moved without the `svn` command.

'~'

Object was being maintained as a file but has since been replaced by a directory or the opposite has occurred.

The second column shows the status of properties. The meaning of all other columns can be read in the subversion book.

Use the command `svn help` to obtain the description of a parameter of a command:

```
svn help proplist
proplist (plist, pl): List all properties on files, dirs, or revisions.
usage: 1. proplist [PATH...]
       2. proplist --revprop -r REV [URL]

  1. Lists versioned props in working copy.
  2. Lists unversioned remote props on repos revision.
...
```

47.5.3 For More Information

The first point of reference is the home page of the subversion project at http://subversion.tigris.org/. A highly recommendable book can be found in the directory file:///usr/share/doc/packages/subversion/html/book.html after installation of the package subversion-doc and is also available online at http://svnbook.red-bean.com/svnbook/index.html.

47.6 Introduction to rsync

rsync is useful when large amounts of data need to be transmitted regularly while not changing too much. This is, for example, often the case when creating backups. Another application concerns staging servers. These are servers that store complete directory trees of Web servers that are regularly mirrored onto a Web server in a DMZ.

47.6.1 Configuration and Operation

rsync can be operated in two different modes. It can be used to archive or copy data. To accomplish this, only a remote shell, like ssh, is required on the target system. However, rsync can also be used as a daemon to provide directories to the network.

The basic mode of operation of rsync does not require any special configuration. rsync directly allows mirroring complete directories onto another system. As an example, the following command creates a backup of the home directory of tux on a backup server named sun:

```
rsync -baz -e ssh /home/tux/ tux@sun:backup
```

The following command is used to play the directory back:

```
rsync -az -e ssh tux@sun:backup /home/tux/
```

Up to this point, the handling does not differ much from that of a regular copying tool, like scp.

rsync should be operated in “rsync” mode to make all its features fully available. This is done by starting the rsyncd daemon on one of the systems. Configure it in the file /etc/rsyncd.conf. For example, to make the directory /srv/ftp available with rsync, use the following configuration:

```
gid = nobody
uid = nobody
read only = true
use chroot = no
transfer logging = true
log format = %h %o %f %l %b
log file = /var/log/rsyncd.log

[FTP]
        path = /srv/ftp
        comment = An Example
```

Then start rsyncd with `rcrsyncd start`. rsyncd can also be started automatically during the boot process. Set this up by activating this service in the runlevel editor provided by YaST or by manually entering the command `insserv rsyncd`. rsyncd can alternatively be started by xinetd. This is, however, only recommended for servers that rarely use rsyncd.

The example also creates a log file listing all connections. This file is stored in `/var/log/rsyncd.log`.

It is then possible to test the transfer from a client system. Do this with the following command:

```
rsync -avz sun::FTP
```

This command lists all files present in the directory `/srv/ftp` of the server. This request is also logged in the log file `/var/log/rsyncd.log`. To start an actual transfer, provide a target directory. Use `.` for the current directory. For example:

```
rsync -avz sun::FTP .
```

By default, no files are deleted while synchronizing with rsync. If this should be forced, the additional option `--delete` must be stated. To ensure that no newer files are deleted, the option `--update` can be used instead. Any conflicts that arise must be resolved manually.

47.6.2 For More Information

Important information about rsync is provided in the man pages `man rsync` and `man rsyncd.conf`. A technical reference about the operating principles of rsync is featured in `/usr/share/doc/packages/rsync/tech_report.ps`. Find latest news about rsync on the project Web site at http://rsync.samba.org/.

47.7 Introduction to mailsync

mailsync is mainly suitable for the following three tasks:

- Synchronization of locally stored e-mails with mails stored on a server
- Migration of mailboxes to a different format or to a different server
- Integrity check of a mailbox or search for duplicates

47.7.1 Configuration and Use

mailsync distinguishes between the mailbox itself (the *store*) and the connection between two mailboxes (the *channel*). The definitions of the stores and channels are stored in `~/.mailsync`. The following paragraphs explain a number of store examples.

A simple definition might appear as follows:

```
store saved-messages {
   pat Mail/saved-messages
prefix  Mail/
}
```

`Mail/` is a subdirectory of the user's home directory that contains e-mail folders, including the folder `saved-messages`. If mailsync is started with `mailsync -m saved-messages`, it lists an index of all messages in `saved-messages`. If the following definition is made

```
store localdir {
pat     Mail/*
prefix  Mail/
}
```

the command `mailsync -m localdir` lists all messages stored under `Mail/`. In contrast, the command `mailsync localdir` lists the folder names. The specifications of a store on an IMAP server appear as follows:

```
store imapinbox {
server {mail.edu.harvard.com/user=gulliver}
ref    {mail.edu.harvard.com}
pat    INBOX
}
```

The above example merely addresses the main folder on the IMAP server. A store for the subfolders would appear as follows:

```
store imapdir {
server {mail.edu.harvard.com/user=gulliver}
ref {mail.edu.harvard.com}
pat INBOX.*
prefix  INBOX.
}
```

If the IMAP server supports encrypted connections, the server specification should be changed to

```
server {mail.edu.harvard.com/ssl/user=gulliver}
```

or, if the server certificate is not known, to

```
server {mail.edu.harvard.com/ssl/novalidate-cert/user=gulliver}
```

The prefix is explained later.

Now the folders under `Mail/` should be connected to the subdirectories on the IMAP server:

```
channel folder localdir imapdir {
msinfo .mailsync.info
}
```

mailsync uses the `msinfo` file to keep track of the messages that have already been synchronized.

The command `mailsync folder` does the following:

- Expands the mailbox pattern on both sides.
- Removes the prefix from the resulting folder names.
- Synchronizes the folders in pairs (or creates them if they do not exist).

Accordingly, the folder `INBOX.sent-mail` on the IMAP server is synchronized with the local folder `Mail/sent-mail` (provided the definitions explained above exist). The synchronization between the individual folder is performed as follows:

- If a message already exists on both sides, nothing happens.

- If the message is missing on one side and is new (not listed in the `msinfo` file), it is transmitted there.

- If the message merely exists on one side and is old (already listed in the `msinfo` file), it is deleted there (because the message that had obviously existed on the other side was deleted).

To know in advance which messages will be transmitted and which will be deleted during a synchronization, start mailsync with a channel *and* a store with `mailsync folder localdir`. This command produces a list of all messages that are new on the local host as well as a list of all messages that would be deleted on the IMAP side during a synchronization. Similarly, the command `mailsync folder imapdir` produces a list of all messages that are new on the IMAP side and a list of all messages that would be deleted on the local host during a synchronization.

47.7.2 Possible Problems

In the event of a data loss, the safest method is to delete the relevant channel log file `msinfo`. Accordingly, all messages that only exist on one side are viewed as new and are therefore transmitted during the next synchronization.

Only messages with a message ID are included in the synchronization. Messages lacking a message ID are simply ignored, which means they are not transmitted or deleted. A missing message ID is usually caused by faulty programs when sending or writing a message.

On certain IMAP servers, the main folder is addressed with `INBOX` and subfolders are addressed with a randomly selected name (in contrast to INBOX and INBOX.name). Therefore, for such IMAP servers, it is not possible to specify a pattern exclusively for the subfolders.

After the successful transmission of messages to an IMAP server, the mailbox drivers (c-client) used by mailsync set a special status flag. For this reason, some e-mail programs, like mutt, are not able to recognize these messages as new. Disable the setting of this special status flag with the option `-n`.

47.7.3 For More Information

The README in /usr/share/doc/packages/mailsync/, which is included in mailsync, provides additional information. In this connection, RFC 2076 “Common Internet Message Headers” is of special interest.

Samba 48

Using Samba, a Unix machine can be configured as a file and print server for DOS, Windows, and OS/2 machines. Samba has developed into a fully-fledged and rather complex product. In addition to describing the basic functionality, this chapter introduces the basics of the Samba configuration and describes the YaST modules you can use for configuring Samba in your network.

Detailed Samba information is available in the digital documentation. Enter `apropos samba` at the command line to display some manual pages or just browse the `/usr/share/doc/packages/samba` directory if Samba is installed for more online documentation and examples. A commented example configuration (`smb.conf.SuSE`) can be found in the `examples` subdirectory.

Some important new features of the enclosed version 3 of the `samba` package include:

- Support for Active Directory
- Improved Unicode support
- The internal authentication mechanisms have been completely revised
- Improved support for the Windows 200x and XP printing system
- Servers can be set up as member servers in Active Directory domains
- Adoption of an NT4 domain, enabling the migration from the latter to a Samba domain

TIP: Migration to Samba3

There are some special points to take into account when migrating from Samba 2.x to Samba 3. A discussion of this topic is included in the Samba HOWTO Collection, where an entire chapter is dedicated to it. After installing the `samba-doc` package, find the HOWTO in `/usr/share/doc/packages/samba/Samba-HOWTO-Collection.pdf`.

Samba uses the SMB protocol (server message block) that is based on the NetBIOS services. Due to pressure from IBM, Microsoft released the protocol so other software manufacturers could establish connections to a Microsoft domain network. With Samba, the SMB protocol works on top of the TCP/IP protocol, so the TCP/IP protocol must be installed on all clients.

NetBIOS is a software interface (API) designed for communication between machines. Here, a name service is provided. It enables machines connected to the net to reserve names for themselves. After reservation, these machines can be addressed by name. There is no central process that checks names. Any machine on the network can reserve as many names as it wants, if the names are not already in use. The NetBIOS interface can now be implemented for different network architectures. An implementation that works relatively closely with network hardware is called NetBEUI, but this is often referred to as NetBIOS. Network protocols implemented with NetBIOS are IPX from Novell (NetBIOS via TCP/IP) and TCP/IP.

The NetBIOS names sent via TCP/IP have nothing in common with the names used in `/etc/hosts` or those defined by DNS. NetBIOS uses its own, completely independent naming convention. However, it is recommended to use names that correspond to DNS hostnames to make administration easier. This is the default used by Samba.

All common operating systems, such as Mac OS X, Windows, and OS/2, support the SMB protocol. The TCP/IP protocol must be installed on all computers. Samba provides a client for the different UNIX flavors. For Linux, there is a kernel module for SMB that allows the integration of SMB resources on the Linux system level.

SMB servers provide hardware space to their clients by means of shares. A share includes a directory and its subdirectories on the server. It is exported by means of a name and can be accessed by its name. The share name can be set to any name—it does not have to be the name of the export directory. A printer is also assigned a name. Clients can access the printer by its name.

48.1 Configuring the Server

If you intend to use Samba as a server, install `samba`. Start the services required for Samba with `rcnmb start && rcsmb start` and stop them with `rcsmb stop && rcnmb stop`.

The main configuration file of Samba is `/etc/samba/smb.conf`. This file can be divided into two logical parts. The `[global]` section contains the central and global settings. The `[share]` sections contain the individual file and printer shares. By means of this approach, details regarding the shares can be set differently or globally in the `[global]` section, which enhances the structural transparency of the configuration file.

48.1.1 The global Section

The following parameters of the `[global]` section need some adjustment to match the requirements of your network setup so other machines can access your Samba server via SMB in a Windows environment.

workgroup = TUX-NET

This line assigns the Samba server to a workgroup. Replace `TUX-NET` with an appropriate workgroup of your networking environment. Your Samba server appears under its DNS name unless this name has been assigned to any other machine in the network. If the DNS name is not available, set the server name using `netbiosname=MYNAME`. See `mansmb.conf` for more details about this parameter.

os level = 2

This parameter triggers whether your Samba server tries to become LMB (local master browser) for its work group. Choose a very low value to spare the existing Windows network from any disturbances caused by a misconfigured Samba server. More information about this important topic can be found in the files `BROWSING.txt` and `BROWSING-Config.txt` under the `textdocs` subdirectory of the package documentation.

If no other SMB server is present in your network (such as a Windows NT or 2000 server) and you want the Samba server to keep a list of all systems present in the local environment, set the `os level` to a higher value (for example, `65`). Your Samba server is then chosen as LMB for your local network.

When changing this setting, consider carefully how this could affect an existing Windows network environment. First test the changes in an isolated network or at a noncritical time of day.

wins support and wins server

To integrate your Samba server into an existing Windows network with an active WINS server, enable the `wins server` option and set its value to the IP address of that WINS server.

If your Windows machines are connected to separate subnets and should still be aware of each other, you need to set up a WINS server. To turn a Samba server into such a WINS server, set the option `wins support = Yes`. Make sure that only one Samba server of the network has this setting enabled. The options `wins server` and `wins support` must never be enabled at the same time in your `smb.conf` file.

48.1.2 Shares

The following examples illustrate how a CD-ROM drive and the user directories (`homes`) are made available to the SMB clients.

[cdrom]

To avoid having the CD-ROM drive accidentally made available, these lines are deactivated with comment marks (semicolons in this case). Remove the semicolons in the first column to share the CD-ROM drive with Samba.

Example 48.1 *A CD-ROM Share*

```
;[cdrom]
;       comment = Linux CD-ROM
;       path = /media/cdrom
;       locking = No
```

`[cdrom]` and `comment`

The entry `[cdrom]` is the name of the share that can be seen by all SMB clients on the net. An additional `comment` can be added to further describe the share.

path = /media/cdrom

path exports the directory /media/cdrom.

By means of a very restrictive default configuration, this kind of share is only made available to the users present on this system. If this share should be made available to everybody, add a line guest ok = yes to the configuration. This setting gives read permissions to anyone on the network. It is recommended to handle this parameter with great care. This applies even more to the use of this parameter in the [global] section.

[homes]

The [home] share is of special importance here. If the user has a valid account and password for the Linux file server and his own home directory, he can be connected to it.

Example 48.2 *homes Share*

```
[homes]
 comment = Home Directories
 valid users = %S
 browseable = No
 read only = No
 create mask = 0640
 directory mask = 0750
```

[homes]

As long as there is no other share using the share name of the user connecting to the SMB server, a share is dynamically generated using the [homes] share directives. The resulting name of the share is the username.

valid users = %S

%S is replaced with the concrete name of the share as soon as a connection has been successfully established. For a [homes] share, this is always the username. As a consequence, access rights to a user's share are restricted exclusively to the user.

browseable = No

This setting makes the share invisible in the network environment.

read only = No

By default, Samba prohibits write access to any exported share by means of the read only = Yes parameter. To make a share writable, set the value read only = No, which is synonymous with writable = Yes.

`create mask = 0640`

Systems that are not based on MS Windows NT do not understand the concept of UNIX permissions, so they cannot assign permissions when creating a file. The parameter `create mask` defines the access permissions assigned to newly created files. This only applies to writable shares. In effect, this setting means the owner has read and write permissions and the members of the owner's primary group have read permissions. `valid users = %S` prevents read access even if the group has read permissions. For the group to have read or write access, deactivate the line `valid users = %S`.

48.1.3 Security Levels

The SMB protocol comes from the DOS and Windows world and directly takes into consideration the problem of security. Each share access can be protected with a password. SMB has three possible ways of checking the permissions:

Share Level Security (security = share):

A password is firmly assigned to a share. Everyone who knows this password has access to that share.

User Level Security (security = user):

This variation introduces the concept of the user to SMB. Each user must register with the server with his own password. After registration, the server can grant access to individual exported shares dependent on usernames.

Server Level Security (security = server):

To its clients, Samba pretends to be working in user level mode. However, it passes all password queries to another user level mode server, which takes care of authentication. This setting expects an additional parameter (`password server =`).

The distinction between share, user, and server level security applies to the entire server. It is not possible to offer individual shares of a server configuration with share level security and others with user level security. However, you can run a separate Samba server for each configured IP address on a system.

More information about this subject can be found in the Samba HOWTO Collection. For multiple servers on one system, pay attention to the options `interfaces` and `bind interfaces only`.

TIP

For simple administration tasks with the Samba server, there is also the program swat. It provides a simple Web interface with which to configure the Samba server conveniently. In a Web browser, open `http://localhost:901` and log in as user `root`. However, swat must also be activated in the files `/etc/xinetd.d/samba` and `/etc/services`. To do so in `/etc/xinetd.d/samba`, edit the `disable` line so it reads `disable = no`. More information about swat is provided in the man page.

48.2 Samba as Login Server

In networks where predominantly Windows clients are found, it is often preferable that users may only register with a valid account and password. This can done with the help of a Samba server. In a Windows-based network, this task is handled by a Windows NT server configured as a primary domain controller (PDC). The entries that must be made in the `[global]` section of `smb.conf` are shown in Example 48.3, "Global Section in smb.conf" (page 743).

Example 48.3 *Global Section in smb.conf*

```
[global]
  workgroup = TUX-NET
  domain logons = Yes
  domain master = Yes
```

If encrypted passwords are used for verification purposes—this is the default setting with well-maintained MS Windows 9x installations, MS Windows NT 4.0 from service pack 3, and all later products—the Samba server must be able to handle these. The entry `encrypt passwords = yes` in the `[global]` section enables this (with Samba version 3, this is now the default). In addition, it is necessary to prepare user accounts and passwords in an encryption format that conforms with Windows. Do this with the command `smbpasswd -a name`. Create the domain account for the computers, required by the Windows NT domain concept, with the following commands:

Example 48.4 *Setting Up a Machine Account*

```
useradd hostname\$
smbpasswd -a -m hostname
```

With the useradd command, a dollar sign is added. The command smbpasswd inserts this automatically when the parameter -m is used. The commented configuration example (/usr/share/doc/packages/Samba/examples/smb.conf.SuSE) contains settings that automate this task.

Example 48.5 *Automated Setup of a Machine Account*

```
add machine script = /usr/sbin/useradd -g nogroup -c "NT Machine Account" \
-s /bin/false %m\$
```

To make sure that Samba can execute this script correctly, choose a Samba user with the required administrator permissions. To do so, select one user and add it to the ntadmin group. After that, all users belonging to this Linux group can be assigned Domain Admin status with the command:

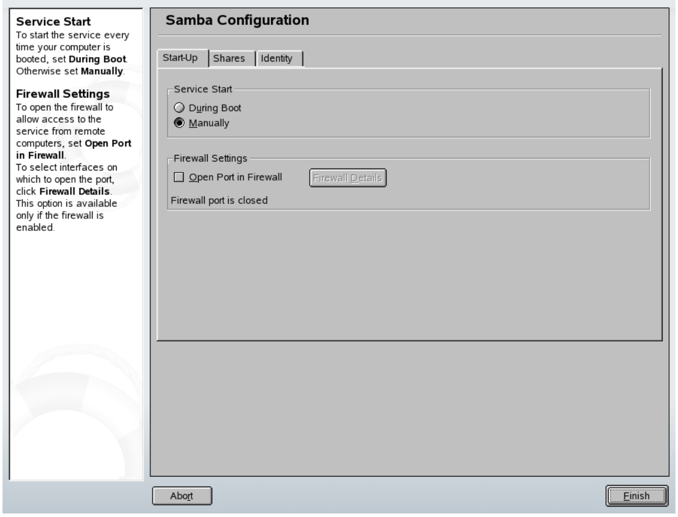
```
net groupmap add ntgroup="Domain Admins" unixgroup=ntadmin
```

More information about this topic is provided in Chapter 12 of the Samba HOWTO Collection, found in /usr/share/doc/packages/samba/Samba-HOWTO-Collection.pdf.

48.3 Configuring a Samba Server with YaST

Start the server configuration by selecting the workgroup or domain that your new Samba server should control. Select an existing one from *Workgroup or Domain Name* or enter a new one. In the next step, specify whether your server should act as PDC (primary domain controller) or as BDC (backup domain controller).

Figure 48.1 *Samba Configuration—Start Up*



Service Start
To start the service every time your computer is booted, set **During Boot**. Otherwise set **Manually**.

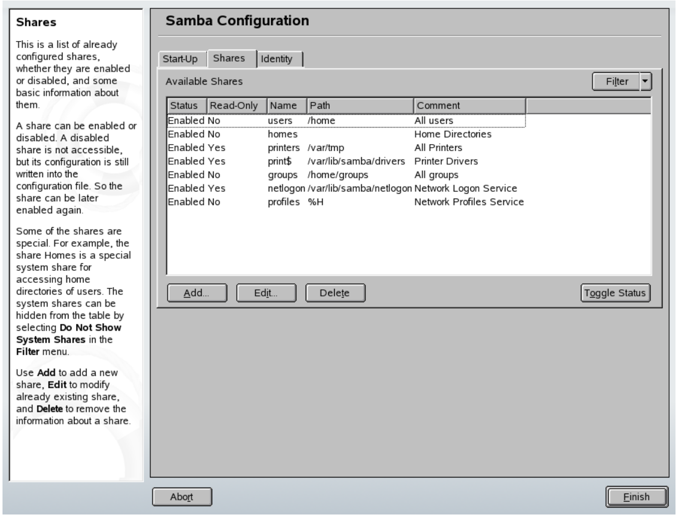
Firewall Settings
To open the firewall to allow access to the service from remote computers, set **Open Port in Firewall**. To select interfaces on which to open the port, click **Firewall Details**. This option is available only if the firewall is enabled.

Service Start
○ During Boot
● Manually

Firewall Settings
□ Open Port in Firewall
Firewall port is closed

Activate Samba in *Start Up*, which is shown in Figure 48.1, “Samba Configuration—Start Up” (page 745). Use *Open Ports in Firewall* and *Firewall Details* to adapt the firewall on the server in such a way that the ports for the `netbios-ns`, `netbios-dgm`, `netbios-ssn`, and `microsoft-ds` services are open on all external and internal interfaces, ensuring a smooth operation of the Samba server.

***Figure 48.2** Samba Configuration—Shares*



Shares

This is a list of already configured shares, whether they are enabled or disabled, and some basic information about them.

A share can be enabled or disabled. A disabled share is not accessible, but its configuration is still written into the configuration file. So the share can be later enabled again.

Some of the shares are special. For example, the share Homes is a special system share for accessing home directories of users. The system shares can be hidden from the table by selecting **Do Not Show System Shares** in the **Filter** menu.

Use **Add** to add a new share, **Edit** to modify already existing share, and **Delete** to remove the information about a share.

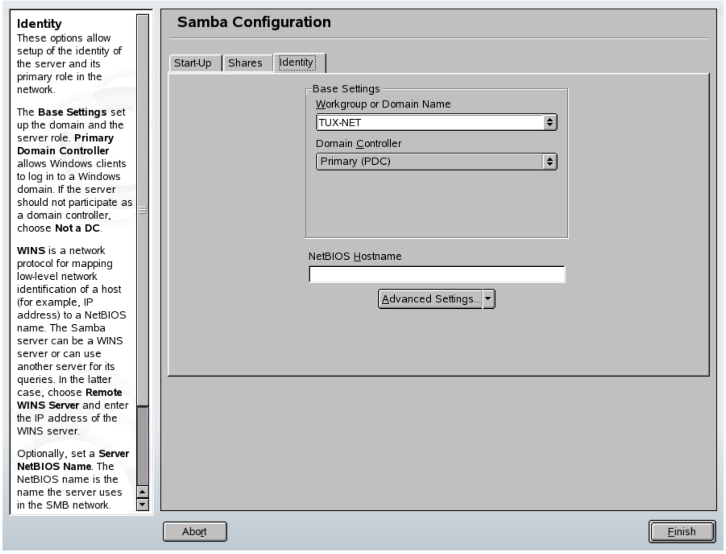
Samba Configuration

Available Shares

Status	Read-Only	Name	Path	Comment
Enabled	No	users	/home	All users
Enabled	No	homes		Home Directories
Enabled	Yes	printers	/var/tmp	All Printers
Enabled	Yes	print$	/var/lib/samba/drivers	Printer Drivers
Enabled	No	groups	/home/groups	All groups
Enabled	Yes	netlogon	/var/lib/samba/netlogon	Network Logon Service
Enabled	No	profiles	%H	Network Profiles Service

In *Shares* (Figure 48.2, “Samba Configuration—Shares” (page 746)), determine the Samba shares to activate. Use *Toggle Status* to switch between *Active* and *Inactive*. Click *Add* to add new shares.

Figure 48.3 *Samba Configuration—Identity*



Identity
These options allow setup of the identity of the server and its primary role in the network.

The **Base Settings** set up the domain and the server role. **Primary Domain Controller** allows Windows clients to log in to a Windows domain. If the server should not participate as a domain controller, choose **Not a DC**.

WINS is a network protocol for mapping low-level network identification of a host (for example, IP address) to a NetBIOS name. The Samba server can be a WINS server or can use another server for its queries. In the latter case, choose **Remote WINS Server** and enter the IP address of the WINS server.

Optionally, set a **Server NetBIOS Name**. The NetBIOS name is the name the server uses in the SMB network.

Samba Configuration

Base Settings
Workgroup or Domain Name: TUX-NET
Domain Controller: Primary (PDC)

NetBIOS Hostname:

In *Identity*, shown in Figure 48.3, “Samba Configuration—Identity” (page 747), determine the domain with which the host is associated (*Base Settings*) and whether to use an alternative hostname in the network (*NetBIOS Host Name*).

48.4 Configuring Clients

Clients can only access the Samba server via TCP/IP. NetBEUI and NetBIOS via IPX cannot be used with Samba.

48.4.1 Configuring a Samba Client with YaST

Configure a Samba client to access resources (files or printers) on the Samba server. Enter the domain or workgroup in the dialog *Samba Workgroup*. Click *Browse* to display all available groups and domains, which can be selected with the mouse. If you activate *Also Use SMB Information for Linux Authentication*, the user authentication runs over the Samba server. After completing all settings, click *Finish* to finish the configuration.

48.4.2 Windows 9x and ME

Windows 9x and ME already have built-in support for TCP/IP. However, this is not installed as the default. To add TCP/IP, go to *Control Panel* → *System* and choose *Add* → *Protocols* → *TCP/IP from Microsoft*. After rebooting your Windows machine, find the Samba server by double-clicking the desktop icon for the network environment.

TIP

To use a printer on the Samba server, install the standard or Apple-PostScript printer driver from the corresponding Windows version. It is best to link this to the Linux printer queue, which accepts Postscript as an input format.

48.5 Optimization

`socket options` is one possible optimization provided with the sample configuration that ships with your Samba version. Its default configuration refers to a local ethernet network. For additional information about `socket options`, refer to the relevant section of the manual pages of `smb.conf` and to the manual page of `socket(7)`. Further information is provided in the Samba performance tuning chapter of the Samba HOWTO Collection.

The standard configuration in `/etc/samba/smb.conf` is designed to provide useful settings based on the default settings of the Samba team. However, a ready-to-use configuration is not possible, especially for the network configuration and the workgroup name. The commented sample configuration `examples/smb.conf.SuSE` contains information that is helpful for adaption to local requirements.

TIP

The Samba HOWTO Collection provided by the Samba team includes a section about troubleshooting. In addition to that, Part V of the document provides a step-by-step guide to checking your configuration.

49 The Proxy Server Squid

Squid is a widely-used proxy cache for Linux and UNIX platforms. This chapter discusses its configuration, the settings required to get it running, how to configure the system to do transparent proxying, how to gather statistics about using the cache with the help of programs, like Calamaris and cachemgr, and how to filter Web contents with squidGuard.

Squid acts as a proxy cache. It redirects object requests from clients (in this case, from Web browsers) to the server. When the requested objects arrive from the server, it delivers the objects to the client and keeps a copy of them in the hard disk cache. One of the advantages of caching is that several clients requesting the same object can be served from the hard disk cache. This enables clients to receive the data much faster than from the Internet. This procedure also reduces the network traffic.

Along with the actual caching, Squid offers a wide range of features such as distributing the load over intercommunicating hierarchies of proxy servers, defining strict access control lists for all clients accessing the proxy, allowing or denying access to specific Web pages with the help of other applications, and generating statistics about frequently-visited Web pages for the assessment of the users' surfing habits. Squid is not a generic proxy. It normally proxies only HTTP connections. It does also support the protocols FTP, Gopher, SSL, and WAIS, but it does not support other Internet protocols, such as Real Audio, news, or video conferencing. Because Squid only supports the UDP protocol to provide communication between different caches, many other multimedia programs are not supported.

49.1 Some Facts about Proxy Caches

As a proxy cache, Squid can be used in several ways. When combined with a firewall, it can help with security. Multiple proxies can be used together. It can also determine what types of objects should be cached and for how long.

49.1.1 Squid and Security

It is possible to use Squid together with a firewall to secure internal networks from the outside using a proxy cache. The firewall denies all clients access to external services except Squid. All Web connections must be established by way of the proxy.

If the firewall configuration includes a DMZ, the proxy should operate within this zone. In this case, it is important that all computers in the DMZ send their log files to hosts inside the secure network. The possibility of implementing a *transparent* proxy is covered in Section 49.5, “Configuring a Transparent Proxy” (page 760).

49.1.2 Multiple Caches

Several proxies can be configured in such a way that objects can be exchanged between them. This reduces the total system load and increases the chances of finding an object already existing in the local network. It is also possible to configure cache hierarchies, so a cache is able to forward object requests to sibling caches or to a parent cache—causing it to get objects from another cache in the local network or directly from the source.

Choosing the appropriate topology for the cache hierarchy is very important, because it is not desirable to increase the overall traffic on the network. For a very large network, it would make sense to configure a proxy server for every subnetwork and connect them to a parent proxy, which in turn is connected to the proxy cache of the ISP.

All this communication is handled by ICP (Internet cache protocol) running on top of the UDP protocol. Data transfers between caches are handled using HTTP (hypertext transmission protocol) based on TCP.

To find the most appropriate server from which to get the objects, one cache sends an ICP request to all sibling proxies. These answer the requests via ICP responses with a

HIT code if the object was detected or a MISS if it was not. If multiple HIT responses were found, the proxy server decides from which server to download, depending on factors such as which cache sent the fastest answer or which one is closer. If no satisfactory responses are received, the request is sent to the parent cache.

TIP

To avoid duplication of objects in different caches in the network, other ICP protocols are used, such as CARP (cache array routing protocol) or HTCP (hypertext cache protocol). The more objects maintained in the network, the greater the possibility of finding the desired one.

49.1.3 Caching Internet Objects

Not all objects available in the network are static. There are a lot of dynamically generated CGI pages, visitor counters, and encrypted SSL content documents. Objects like this are not cached because they change each time they are accessed.

The question remains as to how long all the other objects stored in the cache should stay there. To determine this, all objects in the cache are assigned one of various possible states. Web and proxy servers find out the status of an object by adding headers to these objects, such as “Last modified” or “Expires” and the corresponding date. Other headers specifying that objects must not be cached are used as well.

Objects in the cache are normally replaced, due to a lack of free hard disk space, using algorithms such as LRU (last recently used). Basically this means that the proxy expunges the objects that have not been requested for the longest time.

49.2 System Requirements

The most important thing is to determine the maximum load the system must bear. It is, therefore, important to pay more attention to the load peaks, because these might be more than four times the day's average. When in doubt, it would be better to overestimate the system's requirements, because having Squid working close to the limit of its capabilities could lead to a severe loss in the quality of the service. The following sections point to the system factors in order of significance.

49.2.1 Hard Disks

Speed plays an important role in the caching process, so this factor deserves special attention. For hard disks, this parameter is described as *random seek time*, measured in milliseconds. Because the data blocks that Squid reads from or writes to the hard disk tend to be rather small, the seek time of the hard disk is more important than its data throughput. For the purposes of a proxy, hard disks with high rotation speeds are probably the better choice, because they allow the read-write head to be positioned in the required spot more quickly. One possibility to speed up the system is to use a number of disks concurrently or to employ striping RAID arrays.

49.2.2 Size of the Disk Cache

In a small cache, the probability of a HIT (finding the requested object already located there) is small, because the cache is easily filled so the less requested objects are replaced by newer ones. If, for example, one GB is available for the cache and the users only surf ten MB per day, it would take more than one hundred days to fill the cache.

The easiest way to determine the needed cache size is to consider the maximum transfer rate of the connection. With a 1 Mbit/s connection, the maximum transfer rate is 125 KB/s. If all this traffic ends up in the cache, in one hour it would add up to 450 MB and, assuming that all this traffic is generated in only eight working hours, it would reach 3.6 GB in one day. Because the connection is normally not used to its upper volume limit, it can be assumed that the total data volume handled by the cache is approximately 2 GB. This is why 2 GB of disk space is required in the example for Squid to keep one day's worth of browsed data cached.

49.2.3 RAM

The amount of memory (RAM) required by Squid directly correlates to the number of objects in the cache. Squid also stores cache object references and frequently requested objects in the main memory to speed up retrieval of this data. Random access memory is much faster than a hard disk.

In addition to that, there is other data that Squid needs to keep in memory, such as a table with all the IP addresses handled, an exact domain name cache, the most frequently requested objects, access control lists, buffers, and more.

It is very important to have sufficient memory for the Squid process, because system performance is dramatically reduced if it must be swapped to disk. The cachemgr.cgi tool can be used for the cache memory management. This tool is introduced in Section 49.6, "cachemgr.cgi" (page 763).

49.2.4 CPU

Squid is not a program that requires intensive CPU usage. The load of the processor is only increased while the contents of the cache are loaded or checked. Using a multiprocessor machine does not increase the performance of the system. To increase efficiency, it is better to buy faster disks or add more memory.

49.3 Starting Squid

Squid is already preconfigured in SUSE Linux, so you can start it right after the installation. To ensure a smooth start-up, the network should be configured in such a way that at least one name server and the Internet can be reached. Problems can arise if a dial-up connection is used with a dynamic DNS configuration. In cases such as this, at least the name server should be clearly entered, because Squid does not start if it does not detect a DNS server in `/etc/resolv.conf`.

49.3.1 Commands for Starting and Stopping Squid

To start Squid, enter `rcsquid start` at the command line as `root`. For the initial start-up, the directory structure must first be defined in `/var/squid/cache`. This is done by the start script `/etc/init.d/squid` automatically and can take a few seconds or even minutes. If `done` appears to the right in green, Squid has been successfully loaded. To test the functionality of Squid on the local system, enter `localhost` as the proxy and `3128` as the port in the browser.

To allow all users to access Squid and, through it, the Internet, change the entry in the configuration file `/etc/squid/squid.conf` from `http_access deny all` to `http_access allow all`. However, in doing so, consider that Squid is made completely accessible to anyone by this action. Therefore, define ACLs that control access to the proxy. More information about this is available in Section 49.4.2, “Options for Access Controls” (page 758).

After modifying the configuration file /etc/squid/squid.conf, Squid must reload the configuration file. Do this with rcsquid reload. Alternatively, completely restart Squid with rcsquid restart.

The command rcsquid status can be used to check if the proxy is running. The command rcsquid stop causes Squid to shut down. This can take a while, because Squid waits up to half a minute (shutdown_lifetime option in /etc/squid/squid.conf) before dropping the connections to the clients and writing its data to the disk.

WARNING: Terminating Squid

Terminating Squid with kill or killall can damage the cache. To be able to restart Squid, the damaged cache must be deleted.

If Squid dies after a short period of time even though it was started successfully, check whether there is a faulty name server entry or whether the /etc/resolv.conf file is missing. Squid logs the cause of a start-up failure in the file /var/squid/logs/cache.log. If Squid should be loaded automatically when the system boots, use the YaST runlevel editor to activate Squid for the desired runlevels. See Section “System Services (Runlevel)” (Chapter 3, *System Configuration with YaST*, ↑Start-Up).

An uninstall of Squid does not remove the cache hierarchy or the log files. To remove these, delete the /var/cache/squid directory manually.

49.3.2 Local DNS Server

Setting up a local DNS server makes sense even if it does not manage its own domain. It then simply acts as a caching-only name server and is also able to resolve DNS requests via the root name servers without requiring any special configuration (see Section 40.3, “Starting the Name Server BIND” (page 601)). How this can be done depends on whether you chose dynamic DNS during the configuration of the Internet connection.

Dynamic DNS

Normally, with dynamic DNS, the DNS server is set by the provider during the establishment of the Internet connection and the local file /etc/resolv.conf is adjusted automatically. This behavior is achieved by way of the sysconfig variable MODIFY_RESOLV_CONF_DYNAMICALLY, which is set to YES. Set this variable to NO with the YaST sysconfig editor (see Section 28.3.1, “Changing the System Configuration Using the YaST sysconfig Editor” (page 424)). Then enter the local DNS server in the file /etc/resolv.conf with the IP address 127.0.0.1 for localhost. This way Squid can always find the local name server when it starts.

To make the provider's name server accessible, enter it in the configuration file /etc/named.conf under forwarders along with its IP address. With dynamic DNS, this can be achieved automatically during connection establishment by setting the sysconfig variable MODIFY_NAMED_CONF_DYNAMICALLY to YES.

Static DNS

With static DNS, no automatic DNS adjustments take place while establishing a connection, so there is no need to change any sysconfig variables. You must, however, enter the local DNS server in the file /etc/resolv.conf as described above. Additionally, the providers static name server must be entered manually in the file /etc/named.conf under forwarders along with its IP address.

TIP: DNS and Firewall

If you have a firewall running, make sure DNS requests can pass it.

49.4 The Configuration File /etc/squid/squid.conf

All Squid proxy server settings are made in the /etc/squid/squid.conf file. To start Squid for the first time, no changes are necessary in this file, but external clients are initially denied access. The proxy is available for the localhost. The default port is 3128. The preinstalled /etc/squid/squid.conf provides detailed information about the options and many examples. Nearly all entries begin with # (the lines are commented) and the relevant specifications can be found at the end of the line. The given values almost always correlate with the default values, so removing the comment signs without changing any of the parameters actually has little effect in most cases. If possible, leave the sample as it is and insert the options along with the modified parameters in the line below. In this way, easily interpret the default values and the changes.

TIP: Adapting the Configuration File after an Update

If you have updated from an earlier Squid version, it is recommended to edit the new `/etc/squid/squid.conf` and only apply the changes made in the previous file. If you try to implement the old `squid.conf`, risk that the configuration no longer functions, because options are sometimes modified and new changes added.

49.4.1 General Configuration Options (Selection)

http_port 3128

This is the port on which Squid listens for client requests. The default port is `3128`, but `8080` is also common. If desired, specify several port numbers separated by blank spaces.

cache_peer *hostname type proxy-port icp-port*

Here, enter a parent proxy, for example, if you want to use the proxy of your ISP. As *hostname*, enter the name and IP address of the proxy to use and, as *type*, enter `parent`. For *proxy-port*, enter the port number that is also set by the operator of the parent for use in the browser, usually `8080`. Set the *icp-port* to `7` or `0` if the ICP port of the parent is not known and its use is irrelevant to the provider. In addition, `default` and `no-query` should be specified after the port numbers to prohibit the use of the ICP protocol. Squid then behaves like a normal browser as far as the provider's proxy is concerned.

cache_mem 8 MB

This entry defines the amount of memory Squid can use for the caches. The default is `8 MB`.

cache_dir ufs /var/cache/squid/ 100 16 256

The entry *cache_dir* defines the directory where all the objects are stored on disk. The numbers at the end indicate the maximum disk space in MB to use and the number of directories in the first and second level. The `ufs` parameter should be left alone. The default is 100 MB occupied disk space in the `/var/cache/squid` directory and creation of 16 subdirectories inside it, each containing 256 more subdirectories. When specifying the disk space to use, leave sufficient reserve disk space. Values from a minimum of 50% to a maximum of 80% of the available disk space make the most sense here. The last two numbers for the directories should only be increased with caution, because too many directories can also lead to performance problems. If you have several disks that share the cache, enter several *cache_dir* lines.

cache_access_log /var/log/squid/access.log
Path for log messages.

cache_log /var/log/squid/cache.log
Path for log messages.

cache_store_log /var/log/squid/store.log
Path for log messages.

These three entries specify the paths where Squid logs all its actions. Normally, nothing is changed here. If Squid is experiencing a heavy usage burden, it might make sense to distribute the cache and the log files over several disks.

emulate_httpd_log off
If the entry is set to *on*, obtain readable log files. Some evaluation programs cannot interpret this, however.

client_netmask 255.255.255.255
With this entry, mask IP addresses in the log files to hide the clients' identity. The last digit of the IP address is set to zero if you enter `255.255.255.0` here.

ftp_user Squid@
With this, set the password Squid should use for the anonymous FTP login. It can make sense to specify a valid e-mail address here, because some FTP servers check these for validity.

cache_mgr webmaster
An e-mail address to which Squid sends a message if it unexpectedly crashes. The default is *webmaster*.

logfile_rotate 0
If you run `squid -k rotate`, `Squid` can rotate secured log files. The files are numbered in this process and, after reaching the specified value, the oldest file is overwritten. The default value is `0` because archiving and deleting log files in SUSE Linux is carried out by a cron job set in the configuration file `/etc/logrotate/squid`.

append_domain <domain>

With *append_domain*, specify which domain to append automatically when none is given. Usually, your own domain is entered here, so entering *www* in the browser accesses your own Web server.

forwarded_for on

If you set the entry to *off*, Squid removes the IP address and the system name of the client from HTTP requests.

negative_ttl 5 minutes; negative_dns_ttl 5 minutes

Normally, you do not need to change these values. If you have a dial-up connection, however, the Internet may, at times, not be accessible. Squid makes a note of the failed requests then refuses to issue new ones, although the Internet connection has been reestablished. In a case such as this, change the *minutes* to *seconds* then, after clicking *Reload* in the browser, the dial-up process should be reengaged after a few seconds.

never_direct allow *`acl_name`*

To prevent Squid from taking requests directly from the Internet, use the above command to force connection to another proxy. This must have previously been entered in *cache_peer*. If `all` is specified as the *`acl_name`*, force all requests to be forwarded directly to the *parent*. This might be necessary, for example, if you are using a provider that strictly stipulates the use of its proxies or denies its firewall direct Internet access.

49.4.2 Options for Access Controls

Squid provides a detailed system for controlling the access to the proxy. By implementing ACLs, it can be configured easily and comprehensively. This involves lists with rules that are processed sequentially. ACLs must be defined before they can be used. Some default ACLs, such as *all* and *localhost*, already exist. However, the mere definition of an ACL does not mean that it is actually applied. This only happens in conjunction with *http_access* rules.

acl <acl_name> <type> <data>

An ACL requires at least three specifications to define it. The name <*acl_name*> can be chosen arbitrarily. For <*type*>, select from a variety of different options, which can be found in the *ACCESS CONTROLS* section in the `/etc/squid/squid.conf` file. The specification for <*data*> depends on the individual ACL type and can also be read from a file, for example, via hostnames, IP addresses, or URLs. The following are some simple examples:

```
acl mysurfers srcdomain .my-domain.com
acl teachers src 192.168.1.0/255.255.255.0
acl students src 192.168.7.0-192.168.9.0/255.255.255.0
acl lunch time MTWHF 12:00-15:00
```

http_access allow <acl_name>

http_access defines who is allowed to use the proxy and who can access what on the Internet. For this, ACLs must be given. *localhost* and *all* have already been defined above, which can deny or allow access via *deny* or *allow*. A list containing any number of *http_access* entries can be created, processed from top to bottom, and, depending on which occurs first, access is allowed or denied to the respective URL. The last entry should always be *http_access deny all*. In the following example, the *localhost* has free access to everything while all other hosts are denied access completely.

```
http_access allow localhost
http_access deny all
```

In another example using these rules, the group `teachers` always has access to the Internet. The group `students` only gets access Monday to Friday during lunch time.

```
http_access deny localhost
http_access allow teachers
http_access allow students lunch time
http_access deny all
```

The list with the *http_access* entries should only be entered, for the sake of readability, at the designated position in the `/etc/squid/squid.conf` file. That is, between the text

```
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR
# CLIENTS
```

and the last

```
http_access deny all
```

redirect_program /usr/bin/squidGuard

With this option, specify a redirector such as squidGuard, which allows blocking unwanted URLs. Internet access can be individually controlled for various user groups with the help of proxy authentication and the appropriate ACLs. squidGuard is a separate package that can be installed and configured.

auth_param basic program /usr/sbin/pam_auth

If users must be authenticated on the proxy, set a corresponding program, such as pam_auth. When accessing pam_auth for the first time, the user sees a login window in which to enter the username and password. In addition, an ACL is still required, so only clients with a valid login can use the Internet:

```
acl password proxy_auth REQUIRED

http_access allow password
http_access deny all
```

The *REQUIRED* after *proxy_auth* can be replaced with a list of permitted usernames or with the path to such a list.

ident_lookup_access allow <acl_name>

With this, have an ident request run for all ACL-defined clients to find each user's identity. If you apply *all* to the *<acl_name>*, this is valid for all clients. Also, an ident daemon must be running on all clients. For Linux, install the pidentd package for this purpose. For Microsoft Windows, free software is available for download from the Internet. To ensure that only clients with a successful ident lookup are permitted, define a corresponding ACL here:

```
acl identhosts ident REQUIRED

http_access allow identhosts
http_access deny all
```

Here, too, replace *REQUIRED* with a list of permitted usernames. Using *ident* can slow down the access time quite a bit, because ident lookups are repeated for each request.

49.5 Configuring a Transparent Proxy

The usual way of working with proxy servers is the following: the Web browser sends requests to a certain port in the proxy server and the proxy provides these required objects, whether they are in its cache or not. When working in a network, several situations may arise:

- For security reasons, it is recommended that all clients use a proxy to surf the Internet.

- All clients must use a proxy, regardless of whether they are aware of it.

- The proxy in a network is moved, but the existing clients should retain their old configuration.

In all these cases, a transparent proxy may be used. The principle is very easy: the proxy intercepts and answers the requests of the Web browser, so the Web browser receives the requested pages without knowing from where they are coming. As the name indicates, the entire process is done transparently.

49.5.1 Configuration Options in /etc/squid/squid.conf

The options to activate in the `/etc/squid/squid.conf` file to get the transparent proxy up and running are:

- `httpd_accel_host` virtual

- `httpd_accel_port` 80

  The port number where the actual HTTP server is located

- `httpd_accel_with_proxy` on

- `httpd_accel_uses_host_header` on

49.5.2 Firewall Configuration with SuSEfirewall2

Now redirect all incoming requests via the firewall with help of a port forwarding rule to the Squid port. To do this, use the enclosed tool SuSEfirewall2. Its configuration file can be found in `/etc/sysconfig/SuSEfirewall2`. The configuration file consists of well-documented entries. Even to set only a transparent proxy, you must configure some firewall options:

- Device pointing to the Internet: `FW_DEV_EXT`="eth1"

- Device pointing to the network: `FW_DEV_INT`="eth0"

Define ports and services (see `/etc/services`) on the firewall that are accessed from untrusted (external) networks such as the Internet. In this example, only Web services are offered to the outside:

```
FW_SERVICES_EXT_TCP="www"
```

Define ports or services (see `/etc/services`) on the firewall that are accessed from the secure (internal) network, both via TCP and UDP:

```
FW_SERVICES_INT_TCP="domain www 3128"
FW_SERVICES_INT_UDP="domain"
```

This allows accessing Web services and Squid (whose default port is `3128`). The service “domain” stands for DNS (domain name service). This service is commonly used. Otherwise, simply take it out of the above entries and set the following option to `no`:

```
FW_SERVICE_DNS="yes"
```

The most important option is option number `15`:

Example 49.1 *Firewall Configuration: Option 15*

```
#
# 15.)
# Which accesses to services should be redirected to a local port
# on the firewall machine?
#
# This can be used to force all internal users to surf via your
# Squid proxy, or transparently redirect incoming Web traffic to
# a secure Web server.
#
# Choice: leave empty or use the following explained syntax of
# redirecting rules, separated with spaces.
# A redirecting rule consists of 1) source IP/net,
# 2) destination IP/net, 3) original destination port and
# 4) local port to redirect the traffic to, separated by a colon,
# e.g. "10.0.0.0/8,0/0,80,3128 0/0,172.20.1.1,80,8080"
#
```

The comments above show the syntax to follow. First, enter the IP address and the netmask of the internal networks accessing the proxy firewall. Second, enter the IP address and the netmask to which these clients send their requests. In the case of Web browsers, specify the networks `0/0`, a wild card that means “to everywhere.” After that, enter the original port to which these requests are sent and, finally, the port to which all these requests are redirected. Because Squid supports protocols other than

HTTP, redirect requests from other ports to the proxy, such as FTP (port 21), HTTPS, or SSL (port 443). In this example, Web services (port 80) are redirected to the proxy port (port 3128). If there are more networks or services to add, they must be separated by a blank space in the respective entry.

```
FW_REDIRECT_TCP="192.168.0.0/16,0/0,80,3128 192.168.0.0/16,0/0,21,3128"
FW_REDIRECT_UDP="192.168.0.0/16,0/0,80,3128 192.168.0.0/16,0/0,21,3128"
```

To start the firewall and the new configuration with it, change an entry in the /etc/sysconfig/SuSEfirewall2 file. The entry START_FW must be set to "yes".

Start Squid as shown in Section 49.3, “Starting Squid” (page 753). To check if everything is working properly, check the Squid logs in /var/log/squid/access.log. To verify that all ports are correctly configured, perform a port scan on the machine from any computer outside your network. Only the Web services (port 80) should be open. To scan the ports with nmap, the command syntax is nmap -O IP_address.

49.6 cachemgr.cgi

The cache manager (cachemgr.cgi) is a CGI utility for displaying statistics about the memory usage of a running Squid process. It is also a more convenient way to manage the cache and view statistics without logging the server.

49.6.1 Setup

First, a running Web server on your system is required. To check if Apache is already running, as root enter the command rcapache status. If a message like this appears:

```
Checking for service httpd: OK
Server uptime: 1 day 18 hours 29 minutes 39 seconds
```

Apache is running on the machine. Otherwise, enter rcapache start to start Apache with the SUSE Linux default settings. The last step to set it up is to copy the file cachemgr.cgi to the Apache directory cgi-bin:

```
cp /usr/share/doc/packages/squid/scripts/cachemgr.cgi /srv/www/cgi-bin/
```

49.6.2 Cache Manager ACLs in /etc/squid/squid.conf

There are some default settings in the original file required for the cache manager. The first ACL is the most important, as the cache manager tries to communicate with Squid over the cache_object protocol.

```
acl manager proto cache_object
acl localhost src 127.0.0.1/255.255.255.255
```

The following rules should also be contained:

```
http_access allow manager localhost
http_access deny manager
```

The following rules assume that the Web server and Squid are running on the same machine. If the communication between the cache manager and Squid originates at the Web server on another computer, include an extra ACL as in Example 49.2, "Access Rules" (page 764).

Example 49.2 *Access Rules*

```
acl manager proto cache_object
acl localhost src 127.0.0.1/255.255.255.255
acl webserver src 192.168.1.7/255.255.255.255 # webserver IP
```

Then add the rules in Example 49.3, "Access Rules" (page 764).

Example 49.3 *Access Rules*

```
http_access allow manager localhost
http_access allow manager webserver
http_access deny manager
```

Configure a password for the manager for access to more options, like closing the cache remotely or viewing more information about the cache. For this, configure the entry `cachemgr_passwd` with a password for the manager and the list of options to view. This list appears as a part of the entry comments in `/etc/squid/squid.conf`.

Restart Squid every time the configuration file is changed. Do this easily with `rcsquid reload`.

49.6.3 Viewing the Statistics

Go to the corresponding Web site—http://webserver.example.org/cgi-bin/cachemgr.cgi. Press *continue* and browse through the different statistics. More details for each entry shown by the cache manager is in the Squid FAQ at http://www.squid-cache.org/Doc/FAQ/FAQ-9.html.

49.7 squidGuard

This section is not intended to explain an extensive configuration of squidGuard, only to introduce it and give some advice for using it. For more in-depth configuration issues, refer to the squidGuard Web site at http://www.squidguard.org.

squidGuard is a free (GPL), flexible, and fast filter, redirector, and access controller plug-in for Squid. It lets you define multiple access rules with different restrictions for different user groups on a Squid cache. squidGuard uses Squid's standard redirector interface.

squidGuard can do the following:

- Limit the Web access for some users to a list of accepted or well-known Web servers or URLs.
- Block access to some listed or blacklisted Web servers or URLs for some users.
- Block access to URLs matching a list of regular expressions or words for some users.
- Redirect blocked URLs to an “intelligent” CGI-based information page.
- Redirect unregistered users to a registration form.
- Redirect banners to an empty GIF.
- Use different access rules based on time of day, day of the week, date, etc.
- Use different rules for different user groups.

squidGuard and Squid cannot be used to:

- Edit, filter, or censor text inside documents.
- Edit, filter, or censor HTML-embedded script languages, such as JavaScript or VBscript.

Before it can be used, install squidGuard. Provide a minimal configuration file as /etc/squidguard.conf. Find configuration examples in http://www.squidguard.org/config/. Experiment later with more complicated configuration settings.

Next, create a dummy “access denied” page or a more or less complex CGI page to redirect Squid if the client requests a blacklisted Web site. Using Apache is strongly recommended.

Now, configure Squid to use squidGuard. Use the following entry in the /etc/squid/squid.conf file:

```
redirect_program /usr/bin/squidGuard
```

Another option called redirect_children configures the number of “redirect” (in this case squidGuard) processes running on the machine. squidGuard is fast enough to handle many requests: on a 500 MHz Pentium with 5,900 domains and 7,880 URLs (totalling 13,780), 100,000 requests can be processed within 10 seconds. Therefore, it is not recommended to set more than four processes, because the allocation of these processes would consume an excessive amount of memory

```
redirect_children 4
```

Last, have Squid load the new configuration by running rcsquid reload. Now, test your settings with a browser.

49.8 Cache Report Generation with Calamaris

Calamaris is a Perl script used to generate reports of cache activity in ASCII or HTML format. It works with native Squid access log files. The Calamaris home page is located at http://Calamaris.Cord.de/. The program is quite easy to use.

Log in as `root` then enter `cat access.log.files | calamaris` *`options`* `> reportfile`. It is important when piping more than one log file that the log files are chronologically ordered with older files first. These are some options of the program:

-a
: output all available reports

-w
: output as HTML report

-l
: include a message or logo in report header

More information about the various options can be found in the program's manual page with `man calamaris`.

A typical example is:

```
cat access.log.2 access.log.1 access.log | calamaris -a -w \
> /usr/local/httpd/htdocs/Squid/squidreport.html
```

This puts the report in the directory of the Web server. Apache is required to view the reports.

Another powerful cache report generator tool is SARG (Squid Analysis Report Generator). More information about this is available at: `http://web.onda.com.br/orso/`.

49.9 For More Information

Visit the home page of Squid at `http://www.squid-cache.org/`. Here, find the "Squid User Guide" and a very extensive collection of FAQs on Squid.

Following the installation, a small howto about transparent proxies is available in `howtoenh` as `/usr/share/doc/howto/en/txt/TransparentProxy.gz`. In addition, mailing lists are available for Squid at `squid-users@squid-cache.org`. The archive for this is located at `http://www.squid-cache.org/mail-archive/squid-users/`.

Index