

SUSE Linux Enterprise Desktop

10

July 04, 2006

www.novell.com



Deployment Guide

List of Authors: Jörg Arndt, Steve Bearnson, Stefan Behlert, Frank Bodammer, James Branam, Volker Buzek, Klara Cihlarova, Catherine Craft, Olaf Dabrunz, Stefan Dirsch, Olaf Donjak, Roman Drahtmüller, Thorsten Dubiel, Torsten Duwe, Thomas Fehr, Stefan Fent, Werner Fink, Jakub Friedl, Kurt Garloff, Joachim Gleißner, Todd Grant, Carsten Groß, Andreas Grünbacher, Berthold Gunreben, Franz Hassels, Andreas Jaeger, Jana Jaeger, Denise Jewkes, 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, Scott Rhoades, Thomas Rölz, Heiko Rommel, Tanja Roth, Marcus Schäfer, Thomas Schraitle, Kay Sievers, Klaus Singvogel, Frank Sundermeyer, Elisabeth Tobiasson, Hendrik Vogelsang, Klaus G. Wagner, Rebecca Walter, Christian Zoz

This publication is intellectual property of Novell Inc.

Its contents can be duplicated, either in part or in whole, provided that a copyright label is visibly located on each copy.

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.

Novell, the Novell logo, the N logo and SUSE are registered trademarks of Novell, Inc. in the United States and other countries. * Linux is a registered trademark of Linus Torvalds. All other third party trademarks are the property of their respective owners.

Contents

About This Guide	xiii
Part I Deployment	17
1 Planning for SUSE Linux Enterprise Desktop	19
1.1 Hardware Requirements	20
1.2 Reasons to Use SUSE Linux Enterprise Desktop	20
2 Deployment Strategies	23
2.1 Deploying up to 10 Workstations	23
2.2 Deploying up to 100 Workstations	25
2.3 Deploying More than 100 Workstations	32
3 Installation with YaST	33
3.1 System Start-Up for Installation	33
3.2 The Boot Screen	36
3.3 Language Selection	37
3.4 License Agreement	38
3.5 System Analysis	38
3.6 Time Zone	38
3.7 Installation Summary	39
3.8 Configuration	50
3.9 Graphical Login	58
4 Remote Installation	59
4.1 Installation Scenarios for Remote Installation	59

4.2	Setting Up the Server Holding the Installation Sources	68
4.3	Preparing the Boot of the Target System	77
4.4	Booting the Target System for Installation	87
4.5	Monitoring the Installation Process	92
5	Automated Installation	95
5.1	Simple Mass Installation	95
5.2	Rule-Based Autoinstallation	106
5.3	For More Information	112
6	Advanced Disk Setup	113
6.1	LVM Configuration	113
6.2	Soft RAID Configuration	119
7	System Configuration with YaST	125
7.1	YaST Language	126
7.2	The YaST Control Center	126
7.3	Software	127
7.4	Hardware	140
7.5	System	148
7.6	Network Devices	158
7.7	Network Services	159
7.8	AppArmor	163
7.9	Security and Users	163
7.10	Miscellaneous	167
7.11	YaST in Text Mode	170
7.12	Update from the Command Line	173
7.13	SaX2	176
7.14	Troubleshooting	182
7.15	For More Information	183
8	Updating SUSE Linux Enterprise	185
8.1	Updating SUSE Linux Enterprise	185
8.2	Installing Service Packs	187
8.3	Software Changes from Version 9 to Version 10	194

Part II Administration 207

9 GNOME Configuration for Administrators 209

9.1	Using GConf for Defaults	210
9.2	Customizing Menus	233
9.3	Installing Themes	237
9.4	Configuring Fonts	243
9.5	MIME Types	244
9.6	Setting Screensavers	246
9.7	Session Management	248
9.8	Improving Performance	249
9.9	Hidden Directories	258
9.10	Security Note on Configuring SMB Printers	260
9.11	Disabling GNOME Desktop Features	261
9.12	Starting Applications Automatically	263
9.13	Automounting and Managing Media Devices	264
9.14	Changing Preferred Applications	264
9.15	Managing Profiles Using Sabayon	264
9.16	Adding Document Templates	269

10 KDE Configuration for Administrators 271

10.1	Managing Profiles Using the KIOSK Admin Tool	271
10.2	Managing Profiles Manually	279

11 Active Directory Support 285

11.1	Integrating Linux and AD Environments	285
11.2	Background Information for Linux AD Support	286
11.3	Configuring a Linux Client for Active Directory	291
11.4	Logging In to an AD Domain	294
11.5	Changing Passwords	296

12 Access Control Lists in Linux 299

12.1	Traditional File Permissions	299
12.2	Advantages of ACLs	301
12.3	Definitions	301
12.4	Handling ACLs	302
12.5	ACL Support in Applications	310
12.6	For More Information	310

13	System Monitoring Utilities	311
13.1	Debugging	312
13.2	Files and File Systems	314
13.3	Hardware Information	316
13.4	Networking	318
13.5	The <code>/proc</code> File System	319
13.6	Processes	323
13.7	System Information	326
13.8	User Information	330
13.9	Time and Date	331
14	Working with the Shell	333
14.1	Getting Started with the Bash Shell	334
14.2	Users and Access Permissions	346
14.3	Important Linux Commands	349
14.4	The vi Editor	360
Part III	System	365
15	32-Bit and 64-Bit Applications in a 64-Bit System Environment	367
15.1	Runtime Support	367
15.2	Software Development	368
15.3	Software Compilation on Biarch Platforms	368
15.4	Kernel Specifications	370
16	Bootng and Configuring a Linux System	371
16.1	The Linux Boot Process	371
16.2	The init Process	375
16.3	System Configuration via <code>/etc/sysconfig</code>	383
17	The Boot Loader	387
17.1	Selecting a Boot Loader	388
17.2	Bootng with GRUB	388
17.3	Configuring the Boot Loader with YaST	397
17.4	Uninstalling the Linux Boot Loader	402
17.5	Creating Boot CDs	402
17.6	The Graphical SUSE Screen	403
17.7	Troubleshooting	404
17.8	For More Information	406

18	Special Features of SUSE Linux Enterprise	407
18.1	Information about Special Software Packages	407
18.2	Virtual Consoles	414
18.3	Keyboard Mapping	414
18.4	Language and Country-Specific Settings	415
19	Printer Operation	421
19.1	Workflow of the Printing System	422
19.2	Methods and Protocols for Connecting Printers	423
19.3	Installing the Software	423
19.4	Configuring the Printer	424
19.5	Configuration for Applications	431
19.6	Special Features in SUSE Linux Enterprise	431
19.7	Troubleshooting	436
20	Dynamic Kernel Device Management with udev	445
20.1	The <code>/dev</code> Directory	445
20.2	Kernel uevents and udev	446
20.3	Drivers, Kernel Modules, and Devices	446
20.4	Booting and Initial Device Setup	447
20.5	Debugging udev Events	447
20.6	Influencing Kernel Device Event Handling with udev Rules	448
20.7	Persistent Device Naming	449
20.8	The Replaced hotplug Package	450
20.9	For More Information	451
21	File Systems in Linux	453
21.1	Terminology	453
21.2	Major File Systems in Linux	454
21.3	Some Other Supported File Systems	460
21.4	Large File Support in Linux	461
21.5	For More Information	462
22	The X Window System	465
22.1	X11 Setup with SaX2	465
22.2	Optimizing the X Configuration	467
22.3	Installing and Configuring Fonts	473
22.4	OpenGL—3D Configuration	478

23 Authentication with PAM	483
23.1 Structure of a PAM Configuration File	484
23.2 The PAM Configuration of sshd	485
23.3 Configuration of PAM Modules	488
23.4 For More Information	490
24 Mobile Computing with Linux	491
24.1 Laptops	491
24.2 Mobile Hardware	498
24.3 Cellular Phones and PDAs	499
24.4 For More Information	500
25 PCMCIA	501
25.1 Controlling PCMCIA Cards Using pccardctl	502
25.2 PCMCIA in Detail	502
25.3 Troubleshooting	505
26 System Configuration Profile Management	509
26.1 Terminology	509
26.2 Setting Up SCPM	510
26.3 Configuring SCPM Using a Graphical User Interface	511
26.4 Configuring SCPM Using the Command Line	517
26.5 Troubleshooting	521
26.6 For More Information	522
27 Power Management	523
27.1 Power Saving Functions	524
27.2 APM	525
27.3 ACPI	526
27.4 Rest for the Hard Disk	534
27.5 The powersave Package	535
27.6 The YaST Power Management Module	543
28 Wireless Communication	549
28.1 Wireless LAN	549
28.2 Bluetooth	559
28.3 Infrared Data Transmission	570

Part IV	Services	575
29	Basic Networking	577
29.1	IP Addresses and Routing	581
29.2	IPv6—The Next Generation Internet	583
29.3	Name Resolution	593
29.4	Configuring a Network Connection with YaST	594
29.5	Managing Network Connections with NetworkManager	609
29.6	Configuring a Network Connection Manually	612
29.7	smpppd as Dial-up Assistant	628
30	SLP Services in the Network	631
30.1	Registering Your Own Services	631
30.2	SLP Front-Ends in SUSE Linux Enterprise	632
30.3	Activating SLP	633
30.4	For More Information	633
31	Time Synchronization with NTP	635
31.1	Configuring an NTP Client with YaST	635
31.2	Configuring xntp in the Network	638
31.3	Setting Up a Local Reference Clock	639
32	Using NIS	641
32.1	Configuring NIS Clients	641
33	Configuring eDirectory Authentication	643
33.1	Setting Up Workstations to Use eDirectory Authentication	644
33.2	Using iManager to Enable Users for eDirectory Authentication	648
33.3	Turning Off LUM and eDirectory Authentication	651
34	LDAP—A Directory Service	653
34.1	LDAP versus NIS	654
34.2	Structure of an LDAP Directory Tree	655
34.3	Configuring an LDAP Client with YaST	658
34.4	Configuring LDAP Users and Groups in YaST	666
34.5	For More Information	667

35 Samba	669
35.1 Terminology	669
35.2 Starting and Stopping Samba	671
35.3 Configuring a Samba Server	671
35.4 Configuring Clients	677
35.5 Samba as Login Server	678
35.6 For More Information	679
36 Sharing File Systems with NFS	681
36.1 Importing File Systems with YaST	681
36.2 Importing File Systems Manually	682
36.3 For More Information	682
37 File Synchronization	685
37.1 Available Data Synchronization Software	685
37.2 Determining Factors for Selecting a Program	688
37.3 Introduction to Unison	692
37.4 Introduction to CVS	694
37.5 Introduction to Subversion	697
37.6 Introduction to rsync	700
37.7 Introduction to mailsync	702
Part V Security	707
38 Masquerading and Firewalls	709
38.1 Packet Filtering with iptables	709
38.2 Masquerading Basics	711
38.3 Firewalling Basics	712
38.4 SuSEfirewall2	713
38.5 For More Information	718
39 SSH: Secure Network Operations	719
39.1 The OpenSSH Package	719
39.2 The ssh Program	720
39.3 scp—Secure Copy	720
39.4 sftp—Secure File Transfer	721
39.5 The SSH Daemon (sshd)—Server-Side	721
39.6 SSH Authentication Mechanisms	722
39.7 X, Authentication, and Forwarding Mechanisms	723

40	Network Authentication—Kerberos	725
40.1	Kerberos Terminology	725
40.2	How Kerberos Works	727
40.3	Users' View of Kerberos	730
40.4	For More Information	731
41	Encrypting Partitions and Files	733
41.1	Setting Up a Crypto File System with YaST	734
41.2	Using vi to Encrypt Single Files	736
42	Confining Privileges with AppArmor	739
42.1	Installing Novell AppArmor	740
42.2	Enabling and Disabling Novell AppArmor	740
42.3	Getting Started with Profiling Applications	742
43	Security and Confidentiality	749
43.1	Local Security and Network Security	749
43.2	Some General Security Tips and Tricks	758
43.3	Using the Central Security Reporting Address	761
Part VI	Troubleshooting	763
44	Help and Documentation	765
44.1	Using the SUSE Help Center	765
44.2	Man Pages	769
44.3	Info Pages	770
44.4	The Linux Documentation Project	770
44.5	Wikipedia: The Free Online Encyclopedia	771
44.6	Guides and Books	771
44.7	Package Documentation	772
44.8	Usenet	773
44.9	Standards and Specifications	773
45	Common Problems and Their Solutions	777
45.1	Finding Information	777
45.2	Installation Problems	778
45.3	Boot Problems	786
45.4	Login Problems	789
45.5	Network Problems	796

45.6	Data Problems	800
------	-------------------------	-----

Index		813
--------------	--	------------

About This Guide

This guide is intended for use by professional network and system administrators during the actual planning, deployment, configuration, and operation of SUSE® Linux Enterprise. As such, it is solely concerned with ensuring that SUSE Linux Enterprise is properly configured and that the required services on the network are available to allow it to function properly as initially installed. This guide does not cover the process of ensuring that SUSE Linux Enterprise offers proper compatibility with your enterprise's application software or that its core functionality meets those requirements. It assumes that a full requirements audit has been done and the installation has been requested or that a test installation, for the purpose of such an audit, has been requested.

This guide contains the following:

Deployment

Before you install SUSE Linux Enterprise, choose the deployment strategy and disk setup that is best suited for your scenario. Learn how to install your system manually, how to use network installation setups, and how to perform an autoinstallation. Configure the installed system with YaST to adapt it to your requirements.

Administration

SUSE Linux Enterprise offers a wide range of tools to customize various aspects of the system. This part introduces a few of them.

System

Learn more about the underlying operating system by studying this part. SUSE Linux Enterprise supports a number of hardware architectures and you can use this to adapt your own applications to run on SUSE Linux Enterprise. The boot loader and boot procedure information assists you in understanding how your Linux system works and how your own custom scripts and applications may blend in with it.

Services

SUSE Linux Enterprise is designed to be a network operating system. SUSE® Linux Enterprise Desktop includes client support for many network services. It integrates well into heterogeneous environments including MS Windows clients and servers.

Security

This edition of SUSE Linux Enterprise includes several security-related features. It ships with Novell® AppArmor, which enables you to protect your applications by restricting privileges. Secure login, firewalling, and file system encryption are covered as well.

Troubleshooting

SUSE Linux Enterprise includes a wealth of applications, tools, and documentation should you need them in case of trouble. Some of the most common problems that can occur with SUSE Linux Enterprise and their solutions are discussed in detail.

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 and enter your comments there.

2 Documentation Updates

For the latest version of this documentation, see the SUSE Linux Enterprise Desktop Web site [<http://www.novell.com/documentation/sled10/index.html>].

3 Additional Documentation

For additional documentation on this product, refer to <http://www.novell.com/documentation/sled10/index.html>:

GNOME User Guide

A comprehensive guide to the GNOME desktop and its most important applications.

KDE User Guide

A comprehensive guide to the KDE desktop and its most important applications.

Network Connectivity Guide

An in-depth introduction to networking using NetworkManager.

Novell AppArmor 2.0 Administration Guide

An in-depth administration guide to Novell AppArmor that introduces you to application confinement for heightened security in your environment.

For a documentation overview on the SUSE® Linux Enterprise Server product, refer to <http://www.novell.com/documentation/sles10/index.html>. The following manuals are exclusively available for SUSE Linux Enterprise Server:

Start-Up Guide

Basic information about installation types and work flows.

Architecture-Specific Information

Architecture-specific information needed to prepare a SUSE Linux Enterprise Server target for installation.

Installation and Administration

In-depth installation and administration for SUSE Linux Enterprise Server.

Many chapters in this manual contain links to additional documentation resources. This includes additional documentation that is available on the system as well as documentation available on the Internet.

4 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; keys are shown in uppercase as on a keyboard
- *File, File* → *Save As*: menu items, buttons

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

Part I. Deployment

Planning for SUSE Linux Enterprise Desktop

1

This chapter is addressed mainly to corporate system administrators who face the task of having to deploy SUSE® Linux Enterprise Desktop at their site. Rolling out SUSE Linux Enterprise Desktop to an entire site should involve careful planning and consideration of the following questions:

For which purpose will the SUSE Linux Enterprise Desktop workstations be used?

Determine the purpose for which SUSE Linux Enterprise Desktop should be used and make sure that hardware and software able to match these requirements are used. Consider testing your setup on a single machine before rolling it out to the entire site.

How many workstations should be installed?

Determine the scope of your deployment of SUSE Linux Enterprise Desktop. Depending on the number of installation planned, consider different approaches to the installation or even a mass installation using SUSE Linux Enterprises unique AutoYaST technology. For more information about this subject, refer to [Chapter 2, Deployment Strategies](#) (page 23).

How do you get software updates for your deployment?

All patches provided by Novell for your product are available for download to registered users. Register and find the patch support database at <http://www.novell.com/suselinuxportal>.

Do you need help for your local deployment?

Novell provides training, support, and consulting for all topics around SUSE Linux Enterprise Desktop. Find more information about this at <http://www.novell.com/products/desktop/>.

1.1 Hardware Requirements

SUSE Linux Enterprise Desktop requires certain minimum hardware requirements to be met before you can successfully install and run SUSE Linux Enterprise Desktop. A minimum installation of SUSE Linux Enterprise Desktop containing the most basic, essential software and a very minimalistic graphical user interface requires at least:

- Intel* Pentium* III, 500 MHz
- 256 MB of physical RAM
- 800 MB of available disk space
- 800 x 600 display resolution

For a standard installation of SUSE Linux Enterprise Desktop including the desktop environment of your choice (GNOME or KDE) and a wealth of applications, the following configuration is recommended:

- Intel Pentium IV, 2.4 GHz or higher or any AMD64 or Intel EM64T processor
- 1–2 physical CPUs
- 512 MB physical RAM or higher
- 1024 x 768 display resolution (or higher)

1.2 Reasons to Use SUSE Linux Enterprise Desktop

Let the following items guide you in your selection of SUSE Linux Enterprise Desktop and while determining the purpose of the installed systems:

Wealth of Applications

SUSE Linux Enterprise Desktop's broad offer of software makes it appeal to both professional users in a corporate environment and to home users or users in smaller networks.

Ease of Use

SUSE Linux Enterprise Desktop comes with two enterprise-ready desktop environments, GNOME and KDE. Both enable users to comfortably adjust to a Linux system while maintaining their efficiency and productivity. To explore the desktops in detail, refer to *GNOME User Guide* and *KDE User Guide*.

Support for Mobile Users

With the NetworkManager technology fully integrated into SUSE Linux Enterprise Desktop and its two desktop environments, mobile users will enjoy the freedom of easily joining and switching wired and wireless networks.

Seamless Integration into Existing Networks

SUSE Linux Enterprise Desktop was designed to be a versatile network citizen. It cooperates with various different network types:

Pure Linux Networks SUSE Linux Enterprise Desktop is a complete Linux client and supports all the protocols used in traditional Linux and Unix* environments. It integrates well with networks consisting of other SUSE Linux or SUSE Linux Enterprise machines. LDAP, NIS, and local authentication are supported.

Windows Networks SUSE Linux Enterprise Desktop supports Active Directory as an authentication source. It offers you all the advantages of a secure and stable Linux operating system plus convenient interaction with other Windows clients and means to manipulate your Windows user data from a Linux client. Explore this feature in detail in [Chapter 11, *Active Directory Support*](#) (page 285).

Windows and Novell Networks Being backed by Novell and their networking expertise, SUSE Linux Enterprise Desktop naturally offers you support for Novell technologies, like GroupWise, Novell Client for Linux, and iPrint, and it also offers authentication support for Novell eDirectory services.

Application Security with Novell AppArmor

SUSE Linux Enterprise Desktop enables you to secure your applications by enforcing security profiles tailor-made for your applications. To learn more about Novell AppArmor, refer to <http://www.novell.com/documentation/apparmor/>.

Deployment Strategies

There are several different ways to deploy SUSE® Linux Enterprise. Choose from various approaches ranging from a local installation using physical media or a network installation server to a mass deployment using a remote-controlled, highly-customized, and automated installation technique. Select the method that best matches your requirements.

2.1 Deploying up to 10 Workstations

If your deployment of SUSE Linux Enterprise only involves 1 to 10 workstations, the easiest and least complex way of deploying SUSE Linux Enterprise is a plain manual installation as featured in [Chapter 3, *Installation with YaST*](#) (page 33). Manual installation can be done in several different ways depending on your requirements:

[Installing from the SUSE Linux Enterprise Media](#) (page 24)

Consider this approach if you want to install a single, disconnected workstation.

[Installing from a Network Server Using SLP](#) (page 24)

Consider this approach if you have a single workstation or a small number of workstations and if a network installation server announced via SLP is available.

[Installing from a Network Server](#) (page 25)

Consider this approach if you have a single workstation or a small number of workstations and if a network installation server is available.

Table 2.1 *Installing from the SUSE Linux Enterprise Media*

Installation Source	SUSE Linux Enterprise media kit
Tasks Requiring Manual Interaction	<ul style="list-style-type: none">• Inserting the installation media• Booting the installation target• Changing media• Determining the YaST installation scope• Configuring the system with YaST system
Remotely Controlled Tasks	None
Details	Section “Installing from the SUSE Linux Enterprise Media” (page 34)

Table 2.2 *Installing from a Network Server Using SLP*

Installation Source	Network installation server holding the SUSE Linux Enterprise installation media
Tasks Requiring Manual Interaction	<ul style="list-style-type: none">• Inserting the boot disk• Booting installation target• Determining the YaST installation scope• Configuring the system with YaST
Remotely Controlled Tasks	None, but this method can be combined with VNC
Details	Section “Installing from a Network Server Using SLP” (page 34)

Table 2.3 *Installing from a Network Server*

Installation Source	Network installation server holding the SUSE Linux Enterprise installation media
Tasks Requiring Manual Interaction	<ul style="list-style-type: none">• Inserting the boot disk• Providing boot options• Booting the installation target• Determining the YaST installation scope• Configuring the system with YaST
Remotely Controlled Tasks	None, but method can be combined with VNC
Details	Section “Installing from a Network Server” (page 35)

2.2 Deploying up to 100 Workstations

With a growing numbers of workstations to install, you certainly do not want to install and configure each one of them manually. There are many automated or semiautomated approaches as well as several options to perform an installation with minimal to no physical user interaction.

Before considering a fully-automated approach, take into account that the more complex the scenario gets the longer it takes to set up. If a time limit is associated with your deployment, it might be a good idea to select a less complex approach that can be carried out much more quickly. Automation makes sense for huge deployments and those that need to be carried out remotely.

Choose from the following options:

[Simple Remote Installation via VNC—Static Network Configuration](#) (page 27)

Consider this approach in a small to medium scenario with a static network setup. A network, network installation server, and VNC viewer application are required.

[Simple Remote Installation via VNC—Dynamic Network Configuration](#) (page 27)

Consider this approach in a small to medium scenario with dynamic network setup through DHCP. A network, network installation server, and VNC viewer application are required.

[Remote Installation via VNC—PXE Boot and Wake on LAN](#) (page 28)

Consider this approach in a small to medium scenario that should be installed via network and without physical interaction with the installation targets. A network, a network installation server, network boot images, network bootable target hardware, and a VNC viewer application are required.

[Simple Remote Installation via SSH—Static Network Configuration](#) (page 28)

Consider this approach in a small to medium scenario with static network setup. A network, network installation server, and SSH client application are required.

[Remote Installation via SSH—Dynamic Network Configuration](#) (page 29)

Consider this approach in a small to medium scenario with dynamic network setup through DHCP. A network, network installation server, and SSH client application are required.

[Remote Installation via SSH—PXE Boot and Wake on LAN](#) (page 29)

Consider this approach in a small to medium scenario that should be installed via network and without physical interaction with the installation targets. A network, a network installation server, network boot images, network bootable target hardware, and an SSH client application are required.

[Simple Mass Installation](#) (page 30)

Consider this approach for large deployments to identical machines. If configured to use network booting, physical interaction with the target systems is not needed at all. A network, a network installation server, a remote controlling application such as a VNC viewer or an SSH client, and an AutoYaST configuration profile are required. If using network boot, a network boot image and network bootable hardware are required as well.

[Rule-Based Autoinstallation](#) (page 31)

Consider this approach for large deployments to various types of hardware. If configured to use network booting, physical interaction with the target systems is not needed at all. A network, a network installation server, a remote controlling application such as a VNC viewer or an SSH client, and several AutoYaST configuration profiles as well as a rule setup for AutoYaST are required. If using network boot, a network boot image and network bootable hardware are required as well.

Table 2.4 *Simple Remote Installation via VNC—Static Network Configuration*

Installation Source	Network
Preparations	<ul style="list-style-type: none">• Setting up an installation source• Booting from the installation media
Control and Monitoring	Remote: VNC
Best Suited For	small to medium scenarios with varying hardware
Drawbacks	<ul style="list-style-type: none">• Each machine must be set up individually• Physical access is needed for booting
Details	Section 4.1.1, “Simple Remote Installation via VNC—Static Network Configuration” (page 60)

Table 2.5 *Simple Remote Installation via VNC—Dynamic Network Configuration*

Installation Source	Network
Preparations	<ul style="list-style-type: none">• Setting up the installation source• Booting from the installation media
Control and Monitoring	Remote: VNC
Best Suited For	Small to medium scenarios with varying hardware
Drawbacks	<ul style="list-style-type: none">• Each machine must be set up individually• Physical access is needed for booting

Details [Section 4.1.2, “Simple Remote Installation via VNC—Dynamic Network Configuration”](#) (page 61)

Table 2.6 *Remote Installation via VNC—PXE Boot and Wake on LAN*

Installation Source	Network
Preparations	<ul style="list-style-type: none">• Setting up the installation source• Configuring DHCP, TFTP, PXE boot, and WOL• Booting from the network
Control and Monitoring	Remote: VNC
Best Suited For	<ul style="list-style-type: none">• Small to medium scenarios with varying hardware• Completely remote installs; cross-site deployment
Drawbacks	Each machine must be set up manually
Details	Section 4.1.3, “Remote Installation via VNC—PXE Boot and Wake on LAN” (page 63)

Table 2.7 *Simple Remote Installation via SSH—Static Network Configuration*

Installation Source	Network
Preparations	<ul style="list-style-type: none">• Setting up the installation source• Booting from the installation media
Control and Monitoring	Remote: SSH
Best Suited For	<ul style="list-style-type: none">• Small to medium scenarios with varying hardware

	<ul style="list-style-type: none"> • Low bandwidth connections to target
Drawbacks	<ul style="list-style-type: none"> • Each machine must be set up individually • Physical access is needed for booting
Details	Section 4.1.4, “Simple Remote Installation via SSH—Static Network Configuration” (page 64)

Table 2.8 *Remote Installation via SSH—Dynamic Network Configuration*

Installation Source	Network
Preparations	<ul style="list-style-type: none"> • Setting up the installation source • Booting from installation media
Control and Monitoring	Remote: SSH
Best Suited For	<ul style="list-style-type: none"> • Small to medium scenarios with varying hardware • Low bandwidth connections to target
Drawbacks	<ul style="list-style-type: none"> • Each machine must be set up individually • Physical access is needed for booting
Details	Section 4.1.5, “Simple Remote Installation via SSH—Dynamic Network Configuration” (page 65)

Table 2.9 *Remote Installation via SSH—PXE Boot and Wake on LAN*

Installation Source	Network
Preparations	<ul style="list-style-type: none"> • Setting up the installation source

	<ul style="list-style-type: none"> • Configuring DHCP, TFTP, PXE boot, and WOL • Booting from the network
Control and Monitoring	Remote: SSH
Best Suited For	<ul style="list-style-type: none"> • Small to medium scenarios with varying hardware • Completely remote installs; cross-site deployment • Low bandwidth connections to target
Drawbacks	Each machine must be set up individually
Details	Section 4.1.6, “Remote Installation via SSH—PXE Boot and Wake on LAN” (page 67)

Table 2.10 *Simple Mass Installation*

Installation Source	Preferably network
Preparations	<ul style="list-style-type: none"> • Gathering hardware information • Creating AutoYaST profile • Setting up the installation server • Distributing the profile • Setting up network boot (DHCP, TFTP, PXE, WOL) <p><i>or</i></p> <p>Booting the target from installation media</p>
Control and Monitoring	Local or remote through VNC or SSH

Best Suited For	<ul style="list-style-type: none"> • Large scenarios • Identical hardware • No access to system (network boot)
Drawbacks	Applies only to machines with identical hardware
Details	Section 5.1, “Simple Mass Installation” (page 95)

Table 2.11 *Rule-Based Autoinstallation*

Installation Source	Preferably network
Preparations	<ul style="list-style-type: none"> • Gathering hardware information • Creating AutoYaST profiles • Creating AutoYaST rules • Setting up the installation server • Distributing the profile • Setting up network boot (DHCP, TFTP, PXE, WOL) <p><i>or</i></p> <p>Booting the target from installation media</p>
Control and Monitoring	Local or remote through SSH or VNC
Best Suited For	<ul style="list-style-type: none"> • Varying hardware • Cross-site deployments
Drawbacks	Complex rule setup

2.3 Deploying More than 100 Workstations

Most of the considerations brought up for medium installation scenarios in [Section 2.1, “Deploying up to 10 Workstations”](#) (page 23) still hold true for large scale deployments. However, with a growing number of installation targets, the benefits of a fully automated installation method outweigh its disadvantages.

It pays off to invest a considerable amount of time to create a sophisticated rule and class framework in AutoYaST to match the requirements of a huge deployment site. Not having to touch each target separately can save you a tremendous amount of time depending on the scope of your installation project.

Installation with YaST

The installation of a SUSE Linux Enterprise system is performed by the system assistant YaST. Adjust the default settings as described here to install a system that meets your needs. Background information is provided where appropriate.

3.1 System Start-Up for Installation

Insert the first SUSE Linux Enterprise CD or the DVD into the drive. Then reboot the computer to start the installation program from the medium in the drive.

3.1.1 Boot Options

Boot options other than CD or DVD exist and can be used if problems arise booting from CD or DVD. These options are described in [Table 3.1, “Boot Options”](#) (page 33).

Table 3.1 *Boot Options*

Boot Option	Description
CD-ROM	This is the easiest boot option. This option can be used if the system has a local CD-ROM drive that is supported by Linux.
Floppy	The images for generating boot floppies are located on CD 1 in the <code>/boot</code> directory. A README is available in the same directory.

Boot Option	Description
PXE or BOOTP	This must be supported by the system's BIOS or firmware and a boot server must be available in the network. This task can also be handled by another SUSE Linux Enterprise system.
Hard Disk	SUSE Linux Enterprise can also be booted from the hard disk. To do this, copy the kernel (<code>linux</code>) and the installation system (<code>initrd</code>) from the directory <code>/boot/loader</code> on CD 1 to the hard disk and add the appropriate entry to the boot loader.

3.1.2 Selecting the Source of the Installation Data

You can install from a local installation source, such as the SUSE Linux Enterprise CDs or DVD or get the installation sources from an FTP, HTTP, or NFS server. Any of these approaches require physical access to the system to install and user interaction during the installation. The installation procedure is basically the same, no matter which installation source or method you prefer.

Installing from the SUSE Linux Enterprise Media

Install from physical boot media (your SUSE Linux Enterprise media kit) as follows:

- 1 Insert the media into your CD or DVD drive.
- 2 Reboot the system.
- 3 At the boot screen, select *Installation* and follow the instructions given in [Section 3.2, “The Boot Screen”](#) (page 36) and the following sections.

Installing from a Network Server Using SLP

If your network setup supports OpenSLP and your network installation source has been configured to announce itself via OpenSLP (see [Section 4.2, “Setting Up the Server](#)

[Holding the Installation Sources](#)” (page 68)), proceed as follows to install SUSE Linux Enterprise.

- 1 Set up an installation server as described in [Section 4.2, “Setting Up the Server Holding the Installation Sources](#)” (page 68).
- 2 Insert the first CD of the media kit into the CD-ROM drive and reboot the machine.
- 3 At the boot screen, select *Installation*, press `[F3]` and `[F4]` then select *SLP*.

The installation program retrieves the location of the network installation source using OpenSLP and configures the network connection with DHCP. If the DHCP network configuration fails, you are prompted to enter the appropriate parameters manually. The installation then proceeds normally.

- 4 Finish the installation as if you had chosen to install from physical media.

Installing from a Network Server

To perform a manual installation using a network installation source, proceed as follows:

- 1 Set up an installation server as described in [Section 4.2, “Setting Up the Server Holding the Installation Sources](#)” (page 68).
- 2 Insert the first CD or DVD of the media kit into the corresponding drive then reboot the machine.
- 3 At the boot screen, select *Installation* and use the boot options prompt to pass additional information, such as:

- Location of the installation server:

```
install=protocol:inst_source
```

Replace *protocol* with the protocol prefix for the service used by the installation server (*nfs*, *http*, or *ftp*). Replace *inst_source* with the IP address of the installation server.

- Network configuration parameters if your setup does not support DHCP configuration (see [Section 4.4.3, “Using Custom Boot Options](#)” (page 89) for reference).

- 4 Press `[Enter]` to boot for installation. If no network parameters have been specified at the boot options prompt, the installation routines try to set up the network using DHCP. If this fails, you are prompted for these parameters. After you have provided them, the installation proceeds.
- 5 Finish the installation as if you had chosen to install from the physical media.

3.2 The Boot Screen

The boot screen displays a number of options for the installation procedure. *Boot from Hard Disk* boots the installed system. This item is selected by default, because the CD is often left in the drive. To install the system, select one of the installation options with the arrow keys. The relevant options are:

Installation

The normal installation mode. All modern hardware functions are enabled.

Installation—ACPI Disabled

If the normal installation fails, this might be due to the system hardware not supporting ACPI (advanced configuration and power interface). If this seems to be the case, use this option to install without ACPI support.

Installation—Safe Settings

Boots the system with the DMA mode (for CD-ROM drives) and power management functions disabled. Experts can also use the command line to enter or change kernel parameters.

Use the function keys indicated in the bar at the bottom of the screen to change a number of installation settings.

`[F1]`

Get context-sensitive help for the active element of the boot screen.

`[F2]`

Select the display language for the installation.

`[F3]`

See other options that can be set for installation.

After you press **F3**, some other options can be set:

F3 Select various graphical display modes for the installation. Select the text mode if the graphical installation causes problems.

F4 Normally, the installation is performed from the inserted installation medium. Here, select other sources, like FTP or NFS servers. If the installation is carried out in a network with an SLP server, select one of the installation sources available on the server with this option. Information about SLP is available in [Chapter 30, *SLP Services in the Network*](#) (page 631).

F5 Use this to tell the system that you have an optional disk with a driver update for SUSE Linux Enterprise Desktop. You will be asked to insert the update disk at the appropriate point in the installation process.

A few seconds after starting the installation, SUSE Linux Enterprise loads a minimal Linux system to run the installation procedure. If you want to know what is going on during the boot process, press **Esc** to see messages and copyright notices scroll by. At the end of the loading process, the YaST installation program starts. After a few more seconds, the screen should display the graphical installer.

The actual installation of SUSE Linux Enterprise begins at this point. All YaST screens have a common layout. All buttons, entry fields, and lists can be accessed with the mouse or the keyboard. If your mouse pointer does not move, the mouse has not been detected automatically. In this case, use the keyboard for the time being. Navigation with the keyboard is similar to the description in [Section 7.11.1, “Navigation in Modules”](#) (page 171).

3.3 Language Selection

YaST and SUSE Linux Enterprise in general can be configured to use different languages according to your needs. The language selected here is also used for the keyboard layout. In addition, YaST uses the language setting to guess a time zone for the system clock. These settings can be modified later along with the selection of secondary languages to install on your system. If your mouse does not work, select the language with the

arrow keys and press **Tab** until *Next* is highlighted. Then press **Enter** to confirm your language selection.

3.4 License Agreement

Read the license agreement that is displayed on screen thoroughly. If you agree to the terms, choose *Yes, I Agree to the License Agreement* and click *Next* to confirm your selection. If you do not agree to the license agreement, you are not allowed to install SUSE Linux Enterprise and the installation terminates.

3.5 System Analysis

Select *New installation* or *Update an existing system*. Updating is only possible if a SUSE Linux Enterprise system is already installed. When a SUSE Linux Enterprise system is already installed, use *Other* to access two advanced options: boot the installed system with *Boot installed system* or, if the installed system fails to boot, you can try to fix the problem with *Repair installed system*. If no SUSE Linux Enterprise system is installed, you can only perform a new installation.

The following sections describe the procedure of installing a new system. Find detailed instructions for a system update in [Section 7.3.7, “Updating the System”](#) (page 137). Find a description of the system repair options in [Section “Using YaST System Repair”](#) (page 803).

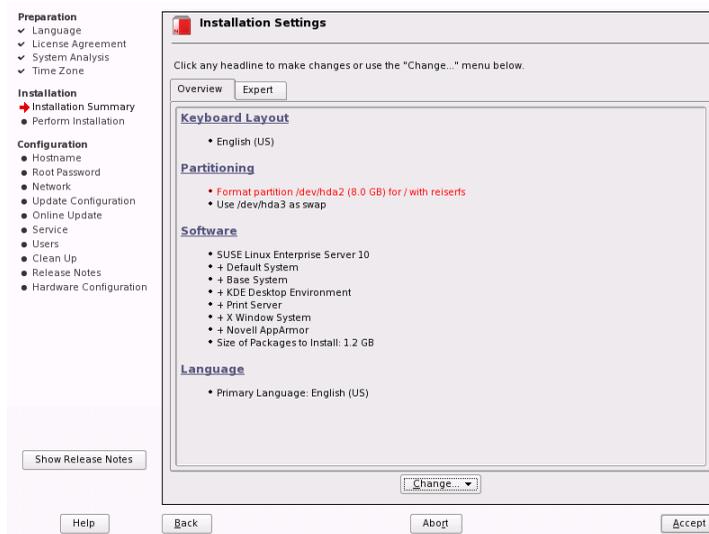
3.6 Time Zone

In this dialog, select your region and time zone from the lists. During installation, both are preselected according to the selected installation language. Choose between *Local Time* and *UTC (GMT)* under *Hardware Clock Set To*. The selection depends on how the BIOS hardware clock is set on your machine. If it is set to GMT, which corresponds to UTC, your system can rely on SUSE Linux Enterprise to switch from standard time to daylight saving time and back automatically. Click *Change* to set the current date and time. When finished, click *Next* to continue the installation.

3.7 Installation Summary

After a thorough system analysis, YaST presents reasonable suggestions for all installation settings. The options that sometimes need manual intervention in common installation situations are presented under the *Overview* tab. Find more special options in the *Expert* tab. After configuring any of the items presented in these dialogs, you are always returned to the summary window, which is updated accordingly. The individual settings are discussed in the following sections.

Figure 3.1 Installation Settings



3.7.1 Partitioning

In most cases, YaST proposes a reasonable partitioning scheme that can be accepted without change. YaST can also be used to customize the partitioning. This section describes the necessary steps.

Partition Types

Every hard disk has a partition table with space for four entries. An entry in the partition table can correspond to a primary partition or an extended partition. Only one extended partition entry is allowed, however.

A primary partition simply consists of a continuous range of cylinders (physical disk areas) assigned to a particular operating system. With primary partitions only, you would be limited to four partitions per hard disk, because more do not fit in the partition table. This is why extended partitions are used. Extended partitions are also continuous ranges of disk cylinders, but an extended partition may itself be subdivided into *logical partitions*. Logical partitions do not require entries in the partition table. In other words, an extended partition is a container for logical partitions.

If you need more than four partitions, create an extended partition as the fourth partition or earlier. This extended partition should span the entire remaining free cylinder range. Then create multiple logical partitions within the extended partition. The maximum number of logical partitions is 15 on SCSI, SATA, and Firewire disks and 63 on (E)IDE disks. It does not matter which types of partitions are used for Linux. Primary and logical partitions both work fine.

Required Disk Space

YaST normally proposes a reasonable partitioning scheme with sufficient disk space. If you want to implement your own partitioning scheme, consider the following recommendations concerning the requirements for different system types.

Minimal System: 500 MB

No graphical interface (X Window System) is installed, which means that only console applications can be used. Also, only a very basic selection of software is installed.

The partitions to create depend on the available space. The following are some basic partitioning guidelines:

Up to 4 GB:

One partition for the swap space and one root partition (/). In this case, the root partition must allow for those directories that often reside on their own partitions if more space is available.

4 GB or More:

A swap partition, a root partition (1 GB), and one partition each for the following directories as needed: `/usr` (4 GB or more), `/opt` (4 GB or more), and `/var` (1 GB). If you do not want to have separate partitions for these directories, add the suggested disk space to the root partition. The rest of the available space can be used for `/home`.

Depending on the hardware, it might also be useful to create a boot partition (`/boot`) to hold the boot mechanism and the Linux kernel. This partition should be located at the start of the disk and should be at least 8 MB or one cylinder. As a rule of thumb, always create such a partition if it was included in YaST's original proposal. If you are unsure about this, create a boot partition to be on the safe side.

You should also be aware that some (mostly commercial) programs install their data in `/opt`. Therefore, either create a separate partition for `/opt` or make the root partition large enough.

Partitioning with YaST

When you select the partitioning item in the suggestion window for the first time, the YaST partitioning dialog displays the proposed partition settings. Accept these current settings as they are or change them before continuing. Alternatively, discard all the settings and start over from scratch.

Nothing in the partitioning setup is changed if you select *Accept Proposal*. If you select *Base Partition Setup on This Proposal*, the *Expert Partitioner* opens. It allows tweaking the partition setup in every detail. This dialog is explained in [Section 7.5.6, “Partitioner”](#) (page 150). The original setup as proposed by YaST is offered there as a starting point.

Selecting *Create Custom Partition Setup* opens the dialog for hard disk selection. Use the list to choose among the existing hard disks on your system. SUSE Linux Enterprise will be installed on the disk selected in this dialog.

The next step is to determine whether the entire disk should be used (*Use Entire Hard Disk*) or whether to use any existing partitions (if available) for the installation. If a Windows operating system was found on the disk, you are asked whether to delete or resize the partition. Before doing so, read [Section “Resizing a Windows Partition”](#) (page 42). If desired, go to the *Expert Partitioner* dialog to create a custom partition setup as described in [Section 7.5.6, “Partitioner”](#) (page 150).

WARNING: Using the Entire Hard Disk for Installation

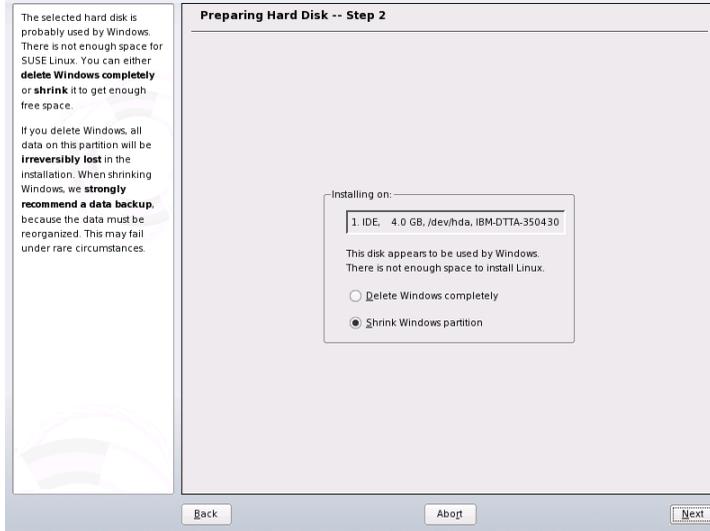
If you choose *Use Entire Hard Disk*, all existing data on that disk is completely erased later in the installation process and is then lost.

YaST checks during the installation whether the disk space is sufficient for the software selection made. If not, YaST automatically changes the software selection. The proposal dialog displays a notice to inform you about this. As long as there is sufficient disk space available, YaST simply accepts your settings and partitions the hard disk accordingly.

Resizing a Windows Partition

If a hard disk containing a Windows FAT or NTFS partition is selected as the installation target, YaST offers to delete or shrink this partition. In this way, you can install SUSE® Linux Enterprise even if there is currently not enough space on the hard disk. This functionality is especially useful if the selected hard disk contains only one Windows partition that covers the entire hard disk. This is sometimes the case on computers where Windows comes preinstalled. If YaST sees that there is not enough space on the selected hard disk, but that space could be made available by deleting or shrinking a Windows partition, it presents a dialog in which to choose one of these two options.

Figure 3.2 Possible Options for Windows Partitions



If you select *Delete Windows Completely*, the Windows partition is marked for deletion and the space is used for the installation of SUSE Linux Enterprise.

WARNING: Deleting Windows

If you delete Windows, all data will be lost beyond recovery as soon as the formatting starts.

To shrink the Windows partition, interrupt the installation and boot Windows to prepare the partition from there. Although this step is not strictly required for FAT partitions, it speeds up the resizing process and also makes it safer. These steps are vital for NTFS partitions.

FAT File System

In Windows, first run scandisk to make sure that the FAT partition is free of lost file fragments and crosslinks. After that, run defrag to move files to the beginning of the partition. This accelerates the resizing procedure in Linux.

If you have optimized virtual memory settings for Windows so a contiguous swap file is used with the same initial (minimum) and maximum size limit, consider another step. With these Windows settings, the resizing might split the swap file into

many small parts scattered all over the FAT partition. Also, the entire swap file would need to be moved during the resizing, which makes the process rather slow. It is therefore useful to disable these Windows optimizations for the time being and reenabling them after the resizing has been completed.

NTFS File System

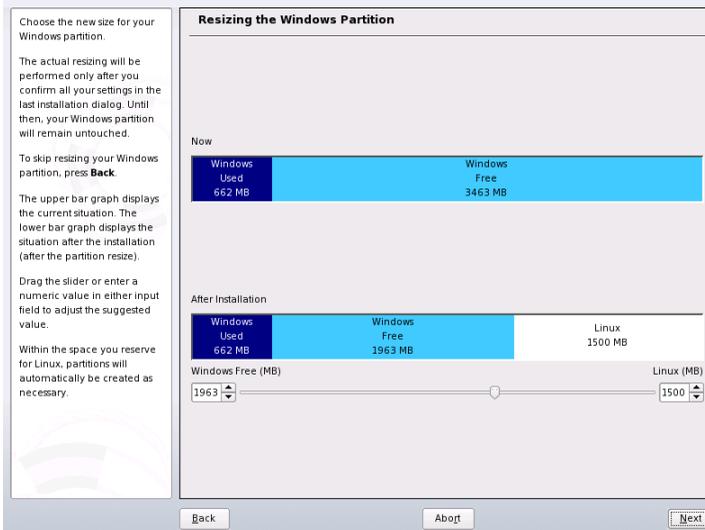
In Windows, run scandisk and defrag to move the files to the beginning of the hard disk. In contrast to the FAT file system, you must perform these steps. Otherwise the NTFS partition cannot be resized.

IMPORTANT: Disabling the Windows Swap File

If you operate your system with a permanent swap file on an NTFS file system, this file may be located at the end of the hard disk and remain there despite defrag. Therefore, it may be impossible to shrink the partition sufficiently. In this case, temporarily deactivate the swap file (the virtual memory in Windows). After the partition has been resized, reconfigure the virtual memory.

After these preparations, return to the Linux partitioning setup and select *Shrink Windows Partition*. After a quick check of the partition, YaST opens a dialog with a suggestion for resizing the Windows partition.

Figure 3.3 Resizing the Windows Partition



The first bar graph shows how much disk space is currently occupied by Windows and how much space is still available. The second bar graph shows how the space would be distributed after the resizing, according to YaST's current proposal. See [Figure 3.3, “Resizing the Windows Partition”](#) (page 44). Accept the proposed settings or use the slider to change the partition sizing (within certain limits).

If you leave this dialog by selecting *Next*, the settings are stored and you are returned to the previous dialog. The actual resizing takes place later, before the hard disk is formatted.

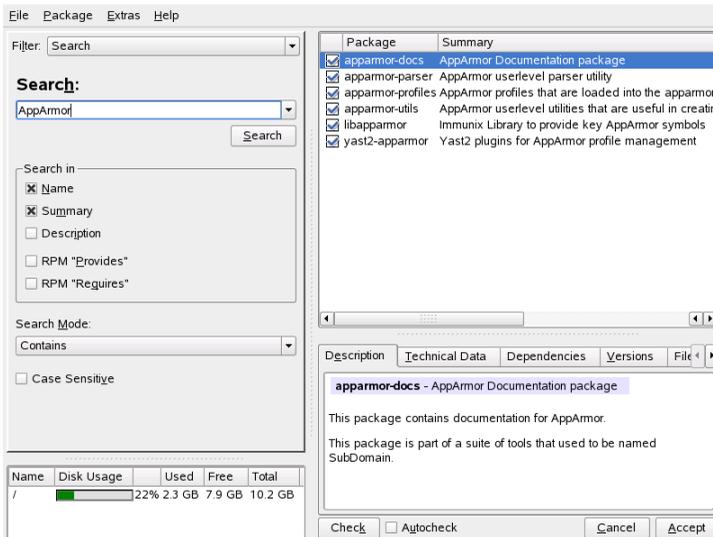
IMPORTANT: Windows Systems Installed on NTFS Partitions

By default, the Windows versions NT, 2000, and XP use the NTFS file system. Unlike FAT file systems, NTFS file systems can only be read from Linux. This means you can read your Windows files from Linux, but you cannot edit them. If you want write access to your Windows data and do not need the NTFS file system, reinstall Windows on a FAT32 file system. In this case, you will have full access to your Windows data from SUSE Linux Enterprise.

3.7.2 Software

SUSE Linux Enterprise contains a number of software packages for various application purposes. Click *Software* in the suggestion window to start the software selection and modify the installation scope according to your needs. Select your categories from the list in the middle and see the description in the right window. Each category contains a number of software packages that meet most requirements for that category. For more detailed selection of software packages to install, select *Details* to switch to the YaST Package Manager.

Figure 3.4 *Installing and Removing Software with the YaST Package Manager*



Changing the Installation Scope

If you have specific software needs, modify the current selection with the package manager, which greatly eases this task. The package manager offers various filter criteria to simplify selection from the numerous packages in SUSE Linux Enterprise.

The filter selection box is located at the top left under the menu bar. The default filter is *Patterns*. Patterns install packages based on the intended use or task of your system. The groups included in the current system type are preselected. Click the check boxes to select or deselect groups for installation.

The right part of the window displays a table listing the individual packages included in the current group. The table column furthest to the left shows the current status of each package. Two status flags are especially relevant for the installation: *Install* (the box in front of the package name is checked) and *Do Not Install* (the box is empty). To select or deselect individual software packages, click the status box until the desired status is displayed. Alternatively, right-click the package line to access a pop-up menu listing all the possible status settings.

Other Filters

Click the filter selection box to view the other possible filters. The selection according to *Package Groups* can also be used for the installation. This filter sorts the program packages by subjects in a tree structure to the left. The more you expand the branches, the more specific the selection of packages is and the fewer packages are displayed in the list of associated packages to the right.

Use *Search* to search for a specific package. This is explained in detail in [Section 7.3.1, “Installing and Removing Software”](#) (page 127).

Package Dependencies and Conflicts

You cannot simply install any combination of software packages. The different software packages must be compatible. Otherwise they might interfere with each other and cause conflicts that affect the system as a whole. Therefore, you may see alerts about unresolved package dependencies or conflicts after selecting or deselecting software packages in this dialog. If installing SUSE Linux Enterprise for the first time or if you do not understand the alerts, read [Section 7.3.1, “Installing and Removing Software”](#) (page 127), which provides detailed information about the operation of the package manager.

Exiting the Software Selection

When satisfied with your software selection and all package dependencies or conflicts are resolved, click *Accept* to apply your changes and exit the module. During the installation, the changes are recorded internally and applied later when the actual installation starts.

3.7.3 Language

The language was selected at the beginning of the installation as described in [Section 3.3, “Language Selection”](#) (page 37). However, you can change this setting here and also select any additional languages to install on your system. In the upper part of this dialog, select the primary language. This is the language that will be activated after installation. Adapt your keyboard and time zone settings to the selected primary language by selecting those options, if desired. Optionally, use *Details* to set the language for the user `root`. There are three options:

ctype only

The value of the variable `LC_CTYPE` in the file `/etc/sysconfig/language` is adopted for the user `root`. This sets the localization for language-specific function calls.

yes

The user `root` has the same language settings as the local user.

no

The language settings for the user `root` are not affected by the language selection. All `locale` variables are unset.

Make the setting for the locale explicitly with *Detailed Locale Setting*.

The list in the lower part of the language dialog allows for selection of additional languages to install. For all the languages selected in this list, YaST checks if there are any language-specific packages for any packages in your current software selection. If so, these packages are installed.

Click *Accept* to complete the configuration.

3.7.4 System

This dialog presents all the hardware information YaST could obtain about your computer. Select any item in the list and click *Details* to see detailed information about the selected item. You may also add PCI IDs to device drivers with this dialog.

3.7.5 Keyboard Layout

Select the keyboard layout from the list. By default, the layout corresponds to the selected language. After changing the layout, test the characters that are special to the selected language layout to make sure that the selection is correct. To set special options regarding keyboard behavior, click *Expert Settings*. Find more information about that in [Section 7.4.9, “Keyboard Layout”](#) (page 143). When finished, click *Accept* to return to the installation settings dialog.

3.7.6 Booting

During the installation, YaST proposes a boot configuration for your system. Normally, you can leave these settings unchanged. However, if you need a custom setup, modify the proposal for your system.

One possibility is to configure the boot mechanism to rely on a special boot floppy. Although this has the disadvantage that it requires the floppy to be in the drive when booting, it leaves an existing boot mechanism untouched. Normally this should not be necessary, however, because YaST can configure the boot loader to boot other existing operating systems as well. Another possibility with the configuration is to change the location of the boot mechanism on the hard disk.

To change the boot configuration proposed by YaST, select *Booting* to open a dialog in which to change many details of the boot mechanism. For information, read [Section 17.3, “Configuring the Boot Loader with YaST”](#) (page 397).

3.7.7 Default Runlevel

SUSE Linux Enterprise can boot to different runlevels. Normally there should be no need to change anything here, but if necessary set the default runlevel with this dialog. Refer to [Section 7.5.11, “System Services \(Runlevel\)”](#) (page 156) for information about runlevel configuration.

3.7.8 Time Zone

In this dialog, change your region and time zone by selecting them from the lists. Choose between *Local Time* and *UTC (GMT)* under *Hardware Clock Set To*. The selection depends on how the BIOS hardware clock is set on your machine. If it is set to GMT, which corresponds to UTC, your system can rely on SUSE Linux Enterprise to switch from standard time to daylight saving time and back automatically. Click *Change* to set the current date and time. When finished, click *Accept* to return to the installation settings dialog.

3.7.9 Performing the Installation

After making all installation settings, click *Accept* in the suggestion window to begin the installation. Confirm with *Install* in the dialog that opens. The installation usually takes between 15 and 30 minutes, depending on the system performance and the software selected. As soon as all packages are installed, YaST boots into the new Linux system, after which you can configure the hardware and set up system services.

3.8 Configuration

After completing the basic system setup and the installation of all selected software packages, provide a password for the account of the system administrator (the `root` user). You can then configure your Internet access and network connection. With a working Internet connection, you can perform an update of the system as part of the installation. You can also configure an authentication server for centralized user administration in a local network. Finally, configure the hardware devices connected to the machine.

3.8.1 Hostname

The hostname is the computer's name in the network. The fully qualified domain name, needed here, includes the name of the domain to which the computer belongs. Each server and client in the network should have a unique hostname.

If you are located in a local network, you might receive your hostname over DHCP, in which case you should not modify the name. To receive the hostname over DHCP, select *Change Hostname via DHCP*.

3.8.2 root Password

`root` is the name of the superuser, the administrator of the system. Unlike regular users, which may or may not have permission to do certain things on the system, `root` has unlimited power to do anything: change the system configuration, install programs, and set up new hardware. If users forget their passwords or have other problems with the system, `root` can help. The `root` account should only be used for system admin-

istration, maintenance, and repair. Logging in as `root` for daily work is rather risky: a single mistake could lead to irretrievable loss of many system files.

For verification purposes, the password for `root` must be entered twice. Do not forget the `root` password. Once entered, this password cannot be retrieved.

3.8.3 Network Configuration

You can now choose whether to use NetworkManager or the traditional method to manage all your network devices. NetworkManager is the new tool enabling automatic connection establishment with minimal user intervention. It is ideal for mobile computing. Also configure the network devices of your system and make security settings, for example, for a firewall or proxy. To configure your network hardware at this stage, refer to [Section 29.4, “Configuring a Network Connection with YaST”](#) (page 594). Otherwise, select *Skip Configuration* and click *Next*. Network hardware can also be configured after the system installation has been completed.

NOTE: Network Devices and Update

If you skip the network device configuration, your system will be offline and unable to retrieve any available updates or include them in the installation.

As well as device configuration, configure network accessibility–related settings:

Firewall Configuration

When you connect to a network, a firewall is started automatically on the configured interface. The configuration proposal for the firewall is updated automatically every time the configuration of the interfaces or services is modified. To adapt the automatic settings to your own preferences, click *Change* → *Firewall*. In the dialog that opens, determine whether the firewall should be started. If you do not want the firewall to be started, select the appropriate option and exit the dialog. To start and configure the firewall, click *Next* for a series of dialogs similar to those described in [Section 38.4.1, “Configuring the Firewall with YaST”](#) (page 714).

VNC Remote Administration

To administer your machine remotely by VNC, click *Change* → *VNC Remote Administration*, enable remote administration, and open the port in the firewall. If you have multiple network devices and want to select on which to open the port, click

Firewall Details and select the network device. You can also use SSH, a more secure option, for remote administration.

Proxy

If you have a proxy server in your network to control access to the network, enter the server name and all other required information to enable access to the Internet.

Internet Connection Test

If you have configured an Internet connection, you can test it now. For this purpose, YaST establishes a connection to the SUSE Linux Enterprise server and checks if any product updates are available for your version of SUSE Linux Enterprise. If there are such updates, they can be included in the installation. Also, the latest release notes are downloaded. You can read them at the end of the installation.

To start the test, select *Yes, Test Connection to the Internet* and click *Next*. In the next dialog, view the progress of the test and the results of the test. If the test fails, click *Back* to return in the previous dialog and correct the configuration or skip the test. If you need more information about the test process, click *View Logs*.

If you do not want to test the connection at this point, select *No, Skip This Test* then *Next*. This also skips downloading product updates and release notes.

If you have multiple network interfaces in your system, verify that the the right card is used to connect to the Internet. To do so, click *Change device*.

3.8.4 Customer Center

To get technical support and product updates, first register and activate your product. *Novell Customer Center Configuration* provides assistance for doing so. If you are offline or want to skip this step, select *Configure Later*.

In *Include for Convenience*, select whether to obtain some of the necessary information from your system. This simplifies the registration process. If you want to see what is required to register your system or what happens with your data, use *Details*.

3.8.5 Online Update

If YaST was able to connect to the SUSE Linux Enterprise servers, select whether to perform a YaST online update. If there are any patched packages available on the servers, download and install them now to fix known bugs or security issues.

3.8.6 Users

This step has two parts. In the first part, choose the user authentication method. The second part depends on the selected authentication method.

User Authentication

If network access was configured successfully during the previous steps of the installation, you now have four possibilities for managing user accounts on your system.

Local (/etc/passwd)

Users are administered locally on the installed host. This is a suitable option for stand-alone workstations. User data is managed by the local file `/etc/passwd`. All users who are entered in this file can log in to the system even if no network is available.

LDAP

Users are administered centrally on an LDAP server for all systems in the network.

NIS

Users are administered centrally on a NIS server for all systems in the network.

Windows Domain

SMB authentication is often used in mixed Linux and Windows networks.

NOTE: Content of the Authentication Menu

If you use the custom package selection and one or more authentication methods are missing from the menu, you probably did not select the packages required for it.

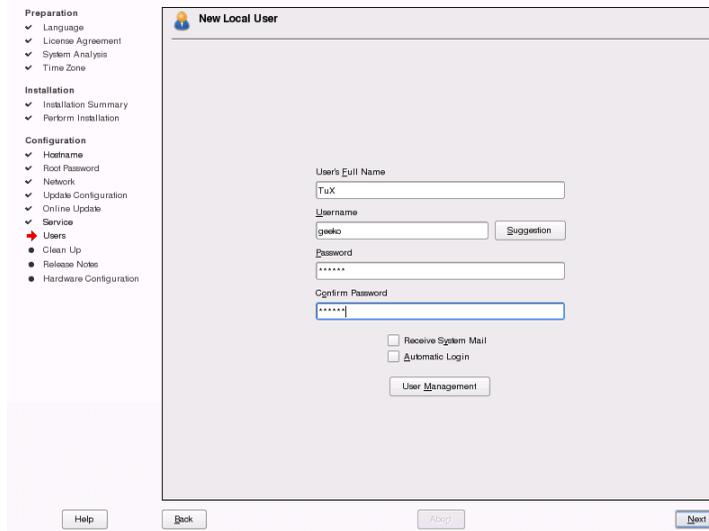
If all requirements are met, YaST opens a dialog in which to select the user administration method. If you do not have the necessary network connection, create local user accounts.

Creating Local User Accounts

Linux is an operating system that allows several users to work on the same system at the same time. Each user needs a user account to log in to the system. By having user accounts, the system gains a lot in terms of security. For instance, regular users cannot change or delete files needed for the system to work properly. At the same time, the personal data of a given user cannot be modified, viewed, or tampered with by other users. Users can set up their own working environments and always find them unchanged when logging back in.

If you decide against using an authentication server for user authentication, create local users. Any data related to user accounts (name, login, password, etc.) is stored and managed on the installed system.

Figure 3.5 *Entering the Username and Password*



A local user account can be created using the dialog shown in [Figure 3.5, “Entering the Username and Password”](#) (page 54). After entering the first name and last name,

specify a username (login). Click *Suggestion* for the system to generate a username automatically.

Finally, enter a password for the user. Reenter it for confirmation (to ensure that you did not type something else by mistake).

To provide effective security, a password should be between five and eight characters long. The maximum length for a password is 128 characters. However, if no special security modules are loaded, only the first eight characters are used to discern the password. Passwords are case-sensitive. Special characters like umlauts are not allowed. Other special characters (7-bit ASCII) and the digits 0 to 9 are allowed.

Two additional options are available for local users:

Receive System Messages via E-Mail

Checking this box sends the user messages created by the system services. These are usually only sent to `root`, the system administrator. This option is useful for the most frequently used account, because it is highly recommended to log in as `root` only in special cases.

Automatic Login

This option is only available if KDE is used as the default desktop. It automatically logs the current user into the system when it starts. This is mainly useful if the computer is operated by only one user.

WARNING: Automatic Login

With the automatic login enabled, the system boots straight into your desktop with no authentication at all. If you store sensitive data on your system, you should not enable this option if the computer can also be accessed by others.

Click *User Management* to create more than one user. Refer to [Section 7.9.1, “User Management”](#) (page 164) for more information about user management.

Configuring the Host as an LDAP Client

To implement user administration by LDAP, configure an LDAP client in the next step. LDAP authentication relies on a central LDAP server located in your network providing the authentication data. This task can be handled by a SUSE Linux Enterprise Server machine.

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. Enter the IP address of the LDAP server to use and the LDAP base DN to select the search base on the LDAP server. To retrieve the base DN automatically, click *Fetch DN*. YaST then checks for any LDAP database on the specified server address. Choose the appropriate base DN from the search results given by YaST. If TLS or SSL protected communication with the server is required, select *LDAP TLS/SSL*. If the LDAP server still uses LDAPv2, explicitly enable the use of this protocol version by selecting *LDAP Version 2*. Select *Start Automounter* to mount remote directories on your client, such as a remotely managed home directory. Click *Finish* to apply your settings. LDAP client configuration is discussed in further detail in [Section 34.3, “Configuring an LDAP Client with YaST”](#) (page 658).

Configuring the Host as a NIS Client

To implement user administration by NIS, configure a NIS client in the next step. NIS authentication relies on a central NIS server located in your network providing the authentication data. This task can be handled by a SUSE Linux Enterprise Server machine.

In the NIS client dialog, first select whether the host has a static IP address or gets one with DHCP. If you select DHCP, you cannot specify a NIS domain or NIS server address, because these are provided by the DHCP server. If a static IP address is used, specify the NIS domain and the NIS server manually.

To search for NIS servers broadcasting in the network, check the relevant option. You can also specify several NIS domains and set a default domain. For each domain, select *Edit* to specify several server addresses or enable the broadcast function on a per-domain basis.

In the expert settings, use *Answer Remote Hosts* to allow other network hosts to query which server your client is using. If you activate *Broken Server*, responses from servers on unprivileged ports are also accepted. For more information, refer to the man page of `yplibind`.

Configuring the Host as a Windows Domain Member

To implement user administration using a Samba or Windows server, configure a Samba client in the next step. A SUSE Linux Enterprise Server machine or a Windows server can be set up as a Samba server.

In the *Windows Domain Membership* dialog, enter the NT or Active Directory domain or Samba workgroup to join or use *Browse* to select from a list of available domains. Select *Create Home Directory on Login* if you want to create home directories for any user logging in to the domain from your local machine. Click *Finish* to apply your settings and provide the necessary credentials. For further details on SUSE Linux Enterprise support for AD domains, see [Chapter 11, Active Directory Support](#) (page 285).

3.8.7 Cleanup

This step does not require any user interaction. The installation program launches the SuSEconfig script to write the system configuration. Depending on the CPU and the amount of memory, this process can take some time.

3.8.8 Release Notes

After completing the user authentication setup, YaST displays the release notes. Reading them is advised because they contain important up-to-date information that was not available when the manuals were printed. If you have installed update packages, read the most recent version of the release notes, as fetched from SUSE Linux Enterprise's servers.

3.8.9 Hardware Configuration

At the end of the installation, YaST opens a dialog for the configuration of the graphics card and other hardware components connected to the system. Click the individual components to start the hardware configuration. For the most part, YaST detects and configures the devices automatically.

You can skip any peripheral devices and configure them later. To skip the configuration, select *Skip Configuration* and click *Next*.

However, you should configure the graphics card right away. Although the display settings as autoconfigured by YaST should be generally acceptable, most users have very strong preferences as far as resolution, color depth, and other graphics features are concerned. To change these settings, select the respective item and set the values as desired. To test your new configuration, click *Test the Configuration*.

3.8.10 Completing Installation

After a successful installation, YaST shows the *Installation Completed* dialog. In this dialog, select whether to clone your newly installed system for AutoYaST. To clone your system, select *Clone This System for AutoYaST*. The profile of the current system is stored in `/root/autoyast.xml`.

AutoYaST is a system for installing one or more SUSE Linux Enterprise systems automatically without user intervention. AutoYaST installations are performed using a control file with installation and configuration data.

Finish the installation of SUSE Linux Enterprise with *Finish* in the final dialog.

3.9 Graphical Login

SUSE Linux Enterprise is now installed. Unless you enabled the automatic login function, you should see the graphical login on your screen where you can enter your username and password to log in to the system. If automatic login is activated, the desktop starts automatically.

4

Remote Installation

SUSE® Linux Enterprise can be installed in several different ways. As well as the usual CD or DVD installation covered in [Chapter 3, *Installation with YaST*](#) (page 33), you can choose from various network-based approaches or even take a completely hands-off approach to the installation of SUSE Linux Enterprise.

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 Enterprise 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.

4.1 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 `telinit 3`, and start `SaX2` to configure the graphics hardware as described in [Section 22.1, “X11 Setup with SaX2”](#) (page 465).

4.1.1 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 3, *Installation with YaST*](#) (page 33).

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 4.2, “Setting Up the Server Holding the Installation Sources”](#) (page 68). Choose an NFS, HTTP, or FTP network server. For an SMB installation source, refer to [Section 4.2.5, “Managing an SMB Installation Source”](#) (page 76).

- 2 Boot the target system using the first CD or DVD of the SUSE Linux Enterprise 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 4.4, “Bootting the Target System for Installation”](#) (page 87).

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 4.5.1, “VNC Installation”](#) (page 92).
- 5 Perform the installation as described in [Chapter 3, *Installation with YaST*](#) (page 33). Reconnect to the target system after it reboots for the final part of the installation.
- 6 Finish the installation.

4.1.2 Simple Remote Installation via VNC—Dynamic Network Configuration

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, or 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 4.2, “Setting Up the Server Holding the Installation Sources”](#) (page 68). Choose an NFS, HTTP, or FTP network server. For an SMB installation source, refer to [Section 4.2.5, “Managing an SMB Installation Source”](#) (page 76).

2 Boot the target system using the first CD or DVD of the SUSE Linux Enterprise 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 4.4, “Booting the Target System for Installation”](#) (page 87).

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 4.5.1, “VNC Installation”](#) (page 92).

5 Perform the installation as described in [Chapter 3, *Installation with YaST*](#) (page 33). Reconnect to the target system after it reboots for the final part of the installation.

6 Finish the installation.

4.1.3 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 4.2, “Setting Up the Server Holding the Installation Sources”](#) (page 68). Choose an NFS, HTTP, or FTP network server or configure an SMB installation source as described in [Section 4.2.5, “Managing an SMB Installation Source”](#) (page 76).
- 2** Set up a TFTP server to hold a boot image that can be pulled by the target system. This is described in [Section 4.3.2, “Setting Up a TFTP Server”](#) (page 79).
- 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 4.3.1, “Setting Up a DHCP Server”](#) (page 77).
- 4** Prepare the target system for PXE boot. This is described in further detail in [Section 4.3.5, “Preparing the Target System for PXE Boot”](#) (page 86).

- 5 Initiate the boot process of the target system using Wake on LAN. This is described in [Section 4.3.7, “Wake on LAN”](#) (page 86).
- 6 On the controlling workstation, open a VNC viewing application or Web browser and connect to the target system as described in [Section 4.5.1, “VNC Installation”](#) (page 92).
- 7 Perform the installation as described in [Chapter 3, *Installation with YaST*](#) (page 33). Reconnect to the target system after it reboots for the final part of the installation.
- 8 Finish the installation.

4.1.4 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 3, *Installation with YaST*](#) (page 33).

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 working SSH client software
- Physical boot medium (CD, DVD, or 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 4.2, “Setting Up the Server Holding the Installation Sources”](#) (page 68). Choose an NFS, HTTP, or FTP network server. For an SMB installation source, refer to [Section 4.2.5, “Managing an SMB Installation Source”](#) (page 76).
- 2 Boot the target system using the first CD or DVD of the SUSE Linux Enterprise 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 4.4.3, “Using Custom Boot Options”](#) (page 89).

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 94).
- 5 Perform the installation as described in [Chapter 3, *Installation with YaST*](#) (page 33). Reconnect to the target system after it reboots for the final part of the installation.
- 6 Finish the installation.

4.1.5 Simple Remote Installation via SSH—Dynamic Network Configuration

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 working SSH client software
- 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 4.2, “Setting Up the Server Holding the Installation Sources”](#) (page 68). Choose an NFS, HTTP, or FTP network server. For an SMB installation source, refer to [Section 4.2.5, “Managing an SMB Installation Source”](#) (page 76).
- 2** Boot the target system using the first CD or DVD of the SUSE Linux Enterprise 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 4.4.3, “Using Custom Boot Options”](#) (page 89) 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 94).
- 5** Perform the installation as described in [Chapter 3, *Installation with YaST*](#) (page 33). Reconnect to the target system after it reboots for the final part of the installation.
- 6** Finish the installation.

4.1.6 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 4.2, “Setting Up the Server Holding the Installation Sources”](#) (page 68). Choose an NFS, HTTP, or FTP network server. For the configuration of an SMB installation source, refer to [Section 4.2.5, “Managing an SMB Installation Source”](#) (page 76).
- 2 Set up a TFTP server to hold a boot image that can be pulled by the target system. This is described in [Section 4.3.2, “Setting Up a TFTP Server”](#) (page 79).
- 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 4.3.1, “Setting Up a DHCP Server”](#) (page 77).
- 4 Prepare the target system for PXE boot. This is described in further detail in [Section 4.3.5, “Preparing the Target System for PXE Boot”](#) (page 86).
- 5 Initiate the boot process of the target system using Wake on LAN. This is described in [Section 4.3.7, “Wake on LAN”](#) (page 86).

- 6 On the controlling workstation, start an SSH client and connect to the target system as described in [Section 4.5.2, “SSH Installation”](#) (page 94).
- 7 Perform the installation as described in [Chapter 3, *Installation with YaST*](#) (page 33). Reconnect to the target system after it reboots for the final part of the installation.
- 8 Finish the installation.

4.2 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 Enterprise, 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 10 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 4.2.5, “Managing an SMB Installation Source”](#) (page 76) for details.

4.2.1 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 *Server Configuration*.

- 4 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.
- 5 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 export options. The NFS server will be accessible under `nfs://Server-IP/Name`. Details of NFS and exports can be found in [Chapter 36, *Sharing File Systems with NFS*](#) (page 681).

- 6 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 find the network installation source without any further configuration. For details on this option, refer to [Section 4.4, “Booting the Target System for Installation”](#) (page 87).

- 7 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.

4.2.2 Setting Up an NFS Installation Source Manually

IMPORTANT

We assume you are using any kind of SUSE Linux-based operating system on the machine that will serve as installation server. If this is not the case, turn to the others vendor's documentation on NFS instead of following these directions.

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 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.

On SUSE Linux and SUSE Linux Enterprise Server you can 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 Enterprise installation sources is automatically started and integrated into the boot process.

If you prefer manually exporting 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 with `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 with `rcslpd start`.

For more information about OpenSLP, refer to the package documentation located under `/usr/share/doc/packages/openslp/` or refer to [Chapter 30, SLP Services in the Network](#) (page 631).

4.2.3 Setting Up an FTP Installation Source Manually

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 4.2.2, “Setting Up an NFS Installation Source Manually”](#) (page 70).
- 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 4.2.2, “Setting Up an NFS Installation Source Manually”](#) (page 70), [Step 3](#) (page 71)).

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` with `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 with `rcslpd start`.

4.2.4 Setting Up an HTTP Installation Source Manually

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 4.2.2, “Setting Up an NFS Installation Source Manually”](#) (page 70).
- 2** Configure the HTTP server to distribute the contents of your installation directory:
 - a** Install the Web server Apache.

- 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** Reload the HTTP server configuration using `rcapache2 reload`.

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`.

4.2.5 Managing an SMB Installation Source

Using SMB, 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 Enterprise 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*. Replace *product* with the actual product name.
- 5 Copy the contents of all SUSE Linux Enterprise CDs or DVDs to the `INSTALL/product` directory.

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

- 1 Boot the installation target.
- 2 Select *Installation*.
- 3 Press `F3` and `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.

4.3 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.

4.3.1 Setting Up a DHCP Server

There are two ways to set up a DHCP server. For SUSE Linux Enterprise Server 9 and higher, YaST provides a graphical interface to the process. Users of any other SUSE Linux-based products and non-SUSE Linux users should manually edit the configuration files or use the front-end provided by their operating system vendors.

IMPORTANT

The following sections just cover the configuration changes needed to make your DHCP server ready for PXE boot. For more information about the configuration of DHCP, turn to the manuals of your operating system vendor.

Setting Up a DHCP Server with YaST

To announce the TFTP server's location to the network clients and specify the boot image file the installation target should use, add two declarations to your DHCP server configuration.

- 1 Log in as `root` to the machine hosting the DHCP server.
- 2 Start *YaST* → *Network Services* → *DHCP Server*.
- 3 Complete the setup wizard for basic DHCP server setup.
- 4 Select *Expert Settings* and select *Yes* when warned about leaving the start-up dialog.

- 5 In the *Configured Declarations* dialog, select the subnet in which the new system should be located and click *Edit*.
- 6 In the *Subnet Configuration* dialog select *Add* to add a new option to the subnet's configuration.
- 7 Select `filename` and enter `pxelinux.0` as the value.
- 8 Add another option (`next-server`) and set its value to the address of the TFTP server.
- 9 Select *OK* and *Finish* to complete the DHCP server configuration.

To configure DHCP to provide a static IP address to a specific host, enter the *Expert Settings* of the DHCP server configuration module ([Step 4](#) (page 77)) and add a new declaration of the host type. Add the options `hardware` and `fixed-address` to this host declaration and provide the appropriate values.

Setting Up a DHCP Server Manually

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" specifies 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" specifies 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 must 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.

4.3.2 Setting Up a TFTP Server

Set up a TFTP server with YaST on SUSE Linux Enterprise Server and SUSE Linux or set it up 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.

Setting Up a TFTP Server Manually

- 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 4.3.3, “Using PXE Boot”](#) (page 81).
- 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`.

4.3.3 Using 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=kernel module
```

By means of this entry, enter the network kernel module needed to support network installation on the PXE client. Replace *kernel module* with the appropriate module name for your network device.

```
netdevice=interface
```

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 4.4, “Booting the Target System for Installation”](#) (page 87).

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 harrdisk
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. Find more information about the options available in the documentation of the `syslinux` package located under `/usr/share/doc/packages/syslinux/`.

4.3.4 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
```

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 COM-BOOT 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	Perform a local boot with the Universal Network Driver Interface (UNDI) driver still resident in memory
5	Perform 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 canceled as soon as the user types anything on the keyboard, assuming the user will complete 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**.

4.3.5 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: BIOS Boot Order

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.

4.3.6 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.

4.3.7 Wake on LAN

Wake on LAN allows a machine to be turned on by a special network packet 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 turning on the wrong machine.

IMPORTANT: Wake on LAN across Different Network Segments

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.

Users of SUSE Linux Enterprise Server 9 and higher can use a YaST module called WOL to easily configure Wake on LAN. Users of other versions of SUSE Linux-based operating systems can use a command line tool.

4.3.8 Wake on LAN with YaST

- 1 Log in as `root`.
- 2 Start *YaST* → *Network Services* → *WOL*.
- 3 Click *Add* and enter the hostname and MAC address of the target system.
- 4 To turn on this machine, select the appropriate entry and click *Wake up*.

4.3.9 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-wake mac_of_target
```

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

4.4 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 4.3.7, “Wake on LAN”](#) (page 86) and [Section 4.3.3, “Using PXE Boot”](#) (page 81). You can either use the default boot options and function 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.

4.4.1 Using the Default Boot Options

The boot options are described in detail in [Chapter 3, *Installation with YaST*](#) (page 33). Generally, just selecting *Installation* starts the installation boot process.

If problems occur, use *Installation—ACPI Disabled* or *Installation—Safe Settings*. For more information about troubleshooting the installation process, refer to [Section 45.2, “Installation Problems”](#) (page 778).

4.4.2 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 (see [Section 4.4.3, “Using Custom Boot Options”](#) (page 89)).

See the table below for a complete set of the options available. To access the complete set of F keys available, first press [\[F3\]](#).

Table 4.1 *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	<ul style="list-style-type: none">• Text mode• VESA• resolution #1• resolution #2• ...	<ul style="list-style-type: none">• Default value depends on your graphics hardware
[F4]	Select the installation source	<ul style="list-style-type: none">• CD-ROM or DVD• SLP• FTP	CD-ROM or DVD

Key	Purpose	Available Options	Default Value
		<ul style="list-style-type: none"> • HTTP • NFS • SMB • Hard Disk 	
F5	Apply driver update disk	Driver	None

4.4.3 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 4.2 *Installation (Boot) Scenarios Used in This Chapter*

Installation Scenario	Parameters Needed for Booting	Boot Options
Chapter 3, <i>Installation with YaST</i> (page 33)	None: system boots automatically	None needed

Installation Scenario	Parameters Needed for Booting	Boot Options
Section 4.1.1, “Simple Remote Installation via VNC—Static Network Configuration” (page 60)	<ul style="list-style-type: none"> • Location of the installation server • Network device • IP address • Netmask • Gateway • VNC enablement • VNC password 	<ul style="list-style-type: none"> • <code>install=(nfs,http,ftp,smb):://path_to_instmedia</code> • <code>netdevice=some_netdevice</code> (only needed if several network devices are available) • <code>hostip=some_ip</code> • <code>netmask=some_netmask</code> • <code>gateway=ip_gateway</code> • <code>vnc=1</code> • <code>vncpassword=some_password</code>
Section 4.1.2, “Simple Remote Installation via VNC—Dynamic Network Configuration” (page 61)	<ul style="list-style-type: none"> • Location of the installation server • VNC enablement • VNC password 	<ul style="list-style-type: none"> • <code>install=(nfs,http,ftp,smb):://path_to_instmedia</code> • <code>vnc=1</code> • <code>vncpassword=some_password</code>
Section 4.1.3, “Remote Installation via VNC—PXE Boot and Wake on LAN” (page 63)	<ul style="list-style-type: none"> • Location of the installation server • Location of the TFTP server • VNC enablement • VNC password 	Not applicable; process managed through PXE and DHCP
Section 4.1.4, “Simple Remote Installation via SSH—Static Network Configuration” (page 64)	<ul style="list-style-type: none"> • Location of the installation server • Network device • IP address 	<ul style="list-style-type: none"> • <code>install=(nfs,http,ftp,smb):://path_to_instmedia</code>

Installation Scenario	Parameters Needed for Booting	Boot Options
Section 4.1.5, “Simple Remote Installation via SSH—Dynamic Network Configuration” (page 65)	<ul style="list-style-type: none"> • Netmask • Gateway • SSH enablement • SSH password 	<ul style="list-style-type: none"> • <code>netdevice=some_netdevice</code> (only needed if several network devices are available) • <code>hostip=some_ip</code> • <code>netmask=some_netmask</code> • <code>gateway=ip_gateway</code> • <code>usessh=1</code> • <code>sshpassword=some_password</code>
Section 4.1.6, “Remote Installation via SSH—PXE Boot and Wake on LAN” (page 67)	<ul style="list-style-type: none"> • Location of the installation server • SSH enablement • SSH password 	<ul style="list-style-type: none"> • <code>install=(nfs,http,ftp,smb):://path_to_instmedia</code> • <code>usessh=1</code> • <code>sshpassword=some_password</code> <p>Not applicable; process managed through PXE and DHCP</p>

TIP: More Information about linuxrc Boot Options

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

4.5 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.

4.5.1 VNC Installation

Using any VNC viewer software, you can remotely control the installation of SUSE Linux Enterprise 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 4.4.3, “Using Custom Boot Options”](#) (page 89)). 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.

4.5.2 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 4.4.3, “Using Custom Boot Options”](#) (page 89) 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 the password, enter the password that has been set with 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 3, *Installation with YaST*](#) (page 33).

Automated Installation

AutoYaST allows you to install SUSE® Linux Enterprise on a large number of machines in parallel. The AutoYaST technology offers great flexibility to adjust deployments to heterogeneous hardware. This chapter tells you how to prepare a simple automated installation and lay out an advanced scenario involving different hardware types and installation purposes.

5.1 Simple Mass Installation

IMPORTANT: Identical Hardware

This scenario assumes you are rolling out SUSE Linux Enterprise to a set of machines with exactly the same hardware configuration.

To prepare for an AutoYaST mass installation, proceed as follows:

- 1 Create an AutoYaST profile that contains the installation details needed for your deployment as described in [Section 5.1.1, “Creating an AutoYaST Profile”](#) (page 96).
- 2 Determine the source of the AutoYaST profile and the parameter to pass to the installation routines as described in [Section 5.1.2, “Distributing the Profile and Determining the `autoyast` Parameter”](#) (page 98).
- 3 Determine the source of the SUSE Linux Enterprise installation data as described in [Section 5.1.3, “Providing the Installation Data”](#) (page 100).

- 4 Determine and set up the boot scenario for autoinstallation as described in [Section 5.1.4, “Setting Up the Boot Scenario”](#) (page 101).
- 5 Pass the command line to the installation routines by adding the parameters manually or by creating an `info` file as described in [Section 5.1.5, “Creating the info File”](#) (page 103).
- 6 Start the autoinstallation process as described in [Section 5.1.6, “Initiating and Monitoring the Autoinstallation”](#) (page 106).

5.1.1 Creating an AutoYaST Profile

An AutoYaST profile tells AutoYaST what to install and how to configure the installed system to get a completely ready-to-use system in the end. It can be created in several different ways:

- Clone a fresh installation from a reference machine to a set of identical machines
- Use the AutoYaST GUI to create and modify a profile to meet your requirements
- Use an XML editor and create a profile from scratch

To clone a fresh reference installation, proceed as follows:

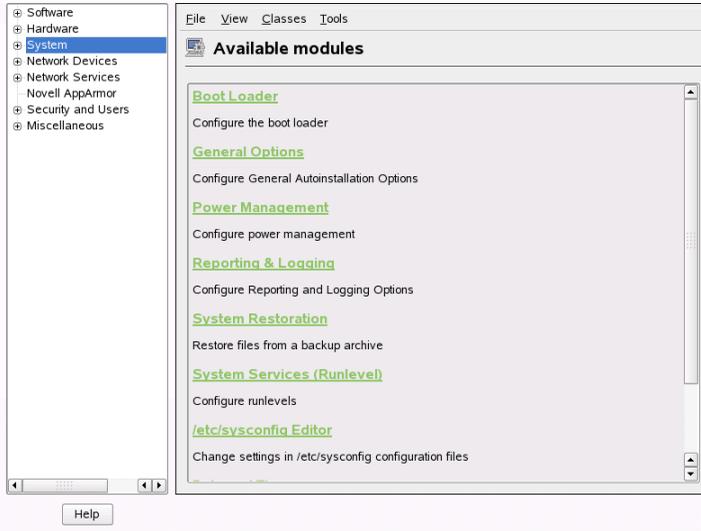
- 1 Perform a normal installation.
- 2 After you complete the hardware configuration and read the release notes, check *Create Profile For AutoYaST*, if it is not yet checked by default. This creates a ready-to-use profile as `/root/autoyast.xml` that can be used to create clones of this particular installation.

To use the AutoYaST GUI to create a profile from an existing system configuration and modify it to your needs, proceed as follows:

- 1 As `root`, start YaST.
- 2 Select *Miscellaneous* → *Autoinstallation* to start the graphical AutoYaST front-end.

- 3** Select *Tools* → *Create Reference Control File* to prepare AutoYaST to mirror the current system configuration into an AutoYaST profile.
- 4** As well as the default resources, like boot loader, partitioning, and software selection, you can add various other aspects of your system to the profile by checking the items in the list in *Create a Reference Control File*.
- 5** Click *Create* to have YaST gather all the system information and write it to a new profile.
- 6** To proceed, choose one of the following:
 - If the profile is complete and matches your requirements, select *File* → *Save as* and enter a filename for the profile, such as `autoyast.xml`.
 - Modify the reference profile by selecting the appropriate configuration aspects (such as “Hardware/Printer”) from the tree view to the left and clicking *Configure*. The respective YaST module starts but your settings are written to the AutoYaST profile instead of applied to your system. When done, select *File* → *Save as* and enter a suitable name for the profile.
- 7** Leave the AutoYaST module with *File* → *Exit*.

Figure 5.1 *Editing an AutoYaST Profile with the AutoYaST Front-End*



5.1.2 Distributing the Profile and Determining the autoyast Parameter

The AutoYaST profile can be distributed in several different ways. Depending on the protocol used to distribute the profile data, different AutoYaST parameters are used to make the profile location known to the installation routines on the client. The location of the profile is passed to the installation routines by means of the boot prompt or an `info` file that is loaded upon boot. The following options are available:

Profile Location	Parameter	Description
File	<code>autoyast=file:// path</code>	Makes the installation routines look for the control file in specified path (relative to source root directory— <code>file:///autoyast.xml</code> if in the top directory of a CD-ROM).

Profile Location	Parameter	Description
Device	<code>autoyast=device:// path</code>	Makes the installation routines look for the control file on a storage device. Only the device name is needed— <code>/dev/sda1</code> is wrong, use <code>sda1</code> instead.
Floppy	<code>autoyast=floppy:// path</code>	Makes the installation routines look for the control file on a floppy in the floppy drive. This option is especially useful, if you want to boot from CD-ROM.
NFS	<code>autoyast=nfs:// server/path</code>	Has the installation routines retrieve the control file from an NFS server.
HTTP	<code>autoyast=http:// server/path</code>	Has the installation routines retrieve the control file from an HTTP server.
HTTPS	<code>autoyast=https:// server/path</code>	Has the installation routines retrieve the control file from an HTTPS server.
TFTP	<code>autoyast=tftp:// server/path</code>	Has the installation routines retrieve the control file from a TFTP server.
FTP	<code>autoyast=ftp:// server/path</code>	Has the installation routines retrieve the control file from an FTP server.

Replace the *server* and *path* placeholders with values matching your actual setup.

AutoYaST includes a feature that allows binding certain profiles to the client's MAC address. Without having to alter the `autoyast=` parameter, you can have the same setup install several different instances using different profiles.

To use this, proceed as follows:

- 1 Create separate profiles with the MAC address of the client as the filename and put them on the HTTP server that holds your AutoYaST profiles.

- 2 Omit the exact path including the filename when creating the `autoyast=` parameter, for example:

```
autoyast=http://192.0.2.91/
```

- 3 Start the autoinstallation.

YaST tries to determine the location of the profile in the following way:

1. YaST searches for the profile using its own IP address in uppercase hexadecimal, for example, `192.0.2.91` is `C000025B`.
2. If this file is not found, YaST removes one hex digit and tries again. This action is repeated eight times until the file with the correct name is found.
3. If that still fails, it tries looking for a file with the MAC address of the clients as the filename. The MAC address of the example client is `0080C8F6484C`.
4. If the MAC address–named file cannot be found, YaST searches for a file named `default` (in lowercase). An example sequence of addresses where YaST searches for the AutoYaST profile looks as follows:

```
C000025B
C000025
C00002
C0000
C000
C000
C00
C00
C0
C
0080C8F6484C
default
```

5.1.3 Providing the Installation Data

The installation data can be provided by means of the product CDs or DVDs or using a network installation source. If the product CDs are used as the installation source, physical access to the client to install is needed, because the boot process needs to be initiated manually and the CDs need to be changed.

To provide the installation sources over the network, set up a network installation server (HTTP, NFS, FTP) as described in [Section 4.2.1, “Setting Up an Installation](#)

[Server Using YaST](#)” (page 68). Use an `info` file to pass the server's location to the installation routines.

5.1.4 Setting Up the Boot Scenario

The client can be booted in several different ways:

Network Boot

As for a normal remote installation, autoinstallation can be initiated with Wake on LAN and PXE, the boot image and control file can be pulled in via TFTP, and the installation sources from any network installation server.

Bootable CD-ROM

You can use the original SUSE Linux Enterprise media to boot the system for autoinstallation and pull in the control file from a network location or a floppy. Alternatively, create your own custom CD-ROM holding both the installation sources and the AutoYaST profile.

The following sections provide a basic outline of the procedures for network boot or boot from CD-ROM.

Preparing for Network Boot

Network booting with Wake on LAN, PXE, and TFTP is discussed in [Section 4.1.3, “Remote Installation via VNC—PXE Boot and Wake on LAN”](#) (page 63). To make the setup introduced there work for autoinstallation, modify the featured PXE Linux configuration file (`/srv/tftp/pxelinux.cfg/default`) to contain the `autoyast` parameter pointing to the location of the AutoYaST profile. An example entry for a standard installation looks like this:

```
default linux
# default label linux
kernel linux append initrd=initrd ramdisk_size=65536 insmod=e100 \
install=http://192.168.0.22/install/suse-enterprise/
```

The same example for autoinstallation looks like this:

```
default linux
# default label linux
kernel linux append initrd=initrd ramdisk_size=65536 insmod=e100 \
```

```
install=http://192.168.0.22/install/suse-enterprise/ \  
autoyast=nfs://192.168.0.23/profiles/autoyast.xml
```

Replace the example IP addresses and paths with the data used in your setup.

Preparing to Boot from CD-ROM

There are several ways in which booting from CD-ROM can come into play in AutoYaST installations. Choose from the following scenarios:

Boot from SUSE Linux Enterprise Media, Get the Profile over the Network

Use this approach if a totally network-based scenario is not possible (for example, if your hardware does not support PXE) and you have physical access to system to install during most of the process.

You need:

- The SUSE Linux Enterprise media
- A network server providing the profile data (see [Section 5.1.2, “Distributing the Profile and Determining the autoyast Parameter”](#) (page 98) for details)
- A floppy containing the `info` file that tells the installation routines where to find the profile

or

Access to the boot prompt of the system to install where you manually enter the `autoyast= parameter`

Boot and Install from SUSE Linux Enterprise Media, Get the Profile from a Floppy

Use this approach if an entirely network-based installation scenario would not work. It requires physical access to the system to install for turning on the target machine, or, in the second case, to enter the profile's location at the boot prompt. In both cases, you may also need to change media depending on the scope of installation.

You need:

- The SUSE Linux Enterprise media

- A floppy holding both the profile and the `info` file

or

Access to the boot prompt of the target to enter the `autoyast=` parameter

Boot and Install from Custom Media, Get the Profile from the Media

If you just need to install a limited number of software packages and the number of targets is relatively low, creating your own custom CD holding both the installation data and the profile itself might prove a good idea, especially if no network is available in your setup.

5.1.5 Creating the `info` File

The installation routines at the target need to be made aware of all the different components of the AutoYaST framework. This is done by creating a command line containing all the parameters needed to locate the AutoYaST components, installation sources, and the parameters needed to control the installation process.

Do this by manually passing these parameters at the boot prompt of the installation or by providing a file called `info` that is read by the installation routines (`linuxrc`). The former requires physical access to any client to install, which makes this approach unsuitable for large deployments. The latter enables you to provide the `info` file on some media that is prepared and inserted into the clients' drives prior to the autoinstallation. Alternatively, use PXE boot and include the `linuxrc` parameters in the `pxelinux.cfg/default` file as shown in [Section “Preparing for Network Boot”](#) (page 101).

The following parameters are commonly used for `linuxrc`. For more information, refer to the AutoYaST package documentation under `/usr/share/doc/packages/autoyast`.

IMPORTANT: Separating Parameters and Values

When passing parameters to `linuxrc` at the boot prompt, use `=` to separate parameter and value. When using an `info` file, separate parameter and value with `:`.

Keyword	Value
netdevice	The network device to use for network setup (for BOOTP/DHCP requests). Only needed if several network devices are available.
hostip	When empty, the client sends a BOOTP request. Otherwise the client is configured using the specified data.
netmask	Netmask.
gateway	Gateway.
nameserver	Name server.
autoyast	Location of the the control file to use for the automatic installation, such as autoyast=http://192.168.2.1/profiles/.
install	Location of the installation source, such as install=nfs://192.168.2.1/CDs/.
vnc	If set to 1, enables VNC remote controlled installation.
vncpassword	The password for VNC.
usessh	If set to 1, enables SSH remote controlled installation.

If your autoinstallation scenario involves client configuration via DHCP and a network installation source and you want to monitor the installation process using VNC, your `info` would look like this:

```
autoyast:profile_source install:install_source vnc:1 vncpassword:some_password
```

If you prefer a static network setup at installation time, your `info` file would look like the following:

```
autoyast:profile_source \  
install:install_source \  
hostip:some_ip \  

```

```
netmask:some_netmask \  
gateway:some_gateway
```

The \ indicate that the line breaks have only been added for the sake of readability. All options must be entered as one continuous string.

The `info` data can be made available to `linuxrc` in various different ways:

- As a file in the root directory of a floppy that is in the client's floppy drive at installation time.
- As a file in the root directory of the initial RAM disk used for booting the system provided either from custom installation media or via PXE boot.
- As part of the AutoYaST profile. In this case, the AutoYaST file needs to be called `info` to enable `linuxrc` to parse it. An example for this approach is given below.

`linuxrc` looks for a string (`start_linuxrc_conf`) in the profile that represents the beginning of the file. If it is found, it parses the content starting from that string and finishes when the string `end_linuxrc_conf` is found. The options are stored in the profile as follows:

```
....  
  <install>  
....  
  <init>  
    <info_file>  
<![CDATA[  
#  
# Don't remove the following line:  
# start_linuxrc_conf  
#  
install: nfs:server/path  
vnc: 1  
vncpassword: test  
autoyast: file:///info  
  
# end_linuxrc_conf  
# Do not remove the above comment  
#  
]]>  
  
    </info_file>  
  </init>  
.....  
  </install>  
....
```

linuxrc loads the profile containing the boot parameters instead of the traditional `info` file. The `install:` parameter points to the location of the installation sources. `vnc` and `vncpassword` indicate the use of VNC for installation monitoring. The `autoyast` parameter tells linuxrc to treat `info` as an AutoYaST profile.

5.1.6 Initiating and Monitoring the Autoinstallation

After you have provided all the infrastructure mentioned above (profile, installation source, and `info` file), you can go ahead and start the autoinstallation. Depending on the scenario chosen for booting and monitoring the process, physical interaction with the client may be needed:

- If the client system boots from any kind of physical media, either product media or custom CDs, you need to insert these into the client's drives.
- If the client is not switched on via Wake on LAN, you need to at least switch on the client machine.
- If you have not opted for remote controlled autoinstallation, the graphical feedback from AutoYaST is sent to the client's attached monitor or, if you use a headless client, to a serial console.

To enable remote controlled autoinstallation, use the VNC or SSH parameters described in [Section 5.1.5, “Creating the `info` File”](#) (page 103) and connect to the client from another machine as described in [Section 4.5, “Monitoring the Installation Process”](#) (page 92).

5.2 Rule-Based Autoinstallation

The following sections introduce the basic concept of rule-based installation using AutoYaST and provide an example scenario that enables you to create your own custom autoinstallation setup.

5.2.1 Understanding Rule-Based Autoinstallation

Rule-based AutoYaST installation allows you to cope with heterogeneous hardware environments:

- Does your site contain hardware of different vendors?
- Are the machines on your site of different hardware configuration (for example, using different devices or using different memory and disk sizes)?
- Do you intend to install across different domains and need to distinguish between them?

What rule-based autoinstallation does is, basically, generate a custom profile to match a heterogeneous scenario by merging several profiles into one. Each rule describes one particular distinctive feature of your setup (such as disk size) and tells AutoYaST which profile to use when the rule matches. Several rules describing different features of your setup are combined in an AutoYaST `rules.xml` file. The rule stack is then processed and AutoYaST generates the final profile by merging the different profiles matching the AutoYaST rules into one. To illustrate this procedure, refer to [Section 5.2.2, “Example Scenario for Rule-Based Autoinstallation”](#) (page 108).

Rule-based AutoYaST offers you great flexibility in planning and executing your SUSE Linux Enterprise deployment. You can:

- Create rules for matching any of the predefined system attributes in AutoYaST
- Combine multiple system attributes (such as disk size and kernel architecture) into one rule by using logical operators
- Create custom rules by running shell scripts and passing their output to the AutoYaST framework. The number of custom rules is limited to five.

NOTE

For more information about rule creation and usage with AutoYaST, refer to the package's documentation under `/usr/share/doc/packages/autoyast2/html/index.html`, *Chapter Rules and Classes*.

To prepare for a rule-based AutoYaST mass installation, proceed as follows:

- 1 Create several AutoYaST profiles that contain the installation details needed for your heterogeneous setup as described in [Section 5.1.1, “Creating an AutoYaST Profile”](#) (page 96).
- 2 Define rules to match the system attributes of your hardware setup as shown in [Section 5.2.2, “Example Scenario for Rule-Based Autoinstallation”](#) (page 108).
- 3 Determine the source of the AutoYaST profile and the parameter to pass to the installation routines as described in [Section 5.1.2, “Distributing the Profile and Determining the `autoyast` Parameter”](#) (page 98).
- 4 Determine the source of the SUSE Linux Enterprise installation data as described in [Section 5.1.3, “Providing the Installation Data”](#) (page 100)
- 5 Pass the command line to the installation routines by adding the parameters manually or by creating an `info` file as described in [Section 5.1.5, “Creating the `info` File”](#) (page 103).
- 6 Determine and set up the boot scenario for autoinstallation as described in [Section 5.1.4, “Setting Up the Boot Scenario”](#) (page 101).
- 7 Start the autoinstallation process as described in [Section 5.1.6, “Initiating and Monitoring the Autoinstallation”](#) (page 106).

5.2.2 Example Scenario for Rule-Based Autoinstallation

To get a basic understanding of how rules are created, think of the following example, depicted in [Figure 5.2, “AutoYaST Rules”](#) (page 110). One run of AutoYaST installs the following setup:

A Print Server

This machine just needs a minimal installation without a desktop environment and a limited set of software packages.

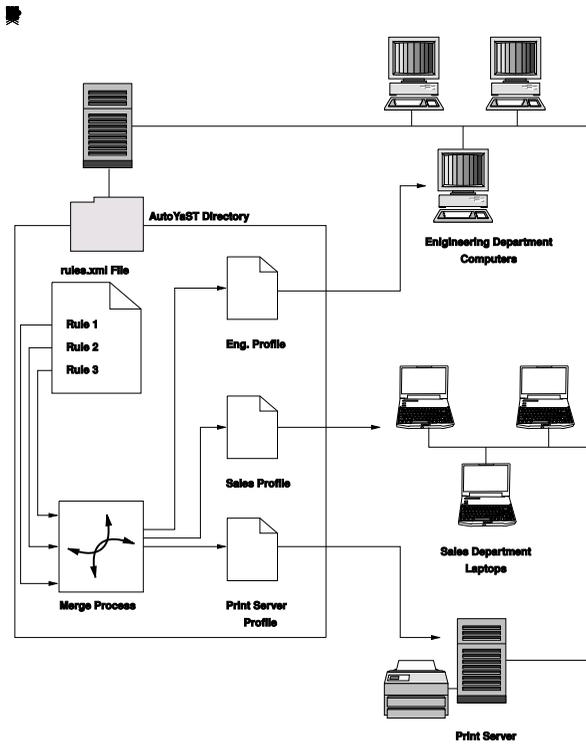
Workstations in the Engineering Department

These machines need a desktop environment and a broad set of development software.

Laptops in the Sales Department

These machines need a desktop environment and a limited set of specialized applications, such as office and calendaring software.

Figure 5.2 *AutoYaST Rules*



In a first step, use one of the methods outlined in [Section 5.1.1, “Creating an AutoYaST Profile”](#) (page 96) to create profiles for each use case. In this example, you would create `print.xml`, `engineering.xml`, and `sales.xml`.

In the second step, create rules to distinguish the three hardware types from one another and to tell AutoYaST which profile to use. Use an algorithm similar to the following to set up the rules:

1. Does the machine have an IP of *192.168.27.11*? Then make it the print server.
2. Does the machine have PCMCIA hardware and feature an Intel chipset? Then consider it an Intel laptop and install the sales department software selection.
3. If none of the above is true, consider the machine a developer workstation and install accordingly.

Roughly sketched, this translates into a `rules.xml` file with the following content:

```
<?xml version="1.0"?>
<!DOCTYPE autoinstall SYSTEM "/usr/share/autoinstall/dtd/rules.dtd">
<autoinstall xmlns="http://www.suse.com/1.0/yast2ns"
xmlns:config="http://www.suse.com/1.0/configns">
  <rules config:type="list">
    <rule>
      <hostaddress>
        <match>192.168.27.11</match>
        <match_type>exact</match_type>
      </hostaddress>
      <result>
        <profile>print.xml</profile>
        <continue config:type="boolean">false</continue>
      </result>
    </rule>
    <rule>
      <haspcmcia>
        <match>1</match>
        <match_type>exact</match_type>
      </haspcmcia>
      <custom1>
        <script>
if grep -i intel /proc/cpuinfo > /dev/null; then
echo -n "intel"
else
echo -n "non_intel"
fi;
        </script>
        <match>*</match>
        <match_type>exact</match_type>
      </custom1>
    </rule>
  </rules>
</autoinstall>
```

```

    <result>
      <profile>sales.xml</profile>
      <continue config:type="boolean">false</continue>
    </result>
    <operator>and</operator>
  </rule>
</rule>
<rule>
  <haspcmcia>
    <match>0</match>
    <match_type>exact</match_type>
  </haspcmcia>
</result>
  <profile>engineering.xml</profile>
  <continue config:type="boolean">false</continue>
</result>
</rule>
</rules>
</autoinstall>

```

When distributing the rules file, make sure that the `rules` directory resides under the `profiles` directory specified in the `autoyast=protocol:serverip/profiles/URL`. AutoYaST looks for a `rules` subdirectory containing a file named `rules.xml` first then loads and merges the profiles specified in the rules file.

The rest of the autoinstallation procedure is carried out as usual.

5.3 For More Information

For in-depth information about the AutoYaST technology, refer to the documentation installed along with the software. It is located under `/usr/share/doc/packages/autoyast2`. The most recent edition of this documentation can be found at http://www.suse.de/~ug/autoyast_doc/index.html.

Advanced Disk Setup

Sophisticated system configurations require particular disk setups. To get persistent device naming with SCSI devices, use a specific start-up script or udev. 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.

6.1 LVM Configuration

This section briefly describes the principles behind LVM and its basic features that make it useful under many circumstances. In [Section 6.1.2, “LVM Configuration with YaST”](#) (page 115), learn how to set up LVM with YaST.

WARNING

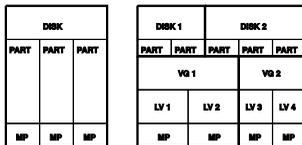
Using LVM might be associated with increased risk, such as data loss. Risks also include application crashes, power failures, and faulty commands. Save your data before implementing LVM or reconfiguring volumes. Never work without a backup.

6.1.1 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”](#) (page 40) and [Section 7.5.6, “Partitioner”](#) (page 150).

Figure 6.1 *Physical Partitioning versus LVM*



[Figure 6.1, “Physical Partitioning versus LVM”](#) (page 114) 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, it is possible to 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.

6.1.2 LVM Configuration with YaST

The YaST LVM configuration can be reached from the YaST Expert Partitioner (see [Section 7.5.6, “Partitioner”](#) (page 150)). This 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 6.2, “Creating a Volume Group”](#) (page 116)). 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 Enterprise 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 6.2 *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

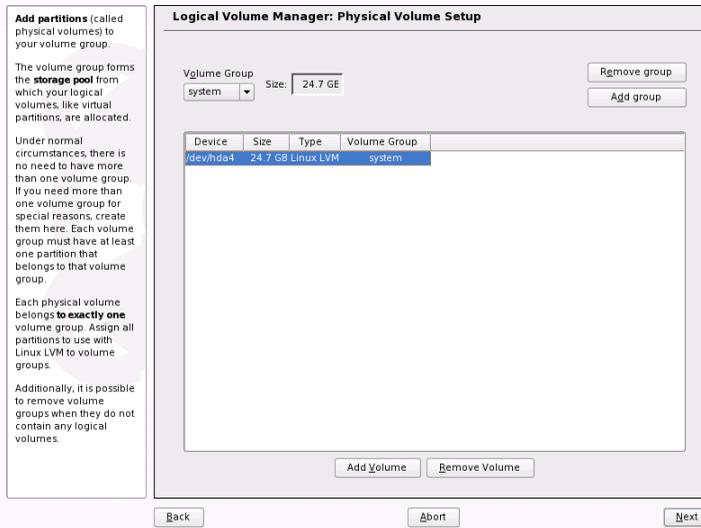
OK Cancel

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 6.3 *Physical Volume Setup*



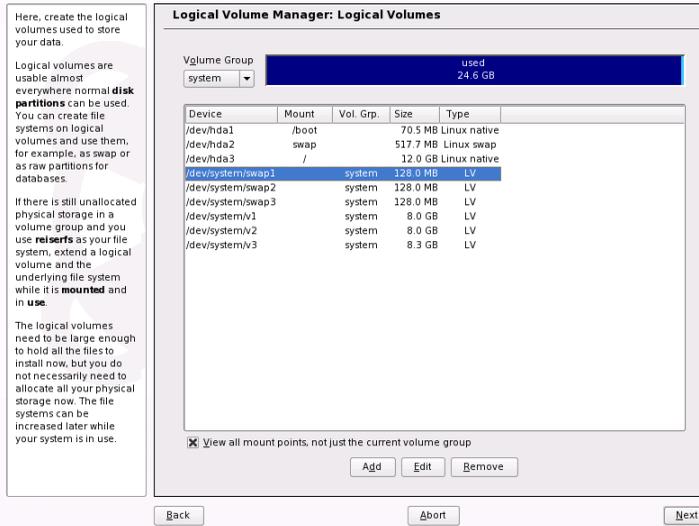
To add a previously unassigned partition to the selected volume group, first click the partition then *Add Volume*. At this point, the name of the volume group is entered next to the selected partition. Assign all partitions reserved for LVM to a volume group. Otherwise, the space on the partition remains 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 6.4 Logical Volume Management



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 6.5 *Creating Logical Volumes*

Create Logical Volume

Logical volume name

(e.g. var, opt)

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

12.0 MB

max = 52.0 MB max

Stripes

1

Stripe Size

64

Fstab Options

Mount Point

/home

OK Cancel

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.

6.2 Soft RAID Configuration

The purpose of RAID (redundant array of independent disks) is to combine several hard disk partitions into one large *virtual* hard disk to optimize performance, data secu-

rity, or both. 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. Soft RAID provides the advantages of RAID systems without the additional cost of hardware RAID controllers. However, this requires some CPU time and has memory requirements that make it unsuitable for real high performance computers.

6.2.1 RAID Levels

SUSE® Linux Enterprise 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 with 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. However, if damage is not detected, it also may happen that damaged data is mirrored to the correct disk and data corruption happens that way. The writing performance suffers a little in the copying process compared to when using single disk access (10 to 20 % 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 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.

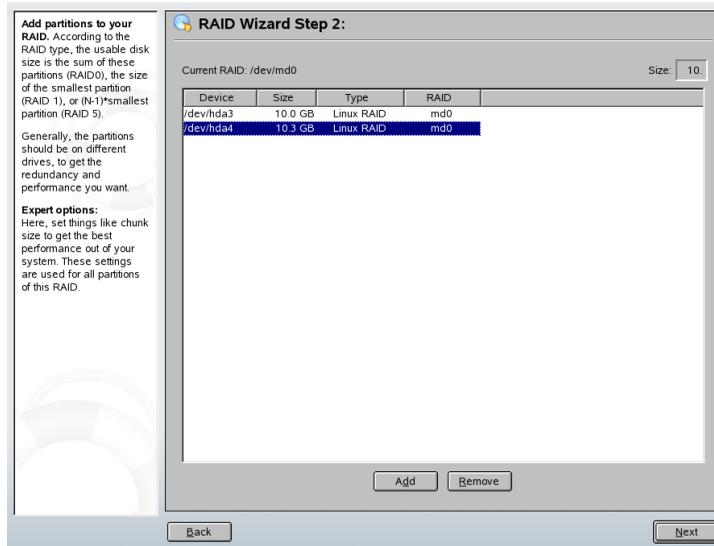
6.2.2 Soft RAID Configuration with YaST

The YaST soft RAID configuration can be reached from the YaST Expert Partitioner, described in [Section 7.5.6, “Partitioner”](#) (page 150). This 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.

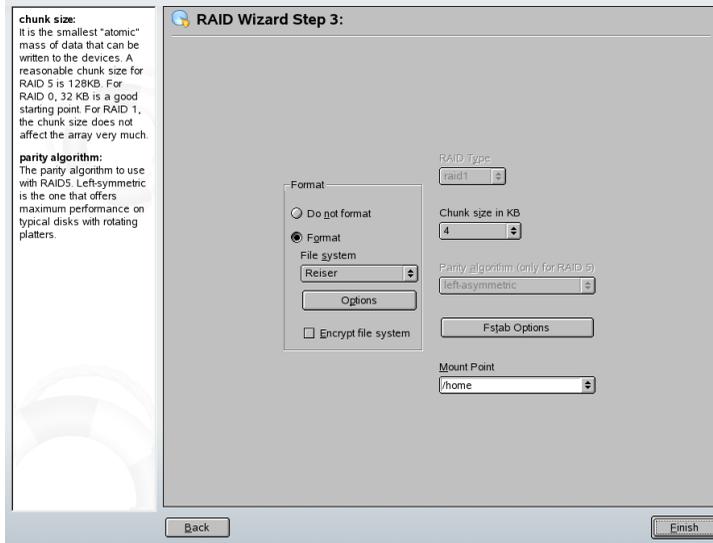
In the next dialog, choose between RAID levels 0, 1, and 5 (see [Section 6.2.1, “RAID Levels”](#) (page 120) for details). After *Next* is clicked, the following dialog lists all partitions with either the “Linux RAID” or “Linux native” type (see [Figure 6.6, “RAID Partitions”](#) (page 122)). No swap or DOS partitions are shown. If a partition is already assigned to a RAID volume, the name of the RAID device (for example, `/dev/md0`) is shown in the list. Unassigned partitions are indicated with “--”.

Figure 6.6 RAID Partitions



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 6.7, “File System Settings”](#) (page 123)).

Figure 6.7 File System Settings



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.

6.2.3 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.

6.2.4 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://marc.theaimsgroup.com/?l=linux-raid&r=1&w=2>.

System Configuration with YaST

7

In SUSE Linux Enterprise Server, YaST handles both the installation and configuration of your system. This chapter describes the configuration of system components (hardware), network access, and security settings, and administration of users. Find a short introduction to the text-based YaST interface in [Section 7.11, “YaST in Text Mode”](#) (page 170). For a description of manual system configuration, see [Section 16.3, “System Configuration via /etc/sysconfig”](#) (page 383).

Configure the system with YaST using various YaST modules. Depending on the hardware platform and the installed software, there are different ways to access YaST in the installed system.

In KDE or GNOME, start the YaST Control Center from the main menu. The individual YaST configuration modules are also integrated in the KDE Control Center. Before YaST starts, you are prompted to enter the `root` password, because YaST needs system administrator permissions to change the system files.

To start YaST from the command line, enter the commands `su` (for changing to the user `root`) and `yast2`. To start the text version, enter `yast` instead of `yast2`. Also use the command `yast` to start the program from one of the virtual consoles.

For hardware platforms that do not support a display device of their own and for remote administration on other hosts, run YaST remotely. First, open a console on the host on which to display YaST and enter the command

```
ssh -X root@<system-to-configure> to log in to the system to configure as root and redirect the X server output to your terminal. Following the successful SSH login, enter yast2 to start YaST in graphical mode.
```

To start YaST in text mode on another system, use `ssh root@<system-to-configure>` to open the connection. Then start YaST with `yast`.

To save time, the individual YaST modules can be started directly. To start a module, enter `yast2 module_name`. View a list of all module names available on your system with `yast2 -l` or `yast2 --list`. Start the network module, for example, with `yast2 lan`.

7.1 YaST Language

To change the language of YaST, select *System* → *Language Selection* in the YaST Control Center. Choose a language, exit the YaST Control Center, log out of the system, then log in again. The next time you start YaST, the new language setting is used. This also changes the language for the entire system.

If you need work in a different language but do not want to change the system language setting, you can temporarily change the `LANG` variable. To do so, export `LANG` with your preferred language. For example, for English, enter the command:

```
export LANG="en_US"; yast2
```

This command changes the `LANG` setting only in your current session. The language setting of other users and your other sessions, like terminal windows, remains unchanged.

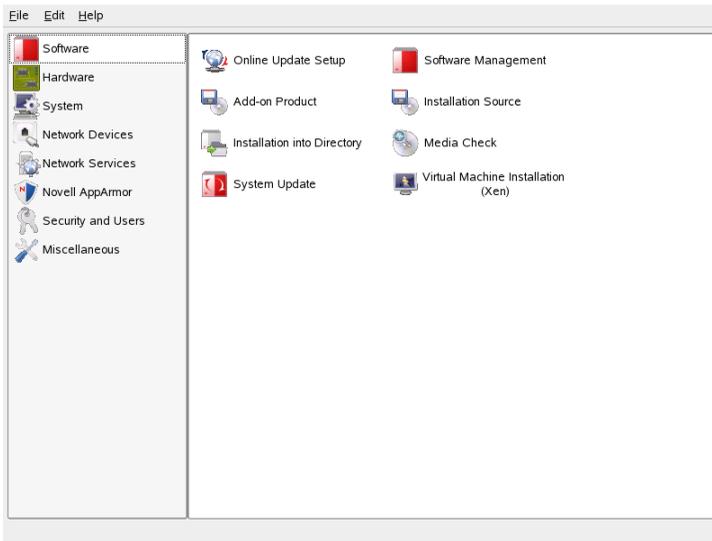
If you run YaST remotely over SSH, YaST uses the language settings of your local system.

7.2 The YaST Control Center

When you start YaST in the graphical mode, the YaST Control Center, as shown in [Figure 7.1, “The YaST Control Center”](#) (page 127), opens. The left frame contains the available categories. When you click a category, its contents are listed in the right frame. Then select the desired module. For example, if you select *Hardware* and click *Sound* in the right frame, a configuration dialog opens for the sound card. The configuration of the individual items usually consists of several steps. Press *Next* to proceed to the following step.

The left frame of most modules displays the help text, which offers suggestions for configuration and explains the required entries. To get help in modules without a help frame, press **F1** or choose *Help*. After selecting the desired settings, complete the procedure by pressing *Accept* on the last page of the configuration dialog. The configuration is then saved.

Figure 7.1 *The YaST Control Center*

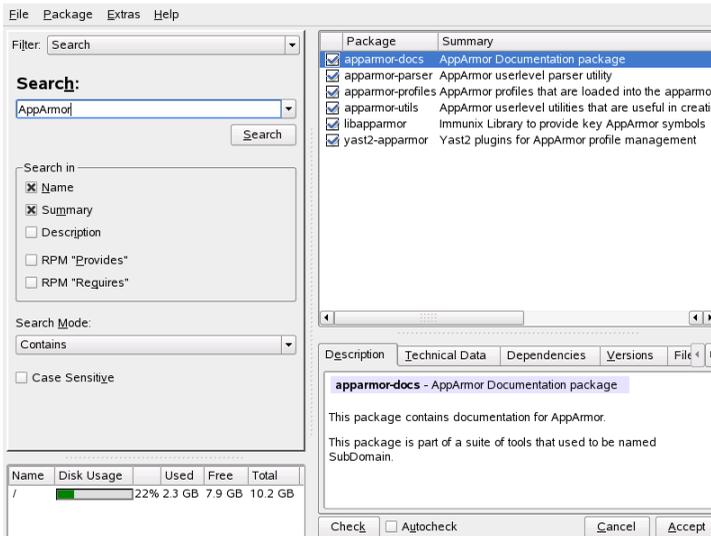


7.3 Software

7.3.1 Installing and Removing Software

To install, uninstall, and update software on your machine, use *Software* → *Software Management*. This opens a package manager dialog as shown in [Figure 7.2, “YaST Package Manager”](#) (page 128).

Figure 7.2 YaST Package Manager



In SUSE® Linux Enterprise, software is available in the form of RPM packages. Normally, a package contains everything needed for a program: the program itself, the configuration files, and all documentation. A list of individual packages is displayed to the right in the individual package window. The content of this list is determined by the currently selected filter. If, for example, the *Patterns* filter is selected, the individual package window displays all packages of the current selection.

In the package manager, each package has a status that determines what to do with the package, such as “Install” or “Delete.” This status is shown by a symbol in a status box at the beginning of the line. Change the status by clicking or selecting the desired status from the menu that opens when the item is right-clicked. Depending on the current situation, some of the possible status flags may not be available for selection. For example, a package that has not yet been installed cannot be set to “Delete.” View the available status flags with *Help* → *Symbols*.

The font color used for various packages in the individual package window provides additional information. Installed packages for which a newer version is available on the installation media are displayed in blue. Installed packages whose version numbers are higher than those on the installation media are displayed in red. However, because the version numbering of packages is not always linear, the information may not be

perfect, but should be sufficient to indicate problematic packages. If necessary, check the version numbers.

Installing Packages

To install packages, select packages for installation and click *Accept*. Selected packages should have the *Install* status icon. The package manager automatically checks the dependencies and selects any other required packages (resolution of dependencies). To view other packages required for installation before clicking *Accept*, choose *Extras* → *Show Automatic Package Changes* from the main menu. After installing packages, continue working with the package manager by clicking *Install More* or close it by clicking *Finish*.

The package manager provides preselected groups for installation. You can select an entire group instead of single packages. To view these groups, use *Filter* in the left frame.

TIP: List of All Available Packages

To display all packages on your installation media, use the filter *Package Groups* and select *zzz All* at the bottom of the tree. SUSE Linux Enterprise contains a number of packages and it might take some time to display this long list.

The Patterns filter groups the program packages according to their application purpose, such as multimedia or office applications. The various groups of the Patterns filter are listed with the installed packages preselected. Click the status box at the beginning of a line to install or uninstall this pattern. Select a status directly by right-clicking the pattern and using the context menu. From the individual package overview to the right, which displays the packages included in the current pattern, select and deselect individual packages.

To find language-specific packages, such as translated texts for the user interface of programs, documentation, and fonts, use the *Language* filter. This filter shows a list of all languages supported by SUSE Linux Enterprise. If you select one of these, the right frame shows all packages available for this language. Among these, all packages applying to your current software selection are automatically tagged for installation.

NOTE

Because language-specific packages may depend on other packages, the package manager may select additional packages for installation.

Packages and Installation Sources

If you want to find only packages from the specific source, use the *Installation Sources* filter. In the default configuration, this filter shows a list of all packages from the selected source. To restrict the list, use a secondary filter.

To view a list of the all installed packages from the selected installation source, select the filter *Installation Sources* then select *Installation Summary* from *Secondary Filters* and deactivate all check boxes except *Keep*.

The package status in the individual package window can be changed as usual. However, the changed package may no longer meet the search criteria. To remove such packages from the list, update the list with *Update List*.

Installing Source Packages

A package containing the source files for the program is usually available. The sources are not needed for running the program, but you may want to install the sources to compile a custom version of the program.

To install sources for selected program, mark the check box in the *Source* column. If you cannot see a check box, your installation sources do not contain the source of the package.

Removing Packages

To remove packages, assign the correct status to the packages to remove and click *Accept*. Selected packages should have the *Delete* status. If a package required by other installed packages is marked for deletion, the package manager issues an alert with detailed information and alternative solutions.

Reinstalling Packages

If you find damaged files that belong to package or you want to reinstall the original version of a package from your installation media, reinstall the package. To reinstall packages, select packages for reinstallation and click *Accept*. Selected packages should have the *Update* status. If any dependency issues arise with installed packages, the package manager issues an alert with detailed information and alternative solutions.

Searching for Packages, Applications, and Files

To find a specific package, use the *Search* filter. Enter a search string and click *Search*. By specifying various search criteria, you can restrict the search to display a few or even only one package. You can also define special search patterns using wild cards and regular expressions in *Search Mode*.

TIP: Quick Search

In addition to the *Search* filter, all lists of the package manager feature a quick search. Simply enter a letter to move the cursor to the first package in the list whose name begins with this letter. The cursor must be in the list (by clicking the list).

To find a package by name, select *Name*, enter the name of the package to find in the search field, and click *Search*. To find a package by text in the description, select *Summary* and *Descriptions*, enter a search string, and click *Search*.

To search for the package that contains a certain file, enter the name of the file, select *RPM "Provides"*, and click *Search*. To find all packages that depend on a particular package, select *RPM "Requires"*, enter the name of package, and click *Search*.

If you are familiar with the package structure of SUSE Linux Enterprise, you can use the *Package Groups* filter to find packages by subject. This filter sorts the program packages by subjects, such as applications, development, and hardware, in a tree structure to the left. The more you expand the branches, the more specific the selection is. This means fewer packages are displayed in the individual package window.

Installation Summary

After selecting the packages for installation, update, or deletion, view the installation summary with *Installation Summary*. It shows how packages will be affected when you click *Accept*. Use the check boxes to the left to filter the packages to view in the individual package window. For example, to check which packages are already installed, deactivate all check boxes except *Keep*.

The package status in the individual package window can be changed as usual. However, the respective package may no longer meet the search criteria. To remove such packages from the list, update the list with *Update List*.

Information about Packages

Get information about the selected package with the tabs in the bottom right frame. If another version of the package is available, you get information about both versions.

The *Description* tab with the description of the selected package is automatically active. To view information about package size, version, installation media, and other technical details, select *Technical Data*. Information about provided and required files is in *Dependencies*. To view available versions with their installation sources, click *Versions*.

Disk Usage

During the selection of the software, the resource window at the bottom left of the module displays the prospective disk usage of all mounted file systems. The colored bar graph grows with every selection. As long as it remains green, there is sufficient space. The bar color slowly changes to red as you approach the limit of disk space. If you select too many packages for installation, an alert is displayed.

Checking Dependencies

Some packages depend on other packages. This means that the software of the package only works properly if another package is also installed. There are some packages with identical or similar functionalities. If these packages use the same system resource, they should not be installed at the same time (package conflict).

When the package manager starts, it examines the system and displays installed packages. When you select to install and remove packages, the package manager automatically checks the dependencies and selects any other required packages (resolution of dependencies). If you select or deselect conflicting packages, the package manager indicates this and submits suggestions for solving the problem (resolution of conflicts).

Check Dependencies and *Autocheck* are located under the information window. If you click *Check Dependencies*, the package manager checks if the current package selection results in any unresolved package dependencies or conflicts. In the event of unresolved dependencies, the required additional packages are selected automatically. For package conflicts, the package manager opens a dialog that shows the conflict and offers various options for solving the problem.

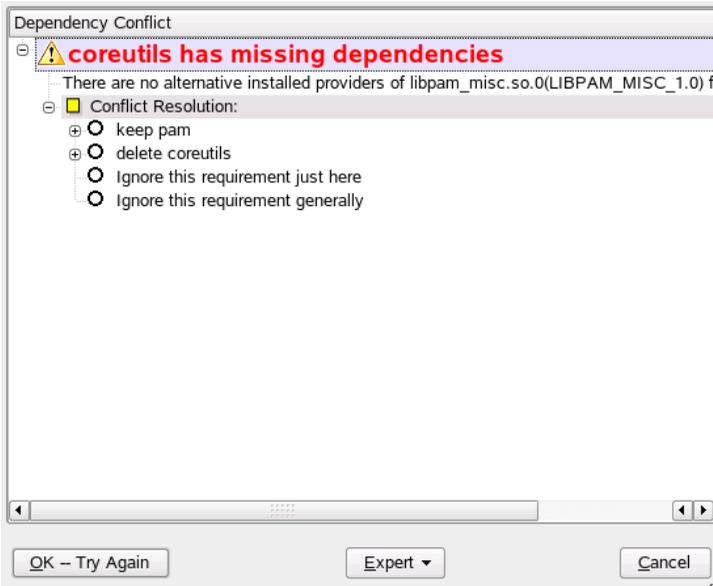
If you activate *Autocheck*, any change of a package status triggers an automatic check. This is a useful feature, because the consistency of the package selection is monitored permanently. However, this process consumes resources and can slow down the package manager. For this reason, the autocheck is not activated by default. In either case, a consistency check is performed when you confirm your selection with *Accept*.

For example, `sendmail` and `postfix` may not be installed concurrently. [Figure 7.3, “Conflict Management of the Package Manager”](#) (page 134) shows the conflict message prompting you to make a decision. `postfix` is already installed. Accordingly, you can refrain from installing `sendmail`, remove `postfix`, or take the risk and ignore the conflict.

WARNING: Handling Package Conflicts

Unless you are very experienced, follow the suggestions of YaST when handling package conflicts, because otherwise the stability and functionality of your system could be endangered by the existing conflict.

Figure 7.3 Conflict Management of the Package Manager



Installing -devel Packages

The package manager provides functions for quick and easy installation of devel and debug packages. To install all devel packages for your installed system, choose *Extras* → *Install All Matching — -devel Packages*. To install all debug packages for your installed system, choose *Extras* → *Install All Matching — -debuginfo Packages*.

7.3.2 Installing Add-On Products

Add-on products are extensions for your system. You can install a third party add-on product or a special extension of your SUSE Linux Enterprise, for example, the SDK add-on or a CD with binary drivers. To install a new add-on, use *Software* → *Add-On Product*. You can select various types of product media, like CD, FTP or local directory. You can work also directly with ISO files. To add an add-on as ISO file media, select *Local Directory* then choose *ISO Images*.

After successfully adding the add-on media, the package manager window appears. If the add-on provides a new pattern, see the new item in the *Patterns* filter. To view the

list of all packages from the selected installation source, select the filter *Installation Sources* and choose the installation source to view. To view packages from a selected add-on by package groups, select the secondary filter *Package Groups*.

Binary Drivers

Some hardware needs binary-only drivers for correct function. If you have such hardware, refer to the release notes for more information about availability of binary drivers for your system. To read the release notes, open YaST and select *Miscellaneous* → *Release Notes*.

SUSE Software Development Kit (SDK) 10

SUSE Software Development Kit 10 is an add-on for SUSE Linux Enterprise 10. It is a complete tool kit for application development. In fact, to provide a comprehensive build system, SUSE Software Development Kit 10 includes all the open source tools that were used to build the SUSE Linux Enterprise Server product. It provides you, as a developer, independent software vendor (ISV), or independent hardware vendor (IHV), with all the tools needed to port applications to all the platforms supported by the SUSE Linux Enterprise Desktop and SUSE Linux Enterprise Server.

SUSE Software Development Kit also contains integrated development environments (IDEs), debuggers, code editors, and other related tools. It supports most major programming languages (including C, C++, Java, and most scripting languages). For your convenience, SUSE Software Development Kit includes multiple Perl packages that are not included in the SUSE Linux Enterprise.

For detailed information, refer to <http://developer.novell.com/ndk/susesdk.htm>. Use the YaST add-on installer and package manager to install SUSE Software Development Kit 10.

7.3.3 Selecting the Installation Source

You can use multiple installation sources of several types. Select them and enable their use for installation or update using *Software* → *Installation Source*. When started, it displays a list of all previously registered sources. Following a normal installation from CD, only the installation CD is listed. Click *Add* to include additional sources in this list. Sources can be CDs, DVDs, or network sources, such as NFS and FTP servers.

Even directories on the local hard disk can be selected as the installation medium. See the detailed YaST help text for more details.

All registered sources have an activation status in the first column of the list. Enable or disable individual installation sources by clicking *Activate or Deactivate*. During the installation of software packages or updates, YaST selects a suitable entry from the list of activated installation sources. When you exit the module with *Close*, the current settings are saved and applied to the configuration modules *Software Management* and *System Update*.

7.3.4 Updating Software Online

Install important updates and improvements with YaST Online Update (YOU). The current patches for your SUSE product are available from the SUSE catalogs. To add or remove catalogs, use the *Software → Installation Source* module, described in [Section 7.3.3, “Selecting the Installation Source”](#) (page 135).

Find the list of available patches on the left. Patches are sorted by security importance:

Security

You must install these patches. Not installing the patches is a real security hazard.

Recommended

You should install these patches, because your computer could be compromised.

Optional

You can install these patches, but if you do not install them your computer remains secure.

To install a patch, select it in the list and click *Accept*. You can select multiple patches. To cancel your changes, click *Cancel*.

If you need special settings, for example, if your computer is behind a proxy server, use the command line tool rug. It is described in [Section 7.12, “Update from the Command Line”](#) (page 173).

7.3.5 Automatic Online Update

Software → *Online Update Setup* allows you to schedule automatic online updates. First enable automatic online update by activating *Enable Automatic Update* then set the time of the update. If you want to have full control over installed patches, you can schedule only the download of patches and install patches manually later. To download patches only, check *Only Download Patches*.

Some patches need some additional actions, for example, patches for the kernel require a reboot for activation. Information about the additional actions are provided with pre-installation information. To use automatic update only for normal patches without preinstallation information, check *Skip Patches with Preinstall Information*. Click *Finish* to exit the dialog.

7.3.6 Updating from a Patch CD

The *Patch CD Update* module from the *Software* section installs patches from CD, not from an FTP server. The advantage lies in a much faster update with CD. After the patch CD is inserted, all patches on the CD are displayed in the dialog. Select the desired packages for installation from the list of patches. The module issues an error message if no patch CD is present. Insert the patch CD then restart the module.

7.3.7 Updating the System

Update the version of SUSE Linux Enterprise installed on your system with *Software* → *System Update*. During operation, you can only update application software, not the base system. To update the base system, boot the computer from an installation medium, such as CD. When selecting the installation mode in YaST, select *Update*.

The procedure for updating the system is similar to a new installation. Initially, YaST examines the system, determines a suitable update strategy, and presents the results in a suggestion dialog. Click *Change* or the individual items to change any details.

Update Options

Set the update method for your system. Two options are available.

Update with Installation of New Software and Features Based on the Selection

To update the entire system to the latest versions of software, select one of the predefined selections. These selections ensure that packages that did not exist previously are also installed.

Only Update Installed Packages

This option merely updates packages that already exist on the system. No new features are installed.

Additionally, you can use *Delete Outdated Packages* to remove packages that do not exist in the new version. By default, this option is preselected to prevent outdated packages from unnecessarily occupying hard disk space.

Packages

Click *Packages* to start the package manager and select or deselect individual packages for update. Any package conflicts should be resolved with the consistency check. The use of the package manager is covered in detail in [Section 7.3.1, “Installing and Removing Software”](#) (page 127).

Backup

During the update, the configuration files of some packages may be replaced by those of the new version. Because you may have modified some of the files in your current system, the package manager normally makes backup copies of the replaced files. With this dialog, determine the scope of these backups.

IMPORTANT: Scope of the Backup

This backup does not include the software. It only contains configuration files.

Language

Primary and other languages currently installed on the system are listed here. Change them by clicking *Language* in the displayed configuration or with *Change → Language*. Optionally, adapt the keyboard layout and time zone to the region where the primary language is spoken. Find more about language selection in [Section 7.5.14, “Language Selection”](#) (page 157).

Important Information about Updates

The system update is a very complex procedure. For each program package, YaST must first check which version is installed on the computer then determine what needs to be done to replace the old version with the new version correctly. YaST also tries to adopt any personal settings of the installed packages.

In most cases, YaST replaces old versions with new ones without problems. A backup of the existing system should be performed prior to updating to ensure that existing configurations are not lost during the update. Conflicts can then be resolved manually after the update has finished.

7.3.8 Installing into a Directory

This YaST module allows you to install packages into a directory specified by you. Select where to place the root directory, how to name directories, and the type of system and software to install. After entering this module, YaST determines the system settings and lists the default directory, installation instructions, and software to install. Edit these settings by clicking *Change*. All changes must be confirmed by clicking *Accept*. After changes have been made, click *Next* until informed that the installation is complete. Click *Finish* to exit the dialog.

7.3.9 Installing into a Directory for Xen

Install packages into a directory for Xen with *Software* → *Installation into a Directory (Xen)*. Xen is a virtual machine monitor (VMM) for x86-compatible computers that enables you to run more than one virtual machine, each with its own OS, on a single physical system and with excellent performance. After this module starts, YaST determines the system settings and lists the default directory, installation instructions, and software to install. Edit the defaults by clicking *Change*. Confirm all changes by clicking *Accept*. After making all desired changes, click *Next* until you are informed that the installation is complete. Click *Finish* to exit the dialog.

7.3.10 Checking Media

If you encounter any problems using the SUSE Linux Enterprise installation media, you can check the CDs or DVDs with *Software* → *Media Check*. Media problems are more likely to occur with media you burn yourself. To check that a SUSE Linux Enterprise CD or DVD is error-free, insert the medium into the drive and run this module. Click *Start* for YaST to check the MD5 checksum of the medium. This may take several minutes. If any errors are detected, you should not use this medium for installation.

7.3.11 Registering SUSE Linux Enterprise

The registration and activation of your product is a precondition for technical support and product updates. If you skipped the registration during installation, you can register later with the help of the *Product Registration* module from *Software*. Before you start registration, prepare your contact e-mail and registration key.

In *Include for Convenience*, select whether to obtain some of the necessary information from your system. This simplifies the registration process. If you want to see what is required to register your system or what happens with your data, use *Details*.

To register, you can also use the command line tool `suse_register`. Refer to `suse_register --help` for directions.

To register your system without Internet access from a different computer, you must obtain a registration URL first. To obtain the URL, run `suse_register` without options on your new system. From the `suse_register` output, copy the URL and enter it in the URL field of a browser on a different computer.

TIP: Technical Support

Find more information about the technical support at <http://www.novell.com/support/products/linuxenterpriseserver/>.

7.4 Hardware

New hardware must first be installed or connected as directed by the vendor. Turn on external devices and start the appropriate YaST module. Most devices are automatically

detected by YaST and the technical data is displayed. If the automatic detection fails, YaST offers a list of devices (model, vendor, etc.) from which to select the suitable device. Consult the documentation enclosed with your hardware for more information.

IMPORTANT: Model Designations

If your model is not included in the device list, try a model with a similar designation. However, in some cases the model must match exactly, because similar designations do not always indicate compatibility.

7.4.1 Bluetooth

Configure Bluetooth devices with *Hardware* → *Bluetooth*. Click *Enable Bluetooth Services* to begin configuration. Bluetooth configuration is covered in detail in [Section “Configuring Bluetooth with YaST”](#) (page 561).

7.4.2 Infrared Device

Configure an infrared device with *Hardware* → *Infrared Device*. Click *Start IrDa* to begin configuration. You can configure *Port* and *Limit Baud Rate* here. Find information about infrared devices in [Section 28.3, “Infrared Data Transmission”](#) (page 570).

7.4.3 Graphics Card and Monitor

Configure graphics cards and monitors with *Hardware* → *Graphics Card and Monitor*. It uses the the SaX2 interface, described in [Section 7.13, “SaX2”](#) (page 176).

7.4.4 Printer

Configure a printer with *Hardware* → *Printer*. If a printer is properly connected to the system, it should be detected automatically. Find detailed instructions for configuring printers with YaST in [Section 19.4, “Configuring the Printer”](#) (page 424).

7.4.5 Hard Disk Controller

Normally, the hard disk controller of your system is configured during the installation. If you add controllers, integrate these into the system with *Hardware* → *Disk Controller*. You can also modify the existing configuration, but this is generally not necessary.

The dialog presents a list of detected hard disk controllers and enables assignment of the suitable kernel module with specific parameters. Use *Test Loading of Module* to check if the current settings work before they are saved permanently in the system.

WARNING: Configuration of the Hard Disk Controller

It is advised to test the settings before making them permanent in the system. Incorrect settings can prevent the system from booting.

7.4.6 Hardware Information

Display detected hardware and technical data using *Hardware* → *Hardware Information*. Click any node of the tree for more information about a device. This module is especially useful, for example, when submitting a support request for which you need information about your hardware.

Save the hardware information displayed to a file by clicking *Save to File*. Select the desired directory and filename then click *Save* to create the file.

7.4.7 IDE DMA Mode

Activate and deactivate the DMA mode for your IDE hard disks and your IDE CD and DVD drives in the installed system with *Hardware* → *IDE DMA Mode*. This module does not have any effect on SCSI devices. DMA modes can substantially increase the performance and data transfer speed in your system.

During installation, the current SUSE Linux Enterprise kernel automatically activates DMA for hard disks but not for CD drives, because default DMA activation for all drives often causes problems with CD drives. Use the DMA module to activate DMA for your drives. If the drive supports the DMA mode without any problems, the data transfer rate of your drive can be increased by activating DMA.

NOTE

DMA (direct memory access) means that your data can be transferred directly to the RAM, bypassing the processor control.

7.4.8 Joystick

Configure a joystick connected to the sound card with *Hardware* → *Joystick*. Select your joystick type in the list provided. If your joystick is not listed, select *Generic Analog Joystick*. After selecting your joystick, make sure that it is connected then click *Test* to test the functionality. Click *Continue* and YaST installs the required files. After the *Joystick Test* window appears, test the joystick by moving it in all directions and pressing all buttons. Each movement should be displayed in the window. If you are satisfied with the settings, click *OK* to return to the module and *Finish* to complete configuration.

If you have a USB device, this configuration is not necessary. Plug in the joystick and start using it.

7.4.9 Keyboard Layout

To configure the keyboard for the console, run YaST in text mode then use *Hardware* → *Keyboard Layout*. After clicking the module, the current layout is displayed. To choose another keyboard layout, select the desired layout from the list provided. Test the layout in *Test* by pressing keys on the keyboard.

Fine-tune the settings by clicking *Expert Settings*. You can adjust the key repeat rate and delay and configure the start-up state by choosing the desired settings in *Start-Up States*. For *Devices to Lock*, enter a space-separated list of devices to which to apply the `[Scroll Lock]`, `[Num Lock]`, and `[Caps Lock]` settings. Click *OK* to complete the fine-tuning. Finally, after all selections have been made, click *Accept* for your changes to take effect.

To set up the keyboard for the graphical environment, run the graphical YaST then select *Keyboard Layout*. Find information about the graphical configuration in [Section 7.13.3, “Keyboard Properties”](#) (page 181).

7.4.10 Mouse Model

When configuring the mouse for the graphical environment, click *Mouse Model* to access the SaX2 mouse configuration. Refer to [Section 7.13.2, “Mouse Properties”](#) (page 180) for details.

To configure your mouse for the text environment, use YaST in text mode. After entering text mode and selecting *Hardware* → *Mouse Model*, use the keyboard arrow keys to choose your mouse from the provided list. Then click *Accept* to save the settings and exit the module.

7.4.11 Scanner

Connect and turn on your scanner then select *Hardware* → *Scanner* to configure it. Most supported scanners are detected automatically. Select the scanner to configure and click *Edit*. If your scanner is not listed, click *Add* to open the manual configuration dialog. Select the appropriate vendor and model from the list and click *Next* to proceed with the installation. To modify a configured scanner, select it then click *Edit*.

After the scanner has been determined by either automatic detection or user selection, installation is carried out. Click *Finish* to complete the installation. If the installation is successful, a corresponding message appears. To test your scanner after installation, insert a document into your scanner and click *Other* → *Test*.

Scanner Not Detected

Only supported scanners can be detected automatically. Scanners connected to another network host cannot be detected. The manual configuration distinguishes three types of scanners: USB scanners, SCSI scanners, and network scanners.

USB Scanner

After the scanner is selected, YaST attempts to load the USB modules. If your scanner is very new, the modules may not be loaded automatically. In this case, continue automatically to a dialog in which to load the USB module manually. Refer to the YaST help text for more information.

SCSI Scanner

SCSI devices are normally detected. Specify the device, such as `/dev/sg0`. If problems arise, refer to the YaST help text. Remember always to shut down the system before connecting or disconnecting a SCSI scanner.

Network Scanner

Enter the IP address or the hostname. To configure a network scanner, refer to the database article *Scanning in Linux* (<http://en.opensuse.org/SDB:SDB>).

If your scanner is not detected, the device is probably not supported. However, sometimes even supported scanners are not detected. If this is the case, proceed with the manual scanner selection. If you can identify your scanner in the list of vendors and models, select it. If not, select *Cancel*. Information about scanners that work with Linux is provided at <http://cdb.suse.de/> and <http://www.sane-project.org/>.

WARNING: Assigning a Scanner Manually

Assign the scanner manually only if you are absolutely sure. An incorrect selection could damage your hardware.

Troubleshooting

Your scanner may not have been detected for one of the following reasons:

- The scanner is not supported. Check <http://cdb.suse.de/> for a list of Linux-compatible devices.
- The SCSI controller was not installed correctly.
- There were termination problems with your SCSI port.
- The SCSI cable is too long.
- The scanner has a SCSI light controller that is not supported by Linux.
- The scanner is defective.

WARNING

SCSI scanners should not be connected or disconnected while the system is running. Shut the system down first.

7.4.12 TV and Radio Cards

Configure TV and radio cards with *Hardware* → *TV Card*. If your card was automatically detected, it is displayed in the list. In this case, select the card and click *Edit*. If your card was not detected, click *Add*. If you have already configured TV or radio cards, select a card to modify then click *Edit*.

During the automatic hardware detection, YaST attempts to assign the correct tuner to your card. If you are not sure, simply keep the setting *Default (recognized)* and check whether it works. If you cannot set all channels, click *Select Tuner* and select the correct tuner type from the list.

If you are familiar with the technical details, you can use the expert dialog to make settings for a TV or radio card. Select a kernel module and its parameters in this dialog. Also check all parameters of your TV card driver. To do this, select the respective parameters and enter the new value in the parameter line. Confirm the new values with *Apply* or restore the default values with *Reset*.

Configure audio settings if your TV or radio card is connected to the installed sound card. Make the connection with a cable from output of the TV or radio card to the external audio input of the sound card. If you have not yet configured your sound card, select *Configure Sound Card* to configure it as described in [Section 7.4.13, “Sound”](#) (page 147).

If your TV or radio card has speaker jacks, you can also connect the speakers directly without using the sound card. There are also TV cards without any sound function, which do not require an audio configuration, such as those for CCD cameras.

When editing a configuration, you can also configure the TV stations by clicking *TV Channel*. Set the proper *TV Standard* and *Frequency Table* for your area and click *Scan the Channels*. A list of stations appears. After scanning has been completed, click *OK* to return to the configuration dialog.

7.4.13 Sound

Use *Hardware* → *Sound* to configure a sound card. Most sound cards are detected automatically and listed. Select the one to configure or modify then click *Edit*. Use *Delete* to remove a sound card. This deactivates existing entries of configured sound cards in `/etc/modprobe.d/sound`.

Click *Other* to open a dialog in which to customize the sound module options manually. With *Add*, configure additional sound cards. If YaST detects another sound card, select it then use *Edit*.

The volume and configuration of all sound cards installed are saved when you click *Finish*. The mixer settings are saved to the file `/etc/asound.conf` and the ALSA configuration data is appended at the end of the files `/etc/modprobe.d/sound` and `/etc/sysconfig/hardware`.

If YaST is unable to detect your sound card automatically, proceed as follows:

- 1 Click *Add* to open a dialog in which to select a sound card vendor and model. Refer to your sound card documentation for the information required. Find a reference list of sound cards supported by ALSA with their corresponding sound modules in `/usr/share/doc/packages/alsa/cards.txt` and at <http://www.alsa-project.org/~goemon/>. After making your selection, click *Next*.
- 2 In *Setup Dialog*, choose the configuration level in the first setup screen. With *Quick Automatic Setup*, you are not required to go through any of the further configuration steps and no sound test is performed. The sound card is configured automatically. With *Normal Setup*, you can adjust the output volume and play a test sound. *Advanced setup with possibility to change options* allows you to customize the sound card options manually.

In this dialog, there is also a shortcut to joystick configuration. Click it and select the joystick type in the following dialog. Click *Next* to continue.

- 3 In *Sound Card Volume*, test your sound configuration and make adjustments to the volume. You should start at about ten percent to avoid damage to your speakers or hearing. A test sound should be audible when you click *Test*. If you cannot hear anything, increase the volume. Press *Continue* to complete the sound configuration. The volume setting is then saved.

If you use a Creative Soundblaster Live or AWE sound card, copy SF2 sound fonts to your hard disk from the original Soundblaster driver CD-ROM with *Install Sound Fonts*. The sound fonts are saved in the directory `/usr/share/sfbank/creative/`.

For playback of MIDI files, check *Start Sequencer*. This way, the modules for sequencer support are loaded along with the sound modules.

7.5 System

This group of modules is designed to help you manage your system. All modules in this group are system-related and serve as valuable tools for ensuring that your system runs properly and your data is managed efficiently.

7.5.1 Backup

Create a backup of both your system and data using *System* → *System Backup*. However, the backup created by the module does not include the entire system. The system is backed up by saving important storage areas on your hard disk that may be crucial when trying to restore a system, such as the partition table or master boot record (MBR). It can also include the XML configuration acquired from the installation of the system, which is used for AutoYaST. Data is backed up by saving changed files of packages accessible on installation media, entire packages that are unaccessible (such as online updates), and files not belonging to packages, such as many of the configuration files in `/etc` or the directories under `/home`.

7.5.2 Restoration

With *System* → *System Restoration*, restore your system from a backup archive created with *System Backup*. First, specify where the archives are located (removable media, local hard disks, or network file systems). Click *Next* to view the description and contents of the individual archives and select what to restore from the archives.

You can also uninstall packages that were added since the last backup and reinstall packages that were deleted since the last backup. These two steps enable you to restore the exact system state at the time of the last backup.

WARNING: System Restoration

Because this module normally installs, replaces, or uninstalls many packages and files, use it only if you have experience with backups. Otherwise you may lose data.

7.5.3 Boot and Rescue Disks

Create boot and rescue disks with *System* → *Boot or Rescue Floppy*. These floppy disks are helpful if the boot configuration of your system is damaged. The rescue disk is especially necessary if the file system of the root partition is damaged.

The following options are available:

Standard Boot Floppy

Use this option to create the standard boot floppies with which to boot an installed system. Depending on the architecture, the actual number of boot disks may vary, but you should create all the boot disks presented in the dialog because all these disks are necessary for booting. They are also needed for starting the rescue system.

Rescue Floppy

This disk contains a special environment that allows you to perform maintenance tasks in your installed system, such as checking and repairing the file system and updating the boot loader. To start the rescue system, boot with the standard boot disks then select *Manual Installation* → *Start Installation or System* → *Rescue System*. Insert the rescue disk when prompted.

Custom Floppy

Use this to write any existing floppy disk image from the hard disk to a floppy disk.

Download Floppy Image

With this, enter a URL and authentication data to download a floppy disk image from the Internet.

To create one of these floppy disks, select the corresponding option and click *Next*. Insert a floppy disk when prompted. Click *Next* again to create the floppy disk.

7.5.4 Boot Loader Configuration

To configure booting for systems installed on your computer, use the *System → Boot Loader* module. A detailed description of how to configure the boot loader with YaST is available in [Section 17.3, “Configuring the Boot Loader with YaST”](#) (page 397).

7.5.5 LVM

The logical volume manager (LVM) is a tool for custom partitioning of hard disks with logical drives. Find information about LVM in [Section 6.1, “LVM Configuration”](#) (page 113).

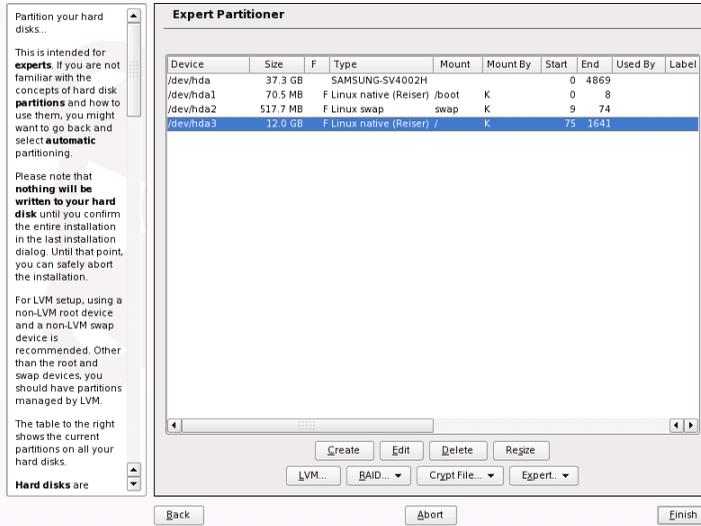
7.5.6 Partitioner

With the expert dialog, shown in [Figure 7.4, “The YaST Partitioner”](#) (page 151), manually modify the partitioning of one or several hard disks. Partitions can be added, deleted, resized, and edited. Also access the soft RAID and LVM configuration from this YaST module.

WARNING

Although it is possible to modify the partitions in the installed system, this should be handled only by experts. Otherwise the risk of making a mistake that causes data loss is very high. If you repartition a hard disk in use, reboot the system right afterwards. It is safer to use the rescue system than repartition the system while running.

Figure 7.4 *The YaST Partitioner*



All existing or suggested partitions on all connected hard disks are displayed in the list of the YaST *Expert Partitioner* dialog. Entire hard disks are listed as devices without numbers, such as `/dev/hda` or `/dev/sda`. Partitions are listed as parts of these devices, such as `/dev/hda1` or `/dev/sda1`. The size, type, file system, and mount point of the hard disks and their partitions are also displayed. The mount point describes where the partition appears in the Linux file system tree.

If you run the expert dialog during installation, any free hard disk space is also listed and automatically selected. To provide more disk space to , free the needed space starting from the bottom toward the top of the list (starting from the last partition of a hard disk toward the first). For example, if you have three partitions, you cannot use the second exclusively for and retain the third and first for other operating systems.

Creating a Partition

Select *Create*. If several hard disks are connected, a selection dialog appears in which to select a hard disk for the new partition. Then specify the partition type (primary or extended). Create up to four primary partitions or up to three primary partitions and one extended partition. Within the extended partition, create several logical partitions (see [Section “Partition Types”](#) (page 40)).

Select the file system to use and a mount point, if necessary. YaST suggests a mount point for each partition created. Details of the parameters are provided in the next section. Select *OK* to apply your changes. The new partition is then listed in the partition table. If you click *Next*, the current values are adopted. During installation you are then returned to the suggestion screen.

Partitioning Parameters

When you create a new partition or modify an existing partition, set various parameters. For new partitions, suitable parameters are set by YaST and usually do not require any modification. To make manual settings, proceed as follows:

1. Select the partition.
2. Click *Edit* to edit the partition and set the parameters:

File System ID

Even if you do not want to format the partition at this stage, assign it a file system ID to ensure that the partition is registered correctly. Possible values include *Linux*, *Linux swap*, *Linux LVM*, and *Linux RAID*. For LVM and RAID details, refer to [Section 6.1, “LVM Configuration”](#) (page 113) and [Section 6.2, “Soft RAID Configuration”](#) (page 119).

File System

To format the partition immediately within the scope of the installation, specify one of the following file systems for the partition: *Swap*, *Ext2*, *Ext3*, *ReiserFS*, or *JFS*. Refer to [Chapter 21, *File Systems in Linux*](#) (page 453) for details on the various file systems.

File System Options

Set various parameters for the selected file system here.

Encrypt File System

If you activate the encryption, all data is written to the hard disk in encrypted form. This increases the security of sensitive data, but slightly reduces the system speed, because the encryption takes some time. More information about the encryption of file systems is provided in [Chapter 41, *Encrypting Partitions and Files*](#) (page 733).

Fstab Options

Here, specify various parameters for the administration file of the file systems (`/etc/fstab`). For example, change the file system identification from the device name, which is default, to a volume label. In the volume label, you can use all characters except `/` and space.

Mount Point

Specify the directory at which the partition should be mounted in the file system tree. Select from various YaST proposals or enter any other name.

3. Select *Next* to activate the partition.

If you partition manually, create a swap partition of at least 256 MB. The swap partition is used to free the main memory of data that is not used at the present moment. This keeps the main memory free for the most frequently-used data.

Expert Options

Expert opens a menu containing the following commands:

Reread Partition Table

Rereads the partitioning from disk. For example, you need this after manual partitioning in the text console.

Delete Partition Table and Disk Label

This completely overwrites the old partition table. For example, this can be helpful if you have problems with unconventional disk labels. Using this method, all data on the hard disk is lost.

More Partitioning Tips

If the partitioning is performed by YaST and other partitions are detected in the system, these partitions are also entered in the file `/etc/fstab` to enable easy access to this data. This file contains all partitions in the system with their properties, such as the file system, mount point, and user permissions.

Example 7.1 */etc/fstab: Partition Data*

```
/dev/sda1 /data1 auto noauto,user 0 0
/dev/sda5 /data2 auto noauto,user 0 0
/dev/sda6 /data3 auto noauto,user 0 0
```

The partitions, regardless of whether they are Linux or FAT partitions, are specified with the options `noauto` and `user`. This allows any user to mount or unmount these partitions as needed. For security reasons, YaST does not automatically enter the `exec` option here, which is needed for executing programs from the location. However, to run programs from there, you can enter this option manually. This measure is necessary if you encounter system messages such as `bad interpreter` or `Permission denied`.

Partitioning and LVM

From the expert partitioner, access the LVM configuration with *LVM* (see [Section 6.1, “LVM Configuration”](#) (page 113)). However, if a working LVM configuration already exists on your system, it is automatically activated as soon as you enter the LVM configuration for the first time in a session. In this case, any disks containing a partition belonging to an activated volume group cannot be repartitioned because the Linux kernel cannot reread the modified partition table of a hard disk when any partition on this disk is in use. However, if you already have a functioning LVM configuration on your system, physical repartitioning should not be necessary. Instead, change the configuration of the logical volumes.

At the beginning of the physical volumes (PVs), information about the volume is written to the partition. To reuse such a partition for other non-LVM purposes, it is advisable to delete the beginning of this volume. For example, in the VG `system` and PV `/dev/sda2`, do this with the command `dd if=/dev/zero of=/dev/sda2 bs=512 count=1`.

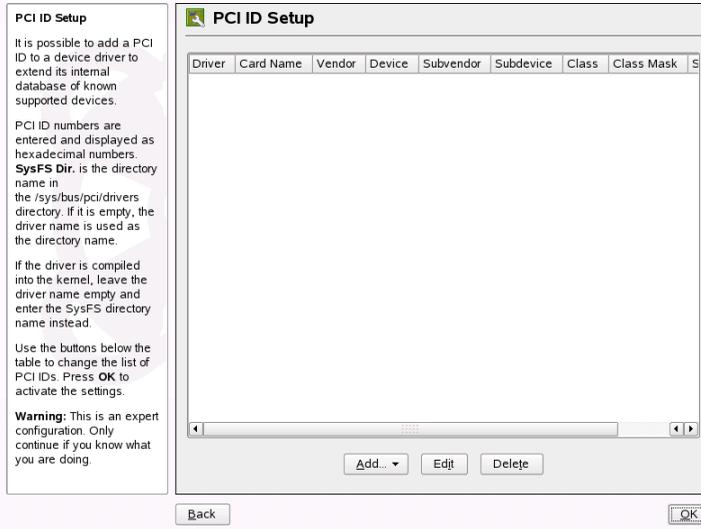
WARNING: File System for Booting

The file system used for booting (the root file system or `/boot`) must not be stored on an LVM logical volume. Instead, store it on a normal physical partition.

7.5.7 PCI Device Drivers

Each kernel driver contains a list of device IDs of all devices it supports. If a new device is not in any driver's database, the device is treated as unsupported, even if it can be used with an existing driver. With this YaST module from *System* section, you can add PCI IDs. Only advanced users should attempt to use this YaST module.

Figure 7.5 Adding a PCI ID



To add an ID, click *Add* and select how to assign it: by selecting a PCI device from a list or by manually entering PCI values. In the first option, select the PCI device from the provided list then enter the driver or directory name. If the directory is left empty, the driver name is used as the directory name. When assigning PCI ID values manually, enter the appropriate data to set up a PCI ID. Click *OK* to save your changes.

To edit a PCI ID, select the device driver from the list and click *Edit*. Edit the information and click *OK* to save your changes. To delete an ID, select the driver and click *Delete*. The ID immediately disappears from the list. When finished, click *OK*.

7.5.8 Power Management

The *System* → *Power Management* module helps you work with saving energy technologies. It is especially important on laptops to extend their operational time. Find detailed information about using this module in [Section 27.6, “The YaST Power Management Module”](#) (page 543).

7.5.9 Powertweak Configuration

Powertweak is a SUSE Linux utility for tweaking your system to peak performance by tuning some kernel and hardware configurations. It should be used only by advanced users. After starting it with *System* → *Powertweak*, it detects your system settings and lists them in tree form in the left frame of the module. You can also use *Search* to find a configuration variable. Select the option to tweak to display it on the screen along with its directory and settings. To save the settings, click *Finish* then confirm it by clicking *OK*.

7.5.10 Profile Manager

Create, manage, and switch among system configurations with *System* → *Profile Management*, the YaST system configuration profile management (SCPM) module. This is especially useful for mobile computers that are used in different locations (in different networks) and by different users. Nevertheless, this feature is useful even for stationary machines, because it enables the use of various hardware components or test configurations. For more information about SCPM basics and handling, refer to [Chapter 26, System Configuration Profile Management](#) (page 509).

7.5.11 System Services (Runlevel)

Configure runlevels and the services that start in them with *System* → *System Services (Runlevel)*. For more information about the runlevels in SUSE Linux Enterprise and a description of the YaST runlevel editor, refer to [Section 16.2.3, “Configuring System Services \(Runlevel\) with YaST”](#) (page 382).

7.5.12 /etc/sysconfig Editor

The directory `/etc/sysconfig` contains the files with the most important settings for SUSE Linux Enterprise. Use *System* → */etc/sysconfig Editor* to modify the values and save them to the individual configuration files. Generally, manual editing is not necessary, because the files are automatically adapted when a package is installed or a service is configured. More information about `/etc/sysconfig` and the YaST `sysconfig` editor is available in [Section 16.3.1, “Changing the System Configuration Using the YaST `sysconfig` Editor”](#) (page 384).

7.5.13 Time and Date Configuration

The time zone is initially set during installation, but you can change it with *System* → *Date and Time*. Also use this to change the current system date and time.

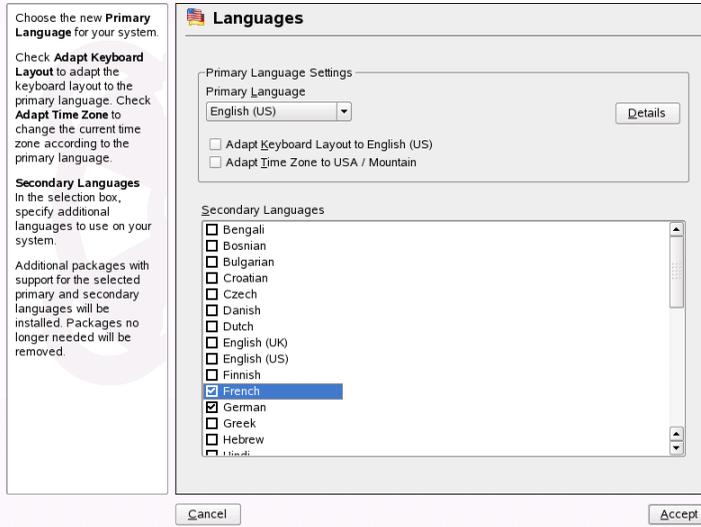
To change the time zone, select the region in the left column and the location or time zone in the right column. With *Hardware Clock Set To*, set whether the system clock should use *Local Time* or *UTC* (Coordinated Universal Time). *UTC* is often used in Linux systems. Machines with additional operating systems, such as Microsoft Windows, mostly use local time.

Set the current system time and date with *Change*. In the dialog that opens, modify the time and date by entering new values or adjusting them with the arrow buttons. Press *Apply* to save the changes.

7.5.14 Language Selection

The primary and secondary languages for your system are set during installation. However, they can be changed at any time using *System* → *Language*. The primary language set in YaST applies to the entire system, including YaST and the desktop environment. This is the language you expect to use most of the time. Secondary languages are languages that are sometimes needed by users for a variety of purposes, such as desktop language or word processing.

Figure 7.6 *Setting the Language*



Select the main language to use for your system in *Primary Language*. To adjust the keyboard or time zone to this setting, enable *Adapt Keyboard Layout* or *Adapt Time Zone*.

Set how locale variables are set for the `root` user with *Details*. Also use *Details* to set the primary language to a dialect not available in the main list. These settings are written into the file `/etc/sysconfig/language`.

7.6 Network Devices

All network devices connected to the system must be initialized before they can be used by a service. The detection and configuration of these devices is done in the module group *Network Devices*.

7.6.1 DSL, ISDN, Modem, or Network Card

To configure a DSL, ISDN, or network interface or a modem, select the appropriate module from the *Network Devices* section. For a device that is detected automatically,

select it from the list then click *Edit*. If your device has not been detected, click *Add* and select it manually. To edit an existing device, select it then click *Edit*. For more detailed information, see [Section 29.4, “Configuring a Network Connection with YaST”](#) (page 594). For wireless network interfaces, see [Chapter 28, *Wireless Communication*](#) (page 549).

TIP: CDMA and GPRS Modems

You can configure supported CDMA and GPRS modems as regular modems in the YaST modem module.

7.6.2 Fax

Configure a fax system with *Network Devices* → *Fax*. Set up the fax system for one or more users, but each user must have a unique fax number. When adding or editing users, configure the username, fax numbers, outgoing MSN, station ID, headline, and desired action.

7.6.3 Phone Answering Machine

Configure your SUSE Linux Enterprise system to function as a telephone answering machine with *Network Devices* → *Phone Answering Machine*. Configure it for one or more users, but each user must have a unique telephone number. When adding or editing users, configure the username, telephone numbers, delay, duration, and desired action. Assign a PIN (personal identification number) to provide the user with remote access to the machine.

7.7 Network Services

This group contains tools to configure all kinds of services in the network. These include name resolution, user authentication, and file services.

7.7.1 Mail Transfer Agent

You can configure your mail settings in *Network Services* → *Mail Transfer Agent* if you send your e-mail with sendmail, postfix, or the SMTP server of your provider. You can fetch mail via the fetchmail program, for which you can also enter the details of the POP3 or IMAP server of your provider. Alternatively, use a mail program of your choice, such as KMail or Evolution, to set your access data. In this case, you do not need this module.

To configure your mail with YaST, specify the type of your connection to the Internet in the first dialog. Choose one of the following options:

Permanent

Select this option if you have a dedicated line to the Internet. Your machine is online permanently, so no dial-up is required. If your system is part of a local network with a central e-mail server, select this option to ensure permanent access to your e-mail messages.

Dial-Up

This item is relevant for users who have a computer at home, are not located in a network, and occasionally connect to the Internet.

No Connection

If you do not have access to the Internet and are not located in a network, you cannot send or receive e-mail.

Activate virus scanning for your incoming and outgoing e-mail with AMaViS by selecting that option. The package is installed automatically as soon as you activate the mail filtering feature. In the following dialogs, specify the outgoing mail server (usually the SMTP server of your provider) and the parameters for incoming mail. Set the diverse POP or IMAP servers for mail reception by various users. Using this dialog, you can also assign aliases, use masquerading, or set up virtual domains. Click *Finish* to exit the mail configuration.

7.7.2 Other Available Services

Many other network modules are available in YaST *Network Services*.

DNS and Hostname

Use this module to configure the hostname and DNS if these settings were not already made while configuring the network devices. Also use it to change the hostname and domain name. If the provider has been configured correctly for DSL, modem, or ISDN access, the list of name servers contains the entries that were extracted automatically from the provider data. If you are located in a local network, you might receive your hostname via DHCP, in which case you should not modify the name.

Hostnames

When booting and in small networks, you can use *Hostnames* for hostname resolution instead of DNS. The entries in this module reflect the data of the file `/etc/hosts`. For more information, read [Section “ /etc/hosts ”](#) (page 618).

Kerberos Client

If you have a Kerberos server in your network for network authentication, use *Kerberos Client*.

LDAP Client

If using LDAP for user authentication in the network, configure the client in *LDAP Client*. Information about LDAP and a detailed description of the client configuration with YaST are available in [Section 34.3, “Configuring an LDAP Client with YaST”](#) (page 658).

NFS Client

With NFS client, mount directories provided by NFS server in your own file trees. Use *NFS Client* to configure your system to access an NFS server in the network. A description of the YaST module and background information about NFS are provided in [Chapter 36, *Sharing File Systems with NFS*](#) (page 681).

NIS Client

If you run NIS server to administer user data on a central place and distribute it to the clients, configure the client here. Detailed information about NIS client and configuration with YaST is available in [Section 32.1, “Configuring NIS Clients”](#) (page 641).

NTP Client

NTP (network time protocol) is a protocol for synchronizing hardware clocks over a network. Information about NTP and instructions for configuring it with YaST are available in [Chapter 31, *Time Synchronization with NTP*](#) (page 635).

Network Services (xinetd)

Configure the network services (such as *finger*, *talk*, and *ftp*) to start when SUSE Linux Enterprise boots using *Network Services*. These services enable external hosts to connect to your computer. Various parameters can be configured for every service. By default, the master service that manages the individual services (*inetd* or *xinetd*) is not started.

When this module starts, choose whether to start *inetd* or *xinetd*. The selected daemon can be started with a standard selection of services. Alternatively, compose your own selection of services with *Add*, *Delete*, and *Edit*.

WARNING: Configuring Network Services (xinetd)

The composition and adjustment of network services on a system is a complex procedure that requires a comprehensive understanding of the concept of Linux services. The default settings are usually sufficient.

Proxy

Configure Internet proxy client settings in *Proxy*. Click *Enable Proxy* then enter the desired proxy settings. You can test these settings by clicking *Test Proxy Settings*. A small window informs you whether your proxy settings work correctly. After your settings have been entered and tested, save them by clicking *Accept*.

Remote Administration

To administer your machine remotely from another machine, use *Remote Administration*. To maintain your system remotely, use a VNC client, such as *krdc*, or a Java-enabled browser. Although remote administration using VNC is simple and fast, it is less secure than using SSH, so you should always keep this in mind when using a VNC server. Find detailed information about installing with a VNC client in [Section 4.1.1, “Simple Remote Installation via VNC—Static Network Configuration”](#) (page 60).

Allow remote administration by selecting *Allow Remote Administration* in *Remote Administration Settings*. Selecting *Do Not Allow Remote Administration* disables this function. Click *Open Port in Firewall* to allow access to your computer. Clicking *Firewall Details* displays network interfaces with open ports in the firewall. Select the desired interface and click *OK* to return to the main dialog. Click *Accept* to complete the configuration.

The YaST *Remote Administration* module is highly recommended for configuring VNC on your machine. Although the SaX2 interface also allows you to set remote access properties, it is not a substitute for YaST. It only enables you to configure your X server as a host for VNC sessions. For more information, refer to [Section 7.13.6, “Remote Access Properties”](#) (page 182).

Routing

Use *Routing* to configure the paths data takes over the network. In most cases, only enter the IP address of the system through which to send all data in *Default Gateway*. To create more complicated configurations, use *Expert Configuration*.

Windows Domain Membership

In a heterogeneous network consisting of Linux and Windows hosts, Samba controls the communication between the two worlds. With the *Samba Client* module, you can configure your computer as member of a Windows domain. Find information about Samba and the configuration of clients in [Chapter 35, *Samba*](#) (page 669).

7.8 AppArmor

Novell AppArmor is designed to provide easy-to-use application security for both servers and workstations. Novell AppArmor is an access control system that lets you specify which files each program may read, write, and execute. To enable or disable Novell AppArmor on your system, use *AppArmor Control Panel*. Information about Novell AppArmor and a detailed description of the configuration with YaST are available in *Novell AppArmor 2.0 Administration Guide* ([↑Novell AppArmor 2.0 Administration Guide](#)).

7.9 Security and Users

A basic aspect of Linux is its multiuser capability. Consequently, several users can work independently on the same Linux system. Each user has a user account identified by a login name and a personal password for logging in to the system. All users have their own home directories where personal files and configurations are stored.

7.9.1 User Management

Create and edit users with *Security and Users* → *User Management*. It provides an overview of users in the system, including NIS, LDAP, Samba, and Kerberos users if requested. If you are part of an extensive network, click *Set Filter* to list all users categorically (for example, `root` or NIS users). You can also customize filter settings by clicking *Customize Filter*.

To add new users, click *Add* and enter the appropriate data. Complete the addition by clicking *Accept*. The new user can immediately log in using the newly created login name and password.

TIP: Autologin

If you are the only user of your system, you can configure autologin. Autologin automatically logs a user into the system after it starts. To activate autologin, select the user from the list of users and click *Login Settings*. Then choose *Autologin* and click *OK*.

Disable user login with the corresponding option. Fine-tune user profiles in *Details*. Here, manually set the user ID, home directory, default login shell, and assign the new user to specific groups. Configure the validity of the password in *Password Settings*. Click *Accept* to save all changes.

To delete a user, select the user from the list and click *Delete*. Then mark whether to delete the home directory and click *Yes* to confirm.

For advanced user administration, use *Expert Options* to define the default settings for the creation of new users. Select the user authentication method (such as NIS, LDAP, Kerberos, or Samba), login settings (only with KDM or GDM), and the algorithm for password encryption. *Default for New Users* and *Password Encryption* apply only to local users. *Authentication and User Sources* provides a configuration overview and the option to configure the client. Advanced client configuration is also possible using this module. After accepting the configuration, return to the initial configuration overview. Click *Write Changes Now* to save all changes without exiting the configuration module.

7.9.2 Group Management

To create and edit groups, select *Security and Users* → *Group Management* or click *Groups* in the user administration module. Both dialogs have the same functionality, allowing you to create, edit, or delete groups.

The module gives an overview of all groups. As in the user management dialog, change filter settings by clicking *Set Filter*.

To add a group, click *Add* and fill in the appropriate data. Select group members from the list by checking the corresponding box. Click *Accept* to create the group. To edit a group, select the group to edit from the list and click *Edit*. Make all necessary changes then save them with *Accept*. To delete a group, simply select it from the list and click *Delete*.

Click *Expert Options* for advanced group management. Find more about these options in [Section 7.9.1, “User Management”](#) (page 164).

7.9.3 Local Security

To apply a set of security settings to your entire system, use *Security and Users* → *Local Security*. These settings include security for booting, login, passwords, user creation, and file permissions. SUSE Linux Enterprise offers three preconfigured security sets: *Home Workstation*, *Networked Workstation*, and *Networked Server*. Modify the defaults with *Details*. To create your own scheme, use *Custom Settings*.

The detailed or custom settings include:

Password Settings

To have new passwords checked by the system for security before they are accepted, click *Check New Passwords* and *Test for Complicated Passwords*. Set the minimum password length for newly created users. Define the period for which the password should be valid and how many days in advance an expiration alert should be issued when the user logs in to the text console.

Boot Settings

Set how the key combination `Ctrl` + `Alt` + `Del` should be interpreted by selecting the desired action. Normally, this combination, when entered in the text console, causes the system to reboot. Do not modify this setting unless your machine or

server is publicly accessible and you are afraid someone could carry out this action without authorization. If you select *Stop*, this key combination causes the system to shut down. With *Ignore*, this key combination is ignored.

If you use the KDE login manager (KDM), set permissions for shutting down the system in *Shutdown Behavior of KDM*. Give permission to *Only root* (the system administrator), *All Users*, *Nobody*, or *Local Users*. If *Nobody* is selected, the system can only be shut down from the text console.

Login Settings

Typically, following a failed login attempt, there is a waiting period lasting a few seconds before another login is possible. This makes it more difficult for password sniffers to log in. Optionally activate *Record Successful Login Attempts* and *Allow Remote Graphical Login*. If you suspect someone is trying to discover your password, check the entries in the system log files in `/var/log`. To grant other users access to your graphical login screen over the network, enable *Allow Remote Graphical Login*. Because this access possibility represents a potential security risk, it is inactive by default.

User Addition

Every user has a numerical and an alphabetical user ID. The correlation between these is established using the file `/etc/passwd` and should be as unique as possible. Using the data in this screen, define the range of numbers assigned to the numerical part of the user ID when a new user is added. A minimum of 500 is suitable for users. Automatically generated system users start with 1000. Proceed in the same way with the group ID settings.

Miscellaneous Settings

To use predefined file permission settings, select *Easy*, *Secure*, or *Paranoid*. *Easy* should be sufficient for most users. The setting *Paranoid* is extremely restrictive and can serve as the basic level of operation for custom settings. If you select *Paranoid*, remember that some programs might not work correctly or even at all, because users no longer have permission to access certain files.

Also set which user should launch the `updatedb` program, if installed. This program, which automatically runs on a daily basis or after booting, generates a database (locatedb) in which the location of each file on your computer is stored. If you select *Nobody*, any user can find only the paths in the database that can be seen by any other (unprivileged) user. If `root` is selected, all local files are indexed, because the user `root`, as superuser, may access all directories. Make sure that

the options *Current Directory in root's Path* and *Current Directory in Path of Regular Users* are deactivated. Only advanced users should consider using these options because these settings may pose a significant security risk if used incorrectly. To have some control over the system even if it crashes, click *Enable Magic SysRq Keys*.

Click *Finish* to complete your security configuration.

7.9.4 Firewall

SuSEfirewall2 can protect your machine against attacks from the Internet. Configure it with *Security and Users* → *Firewall*. Find detailed information about SuSEfirewall2 in [Chapter 38, Masquerading and Firewalls](#) (page 709).

TIP: Automatic Activation of the Firewall

YaST automatically starts a firewall with suitable settings on every configured network interface. Start this module only if you want to reconfigure the firewall with custom settings or deactivate it.

7.10 Miscellaneous

The YaST Control Center has several modules that cannot easily be classified into the first six module groups. They can be used for things like viewing log files and installing drivers from a vendor CD.

7.10.1 Autoinstallation

The part of SUSE Linux Enterprise is the AutoYaST tool for automated installation. In *Miscellaneous* → *Autoinstallation*, prepare profiles for this tool. Find detailed information about automated installation with AutoYaST in [Chapter 5, Automated Installation](#) (page 95). Information about using the *Autoinstallation* module is provided in [Section 5.1.1, “Creating an AutoYaST Profile”](#) (page 96).

7.10.2 Support Query

Miscellaneous → *Support Query* offers the possibility to collect all system information needed by the support team to find your problem so you can get help to solve it as soon as possible. Regarding your query, select the problem category in the following window. When all information is gathered, attach it to your support request.

7.10.3 Release Notes

The release notes are an important source about installation, update, configuration, and technical issues. The release notes are continuously updated and published through online update. Use *Miscellaneous* → *Release Notes* to view the release notes.

7.10.4 Start-Up Log

View information concerning the start-up of the computer in *Miscellaneous* → *Start-Up Log*. This is one of the first places you might want to look when encountering problems with the system or when troubleshooting. It shows the boot log `/var/log/boot.msg`, which contains the screen messages displayed when the computer starts. Viewing the log can help determine if the computer started properly and if all services and functions were started correctly.

7.10.5 System Log

Use *Miscellaneous* → *System Log* to view the system log that keeps track of the operations of your computer in `var/log/messages`. Kernel messages, sorted according to date and time, are also recorded here. View the status of certain system components using the box at the top. The following options are possible from the system log and boot log modules:

`/var/log/messages`

This is the general system log file. Here, view kernel messages, users logging in as `root`, and other useful information.

`/proc/cpuinfo`

This displays processor information, including its type, make, model, and performance.

`/proc/dma`

This shows which DMA channels are currently being used.

`/proc/interrupts`

This shows which interrupts are in use and how many of each have been in use.

`/proc/iomem`

This displays the status of input/output memory.

`/proc/ioports`

This shows which I/O ports are in use at the moment.

`/proc/meminfo`

This displays memory status.

`/proc/modules`

This displays the individual modules.

`/proc/mounts`

This displays devices currently mounted.

`/proc/partitions`

This shows the partitioning of all hard disks.

`/proc/version`

This displays the current version of Linux.

`/var/log/YaST2/y2log`

This displays all YaST log messages.

`/var/log/boot.msg`

This displays information concerning the start-up of the system.

`/var/log/faillog`

This displays login failures.

`/var/log/warn`

This displays all system warnings.

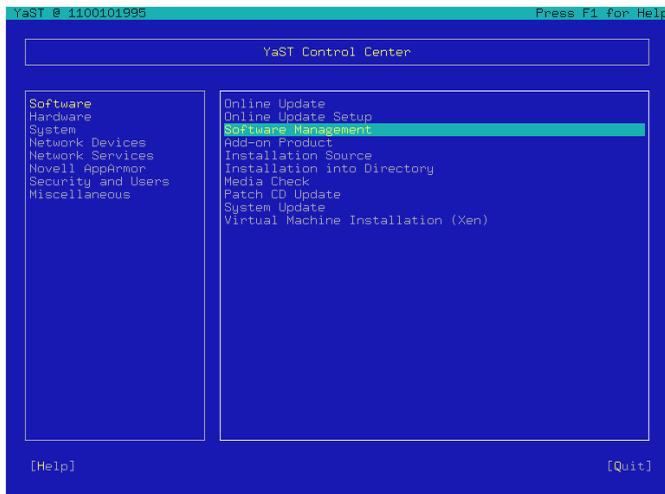
7.10.6 Vendor Driver CD

Install device drivers from a Linux driver CD that contains drivers for SUSE Linux Enterprise with *Miscellaneous* → *Vendor Driver CD*. When installing SUSE Linux Enterprise from scratch, use this YaST module to load the required drivers from the vendor CD after the installation.

7.11 YaST in Text Mode

When YaST is started in text mode, the YaST Control Center appears first. See [Figure 7.7](#), “Main Window of YaST in Text Mode” (page 170). The main window consists of three areas. The left frame, which is surrounded by a thick white border, features the categories to which the various modules belong. The active category is indicated by a colored background. The right frame, which is surrounded by a thin white border, provides an overview of the modules available in the active category. The bottom frame contains the buttons for *Help* and *Exit*.

Figure 7.7 Main Window of YaST in Text Mode



When the YaST Control Center is started, the category *Software* is selected automatically. Use and to change the category. To start a module from the selected category, press . The module selection now appears with a thick border. Use and to select

the desired module. Keep the arrow keys pressed to scroll through the list of available modules. When a module is selected, the module title appears with a colored background and a brief description is displayed in the bottom frame.

Press **[Enter]** to start the desired module. Various buttons or selection fields in the module contain a letter with a different color (yellow by default). Use **[Alt] + [yellow_letter]** to select a button directly instead of navigating there with **[Tab]**. Exit the YaST Control Center by pressing the *Exit* button or by selecting *Exit* in the category overview and pressing **[Enter]**.

7.11.1 Navigation in Modules

The following description of the control elements in the YaST modules assumes that all function keys and **[Alt]** key combinations work and are not assigned different global functions. Read [Section 7.11.2, “Restriction of Key Combinations”](#) (page 172) for information about possible exceptions.

Navigation among Buttons and Selection Lists

Use **[Tab]** and **[Alt] + [Tab]** or **[Shift] + [Tab]** to navigate among the buttons and the frames containing selection lists.

Navigation in Selection Lists

Use the arrow keys (**[↑]** and **[↓]**) to navigate among the individual elements in an active frame containing a selection list. If individual entries within a frame exceed its width, use **[Shift] + [→]** or **[Shift] + [←]** to scroll horizontally to the right and left. Alternatively, use **[Ctrl] + [E]** or **[Ctrl] + [A]**. This combination can also be used if using **[→]** or **[←]** would result in changing the active frame or the current selection list, as in the Control Center.

Buttons, Radio Buttons, and Check Boxes

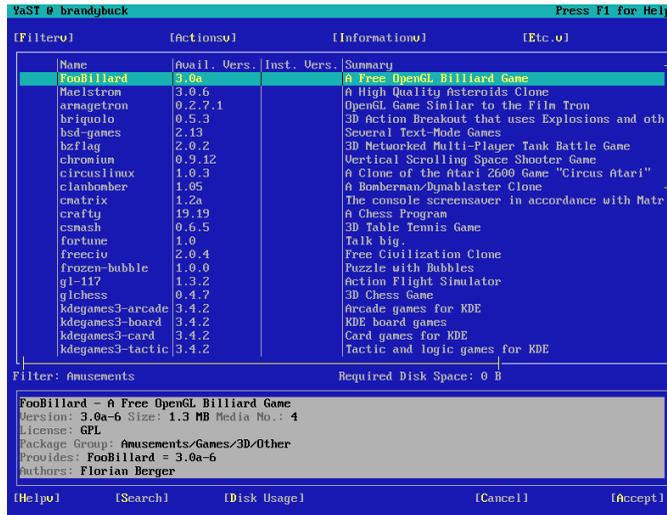
To select buttons with empty square brackets (check boxes) or empty parentheses (radio buttons), press **[Space]** or **[Enter]**. Alternatively, radio buttons and check boxes can be selected directly with **[Alt] + [yellow_letter]**. In this case, you do not need to confirm with **[Enter]**. If you navigate to an item with **[Tab]**, press **[Enter]** to execute the selected action or activate the respective menu item.

Function Keys

The F keys (**[F1]** to **[F12]**) enable quick access to the various buttons. Which function keys are actually mapped to which buttons depends on the active YaST module,

because the different modules offer different buttons (Details, Info, Add, Delete, etc.). Use **[F10]** for *OK*, *Next*, and *Finish*. Press **[F1]** to access the YaST help, which shows the functions mapped to the individual F keys.

Figure 7.8 *The Software Installation Module*



7.11.2 Restriction of Key Combinations

If your window manager uses global **[Alt]** combinations, the **[Alt]** combinations in YaST might not work. Keys like **[Alt]** or **[Shift]** can also be occupied by the settings of the terminal.

Replacing **[Alt]** with **[Esc]**

[Alt] shortcuts can be executed with **[Esc]** instead of **[Alt]**. For example, **[Esc] + [H]** replaces **[Alt] + [H]**.

Backward and Forward Navigation with **[Ctrl] + [F]** and **[Ctrl] + [B]**

If the **[Alt]** and **[Shift]** combinations are occupied by the window manager or the terminal, use the combinations **[Ctrl] + [F]** (forward) and **[Ctrl] + [B]** (backward) instead.

Restriction of Function Keys

The F keys are also used for functions. Certain function keys might be occupied by the terminal and may not be available for YaST. However, the `[Alt]` key combinations and function keys should always be fully available on a pure text console.

7.11.3 Starting the Individual Modules

To save time, the individual YaST modules can be started directly. To start a module, enter:

```
yast <module_name>
```

View a list of all module names available on your system with `yast -l` or `yast --list`. Start the network module, for example, with `yast lan`.

7.12 Update from the Command Line

SUSE Linux Enterprise comes with a new command line tool for installing and updating packages, `rug`. It works with the `rcd` daemon to install, update, and remove software according to the commands given. It can install software from local files or from servers. You may use one or more remote servers, known as services. Supported services are `mount` for local files and `yum` or `ZENworks` for servers.

`rug` sorts software from services into catalogs (also known as channels), groups of similar software. For example, one catalog might contain software from an update server and another some software from a third-party software vendor. Subscribe to individual catalogs to control the display of available packages and prevent the accidental installation of unwanted software. Operations are normally performed only on software from catalogs to which you are subscribed.

The most commonly used command is `rug update`, which downloads and installs patches in catalogs to which you are subscribed. If you only want to update software, this is the only command you need. To obtain a list of all packages from one catalog, use `rug pa catalogname`. Replace `catalogname` with name of your catalog. To list all available services, use `rug sl`. Some other useful `rug` commands and their functions are shown in [Table 7.1, “rug Commands”](#) (page 174).

Table 7.1 *rug Commands*

Command	Function
ca	List the catalogs
sa	Add a service
reg	Register a service
sub	Subscribe to a catalog
refresh	Refresh the lists of patches

7.12.1 rug User Management

One of the biggest advantages of rug is user management. Normally only `root` can update or install new packages. With rug, you can distribute the right to update the system to other users and restrict them, for example, only to the update right without the possibility to remove software. Privileges you can grant are:

install

User may install new software

lock

User may set package locks

remove

User may remove software

subscribe

User may change channel subscriptions

trusted

User is considered trusted, so may install packages without package signatures

upgrade

User may update software packages

view

This allows the user to see which software is installed on the machine and which software is in available channels. The option is relevant only to remote users, local users are normally permitted to view installed and available packages.

superuser

Permits all rug commands except user management and settings, which must be done locally.

To give a user permission to update the system, use the command `rug ua username upgrade`. Replace *username* with the name of the user. To revoke the privileges of a user, use command `rug ud username`. To list users with their rights, use `rug ul`.

To change the current privileges of a user, use `rug ue username`. Replace *username* with name of the desired user. The edit command is interactive. It lists privileges of the selected user and the offers you a prompt. Enter the plus (+) or minus (-) symbol and the name of the privilege then press `[Enter]`. For example, to permit the user to delete software, enter `+remove`. To save and quit, press `[Enter]` on a blank line.

7.12.2 Scheduling Updates

Using `rug`, the system can be updated automatically, for example, with scripts. The simplest example is the fully automatic update. To do this, as `root` configure a cron job that executes `rug up -y`. The `up -y` option downloads and installs the patches from your catalogs without confirmation.

However, you may not want the patches installed automatically. Instead, you may want to retrieve the patches and select the patches for installation at a later time. To download patches only, use the command `rug up -dy`. The `up -dy` option downloads the patches from your catalogs without confirmation and saves them to the rug cache. The default location of the rug cache is `/var/cache/redcarpet`.

7.12.3 Configuring rug

`rug` comes with many preferences to provide the update functionality in different network configurations. To list the preferences that may be set, use `rug get`. To set a preference variable, enter `rug set`. For example, adjust settings if you need to update your system,

but the computer is behind a proxy server. Before downloading updates, send your username and password to the proxy server. To do so, use the commands:

```
rug set proxy-url url_path
    rug set proxy-username name
    rug set proxy-password password
```

Replace `url_path` with the name of your proxy server. Replace `name` with your username. Replace `password` with your password.

7.12.4 For More Information

For more information about updating from the command line, enter `rug --help` or see the `rug(1)` man page. The `--help` option is also available for all `rug` commands. If, for example, you want to read help for `rug update`, enter `rug update --help`.

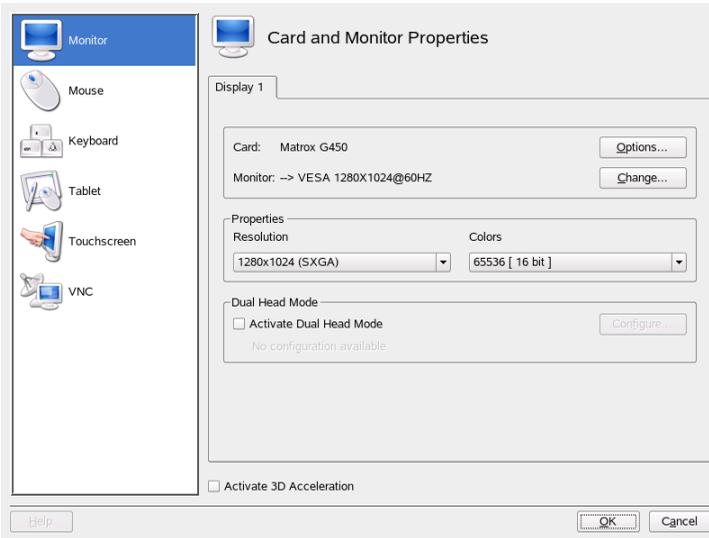
7.13 SaX2

Configure the graphical environment of your system with *Hardware* → *Graphics Card and Monitor*. This opens the SUSE Advanced X11 Configuration interface (SaX2), where you can configure devices such as your mouse, keyboard, or display devices. This interface can also be accessed from the main menu by clicking *System* → *Configuration* → *SaX2*.

7.13.1 Card and Monitor Properties

Adjust the settings for your graphics card and display device in *Card and Monitor Properties*. If you have more than one graphics card installed, each device is shown in a separate dialog reachable by a tab. At the top of the dialog, see the current settings for the selected graphics card and the monitor that is attached to it. If more than one screen can be connected to the card (dual head), the monitor on the primary output is shown. Normally, the card and display device are detected automatically by the system during installation. However, you can tune many parameters manually or even change the display device completely.

Figure 7.9 Card and Monitor Properties



TIP: Autodetecting New Display Hardware

If you change your display hardware after installation, use `sax2 -r` on the command line to cause SaX2 to detect your hardware. You must be `root` to run SaX2 from the command line.

Graphics Card

It is not possible to change the graphics card because only known models are supported and these are detected automatically. However, you can change many options that affect the behavior of the card. Normally, this should not be necessary because the system already has set them up appropriately during installation. If you are an expert and want to tweak some of the options, click *Options* next to the graphics card and select the option to change. To assign a value needed to a certain option, enter this value in the dialog that appears after selecting that option. Click *OK* to close the options dialog.

Monitor

To change the current settings for the monitor, click *Change* next to the monitor. A new dialog opens in which to adjust various monitor-specific settings. This dialog has several tabs for various aspects of monitor operation. Select the first tab to manually select the vendor and model of the display device in two lists. If your monitor is not listed, you can choose one of the VESA or LCD modes that suit your needs or, if you have a vendor driver disk or CD, click *Utility Disk* and follow the instructions on the screen to use it. Check *Activate DPMS* to use display power management signaling. *Display Size*, with the geometrical properties of the monitor, and *Sync Frequencies*, with the ranges for the horizontal and vertical sync frequencies of your monitor, are normally set up correctly by the system, but you can modify these values manually. After making all adjustments, click *OK* to close this dialog.

WARNING: Changing Monitor Frequencies

Although there are safety mechanisms, you should still be very careful when changing the allowed monitor frequencies manually. Incorrect values might destroy your monitor. You should always refer to the monitor's manual before changing frequencies.

Resolution and Color Depth

The resolution and color depth can be chosen directly from two lists in the middle of the dialog. The resolution you select here marks the highest resolution to use. All common resolutions down to 640x480 are also added to the configuration automatically. Depending on the graphical desktop used, you can switch to any of these later without the need for reconfiguration.

Dual Head

If you have a graphics card with two outputs installed in your computer, you can connect two screens to your system. Two screens that are attached to the same graphics card are referred to as *dual head*. SaX2 automatically detects multiple display devices in the system and prepares the configuration accordingly. To use the dual head mode of a graphics card, check *Activate Dual Head Mode* at the bottom of the dialog and click *Configure* to set the dual head options and the arrangement of the screens in the dual head dialog.

The tabs in the row at the top of the dialog each correspond to a graphics card in your system. Select the card to configure and set its multihead options in the dialog below. In the upper part of the multihead dialog, click *Change* to configure the additional screen. The possible options are the same as for the first screen. Choose the resolution to use for this screen from the list. Select one of three possible multihead modes.

Traditional Multihead

Each monitor represents an individual unit. The mouse pointer can switch between the screens.

Cloned Multihead

In this mode, all monitors display the same contents. The mouse is only visible on the main screen.

Xinerama Multihead

All screens combine to form a single large screen. Program windows can be positioned freely on all screens or scaled to a size that fills more than one monitor.

NOTE

Linux currently does not offer 3D support for Xinerama multihead environments. In this case, SaX2 deactivates the 3D support.

The arrangement of the dual head environment describes the sequence of the individual screens. By default, SaX2 configures a standard layout that follows the sequence of the detected screens, arranging all screens in a row from left to right. In the *Arrangement* part of the dialog, determine the way the monitors are arranged by selecting one of the sequence buttons. Click *OK* to close the dialog.

TIP: Using a Beamer with Laptop Computers

To connect a beamer to a laptop computer, activate dual head mode. In this case, SaX2 configures the external output with a resolution of 1024x768 and a refresh rate of 60 Hz. These values suit most beamers very well.

Multihead

If you have more than one graphics card installed in your computer, you can connect more than one screen to your system. Two or more screens that are attached to different

graphics cards are referred to as *multihead*. SaX2 automatically detects multiple graphics cards in the system and prepares the configuration accordingly. By default, SaX2 configures a standard layout that follows the sequence of the detected graphics cards, arranging all screens in a row from left to right. The additional *Arrangement* tab allows for changing this layout manually. Drag the icons representing the individual screens in the grid and click *OK* to close the dialog.

Testing the Configuration

Click *OK* in the main window after completing the configuration of your monitor and your graphics card, then test your settings. This ensures that your configuration is suitable for your devices. If the image is not steady, terminate the test immediately by pressing `Ctrl+Alt+Backspace` and reduce the refresh rate or the resolution and color depth.

NOTE

Regardless of whether you run a test, all modifications are only activated when you restart the X server.

7.13.2 Mouse Properties

Adjust the settings for your mouse in *Mouse Properties*. If you have more than one mouse with different drivers installed, each driver is shown in a separate tab. Multiple devices operated by the same driver are shown as one mouse. Activate or deactivate the currently selected mouse with the check box at the top of the dialog. Below the check box, see the current settings for that mouse. Normally, the mouse is detected automatically, but you can change it manually if the automatic detection fails. Refer to the documentation for your mouse for a description of the model. Click *Change* to select the vendor and model from two lists then click *OK* to confirm your selection. In the options part of the dialog, set various options for operating your mouse.

Activate 3-Button Emulation

If your mouse has only two buttons, a third button is emulated when you click both buttons simultaneously.

Activate Mouse Wheel

Check this box to use a scroll wheel.

Emulate Wheel with Mouse Button

If your mouse does not have a scroll wheel but you want to use similar functionality, you can assign an additional button for this. Select the button to use. While pressing this button, any movement of the mouse is translated into scroll wheel commands. This feature is especially useful with trackballs.

When you are satisfied with your settings, click *OK* to confirm your changes.

NOTE

Any changes you make here take effect only after you restart the X server.

7.13.3 Keyboard Properties

Use this dialog to adjust the settings for operating your keyboard in the graphical environment. In the upper part of the dialog, select the type, language layout, and variant. Use the test field at the bottom of the dialog to check if special characters are displayed correctly. Select additional layouts and variants to use from the list in the middle. Depending on the type of your desktop, these may be switched in the running system without the need for reconfiguration. After you click *OK*, the changes are applied immediately.

7.13.4 Tablet Properties

Use this dialog to configure a graphics tablet attached to your system. Click the *Graphics Tablet* tab to select vendor and model from the lists. Currently, SUSE® Linux Enterprise supports only a limited number of graphics tablets. To activate the tablet, check *Activate This Tablet* at the top of the dialog.

In the *Port and Mode* dialog, configure the connection to the tablet. SaX2 enables the configuration of graphics tablets connected to the USB port or the serial port. If your tablet is connected to the serial port, verify the port. `/dev/ttyS0` refers to the first serial port. `/dev/ttyS1` refers to the second. Additional ports use similar notation. Choose appropriate *Options* from the list and select the *Primary Tablet Mode* suitable for your needs.

If your graphics tablet supports electronic pens, configure them in *Electronic Pens*. Add eraser and pen and set their properties after clicking *Properties*.

When you are satisfied with the settings, click *OK* to confirm your changes.

7.13.5 Touchscreen Properties

Use this dialog to configure touchscreens attached to your system. If you have more than one touchscreen installed, each device is shown in a separate dialog reachable by a tab. To activate the currently selected touchscreen, check *Assign a Touchscreen to Display* at the top of the dialog. Select vendor and model from the lists below and set an appropriate *Connection Port* at the bottom. You can configure touchscreens connected to the USB port or the serial port. If your touchscreen is connected to the serial port, verify the port. `/dev/ttyS0` refers to the first serial port. `/dev/ttyS1` refers to the second. Additional ports use similar notation. When you are satisfied with your settings, click *OK* to confirm your changes.

7.13.6 Remote Access Properties

VNC (*Virtual Network Computing*) is a client-server solution that gives access a remote X server with a slim and easy-to-use client. This client is available for a variety of operating systems, including Microsoft Windows, Apple's MacOS, and Linux. Find additional information about VNC at <http://www.realvnc.com/>.

Use this dialog to configure your X server as a host for VNC sessions. If you want VNC clients to connect to your X server, check *Allow Access to Display Using VNC Protocol*. Set a password to restrict access to your VNC-enabled X server. Check *Allow Multiple VNC Connections* if more than one VNC client should connect to the X server at the same time. Allow HTTP access by checking *Activate HTTP Access* and setting the port to be use in *HTTP Port*.

When you are satisfied with your settings, click *OK* to save your changes.

7.14 Troubleshooting

All error messages and alerts are logged in the directory `/var/log/YaST2`. The most important file for finding YaST problems is `y2log`.

7.15 For More Information

More information about YaST can be found on the following Web sites and directories:

- `/usr/share/doc/packages/yast2`—Local YaST development documentation
- http://www.opensuse.org/YaST_Development—The YaST project page in the openSUSE wiki
- <http://forge.novell.com/modules/xfmod/project/?yast>—Another YaST project page

Updating SUSE Linux Enterprise

8

SUSE® Linux Enterprise provides the option of updating an existing system to the new version without completely reinstalling it. No new installation is needed. Old data such as home directories and system configuration, is kept intact. During the life cycle of the product, you can apply Service Packs to increase system security and correct software defects.

8.1 Updating SUSE Linux Enterprise

Software tends to “grow” from version to version. Therefore, take a look at the available partition space with `df` before updating. If you suspect you are running short of disk space, secure your data before updating and repartition your system. There is no general rule of thumb regarding how much space each partition should have. Space requirements depend on your particular partitioning profile and the software selected.

8.1.1 Preparations

Before updating, copy the old configuration files to a separate medium, such as streamer, removable hard disk, USB stick, or ZIP drive, to secure the data. This primarily applies to files stored in `/etc` as well as some of the directories and files in `/var` and `/opt`. You may also want to write the user data in `/home` (the `HOME` directories) to a backup medium. Back up this data as `root`. Only `root` has read permission for all local files.

Before starting your update, make note of the root partition. The command `df /` lists the device name of the root partition. In [Example 8.1, “List with `df -h`”](#) (page 186), the root partition to write down is `/dev/hda3` (mounted as `/`).

Example 8.1 *List with `df -h`*

```
Filesystem      Size  Used Avail Use% Mounted on
/dev/hda3        74G   22G   53G  29% /
tmpfs           506M    0  506M   0% /dev/shm
/dev/hda5       116G   5.8G  111G   5% /home
/dev/hda1        39G   1.6G   37G   4% /windows/C
/dev/hda2        4.6G   2.6G   2.1G  57% /windows/D
```

8.1.2 Possible Problems

If you update a default system from the previous version to this version, YaST works out necessary changes and performs them. Depending on your customizations, some steps or the entire update procedure may fail and you must resort to copying back your backup data. Check the following issues before starting the system update.

Checking `passwd` and `group` in `/etc`

Before updating the system, make sure that `/etc/passwd` and `/etc/group` do not contain any syntax errors. For this purpose, start the verification utilities `pwck` and `grpck` as `root` and eliminate any reported errors.

PostgreSQL

Before updating PostgreSQL (`postgres`), dump the databases. See the manual page of `pg_dump`. This is only necessary if you actually used PostgreSQL prior to your update.

8.1.3 Updating with YaST

Following the preparation procedure outlined in [Section 8.1.1, “Preparations”](#) (page 185), you can now update your system:

- 1 Optionally, prepare an installation server. For background information, see [Section 4.2.1, “Setting Up an Installation Server Using YaST”](#) (page 68).

- 2 Boot the system as for the installation, described in [Section 3.1, “System Start-Up for Installation”](#) (page 33). In YaST, choose a language and select *Update* in the *Installation Mode* dialog. Do not select *New Installation*.
- 3 YaST determines whether there are multiple root partitions. If there is only one, continue with the next step. If there are several, select the right partition and confirm with *Next* (`/dev/hda3` was selected in the example in [Section 8.1.1, “Preparations”](#) (page 185)). YaST reads the old `fstab` on this partition to analyze and mount the file systems listed there.
- 4 In the *Installation Settings* dialog, adjust the settings according to your requirements. Normally, you can leave the default settings untouched, but if you intend to enhance your system, check the packages offered in the *Software Selection* submenus or add support for additional languages.

You also have the possibility to make backups of various system components. Selecting backups slows down the update process. Use this option if you do not have a recent system backup.

- 5 In the following dialog, choose to update only the software that is already installed or to add new software components to the system (upgrade mode). It is advisable to accept the suggested composition. Adjustments can be made later with YaST.

8.2 Installing Service Packs

Use Service Packs to update a SUSE Linux Enterprise installation. There are several different ways in which you can apply a Service Pack. You can either update the existing installation or start a whole new installation using the Service Pack media. Possible scenarios for updating the system and setting up a central network installation source are described here.

TIP

Read the installation instructions on the Service Pack media for further changes.

8.2.1 Setting Up a Network Installation Source for Service Pack Media

As with the initial installation of SUSE Linux Enterprise, it is much more efficient having a central installation source on your network to serve all clients rather than installing all of them separately using a set of physical media.

Configuring a Network Installation Source on SUSE Linux Enterprise Using YaST

Basically, follow the procedure outlined in [Section 4.2.1, “Setting Up an Installation Server Using YaST”](#) (page 68). Just add another installation source called `SLE-10-SP-x-arch`, `SLES-10-SP-x-arch`, or `SLED-10-SP-x-arch` (where *x* is the number of the Service Pack and *arch* is the name of your hardware architecture) and make it available via NFS, HTTP, or FTP.

Setting Up a Network Installation Source Manually

The procedure for manual setup of a network installation source is very similar to the one described in [Section 4.2.2, “Setting Up an NFS Installation Source Manually”](#) (page 70). A few minor changes must be applied to the procedures for setting up FTP, HTTP, and SMB installation sources.

When copying the installation media, proceed as follows:

- 1 Log in as `root`.
- 2 Change to the directory that is already holding the original installation media (in this example, in `/install/sle` as `SLE-10-arch/CD1`) by entering

```
cd /install/sle
```

- 3 Create a new subdirectory called, for example, `SLE-10-SP-x-arch` (replacing *x* with the number of the SP and *arch* with the name of your hardware architecture) by entering

```
mkdir SLE-10-SP-x-arch
```

- 4 Copy the contents of each SP installation medium to its own subdirectory. Once done the directory hierarchy is as follows:

```
/install/sle/SLE-10-arch/CD1
                        /CD2
                        /CD3
                        /CD4
/install/sle/SLE-10-SP-x-arch/CD1
                        /CD2
                        /CD3
```

- 5 In `SLE-10-arch/CD1`, create a file called `add_on_products`. The contents of `add_on_products` determines which Service Pack should be added to your SUSE Linux Enterprise 10 as an add-on product. The format of the file is as follows:

```
media_url [path_on_media [product_1 [product_2 [...]]]
```

For example, if you want to offer the SP via NFS from `sun.example.com` write the following in `add_on_products` (specifying `path_on_media`, `product_1`, etc., is not required for Service Packs):

```
nfs://sun.example.com/install/sle/SLE-10-SP-x-arch/CD1
```

Replace `x` with the actual number of the Service Pack and `arch` with the name of your architecture.

- 6 Make the sources available via NFS, FTP, or HTTP as described in [Section 4.2, “Setting Up the Server Holding the Installation Sources”](#) (page 68).

8.2.2 Installing a Service Pack

NOTE

To update an existing SUSE Linux Enterprise 10 system to a SUSE Linux Enterprise 10 Service Pack (SP), see [Section 8.2.3, “Updating to a Service Pack”](#) (page 193).

Installing a SUSE Linux Enterprise Service Pack is very similar to installing the original SUSE Linux Enterprise media. As with the original installation, you can choose between installing from a local CD or DVD drive or from a network server. For more information about both these possibilities, see [Section 3.1.2, “Selecting the Source of the Installation Data”](#) (page 34).

Installing from a Local CD or DVD Drive

Before starting a new installation of a SUSE Linux Enterprise SP, ensure that the following prerequisite items are available:

- The original SUSE Linux Enterprise installation media (CDs or DVD)
- All of the Service Pack installation media (CDs or DVD)

There are two ways to install an SUSE Linux Enterprise SP system from scratch: either boot from the original installation medium and register the Service Pack as an add-on product as outlined in [Booting from the Original Installation Medium](#) (page 190) or boot from the SP medium and insert the original media, when YaST asks for them—see [Booting from the Service Pack Medium](#) (page 190).

Procedure 8.1 *Booting from the Original Installation Medium*

- 1** Insert the original SUSE Linux Enterprise 10 boot medium and boot your machine. For detailed information, see [Section 3.1, “System Start-Up for Installation”](#) (page 33).
- 2** In the YaST, *Installation Mode* dialog, select *New Installation* and activate *Include Add-On Products from Separate Media*.
- 3** YaST asks you to register the SP medium as an add-on product. Once done, run the regular installation as described in [Chapter 3, *Installation with YaST*](#) (page 33).

Procedure 8.2 *Booting from the Service Pack Medium*

- 1** Insert the first SUSE Linux Enterprise SP medium (CD or DVD number 1) and boot your machine. A boot screen similar to the original installation of SUSE Linux Enterprise 10 is displayed.
- 2** Select *Installation* to boot the SP kernel. The kernel loads. Wait until you are prompted to insert CD1 of the product.
- 3** Insert the original SUSE Linux Enterprise 10 installation medium number 1 then click *OK* to run the regular installation.
- 4** Accept the license agreement then select a language, default desktop, and other installation settings.

- 5 Click *Yes, Install* to start the installation.
- 6 Insert the appropriate media when prompted. Both the SP media and the original product media are required, depending on the software installed.
- 7 Continue as usual with the installation (entering a password for `root`, completing the network configuration, testing your Internet connection, activating the ZENworks® Online Update Service, selecting the user authentication method, and entering a username and password).

For instructions on installing SLED, see the *SUSE Linux Enterprise Desktop 10 Installation Quick Start* [<http://www.novell.com/documentation/sled10/>] or `quickstart-en.pdf` at the root of the SUSE Linux Enterprise Desktop CD 1 or DVD.

Network Installation

Before starting a network installation of an SUSE Linux Enterprise SP, make sure that the following requirements are met:

- A network installation source set up according to [Section 8.2.1, “Setting Up a Network Installation Source for Service Pack Media”](#) (page 188).
- A working network connection both on the installation server and the target machine that includes a name service, DHCP (optional, but needed for PXE boot), and OpenSLP (optional).
- The SUSE Linux Enterprise SP CD or DVD number 1 to boot the target system *or* a target system set up for PXE boot according to [Section 4.3.5, “Preparing the Target System for PXE Boot”](#) (page 86).

Network Installation—Boot from CD or DVD

To perform a network installation using the SP CD or DVD as the boot medium, proceed as follows:

- 1 Insert the SUSE Linux Enterprise SP CD or DVD number 1 and boot your machine. A boot screen similar to the original installation of SUSE Linux Enterprise 10 is displayed.

- 2 Select *Installation* to boot the SP kernel from CD, then use **[F3]** to enable *Further Options*, and finally **[F4]** to select a type of network installation source (FTP, HTTP, NFS, or SMB).
- 3 Provide the appropriate path information or select *SLP* as the installation source.
- 4 Select the appropriate installation server from those offered or use the boot options prompt to provide the type of installation source and its actual location as in [Section “Installing from a Network Server”](#) (page 35). YaST starts.
- 5 Accept the license agreement then select a language, default desktop, and other installation settings.
- 6 Click *Yes, Install* to start the installation.
- 7 Continue as usual with the installation (entering a password for `root`, completing the network configuration, testing your Internet connection, activating the Online Update service, selecting the user authentication method, and entering a username and password).

For detailed instructions for installing SUSE Linux Enterprise, see [Chapter 3, *Installation with YaST*](#) (page 33).

Network Installation—PXE Boot

To perform a network installation of a SUSE Linux Enterprise Service Pack via network, proceed as follows:

- 1 Adjust the setup of your DHCP server to provide the address information needed for PXE boot according to [Section 4.3.5, “Preparing the Target System for PXE Boot”](#) (page 86).
- 2 Set up a TFTP server to hold the boot image needed for PXE boot.

Use the first CD or DVD of your SUSE Linux Enterprise Service Pack for this and otherwise follow the instructions in [Section 4.3.2, “Setting Up a TFTP Server”](#) (page 79).

- 3 Prepare PXE boot and Wake-on-LAN on the target machine.

- 4 Initiate the boot of the target system and use VNC to remotely connect to the installation routine running on this machine. See [Section 4.5.1, “VNC Installation”](#) (page 92) for more information.
- 5 Accept the license agreement then select a language, default desktop, and other installation settings.
- 6 Click *Yes, Install* to start the installation.
- 7 Continue as usual with the installation (entering a password for `root`, completing the network configuration, testing your Internet connection, activating the Online Update service, selecting the user authentication method, and entering a username and password).

For detailed instructions for installing SUSE Linux Enterprise, see [Chapter 3, *Installation with YaST*](#) (page 33).

8.2.3 Updating to a Service Pack

You need access to the original SUSE Linux Enterprise 10 CDs or DVD to complete the update.

- 1 In a running SUSE Linux Enterprise system, insert the SUSE Linux Enterprise SP CD or DVD number 1 into your CD drive then click *Yes* when the hardware detection notice appears.

If the boot medium is not detected automatically, try reinserting it. If autodetection continues to fail, select *System* → *Administrator Settings* → *Software* → *Patch CD Update*.

- 2 If you are not logged in as `root`, enter the `root` password when prompted. The *Online Update* dialog appears.
- 3 Click *Next* to begin the update process. *Download and Installation Log* tracks the update progress.
- 4 When prompted for the *Patch CD CD2*, insert *SUSE Linux Enterprise SP CD2* then click *OK*. This prompt does not appear if you are updating from a DVD.

- 5 Click *Finish* when you see *Installation Finished* reported near the end of the progress log.

8.3 Software Changes from Version 9 to Version 10

The individual aspects changed from version 9 to version 10 are outlined in the following in detail. This summary indicates, for example, whether basic settings have been completely reconfigured, whether configuration files have been moved to other places, or whether common applications have been significantly changed. Significant modifications that affect the daily use of the system at either the user level or the administrator level are mentioned here.

8.3.1 Multiple Kernels

It is possible to install multiple kernels side by side. This feature is meant to allow administrators to upgrade from one kernel to another by installing the new kernel, verifying that the new kernel works as expected, then uninstalling the old kernel. While YaST does not yet support this feature, kernels can easily be installed and uninstalled from the shell using `rpm -i package.rpm`.

The default boot loader menus contain one kernel entry. Before installing multiple kernels, it is useful to add an entry for the extra kernels, so they can be selected easily. The kernel that was active before installing the new kernel can be accessed as `vmlinuz.previous` and `initrd.previous`. By creating a boot loader entry similar to the default entry and having this entry refer to `vmlinuz.previous` and `initrd.previous` instead of `vmlinuz` and `initrd`, the previously active kernel can be accessed. Alternatively, GRUB and LILO support wild card boot loader entries. Refer to the GRUB info pages (`info grub`) and to the `lilo.conf(5)` manual page for details.

8.3.2 Changes with Kernel Modules

The following kernel modules are no longer available:

- `km_fcdsl`—AVM Fritz!Card DSL
- `km_fritzcapi`—AVM FRITZ! ISDN Adapters

The following kernel module package was changed internally:

- `km_wlan`—Various drivers for wireless LAN cards. The `madwifi` driver for Atheros WLAN cards from `km_wlan` was removed. `madwifi` is available as a stand-alone package.

For technical reasons, it was necessary to drop support for Ralink WLAN cards. The following modules were not part of the distribution and will not be added in the future:

- `ati-fglrx`—ATI FireGL Graphics Cards
- `nvidia-gfx`—NVIDIA gfx driver
- `km_smartlink-softmodem`—Smart Link Soft Modem

8.3.3 Stricter tar Syntax

The `tar` usage syntax is stricter now. The `tar` options must come before the file or directory specifications. Appending options, like `--atime-preserve` or `--numeric-owner`, after the file or directory specification makes `tar` fail. Check your backup scripts. Commands such as the following no longer work:

```
tar czf etc.tar.gz /etc --atime-preserve
```

See the `tar` info pages for more information.

8.3.4 Kerberos for Network Authentication

`Kerberos` is the default for network authentication instead of `heimdal`. Converting an existing `heimdal` configuration automatically is not possible. During a system update, backup copies of configuration files are created as shown in [Table 8.1, “Backup Files”](#) (page 196).

Table 8.1 *Backup Files*

Old File	Backup File
<code>/etc/krb5.conf</code>	<code>/etc/krb5.conf.heimdal</code>
<code>/etc/krb5.keytab</code>	<code>/etc/krb5.keytab.heimdal</code>

The client configuration (`/etc/krb5.conf`) is very similar to the one of heimdal. If nothing special was configured, it is enough to replace the parameter `kpasswd_server` with `admin_server`.

It is not possible to copy the server-related (`kdc` and `kadmind`) data. After the system update, the old heimdal database is still available under `/var/heimdal`. MIT kerberos maintains the database under `/var/lib/kerberos/krb5kdc`. For more information, see [Chapter 40, Network Authentication—Kerberos](#) (page 725).

8.3.5 Hotplug Events Handled by the udev Daemon

Hotplug events are now completely handled by the udev daemon (`udev`). The event multiplexer system in `/etc/hotplug.d` and `/etc/dev.d` is no longer used. Instead, `udev` calls all hotplug helper tools directly according to its rules. Udev rules and helper tools are provided by `udev` and various other packages.

8.3.6 Firewall Activation During the Installation

To increase security, the enclosed firewall solution `SuSEFirewall2` is activated at the end of the installation in the proposal dialog. This means that all ports are closed initially and can be opened in the proposal dialog if necessary. By default, you cannot log in from remote systems. It also interferes with network browsing and multicast applications, such as SLP, Samba ("Network Neighborhood"), and some games. You can fine-tune the firewall settings using YaST.

If network access is required during the installation or configuration of a service, the respective YaST module opens the needed TCP and UDP ports of all internal and external interfaces. If this is not desired, close the ports in the YaST module or specify other detailed firewall settings.

8.3.7 KDE and IPv6 Support

By default, IPv6 support is not enabled for KDE. You can enable it using the `/etc/sysconfig` editor of YaST. The reason for disabling this feature is that IPv6 addresses are not properly supported by all Internet service providers and, as a consequence, this would lead to error messages while browsing the Web and delays while displaying Web pages.

8.3.8 Online Update and Delta Packages

Online Update now supports a special kind of RPM package that only stores the binary difference from a given base package. This technique significantly reduces the package size and download time at the expense of higher CPU load for reassembling the final package. See `/usr/share/doc/packages/deltarpm/README` for technical details.

8.3.9 Print System Configuration

At the end of the installation (proposal dialog), the ports needed for the print system must be open in the firewall configuration. Port 631/TCP and port 631/UDP are needed for CUPS and should not be closed for normal operation. Port 515/TCP (for the old LPD protocol) and the ports used by Samba must also be open for printing via LPD or SMB.

8.3.10 Change to X.Org

The change from XFree86 to X.Org is facilitated by compatibility links that enable access to important files and commands with the old names.

Table 8.2 *Commands*

XFree86	X.Org
XFree86	Xorg
xf86config	xorgconfig
xf86cfg	xorgcfg

Table 8.3 *Log Files in /var/log*

XFree86	X.Org
XFree86.0.log	Xorg.0.log
XFree86.0.log.old	Xorg.0.log.old

In the course of the change to X.Org, the packages were renamed from XFree86* to xorg-x11*.

8.3.11 X.Org Configuration File

The configuration tool SaX2 writes the X.Org configuration settings into `/etc/X11/xorg.conf`. During an installation from scratch, no compatibility link from `XF86Config` to `xorg.conf` is created.

8.3.12 XView and OpenLook Support Dropped

The packages `xview`, `xview-devel`, `xview-devel-examples`, `olvwm`, and `xtoolpl` were dropped. In the past, only the XView (OpenLook) base system was provided. The XView libraries are no longer provided after the system update. Even more important, OLVWM (OpenLook Virtual Window Manager) is no longer available.

8.3.13 Terminal Emulators for X11

A number of terminal emulators were removed because they are either no longer maintained or do not work in the default environment, especially by not supporting UTF-8. SUSE Linux Enterprise Server offers standard terminals, such as xterm, the KDE and GNOME terminals, and mlterm (Multilingual Terminal Emulator for X), which might be a replacement for aterm and eterm.

8.3.14 OpenOffice.org (OOo)

Directories

OOo is now installed in `/usr/lib/ooo-2.0` instead of `/opt/OpenOffice.org`. The default directory for user settings is now `~/.ooo-2.0` instead of `~/OpenOffice.org1.1`.

Wrapper

There are some new wrappers for starting the OOo components. The new names are shown in [Table 8.4, “Wrapper”](#) (page 199).

Table 8.4 *Wrapper*

Old	New
<code>/usr/X11R6/bin/OOo-calc</code>	<code>/usr/bin/ocalc</code>
<code>/usr/X11R6/bin/OOo-draw</code>	<code>/usr/bin/oodraw</code>
<code>/usr/X11R6/bin/OOo-impress</code>	<code>/usr/bin/ooimpress</code>
<code>/usr/X11R6/bin/OOo-math</code>	<code>/usr/bin/oomath</code>
<code>/usr/X11R6/bin/OOo-padmin</code>	<code>/usr/sbin/oopadmin</code>
<code>/usr/X11R6/bin/OOo-setup</code>	–
<code>/usr/X11R6/bin/OOo-template</code>	<code>/usr/bin/oofromtemplate</code>
<code>/usr/X11R6/bin/OOo-web</code>	<code>/usr/bin/ooweb</code>

Old	New
<code>/usr/X11R6/bin/OOo-writer</code>	<code>/usr/bin/oowriter</code>
<code>/usr/X11R6/bin/OOo</code>	<code>/usr/bin/ooffice</code>
<code>/usr/X11R6/bin/OOo-wrapper</code>	<code>/usr/bin/ooo-wrapper</code>

The wrapper now supports the option `--icons-set` for switching between KDE and GNOME icons. The following options are no longer supported:

`--default-configuration`, `--gui`, `--java-path`, `--skip-check`, `--lang` (the language is now determined by means of locales), `--messages-in-window`, and `--quiet`.

KDE and GNOME Support

KDE and GNOME extensions are available in the `OpenOffice_org-kde` and `OpenOffice_org-gnome` packages.

8.3.15 Sound Mixer `kmix`

The sound mixer `kmix` is preset as the default. For high-end hardware, there are other mixers, like `QAMix`, `KAMix`, `envy24control` (only ICE1712), or `hdspmixer` (only RME Hammerfall).

8.3.16 DVD Burning

In the past, a patch was applied to the `cdrecord` binary from the `cdrecord` package to support burning DVDs. Instead, a new binary `cdrecord-dvd` is installed that has this patch.

The `growisofs` program from the `dvd+rw-tools` package can now burn all DVD media (DVD+R, DVD-R, DVD+RW, DVD-RW, DVD+RL). Try using that one instead of the patched `cdrecord-dvd`.

8.3.17 Starting Manual Installation at the Kernel Prompt

The *Manual Installation* mode is gone from the boot loader screen. You can still get `linuxrc` into manual mode using `manual=1` at the boot prompt. Normally this is not necessary because you can set installation options at the kernel prompt directly, such as `textmode=1` or a URL as the installation source.

8.3.18 JFS: Not Supported Anymore

Due to technical problems with JFS, it is no longer supported. The kernel file system driver is still there, but YaST does not offer partitioning with JFS.

8.3.19 AIDE as a Tripwire Replacement

As an intrusion detection system, use AIDE (package name `aide`), which is released under the GPL. Tripwire is no longer available on SUSE Linux.

8.3.20 PAM Configuration

New Configuration Files (containing comments for more information)

`common-auth`

Default PAM configuration for auth section

`common-account`

Default PAM configuration for account section

`common-password`

Default PAM configuration for password changing

`common-session`

Default PAM configuration for session management

You should include these default configuration files from within your application-specific configuration file, because it is easier to modify and maintain one file instead of

the approximately forty files that used to exist on the system. If you install an application later, it inherits the already applied changes and the administrator is not required to re-member to adjust the configuration.

The changes are simple. If you have the following configuration file (which should be the default for most applications):

```
##PAM-1.0
auth      required      pam_unix2.so
account   required      pam_unix2.so
password  required      pam_pwcheck.so
password  required      pam_unix2.so      use_first_pass use_authtok
#password required      pam_make.so        /var/yp
session   required      pam_unix2.so
```

you can change it to:

```
##PAM-1.0
auth      include      common-auth
account   include      common-account
password  include      common-password
session   include      common-session
```

8.3.21 Becoming the Superuser Using `su`

By default, calling `su` to become `root` does not set the `PATH` for `root`. Either call `su -` to start a login shell with the complete environment for `root` or set `ALWAYS_SET_PATH` to `yes` in `/etc/default/su` if you want to change the default behavior of `su`.

8.3.22 Changes in the powersave Package

The configuration files in `/etc/sysconfig/powersave` have changed:

Table 8.5 *Split Configuration Files in `/etc/sysconfig/powersave`*

Old	Now Split Into
<code>/etc/sysconfig/powersave/ common</code>	<code>common</code>

Old	Now Split Into
	cpufreq
	events
	battery
	sleep
	thermal

`/etc/powersave.conf` has become obsolete. Existing variables have been moved to the files listed in [Table 8.5, “Split Configuration Files in `/etc/sysconfig/powersave`”](#) (page 202). If you changed the “event” variables in `/etc/powersave.conf`, these must now be adapted in `/etc/sysconfig/powersave/events`.

The names of sleep states have changed from:

- suspend (ACPI S4, APM suspend)
- standby (ACPI S3, APM standby)

To:

- suspend to disk (ACPI S4, APM suspend)
- suspend to ram (ACPI S3, APM suspend)
- standby (ACPI S1, APM standby)

8.3.23 Powersave Configuration Variables

Names of the powersave configuration variables are changed for consistency, but the sysconfig files are still the same. Find more information in [Section 27.5.1, “Configuring the powersave Package”](#) (page 535).

8.3.24 PCMCIA

`cardmgr` no longer manages PC cards. Instead, as with Cardbus cards and other subsystems, a kernel module manages them. All necessary actions are executed by `hotplug`. The `pcmcia` start script has been removed and `cardctl` is replaced by `pccardctl`. For more information, see `/usr/share/doc/packages/pcmciautils/README.SUSE`.

8.3.25 Setting Up D-BUS for Interprocess Communication in `.xinitrc`

Many applications now rely on D-BUS for interprocess communication (IPC). Calling `dbus-launch` starts `dbus-daemon`. The systemwide `/etc/X11/xinit/xinitrc` uses `dbus-launch` to start the window manager.

If you have a local `~/.xinitrc` file, you must change it accordingly. Otherwise applications like `f-spot`, `banshee`, `tomboy`, or `Network Manager banshee` might fail. Save your old `~/.xinitrc`. Then copy the new template file into your home directory with:

```
cp /etc/skel/.xinitrc.template ~/.xinitrc
```

Finally, add your customizations from the saved `.xinitrc`.

8.3.26 NTP-Related Files Renamed

For reasons of compatibility with LSB (Linux Standard Base), most configuration files and the init script were renamed from `xntp` to `ntp`. The new filenames are:

- `/etc/slp.reg.d/ntp.reg`
- `/etc/init.d/ntp`
- `/etc/logrotate.d/ntp`
- `/usr/sbin/rcntp`

- `/etc/sysconfig/ntp`

8.3.27 File System Change Notification for GNOME Applications

For proper functionality, GNOME applications depend on file system change notification support. For local-only file systems, install the `gamin` package (preferred) or run the `FAM` daemon. For remote file systems, run `FAM` on both the server and client and open the firewall for RPC calls by `FAM`.

GNOME (`gnome-vfs2` and `libgda`) contains a wrapper that picks `gamin` or `fam` to provide file system change notification:

- If the `FAM` daemon is not running, `gamin` is preferred (Rationale: `Inotify` is supported only by `gamin` and it is more efficient for local file systems).
- If the `FAM` daemon is running, `FAM` is preferred (Rationale: If `FAM` is running, you probably want remote notification, which is supported only by `FAM`).

8.3.28 Starting an FTP Server (`vsftpd`)

By default, `xinetd` no longer starts the `vsftpd` FTP server. It is now a stand-alone daemon and you must configure it with the YaST runtime editor.

8.3.29 Firefox 1.5: The URL Open Command

With Firefox 1.5, the method for applications to open a Firefox instance or window has changed. The new method was already partly available in former versions where the behavior was implemented in the wrapper script.

If your application does not use `mozilla-xremote-client` or `firefox-remote`, you do not need to change anything. Otherwise the new command to open a URL is `firefox url` and it does not matter whether Firefox is already running or not. If it is already running, it follows the preference configured in *Open links from other applications in*.

From the command line, you can influence the behavior by using `firefox -new-window url` or `firefox -new-tab url`.

Part II. Administration

GNOME Configuration for Administrators

9

This chapter discusses the following topics:

- [Section 9.1, “Using GConf for Defaults”](#) (page 210)
- [Section 9.2, “Customizing Menus”](#) (page 233)
- [Section 9.3, “Installing Themes”](#) (page 237)
- [Section 9.4, “Configuring Fonts”](#) (page 243)
- [Section 9.5, “MIME Types”](#) (page 244)
- [Section 9.6, “Setting Screensavers”](#) (page 246)
- [Section 9.7, “Session Management”](#) (page 248)
- [Section 9.8, “Improving Performance”](#) (page 249)
- [Section 9.9, “Hidden Directories”](#) (page 258)
- [Section 9.10, “Security Note on Configuring SMB Printers”](#) (page 260)
- [Section 9.11, “Disabling GNOME Desktop Features”](#) (page 261)
- [Section 9.12, “Starting Applications Automatically”](#) (page 263)
- [Section 9.13, “Automounting and Managing Media Devices”](#) (page 264)
- [Section 9.14, “Changing Preferred Applications”](#) (page 264)

- [Section 9.15, “Managing Profiles Using Sabayon”](#) (page 264)
- [Section 9.16, “Adding Document Templates”](#) (page 269)

9.1 Using GConf for Defaults

GConf is a system for storing application preferences that simplifies the administration of user preferences. GConf lets system administrators do the following:

- Set mandatory values for particular preferences for all users. This controls whether users can update particular preferences.
- Set default values for particular preferences for all users.
- Use suggested values for preferences that are specified in definition files for the preferences.
- Read documentation on each preference.

GConf also notifies applications when a preference value changes, locally or across a network. Therefore, when you change a preference, all applications that use the preference are immediately updated.

GConf provides a preferences database, similar to a simple file system. The file system contains keys organized into a hierarchy. Each key is either a directory containing more keys or it has a value. For example, the `key/apps/metacity/general/titlebar_font` contains an integer value giving the size of the titlebar font for the Metacity window manager.

GConf has the following components:

- [GConf Repository](#) (page 211)
- [GConf Daemon](#) (page 216)
- [GConf Command Line Tool](#) (page 217)
- [Configuration Editor](#) (page 230)

9.1.1 GConf Repository

Each preference in the GConf repository is expressed as a key-value pair. A GConf preference key is an element in the repository that corresponds to an application preference. For example, the

`/apps/gnome-session/options/show_splash_screen` preference key corresponds to the Show Splash Screen on Login option in the Sessions preference tool. The GNOME Desktop user interface does not contain all of the preference keys in the GConf repository. For example, the Panel preference tool does not contain an option that corresponds to the `/apps/panel/global/tooltips_enabled` key.

The repository is structured like a simple hierarchical file system. The repository contains the following:

- Directories that correspond to applications that use the GConf repository. For example, the file system contains the directory `/apps/metacity`.
- Subdirectories that correspond to categories of preferences. For example, the file system contains the directory `/apps/metacity/general`.
- Special files that list the preference keys in the directory and contain information about the keys. For example, a file that contains information about the keys that relate to the HTTP proxy preferences is in the directory `/system/http_proxy`.
- A `/schemas` directory that contains files that describe all of the preference keys.

Preference keys typically have simple values such as strings, integers, or lists of strings and integers. The format of the preference key in the repository depends on the backend module that is used to read the repository. The following is an example of the `/desktop/gnome/interface/font_name` preference key when an Extensible Markup Language (XML) backend module is used to read the repository:

```
<entry name="font_name" mtime="1038323555" muser="user123" type="string">
<stringvalue>Sans 10</stringvalue>
</entry>
```

NOTE

When this guide refers to a preference key, the path to the key is added to the name of the key. For example, the `font_name` preference key in the

`/desktop/gnome/interface` subdirectory is referred to as `/desktop/gnome/interface/font_name`.

GConf Configuration Sources

The GConf repository contains a series of storage locations that are called configuration sources. The configuration sources are listed in the `/etc/opt/gnome/opt/gnome/gconf/gconf-version-number/path` GConf path file. Each user has a path file. The path file specifies the following information for each configuration source:

- Backend module to use to read the repository
- Permissions on the repository
- Location of the repository

The GConf path file also contains include instructions. By default, the contents of the GConf path file are as follows:

```
xml:readonly:/etc/opt/gnome/opt/gnome/gconf/gconf.xml.mandatory
include /etc/opt/gnome/opt/gnome/gconf/2/local-mandatory.path
include "$HOME/.gconf.path"
include /etc/opt/gnome/opt/gnome/gconf/2/local-defaults.path
xml:readwrite:$HOME/.gconf
xml:readonly:/etc/opt/gnome/opt/gnome/gconf/gconf.xml.defaults
```

When GConf searches for a preference value, it reads the configuration sources in the order specified in the path file. The following table describes the configuration sources in the path file:

Table 9.1 *Configuration Sources in the Path File*

Configuration Source	Description
Mandatory	The permissions on this configuration source are set to Read Only. Users cannot overwrite the values in this source, so the preferences in the source are mandatory.

Configuration Source	Description
User	This configuration source is stored in the <code>.gconf</code> directory in the home directory of the user. When the user sets a preference, the new preference information is added to this location. You can use the Configuration Editor to modify the user configuration source.
Default	This configuration source contains the default preference settings.

The sequence of the configuration sources in the path file ensures that mandatory preference settings override user preference settings. The sequence also ensures that user preference settings override default preference settings. That is, GConf applies preferences in the following order of priority:

1. Mandatory preferences
2. User-specified preferences
3. Default preferences

The include instructions in the GConf path file enable system administrators to specify other configuration sources:

Table 9.2 *Other Configuration Sources*

Included Configuration Source	Description
<code>/etc/opt/gnome/opt/gnome/gconf/2/local-mandatory.path</code>	Stores mandatory preference values for a particular system.
<code>\$(HOME)/.gconf.path</code>	Specifies the location of the configuration source in the <code>.gconf.path</code> file in the user's home directory.

Included Configuration Source	Description
<code>/etc/opt/gnome/opt/gnome/gconf/2/local-defaults.path</code>	Stores default preference values for a particular system.

GConf Schema

A GConf schema is a collective term for a GConf schema key and a GConf schema object. The following table describes schema keys and schema objects and their relationship to preference keys:

Table 9.3 *Schema Keys and Objects*

Item	Description
Preference key	An element in the GConf repository that corresponds to an application preference.
Schema key	A key that stores a schema object for a preference key.
Schema object	An element in a configuration source that contains information for a preference key, such as the following: <ul style="list-style-type: none"> • The name of the application that uses the preference key • The type of value required for the preference key (for example, integer, boolean, etc.) • A default value for the preference key • Brief documentation on the preference key

The following are examples of a preference key, a schema key, and a schema object:

Table 9.4 *Preference Key, Schema Key, and Schema Object Examples*

Preference key:	<code>/desktop/gnome/interface/font_name</code>
-----------------	---

Schema key: `/schemas/desktop/gnome/interface/font_name`

Schema object: `<schema>
<applyto>/desktop/gnome/interface/font_name</applyto>
<key>/schemas/desktop/gnome/interface/font_name</key>
<owner>gnome</owner> <type>string</type>
<default>Sans 10</default> <locale
name="C"> <short>Default font</short>
<long>Name of the default font used by
gtk+.</long> </locale>`

You can associate a schema key with a preference key. For example, the following `/desktop/gnome/interface/font_name` key includes a schema key:

```
<entry name="font_name" mtime="1034873859"  
schema="/schemas/desktop/gnome/interface/font_name"/>
```

When you associate a schema key with a preference key, the preference uses the suggested value that is specified in the schema object of the schema key. The suggested value is contained in the `<default>` element in the schema object. By default, all the preference keys in the default configuration source are associated with schema keys.

Typically, schemas are stored in the default configuration source.

GConf Schema Definition Files

Schemas are generated from schema definition files. A schema definition file defines the characteristics of all of the keys in a particular application. Schema definition files have a `.schemas` extension.

The schema definition files are included in the `/etc/opt/gnome/opt/gnome/gconf/schemas` directory. You can use the schema definition files to create a new configuration source.

Some schema definition files correspond closely to a part of the GNOME Desktop user interface. For example, `system_http_proxy.schemas` corresponds to the Network Proxy preference tool. Other schema definition files contain preference keys that are not present in the GNOME Desktop user interface. For example, the `/apps/panel/global/tooltips_enabled` key is not present.

Some parts of the GNOME Desktop user interface contain preferences that represent preference keys from more than one schema definition file. For example, the Keyboard Shortcuts preference tool contains preferences that represent keys from the `panel-global-config.schemas` and `metacity.schemas` files.

9.1.2 GConf Daemon

The GConf daemon is called `gconfd-2`. It notifies applications when a preference value changes. For example, you might choose to show only icons in toolbars in the Menus & Toolbars preference tool. When you select this option in the preference tool, the toolbars on all open applications are updated instantly. The daemon can operate locally or across a network.

An instance of the GConf daemon is started for each user. It does not have to deal with complex problems such as authentication and data security. When the daemon starts, it loads the GConf path file. The daemon also manages all access between applications and the configuration sources.

When an application requests the value of a preference key, the daemon searches the configuration sources as follows:

1. Search for the value of the preference key in each configuration source, in the order specified in the path file.
2. If the value is found, return the value.
3. If the value is not found, search for the schema key that corresponds to the preference key in each configuration source, in the order specified in the path file.
4. If the schema key is found, check the value of the schema key.
5. If the value of the schema key is a schema object, return the suggested value in the `<default>` element of the schema object.

The GConf daemon also caches preference key values. All applications use this cache, so applications need to access the configuration sources only once.

To terminate the GConf daemon, use the following command:

```
gconftool-2 --shutdown
```

9.1.3 GConf Command Line Tool

GConf includes a command line tool called `gconftool-2`. You can use `gconftool-2` to perform the following tasks:

- Set the values of keys
- Display the values of keys
- Install schemas from schema definition files when you install an application

For example, you would use the following command to display the values of all keys in the `/desktop/gnome` directory and subdirectories:

```
gconftool-2 --recursive-list /desktop/gnome
```

The following table lists some of the options that you can use with the `gconftool-2` command:

Table 9.5 *gconftool-2 Options*

Option	Function
<code>--all-dirs</code>	Lists all subdirectories in a directory that you specify.
<code>--all-entries</code>	Displays the values of all keys in a directory that you specify.
<code>--config-source= configuration-source</code>	Use with the <code>--direct</code> option to specify a configuration source to use. If you do not specify a configuration source with this option, the command runs on all configuration sources in the path file.
<code>--direct</code>	Use with the <code>--config-source</code> option to access a configuration source directly. When you use this option, GConf bypasses the server. Ensure that the GConf daemon, <code>gconfd-2</code> , is not running before you use this option.
<code>--dump</code>	Generates a list that contains all preference keys in a GConf repository directory that you specify. The list

Option	Function
	contains XML descriptions of all the keys, in a <code><gconfentryfile></code> element.
	For example, you can redirect the output from this option to generate a file that lists all keys that are related to your panel configuration. You can use the <code>--load</code> option with this file.
<code>--get</code>	Displays the value of a preference key that you specify. Also displays the values of the elements in the schema object for a schema key that you specify.
<code>--help</code>	Displays a help message about the <code>gconftool-2</code> command and the options that you can use with it.
<code>--load=<i>filename</i></code>	Sets the values of preference keys in the current directory in a configuration source to the values in the file that you specify. The file that you specify must contain XML descriptions of the keys, in a <code><gconfentryfile></code> element.
<code>--long-desc=<i>description</i></code>	Use with the <code>--set-schema</code> option to specify a long description for a schema key.
<code>--makefile-install-rule</code>	Installs schema definition files to applications.
<code>--owner=<i>owner</i></code>	Use with the <code>--set-schema</code> option to specify an owner for a schema key.
<code>--recursive-list</code>	Displays the values of all preference keys in all subdirectories in a directory that you specify.
<code>--recursive-unset</code>	Resets the values of all preference keys, in all subdirectories in a directory, from the user setting to the setting in the default configuration source.

Option	Function
<code>--set</code>	<p>Sets the value of a preference key and writes the value to the user configuration source. Use it with the <code>--type</code> option to specify the data type of the value that you want to set. For example, the following command sets the value of the <code>/apps/gnome-terminal/profiles/Default/background_color</code> key in the user configuration source:</p> <pre data-bbox="475 480 1095 548">gconftool-2 --set "/apps/gnome-terminal/profiles/Default/background_color" --type string "#000000"</pre> <p>You can also use it with the <code>--direct</code> option and the <code>--config-source</code> option to write a value to another configuration source.</p>
<code>--set-schema</code>	<p>Sets the value of an attribute in a schema key and writes the value to the default configuration source.</p> <p>Use it with the following options to specify the attribute that you want to update:</p> <ul data-bbox="495 911 723 1117" style="list-style-type: none"> • <code>--type</code> • <code>--short-desc</code> • <code>--long-desc</code> • <code>--owner</code> <p>For example, the following command sets the short description in the schema key for the <code>/schemas/apps/gnome-terminal/profiles/Default/background_color</code> key:</p> <pre data-bbox="475 1305 1095 1398">gconftool-2 --set-schema "/schemas/apps/gnome-terminal/profiles/Default/background_color" --short-desc "Default background color of terminal"</pre>

Option	Function
<code>--short-desc=description</code>	Use with the <code>--set-schema</code> option to specify a short description for a schema key.
<code>--shutdown</code>	Terminates the GConf daemon.
<code>--type=data-type</code>	Specifies the data type when you set a value of a preference key. You can also use this option when you set the value of an attribute in a schema key. The following are valid data types: <ul style="list-style-type: none"> • <code>bool</code> • <code>float</code> • <code>int</code> • <code>list</code> • <code>pair</code> • <code>string</code>
<code>--unset</code>	Resets the value of a preference key from the user setting to the setting in the default configuration source.
<code>--usage</code>	Displays a brief help message about the <code>gconftool-2</code> command and the options that you can use with it.

9.1.4 Setting Preference Values

You can set a mandatory value or a default value for a preference key. Before you change mandatory preference values or default preference values for users, you must ensure that the GConf daemon is not running for any user.

IMPORTANT

Before you change mandatory preference values or default preference values for users, you must ensure that all users are logged out.

To set a mandatory value or a default value for a preference key, use the `gconftool-2` command, as follows:

```
gconftool-2 --direct --config-source configuration-source --type data-type
--set preference-keyvalue
```

For example, to set `wwwproxy.xyz.com` as the mandatory HTTP proxy host, use the following command:

```
gconftool-2 --direct --config-source
xml:readwrite:/etc/opt/gnome/opt/gnome/gconf/gconf.xml.mandatory --type string
--set /system/http_proxy/host wwwproxy.xyz.com
```

The user *cannot* override this preference value.

You can also use the `gconftool-2` command to set default values. For example, to set the default number of workspaces to five, use the following command:

```
gconftool-2 --direct --config-source
xml:readwrite:/etc/opt/gnome/opt/gnome/gconf/gconf.xml.defaults --type int
--set /apps/metacity/general/num_workspaces 5
```

The user *can* override this preference value.

9.1.5 Setting General Preferences

The following sections describe how to assign mandatory or default values to general preferences:

- [Setting HTTP Proxy Preferences](#) (page 222)
- [Setting Print Manager Preferences](#) (page 222)
- [Setting the Number of Workspaces](#) (page 222)
- [Setting Keyboard Accessibility Preferences](#) (page 223)
- [Setting Keyboard Shortcut Preferences](#) (page 223)

- [Setting Keyboard Shortcut Preferences](#) (page 223)

Setting HTTP Proxy Preferences

To set HTTP proxy preferences, modify the values of the preference keys in the `/system/http_proxy/` location. For example, to set a mandatory value for the HTTP proxy host, use the following command:

```
gconftool-2 --direct --config-source
xml:readwrite:/etc/opt/gnome/opt/gnome/gconf/gconf.xml.mandatory --type string
--set /system/http_proxy/host proxy-name
```

To set a default value for the HTTP proxy host, use the following command:

```
gconftool-2 --direct --config-source
xml:readwrite:/etc/opt/gnome/opt/gnome/gconf/gconf.xml.defaults --type string
--set /system/http_proxy/host proxy-name
```

You can also set other HTTP proxy-related preferences. For more information, see the `system_http_proxy.schemas` schema definition file.

Setting Print Manager Preferences

To set print manager preferences, modify the values of the preference keys in the `/apps/gnome-print-manager` location. For example, if you do not want users to view the print jobs of other users, set a mandatory value as follows:

```
gconftool-2 --direct --config-source
xml:readwrite:/etc/opt/gnome/gconf/gconf.xml.mandatory --type bool --set
/apps/gnome-print-manager/show_all_jobs false
```

To set a default value for this preference, use the following command:

```
gconftool-2 --direct --config-source
xml:readwrite:/etc/opt/gnome/gconf/gconf.xml.defaults --type bool --set
/apps/gnome-print-manager/show_all_jobs false
```

You can also set other print manager preferences. For more information, see the `gnome-print-manager.schemas` schema definition file.

Setting the Number of Workspaces

To set a mandatory number of workspaces, use the following command:

```
gconftool-2 --direct --config-source
xml:readwrite:/etc/opt/gnome/gconf/gconf.xml.mandatory --type int --set
/apps/metacity/general/num_workspaces integer
```

To set a default number of workspaces, use the following command:

```
gconftool-2 --direct --config-source
xml:readwrite:/etc/opt/gnome/gconf/gconf.xml.defaults --type int --set
/apps/metacity/general/num_workspaces integer
```

You can also set other window manager preferences. For more information, see the `metacity.schemas` schema definition file.

Setting Keyboard Accessibility Preferences

To set keyboard accessibility preferences, modify the values of the preference keys in `/desktop/gnome/accessibility/keyboard` location. For example, if you want to set a mandatory value so that keyboard accessibility features are enabled, use the following command:

```
gconftool-2 --direct --config-source
xml:readwrite:/etc/opt/gnome/gconf/gconf.xml.mandatory --type bool --set
/desktop/gnome/ accessibility/keyboard/enable true
```

To set a default value for this preference, use the following command:

```
gconftool-2 --direct --config-source
xml:readwrite:/etc/opt/gnome/gconf/gconf.xml.defaults --type bool --set
/desktop/gnome/
accessibility/keyboard/enable false
```

You can also set other keyboard accessibility preferences. For more information, see the `desktop_gnome_accessibility_keyboard.schemas` schema definition file.

Setting Keyboard Shortcut Preferences

To set keyboard shortcut preferences, modify the values of preference keys in `/apps/metacity/global_keybindings` location. For example, you might want users to use only the `Alt+F3` keyboard shortcut to open the Run Application dialog. To set this mandatory value, use the following command:

```
gconftool-2 --direct --config-source
xml:readwrite:/etc/opt/gnome/gconf/gconf.xml.mandatory --type string --set
/apps/metacity/global_keybindings '<Alt>F3'
```

You can also set other keyboard shortcut preferences. For more information, see the `metacity.schemas` schema definition file.

Setting Panel and Panel Object Preferences

The `panel-default-setup.entries` file specifies the following details of the panels in the GNOME Desktop:

- Number of panels
- Types of the panels
- Properties of the panels
- Contents of the panels

The configuration of individual panels and of panel objects is a complex task. You must first understand the structure of the `panel-default-setup.entries` file. For more information, see [Section “Specifying Individual Panels and Panel Objects”](#) (page 224).

To set preferences for individual panels and panel objects, you must set the values of many preferences in a configuration source. The easiest way to do this is to use the `gconftool-2` command with the `--dump` and `--load` options. For more information, see [Section “Setting Preferences for Individual Panels and Panel Objects”](#) (page 227).

Specifying Individual Panels and Panel Objects

The `panel-default-setup.entries` file contains sections that specify panels and panel contents, and it specifies values for schema keys. This file is located in the `/etc/opt/gnome/gconf/schemas` directory.

The `panel-default-setup.entries` file is structured as follows:

1. Keys that specify the general structure of panels, applets, and other panel objects in the GNOME Desktop.

The following keys specify the number of panels, panel objects, and applets that appear in the GNOME Desktop:

- `/apps/panel/default_setup/general/toplevel_id_list`

- /apps/panel/default_setup/general/object_id_list
- /apps/panel/default_setup/general/applet_id_list

The keys also assign identifiers to each panel, panel object, and applet. For example, the following sample from `panel-default-setup.entries` specifies that one panel appears in the GNOME Desktop:

```
<entry>
  <key>toplevel_id_list</key>
  <schema_key>/schemas/apps/panel/general/toplevel_id_list
  </schema_key>
  <value>
    <list type="string">
      <value>
        <string>bottom_panel</string>
      </value>
    </list>
  </value>
</entry>
```

In the `panel-default-setup.entries` file, the identifier `bottom_panel` identifies the bottom edge panel.

2. Keys that specify the properties of the panels.

The panel property keys are structured as follows:

```
/apps/panel/default_setup/toplevels/panel-name/panel-property-key
```

For example, the

```
/apps/panel/default_setup/toplevels/bottom_panel/size
```

key specifies the size of the bottom panel.

3. Keys that specify the panel objects, the panel object properties, and the panels where the objects reside.

For example, the following sample from `panel-default-setup.entries` specifies a Main Menu object at the left side of the bottom panel:

```
<entrylist base="/apps/panel/default_setup/objects/main_menu">
  <entry>
    <key>object_type</key>
    <schema_key>/schemas/apps/panel/objects/object_type</schema_key>
    <value>
      <string>menu-object</string>
    </value>
  </entry>
```

```

<entry>
  <key>toplevel_id</key>
  <schema_key>/schemas/apps/panel/objects/toplevel_id</schema_key>
  <value>
    <string>bottom_panel</string>
  </value>
</entry>
<entry>
  <key>position</key>
  <schema_key>/schemas/apps/panel/objects/position</schema_key>
  <value>
    <int>0</int>
  </value>
</entry>
.
.
.
</entrylist>

```

4. Keys that specify the applets, the applet preferences, and the panels where the applets reside.

For example, the following sample from `panel-default-setup.entries` specifies the Window List applet, in the bottom panel:

```

<entrylist base="/apps/panel/default_setup/applets/window_list">
  <entry>
    <key>object_type</key>
    <schema_key>/schemas/apps/panel/objects/object_type
    </schema_key>
    <value>
      <string>bonobo-applet</string>
    </value>
  </entry>
  <entry>
    <key>toplevel_id</key>
    <schema_key>/schemas/apps/panel/objects/toplevel_id
    </schema_key>
    <value>
      <string>bottom_panel</string>
    </value>
  </entry>
  <entry>
    <key>position</key>
    <schema_key>/schemas/apps/panel/objects/position
    </schema_key>
    <value>
      <int>2</int>
    </value>
  </entry>
.

```

```

.
.
  <entry>
    <key>bonobo_iid</key>

<schema_key>/schemas/apps/panel/objects/bonobo_iid_type</schema_key>
  <value>
    <string>OAFIID:GNOME_WindowListApplet</string>
  </value>
</entry>
</entrylist>

```

The OAFIID is a unique identifier for an applet. To find the OAFIID for a particular applet, see the `.server` file for the applet in the `/usr/lib/bonobo/servers` directory. For example, the following excerpt from `GNOME_Wncklet_Factory.server` shows the OAFIID for the Window List applet:

```

<oaf_server iid="OAFIID:GNOME_WindowListApplet"
type="factory" location="OAFIID:GNOME_Wncklet_Factory">

```

Setting Preferences for Individual Panels and Panel Objects

- 1 Log in to a GNOME session, then configure the panels as required.
- 2 Use the `--dump` option with the `gconftool-2` command line tool to generate a file that contains an XML description of your panel configuration.

The `--dump` option generates a list that contains all preference keys in a GConf repository directory that you specify.

For example, the following command creates an XML description of the default panel configuration in a file called `my-panel-setup.entries`:

```
gconftool-2 --dump /apps/panel/profiles/default > my-panel-setup.entries
```

- 3 Open the `my-panel-setup.entries` file in a text editor, then modify the file as required.

For example, you might want to change the location of the desktop entry files. The following is an excerpt from a file generated with the `--dump` option:

```

<entry>
  <key>objects/object_16/launcher_location</key>
  <schema_key>/schemas/apps/panel/objects/launcher_location
</schema_key>
  <value>

```

```
<string>hadjaha-00adce02f7.desktop</string>
</value>
</entry>
```

In the above example, you might want to change the reference to `hadjaha-00adce02f7.desktop` to another desktop entry file that is available globally.

When you generate a panel configuration with the `--dump` option, the positions of the panel objects are absolute positions. You might want to change these positions to relative positions. The object at the extreme left of a panel has a position value of 0. The next object has a position value of 1, and so on. If you want object positions to be relative to the right side of the panel, set the value of the `right_stick` key to `True`.

- 4 Use the `--load` option with the `gconftool-2` command line tool to set the values of the default configuration source to the values in the `my-panel-setup.entries` file.

For example, the following command sets the values of the keys in the default configuration source to the values of the corresponding keys in `my-panel-setup.entries`:

```
gconftool-2 --direct --config-source
xml:readwrite:/etc/opt/gnome/gconf/gconf.xml.defaults --load
my-panel-setup.entries
```

9.1.6 Setting Look-and-Feel Preferences

The following sections describe how to assign mandatory or default values to look-and-feel preferences.

Setting Font Preferences

To set font preferences, modify the values of two preference keys. The following table shows the keys to modify and the part of the user interface that the keys correspond to:

Table 9.6 *Font Preference Keys*

GConf Location	User Interface Component
/desktop/gnome/interface/ font_name	Font preference tool, Application font option
/apps/nautilus/preferences/ desktop_font	Font preference tool, Desktop font option

For example, to set Sans 12 as the mandatory application font, use the following command:

```
gconftool-2 --direct --config-source  
xml:readwrite:/etc/opt/gnome/gconf/gconf.xml.mandatory --type string --set  
/desktop/gnome/interface/font_name "Sans 12"
```

To set Palatino 12 as the default desktop object font, use the following command:

```
gconftool-2 --direct --config-source  
xml:readwrite:/etc/opt/gnome/gconf/gconf.xml.defaults --type string --set  
/apps/nautilus/preferences/desktop_font "palatino 12"
```

Setting Background Preferences

To set preferences for the desktop background, modify the values of the preference keys in the `/desktop/gnome/background` location. For example, to set a mandatory image for the background, use the following command:

```
gconftool-2 --direct --config-source  
xml:readwrite:/etc/opt/gnome/gconf/gconf.xml.mandatory --type string --set  
/desktop/gnome/background/picture_filename filename.png
```

To set a default value for this preference, use the following command:

```
gconftool-2 --direct --config-source  
xml:readwrite:/etc/opt/gnome/gconf/gconf.xml.defaults --type string --set  
/desktop/gnome/background/picture_filename filename.png
```

You can also set other background preferences. For more information, see the `desktop_gnome_background.schemas` schema definition file.

Setting Splash Image Preferences

To set splash image preferences, modify the value of the preference keys in the `/apps/gnome-session/options/` location. For example, if you do not want users ever to see a splash image, set a mandatory value as follows:

```
gconftool-2 --direct --config-source
xml:readwrite:/etc/opt/gnome/gconf/gconf.xml.mandatory --type bool --set
/apps/gnome-session/options/show_splash_screen false
```

To set a default value for this preference, use the following command:

```
gconftool-2 --direct --config-source
xml:readwrite:/etc/opt/gnome/gconf/gconf.xml.defaults --type bool --set
/apps/gnome-session/options/show_splash_screen false
```

You can also set other splash image preferences. For more information, see the `gnome-session.schemas` schema definition file.

9.1.7 Restoring Default Preference Values

To restore the default preference values for a user, use the following command:

```
gconftool-2 --direct --config-source user-configuration-source
--recursive-unset
```

Replace *user-configuration-source* with the configuration source in the `.gconf` directory in the home directory of the user.

This command resets the values of all preference keys, in all subdirectories, from the user setting to the setting in the default configuration source.

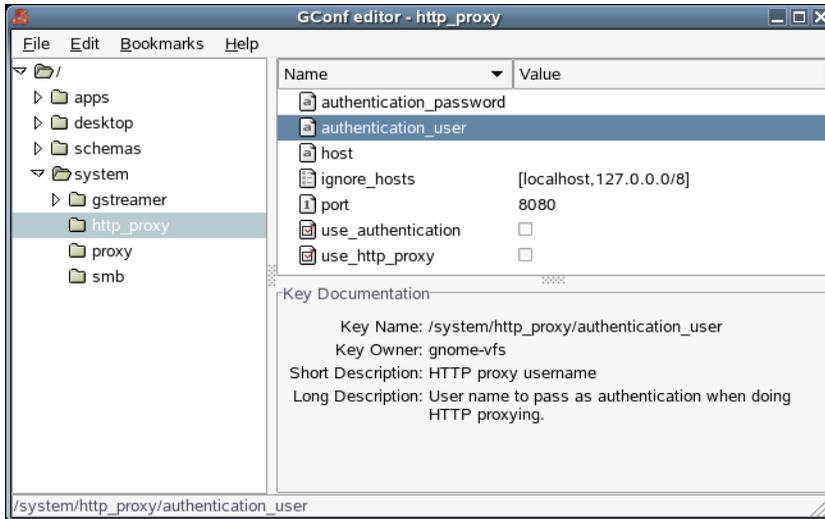
9.1.8 Configuration Editor

The Configuration Editor (GConf Editor) lets you view and edit the values of the keys stored in the GConf repository.

To open the Configuration Editor:

- 1 Press `Alt+F2` to open the Run Application dialog box.
- 2 Type `gconf-editor`, then click *Run*.

Figure 9.1 Configuration Editor Window



The Configuration Editor window contains the following panes:

Tree

Lets you navigate the directories and subdirectories in the GConf repository. Use this pane to display the keys that you want to modify in the modification pane. The tree pane is on the left side of the window.

Modification

Displays the keys in the selected GConf repository directory. Use this pane to select keys that you want to modify, and to modify the values of keys. The modification pane is in the upper part of the right side of the window.

The icons beside the keys in the modification pane indicate what type of value you can enter for the key. For example, the check mark icon beside the `/system/http_proxy/use_http_proxy` key indicates that you can enter a boolean value for the key.

The icons also indicate if you cannot edit the value of a key. For example, a key icon beside a schema key indicates that you cannot modify the value of the key.

Documentation

Displays documentation for the currently selected key. Use this pane to get more information about the GConf preference keys.

You can copy the names of keys so that you can paste them into another application. You can also add bookmarks to keys.

Modifying the Value of a Key

- 1 Use the tree pane to display the key that you want to modify in the modification pane.
- 2 Click the key you want to modify.
- 3 To change the value of an integer key or a string key, click the *Value* column of the key and then type the new value for the key.
- 4 To change the value of a boolean key, click the *Value* column of the key.

Copying a Key Name

- 1 Click the key whose name you want to copy in the modification pane.
- 2 Click *Edit > Copy Key Name*.
- 3 If required, you can paste the name of the key into another application.

Using Bookmarks with Keys

To access a key in your bookmarks, select the key from the Bookmarks menu.

Adding a Bookmark

- 1 Click the key that you want to bookmark in the modification pane.
- 2 Click *Bookmarks > Add Bookmark*.

Deleting a Bookmark

- 1 Click *Bookmarks > Edit Bookmarks*.

An Edit Bookmarks dialog is displayed.

- 2 Select a bookmark from the list on the left, then click *Delete*.
- 3 Click *Close*.

9.2 Customizing Menus

The way in which the GNOME Desktop implements menus enables you to do the following:

- Customize the menu hierarchy easily. The menu hierarchy is not based on the file system hierarchy. You can edit a small number of files to customize the menu hierarchy. You do not need to modify your applications or move files.
- Install applications easily. You do not need to provide information about the menu hierarchy to applications when you install the applications.
- Configure menus so that users cannot modify them.

Menus in the GNOME Desktop use the following components:

- [Desktop entry files](#)
- [Directory entry files](#)

9.2.1 Desktop Entry Files

A desktop entry file is a data file that provides information about an item in a menu. This file specifies the details for the item such as a name, a command to run, or an icon. It also contains keywords which determine the location of the item in the menu hierarchy. Desktop entry files have a `.desktop` file extension.

The following is a sample desktop entry file:

```

[Desktop Entry]
Encoding=UTF-8
Name=Calculator
Comment=Perform calculations
Exec=gcalctool
Icon=gcalctool.png
Terminal=false
Type=Application
Categories=GNOME;Application;Utility;
X-GNOME-DocPath=gcalctool/gcalctool.xml

```

The following table describes the most important keys in desktop entry files.

Table 9.7 *Desktop Entry File Keys*

Desktop Entry Key	Description
Encoding	Specifies the encoding of the desktop entry file.
Name	Specifies the name of the item. This name is displayed on the item in the menu.
Comment	Specifies a short description of the item. The comment is displayed as a tooltip when you point to the item in the menu.
Exec	Specifies a command to execute when you select the item from the menu.
Icon	Specifies the filename of an icon that represents the item. Does not specify the file extension or the path to the filename.
Terminal	Specifies whether the command in the Exec key runs in a terminal window. If the value is True, the command runs in a terminal window. If the command does not create a window in which to run, the value of this key must be True.
Type	Specifies the type of item. This value is one of the following:

Desktop Entry Key	Description
	<ul style="list-style-type: none"> • Application: Use this option for an item that starts an application. • Link: Use this option for an item that links to a file, folder, or FTP site.
Categories	<p>Specifies the keywords that describe the item. The keywords are separated with semicolons (;). To view a list of the standard category keywords, see the desktop menu specification at freedesktop.org [http://www.freedesktop.org]</p> <p>The vfolder information files map the keywords to menus.</p>
X-GNOME-Doc-Path	<p>Specifies the help file to display when you select Help on application-name from the menu item pop-up menu.</p>

For more information on the keys in desktop entry files, see the desktop entry specification at freedesktop.org [<http://www.freedesktop.org>].

NOTE

Panel launchers and desktop objects also use desktop entry files. These desktop entry files provide the same information as for items in a menu. For example, the desktop entry files provide the command to run when a user selects the launcher or object.

9.2.2 Directory Entry Files

A directory entry file is a data file that provides information about a menu. The directory entry file specifies the details for the menu, such as a name, a tooltip, and an icon. Directory entry files have a `.directory` file extension.

The following is a sample directory entry file:

```
[Desktop Entry]
Name=Accessories
Comment=Accessories menu
Icon=gnome-util.png
Type=Directory
```

The following table describes the most important keys in directory entry files.

Table 9.8 *Directory Entry File Keys*

Directory Entry Key	Description
Name	Specifies the name of the menu, which is displayed on the menu.
Comment	Specifies a short description of the menu. The comment is displayed as a tooltip when you point to the menu.
Icon	Specifies the filename of an icon that represents the menu. Does not specify the file extension or the path to the filename.
Type	Specifies the type of menu. The value of this key is always Directory.

9.2.3 Editing Menus

SUSE Link Enterprise Desktop uses the freedesktop.org menu specification. This specification uses the following files and directories:

Table 9.9 *Menu File Locations*

File	Description
/etc/xdg/menus/ applications.menu	This file contains the XML definition for the default application menu layout. If a user has their own applications.menu, it replaces the system wide menu.
/etc/xdg/menus/ applications-merged	This directory contains the default merge directories included in the <DefaultMergeDirs> element. You can add new <Menu> files in this location.

File	Description
<code>/etc/xdg/menus/preferences.menu</code>	This file contains the XML definition for the GNOME Control Center.

For more detailed information on adding and editing menu items, see the Desktop Menu Specification [<http://standards.freedesktop.org/menu-spec/latest>] Web site.

9.3 Installing Themes

A theme is a group of coordinated settings that specifies the visual appearance of a part of the GNOME Desktop. Users can select themes to change the appearance of the GNOME Desktop.

A theme contains settings that affect different parts of the GNOME Desktop, as follows:

Controls

Determines the visual appearance of windows, panels, and applets. It also determines the visual appearance of the GNOME-compliant interface items that appear on windows, panels, and applets (such as menus, icons, and buttons). Some of the controls setting options that are available are designed for special accessibility needs. Users can select an option for the controls setting from the Controls tabbed section in the Theme preference tool.

Window frame

Determines the appearance of the frames around windows only. Users can select an option for the window frame setting from the Window Border tabbed section in the Theme preference tool.

Icon

Determines the appearance of the icons on panels and the desktop background. Users can select an option for the icon setting from the Icons tabbed section in the Theme preference tool.

9.3.1 Theme Index File

Each theme has an index file which defines the characteristics of the theme. The name of the index file is `/opt/gnome/share/themes/theme-name/index.theme`.

The following is a sample theme index file:

```
[Desktop Entry]
Type=X-GNOME-Metatheme
Name=High Contrast Large
Name[es]=Alto contraste grande
Comment=Large black-on-white text and icons
Comment[es]=Textos e iconos grandes en negro sobre blanco
Encoding=UTF-8
[X-GNOME-Metatheme]
GtkTheme=HighContrastLargePrint
IconTheme=HighContrast
MetacityTheme=Atlanta
ApplicationFont=sans 18
```

The following table describes the keys in theme index files:

Table 9.10 *Theme Index File Keys*

Index File Key	Description
Type	Specifies that this theme determines the appearance of several theme options, such as controls, window frames, and icons.
Name	The name of the theme, which is displayed in the Theme preference tool.
Comment	A brief description of the theme, which is displayed under the name of the theme in the Theme preference tool.
GtkTheme	Corresponds to the controls setting in the Theme preference tool. Specifies which controls setting option to apply to windows, panels, and applets.
IconTheme	Corresponds to the icons setting in the Theme preference tool. Specifies which icons setting option to apply to panels and the desktop background.

Index File Key	Description
MetacityTheme	Corresponds to the window frame setting in the Theme preference tool. Specifies which window frame setting option to apply to windows.
ApplicationFont	Corresponds to the application font setting in the Font preference tool.

9.3.2 Installing a New Controls Option

You can add a new option for the controls setting in the Theme preference tool. Controls options reside in the `/opt/gnome/share/themes` directory. The typical structure of a controls option in the file system is as follows.

Option file

```
/opt/gnome/share/themes/option-name/gtk-2.0/gtkrc
```

Image files

```
/opt/gnome/share/themes/option-name/pixmaps/*.*
```

Typically, a new option for the controls setting is supplied as a `.tar.gz` file. To install the new controls option, unzip the `.tar.gz` file and then untar the `.tar` file into the `/opt/gnome/share/themes` directory.

Users can install their own options for the controls setting. If a user installs an option for the controls setting, the option is stored in the `$HOME/.themes` directory.

9.3.3 Installing a New Window Frame Option

You can add a new option for the window frame setting in the Theme preference tool. Window frame options reside in the `/opt/gnome/share/themes/option-name/metacity-1` directory. The typical structure of a window frame option in the file system is as follows.

Option file

```
/opt/gnome/share/themes/option-name/metacity-1/  
metacity-theme-1.xml
```

Image files

```
/opt/gnome/share/themes/option-name/metacity-1/*.*
```

Typically, a new option for the window frame setting is supplied as a `.tar.gz` file. To install the new window frame option, unzip the `.tar.gz` file and then untar the `.tar` file into the `/opt/gnome/share/themes` directory.

Users can install their own options for the window frame setting. If a user installs an option for the window frame setting, the option is stored in the `$HOME/.themes` directory.

9.3.4 Installing a New Icons Option

You can add a new option for the icons setting in the Theme preference tool. Icons options reside in the `/opt/gnome/share/icons/option-name` directory. The typical structure of an icons option in the file system is as follows.

Option file

```
/opt/gnome/share/icons/option-name
```

Image files

```
/opt/gnome/share/icons/option-name/icons/*.*
```

Typically, a new option for the icons setting is supplied as a `.tar.gz` file. To install the new icons option, unzip the `.tar.gz` file and then untar the `.tar` file into the `/opt/gnome/share/icons` directory.

Users can install their own options for the icons setting. If a user installs an option for the icons setting, the option is stored in the `$HOME/.icons/option-name` directory.

9.3.5 Installing Icons for Themes

The GNOME Desktop provides several themes that are designed for users with special visual needs. For example, some of the themes are designed for users with low vision. Several versions of icons might be required so that the icon can be displayed properly in each theme.

You might need to install a new icon for an application. When you install a new icon, you must create several versions of the icon so that the icon displays correctly in the themes. You must create several versions of the following types of icon:

- Icons that are used within applications in the GNOME Desktop
- Icons that are used internally by GTK+ applications or GTK+ stock icons

You can create the icons in several formats (for example, Portable Network Graphic (PNG) format). The suggested size of icons for the desktop environment is 48 x 48 pixels; at this size, most themes can rescale the icons.

When you install a new icon, create the following 48 x 48 pixel versions of the icon:

- Regular icon
- Low-contrast icon
- High-contrast icon
- Inverse high-contrast icon

If possible, also create 16 x 16 pixel versions of the each of the icons above, for themes that do not require large print.

Install the icons to the image files location that is specified for the theme in [Section 9.3.2, “Installing a New Controls Option”](#) (page 239) or [Section 9.3.3, “Installing a New Window Frame Option”](#) (page 239). For example, to add icons to the HighContrast-LargePrint theme, add the icons to the `/opt/gnome/share/themes/`

HighContrastLargePrint/pixmaps directory. Add references to the icons to the relevant theme files. For example, to add icons to the HighContrastLargePrint theme, add references to the icons to the `/opt/gnome/share/themes/HighContrastLargePrint/gtk-2.0/gtkrc` file. Modify the `gtkrc` file for the theme to associate the icon with a GTK stock icon identifier.

For more information on how to create icons for application launchers and panels, see the Icon Themes [<http://www.freedesktop.org/Standards/icon-theme-spec>].

9.3.6 Creating a Custom Controls Option

If the options for the controls setting are not suitable for the needs of your users, you can create a custom controls option.

- 1 Create a directory structure for the option in the `/opt/gnome/share/themes` directory.

Use the same directory structure that other options use. For example, to create an option that is called SmallPrint, create the following directories:

- `/opt/gnome/share/themes/SmallPrint`
- `/opt/gnome/share/themes/SmallPrint/gtk-2.0`

- 2 Locate the `gtkrc` file that is closest to meeting the needs of your users, then copy the file to the `gtk-2.0` directory of your new option.
- 3 Open the `gtkrc` file in a text editor and modify the attributes of the interface elements as needed.
- 4 (Conditional) If the new option includes images, install the images for the new option in the `pixmaps` directory of your new option.

If the new option uses images from another option, you do not need to create copies of the images for the new option. Instead, make sure that the reference to the images in the `pixmap_path` entry in the `gtkrc` file is correct.

Users can now select the new option for the controls setting.

For more information on gtkrc files, see the *GTK+ Reference Manual* [<http://developer.gnome.org/doc/API/2.0/gtk/index.html>].

9.4 Configuring Fonts

The GNOME Desktop uses the fontconfig font configuration and customization library. The fontconfig library can use all kinds of fonts, including PostScript Type 1 fonts and TrueType* fonts. The fontconfig library provides a list of all the fonts available on a system. To compile this list, fontconfig searches the directories listed in the `/etc/fonts/fonts.conf` file. To view all the fonts available on a system, access the `fonts:///` location in the file manager on the system.

For more information about the fontconfig library, see the Fontconfig [<http://freedesktop.org/software/fontconfig>] Web site.

9.4.1 Font Substitution

The fontconfig library performs font substitution when entire fonts or individual characters are not present. If the system needs to display a font that is not available, fontconfig attempts to display another, similar font. For example, if a Web page requests to display the Verdana font, and that font is not installed on the system, fontconfig displays a similar font, such as Helvetica. The list of similar fonts is defined in the `/etc/opt/gnome/fonts/fonts.conf` file.

If the system needs to display a character that is not present in the selected font, fontconfig attempts to display the character in another, similar font. For example, you might select Bitstream Vera Sans as the font for the Text Editor application. The Bitstream Vera font family does not include Cyrillic characters. If you open a document which contains a Cyrillic character, Text Editor uses a similar font that includes Cyrillic characters to display the character.

The fontconfig library also defines aliases for fonts (for example, serif, sans-serif, and monospace). When you select one of the aliases for a font, the system uses the first font that is defined for that alias in the `/etc/opt/gnome/fonts/fonts.conf`.

9.4.2 Adding a Font for All Users

- 1 Copy the font file to one of the directories in the `/etc/opt/gnome/fonts/fonts.conf` file.

Typically, fonts are stored in the `/opt/gnome/share/fonts/` directory.

- 2 (Conditional) The fontconfig library updates the list of fonts automatically. If the list of fonts is not updated, run the following command:

```
fc-cache directory-name
```

9.4.3 Adding a Font for an Individual User

- 1 Copy the font file to the `$HOME/.fonts` directory of the user.

If you drag the font file to the `fonts:///` location in the file manager, the font file is copied to the `$HOME/.fonts` directory.

- 2 (Conditional) The fontconfig library updates the list of fonts automatically. If the list of fonts is not updated, run the following command:

```
fc-cache directory-name
```

9.5 MIME Types

A Multipurpose Internet Mail Extension (MIME) type identifies the format of a file. The MIME type enables applications to read the file. Applications such as Internet browsers and e-mail applications use the MIME type to handle files of different types. For example, an e-mail application can use the MIME type to detect what type of file is attached to an e-mail.

The Nautilus file manager uses MIME types to identify the type of a file. The file manager needs to know the MIME type of a file to perform the following tasks:

- Open the file in an appropriate application
- Display a string that describes the type of file

- Display an appropriate icon to represent the file
- Display a list of other applications that can open the file

It is sometimes necessary to work out the correct MIME type for a file. This is usually done by examining the file's name or contents and looking up the correct MIME type in a database. If you add a new application (that is, extend the database), you must make sure that other applications can recognize the files associated with the application. For example, you might want to add the following:

- image/png files should be edited using the Gimp.
- image/png files should be described in English as Portable Network Graphics files.
- Files whose names end in .png should have the type image/png.

You can use a graphical editor (such as MIME-Editor [<http://rox.sourceforge.net/phpwiki/index.php/MIME-Editor>]) to edit the database, or you can do it manually by creating a file called `$XDG_DATA_HOME/mime/packages/Override.xml` in the format described below. For information on the XDG_ variables, see the Base Directory Specification [http://freedesktop.org/wiki/Standards_2fbasedir_2dspec].

When your new application is installed, it should install a file with the application's name in `$XDG_DATA_DIRS/mime/packages`. For example, running the `./configure && make install` command with the Gimp will create `/usr/local/share/mime/packages/gimp.xml`.

This file has the following format:

```
<?xml version="1.0" encoding="UTF-8"?>
<mime-info xmlns="http://www.freedesktop.org/standards/shared-mime-info">
  <mime-type type="image/png">
    <comment xml:lang="en">PNG image</comment>
    <comment xml:lang="af">png bleed</comment>
    ...
    <magic priority="50">
      <match type="string" value="\x89PNG" offset="0"/>
    </magic>
    <glob pattern="*.png"/>
  </mime-type>
</mime-info>
```

This provides a comment in two languages, a rule to recognize PNG files by their contents, and a rule to recognize PNG files by their names. You can provide information about several types in the single *application.xml* file. You do not need to provide any information which is also in the base package.

You can also add extra elements if they are namespaced to avoid conflicts. For example:

```
<desktop:can-edit-with>gimp.desktop</desktop:can-edit-with>
```

This indicates that the named desktop entry file describes an application that can edit image/png files.

Information added to the database should be static (for example, “The Gimp can edit PNG files.”), not configuration (for example, “The Gimp is the preferred editor for PNG files.”). For more information on storing configuration information, see the Shared Configuration System Spec [http://freedesktop.org/wiki/Standards_2fconfig_2dspec].

After you have installed the *application.xml* file, run the `update-mime-database` command to rebuild the output files. This program checks that the syntax of your file is correct and merges the information in it with the information in the other XML files in the `packages` directory. It then puts the rules for recognizing files into one set of files and the information about each type into other files (for example, `$XDG_DATA_DIR/mime/image/png.xml`) where other programs can easily access it.

When the application is uninstalled, the *application.xml* file is removed. Run `update-mime-database` again to remove the information from the database.

9.6 Setting Screensavers

A screensaver is an application that replaces the image on a screen when the screen is not in use. The screensaver application for the GNOME Desktop is XScreenSaver. The following sections describe how to set preferences for the XScreenSaver application and how to modify the displays that are available for the screensaver.

9.6.1 Setting Screensaver Preferences

Default screensaver preferences are stored in the XScreenSaver file, located in `/usr/X11R6/lib/X11/app-defaults/XScreenSaver`.

To modify screensaver application preferences, users can use the Screensaver preference tool. When a user modifies the screensaver preferences, the preferences are stored in the home directory of the user, in the `$HOME/.xscreensaver` file. For information on screensaver preferences, see the *GNOME Desktop User Guide* [<http://www.gnome.org/learn/users-guide/2.6>].

Users can also run the `/usr/X11R6/bin/xscreensaver-demo` command to open the XScreenSaver dialog.

To set default screensaver preferences for all users, modify the XScreenSaver file. You can also use the XScreenSaver dialog to create a `$HOME/.xscreensaver` file, then copy the file to the location of the XScreenSaver file.

To restore the default settings for a user, delete the `$HOME/.xscreensaver` file from the home directory of the user. If no `$HOME/.xscreensaver` file is present, the default preferences in the XScreenSaver file are used.

TIP

The default display behavior of XScreenSaver is to display a blank screen. The blank screen might confuse users. You might want to change this default display behavior.

To activate changes to the screensaver preferences, use the following command to reload screensaver preferences:

```
xscreensaver-command -restart
```

9.6.2 Modifying Screensaver Displays

The screensaver application allows users to select one or more screensaver displays. A screensaver display is an application that displays images on the screen of the user when the screen is not in use. The screensaver displays are listed in the XScreenSaver file and in the `$HOME/.xscreensaver` file.

To add a new screensaver display, copy the executable file for the display to the directory where the displays are located. Add the command for the screensaver display to the XScreenSaver file or the `$HOME/.xscreensaver` file. Include any arguments that are required to run the screensaver display on the whole screen, rather than in a window. For example, you might want to include the `-root` option to display the screensaver display on the whole screen.

To disable a screensaver display, add a Minus sign (-) to the start of the command for the screensaver display in the preferences file. The following excerpt from a `$HOME/.xscreensaver` file shows a disabled Qix (solid) screensaver display:

```
- "Qix (solid)"  qix -root -solid -segments 100
```

9.7 Session Management

A session occurs between the time that a user logs in to the GNOME Desktop and the time that the user logs out. The session manager starts after the login manager authenticates the user. The session manager lets the user manage the session. For example, a user can save the state of a session and return to that session the next time that he logs in.

At a minimum, the following applications run in a session:

- The session manager, `gnome-session`.
- The GConf X settings daemon, `gnome-settings-daemon`.
- The `gnome-panel` application, which runs the panels in the GNOME Desktop.
- The Metacity window manager.

The following table lists the files that contain default session information:

Table 9.11 *Default Session Information Files*

File	Description
<code>/opt/gnome/share/gnome/default.session</code>	Default session file. Default session details are stored in this file.

File	Description
<code>\$HOME/.gnome2/session</code>	User session file. When a user modifies the session, the details are stored in this file.

To set default session details for all users, modify the default session file.

To restore the default session settings for a user, delete the session file from the home directory of the user. If no user session file is present, the default settings in `/opt/gnome/share/gnome/default.session` are used.

To save the current session as the default session, users can run the `gnome-session-save` command.

GNOME also supports autostart. For more information, see [Section 9.12, “Starting Applications Automatically”](#) (page 263).

9.8 Improving Performance

This section discusses several preference settings you can change to improve the performance of the GNOME Desktop. You can use the `gconftool-2` command to set values for preferences for users. The example commands in this section show how to set values in the user configuration source.

You can also use the `--direct` and `--config-source` options to set mandatory values or default values for preferences. And you can use the `gconftool-2` command in a script to set the values of many preferences. For more information on the `gconftool-2` command and the options that are available with the command, see [Section 9.1, “Using GConf for Defaults”](#) (page 210).

9.8.1 Reducing CPU Usage

There are a number of preferences that you can set to reduce CPU usage by the GNOME Desktop.

Using Theme Options That Require Less CPU Resources

Some window frame theme options load image files to draw the window frame. Other options use simpler techniques to draw the window frame. The Crux window frame option loads image files, but can be slow on systems with limited CPU resources. To reduce CPU usage, use one of the following window frame options:

- Atlanta
- Esco

The following window frame options also use fewer CPU resources than Crux:

- AgingGorilla
- Bright
- Metabox

TIP

Metabox does not work well with inverse controls options such as HighContrastInverse. Use Atlanta with inverse controls options.

To change the window frame theme option, use the following command:

```
gconftool-2 --type string --set /apps/metacity/general/theme option-name
```

For example, to use Atlanta, run the following command:

```
gconftool-2 --type string --set /apps/metacity/general/theme Atlanta
```

Users can also use the Theme preference tool to select the appropriate option.

You can use the Metacity Theme Viewer to measure the performance of a window frame option and to preview the option. To start Metacity Theme Viewer, use the following command:

```
metacity-theme-viewer option-name
```

For example, to measure the performance of Atlanta and preview Atlanta, use the following command:

Turning Off Display of Icons in Menus

Some items in menus display an icon beside the item. To turn off this feature, use the following command:

```
gconftool-2 --type bool --set /desktop/gnome/interface/menus_have_icons false
```

Users can also use the Menus & Toolbars preference tool to deselect the Show Icons in Menus option.

Turning Off the Splash Screen

When users log in to the desktop environment, a splash screen is displayed by default. Icons are displayed on the splash screen while the user logs in. You can turn off the splash screen to reduce CPU usage during login.

To turn off the splash screen, use the following command:

```
gconftool-2 --type bool --set /apps/gnome-session/options/show_splash_screen false
```

Turning Off Panel Animation

When users show or hide panels, the panels can show or hide in an animated style. To turn off panel animation, use the following command:

```
gconftool-2 --type bool --set /apps/panel/global/enable_animations false
```

Users can also use the Panel preference tool to deselect the Drawer and Panel Animation option.

Improving File Manager Performance

The Nautilus file manager includes some features that you can modify to improve performance.

Modifying Performance Preferences

The file manager includes performance-related preferences. Each of these preference can take any of the following three values.

Table 9.12 *Performance-related Preferences*

Value	Description
<code>always</code>	Performs the action for both local files and files on other file systems.
<code>local_only</code>	Performs the action for local files only. Using this value reduces CPU usage.
<code>never</code>	Never performs the action. Using this value reduces CPU usage and network traffic.

The following table describes the performance preferences for the file manager. For the fastest performance, set the value of the preferences to Never.

Table 9.13 *File Manager Performance Preferences*

Preference	Description
<code>show_icon_text</code>	<p>Specifies when to preview the content of text files in the icon that represents the file. To never preview the content of text files, use the following command:</p> <pre>gconftool-2 --type string --set /apps/nautilus/preferences/show_icon_text never</pre>

Users can also perform the following steps:

1. Click *Edit > Preferences* in a file manager window, then click *Preview*.
2. Select an option for the Show Text in Icons preference.

Preference	Description
show_directory_item_counts	<p>Specifies when to show the number of items in folders. To never show the number of items in folders, use the following command:</p> <pre data-bbox="619 375 1022 444">gconftool-2 --type string --set /apps/nautilus/preferences/show_directory_item_counts never</pre> <p>Users can also perform the following steps:</p> <ol data-bbox="639 537 1124 695" style="list-style-type: none"> 1. Click <i>Edit > Preferences</i> in a file manager window, then click <i>Preview</i>. 2. Select an option for the Count Number of Items preference.
show_image_thumbnails	<p>Specifies when to show thumbnails of image files. To never show thumbnails, use the following command:</p> <pre data-bbox="619 878 1009 948">gconftool-2 --type string --set /apps/nautilus/preferences/show_image_thumbnails never</pre> <p>Users can also perform the following steps:</p> <ol data-bbox="639 1040 1124 1198" style="list-style-type: none"> 1. Click <i>Edit > Preferences</i> in a file manager window, then click <i>Preview</i>. 2. Select an option for the Show Thumbnails preference.
preview_sound	<p>Specifies when to preview the content of sound files. To never preview the content of sound files, use the following command:</p> <pre data-bbox="619 1382 1124 1451">gconftool-2 --type string --set /apps/nautilus/preferences/preview_sound never</pre>

Preference	Description
	Users can also perform the following steps: <ol style="list-style-type: none">1. Click <i>Edit > Preferences</i> in a file manager window, then click <i>Preview</i>.2. Select an option for the Preview Sound Files preference.

Turning Off the Side Pane, Toolbar, and Location Bar

The file manager includes preferences that let you turn off the side pane and the toolbar. Turning these off improves file manager performance.

To turn off the side pane, use the following command:

```
gconftool-2 --type bool --set /apps/nautilus/preferences/start_with_sidebar false
```

To turn off the toolbar, use the following command:

```
gconftool-2 --type bool --set /apps/nautilus/preferences/start_with_toolbar false
```

You can also turn off the location bar. Users can use the Ctrl+L keyboard shortcut to display a location bar when required.

To turn off the location bar, use the following command:

```
gconftool-2 --type bool --set /apps/nautilus/preferences/start_with_location_bar false
```

Turning Off the Desktop

The file manager contains a preference that lets users use Nautilus to manage the desktop. You can disable the desktop to improve performance. However, if you disable the desktop, you cannot

- Use the Desktop menu.
- Use the file manager to change the pattern or color of the desktop background.

- Use the desktop objects, such as Trash. (The desktop objects are not displayed on the desktop.)

To disable the desktop, use the following command:

```
gconftool-2 --type bool --set /apps/nautilus/preferences/show_desktop false
```

9.8.2 Reducing X Window System Network Traffic

There are some preferences that you can set to reduce X Window System network traffic on the GNOME Desktop.

Using Theme Options That Create Less Network Traffic

Remote display protocols do not transfer every pixel in a block of pixels if all pixels in the block are the same color. To reduce X Window System network traffic, use one of the following window frame options that uses solid colors:

- Atlanta
- Esco

For information on how to change theme options, see [Section “Using Theme Options That Require Less CPU Resources”](#) (page 250).

Turning Off Display of Icons in Menus

Some items in menus display an icon beside the item. This feature can increase X Window System network traffic if the icon is located on another file system or if the panels are displayed on a remote host.

For information on how to turn off this feature, see [Section “Turning Off Display of Icons in Menus”](#) (page 251).

9.8.3 Reducing Color Usage and Improving Display Quality

Many modern computer systems support 24-bit color (that is, 16,777,216 colors). However, many users still use systems that support only 8-bit color (256 colors). The GNOME Desktop uses the websafe color palette. This palette is a general-purpose palette of 216 colors which is designed to optimize the use of color on systems that support 8-bit color. However, some visual components of the GNOME Desktop are designed for systems that support 24-bit color.

The following display problems might occur on systems that support only 8-bit color:

- Windows, icons, and background images might appear grainy. Many themes, background images, and icons use colors that are not in the websafe color palette. The colors that are not in the palette are replaced with the nearest equivalent or a dithered approximation, which causes the grainy appearance.
- Applications that do not use the websafe color palette have fewer colors available; therefore, color errors might occur. Some colors might not appear in the user interface of the application, and some applications might crash if the application cannot allocate colors.
- Color flashing might occur when users switch between applications that use the websafe color palette and applications that do not use this palette. The applications that do not use the websafe color palette might use a custom colormap. When the custom colormap is used, other visual components might lose colors and then become unviewable.

The following sections describe how to optimize the appearance of the GNOME Desktop for systems that support only 8-bit color.

Using Theme Options That Use the Websafe Color Palette

Some window frame theme options use colors that are in the websafe color palette. Bright and Esco use colors from the websafe color palette and do not have the grainy appearance of other window frame options on 8-bit color displays. Use Bright or Esco for the best color display on 8-bit visual modes.

For information on how to change theme options, see [Section “Using Theme Options That Require Less CPU Resources”](#) (page 250).

Reducing Color Usage by Turning Off Display of Icons in Menus

Some items in menus display an icon beside the item. If the icon contains colors that are not in the websafe color palette, this feature can increase the number of colors used.

For information on how to turn off this feature, see [Section “Turning Off Display of Icons in Menus”](#) (page 251).

Reducing Color Usage by Turning Off the Splash Screen

You can turn off the splash screen to make more colors available for the GNOME Desktop and for applications.

For information on how to turn off the splash screen, see [Section “Turning Off the Splash Screen”](#) (page 251).

Reducing Color Usage by Using a Solid Color for the Background

Use a solid color for the desktop background. This reduces the number of colors used by the GNOME Desktop.

To set a solid color for the background, use the following commands:

```
gconftool-2 --type string --set /desktop/gnome/background/picture_options none
gconftool-2 --type string --set /desktop/gnome/background/color_shading_type
  solid
gconftool-2 --type string --set /desktop/gnome/background/primary_color
  \#hexadecimal-color
```

Users can also use the Background preference tool to choose a solid color for the background.

9.9 Hidden Directories

The following table describes the hidden directories that the GNOME Desktop adds to the home directories of users. A hidden directory is a directory that has a name that begins with a period (.).

Table 9.14 *Hidden Directories Added to Users' Home Directories*

Directory	Description
<code>.esd_auth</code>	Contains the authentication cookie for the GNOME sound daemon, the Enlightened Sound Daemon (ESD).
<code>.gconf</code>	Contains the GConf configuration source for the user. When the user sets a preference, the new preference information is added to this location.
<code>.gconfd</code>	Contains the following GConf daemon details: <ul style="list-style-type: none">• Configuration information• Lock information for objects that are referenced by an Interoperable Object Reference (IOR)• State information for objects that are referenced by an IOR
<code>.gnome</code>	Contains user-specific application data that is not stored in the GConf repository. For example, this directory contains MIME type information and session information for the user.
<code>.gnome-desktop</code>	The Nautilus file manager contains a preference that enables users to use the file manager to manage the desktop. If this option is selected, this directory contains the following: <ul style="list-style-type: none">• Objects on the desktop (for example, the Home object, the Trash object, and other launchers). The objects ap-

Directory	Description
	<p>pear in the directory as desktop entry files. For example, the <code>starthere.desktop</code> file contains a link to the Start Here location.</p> <ul style="list-style-type: none"> • Removable media volumes that are mounted.
	<p>The file manager also contains a preference that enables users to use the home directory as the desktop directory, instead of <code>.gnome-desktop</code>. If a user selects this option, the contents of the home directory are displayed as desktop objects.</p>
<code>.gnome2</code>	<p>Contains user-specific application data that is not stored in the GConf repository, such as the following:</p> <ul style="list-style-type: none"> • Keyboard shortcut information • Window location information • Desktop entry files for panel launchers <p>This directory also contains user-specific menu data. If a user modifies menus, the details are stored here.</p>
<code>.gnome2-private</code>	(Ignore this directory. It currently has no function.)
<code>.metacity</code>	Contains session data for the Metacity window manager.
<code>.nautilus</code>	<p>Contains file manager data that is specific to the use, such as the following:</p> <ul style="list-style-type: none"> • Metadata for the directories with which the user works • Nautilus emblems that the user adds • Nautilus desktop images

Directory	Description
<code>.themes</code>	Contains controls theme options, window frame theme options, and icons theme options that the user adds. The user can add themes from the Theme preference tool.
<code>.thumbnails</code>	Contains image thumbnails for the user. The image thumbnails are used in the file manager. The file manager contains a preference that the user can select to stop generation of thumbnail images.
<code>.xscreensaver</code>	Contains screensaver configuration data and screensaver preference data.

9.10 Security Note on Configuring SMB Printers

Windows network shares are also referred to as Samba or SMB shares. When you configure a printer on an SMB share, you must enter a username and password for the print queue.

The username and password are stored as unencrypted text in the `/etc/opt/gnome/cups/printers.conf` file. This file has read-only permissions for users with root privileges, so any user with root privileges can read the username and password for the print queue.

To reduce the impact of possible security violations, make sure that the username and password required to access the print queue is used only for the print queue. This ensures that any possible security violation is restricted to unauthorized use of the print queue.

9.11 Disabling GNOME Desktop Features

The GNOME Desktop includes features you can use to restrict access to certain of its functions. These disable (or lockdown) features let you restrict the actions that users can perform on a computer. For example, you might want to prevent command line operations on a computer that is for public use at a trade show.

You disable feature by setting GConf keys (see [Section 9.1, “Using GConf for Defaults”](#) (page 210)). You can also use the Configuration Editor application to set GConf keys in a user configuration source (see [Section 9.1.8, “Configuration Editor”](#) (page 230)).

9.11.1 Disabling Lock Screen and Log Out

To disable the lock screen and log out functions, set the `/apps/panel/global/disable_lock_screen` key and the `/apps/panel/global/disable_log_out` key to True.

When you disable the lock screen and log out functions, the following items are removed from the panels:

- Lock Screen and Log Out user menu items from the Main Menu
- Lock and Log Out menu items from the Add to Panel > Actions menu

To open this menu, right-click a vacant space on a panel and then click *Add to Panel > Actions*.

- Lock Screen and Log Out user menu items from the Actions menu in the Menu Bar applet

Additionally, any Lock Screen buttons and Log Out buttons on panels are disabled.

9.11.2 Disabling Command Line Operations

To disable operations from a command line, set the `/desktop/gnome/lockdown/disable_command_line` key to True.

When you disable command line operations, the following changes occur in the user interface:

- The Run Application menu item is removed from the following menus:
 - Main Menu
 - Actions submenu in the Add to Panel menu
 - Actions menu in the Menu Bar applet
- Any Run buttons on panels are disabled

To disable command line operations, you must also remove menu items that start terminal applications. For example, you might want to remove menu items that contain the following commands:

- GNOME Terminal command (`/opt/gnome/bin/gnome-terminal`)
- `/usr/bin/xterm`
- `/usr/bin/setterm`

These items are removed from the following menus:

- Main Menu
- Add to Panel > Launcher From menu

To disable command line operations, you must also disable the Command Line applet. To disable this applet, add the applet to the `/apps/panel/global/disabled_applets` key. When you disable the Command Line applet, it is removed from the Main Menu and the Add to Panel > Utility menu.

9.11.3 Disabling Panel Configuration

To disable panel configuration, set the `/apps/panel/global/locked_down` key to True.

When you disable panel configuration, the following changes occur in the user interface:

- The following items are removed from the Panel and Drawer pop-up menus:
 - Add to Panel
 - Delete This Panel
 - Properties
 - New Panel
- The launcher popup menu is disabled.
- The following items are removed from the Applet pop-up menu:
 - Remove from Panel
 - Lock
 - Move
- The Main Menu pop-up menu is disabled.
- The Launcher drag feature is disabled so that users cannot drag launchers to or from panels.
- The Panel drag feature is disabled so that users cannot drag panels to new locations.

9.12 Starting Applications Automatically

To automatically start applications in GNOME, use one of the following methods:

- **To run applications for every user:** Put `.desktop` files in `/opt/gnome/share/autostart` or `/opt/gnome/share/gnome/autostart`.
- **To run applications for an individual user:** Put `.desktop` files in `~/config/autostart`.

To disable an application that starts automatically, add `X-Autostart-enabled=false` to the `.desktop` file.

9.13 Automounting and Managing Media Devices

The GNOME Volume Manager (`gnome-volume-manager`) monitors volume-related events and responds with a user-specified policy. You can use the GNOME Volume Manager to automatically mount hot-plugged drives and inserted removable media, automatically run programs, automatically play audio CDs and video DVDs, and automatically import photos from a digital camera.

GNOME Volume Manager is started automatically. To disable GNOME Volume Manager, add `X-Autostart-enabled=false` to the `/opt/gnome/share/gnome/autostart/gnome-volume-manager.desktop` file.

You can use the GConf Editor to configure GNOME Volume Manager settings. Open the GConf Editor by pressing `Alt+F2` to open the Run Application dialog box, typing `gconf-editor`, then clicking *Run*. GNOME Volume Manager is located under `/desktop/gnome/volume_manager`.

9.14 Changing Preferred Applications

To change users' preferred applications, edit `/etc/opt/gnome/gnome_defaults.conf`.

After editing the file, run `SuSEconfig --module gnome-vfs2`.

9.15 Managing Profiles Using Sabayon

Sabayon is a system administration tool you can use to create and apply desktop environment profiles. A profile is a collection of default settings and restrictions that can be applied to either individual users or groups of users. Sabayon lets you edit GConf defaults and mandatory keys using a graphical tool.

Profile definition is done through a graphical session similar to the one a user would be running, only inside a desktop window. You can change properties (such as the desktop background, toolbars, and available applets) in the usual way. Sabayon also detects changes to the default settings in most desktop applications.

Files or documents that are left in the simulated home directory or on the desktop are included in the finished profile. This includes many application-specific databases, such as Tomboy notes. Using this mechanism, it's easy to supply introductory notes or templates in a manner easily accessible to new users.

A user profile can inherit its settings from a parent profile, overriding or adding specific values. This enables hierarchical sets of settings. For example, you can define an Employee profile and derive Artist and Quality Assurance profiles from that.

In addition to providing defaults, Sabayon can also lock down settings. This makes the setting resistant to change by users. For instance, you can specify that the desktop background cannot be changed to something other than the default you provide. It prevents casual tampering with settings, potentially reducing the number of helpdesk calls, and enabling kiosk-like environments. However, it does not provide absolute security and should not be relied on for such.

Sabayon also provides a list of settings for applications and generic user interface elements that have built-in lockdown support, including Epiphany, OpenOffice.org, and the GNOME panel. For example, the panel can be set up to allow only specific applets to be added to it and prevent changing its location or size on the screen. Likewise, the Save menu items can be disabled across all applications that use it, preventing users from saving documents.

The profiles are transferable to other computers. They reside in `/etc/opt/gnome/desktop-profiles/`, and each profile is saved in a separate ZIP file.

9.15.1 Creating a Profile

Profiles are saved in ZIP files located in `/etc/opt/gnome/desktop-profiles`. Each profile you save is stored in a separate ZIP file as *name-of-the-profile*.zip. You can copy or move profiles to other computers.

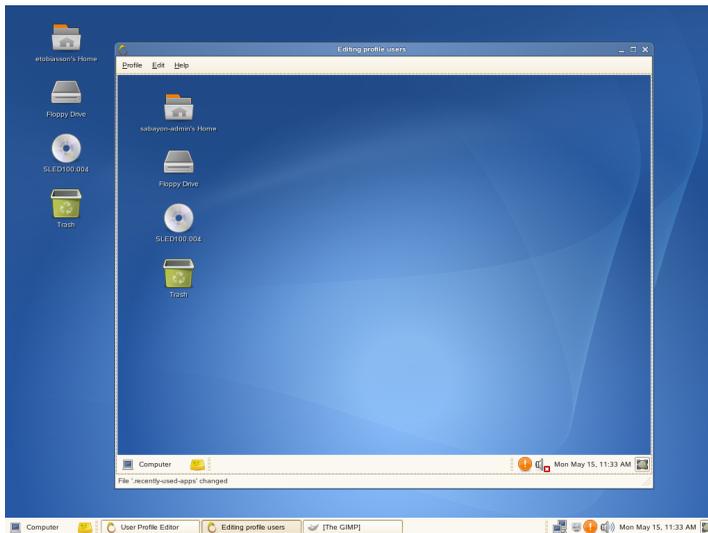
- 1 Click *Computer* → *More Applications* → *System* → *Desktop Profile Editor*.

- 2 If you are not logged in as root, type the root password and then click *Continue*.



- 3 Click *Add*.
- 4 Specify a name for the profile, then click *Add*.
- 5 Select the profile, then click *Edit*.

A new desktop session opens in an Xnest window.



- 6 In the Xnest window, make the changes to the settings that you want.

Each setting you change appears in the Xnest window.

You can choose to make each setting mandatory (click *Edit* → *Enforce Mandatory*), to ignore a setting (click *Edit* → *Changes* → *Ignore*), or make a setting default (don't selecting either *Ignore* or *Mandatory*).

- 7 To lock settings for users, click *Edit* → *Lockdown* in the Xnest window.

You can choose from the following options:

General: Lets you disable the command line, printing, print setup, and the save to disk feature.

Panel: Lets you lock down the panels, disable force quit, disable lock screen, disable log out, and disable any of the applets in the *Disabled Applets* list.

OpenOffice: Lets you define the macro security level for OpenOffice.org documents, load and save options, and user interface options.

Epiphany Web Browser: Lets you hide the menu bar, make the window fullscreen, and disable quit, arbitrary URLs, bookmark and toolbar editing, and unsafe protocols.

8 To save the profile, click *Profile* → *Save*.

The profile is saved in `/etc/opt/gnome/desktop-profiles`.

9 Click *Profile* → *Quit* to close the Xnest window, then click *Close* to exit Sabayon.

9.15.2 Applying a Profile

You can apply a profile to individual users or to all users on a workstation.

1 Click *Computer* → *More Applications* → *System* → *Desktop Profile Editor*.

2 If you are not logged in as root, type the root password and then click *Continue*.

3 Select the profile you want to apply, then click *Users*.



4 Select the users you want to use this profile.

To apply this profile to all users on this workstation, click *Use this profile for all users*.

5 Click *Close*.

9.16 Adding Document Templates

To add document templates for users, fill in the `Templates` directory in a user's home directory. You can do this manually for each user by copying the files into `~/Templates`, or system wide by adding a `Templates` directory with documents to `/etc/skel` before the user is created.

A user creates a new document from a template by right-clicking the desktop and selecting *Create Document*.

KDE Configuration for Administrators

10

KDE is an extensively configurable desktop environment. In addition to being configurable for the individual user, administrators have the possibility to create global configurations. This allows system administrators to provide custom default settings for their environments. Settings can differ between groups and individual users. It is also possible to restrict which settings users can change. Additionally, access to parts of KDE or functions in KDE can be restricted for users and groups.

These global configurations allow administrators to, for example, set up a company-wide desktop following the corporate identity that the user is not allowed to change. It is also feasible to assign task-specific profiles with access to only a limited set of applications to different groups within an organization.

KDE reads and stores all configuration files in fixed directory trees called profiles. A profile is a collection of default settings and restrictions that can be applied to individual users or groups of users. These profiles are handled by the KIOSK framework. Use the graphical KIOSK Admin Tool to generate and manage profiles or manually edit and create files and structures in a profile.

10.1 Managing Profiles Using the KIOSK Admin Tool

The Kiosk Admin Tool allows you to define profiles with desktop policies, environment restrictions, and menu definitions. It allows you to modify existing profiles and lets

you assign them to groups and users. Kiosk also lets you automatically deploy profiles to a remote host.

Start the Kiosk Admin Tool from the KDE main menu or with `[Alt] + [F2]` and the command `kiosktool`.

10.1.1 Creating a New Profile

To create a new profile, click *Add New Profile*. In the dialog that opens, enter a *Profile name* and a *Short description*. You can also specify an owner to which the files of the profile should belong. The user specified here must have write access to the profile directory. You also need to know the password of the user specified here. See [Section “Deploying Profiles to the Local Machine”](#) (page 273) for more information about the profile directory.

It is possible to change the data entered here any time with *Profile Properties*.

10.1.2 Setting Up a Profile

By choosing an existing profile and clicking *Setup Profile*, set up configurations for all KDE components, such as icons, menus, and file associations. After choosing a component, activate a restriction by checking the box of the respective entry. Choosing an entry with the mouse displays a help text explaining the effect the restriction has.

Entries either describe a specific feature that you can `disable` (such as *Disable Logout option*) or describe configuration options that you can `lock down` (such as *Lock down Screen Saver Settings*). By doing so, the feature or configuration option is not available when the profile is used.

Apart from disabling features and locking down configuration options, you can also configure the look and feel of the desktop itself. When selecting the components *Desktop Icon*, *Desktop Background*, *Screen Saver*, *KDE Menu*, and *Panel*, get two additional buttons—*Setup* and *Preview*. When clicking *Setup*, the desktop settings of the currently selected profile are loaded and temporarily overwrite your own desktop settings. Now you can make changes just as you would when configuring your own desktop. When you confirm your changes by clicking *Save*, the changes made are permanently added to the profile and your own desktop settings are restored.

10.1.3 Assigning Profiles to Users and Groups

When you create a profile, it is not “active” by default. First assign it to users or groups first. *Assign Profiles* opens a dialog where you can assign all existing profiles to distinct users or groups. If you are applying more than one profile to a user or group, settings from all profiles are used. If a profile contains settings that conflict with settings in another profile, the settings in the earlier listed profile take precedence. The same rule applies if you apply a profile to a specific user and another profile to a group of which this user is a member.

IMPORTANT: Users and Groups on Remote Hosts

You can assign profiles to groups and users available on the local machine. If you are planning to deploy your profiles to a remote server, make sure that the needed users and groups from the remote host are also available on the local machine (for example, by using NIS).

10.1.4 Deploying Profiles

The KIOSK Admin Tool not only allows you to deploy profiles to the local machine, but also to a remote computer. In doing so, you can, for instance, deploy the profiles onto an NFS server from which they are exported to all clients on the network.

Deploying Profiles to the Local Machine

If you are deploying your profiles to the same machine as the KIOSK Admin Tools is running on, no manual intervention is required—the tool takes care that the profiles are “found” on start-up. By default, all profiles are stored in `/var/lib/kde-profiles` to which only the user `root` is allowed to write. It is recommended not to change this setting.

However, if you need to change the location to which the profiles are written, select *Settings* → *Configure KIOSK Admin Tool* and change the *Base directory*.

It is also possible, although not recommended, to distribute profiles to different locations. Uncheck *Store all profiles under the same directory* in the configuration dialog. Having done so, you must specify the *Directory for this profile* when creating a profile.

Deploying Profiles to a Remote Machine

The KIOSK Admin Tool configuration (*Settings* → *Configure KIOSK Admin Tool*) lets you specify a location on a remote host to which to upload the profiles when exiting the tool. This upload mechanism uses the `fish` protocol. The *Server URL* field in the configuration dialog is initialized with `fish://root@host/`. Replace `root` with the user to which the files on the remote server should belong and `host` with the remote hostname. By default, the same directory as on the local host is used. To change this, click *Open file dialog* to specify a new directory on the remote server. After entering the password for the remote user, you can browse directories. By default, the directory on the local host is appended to the *Server URL* specified. Use *Strip off* to change this.

By default, KDE expects its profiles in `/var/lib/kde-profiles`. If you are deploying them to this directory on a remote machine or to a directory on an NFS server that will be mounted with this path by the clients, no further interaction is required. Otherwise, adjust `/etc/kde3rc`. See <http://websvn.kde.org/trunk/KDE/kdelibs/kdecore/README.kiosk?view=markup> for details.

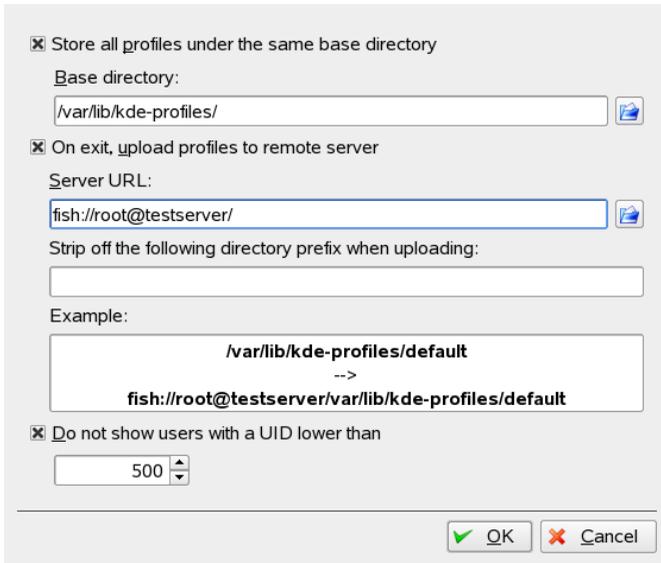
10.1.5 Example: Creating and Assigning a Profile

In the following example, a profile called `myCompany` is created and assigned to the user `tester` on the remote host `testserver`.

- 1 Start the Kiosk Admin Tool from the KDE main menu or with `[Alt] + [F2]` and the command `kiosktool`.
- 2 Open the configuration dialog with *Settings* → *Configure KIOSK*. On the local machine, all profiles are stored in `/var/lib/kde-profiles` by default. Also by default, users with a `UID` lower than 500 are not displayed.

The profile in this example should be deployed to a remote host named `testserver` in the default profile location. Therefore, activate *On exit* and change the *Server URL* to `fish://root@testserver/`.

Figure 10.1 *Configuring the KIOSK Admin Tool*



3 Open the *Add New Profile* dialog and create a new profile called myCompany.

Figure 10.2 *Adding a Profile*



Click *Finished* to save the new profile. You are prompted for the `root` password before the files can be saved.

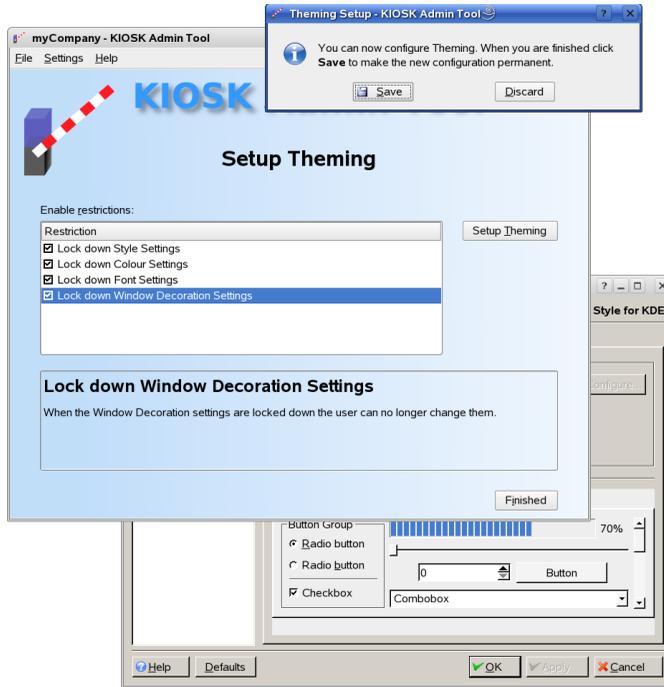
- 4 Clicking *Setup Profile* opens a dialog where you can configure the various aspects of KDE.

Figure 10.3 *Setting Up a Profile*



If you choose, for example, *Theming* then *Setup Theming*, the configuration dialog for the themes opens. All changes you make here do not affect your current desktop, but are added to the profile you are working on after you confirm your changes with *Save* in the *Theming Setup* window.

Figure 10.4 *Setting Up Theming*



After finishing setting up the profile, return to the main menu by clicking *Finished*.

- 5 Assign the profile to distinct users or groups by clicking *Assign Profiles*.

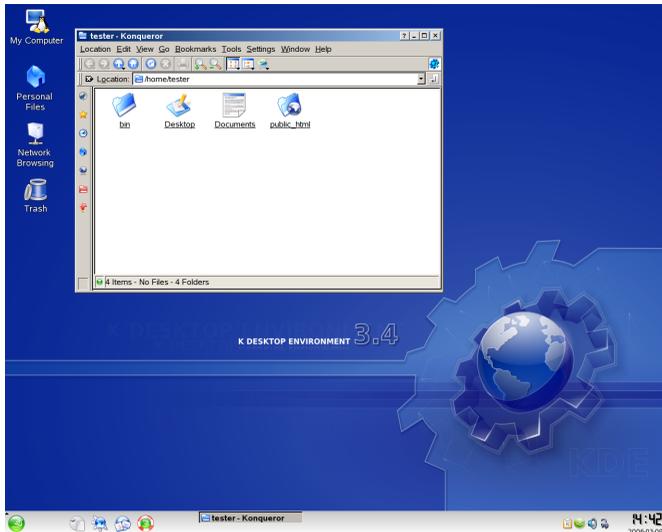
Figure 10.5 *Assigning Profiles*



Return to the main menu by clicking *Finished*.

- 6 Now the profile is available on the local machine. Before deploying it to the remote host, you can test it. Start a new session by right-clicking the desktop and choosing *Switch User* → *Start New Session* then log in as user `tester`.

Figure 10.6 *The Profile in Use*



Return to your own desktop by logging out as `tester`. If you need to make changes, start the setup procedure again. Otherwise leave the KIOSK Admin Tool. On exit, it deploys all profiles to `testserver`. You must enter the `root` password on `testserver` for this operation. Because the profiles are deployed to the default KDE profile location in this example, no further action is required. The next time `tester` logs in on `testserver`, the `myCompany` profile is used.

10.2 Managing Profiles Manually

If you prefer manually editing configuration files over using a graphical tool, the KIOSK framework lets you do this, too. Every configuration file in a profile is a plain text file that can be edited with the editor of your choice. KIOSK's configuration and deployment options are described in detail in The KDE Source Repository at <http://websvn.kde.org/trunk/KDE/kdelibs/kdecore/README.kiosk?view=markup>. Refer to this resource for details. In the following, only the fundamentals needed to use the KIOSK framework are described.

10.2.1 File System Hierarchy

KDE reads and stores files used by the KDE environment itself as well as by the KDE applications in fixed directory trees, also referred to as “profiles” in this context. By default, there are two such directories: `/opt/kde3` and `~/ .kde`. The `~/ .kde` directory contains the user-specific settings. The `/opt/kde3` directory contains data and configuration files that came with the packages. It is not recommended to make any changes there, because they get overwritten with the next update. Therefore, as a system administrator you can create additional trees that are used by the KIOSK framework. The default location for an additional fixed directory tree is `/var/lib/kde-profiles`. You can add custom locations in `/etc/kde3rc`. Refer to the KIOSK documentation for details.

A fixed directory tree consists of the following directories (although not all directories need to be present):

```
bin
    Executables

cgi-bin
    Help center scripts

lib
    Libraries

socket-<HOSTNAME>
    Communication sockets

tmp-<HOSTNAME>
    Temporary files

cache-<HOSTNAME>
    Cached data

share
    Application and configuration data
```

Among others, the `share` directory contains the following subdirectories:

share/applications
 .desktop files for all applications appearing in the KDE menu

share/applnk
 The KDE menu structure

share/config
 Configuration files for applications and components as well as the global configuration file `kdeglobals`

share/icons
 Icons, categorized by theme, dimension, and usage category

share/mimelnk
 .desktop files with mime types

share/wallpapers
 Images that can be used as background pictures

Precedence

KDE scans all directory trees known to the system. When a specific file is present in multiple directory trees, the order of precedence determines which file is used.

When configuration files are scanned, an additional rule applies. Generally, the contents of multiple configuration files with the same name are merged. However, if the same configuration key is defined more than once, the key from the file with the highest precedence determines which value is used.

The rule of precedence is:

1. User directory (`~/ .kde`)
2. Directories configured in `/etc/kde3rc`
3. Systemwide default directory (`/opt/kde3`)

As a user, you can overwrite this order by setting the variable `$KDEDIRS`. Directories should be separated by a colon (:). The first directory has highest precedence and the last one lowest precedence.

10.2.2 Configuration File Format

KDE configurations are stored in text files in UTF-8 format. Each configuration option consists of a key and value pair and is placed inside a group:

```
[Group 1]
key=value
key 2=value 2
```

White space at the beginning or end of keys and values are ignored. However, both may contain spaces as shown in the example above. If a value is supposed to start or end with space or should contain line breaks or special characters, use the following special codes:

- `\s`: space
- `\t`: tab
- `\r`: carriage return
- `\n`: new line
- `\\`: backslash

Shell Expansion

To use dynamically generated values, KDE allows you to use *shell expansions*. If a key is followed by `[$e]`, shell expansions are activated. When using this construct, the value is written to the file the first time it is read. Using `[$!e]`, lock down this behavior so the expansion is evaluated every time the configuration file is read. Shell expansions allow you to either use environment variables or the output of commands as values.

```
[example group]
UserName=$USER
Group=$(id -g)
HomeDirectory=$HOME
```

Localization

All configuration values can be localized with a language code added to the key entry:

```
[example group]
Label=Language
```

```
Label[de]=Sprache
Label[ru]=Язык
```

Configuration Entry Lock Down

All configuration entries can be protected from being overwritten. You can lock down entire configuration files, groups, or individual keys. Do this by adding [Si] on a separate line at the beginning of a file, placing it behind the group name, or adding it behind a key.

```
[example group][Si]
  Label=Language
```

```
[example group 2]
  UserName[$i]=$USER
```

10.2.3 Activating Profiles

Profiles can be created anywhere in the file system. To make the KDE environment read your profiles, you must make them known to the system in `/etc/kde3rc`. The default profile location `/var/lib/kde-profiles/` is already configured there.

By default, a custom profile is not associated to users or groups. You can make this association in the user profile map file at `/etc/kde-user-profile`. The only exception from this is the default profile. If you create a profile named “default” under `/var/lib/kde-profiles/` this is automatically associated to all users on this machine (such a profile does not exist by default).

Find more detailed information about activating profiles and mapping them to users in the KIOSK framework documentation.

10.2.4 Examples

SUSE Linux Enterprise comes with three predefined profiles (redmond, simplified, and Thinclient) located in `/var/lib/kde-profiles`. You may use one of these as a template for your own profile.

Active Directory Support

Active Directory* (AD) is a directory service based on LDAP, Kerberos, and other services that is used by Microsoft Windows to manage resources, services, and people. In an MS Windows network, AD provides information about these objects, restricts access to any of them, and enforces policies. SUSE® Linux Enterprise lets you join existing AD domains and integrate your Linux machine into a Windows environment.

11.1 Integrating Linux and AD Environments

With a Linux client configured as an Active Directory client that is joined to an existing Active Directory domain, benefit from various features not available on a pure SUSE Linux Enterprise Desktop Linux client:

Browsing Shared Files and Folders with SMB

Both Nautilus, the GNOME file manager, and Konqueror, its KDE counterpart, support browsing shared resources through SMB.

Sharing Files and Folders with SMB

Both Nautilus, the GNOME file manager, and Konqueror, its KDE counterpart, support sharing folders and files as in Windows.

Accessing and Manipulating User Data on the Windows Server

Through Nautilus and Konqueror, users are able to access their Windows user data and can edit, create, and delete files and folders on the Windows server. Users can access their data without having to enter their password again and again.

Offline Authentication

Users are able to log in and access their local data on the Linux machine even if they are offline (for example, using a laptop) or the AD server is unavailable for other reasons.

Windows Password Change

This part of AD support in Linux enforces corporate password policies stored in Active Directory. The display managers and console support password change messages and accept your input. You can even use the Linux `passwd` command to set Windows passwords.

Single-Sign-On through Kerberized Applications

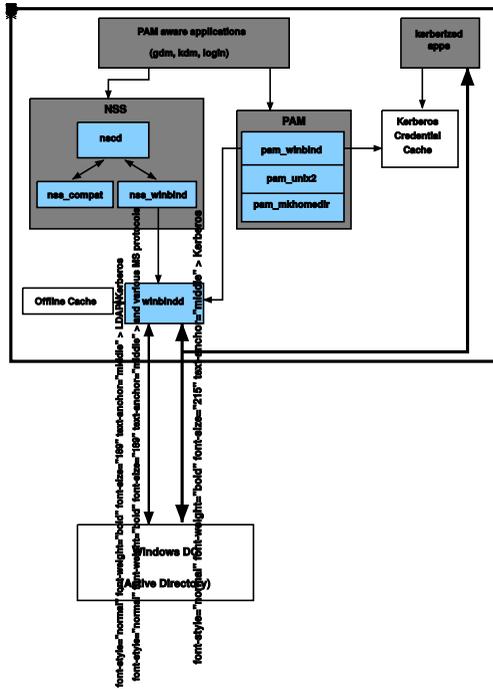
Many applications of both desktops are Kerberos-enabled (*kerberized*), which means they can transparently handle authentication for the user without the need for password reentry at Web servers, proxies, groupware applications, or other locations.

A brief technical background for most of these features is given in the following section. For directions for file and printer sharing, refer to *GNOME User Guide* and *KDE User Guide*, where you can learn more about AD enablement in the GNOME and KDE application worlds.

11.2 Background Information for Linux AD Support

Many system components need to interact flawlessly to integrate a Linux client into an existing Windows Active Directory domain. [Figure 11.1, “Active Directory Authentication Schema”](#) (page 287) highlights the most prominent ones. The following sections focus on the underlying processes of the key events in AD server and client interaction.

Figure 11.1 Active Directory Authentication Schema



To communicate with the directory service, the client needs to share at least two protocols with the server:

LDAP

LDAP is a protocol optimized for managing directory information. A Windows domain controller with AD can use the LDAP protocol to exchange directory information with the clients. To learn more about LDAP in general and about the open

source port of it, OpenLDAP, refer to [Chapter 34, LDAP—A Directory Service](#) (page 653).

Kerberos

Kerberos is a third-party trusted authentication service. All its clients trust Kerberos's judgment of another client's identity, enabling kerberized single-sign-on (SSO) solutions. Windows supports a Kerberos implementation, making Kerberos SSO possible even with Linux clients. To learn more about Kerberos in Linux, refer to [Chapter 40, Network Authentication—Kerberos](#) (page 725).

The following client components process account and authentication data:

Winbind

The most central part of this solution is the winbind daemon that is a part of the Samba project and handles all communication with the AD server.

NSS (*Name Service Switch*)

NSS routines provide name service information. Naming service for both users and groups is provided by `nss_winbind`. This module directly interacts with the winbind daemon.

PAM (*Pluggable Authentication Modules*)

User authentication for AD users is done by the `pam_winbind` module. The creation of user homes for the AD users on the Linux client is handled by `pam_mkhome`. The `pam_winbind` module directly interacts with winbindd. To learn more about PAM in general, refer to [Chapter 23, Authentication with PAM](#) (page 483).

Applications that are PAM-aware, like the login routines and the GNOME and KDE display managers, interact with the PAM and NSS layer to authenticate against the Windows server. Applications supporting Kerberos authentication, such as file managers, Web browsers, or e-mail clients, use the Kerberos credential cache to access user's Kerberos tickets, making them part of the SSO framework.

11.2.1 Domain Join

During domain join, the server and the client establish a secure relation. On the client, the following tasks need to be performed to join the existing LDAP and Kerberos SSO environment provided by the Window domain controller. The entire join process is

handled by the YaST Domain Membership module that can be run during installation or in the installed system:

- 1 The Windows domain controller providing both LDAP and KDC (Key Distribution Center) services is located.
- 2 A machine account for the joining client is created in the directory service.
- 3 An initial ticket granting ticket (TGT) is obtained for the client and stored in its local Kerberos credential cache. The client needs this TGT to get further tickets allowing it to contact other services, like contacting the directory server for LDAP queries.
- 4 NSS and PAM configurations are adjusted to enable the client to authenticate against the domain controller.

During client boot, the winbind daemon is started and retrieves the initial Kerberos ticket for the machine account. winbindd automatically refreshes the machine's ticket to keep it valid. To keep track of the current account policies, winbindd periodically queries the domain controller.

11.2.2 Domain Login and User Homes

The login managers of GNOME and KDE (GDM and KDM) have been extended to allow the handling of AD domain login. Users can choose to log in to the primary domain the machine has joined or to one of the trusted domains with which the domain controller of the primary domain has established a trust relationship.

User authentication is mediated by a number of PAM modules as described in [Section 11.2, “Background Information for Linux AD Support”](#) (page 286). The `pam_winbind` module used to authenticate clients against Active Directory or NT4 domains is fully aware of Windows error conditions that might prohibit a user's login. The Windows error codes are translated into appropriate user-readable error messages that PAM gives at login through any of the supported methods (GDM, KDM, console, and SSH):

```
Password has expired
```

The user sees a message stating that the password has expired and needs to be changed. The system prompts directly for a new password and informs the user if

the new password does not comply with corporate password policies, for example, the password is too short, too simple, or already in the history. If a user's password change fails, the reason is shown and a new password prompt is given.

Account disabled

The user sees an error message stating that his account has been disabled and that he should contact the system administrator.

Account locked out

The user sees an error message stating that his account has been locked and that he should contact the system administrator.

Password has to be changed

The user can log in but receives a warning that the password needs to be changed soon. This warning is sent three days before that password expires. After expiration, the user cannot login again.

Invalid workstation

When a user is just allowed to log in from specific workstations and the current SUSE Linux Enterprise machine is not in that list, a message appears that this user cannot log in from this workstation.

Invalid logon hours

When a user is only allowed to log in during working hours and tries to log in outside working hours, a message shows that login is not possible at this point in time.

Account expired

An administrator can set an expiration time for a specific user account. If that user tries to log in after that time has passed, the user gets a message that the account has expired and cannot be used to log in.

During a successful authentication, `pam_winbind` acquires a ticket granting ticket (TGT) from the Kerberos server of Active Directory and stores it in the user's credential cache. It also takes care of renewing the TGT in the background, not requiring any user interaction.

SUSE Linux Enterprise Desktop supports local home directories for AD users. If configured through YaST as described in [Section 11.3, “Configuring a Linux Client for Active Directory”](#) (page 291), user homes are created at the first login of a Windows (AD) user into the Linux client. These home directories look and feel entirely the same

as standard Linux user home directories and work independently of the AD domain controller. Using a local user home, it is possible to access a user's data on this machine, even when the AD server is disconnected, if the Linux client has been configured to perform offline authentication.

11.2.3 Offline Service and Policy Support

Users in a corporate environment must have the ability to become roaming users, for example, to switch networks or even work disconnected for some time. To enable users to log in to a disconnected machine, extensive caching was integrated into the winbind daemon. The winbind daemon enforces password policies even in the offline state. It tracks the number of failed login attempts and reacts according to the policies configured in Active Directory. Offline support is disabled by default and must be explicitly enabled in the YaST Domain Membership module.

As in Windows, when the domain controller has become unavailable, the user can still access network resources (other than the AD server itself) with valid Kerberos tickets that have been acquired before losing the connection. Password changes cannot be processed unless the domain controller is online. While disconnected from the AD server, a user cannot access any data stored on this server. When a workstation has become disconnected from the network entirely and attaches to the corporate network again later, SUSE Linux Enterprise acquires a new Kerberos ticket as soon as the user has locked and unlocked the desktop (for example, using a desktop screen saver).

11.3 Configuring a Linux Client for Active Directory

Before your client can join an AD domain, some adjustments must be made to your network setup to ensure a flawless interaction of client and server.

DNS

Configure your client machine to use a DNS server that can forward DNS requests to the AD DNS server. Alternatively, configure your machine to use the AD DNS server as the name service data source.

NTP

To succeed with Kerberos authentication, the client must have its time set accurately. It is highly encouraged to use a central NTP time server for this purpose (this can be also the NTP server running on your Active Directory domain controller). If the clockskew between your Linux host and the domain controller exceeds a certain limit, Kerberos authentication fails and the client is logged in only using the weaker NTLM (NT LAN Manager) authentication.

DHCP

If your client uses dynamic network configuration with DHCP, configure DHCP to provide the same IP and hostname to the client. If possible, use static IP addresses to be on the safe side.

Firewall

To browse your network neighborhood, either disable the firewall entirely or mark the interface used for browsing as part of the internal zone.

To change the firewall settings on your client, log in as `root` and start the YaST firewall module. Select *Interfaces*. Select your network interface from the list of interfaces and click *Change*. Select *Internal Zone* and apply your settings with *OK*. Leave the firewall settings with *Next* → *Accept*. To disable the firewall, just set *Service Start* to *Manually* and leave the firewall module with *Next* → *Accept*.

AD Account

You cannot log in to an AD domain unless the AD administrator has provided you with a valid user account for this domain. Use the AD username and password to log in to the AD domain from your Linux client.

Join an existing AD domain during installation or by later activating SMB user authentication with YaST in the installed system. The domain join during installation is covered in [Section “Configuring the Host as a Windows Domain Member”](#) (page 56).

NOTE

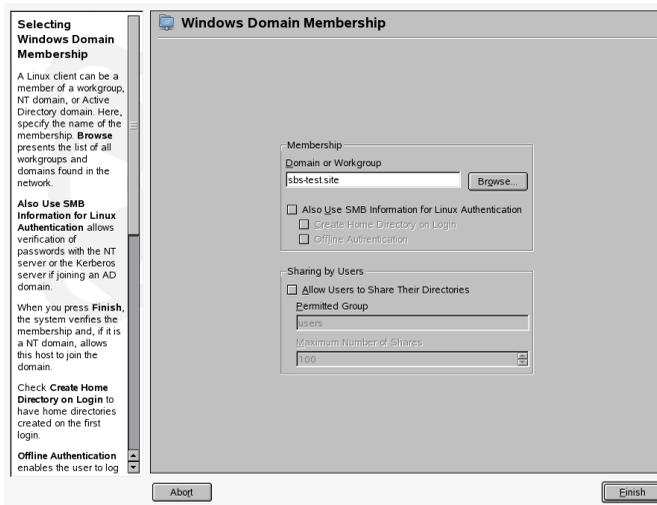
Currently only a domain administrator account, such as `Administrator`, can join SUSE Linux Enterprise Desktop into Active Directory.

To join an AD domain in a running system, proceed as follows:

- 1 Log in as `root` and start YaST.

- 2 Start *Network Services* → *Windows Domain Membership*.
- 3 Enter the domain to join at *Domain or Workgroup* in the *Windows Domain Membership* screen (see [Figure 11.2, “Determining Windows Domain Membership”](#) (page 293)). If the DNS settings on your host are properly integrated with the Windows DNS server, enter the AD domain name in its DNS format (mydomain.mycompany.com). If you enter the short name of your domain (also known as the pre-Windows 2000 domain name), YaST must rely on NetBIOS name resolution instead of DNS to find the correct domain controller. To select from a list of available domains instead, use *Browse* to list the NetBIOS domains then select the desired domain.

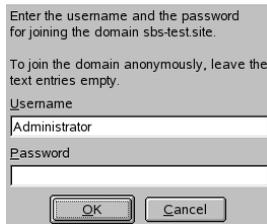
Figure 11.2 *Determining Windows Domain Membership*



- 4 Check *Also Use SMB Information for Linux Authentication* to use the SMB source for Linux authentication.
- 5 Check *Create Home Directory on Login* to automatically create a local home directory for your AD user on the Linux machine.
- 6 Check *Offline Authentication* to allow your domain users to log in even if the AD server is temporarily unavailable or you do not have a network connection.

- 7 Click *Finish* and confirm the domain join when prompted for it.
- 8 Provide the password for the Windows administrator on the AD server and click *OK* (see [Figure 11.3, “Providing Administrator Credentials”](#) (page 294)).

Figure 11.3 *Providing Administrator Credentials*



After you have joined the AD domain, you can log in to it from your workstation using the display manager of your desktop or the console.

11.4 Logging In to an AD Domain

Provided your machine has been configured to authenticate against Active Directory and you have a valid Windows user identity, you can log in to your machine using the AD credentials. Login is supported for both desktop environments (GNOME and KDE), the console, SSH, and any other PAM-aware application.

IMPORTANT: Offline Authentication

SUSE Linux Enterprise supports offline authentication, allowing you to remain logged in to your client machine even if the client machine is disconnected from the network. This enables you to maintain a mobile style of working, for example, it allows you to continue to work even if you are on an airplane and do not have a network connection.

11.4.1 GDM and KDM

To authenticate a GNOME client machine against an AD server, proceed as follows:

- 1 Select the domain.
- 2 Enter your Windows username and press `Enter`.
- 3 Enter your Windows password and press `Enter`.

To authenticate a KDE client machine against an AD server, proceed as follows:

- 1 Select the domain.
- 2 Enter your Windows username.
- 3 Enter your Windows password and press `Enter`.

If configured to do so, SUSE Linux Enterprise creates a user home directory on the local machine on first login of each AD authenticated user. This allows you to benefit from the AD support of SUSE Linux Enterprise while still having a completely capable Linux machine at your disposal.

11.4.2 Console Login

As well as logging in to the AD client machine using a graphical front-end, you can log in using the text-based console login or even remotely using SSH.

To log in to your AD client from a console, enter `DOMAIN\user` at the `login:` prompt and provide the password.

To remotely log in to your AD client machine using SSH, proceed as follows:

- 1 At the login prompt, enter:

```
ssh DOMAIN\\user@hostname
```

The `\` domain and login delimiter is escaped with another `\` sign.

- 2 Provide the user's password.

11.5 Changing Passwords

SUSE Linux Enterprise has the ability to help a user choose a suitable new password that meets the corporate security policy. The underlying PAM module retrieves the current password policy settings from the domain controller. It informs about the specific password quality requirements a user account typically has by means of a message at login time. Like the Windows counterpart, SUSE Linux Enterprise presents a message describing:

- Password history settings
- Minimum password length requirements
- Minimum password age
- Password complexity

The password change process cannot succeed unless all possible requirements have been successfully satisfied. Feedback about the password status is given both through the display managers and the console.

GDM and KDM provide feedback about password expiration and prompt for new passwords in an interactive mode. To change passwords in the display managers, just provide the password information when prompted to do so.

To change your Windows password, you can use the standard Linux utility, `passwd`, instead of having to manipulate this data on the server. To change your Windows password, proceed as follows:

- 1 Log in at the console.
- 2 Enter `passwd`.
- 3 Enter your current password when prompted to do so.
- 4 Enter the new password.
- 5 Reenter the new password for confirmation. If your new password does not comply with the policies on the Windows server, this feedback is given to you and you are prompted for another password.

To change your Windows password from the GNOME desktop, proceed as follows:

- 1** Click the *Computer* icon on the left edge of the panel.
- 2** Select *Control Center*.
- 3** From the *Personal* section, select *Change Password*.
- 4** Enter your old password.
- 5** Enter and confirm the new password.
- 6** Leave the dialog with *Close* to apply your settings.

To change your Windows password from the KDE desktop, proceed as follows:

- 1** Select *Personal Settings* from the main menu.
- 2** Select *Security & Privacy*.
- 3** Click *Password & User Account*.
- 4** Click *Change Password*.
- 5** Enter your current password.
- 6** Enter and confirm the new password and apply your settings with *OK*.
- 7** Leave the *Personal Settings* with *File* → *Quit*.

Access Control Lists in Linux

POSIX ACLs (access control lists) 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/>.

12.1 Traditional File Permissions

The basics of traditional Linux file permissions are explained in [Section 14.2, “Users and Access Permissions”](#) (page 346). More advanced features are the `setuid`, `setgid`, and sticky bit.

12.1.1 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`.

12.1.2 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.

12.1.3 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
```

12.2 Advantages of ACLs

Traditionally, three permission sets 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 as 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 evident if you want to replace 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. With Samba supporting 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*, part of the samba suite, it is even possible to assign permissions to users only existing in the Windows domain without any account on the Linux server.

12.3 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, 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.

12.4 Handling ACLs

[Table 12.1, “ACL Entry Types”](#) (page 303) 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. 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 12.2, “Masking Access Permissions”](#) (page 303) 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 12.1 *ACL Entry Types*

Type	Text Form
owner	user::rwx
named user	user:name:rwx
owning group	group::rwx
named group	group:name:rwx
mask	mask::rwx
other	other::rwx

Table 12.2 *Masking Access Permissions*

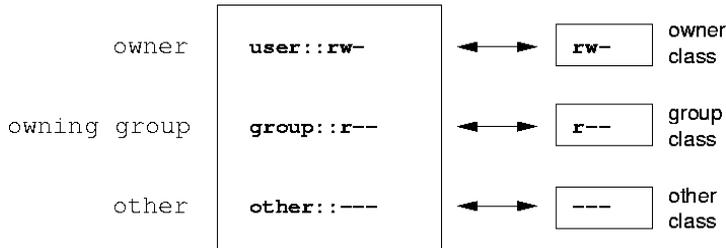
Entry Type	Text Form	Permissions
named user	user:geeko:r-x	r-x
mask	mask::rw-	rw-
	effective permissions:	r--

12.4.1 ACL Entries and File Mode Permission Bits

Figure 12.1, “Minimum ACL: ACL Entries Compared to Permission Bits” (page 304) and Figure 12.2, “Extended ACL: ACL Entries Compared to Permission Bits” (page 304) 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

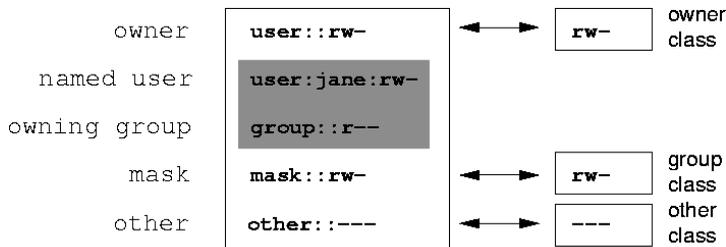
ACL entry owner. *Other class* permissions are mapped to the respective ACL entry. However, the mapping of the *group class* permissions is different in the two cases.

Figure 12.1 *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 12.1, “Minimum ACL: ACL Entries Compared to Permission Bits”](#) (page 304). In the case of an extended ACL—with mask—the group class permissions are mapped to the mask entry. This is shown in [Figure 12.2, “Extended ACL: ACL Entries Compared to Permission Bits”](#) (page 304).

Figure 12.2 *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.

12.4.2 A Directory with an Access ACL

With `getfacl` and `setfacl` on the command line, you can access ACLs. The usage of these commands is demonstrated in the following example.

Before creating 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 `umask` man page.

`mkdir mydir` creates the `mydir` directory with the default permissions as set by `umask`. Use `ls -dl mydir` to check whether 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 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 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 12.2, “Masking Access Permissions”](#) (page 303)). 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`.

12.4.3 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 ways in which the permissions of a directory's default ACL are passed to the files and subdirectories:

- 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 307)). 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.

12.4.4 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”.

12.5 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 do Samba and Konqueror.

Unfortunately, many editors and file managers still lack ACL support. When copying files with Emacs, 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.

12.6 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)`.

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 > or # 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
[...]
```

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.

13.1 Debugging

13.1.1 Specifying the Required Library: `ldd`

Use the command `ldd` to find out which libraries would load the dynamic executable specified as argument.

```
tester@linux:~> ldd /bin/ls
linux-gate.so.1 => (0xffffe000)
librt.so.1 => /lib/librt.so.1 (0xb7f97000)
libacl.so.1 => /lib/libacl.so.1 (0xb7f91000)
libc.so.6 => /lib/libc.so.6 (0xb7e79000)
libpthread.so.0 => /lib/libpthread.so.0 (0xb7e67000)
/lib/ld-linux.so.2 (0xb7fb6000)
libattr.so.1 => /lib/libattr.so.1 (0xb7e63000)
```

Static binaries do not need any dynamic libraries.

```
tester@linux:~> ldd /bin/sash
not a dynamic executable
tester@linux:~> file /bin/sash
/bin/sash: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), for
GNU/Linux 2.6.4, statically linked, for GNU/Linux 2.6.4, stripped
```

13.1.2 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:

```
tester@linux:~> ltrace -c find ~
% time      seconds  usecs/call   calls   function
-----
 34.37     6.758937      245    27554  __errno_location
 33.53     6.593562      788     8358  __fprintf_chk
 12.67     2.490392     144    17212  strlen
 11.97     2.353302     239     9845  readdir64
  2.37     0.466754      27    16716  __ctype_get_mb_cur_max
  1.18     0.231189      27     8531  strcpy
  1.17     0.230765      27     8358  memcpy
[...]
```

% time	seconds	usecs/call	calls	function
34.37	6.758937	245	27554	__errno_location
33.53	6.593562	788	8358	__fprintf_chk
12.67	2.490392	144	17212	strlen
11.97	2.353302	239	9845	readdir64
2.37	0.466754	27	16716	__ctype_get_mb_cur_max
1.18	0.231189	27	8531	strcpy
1.17	0.230765	27	8358	memcpy
[...]				
0.00	0.000036	36	1	textdomain

100.00 19.662715 105717 total

13.1.3 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:

```
tester@linux:~> strace ls
execve("/bin/ls", ["ls"], [/* 61 vars */]) = 0
uname({sys="Linux", node="linux", ...}) = 0
brk(0) = 0x805c000
access("/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or \
    directory)
open("/etc/ld.so.cache", O_RDONLY) = 3
fstat64(3, {st_mode=S_IFREG|0644, st_size=89696, ...}) = 0
mmap2(NULL, 89696, PROT_READ, MAP_PRIVATE, 3, 0) = 0xb7ef2000
close(3) = 0
open("/lib/librt.so.1", O_RDONLY) = 3
read(3, "\177ELF\1\1\1\0\0\0\0\0\0\0\0\3\0\3\0\1\0\0\000\36\0"... , 512) \
    = 512
fstat64(3, {st_mode=S_IFREG|0755, st_size=36659, ...}) = 0
[...]
stat64(1, {st_mode=S_IFCHR|0620, st_rdev=makedev(136, 0), ...}) = 0
mmap2(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) \
    = 0xb7ca7000
write(1, "bin Desktop Documents music\tM"... , 55bin Desktop Documents \
    \ music Music public_html tmp
) = 55
close(1) = 0
munmap(0xb7ca7000, 4096) = 0
exit_group(0) = ?
```

For example, to trace all attempts to open a particular file, use the following:

```
tester@linux:~> strace -e open ls .bashrc
open("/etc/ld.so.cache", O_RDONLY) = 3
open("/lib/librt.so.1", O_RDONLY) = 3
open("/lib/libacl.so.1", O_RDONLY) = 3
open("/lib/libc.so.6", O_RDONLY) = 3
open("/lib/libpthread.so.0", O_RDONLY) = 3
open("/lib/libattr.so.1", O_RDONLY) = 3
[...]
```

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`.

13.2 Files and File Systems

13.2.1 Determine the File Type: `file`

The command `file` determines the type of a file or a list of files by checking `/etc/magic`.

```
tester@linux:~> file /usr/bin/file
/usr/bin/file: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), \
    for GNU/Linux 2.2.5, dynamically linked (uses shared libs), stripped
```

The parameter `-f list` specifies a file with a list of filenames to examine. The `-z` allows `file` to look inside compressed files:

```
tester@linux:~> file /usr/share/man/man1/file.1.gz
usr/share/man/man1/file.1.gz: gzip compressed data, from Unix, max compression
tester@linux:~> file -z /usr/share/man/man1/file.1.gz
/usr/share/man/man1/file.1.gz: ASCII troff or preprocessor input text \
    (gzip compressed data, from Unix, max compression)
```

13.2.2 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:

```
tester@linux:~> mount
/dev/hda3 on / type reiserfs (rw,acl,user_xattr)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
udev on /dev type tmpfs (rw)
devpts on /dev/pts type devpts (rw,mode=0620,gid=5)
/dev/hda1 on /boot type ext2 (rw,acl,user_xattr)
/dev/hda4 on /local type reiserfs (rw,acl,user_xattr)
/dev/fd0 on /media/floppy type subfs (rw,nosuid,nodev,noatime,fs=floppyfs,p
```

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.

```
tester@linux:~> df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/hda3       11G   3.2G   6.9G  32% /
```

```

udev                252M  104K  252M   1% /dev
/dev/hda1           16M   6.6M   7.8M  46% /boot
/dev/hda4           27G   34M   27G   1% /local

```

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 human-readable form:

```

tester@linux:~> du -sh /local
1.7M   /local

```

13.2.3 Additional Information about ELF Binaries

Read the content of binaries with the `readelf` utility. This even works with ELF files that were built for other hardware architectures:

```

tester@linux:~> readelf --file-header /bin/ls
ELF Header:
  Magic:   7f 45 4c 46 01 01 01 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:                   0x8049b60
  Start of program headers:               52 (bytes into file)
  Start of section headers:               81112 (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:              30
  Section header string table index:      29

```

13.2.4 File Properties: `stat`

The command `stat` displays file properties:

```

tester@linux:~> stat /etc/profile
  File: `/etc/profile'
  Size: 7930          Blocks: 16          IO Block: 4096   regular file

```

```
Device: 303h/771d      Inode: 40657      Links: 1
Access: (0644/-rw-r--r--)  Uid: (    0/    root)  Gid: (    0/    root)
Access: 2006-01-06 16:45:43.000000000 +0100
Modify: 2005-11-21 14:54:35.000000000 +0100
Change: 2005-12-19 09:51:04.000000000 +0100
```

The parameter `--filesystem` produces details of the properties of the file system in which the specified file is located:

```
tester@linux:~> stat /etc/profile --filesystem
  File: "/etc/profile"
   ID: 0      Namelen: 255      Type: reiserfs
Block size: 4096      Fundamental block size: 4096
Blocks: Total: 2622526      Free: 1809771      Available: 1809771
Inodes: Total: 0      Free: 0
```

13.3 Hardware Information

13.3.1 PCI Resources: `lspci`

The command `lspci` lists the PCI resources:

```
linux:~ # lspci
00:00.0 Host bridge: Intel Corporation 82845G/GL[Brookdale-G]/GE/PE \
  DRAM Controller/Host-Hub Interface (rev 01)
00:01.0 PCI bridge: Intel Corporation 82845G/GL[Brookdale-G]/GE/PE \
  Host-to-AGP Bridge (rev 01)
00:1d.0 USB Controller: Intel Corporation 82801DB/DBL/DBM \
  (ICH4/ICH4-L/ICH4-M) USB UHCI Controller #1 (rev 01)
00:1d.1 USB Controller: Intel Corporation 82801DB/DBL/DBM \
  (ICH4/ICH4-L/ICH4-M) USB UHCI Controller #2 (rev 01)
00:1d.2 USB Controller: Intel Corporation 82801DB/DBL/DBM \
  (ICH4/ICH4-L/ICH4-M) USB UHCI Controller #3 (rev 01)
00:1d.7 USB Controller: Intel Corporation 82801DB/DBM \
  (ICH4/ICH4-M) USB2 EHCI Controller (rev 01)
00:1e.0 PCI bridge: Intel Corporation 82801 PCI Bridge (rev 81)
00:1f.0 ISA bridge: Intel Corporation 82801DB/DBL (ICH4/ICH4-L) \
  LPC Interface Bridge (rev 01)
00:1f.1 IDE interface: Intel Corporation 82801DB (ICH4) IDE \
  Controller (rev 01)
00:1f.3 SMBus: Intel Corporation 82801DB/DBL/DBM (ICH4/ICH4-L/ICH4-M) \
  SMBus Controller (rev 01)
00:1f.5 Multimedia audio controller: Intel Corporation 82801DB/DBL/DBM \
  (ICH4/ICH4-L/ICH4-M) AC'97 Audio Controller (rev 01)
01:00.0 VGA compatible controller: Matrox Graphics, Inc. G400/G450 (rev 85)
02:08.0 Ethernet controller: Intel Corporation 82801DB PRO/100 VE (LOM) \
  Ethernet Controller (rev 81)
```

Using `-v` results in a more detailed listing:

```
linux:~ # lspci
[...]
```

02:08.0 Ethernet controller: Intel Corporation 82801DB PRO/100 VE (LOM)\
Ethernet Controller (rev 81)
Subsystem: Fujitsu Siemens Computer GmbH: Unknown device 1001
Flags: bus master, medium devsel, latency 66, IRQ 11
Memory at d1000000 (32-bit, non-prefetchable) [size=4K]
I/O ports at 3000 [size=64]
Capabilities: [dc] Power Management version 2

Information about device name resolution is obtained from the 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, use the parameter `-n`.

13.3.2 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` with these USB devices attached: hub, memory stick, hard disk, and mouse.

```
linux:/ # lsusb
Bus 004 Device 007: ID 0ea0:2168 Ours Technology, Inc. Transcend JetFlash \
  2.0 / Astone USB Drive
Bus 004 Device 006: ID 04b4:6830 Cypress Semiconductor Corp. USB-2.0 IDE \
  Adapter
Bus 004 Device 005: ID 05e3:0605 Genesys Logic, Inc.
Bus 004 Device 001: ID 0000:0000
Bus 003 Device 001: ID 0000:0000
Bus 002 Device 001: ID 0000:0000
Bus 001 Device 005: ID 046d:c012 Logitech, Inc. Optical Mouse
Bus 001 Device 001: ID 0000:0000
```

13.3.3 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.

```
linux:/ # scsiinfo -i /dev/sda
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
```

The option `-d` puts out 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.

13.4 Networking

13.4.1 Show the Network Status: `netstat`

`netstat` shows network connections, routing tables (`-r`), interfaces (`-i`), masquerade connections (`-M`), multicast memberships (`-g`), and statistics (`-s`).

```
tester@linux:~> netstat -r
Kernel IP routing table
Destination Gateway Genmask Flags MSS Window irtt Iface
192.168.22.0 * 255.255.254.0 U 0 0 0 eth0
link-local * 255.255.0.0 U 0 0 0 eth0
loopback * 255.0.0.0 U 0 0 0 lo
default 192.168.22.254 0.0.0.0 UG 0 0 0 eth0
```

```

tester@linux:~> netstat -i
Kernel Interface table
Iface   MTU Met   RX-OK RX-ERR RX-DRP RX-OVR   TX-OK TX-ERR TX-DRP TX-OVR Flg
eth0    1500  0 1624507 129056      0      0  7055    0      0      0 BMNRU
lo      16436  0  23728      0      0      0  23728    0      0      0 LRU

```

When displaying network connections or statistics, you can specify the socket type to display: TCP (`-t`), UDP (`-u`), or raw (`-r`). The `-p` option shows the PID and name of the program to which each socket belongs.

The following example lists all TCP connections and the programs using these connections.

```

linux:~ # netstat -t -p
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address      Foreign Address    State      PID/Pro
tcp      0      0 linux:33513       www.novell.com:www-http ESTABLISHED 6862/fi
tcp      0    352 linux:ssh         linux2.:trc-netpoll ESTABLISHED 19422/s
tcp      0      0 localhost:ssh     localhost:17828     ESTABLISHED -

```

In the following, statistics for the TCP protocol are displayed:

```

tester@linux:~> netstat -s -t
Tcp:
  2427 active connections openings
  2374 passive connection openings
  0 failed connection attempts
  0 connection resets received
  1 connections established
  27476 segments received
  26786 segments send out
  54 segments retransmitted
  0 bad segments received.
  6 resets sent
[...]
TCPAbortOnLinger: 0
TCPAbortFailed: 0
TCPMemoryPressures: 0

```

13.5 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:

```
tester@linux:~> 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
[...]
```

Query the allocation and use of interrupts with the following command:

```
tester@linux:~> cat /proc/interrupts
CPU0
 0:   3577519      XT-PIC  timer
 1:     130       XT-PIC  i8042
 2:     0        XT-PIC  cascade
 5:   564535      XT-PIC  Intel 82801DB-ICH4
 7:     1        XT-PIC  parport0
 8:     2        XT-PIC  rtc
 9:     1        XT-PIC  acpi, uhci_hcd:usb1, ehci_hcd:usb4
10:     0        XT-PIC  uhci_hcd:usb3
11:    71772      XT-PIC  uhci_hcd:usb2, eth0
12:   101150      XT-PIC  i8042
14:    33146      XT-PIC  ide0
15:   149202      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. Find information about processes currently running 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/ :

```
tester@linux:~> ls -l /proc/self
lrwxrwxrwx 1 root root 64 2006-01-09 13:03 /proc/self -> 5356
tester@linux:~> ls -l /proc/self/
total 0
dr-xr-xr-x 2 tester users 0 2006-01-09 17:04 attr
-r----- 1 tester users 0 2006-01-09 17:04 auxv
-r--r--r-- 1 tester users 0 2006-01-09 17:04 cmdline
lrwxrwxrwx 1 tester users 0 2006-01-09 17:04 cwd -> /home/tester
-r----- 1 tester users 0 2006-01-09 17:04 environ
lrwxrwxrwx 1 tester users 0 2006-01-09 17:04 exe -> /bin/ls
dr-x----- 2 tester users 0 2006-01-09 17:04 fd
-rw-r--r-- 1 tester users 0 2006-01-09 17:04 loginuid
-r--r--r-- 1 tester users 0 2006-01-09 17:04 maps
-rw----- 1 tester users 0 2006-01-09 17:04 mem
-r--r--r-- 1 tester users 0 2006-01-09 17:04 mounts
-rw-r--r-- 1 tester users 0 2006-01-09 17:04 oom_adj
-r--r--r-- 1 tester users 0 2006-01-09 17:04 oom_score
lrwxrwxrwx 1 tester users 0 2006-01-09 17:04 root -> /
-rw----- 1 tester users 0 2006-01-09 17:04 seccomp
-r--r--r-- 1 tester users 0 2006-01-09 17:04 smaps
-r--r--r-- 1 tester users 0 2006-01-09 17:04 stat
-r--r--r-- 1 tester users 0 2006-01-09 17:04 statm
-r--r--r-- 1 tester users 0 2006-01-09 17:04 status
dr-xr-xr-x 3 tester users 0 2006-01-09 17:04 task
-r--r--r-- 1 tester users 0 2006-01-09 17:04 wchan
```

The address assignment of executables and libraries is contained in the maps file:

```
tester@linux:~> cat /proc/self/maps
08048000-0804c000 r-xp 00000000 03:03 17753      /bin/cat
0804c000-0804d000 rw-p 00004000 03:03 17753      /bin/cat
0804d000-0806e000 rw-p 0804d000 00:00 0         [heap]
b7d27000-b7d5a000 r--p 00000000 03:03 11867     /usr/lib/locale/en_GB.utf8/
b7d5a000-b7e32000 r--p 00000000 03:03 11868     /usr/lib/locale/en_GB.utf8/
b7e32000-b7e33000 rw-p b7e32000 00:00 0
b7e33000-b7f45000 r-xp 00000000 03:03 8837      /lib/libc-2.3.6.so
b7f45000-b7f46000 r--p 00112000 03:03 8837      /lib/libc-2.3.6.so
b7f46000-b7f48000 rw-p 00113000 03:03 8837      /lib/libc-2.3.6.so
b7f48000-b7f4c000 rw-p b7f48000 00:00 0
b7f52000-b7f53000 r--p 00000000 03:03 11842     /usr/lib/locale/en_GB.utf8/
[...]
```

```

b7f5b000-b7f61000 r--s 00000000 03:03 9109      /usr/lib/gconv/gconv-module
b7f61000-b7f62000 r--p 00000000 03:03 9720      /usr/lib/locale/en_GB.utf8/
b7f62000-b7f76000 r-xp 00000000 03:03 8828      /lib/ld-2.3.6.so
b7f76000-b7f78000 rw-p 00013000 03:03 8828      /lib/ld-2.3.6.so
bfd61000-bfd76000 rw-p bfd61000 00:00 0         [stack]
ffffe000-fffff000 ---p 00000000 00:00 0         [vdso]

```

13.5.1 procinfo

Important information from the `/proc` file system is summarized by the command `procinfo`:

```

tester@linux:~> procinfo
Linux 2.6.15-rc5-git3-2-default (geeko@buildhost) (gcc 4.1.0 20051129) #1 Wed

Memory:      Total      Used      Free      Shared      Buffers
Mem:         515584    509472    6112      0           73024
Swap:        658656      0        658656

Bootup: Mon Jan  9 12:59:08 2006   Load average: 0.10 0.04 0.05 1/86 5406

user  :      0:02:07.98   0.8% page in :      442638  disk 1:    20125r 134
nice  :      0:02:20.91   0.9% page out:      134950
system: 0:00:42.93   0.3% page act:      70577
IOWait: 0:01:25.40   0.6% page dea:      11696
hw irq: 0:00:08.94   0.1% page flt:    1423622
sw irq: 0:00:01.29   0.0% swap in :           0
idle  :      4:06:30.54  97.3% swap out:           0
uptime: 4:13:20.72      context :    3813145

irq 0:  3799268 timer          irq 8:      2 rtc
irq 1:    130 i8042          irq 9:      1 acpi, uhci_hcd:usb
irq 2:     0 cascade [4]     irq 10:     0 uhci_hcd:usb3
irq 3:     8                irq 11:    75905 uhci_hcd:usb2, eth
irq 4:     8                irq 12:   101150 i8042
irq 5:   564535 Intel 82801DB-ICH4  irq 14:    33733 ide0
irq 6:     9                irq 15:   157045 ide1
irq 7:     1 parport0 [3]

```

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:

13.6 Processes

13.6.1 Interprocess Communication: `ipcs`

The command `ipcs` produces a list of the IPC resources currently in use:

```
----- Shared Memory Segments -----
key          shmid      owner      perms      bytes      nattch     status
0x00000000  58261504  tester    600        393216     2          dest
0x00000000  58294273  tester    600        196608     2          dest
0x00000000  83886083  tester    666        43264      2
0x00000000  83951622  tester    666        192000     2
0x00000000  83984391  tester    666        282464     2
0x00000000  84738056  root      644        151552     2          dest

----- Semaphore Arrays -----
key          semid      owner      perms      nsems
0x4d038abf  0          tester    600        8

----- Message Queues -----
key          msqid      owner      perms      used-bytes  messages
```

13.6.2 Process List: `ps`

The command `ps` produces a list of processes. Most parameters must be written without a minus sign. Refer to `ps --help` for a brief help or to the man page for extensive help.

To list all processes with user and command line information, use `ps aux`:

```
tester@linux:~> ps aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0   696   272 ?        S    12:59    0:01 init [5]
root         2  0.0  0.0     0     0 ?        SN   12:59    0:00 [ksoftirqd
root         3  0.0  0.0     0     0 ?        S<   12:59    0:00 [events]
[...]
tester   4047  0.0  6.0 158548 31400 ?        Ssl  13:02    0:06 mono-best
tester   4057  0.0  0.7  9036  3684 ?        Sl   13:02    0:00 /opt/gnome
tester   4067  0.0  0.1  2204   636 ?        S    13:02    0:00 /opt/gnome
tester   4072  0.0  1.0 15996   5160 ?        Ss   13:02    0:00 gnome-scre
tester   4114  0.0  3.7 130988 19172 ?        SLl  13:06    0:04 sound-juic
tester   4818  0.0  0.3  4192   1812 pts/0    Ss   15:59    0:00 -bash
tester   4959  0.0  0.1  2324    816 pts/0    R+   16:17    0:00 ps aux
```

To check how many `sshd` processes are running, use the option `-p` together with the command `pidof`, which lists the process IDs of the given processes.

```
tester@linux:~> ps -p `pidof sshd`
  PID TTY          STAT       TIME COMMAND
 3524 ?            Ss          0:00 /usr/sbin/sshd -o PidFile=/var/run/sshd.init.pid
 4813 ?            Ss          0:00 sshd: tester [priv]
 4817 ?            R           0:00 sshd: tester@pts/0
```

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:

```
tester@linux:~> ps ax --format pid,rss,cmd --sort rss
  PID  RSS CMD
    2     0 [ksoftirqd/0]
    3     0 [events/0]
    4     0 [khelper]
    5     0 [kthread]
   11     0 [kblockd/0]
   12     0 [kacpid]
  472     0 [pdflush]
  473     0 [pdflush]
[...]
```

```
4028 17556 nautilus --no-default-window --sm-client-id default2
4118 17800 ksnapshot
4114 19172 sound-juicer
4023 25144 gnome-panel --sm-client-id default1
4047 31400 mono-best --debug /usr/lib/beagle/Best.exe --autostarted
3973 31520 mono-beagled --debug /usr/lib/beagle/BeagleDaemon.exe --bg --aut
```

13.6.3 Process Tree: `pstree`

The command `pstree` produces a list of processes in the form of a tree:

```
tester@linux:~> pstree
init--NetworkManagerD
  |-acpid
  |-3*[automount]
  |-cron
  |-cupsd
  |-2*[dbus-daemon]
  |-dbus-launch
  |-dcopserver
  |-dhcpcd
  |-events/0
  |-gpg-agent
  |-hald--hald-addon-acpi
  |   `--hald-addon-stor
```

```

|-kded
|-kdeinit--kdesu---su---kdesu_stub---yast2---y2controlcenter
|
|   |-kio_file
|   |-klauncher
|   |-konqueror
|   |-konsole--bash---su---bash
|   |   `--bash
|   `--kwin
|-kdesktop---kdesktop_lock---xmatrix
|-kdesud
|-kdm--X
|   `--kdm---startkde---kwrapper
[...]
```

The parameter `-p` adds the process ID to a given name. To have the command lines displayed as well, use the `-a` parameter:

13.6.4 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`:

```

tester@linux:~> top -n 1
top - 17:06:28 up 2:10, 5 users, load average: 0.00, 0.00, 0.00
Tasks: 85 total, 1 running, 83 sleeping, 1 stopped, 0 zombie
Cpu(s): 5.5% us, 0.8% sy, 0.8% ni, 91.9% id, 1.0% wa, 0.0% hi, 0.0% si
Mem: 515584k total, 506468k used, 9116k free, 66324k buffers
Swap: 658656k total, 0k used, 658656k free, 353328k cached

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
    1 root        16   0   700   272  236  S   0.0   0.1   0:01.33  init
    2 root        34  19     0     0     0  S   0.0   0.0   0:00.00  ksoftirqd/0
    3 root        10  -5     0     0     0  S   0.0   0.0   0:00.27  events/0
    4 root        10  -5     0     0     0  S   0.0   0.0   0:00.01  khelper
    5 root        10  -5     0     0     0  S   0.0   0.0   0:00.00  kthread
   11 root        10  -5     0     0     0  S   0.0   0.0   0:00.05  kblockd/0
   12 root        20  -5     0     0     0  S   0.0   0.0   0:00.00  kacpid
  472 root        20   0     0     0     0  S   0.0   0.0   0:00.00  pdflush
  473 root        15   0     0     0     0  S   0.0   0.0   0:00.06  pdflush
  475 root        11  -5     0     0     0  S   0.0   0.0   0:00.00  aio/0
  474 root        15   0     0     0     0  S   0.0   0.0   0:00.07  kswapd0
  681 root        10  -5     0     0     0  S   0.0   0.0   0:00.01  kseriod
  839 root        10  -5     0     0     0  S   0.0   0.0   0:00.02  reiserfs/0
  923 root        13  -4   1712  552  344  S   0.0   0.1   0:00.67  udevd
 1343 root        10  -5     0     0     0  S   0.0   0.0   0:00.00  khubd
 1587 root        20   0     0     0     0  S   0.0   0.0   0:00.00  shpchpd_event
 1746 root        15   0     0     0     0  S   0.0   0.0   0:00.00  wl_control
```

```

1752 root      15   0     0     0     0 S   0.0  0.0    0:00.00 wl_bus_master1
2151 root      16   0  1464  496  416 S   0.0  0.1    0:00.00 acpid
2165 messageb  16   0  3340 1048  792 S   0.0  0.2    0:00.64 dbus-daemon
2166 root      15   0  1840  752  556 S   0.0  0.1    0:00.01 syslog-ng
2171 root      16   0  1600  516  320 S   0.0  0.1    0:00.00 klogd
2235 root      15   0  1736  800  652 S   0.0  0.2    0:00.10 resmgrd
2289 root      16   0  4192 2852 1444 S   0.0  0.6    0:02.05 hald
2403 root      23   0  1756  600  524 S   0.0  0.1    0:00.00 hald-addon-acpi
2709 root      19   0  2668 1076  944 S   0.0  0.2    0:00.00 NetworkManagerD
2714 root      16   0  1756  648  564 S   0.0  0.1    0:00.56 hald-addon-stor

```

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`` returns the UID of the user on the basis of the username and displays his processes.

13.7 System Information

13.7.1 System Activity Information: `sar`

To use `sar`, `sadc` (system activity data collector) needs to be running. Check its status or start it with `rcsysstat {start|status}`.

`sar` can generate extensive reports on almost all important system activities, among them CPU, memory, IRQ usage, IO, or networking. With its many options, it is too complex to explain further here. Refer to the man page for extensive documentation with examples.

13.7.2 Memory Usage: `free`

The utility `free` examines RAM usage. Details of both free and used memory and swap areas are shown:

```

tester@linux:~> free
              total          used          free      shared    buffers     cached
Mem:          515584       501704        13880           0        73040       334592
-/+ buffers/cache:          94072       421512
Swap:          658656              0        658656

```

The options `-b,-k,-m,-g` show output in bytes, KB, MB, or GB, respectively. 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.

13.7.3 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:

```
tester@linux:~> fuser -v /mnt/*

/mnt/notes.txt          USER          PID ACCESS COMMAND
                        tester         26597 f....  less
```

Following termination of the `less` process, which was running on another terminal, the file system can successfully be unmounted.

13.7.4 Kernel Ring Buffer: `dmesg`

The Linux kernel keeps certain messages in a ring buffer. To view these messages, enter the command `dmesg`:

```
$ dmesg
[...]
end_request: I/O error, dev fd0, sector 0
subfs: unsuccessful attempt to mount media (256)
e100: eth0: e100_watchdog: link up, 100Mbps, half-duplex
NET: Registered protocol family 17
IA-32 Microcode Update Driver: v1.14 <tigran@veritas.com>
microcode: CPU0 updated from revision 0xe to 0x2e, date = 08112004
IA-32 Microcode Update Driver v1.14 unregistered
boot splash: status on console 0 changed to on
NET: Registered protocol family 10
Disabled Privacy Extensions on device c0326ea0(10)
IPv6 over IPv4 tunneling driver
powernow: This module only works with AMD K7 CPUs
boot splash: status on console 0 changed to on
```

Older events are logged in the files `/var/log/messages` and `/var/log/warn`.

13.7.5 List of Open Files: `lsdf`

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:

```
tester@linux:~> lsdf -p $$
COMMAND PID USER FD TYPE DEVICE SIZE NODE NAME
bash 5552 tester cwd DIR 3,3 1512 117619 /home/tester
bash 5552 tester rtd DIR 3,3 584 2 /
bash 5552 tester txt REG 3,3 498816 13047 /bin/bash
bash 5552 tester mem REG 0,0 0 [heap] (stat: No such
bash 5552 tester mem REG 3,3 217016 115687 /var/run/nscd/passwd
bash 5552 tester mem REG 3,3 208464 11867 /usr/lib/locale/en_GB.
bash 5552 tester mem REG 3,3 882134 11868 /usr/lib/locale/en_GB.
bash 5552 tester mem REG 3,3 1386997 8837 /lib/libc-2.3.6.so
bash 5552 tester mem REG 3,3 13836 8843 /lib/libdl-2.3.6.so
bash 5552 tester mem REG 3,3 290856 12204 /lib/libncurses.so.5.5
bash 5552 tester mem REG 3,3 26936 13004 /lib/libhistory.so.5.1
bash 5552 tester mem REG 3,3 190200 13006 /lib/libreadline.so.5.
bash 5552 tester mem REG 3,3 54 11842 /usr/lib/locale/en_GB.
bash 5552 tester mem REG 3,3 2375 11663 /usr/lib/locale/en_GB.
bash 5552 tester mem REG 3,3 290 11736 /usr/lib/locale/en_GB.
bash 5552 tester mem REG 3,3 52 11831 /usr/lib/locale/en_GB.
bash 5552 tester mem REG 3,3 34 11862 /usr/lib/locale/en_GB.
bash 5552 tester mem REG 3,3 62 11839 /usr/lib/locale/en_GB.
bash 5552 tester mem REG 3,3 127 11664 /usr/lib/locale/en_GB.
bash 5552 tester mem REG 3,3 56 11735 /usr/lib/locale/en_GB.
bash 5552 tester mem REG 3,3 23 11866 /usr/lib/locale/en_GB.
bash 5552 tester mem REG 3,3 21544 9109 /usr/lib/gconv/gconv-m
bash 5552 tester mem REG 3,3 366 9720 /usr/lib/locale/en_GB.
bash 5552 tester mem REG 3,3 97165 8828 /lib/ld-2.3.6.so
bash 5552 tester 0u CHR 136,5 7 /dev/pts/5
bash 5552 tester 1u CHR 136,5 7 /dev/pts/5
bash 5552 tester 2u CHR 136,5 7 /dev/pts/5
bash 5552 tester 255u CHR 136,5 7 /dev/pts/5
```

The special shell variable `$$`, whose value is the process ID of the shell, has been used.

The command `lsdf` 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:

```
tester@linux:~> lsdf | grep CHR
bash 3838 tester 0u CHR 136,0 2 /dev/pts/0
bash 3838 tester 1u CHR 136,0 2 /dev/pts/0
bash 3838 tester 2u CHR 136,0 2 /dev/pts/0
bash 3838 tester 255u CHR 136,0 2 /dev/pts/0
bash 5552 tester 0u CHR 136,5 7 /dev/pts/5
```

bash	5552	tester	1u	CHR	136,5	7 /dev/pts/5
bash	5552	tester	2u	CHR	136,5	7 /dev/pts/5
bash	5552	tester	255u	CHR	136,5	7 /dev/pts/5
X	5646	root	mem	CHR	1,1	1006 /dev/mem
lsuf	5673	tester	0u	CHR	136,5	7 /dev/pts/5
lsuf	5673	tester	2u	CHR	136,5	7 /dev/pts/5
grep	5674	tester	1u	CHR	136,5	7 /dev/pts/5
grep	5674	tester	2u	CHR	136,5	7 /dev/pts/5

13.7.6 Kernel and udev Event Sequence

Viewer: `udevmonitor`

`udevmonitor` listens to the kernel uevents and events sent out by a udev rule and prints the device path (DEVPATH) of the event to the console. This is a sequence of events while connecting a USB memory stick:

```

UEVENT[1138806687] add@/devices/pci0000:00/0000:00:1d.7/usb4/4-2/4-2.2
UEVENT[1138806687] add@/devices/pci0000:00/0000:00:1d.7/usb4/4-2/4-2.2/4-2.2
UEVENT[1138806687] add@/class/scsi_host/host4
UEVENT[1138806687] add@/class/usb_device/usbdev4.10
UDEV [1138806687] add@/devices/pci0000:00/0000:00:1d.7/usb4/4-2/4-2.2
UDEV [1138806687] add@/devices/pci0000:00/0000:00:1d.7/usb4/4-2/4-2.2/4-2.2
UDEV [1138806687] add@/class/scsi_host/host4
UDEV [1138806687] add@/class/usb_device/usbdev4.10
UEVENT[1138806692] add@/devices/pci0000:00/0000:00:1d.7/usb4/4-2/4-2.2/4-2.2
UEVENT[1138806692] add@/block/sdb
UEVENT[1138806692] add@/class/scsi_generic/sg1
UEVENT[1138806692] add@/class/scsi_device/4:0:0:0
UDEV [1138806693] add@/devices/pci0000:00/0000:00:1d.7/usb4/4-2/4-2.2/4-2.2
UDEV [1138806693] add@/class/scsi_generic/sg1
UDEV [1138806693] add@/class/scsi_device/4:0:0:0
UDEV [1138806693] add@/block/sdb
UEVENT[1138806694] add@/block/sdb/sdb1
UDEV [1138806694] add@/block/sdb/sdb1
UEVENT[1138806694] mount@/block/sdb/sdb1
UEVENT[1138806697] umount@/block/sdb/sdb1

```

13.7.7 Server Resources Used by X11 Clients:

`xrestop`

`xrestop` provides statistics for each connected X11 client's server-side resource. The output is very similar to [Section 13.6.4, “Processes: `top`”](#) (page 325).

```
xrestop - Display: localhost:0
Monitoring 40 clients. XErrors: 0
Pixmaps: 42013K total, Other: 206K total, All: 42219K total
```

res-base	Wins	GCs	Fnts	Pxms	Misc	Pxm mem	Other	Total	PID	Identifier
3e00000	385	36	1	751	107	18161K	13K	18175K	?	NOVELL: SU
4600000	391	122	1	1182	889	4566K	33K	4600K	?	amaroK - S
1600000	35	11	0	76	142	3811K	4K	3816K	?	KDE Deskto
3400000	52	31	1	69	74	2816K	4K	2820K	?	Linux Shel
2c00000	50	25	1	43	50	2374K	3K	2378K	?	Linux Shel
2e00000	50	10	1	36	42	2341K	3K	2344K	?	Linux Shel
2600000	37	24	1	34	50	1772K	3K	1775K	?	Root - Kon
4800000	37	24	1	34	49	1772K	3K	1775K	?	Root - Kon
2a00000	209	33	1	323	238	1111K	12K	1123K	?	Trekstor25
1800000	182	32	1	302	285	1039K	12K	1052K	?	kicker
1400000	157	121	1	231	477	777K	18K	796K	?	kwin
3c00000	175	36	1	248	168	510K	9K	520K	?	de.comp.la
3a00000	326	42	1	579	444	486K	20K	506K	?	[opensuse-
0a00000	85	38	1	317	224	102K	9K	111K	?	Kopete
4e00000	25	17	1	60	66	63K	3K	66K	?	YaST Contr
2400000	11	10	0	56	51	53K	1K	55K	22061	suseplugge
0e00000	20	12	1	50	92	50K	3K	54K	22016	kded
3200000	6	41	5	72	84	40K	8K	48K	?	EMACS
2200000	54	9	1	30	31	42K	3K	45K	?	SUSEWatche
4400000	2	11	1	30	34	34K	2K	36K	16489	kdesu
1a00000	255	7	0	42	11	19K	6K	26K	?	KMix
3800000	2	14	1	34	37	21K	2K	24K	22242	knotify
1e00000	10	7	0	42	9	15K	624B	15K	?	KPowersave
3600000	106	6	1	30	9	7K	3K	11K	22236	konqueror
2000000	10	5	0	21	34	9K	1K	10K	?	klipper
3000000	21	7	0	11	9	7K	888B	8K	?	KDE Wallet

13.8 User Information

13.8.1 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:

```
tester@linux:~> w
 16:33:03 up 3:33, 2 users, load average: 0.14, 0.06, 0.02
USER      TTY      LOGIN@  IDLE   JCPU   PCPU WHAT
tester    :0          16:33   ?xdm?  9.42s  0.15s /bin/sh /opt/kde3/bin/startk
tester    pts/0      15:59    0.00s  0.19s  0.00s w
```

If any users of other systems have logged in remotely, the parameter `-f` shows the computers from which they have established the connection.

13.9 Time and Date

13.9.1 Time Measurement with `time`

Determine the time spent by commands with the `time` utility. This utility is available in two versions: as a shell built-in and as a program (`/usr/bin/time`).

```
tester@linux:~> time find . > /dev/null
```

```
real    0m4.051s
user    0m0.042s
sys     0m0.205s
```


Working with the Shell

When booting your Linux system, you are usually directed to a graphical user interface that guides you through the login process and the following interactions with the system. Although graphical user interfaces have become very important and user-friendly, using them is not the only way to communicate with your system. You can also use a text-oriented communication like a command line interpreter, usually called the shell, where you can enter commands. Because Linux provides options to start shell windows from the graphical user interface, you can easily use both methods.

In administration, shell-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. For Linux “newbies” it might be rather unusual to enter commands in a shell, but you will soon realize that the shell is not only for administrators—in fact, using the shell is often the quickest and easiest way to perform some daily tasks.

There are several shells for UNIX or Linux. The default shell in SUSE® Linux Enterprise is Bash (GNU Bourne-Again Shell).

This chapter deals with a couple of basics you need to know for using the shell. This includes the following topics: how to enter commands, the directory structure of Linux, how to work with files and directories and how to use some basic functions, the user and permission concept of Linux, an overview of important shell commands, and a short introduction to the `vi` editor, which is a default editor always available in Unix and Linux systems.

14.1 Getting Started with the Bash Shell

In Linux, you can use the command line parallel to the graphical user interface and easily switch between them. To start a terminal window from the graphical user interface in KDE, click the Konsole icon in the panel. In GNOME, click the GNOME Terminal icon in the panel.

The Konsole or the GNOME Terminal window appears, showing the prompt on the first line like in [Figure 14.1, “Example of a Bash Terminal Window”](#) (page 334). The prompt usually shows your login name (in this example, `tux`), the hostname of your computer (here, `knox`), and the current path (in this case, your home directory, indicated by the tilde symbol, `~`). When you are logged in on a remote computer this information always shows you which system you are currently working on. When the cursor is after this prompt, you can send commands directly to your computer system.

Figure 14.1 *Example of a Bash Terminal Window*



14.1.1 Entering Commands

A command consists of several elements. The first element is always the actual command, followed by parameters or options. You can type a command and edit it by using `←`, `→`, `←`, `Del`, and `Space`. You can also add options or correct typing errors. The command is executed when you press `Enter`.

IMPORTANT: No News Is Good News

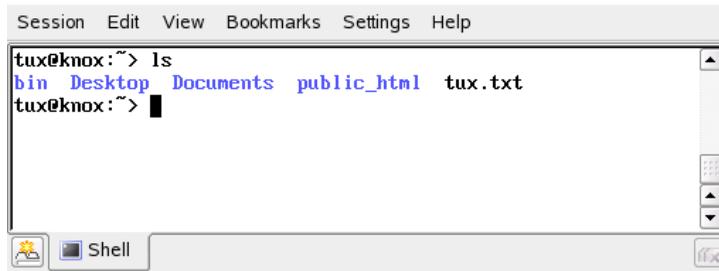
The shell is not verbose: in contrast to some graphical user interfaces, it usually does not provide confirmation messages when commands have been executed. Messages only appear in case of problems or errors.

Also keep this in mind for commands to delete objects. Before entering a command like `rm` for removing a file, you should know if you really want to get rid of the object: it will be deleted irretrievably, without enquiry.

Using Commands without Options

Look at the structure of commands using a simple example: the `ls` command, used to list the contents of a directory. The command can be used with or without options. Entering the plain `ls` command shows the contents of the current directory:

Figure 14.2 *The ls Command*



```
Session Edit View Bookmarks Settings Help
tux@knox:~> ls
bin Desktop Documents public_html tux.txt
tux@knox:~> █
```

The screenshot shows a terminal window with a menu bar (Session, Edit, View, Bookmarks, Settings, Help) and a title bar (Shell). The terminal content shows the prompt `tux@knox:~>`, the command `ls`, and the output listing: `bin Desktop Documents public_html tux.txt`. The directories are colored blue and the file is black. A cursor is visible on the line following the command.

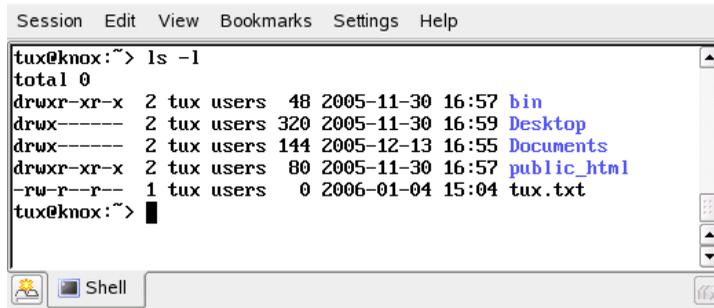
Unlike in other operating systems, files in Linux may have a file extension, such as `.txt`, but do not need to have one. This makes it difficult to differentiate between files and folders in this output of the `ls`. By default, the colors can give you a hint: directories are usually shown in blue, files in black.

Using Commands with Options

A better way to get more details about the contents of a directory is using the `ls` command with a string of options. Options modify the way a command works so that you can get it to do specific tasks. Options are separated from the command with a blank

and are prefixed with a hyphen. The `ls -l` command shows the contents of the same directory in full detail (long listing format):

Figure 14.3 *The `ls -l` Command*



```
tux@knox:~> ls -l
total 0
drwxr-xr-x 2 tux users 48 2005-11-30 16:57 bin
drwx----- 2 tux users 320 2005-11-30 16:59 Desktop
drwx----- 2 tux users 144 2005-12-13 16:55 Documents
drwxr-xr-x 2 tux users 80 2005-11-30 16:57 public_html
-rw-r--r-- 1 tux users 0 2006-01-04 15:04 tux.txt
tux@knox:~>
```

On the left of each object name, information about the object is shown in several columns. The most important are the following: The first column shows the file type of the object (in this example, `d` for directory or `-` for normal files). The next nine columns show the user permissions for the object. Columns 11 and 12 show the name of the file owner and the group (in this case, `tux` and `users`). Find information about user permissions and the user concept of Linux in [Section 14.2, “Users and Access Permissions”](#) (page 346). The next column shows the file size in bytes. Then date and time of the last change are displayed. The last column shows the object name.

If you want to see even more, you can combine two options for the `ls` command and enter `ls -la`. The shell now also shows hidden files in the directory, indicated by a dot in front (for example, `.hiddenfile`).

Getting Help

Nobody is expected to know all options of all commands by heart. If you remember the command name but are not sure about the options, you can enter the command followed by a blank and `--help`. This `--help` option exists for many commands. Entering `ls --help` displays all the options for the `ls` command.

14.1.2 Linux Directory Structure

Because the shell does not offer a graphical overview of directories and files like the tree view in a file manager, it is useful to have some basic knowledge of the default directory structure in 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.

[Figure 14.4](#) shows the standard directory tree in Linux, with the home directories of the example users `xyz`, `linux`, and `tux`. The `/home` directory contains the directories in which the individual users can store their personal files.

NOTE: Home Directory in a Network Environment

If you are working in a network environment, your home directory may not be called `/home`. It can be mapped to any directory in the file system.

The following list provides a brief description of the standard directories in Linux.

Figure 14.4 Excerpt from a Standard Directory Tree

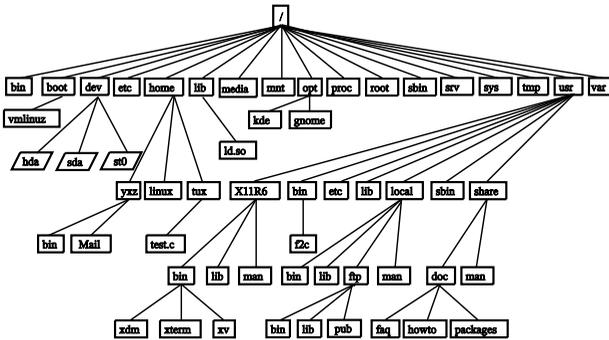


Table 14.1 Overview of a Standard Directory Tree

/	Root directory, starting point of the directory tree
/home	Personal directories of users
/etc	Important files for system configuration
/bin, /sbin	Programs needed early in the boot process (/bin) and for the administrator (/sbin)
/usr, /usr/local	All application programs and local, distribution-independent extensions (/usr/local)
/usr/bin, /usr/sbin	Generally accessible programs (/usr/bin) and reserved for the system administrator (/usr/sbin)

<code>/usr/share/doc</code>	Various documentation files
<code>/tmp, /var/tmp</code>	Temporary files (do not save files in this directory unless you do not need them)
<code>/opt</code>	Optional software, larger add-on program packages (such as KDE, GNOME, and Netscape)

14.1.3 Working with Directories and Files

To address a certain file or directory, you must specify the path leading to that directory or file. There are two ways to specify a path:

- The entire (absolute) path from the root directory to the respective file
- A path starting from the current directory (relative path)

Absolute paths always start with a slash. Relative paths do not have a slash at the beginning.

NOTE: Linux Is Case-Sensitive

Linux distinguishes between uppercase and lowercase in the file system. For example, entering `test.txt` or `Test.txt` makes a difference in Linux. Keep this in mind when entering filenames or paths.

To change directories, use the `cd` command.

- To switch to your home directory, enter `cd`.
- Refer to the current directory with a dot (`.`). This is mainly useful for other commands (`cp`, `mv`, ...).
- The next higher level in the tree is represented by two dots (`..`). For example, to switch to the parent directory of your current directory, enter `cd ..`

Examples of Addressing a File

The `cd` commands in [Section 14.1.3, “Working with Directories and Files”](#) (page 339) used relative paths. You can use also absolute paths. For example, suppose you want to copy a file from your home directory to a subdirectory of `/tmp`:

- 1 First, from your home directory create a subdirectory in `/tmp`:
 - a If your current directory is not your home directory, enter `cd ~` to switch to it. From anywhere in the file system, you can reach your home directory by entering `cd ~`.
 - b In your home directory, enter `mkdir /tmp/test`. `mkdir` stands for “make directory”. This command creates a new directory named `test` in the `/tmp` directory. In this case, use an absolute path to create the directory.
 - c To check what happened, now enter `ls -l /tmp`. The new directory `test` should appear in the list of contents of the `/tmp` directory.
- 2 Now create a new file in your home directory and copy it to the `/tmp/test` directory by using a relative path.
 - a Enter `touch myfile.txt`. The `touch` command with the `myfile.txt` option creates a new, empty file named `myfile.txt` in your current directory.
 - b Check this by entering `ls -l`. The new file should appear in the list of contents.
 - c Enter `cp myfile.txt ../tmp/test`. This copies `myfile.txt` to the directory `/tmp/test` without changing the name of the file.
 - d Check this by entering `ls -l /tmp/test`. The file `myfile.txt` should appear in the list of contents for `/tmp/test`.

To list the contents of home directories of other users, enter `ls ~username`. In the example directory tree in [Figure 14.4, “Excerpt from a Standard Directory Tree”](#) (page 338), one of the sample users is `tux`. In this case, `ls ~tux` would list the contents of the home directory of `tux`.

NOTE: Handling Blanks in Filenames or Directory Names

If a filename contains a space, either escape the space using a back slash (\) in front of the blank 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.

14.1.4 Useful Features of the Shell

Entering commands in Bash can include a lot of typing. In the following, get to know some features of the Bash that can make your work a lot easier and save a lot of typing.

History and Completion

By default, Bash “remembers” commands you have entered. This feature is called *history*. To repeat a command that has been entered before, press `↑` until the desired command appears at the prompt. Press `↓` to move forward through the list of previously entered commands. Use `Ctrl + R` to search in the history.

You can edit the selected command, for example, changing the name of a file, before you execute the command by pressing `Enter`. To edit the command line, just move the cursor to the desired position using the arrow keys and start typing.

Completing a filename or directory name to its full length after typing its first letters is another helpful feature of Bash. To do so, type the first letters then press `→`. If the filename or path can be uniquely identified, it is completed at once and the cursor moves to the end of the filename. You can then enter the next option of the command, if necessary. If the filename or path cannot be uniquely identified (because there are several filenames starting with the same letters), the filename or path is only completed up to the point where again several options are possible. You can then obtain a list of them by pressing `→` a second time. After this, you can enter the next letters of the file or path then try completion again by pressing `→`. When completing filenames and paths with the help of `→`, you can simultaneously check whether the file or path you want to enter really exists (and you can be sure of getting the spelling right).

Wild Cards

Another convenience offered by the shell is wild cards for pathname expansion. Wild cards are characters that can stand for other characters. 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`.
- The command `ls Testfile*` lists the files `Testfile1` and `Testfile2`.
- With `ls Test*`, the list also includes `Testfile`.
- The command `ls *fil*` shows all the sample files.
- 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*.

Viewing Files with Less and More

Linux includes two small programs for viewing text files directly in the shell: `less` and `more`. 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 “Redirection and Pipes”](#) (page 343).

Redirection and Pipes

Normally, the standard output in the shell is your screen or the console window and the standard input is the keyboard. However, the shell provides functions by which you can redirect the input or the output to another object, such as a file or another command. With the help of the symbols `>` and `<`, for example, you can forward the output of a command to a file (output redirection) or use a file as input for a command (input redirection). For example, if you want to write the output of a command such as `ls` to a file, enter `ls -l > file.txt`. This creates a file named `file.txt` that contains the list of contents of your current directory as generated by the `ls` command. However, if a file named `file.txt` already exists, this command overwrites the existing file. To prevent this, use `>>`. Entering `ls -l >> file.txt` simply appends the output of the `ls` command to an already existing file named `file.txt`. If the file does not exist, it is created.

Sometimes it is also useful to use a file as the input for a command. For example, with the `tr` command, you can replace characters redirected from a file and write the result to the standard output, your screen. Suppose you want to replace all characters `t` of your `file.txt` from the example above with `x` and print this to your screen. Do so by entering `tr t x < file.txt`.

Just like the standard output, the standard error output is sent to the console. To redirect the standard error output to a file named `errors`, append `2> errors` to the corre-

sponding command. Both standard output and standard error are saved to one file named `alloutput` if you append `>& alloutput`.

Using *pipelines* or *pipes* is also a sort redirection, although the use of the pipe is not constrained to files. With a pipe (`|`), you can combine several commands, using the output of one command as input for the next command. For example, to view the contents of your current directory in `less`, enter `ls | 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`.

14.1.5 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`, do the following:

- 1 Open a shell.

- 2 Use `cd` to your home directory where the `test` directory is located.
- 3 Enter `tar -cvf testarchive.tar test`. The `-c` option creates the archive, making it a file as directed by `-f`. The `-v` option lists the files as they are processed.
- 4 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.

14.1.6 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 14.3, “Important Linux Commands”](#) (page 349), find a list of the most important commands and a brief description of their functions.

14.2 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 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.

14.2.1 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 14.1, “Sample Output Showing File Permissions”](#) (page 346).

Example 14.1 *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).

Directory Permissions

Access permissions for directories have the type d. For directories, the individual permissions have a slightly different meaning.

Example 14.2 *Sample Output Showing Directory Permissions*

```
drwxrwxr-x 1 tux project3 35 Jun 21 15:15 ProjectData
```

In [Example 14.2, “Sample Output Showing Directory Permissions”](#) (page 347), 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 346), 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**).

14.2.2 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 14.2, “Sample Output Showing Directory Permissions”](#) (page 347) 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 create a new file in the directory 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 14.2, “Sample Output Showing Directory Permissions”](#) (page 347) 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 14.1, “Sample Output Showing File Permissions”](#) (page 346) 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.

14.3 Important Linux Commands

This section gives insight into the most important commands of your SUSE® Linux Enterprise 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`.

14.3.1 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 14.1.5, “Archives and Data Compression”](#) (page 344).

`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

`file [options] [files]`

With `file`, detect the contents of the specified files.

-Z

Tries to look inside compressed files

`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

Specify 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.

14.3.2 System Commands

The following section lists a few of the most important commands needed for retrieving system information and controlling processes and the network.

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 18.1.6, “The free Command”](#) (page 412) 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 imme-

ciate 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.

14.3.3 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.

14.4 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.

14.4.1 Operating Modes

NOTE: Display of Keys

In the following, find several commands that you can enter in *vi* by just pressing keys. These appear in uppercase as on a keyboard. If you need to enter a key in uppercase, this is stated explicitly by showing a key combination including the `[Shift]` key.

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 Shift+Z`. To exit the program saving all changes using the *extended* mode, enter `:-W-Q`. In *extended* mode, `w` stands for write and `q` for quit.

14.4.2 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 `5L` to move the cursor five characters to the right.

A selection of important commands is shown in [Table 14.2, “Simple Commands of the vi Editor”](#) (page 363) This list is far from complete. More complete lists are available in the documentation found in [Section 14.4.3, “For More Information”](#) (page 364)

Table 14.2 *Simple Commands of the vi Editor*

<code>Esc</code>	Change to command mode
<code>I</code>	Change to insert mode (characters appear at the current cursor position)
<code>A</code>	Change to insert mode (characters are inserted after the current cursor position)
<code>Shift</code> + <code>A</code>	Change to insert mode (characters are added at the end of the line)
<code>Shift</code> + <code>R</code>	Change to replace mode (overwrite the old text)
<code>R</code>	Replace the character under the cursor
<code>O</code>	Change to insert mode (a new line is inserted after the current one)
<code>Shift</code> + <code>O</code>	Change to insert mode (a new line is inserted before the current one)
<code>X</code>	Delete the current character
<code>D</code> - <code>D</code>	Delete the current line
<code>D</code> - <code>W</code>	Delete up to the end of the current word
<code>C</code> - <code>W</code>	Change to insert mode (the rest of the current word is overwritten by the next entries you make)
<code>U</code>	Undo the last command
<code>Ctrl</code> + <code>R</code>	Redo the change that was undone
<code>Shift</code> + <code>J</code>	Join the following line with the current one
<code>.</code>	Repeat the last command

14.4.3 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 Enterprise 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/>.

Part III. System

32-Bit and 64-Bit Applications in a 64-Bit System Environment

15

SUSE® Linux Enterprise 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 Enterprise 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 Enterprise 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 Enterprise 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.

15.1 Runtime Support

IMPORTANT: Conflicts between Application Versions

If an application is available both for 32-bit and 64-bit environments, 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.

Subdirectories of 32-bit `/lib` directories whose data content does not depend on the word size are not moved. For example, the X11 fonts are still found in the usual location under `/usr/X11R6/lib/X11/fonts`. This scheme conforms to LSB (Linux Standards Base) and FHS (File System Hierarchy Standard).

15.2 Software Development

A biarch development tool chain allows generation 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 Linux Enterprise environment is designed according to this principle. In the case of manually updated libraries, resolve these issues yourself.

15.3 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 `x86_64` system with `x86` as the second architecture.

1 Use the 32-bit compiler:

```
CC="gcc -m32"
```

2 Instruct the linker to process 32-bit objects (always use `gcc` as the linker front-end):

```
LD="gcc -m32"
```

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 -m32" \
LDFLAGS="-L/usr/lib;" \
    .configure \
    --prefix=/usr \
    --libdir=/usr/lib
```

```
make
make install
```

15.4 Kernel Specifications

The 64-bit kernels for `x86_64` 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 all the APIs used by system programs. This depends on the platform. For this reason, a small number of applications, like `lspci`, must be compiled

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 Novell 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.

Booting and Configuring a Linux System

16

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.

16.1 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 boot 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 17, *The Boot Loader*](#) (page 387).

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 initramfs can be used by the kernel directly. initramfs 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 16.1.1, “initramfs”](#) (page 372).
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 with `udev`. 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 16.1.2, “init on initramfs”](#) (page 373). Find more information about `udev` in [Chapter 20, *Dynamic Kernel Device Management with udev*](#) (page 445).
5. **init** `init` handles the actual booting of the system through several different levels providing different functionality. `init` is described in [Section 16.2, “The init Process”](#) (page 375).

16.1.1 initramfs

initramfs is a small `cpio` archive 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 root file system can be mounted and the 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`. After the modules are loaded, `udev` provides the initramfs with the needed devices. Later in the boot process, after

changing the root file system, it is necessary to regenerate the devices. This is done by `init`.

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 its predecessor, `initrd`—by calling `mkinitrd`. Calling `mkinitrd` without any argument creates an `initramfs`. Calling `mkinitrd -R` creates an `initrd`. In SUSE® Linux Enterprise, 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 only important if you rely on the correct setting of the device files `/dev/sd?`. However, in current systems you also may use the device files below `/dev/disk/` that are sorted in several subdirectories, named `by-id`, `by-path` and `by-uuid`, and always represent the same disk.

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.

16.1.2 `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 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.

Providing Special Block Files

For each loaded module, the kernel generates device events. `udev` handles these events and generates the needed device special files on a RAM file system in `/dev`. Without those special files, the file system would not be accessible.

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 system later. Find information about RAID in [Section 6.2, “Soft RAID Configuration”](#) (page 119). Find information about LVM in [Section 6.1, “LVM Configuration”](#) (page 113).

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 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 16.1.1, “initramfs”](#) (page 372), 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. The names of the modules needed for the boot process are written to `INITRD_MODULES` in `/etc/sysconfig/kernel`. These names are used to generate a custom `initramfs` that is needed to boot the system. If the modules are not needed for boot but for coldplug, the modules are written to `/etc/sysconfig/hardware/hwconfig-*`. All devices that are described with configuration files in this directory are initialized in the boot process.

Loading the Installation System or Rescue System

As soon as the hardware has been properly recognized, the appropriate drivers have been loaded, and `udev` has created the device special files, `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.

16.2 The init Process

The program `init` is the process with process ID 1. It is responsible for initializing the system in the required way. `init` 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 16.2.1, “Runlevels”](#) (page 375)). 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 16.2.2, “Init Scripts”](#) (page 378)).

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.

16.2.1 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 16.1, “Available Runlevels”](#) (page 375). 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 16.1 *Available Runlevels*

Runlevel	Description
0	System halt
S	Single user mode; from the boot prompt, only with US keyboard mapping
1	Single user mode

Runlevel	Description
2	Local multiuser mode without remote network (NFS, etc.)
3	Full multiuser mode with network
4	Not used
5	Full multiuser mode with network and X display manager—KDM, GDM, or XDM
6	System reboot

IMPORTANT: Avoid Runlevel 2 with a Partition Mounted via NFS

You should not use runlevel 2 if your system mounts a partition like `/usr` via NFS. The system might behave unexpectedly if program files or libraries are missing because the NFS service is not available in runlevel 2 (local multiuser mode without remote network).

To change runlevels while the system is running, enter `telinit` 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.

```
telinit 1 or shutdown now
```

The system changes to *single user mode*. This mode is used for system maintenance and administration tasks.

```
telinit 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.

```
telinit 5
```

The graphical environment is enabled. Usually a display manager like XDM, GDM, or KDM is started. If `autologin` is enabled, the local user is logged in to the preselected window manager (GNOME or KDE or any other window manager).

```
telinit 0 or shutdown -h now
    The system halts.
```

```
telinit 6 or shutdown -r now
    The system halts then reboots.
```

Runlevel 5 is the default runlevel in all SUSE Linux Enterprise standard installations. Users are prompted for login with a graphical interface or the default user is logged in automatically. If the default runlevel is 3, the X Window System must be configured properly, as described in [Chapter 22, *The X Window System*](#) (page 465), before the runlevel can be switched to 5. If this is done, check whether the system works in the desired way by entering `telinit 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`. Always let `init` reread `/etc/inittab` with the command `telinit q` before rebooting the machine.

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 `telinit 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 the stop scripts of the current runlevel 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. The same functionality may be achieved with the command `telinit q`.

16.2.2 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 that are run at boot time are called through symbolic links from `/etc/init.d/boot.d`. Scripts for changing the runlevel 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 16.2, “Possible init Script Options”](#) (page 378). Scripts that are run directly by `init` do not have these links. They are run independently from the runlevel when needed.

Table 16.2 *Possible init Script Options*

Option	Description
<code>start</code>	Start service.

Option	Description
<code>stop</code>	Stop service.
<code>restart</code>	If the service is running, stop it then restart it. If it is not running, start it.
<code>reload</code>	Reload the configuration without stopping and restarting the service.
<code>force-reload</code>	Reload the configuration if the service supports this. Otherwise, do the same as if <code>restart</code> had been given.
<code>status</code>	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.

All of these settings may also be changed with the help of the YaST module. If you need to check the status on the command line, use the tool `chkconfig`, described in the `chkconfig(8)` man page.

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 these 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`, `chkconfig`, 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. Find some useful information about init scripts in [Section 16.2.1, “Runlevels”](#) (page 375).

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 must be edited. See [Example 16.1, “A Minimal INIT INFO Block”](#) (page 381).

Example 16.1 *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 16.2.3, “Configuring System Services \(Runlevel\) with YaST”](#) (page 382).

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

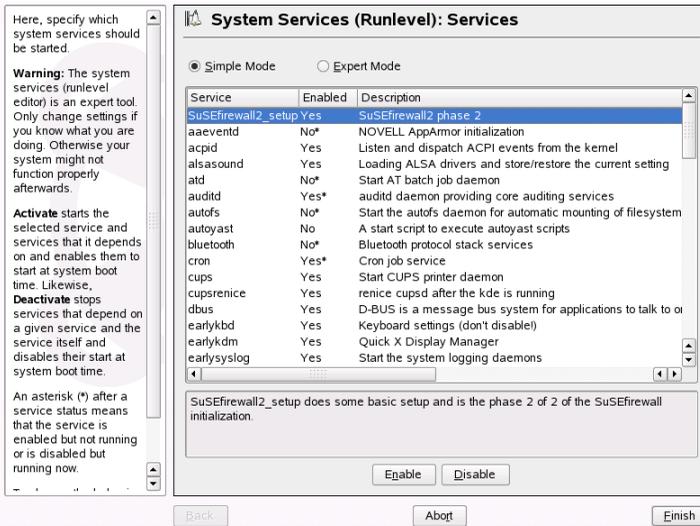
by enabling the corresponding service in the runlevel editor of YaST. Your changes are applied during the next reboot—the new service is started automatically.

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. The manually-added service will be removed with the next run of `insserv`.

16.2.3 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 16.1 *System Services (Runlevel)*



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 Enterprise 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 16.1, “Available Runlevels”](#) (page 375)) 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 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.

16.3 System Configuration via `/etc/sysconfig`

The main configuration of SUSE Linux Enterprise 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 allows most configurations to be made in one central place without fiddling with different configuration files at different places of the operating system.

There are two ways to edit the system configuration. Either use the YaST `sysconfig` Editor or edit the configuration files manually.

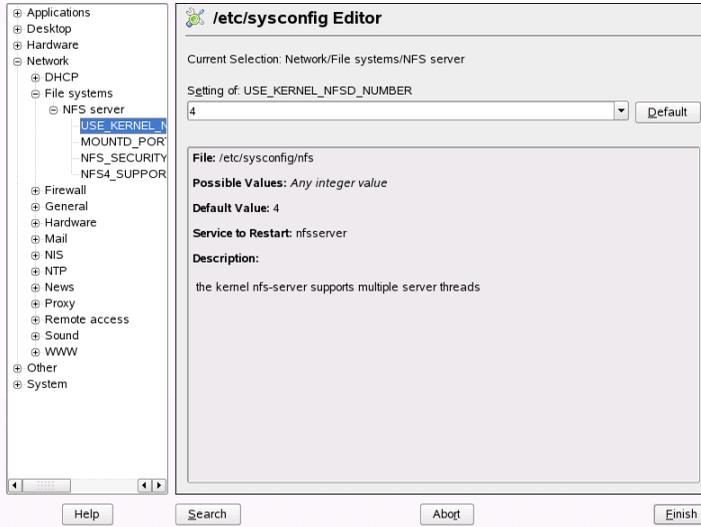
16.3.1 Changing the System Configuration Using the YaST `sysconfig` Editor

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 16.2 System Configuration Using the *sysconfig* Editor



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.

16.3.2 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 may 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.

The Boot Loader

This chapter describes how to configure GRUB, the boot loader used in SUSE® Linux Enterprise. 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 16, *Booting and Configuring a Linux System*](#) (page 371). A boot loader represents the interface between machine (BIOS) and the operating system (SUSE Linux Enterprise). 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 part of a boot loader program or an operating system selector. The next 64 bytes provide space for a partition table of up to four entries (see [Section “Partition Types”](#) (page 40)). The partition table contains information about the partitioning of the hard disk and the file system types. The operating system needs this table for handling the hard disk. With conventional generic code in the MBR, exactly one partition must be marked *active*. 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 some BIOSs, so is not considered for booting.

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 initially empty after setting up a file system other than XFS. 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).

17.1 Selecting a Boot Loader

By default, the boot loader GRUB is used in SUSE Linux Enterprise. However, in some cases and for special hardware and software constellations, LILO may be necessary. If you update from an older SUSE Linux Enterprise 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`.

17.2 Booting with GRUB

GRUB (Grand Unified Bootloader) comprises two stages. stage1 consists of 512 bytes and its only task is to load the second stage of the boot loader. Subsequently, stage2 is loaded. This stage contains the main part of the boot loader.

In some configurations, an intermediate stage 1.5 can be used, which locates and loads stage 2 from an appropriate file system. If possible, this method is chosen by default on installation or when initially setting up GRUB with YaST.

stage2 is able to access many 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 GRUB command line prompts the user for how to proceed (see [Section “Editing Menu Entries during the Boot Procedure”](#) (page 393) for details).

`/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 commands, 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 393)). 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.

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*. It provides an emulation of GRUB in the installed system and can be used to install GRUB or test new settings before applying them. The functionality to install GRUB as the 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.

17.2.1 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 17.3, “Configuring the Boot Loader with YaST”](#) (page 397).

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 391). 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 its command line.

If the kernel does not have built-in drivers for access to the root partition or a recent Linux system with advanced hotplug features is used, `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 into the loaded kernel image, the command `initrd` must follow after the `kernel` command.

The command `root` simplifies the specification of kernel and `initrd` files. The only argument of `root` is a device or a partition. This device is used for all kernel, `initrd`, or other file paths for which no device is explicitly specified until the next `root` command.

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 392).

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. It more closely resembles the simple disk enumeration the BIOS does and the syntax is similar to that used in some BSD derivatives. 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
```

Being dependent on BIOS devices, GRUB does not distinguish between IDE, SATA, SCSI, and hardware RAID devices. All hard disks recognized by the BIOS or other controllers are numbered according to the boot sequence preset in the BIOS.

Unfortunately, it is often not possible 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 17.2.2, “The File `device.map`”](#) (page 394).

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 the top directory of the `/dev/hda5` partition.

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 Enterprise. 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 393).

Editing Menu Entries during the Boot Procedure

In the graphical 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 and get to

the GRUB text-based menu 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. See [Figure 45.1, “US Keyboard Layout”](#) (page 782) for a figure.

Editing menu entries 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.

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.

17.2.2 The File `device.map`

The file `device.map` maps GRUB and BIOS 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 may 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 prompt 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.

IMPORTANT: SATA Disks

Depending on the controller, SATA disks are either recognized as IDE (`/dev/hd x`) or SCSI (`/dev/sd x`) devices.

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
```

17.2.3 The File `/etc/grub.conf`

The third most important GRUB configuration file after `menu.lst` and `device.map` is `/etc/grub.conf`. This file contains the commands, parameters, and options the GRUB shell needs for installing the boot loader correctly:

```
root (hd0,4)
  install /grub/stage1 (hd0,3) /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 the extended partition container

(`/grub/stage1 (hd0,3)`). This is a slightly esoteric configuration, but it is known to work in many cases. `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.

17.2.4 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, encrypt the password using `grub-md5-crypt`:

```
# grub-md5-crypt
Password: ****
Retype password: ****
Encrypted: $1$1S2dv/$JOYcdxIn7CJk9xShzzJVw/
```

- 2 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$1S2dv/$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.

- 3 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:

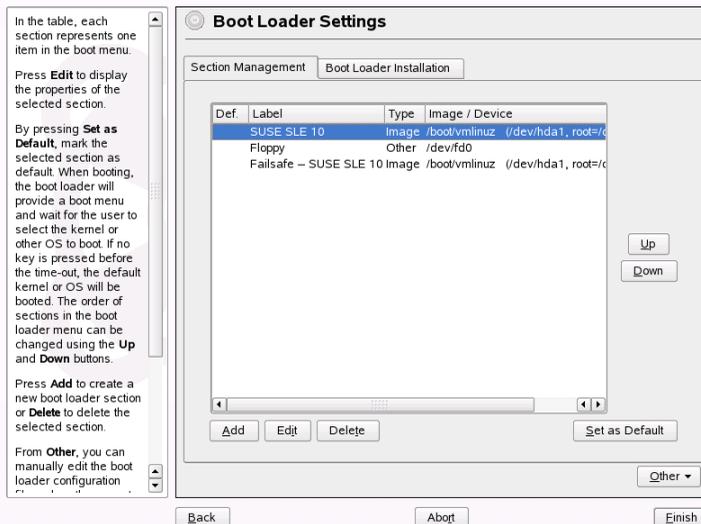
```
Error 32: Must be authenticated
```

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.

17.3 Configuring the Boot Loader with YaST

The easiest way to configure the boot loader in your SUSE Linux Enterprise system is to use the YaST module. In the YaST Control Center, select *System* → *Boot Loader Configuration*. As in [Figure 17.1](#), “Configuring the Boot Loader with YaST” (page 397), this shows the current boot loader configuration of your system and allows you to make changes.

Figure 17.1 *Configuring the Boot Loader with YaST*



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 17.2, “Booting with GRUB”](#) (page 388) first.

Use the *Boot Loader Installation* tab to view and change settings related to type, location, and advanced loader settings.

17.3.1 Boot Loader Type

Set the boot loader type in *Boot Loader Installation*. The default boot loader in SUSE Linux Enterprise is GRUB. To use LILO, proceed as follows:

Procedure 17.1 *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 Enterprise.
 - Read Configuration Saved on Disk
Load your own `/etc/lilo.conf`. This action is not available during the installation of SUSE Linux Enterprise.
- 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.

17.3.2 Boot Loader Location

To change the location of the boot loader, follow these steps:

Procedure 17.2 *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. 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. Unless a /boot partition is necessary or the MBR needs to be used, this is the preferred default.

Other

Use this option to specify the location of the boot loader manually.

- 2 Click *Finish* to apply your changes.

17.3.3 Default System

To change the system that is booted by default, proceed as follows:

Procedure 17.3 *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.

17.3.4 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 17.4 *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*.

17.3.5 Security Settings

Using this YaST module, you can also set a password to protect booting. This gives you an additional level of security.

Procedure 17.5 *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.

17.3.6 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 17.2.2, “The File device.map”](#) (page 394)). To do so, proceed as follows:

Procedure 17.6 *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 Generic 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.

17.4 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.

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*.

17.5 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.

Procedure 17.7 *Creating Boot CDs*

- 1 Create a directory in which to create the ISO image, for example:

```
cd /tmp
mkdir iso
```

- 2 Create a subdirectory for GRUB:

```
mkdir -p iso/boot/grub
```

- 3 Copy the kernel, the files `stage2_eltorito`, `initrd`, `menu.lst`, and `/boot/message` to `iso/boot/`:

```
cp /boot/vmlinuz iso/boot/  
cp /boot/initrd iso/boot/  
cp /boot/message iso/boot/  
cp /boot/grub/menu.lst iso/boot/grub
```

- 4 Adjust the path entries in `iso/boot/menu.lst` 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
```

- 5 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
```

- 6 Write the resulting file `grub.iso` to a CD using your preferred utility.

17.6 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 17, *The Boot Loader*](#) (page 387) 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.

17.7 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`.

17.8 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 Enterprise

18

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).

18.1 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.

18.1.1 The Package `bash` and `/etc/profile`

Bash is the default shell in SUSE® Linux Enterprise. 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.

18.1.2 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 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 username to run the command directly after the time table and before the command. In [Example 18.1, “Entry in /etc/crontab”](#) (page 408), `root` is entered. Package-specific tables, located in `/etc/cron.d`, have the same format. See the `cron` man page (`man cron`).

Example 18.1 *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 execution is 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 scripts at custom times, remove the time stamp files regularly using `/etc/crontab` entries (see [Example 18.2, “/etc/crontab: Remove Time Stamp Files”](#) (page 409), which removes the `hourly` one before every full hour, the `daily` one once a day at 2:14 a.m., etc.).

Example 18.2 */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`.

18.1.3 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.

Configure `logrotate` with the file `/etc/logrotate.conf`. In particular, the `include` specification primarily configures the additional files to read. SUSE® Linux Enterprise ensures that programs that produce log files install individual configuration 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 18.3 *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.

18.1.4 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.

18.1.5 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 18.1, “ulimit: Setting Resources for the User”](#) (page 411).

Table 18.1 *ulimit: Setting Resources for the User*

<code>-m</code>	Maximum size of physical memory
<code>-v</code>	Maximum size of virtual memory
<code>-s</code>	Maximum size of the stack
<code>-c</code>	Maximum size of the core files
<code>-a</code>	Display of limits set

Systemwide entries 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 18.4 *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.

18.1.6 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`.

18.1.7 The File `/etc/resolv.conf`

Domain name resolution is handled through the file `/etc/resolv.conf`.

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.

18.1.8 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 `tinfo`, `xinfo`, or the SUSE help system to view info pages.

18.1.9 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 Enterprise, 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.

18.2 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]`.

18.3 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 Enterprise 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>.

18.4 Language and Country-Specific Settings

SUSE® Linux Enterprise 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 variables can be set with the YaST `sysconfig` editor (see [Section 16.3.1, “Changing the System Configuration Using the YaST sysconfig Editor”](#) (page 384)). 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>]
```

18.4.1 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.

18.4.2 Locale Settings in `~/ .i18n`

If you are not satisfied with locale system defaults, change the settings in `~/ .i18n`. Entries in `~/ .i18n` override system defaults from `/etc/sysconfig/language`. Use the same variable names but without the `RC_` namespace prefixes, for example, use `LANG` instead of `RC_LANG`.

18.4.3 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.

18.4.4 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

CUPS is the standard print system in SUSE® Linux Enterprise. 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 Enterprise 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 19.7.1, “Printers without Standard Printer Language Support”](#) (page 437) 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://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 Enterprise version was released. Thus, the databases may not necessarily indicate the correct status, but only provide an approximation.

19.1 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.

19.2 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://en.opensuse.org/SDB:CUPS_in_a_Nutshell.

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.

19.3 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 Enterprise, 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 19.6.3, “PPD Files in Various Packages”](#) (page 434) and [Section 19.7.2, “No Suitable PPD File Available for a PostScript Printer”](#) (page 437).

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 425)). 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 Enterprise 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.

19.4 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 Enterprise. Because SUSE Linux Enterprise puts great emphasis on security, third-party tools often have difficulties with the security restrictions and cause more complications than benefits. See [Section 19.6.1, “CUPS Server and Firewall”](#) (page 431) and [Section 19.6.2, “Changes in the CUPS Print Service”](#) (page 432) for more information about troubleshooting.

19.4.1 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*).

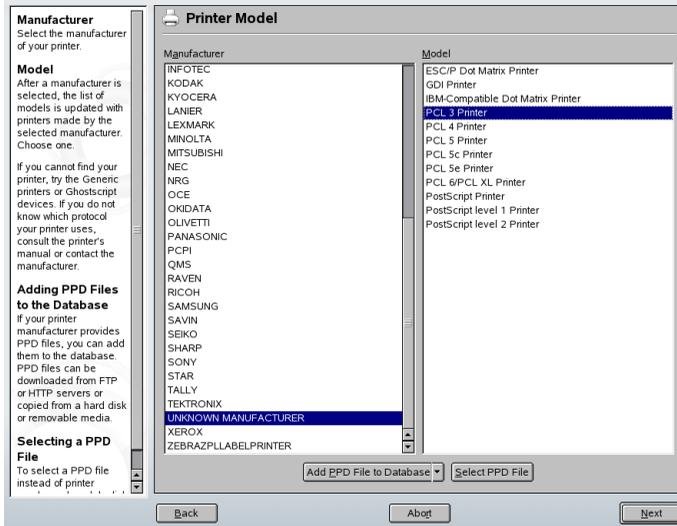
Printer Model and PPD File

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 19.3, “Installing the Software”](#) (page 423) 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 19.1 *Selecting the Printer Model*



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.

19.4.2 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 19.6.2, “Changes in the CUPS Print Service”](#) (page 432)). For guidelines for installation of CUPS in the network, read the article *CUPS in a Nutshell* in the Support Database at http://en.opensuse.org/SDB:CUPS_in_a_Nutshell.

Start the printer configuration then click *Add*. If not told otherwise by the network administrator 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-levell.ppd.gz -E
```

For more options of `lpadmin`, see the `lpadmin(1)` man page.

During printer setup, 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
```

Settings are written to `~/.lpoptions` when a normal user runs `lpoptions`. root settings are written to `/etc/cups/lpoptions`.

19.5 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.

19.6 Special Features in SUSE Linux Enterprise

A number of CUPS features have been adapted for SUSE Linux Enterprise. Some of the most important changes are covered here.

19.6.1 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 (forwarding queue). 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.

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 either scanning local network hosts to see if they offer the IPP service or by listening to IPP broadcasts. This requires the firewall to let incoming packets on port 631/UDP (service IPP client) pass through. This is automatically enabled when you have configured your machine to be in the internal firewall zone. Opening a port to configure access to remote queues in the external zone can be a security risk because an attacker could broadcast a server that might be accepted by users. By default IPP broadcasts are rejected in the external zone. See [Section 38.4.1, “Configuring the Firewall with YaST”](#) (page 714) for details on firewall configuration.

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.

19.6.2 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 CUPS administration Web front-end or the KDE printer administration tool.

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 `rc cups reload`. Use `rc cups 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 433) and [Section “Generalized Functionality for BrowseAllow and BrowseDeny”](#) (page 433) are vital preconditions for this feature, because otherwise the security would not be sufficient for an automatic activation of `cupsd`.

19.6.3 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 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 `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.

19.7 Troubleshooting

The following sections cover some of the most frequently encountered printer hardware and software problems and ways to solve or circumvent these problems.

19.7.1 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.

19.7.2 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.

19.7.3 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 19.1](#), “`/etc/modprobe.conf: Interrupt Mode for the First Parallel Port`” (page 438). 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 19.1 */etc/modprobe.conf: Interrupt Mode for the First Parallel Port*

```
alias parport_lowlevel parport_pc
options parport_pc io=0x378 irq=7
```

19.7.4 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 19.2, “Error Message from the lpd”](#) (page 439), the problem is caused by the remote lpd.

Example 19.2 *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 19.3, “Broadcast from the CUPS Network Server”](#) (page 439).

Example 19.3 *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 19.4.2, “Network Printers”](#) (page 427).

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.

19.7.5 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.

19.7.6 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 reenables printing with the command `/usr/bin/enable`.

19.7.7 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-jobnumber
```

19.7.8 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.

19.7.9 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.

Dynamic Kernel Device Management with udev

20

Since version 2.6, the kernel is capable of adding or removing almost any device in the running system. Changes in device state (whether a device is plugged in or removed) need to be propagated to userspace. Devices need to be configured as soon as they are plugged in and discovered. Users of a certain device need to be informed about any state changes of this device. `udev` provides the needed infrastructure to dynamically maintain the device node files and symbolic links in the `/dev` directory. `udev` rules provide a way to plug external tools into the kernel device event processing. This enables you to customize `udev` device handling, for example, by adding certain scripts to execute as part of kernel device handling, or request and import additional data to evaluate during device handling.

20.1 The `/dev` Directory

The device nodes in the `/dev` directory provide access to the corresponding kernel devices. With `udev`, the `/dev` directory reflects the current state of the kernel. Every kernel device has one corresponding device file. If a device is disconnected from the system, the device node is removed.

The content of the `/dev` directory is kept on a temporary file system and all files are created from scratch at every system start-up. Manually created or changed files intentionally do not survive a reboot. Static files and directories that should always be present in the `/dev` directory regardless of the state of the corresponding kernel device can be placed in the `/lib/udev/devices` directory. At system start-up, the contents of that directory is copied to the `/dev` directory with the same ownership and permissions as the files in `/lib/udev/devices`.

20.2 Kernel uevents and udev

The required device information is exported by the `sysfs` file system. For every device the kernel has detected and initialized, a directory with the device name is created. It contains attribute files with device-specific properties. Every time a device is added or removed, the kernel sends a `uevent` to notify `udev` of the change.

The `udev` daemon reads and parses all provided rules from the `/etc/udev/rules.d/*.rules` files once at start-up and keeps them in memory. If rules files are changed, added, or removed, the daemon receives an event and updates the in-memory representation of the rules.

Every received event is matched against the set of provided rules. The rules can add or change event environment keys, request a specific name for the device node to create, add symlinks pointing to the node, or add programs to run after the device node is created. The driver core `uevents` are received from a kernel `netlink` socket.

20.3 Drivers, Kernel Modules, and Devices

The kernel bus drivers probe for devices. For every detected device, the kernel creates an internal device structure and the driver core sends a `uevent` to the `udev` daemon. Bus devices identify themselves by a specially-formatted ID, which tells what kind of device it is. Usually these IDs consist of vendor and product ID and other subsystem-specific values. Every bus has its own scheme for these IDs, called `MODALIAS`. The kernel takes the device information, composes a `MODALIAS` ID string from it, and sends that string along with the event. For a USB mouse, it looks like this:

```
MODALIAS=usb:v046DpC03Ed2000dc00dsc00dp00ic03isc01ip02
```

Every device driver carries a list of known aliases for devices it can handle. The list is contained in the kernel module file itself. The program `depmod` reads the ID lists and creates the file `modules.alias` in the kernel's `/lib/modules` directory for all currently available modules. With this infrastructure, module loading is as easy as calling `modprobe` for every event that carries a `MODALIAS` key. If `modprobe $MODALIAS` is called, it matches the device alias composed for the device with the

aliases provided by the modules. If a matching entry is found, that module is loaded. All this is triggered by udev and happens automatically.

20.4 Booting and Initial Device Setup

All device events happening during the boot process before the udev daemon is running are lost, because the infrastructure to handle these events lives on the root file system and is not available at that time. To cover that loss, the kernel provides a `uevent` file for every device in the `sysfs` file system. By writing `add` to that file, the kernel resends the same event as the one lost during boot. A simple loop over all `uevent` files in `/sys` triggers all events again to create the device nodes and perform device setup.

As an example, a USB mouse present during boot may not be initialized by the early boot logic, because the driver is not available that time. The event for the device discovery was lost and failed to find a kernel module for the device. Instead of manually searching for possibly connected devices, udev just requests all device events from the kernel after the root file system is available, so the event for the USB mouse device just runs again. Now it finds the kernel module on the mounted root file system and the USB mouse can be initialized.

From userspace, there is no visible difference between a device coldplug sequence and a device discovery during runtime. In both cases, the same rules are used to match and the same configured programs are run.

20.5 Debugging udev Events

The program `udevmonitor` can be used to visualize the driver core events and the timing of the udev event processes.

```
UEVENT [1132632714.285362] add@/devices/pci0000:00/0000:00:1d.1/usb2/2-2
UEVENT [1132632714.288166] add@/devices/pci0000:00/0000:00:1d.1/usb2/2-2/2-2:1.0
UEVENT [1132632714.309485] add@/class/input/input6
UEVENT [1132632714.309511] add@/class/input/input6/mouse2
UEVENT [1132632714.309524] add@/class/usb_device/usbdev2.12
UDEV [1132632714.348966] add@/devices/pci0000:00/0000:00:1d.1/usb2/2-2
UDEV [1132632714.420947] add@/devices/pci0000:00/0000:00:1d.1/usb2/2-2/2-2:1.0
UDEV [1132632714.427298] add@/class/input/input6
UDEV [1132632714.434223] add@/class/usb_device/usbdev2.12
UDEV [1132632714.439934] add@/class/input/input6/mouse2
```

The `UEVENT` lines show the events the kernel has sent over netlink. The `UDEV` lines show the finished udev event handlers. The timing is printed in microseconds. The time between `UEVENT` and `UDEV` is the time udev took to process this event or the udev daemon has delayed its execution to synchronize this event with related and already running events. For example, events for hard disk partitions always wait for the main disk device event to finish, because the partition events may rely on the data the main disk event has queried from the hardware.

`udevmonitor --env` shows the complete event environment:

```
UDEV [1132633002.937243] add@/class/input/input7
UDEV_LOG=3
ACTION=add
DEVPATH=/class/input/input7
SUBSYSTEM=input
SEQNUM=1043
PHYSDEVPATH=/devices/pci0000:00/0000:00:1d.1/usb2/2-2/2-2:1.0
PHYSDEVBUS=usb
PHYSDEVDRIVER=usbhid
PRODUCT=3/46d/c03e/2000
NAME="Logitech USB-PS/2 Optical Mouse"
PHYS="usb-0000:00:1d.1-2/input0"
UNIQ=""
EV=7
KEY=70000 0 0 0 0 0 0 0 0
REL=103
```

udev also sends messages to syslog. The default syslog priority that controls which messages are sent to syslog is specified in the udev configuration file `/etc/udev/udev.conf`. The log priority of the running daemon can be changed with `udevcontrol log_priority=level/number`.

20.6 Influencing Kernel Device Event Handling with udev Rules

A udev rule can match any property the kernel adds to the event itself or any information that the kernel exports to `sysfs`. The rule can also request additional information from external programs. Every event is matched against all provided rules. All rules are located in the `/etc/udev/rules.d` directory.

Every line in the rules file contains at least one key value pair. There are two kinds of keys, match and assignment keys. If all match keys match their values, the rule is applied

and the assignment keys are assigned the specified value. A matching rule may specify the name of the device node, add symlinks pointing to the node, or run a specified program as part of the event handling. If no matching rule is found, the default device node name is used to create the device node. The rule syntax and the provided keys to match or import data are described in the udev man page.

20.7 Persistent Device Naming

The dynamic device directory and the udev rules infrastructure make it possible to provide stable names for all disk devices—regardless of their order of recognition or the connection used for the device. Every appropriate block device the kernel creates is examined by tools with special knowledge about certain buses, drive types, or file systems. Along with the dynamic kernel-provided device node name, udev maintains classes of persistent symbolic links pointing to the device:

```
/dev/disk
|-- by-id
|   |-- scsi-SATA_HTS726060M9AT00_MRH453M4HWHG7B -> ../../sda
|   |-- scsi-SATA_HTS726060M9AT00_MRH453M4HWHG7B-part1 -> ../../sda1
|   |-- scsi-SATA_HTS726060M9AT00_MRH453M4HWHG7B-part6 -> ../../sda6
|   |-- scsi-SATA_HTS726060M9AT00_MRH453M4HWHG7B-part7 -> ../../sda7
|   |-- usb-Generic_STORAGE_DEVICE_02773 -> ../../sdd
|   `-- usb-Generic_STORAGE_DEVICE_02773-part1 -> ../../sdd1
|-- by-label
|   |-- Photos -> ../../sdd1
|   |-- SUSE10 -> ../../sda7
|   `-- devel -> ../../sda6
|-- by-path
|   |-- pci-0000:00:1f.2-scsi-0:0:0:0 -> ../../sda
|   |-- pci-0000:00:1f.2-scsi-0:0:0:0-part1 -> ../../sda1
|   |-- pci-0000:00:1f.2-scsi-0:0:0:0-part6 -> ../../sda6
|   |-- pci-0000:00:1f.2-scsi-0:0:0:0-part7 -> ../../sda7
|   |-- pci-0000:00:1f.2-scsi-1:0:0:0 -> ../../sr0
|   |-- usb-02773:0:0:2 -> ../../sdd
|   |-- usb-02773:0:0:2-part1 -> ../../sdd1
`-- by-uuid
    |-- 159a47a4-e6e6-40be-a757-a629991479ae -> ../../sda7
    |-- 3e999973-00c9-4917-9442-b7633bd95b9e -> ../../sda6
    `-- 4210-8F8C -> ../../sdd1
```

20.8 The Replaced hotplug Package

The formerly used hotplug package is entirely replaced by udev and the udev-related kernel infrastructure. The following parts of the former hotplug infrastructure have been made obsolete or had their functionality taken over by udev:

`/etc/hotplug/*.agent`

No longer needed or moved to `/lib/udev`

`/etc/hotplug/*.rc`

Replaced by the `/sys/*/uevent` trigger

`/etc/hotplug/blacklist`

Replaced by the `blacklist` option in `modprobe.conf`

`/etc/dev.d/*`

Replaced by the udev rule `RUN` key

`/etc/hotplug.d/*`

Replaced by the udev rule `RUN` key

`/sbin/hotplug`

Replaced by `udev` listening to `netlink`; only used in the initial RAM file system until the root file system can be mounted, then it is disabled

`/dev/*`

Replaced by dynamic udev and static content in `/lib/udev/devices/*`

The following files and directories contain the crucial elements of the udev infrastructure:

`/etc/udev/udev.conf`

Main udev configuration file

`/etc/udev/rules.d/*`

udev event matching rules

`/lib/udev/devices/*`

Static `/dev` content

`/lib/udev/*`

Helper programs called from udev rules

20.9 For More Information

For more information about the udev infrastructure, refer to the following man pages:

`udev`

General information about udev, keys, rules, and other important configuration issues.

`udevinfo`

`udevinfo` can be used to query device information from the udev database.

`udev`

Information about the udev event managing daemon.

`udevmonitor`

`udevmonitor` prints the kernel and udev event sequence to the console. This tool is mainly used for debugging purposes.

File Systems in Linux

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 concepts, advantages, and fields of application. Some additional information about LFS (large file support) in Linux is also provided.

21.1 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, pointers to the disk blocks where the file contents are actually stored, and date and time of creation, modification, and access.

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.

21.2 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 replace 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 YaST.

21.2.1 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 version 6.4. ReiserFS was designed by Hans Reiser and the Namesys development team. It has proven itself to be a powerful alternative to 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 21.2.3, “Ext3”](#) (page 456). The default mode is `data=ordered`, which ensures both data and metadata integrity, but uses journaling only for metadata.

21.2.2 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.

21.2.3 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 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.

21.2.4 Converting an Ext2 File System into Ext3

To convert an Ext2 file system to Ext3, proceed as follows:

- 1 Create an Ext3 journal by running `tune2fs -j` as root. 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 the `tune2fs` manual page.

- 2 To ensure that the Ext3 file system is recognized as such, edit the file `/etc/fstab` as root, changing the file system type specified for the corresponding partition from `ext2` to `ext3`. The change takes effect after the next reboot.
- 3 To boot a root file system set up as an Ext3 partition, include the modules `ext3` and `jbd` in the `initrd`. To do this, edit `/etc/sysconfig/kernel` as root, adding `ext3` and `jbd` to the `INITRD_MODULES` variable. After saving the changes, run the `mkinitrd` command. This builds a new `initrd` and prepares it for use.

21.2.5 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 21.2.6, “XFS”](#) (page 458). Using delayed allocation even for metadata can result in better overall layout.

21.2.6 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.

21.3 Some Other Supported File Systems

Table 21.1, “File System Types in Linux” (page 460) 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 21.1 *File System Types in Linux*

<code>cramfs</code>	<i>Compressed ROM file system</i> : A compressed read-only file system for ROMs.
<code>hpfs</code>	<i>High Performance File System</i> : The IBM OS/2 standard file system—only supported in read-only mode.
<code>iso9660</code>	Standard file system on CD-ROMs.
<code>minix</code>	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.
<code>msdos</code>	<i>fat</i> , the file system originally used by DOS, is today used by various operating systems.
<code>ncpfs</code>	File system for mounting Novell volumes over networks.
<code>nfs</code>	<i>Network File System</i> : Here, data can be stored on any machine in a network and access may be granted via a network.
<code>smbfs</code>	<i>Server Message Block</i> is used by products such as Windows to enable file access over a network.
<code>sysv</code>	Used on SCO UNIX, Xenix, and Coherent (commercial UNIX systems for PCs).
<code>ufs</code>	Used by BSD, SunOS, and NeXTstep. Only supported in read-only mode.

umsdos	<i>UNIX on MSDOS</i> : Applied on top of a normal <code>fat</code> file system, achieves UNIX functionality (permissions, links, long filenames) by creating special files.
vfat	<i>Virtual FAT</i> : Extension of the <code>fat</code> file system (supports long filenames).
ntfs	<i>Windows NT file system</i> , read-only.

21.4 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 21.2, “Maximum Sizes of File Systems \(On-Disk Format\)”](#) (page 461) offers an overview of the current limitations of Linux files and file systems.

Table 21.2 *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^{43} -4096 (16 TB-4096 Bytes)
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 TB)	2^{45} (32 TB)

File System	File Size (Bytes)	File System Size (Bytes)
XFS	2^{63} (8 EB)	2^{63} (8 EB)
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 21.2, “Maximum Sizes of File Systems (On-Disk Format)” (page 461) 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 in size. However, this limit is still out of reach for the currently available hardware.

21.5 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://www.ibm.com/developerworks/linux/library/l-jfs.html>
- <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.

The X Window System

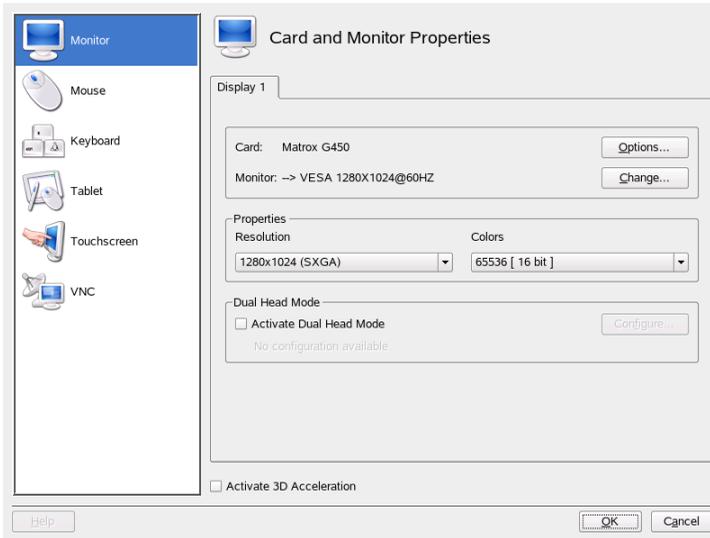
The X Window System (X11) is the de facto standard for graphical user interfaces in UNIX. X is network-based, enabling applications started on one host to be displayed on another host connected over any kind of network (LAN or Internet). This chapter describes the setup and optimization of the X Window System environment, provides background information about the use of fonts in SUSE® Linux Enterprise, and explains the configuration of OpenGL and 3D.

The following text contains several references to documentation that can be found below `/usr/share/doc/packages/Xorg` and `/usr/share/doc/howto/en`. This material along with respective manual pages is available only if the appropriate documentation packages are installed (`xorg-x11-doc`, `xorg-x11-man`, and `howtoenh`).

22.1 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 interface for the individual modules from the YaST control center.

Figure 22.1 *The Main Window of SaX2*



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 7, *System Configuration with YaST*](#) (page 125).

Monitor

For a description of the monitor and graphics card configuration, see [Section 7.13.1, “Card and Monitor Properties”](#) (page 176).

Mouse

For a description of the mouse configuration in the graphical environment, see [Section 7.13.2, “Mouse Properties”](#) (page 180).

Keyboard

For a description of the keyboard configuration in the graphical environment, see [Section 7.13.3, “Keyboard Properties”](#) (page 181).

Tablet

For a description of the graphics tablet configuration, see [Section 7.13.4, “Tablet Properties”](#) (page 181).

Touchscreen

For a description of the touchscreen configuration, see [Section 7.13.5, “Touchscreen Properties”](#) (page 182).

VNC

For a description of the VNC configuration, see [Section 7.13.6, “Remote Access Properties”](#) (page 182).

22.2 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 sections 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 22.1, “Sections in /etc/X11/xorg.conf”](#) (page 468).

Table 22.1 *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 <code>Driver</code> and the options defining the <code>Protocol</code> and <code>Device</code> .
Monitor	Describes the monitor used. The individual elements of this section are the name, which is referred to later in the <code>Screen</code> definition, the <code>bandwidth</code> , and the synchronization frequency limits (<code>HorizSync</code> and <code>VertRefresh</code>). 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 <code>SaX2</code> 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 files in <code>/usr/share/doc/howto/en/html/XFree86-Video-Timings-HOWTO</code> .

Type	Meaning
Device	This section defines a specific graphics card. It is referenced by its descriptive name.
Screen	This section puts together a <code>Monitor</code> and a <code>Device</code> to form all the necessary settings for <code>X.Org</code> . In the <code>Display</code> subsection, specify the size of the virtual screen (<code>Virtual</code>), the <code>ViewPort</code> , and the <code>Modes</code> used with this screen.
<code>ServerLayout</code>	This section defines the layout of a single or multihead configuration. This section binds the input devices <code>InputDevice</code> and the display devices <code>Screen</code> .

`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.

22.2.1 Screen Section

The screen section combines a monitor with a device section and determines the resolution and color depth to use. A screen section might resemble [Example 22.1, “Screen Section of the File `/etc/X11/xorg.conf`”](#) (page 470).

Example 22.1 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 `Identifier` line (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.

22.2.2 Device Section

A device section describes a specific graphics card. You can have as many device entries in `xorg.conf` as you like, provided 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 configuration, 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.

With 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`).

22.2.3 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.

22.3 Installing and Configuring Fonts

The installation of additional fonts in SUSE Linux Enterprise is very easy. Simply copy the fonts to any directory located in the X11 font path (see [Section 22.3.1, “X11 Core Fonts”](#) (page 473)). To enable use of the fonts, the installation directory should be a subdirectory of the directories configured in `/etc/fonts/fonts.conf` (see [Section 22.3.2, “Xft”](#) (page 474)).

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 22.3.3, “CID-Keyed Fonts”](#) (page 478).

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.

22.3.1 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. 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. Unicode fonts are also supported, but their use may be slow and require 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 Enterprise 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 Enterprise contain at least the glyphs needed for European languages (formerly encoded as `iso-8859-*`).

22.3.2 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 re-

spective 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 Enterprise, 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 (`:scalable=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:scalable=true" family style weight
```

The output of this command could look like the following:

```
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 22.2 *Parameters of `fc-list`*

Parameter	Meaning and Possible Values
family	Name of the font family, for example, <code>FreeSans</code> .
foundry	The manufacturer of the font, for example, <code>urw</code> .
style	The font style, such as <code>Medium</code> , <code>Regular</code> , <code>Bold</code> , <code>Italic</code> , or <code>Heavy</code> .
lang	The language that the font supports, for example, <code>de</code> for German, <code>ja</code> for Japanese, <code>zh-TW</code> for traditional Chinese, or <code>zh-CN</code> for simplified Chinese.
weight	The font weight, such as <code>80</code> for regular or <code>200</code> for bold.
slant	The slant, usually <code>0</code> for none and <code>100</code> for italic.
file	The name of the file containing the font.
outline	<code>true</code> for outline fonts or <code>false</code> for other fonts.
scalable	<code>true</code> for scalable fonts or <code>false</code> for other fonts.
bitmap	<code>true</code> for bitmap fonts or <code>false</code> for other fonts.

Parameter	Meaning and Possible Values
pixelsize	Font size in pixels. In connection with fc-list, this option only makes sense for bitmap fonts.

22.3.3 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.

22.4 OpenGL—3D Configuration

22.4.1 Hardware Support

SUSE® Linux Enterprise includes several OpenGL drivers for 3D hardware support. [Table 22.3, “Supported 3D Hardware”](#) (page 478) provides an overview.

Table 22.3 *Supported 3D Hardware*

OpenGL Driver	Supported Hardware
NVIDIA*	NVIDIA* Chips: all except some legacy chipsets (GeForce2 and older)
DRI	Intel* i810/i815/i830M, Intel* 845G/852GM/855GM/865G/915G,915GM/945G

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 you update your system instead, 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.

22.4.2 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.

22.4.3 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.

22.4.4 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`.

22.4.5 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.

22.4.6 Installation Support

Apart from the `software rendering fallback` of the DRI driver, some OpenGL drivers in Linux are still 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 some 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.

22.4.7 For More Information

For information, refer to the README files in `/usr/X11R6/lib/X11/doc`. Find more information about nvidia driver installation at <http://www.suse.de/~sndirsch/nvidia-installer-HOWTO.html>.

Authentication with PAM

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 delegate authentication 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.

23.1 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 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 Enterprise, 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).

23.2 The PAM Configuration of `sshd`

To show how the theory behind PAM works, consider the PAM configuration of `sshd` as a practical example:

Example 23.1 *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 occurred 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 23.2, “Default Configuration for the `auth` Section”](#) (page 486).

Example 23.2 *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 23.3, “Default Configuration for the `account` Section”](#) (page 487). `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 23.4, “Default Configuration for the `password` Section”](#) (page 487).

Example 23.3 *Default Configuration for the `account` Section*

```
account required          pam_unix2.so
```

Example 23.4 *Default Configuration for the `password` Section*

```
password required        pam_pwcheck.so  nullok
password required        pam_unix2.so   nullok use_first_pass use_authok
#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 23.5 *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.

23.3 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`.

23.3.1 `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 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 23.6, “`pam_unix2.conf`”](#) (page 488).

Example 23.6 `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)`.

23.3.2 `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 23.7](#), “`pam_env.conf`” (page 489).

Example 23.7 *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`.

23.3.3 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 23.8](#), “`pam_pwcheck.conf`” (page 489) 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 23.8 *pam_pwcheck.conf*

```
password: nullok
```

23.3.4 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.

23.4 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 sub-directory `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 and made some information available about them at <http://www.suse.de/~kukuk/pam/>.

Mobile Computing with Linux

24

Mobile computing is mostly associated with laptops, PDAs, and cellular phones and the data exchange between them. Mobile hardware components, such as external hard disks, flash drives, or digital cameras, can be connected to laptops or desktop systems. A number of software components are involved in mobile computing scenarios and some applications are tailor-made for mobile use.

24.1 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 25, PCMCIA](#) (page 501).

24.1.1 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 Enterprise Desktop 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:

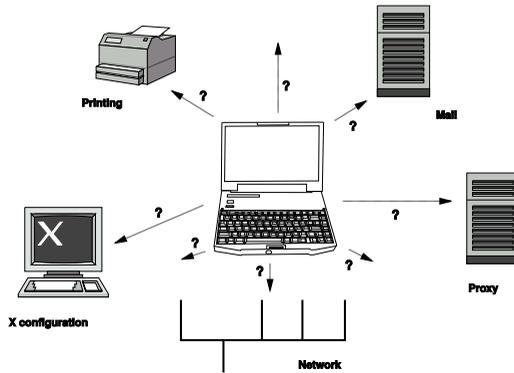
- Throttling the CPU speed
- Switching off the display illumination during pauses
- Manually adjusting the display illumination
- Disconnecting unused, hotplug-enabled accessories (USB CD-ROM, external mouse, unused PCMCIA cards, etc.)
- Spinning down the hard disk when idling

Detailed background information about power management in SUSE Linux Enterprise Desktop and about operating the YaST power management module is provided in [Chapter 27, *Power Management*](#) (page 523).

24.1.2 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 Enterprise Desktop handles this task for you.

Figure 24.1 *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

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.

X

If your laptop is temporarily connected to a beamer or an external monitor, the different display configurations must be available.

SUSE Linux Enterprise Desktop offers several ways of integrating a laptop into existing operating environments:

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. Find information about SCPM in [Chapter 26, System Configuration Profile Management](#) (page 509). You can use the Kicker applet Profile Chooser in KDE to switch between profiles. The application requires the `root` password before switching.

NetworkManager

NetworkManager is especially tailored for mobile networking on laptops. It provides a means to easily and automatically switch between network environments or different types of networks, such as wireless LAN and ethernet. NetworkManager supports WEP and WPA-PSK encryption in wireless LANs. It also supports dial-up connections (with `smpppd`). Both desktop environments (GNOME and KDE) include a front-end to NetworkManager.

Table 24.1 *Use Cases for NetworkManager*

My computer...	Use NetworkManager
is a laptop	Yes
is sometimes attached to different networks	Yes
provides network services (such as DNS or DHCP)	No
only uses a static IP address	No

Use the YaST tools to configure networking whenever NetworkManager should not handle network configuration.

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, sparing the effort of searching for a suitable installation source. Find detailed information about SLP in [Chapter 30, *SLP Services in the Network*](#) (page 631).

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.

24.1.3 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 Enterprise Desktop provides for each task.

System Monitoring

Two KDE system monitoring tools are provided by SUSE Linux Enterprise Desktop:

KPowersave

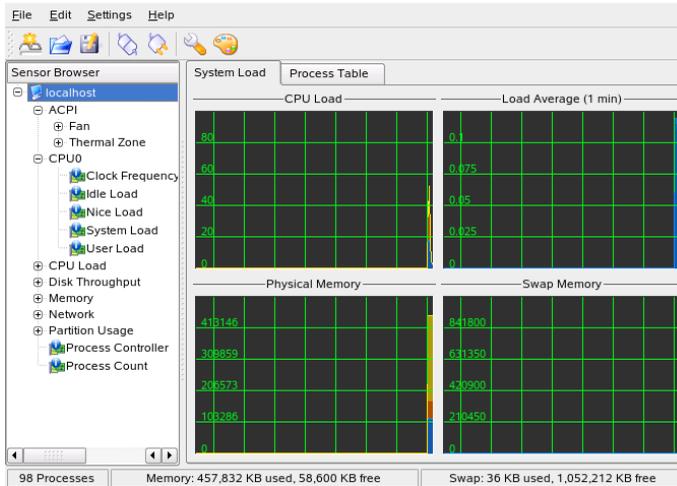
KPowersave is an applet that displays the state of the rechargeable battery in the control panel. The icon adjusts to represent the type of power supply. When working on AC power, a small plug icon is displayed. When working on batteries, the icon changes to a battery. The corresponding menu opens the YaST module for power management after requesting the `root` password. This allows setting the behavior of the system for different power sources. Find information about power management and about the corresponding YaST module in [Chapter 27, *Power Management*](#) (page 523).

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 envi-

ronment. Find more information about this program in its integrated help function or in the SUSE help pages.

Figure 24.2 *Monitoring the Battery State with KSysguard*



In the GNOME desktop, use the panel applet GNOME ACPI and the application System Monitor.

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:

Synchronizing E-Mail

Use an IMAP account for storing your e-mails in the office network. Then access the e-mails from the workstation using any disconnected IMAP-enabled e-mail client, like Mozilla Thunderbird Mail, Evolution, or KMail as described in *Applications*. 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 37, File Synchronization](#) (page 685).

Wireless Communication

As well as connecting to a home or office network with a cable, a laptop can also wirelessly connect 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 intermediaries 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. Find details about WLAN in [Section 28.1, “Wireless LAN”](#) (page 549).

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. Find more information about Bluetooth, its applications, and configuration in [Section 28.2, “Bluetooth”](#) (page 559).

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. Find more information about IrDA in [Section 28.3, “Infrared Data Transmission”](#) (page 570).

24.1.4 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 Enterprise Desktop is described in [Chapter 41, *Encrypting Partitions and Files*](#) (page 733).

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. Find general security issues regarding Linux and networks in [Chapter 43, *Security and Confidentiality*](#) (page 749). Security measures related to wireless networking are provided in [Chapter 28, *Wireless Communication*](#) (page 549).

24.2 Mobile Hardware

SUSE Linux Enterprise Desktop 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 as soon as they are connected with the system over the corresponding interface. The file managers of both GNOME and KDE offer flexible handling of mobile hardware items. To unmount any of these media safely, use the *Eject* feature of either file manager. These are described in more detail in the *GNOME User Guide* and *KDE User Guide*.

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` 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. Do advanced photo processing with The GIMP. For more details on digital cameras and image management, refer to Chapter 18, *Using Digital Cameras with Linux* (↑KDE User Guide).

24.3 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 497). 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 28.2, “Bluetooth”](#) (page 559) and [Section 28.3, “Infrared Data Transmission”](#) (page 570).

The support for synchronizing 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 *Applications*.

The program KPilot as integrated in Kontact is also available as an independent utility. It is described in *Applications*. The program KitchenSync is also available for synchronizing address data.

24.4 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 Enterprise Desktop. 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 Enterprise Desktop 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.

PCMCIA

PCMCIA is often used to refer the hardware itself, although the term originates from the organization that standardized all possible types of PC cards, the *PC Memory Card International Association*. In the beginning, PCMCIA only included PC cards (using a 16-bit bus like ISA cards), but later on CardBus cards (using a 32-bit bus) were included. A wide range of PCMCIA hardware is supported in Linux. Linux additionally includes tools for managing PCMCIA.

PCMCIA cards are mainly used in mobile computing for different purposes. Examples include:

- Ethernet and wireless LAN adapters
- Bluetooth cards
- Memory cards (Flash, SRAM, and others)
- Memory card adapters (SD, MMC, SmartMedia, CompactFlash, MemoryStick)
- Modems

Most of the card management is silently handled by `udev` and `hotplug`. When user interaction is required, you use the `pccardctl` command. For PCMCIA background information, refer to [Section 25.2, “PCMCIA in Detail”](#) (page 502). For details on `pccardctl`, refer to [Section 25.1, “Controlling PCMCIA Cards Using `pccardctl`”](#) (page 502).

25.1 Controlling PCMCIA Cards Using `pccardctl`

Card management is normally handled by `udev` and `hotplug` without requiring any user interaction at all. `pccardctl` offers manual control of the card in case the automated process does not work flawlessly.

The following is a list of the most important `pccardctl` commands. All commands must be executed as `root`:

`pccardctl insert`

If the card has not been detected automatically, notify the client drivers that the card has just been inserted.

`pccardctl eject`

Eject the card manually and notify the client drivers that it will be ejected. Cut power to the socket. This option is especially useful if you noticed problems with suspend and resume as described in [Section 25.3.2, “General Suspend Issues with PCMCIA”](#) (page 507).

`pccardctl suspend`

Shut down and disable power for a socket, but do not eject the card (unbind the appropriate modules).

`pccardctl resume`

After a `pccardctl resume`, bring up power for the socket and restore the configuration from before the `suspend` event.

For further information, refer to the manual page of `pccardctl`.

25.2 PCMCIA in Detail

The following sections outlines what happens in your Linux system when a PCMCIA device is plugged into your machine. Components interact with each other and many requirements need to be met to support a PCMCIA device.

The following is a very rough outline of the PCMCIA initialization process in Linux:

1. The PCMCIA bridge (or socket) must be set up properly as described in [Section 25.2.1, “Bridge Initialization”](#) (page 503). Prerequisites are:
 - an appropriate driver for the bridge
 - additional I/O and memory ranges for PC cards
2. After the bridge is properly set up, the bridge driver detects the presence of a card and triggers its initialization as described in [Section 25.2.2, “Card Initialization”](#) (page 504):
 - a. Determine the card type.
 - b. Supply the proper voltage.
 - c. Assign I/O and memory ranges and IRQ lines to the card.
 - d. Trigger the card or device initialization by binding the appropriate card driver.
 - e. For some cards, the Card Information Structure (CIS) needs to be uploaded.
3. Finally, the interface itself is set up and ready for use. See [Section 25.2.3, “Interface Setup”](#) (page 505) for details on this.

25.2.1 Bridge Initialization

Most PCMCIA bridges are PCI devices and are treated as such. The bridge initialization process can be summarized as follows:

1. Hotplug creates a PCI event.
2. `udev` calls `/sbin/hwup` to load the driver. `/sbin/hwup` checks `/etc/sysconfig/hardware` for an existing device configuration. If an appropriate configuration is found, that configuration is used. Otherwise `/sbin/hwup` calls `modprobe` with the `modalias` string provided by the kernel to load the driver module.
3. New hotplug events are sent (one per PCMCIA socket).
4. The following steps are omitted if only CardBus cards are used:

- a. The `pcmcia_socket` events trigger `udev` to call `/sbin/hwup` and load the `pcmcia` kernel module.
- b. All I/O and memory ranges specified in `/etc/pcmcia/config.opts` are added to the socket.
- c. The card services in the kernel check these ranges. If the memory ranges in `/etc/pcmcia/config.opts` are wrong, this step may crash your machine. See [Section 25.3.1, “Machine Crashes on PCMCIA”](#) (page 505) for information about how to debug and fix this issue.

After these steps have been successfully completed, the bridge is fully initialized. After this, the card itself is initialized as described in the following section.

25.2.2 Card Initialization

The events caused by plugging in a PCMCIA card can be summarized as follows:

1. A hotplug event occurs. For PC cards, this is a `pcmcia` event. For CardBus cards, this is a `pci` event.
2. For any events, `udev` calls `/sbin/hwup` to load a driver module. The module name is either specified in a `hwcfg*` file under `/etc/sysconfig/hardware` or via `modprobe` `modalias`.
3. If needed, device initialization triggers a firmware hotplug event. This searches for firmware and loads it.
4. The device driver registers the interfaces.

After these steps have been completed, the system proceeds with interface setup as described in the next section.

If your card is a PC card, you might need some of the following parameters in `/etc/sysconfig/pcmcia` to get it fully supported and working flawlessly:

PCMCIA_LOAD_CIS

A PC card's firmware is referred to as *CIS* (Card Information Structure). It provides additional implementation details of the card. `hwup` checks the integrity of the card's built-in CIS and tries to load another CIS from disk if the card's CIS proves

to be defective. The default setting is `yes`. To disable CIS loading from disk, set this variable to `no`.

PCMCIA_ALLOW_FUNC_MATCH

Linux device drivers contain a device ID table that tells drivers which devices to handle. This means that only those devices whose IDs are known to the kernel are supported. To support those cards whose ID is not listed, you can use function matching. This means that the driver is not selected by ID, but by the function of the card (such as a network card), and would be responsible for any PC card inserted with that function (such as network cards). The default setting is `yes`. To disable function matching, set this variable to `no`.

PCMCIA_COLDPLUG_REINSERT

Cards that have been inserted before booting sometimes fail to be detected. To prevent that, cause a soft eject and a soft insert of the card by setting `PCMCIA_COLDPLUG_REINSERT` to `yes`. The default setting is `no`.

25.2.3 Interface Setup

Depending on the card type, different interfaces are registered after initialization has been successfully completed. Interface registration is handled by `udev`'s hotplug. For details on `udev` and hotplug, refer to [Chapter 20, *Dynamic Kernel Device Management with udev*](#) (page 445).

25.3 Troubleshooting

The following is a list of the most prominent issues that are occasionally encountered with PCMCIA. More information about this is available in the PCMCIA README (`/usr/share/doc/packages/pcmciautils/README.SuSE`).

25.3.1 Machine Crashes on PCMCIA

Your machine crashes when PCMCIA is started on boot. To find out what caused your machine to crash, set it up manually as described below. In carefully setting up PCMCIA manually, you can clearly identify the step or component that crashed your machine.

Once the culprit has been identified, you can circumvent the problematic step or component.

To manually set up PCMCIA, proceed as follows:

- 1** Prevent PCMCIA from being started on system boot and enable SysRq for easier debugging by appending the following options to the boot prompt:

```
init=3 pcmcia=off sysrq=1
```

For more information about SysRq, refer to `/usr/src/linux/Documentation/sysrq.txt`.

- 2** Boot the system into a text-based environment and log in as `root`.
- 3** Add the appropriate PCMCIA modules to the kernel:

```
/sbin/modprobe yenta_socket  
/sbin/modprobe pcmcia
```

- 4** Start the PCMCIA socket:

```
/sbin/pcmcia-socket-startupN
```

Replace *N* with the number of the socket. Repeat this step for each socket.

- 5** If the previous step crashed your machine, this might have been caused by wrong I/O or memory ranges specified in `/etc/pcmcia/config.opts`. To prevent this, do one of the following:

- Exclude ranges in `/etc/pcmcia/config.opts` and retry the socket setup.
- Add the ranges manually as described below.

After you successfully added the appropriate ranges manually, set them permanently by including them in `/etc/pcmcia/config.opts`.

- 6** After the socket setup has been successfully completed, card initialization and interface setup work as described in [Section 25.2.2, “Card Initialization”](#) (page 504) and [Section 25.2.3, “Interface Setup”](#) (page 505).

To manually add I/O ranges, proceed as follows (for each socket):

- 1 Change into the directory that holds the range configurations (in this case, `pcmcia_socket0`, adapt for other socket numbers):

```
cd /sys/class/pcmcia_socket/pcmcia_socket0
```

- 2 Execute the following command:

```
echo begin - end > available_resources_io
```

Replace *begin* and *end* with the addresses where the new range should start and end. The correct values can only be determined by trial and error.

Manually adding the following ranges:

```
echo 0x800 - 0x8ff > available_resources_io
echo 0xc00 - 0xcff > available_resources_io
```

equals the following line from `/etc/pcmcia/config.opts`:

```
include port 0x800-0x8ff, port 0xc00 0xcff
```

The same procedure applies for the memory ranges under `available_resources_mem`.

IMPORTANT: Identifying Faulty Default Settings

If you find a faulty range in the default configuration file (`/etc/pcmcia/config.opts`) shipped with this product, file a bug against it in <http://bugzilla.novell.com>, so that developers can look into this issue.

25.3.2 General Suspend Issues with PCMCIA

Whenever suspending your system (suspend to disk, suspend to RAM, or standby), do not plug or unplug any hardware items while the system is in suspend mode. Otherwise, the system might not resume properly.

To automatically eject PCMCIA cards on suspend, proceed as follows:

- 1 Log in as `root`.
- 2 Open the file `/etc/powersave/sleep`.

3 Set the following variables:

```
SUSPEND2DISK_EJECT_PCMCIA="yes"  
SUSPEND2RAM_EJECT_PCMCIA="yes"  
STANDBY_EJECT_PCMCIA="yes"
```

4 Save the file to apply your settings.

If additional modules need to be ejected on suspend, proceed as above and add the module names to the following variables:

```
UNLOAD_MODULES_BEFORE_SUSPEND2DISK=""  
UNLOAD_MODULES_BEFORE_SUSPEND2RAM=""  
UNLOAD_MODULES_BEFORE_STANDBY=""
```

For general information about the powersave daemon, refer to [Section 27.5, “The powersave Package”](#) (page 535).

25.3.3 For More Information

Find the latest up-to-date information about PCMCIA in `/usr/share/doc/packages/pcmciautils/README.SuSE`. For a comprehensive overview of PCMCIA hardware and its fields of use, turn to the official PCMCIA Web site (<http://www.pcmcia.org/pccard.htm>). To check whether a certain card or device is generally supported by Linux, refer to the *Linux PCMCIA/CF/CardBus Card Survey* at http://tuxmobil.org/pcmcia_linux.html.

System Configuration Profile Management

26

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. It 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.

26.1 Terminology

The following are some terms used in SCPM documentation and in the YaST module.

system configuration

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.

profile or system profile

A state that has been preserved and can be restored at any time.

active profile

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.

resource

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.

resource group

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.

26.2 Setting Up SCPM

The following sections introduce SCPM configuration by means of a real life example: a mobile computer that is run in several different networks. The major challenges faced in this scenario are:

- Varying network environments, like wireless LAN at home and an ethernet at work
- Different printer configuration at home and at work

To get SCPM up and running and have it manage your changing system configuration, proceed as follows:

- 1 Add the Profile Chooser applet to your panel and configure it to allow user switching as described in [Section 26.3.1, “Configuring the Profile Chooser Panel Applet”](#) (page 511).
- 2 Configure SCPM using the YaST Profile Manager module as described in [Section 26.3.2, “Configuring Basic SCPM Settings”](#) (page 512).
- 3 Create a profile for each of the different setups using SUMF (SCPM Unified Management Front-End) as described in [Section 26.3.3, “Creating a New Profile”](#) (page 514).
- 4 Switch to the profile appropriate for your current situation as described in [Section 26.3.4, “Switching Profiles”](#) (page 515).

If you prefer to control SCPM with its command line interface, refer to [Section 26.4, “Configuring SCPM Using the Command Line”](#) (page 517) for details.

26.3 Configuring SCPM Using a Graphical User Interface

The following sections introduce the graphical tools used for controlling your profile settings.

26.3.1 Configuring the Profile Chooser Panel Applet

Before you can use Profile Chooser to control your system configuration, configure it to be started automatically on login:

- In GNOME, right-click the panel and select Profile Chooser from the list of available applets.
- In KDE, select *System* → *Desktop Applet* → *Profile Chooser* to add Profile Chooser to your panel.

26.3.2 Configuring Basic SCPM Settings

Configure the basic behavior of SCPM through YaST.

- 1 Start YaST from the main menu and select the YaST Profile Manager.
- 2 In *System Configuration Profile Management*, click *Options* and select *Enabled*.
- 3 Determine how verbose SCPM should be by selecting any or both of *Verbose Progress Messages* and *Log Debug Messages*.
- 4 Determine the appropriate switch mode for your setup:
 - Should SCPM list any changed resource when switching to another profile and save these changes to the active profile? Select *Normal* or *Save Changes*.
 - Should SCPM drop any changed resource configuration when switching? Select *Drop Changes*.
- 5 Set the boot mode and determine whether changes to the current profile should be saved or discarded with profile switching triggered at boot time.
- 6 Make sure that all resource groups needed are covered by the active selection, displayed in the *Resource Groups* section. If you need additional resource groups, adjust the resources with *Configure Resources*. For details, refer to [Section 26.3.6, “Configuring Resource Groups”](#) (page 516).

For the example scenario, you do not need to configure additional resources, because printer and network resources are included by default.

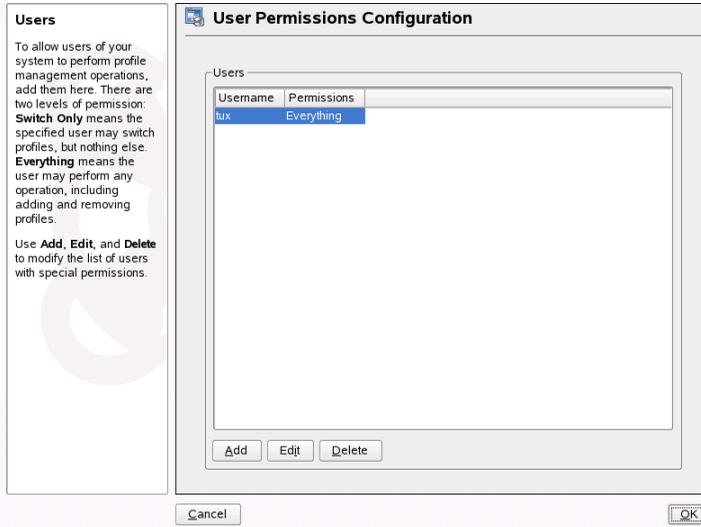
Figure 26.1 *YaST: Basic SCPM Configuration*



To allow users other than `root` to manage profiles, proceed as follows:

- 1 Start YaST from the main menu and select the YaST Profile Manager.
- 2 Check *Permit non-root Users to Manage Profiles*. See [Figure 26.2, “YaST: Configure SCPM Users”](#) (page 514).
- 3 Click *Configure*.
- 4 Click *Add* to add any user who should be able to manage profiles.
- 5 For each user, specify whether to grant switch permissions only or whether this user should be allowed to switch, modify, and create profiles.
- 6 Click *Finish* to apply your settings and close YaST.

Figure 26.2 *YaST: Configure SCPM Users*



26.3.3 Creating a New Profile

After you have enabled SCPM, you have a profile named `default` that contains your current system configuration. Create another profile that matches the requirements of the other setup.

To add a new profile based on the current system configuration, proceed as follows:

- 1 Right-click the Profile Chooser and select *Run Profile Manager (SUMF)*.
- 2 Select *Profiles* → *Add*.
- 3 Enter the name of the new profile and click *OK*.
- 4 Determine whether the new profile should be the active profile.

If you selected *Yes*, SCPM switches to the new profile immediately after it has been created.

For this example, do the following:

- 1 In your home setup, enable SCPM.
- 2 Rename the `default` profile to a more descriptive name by starting SUMF and selecting *Profiles* → *Edit* and entering the new name.
- 3 In your setup at work, start SUMF and create the profile for your system environment at work.

Once you have all desired profiles, you are ready to switch to them whenever a different system setup is required. Switching profiles is described in [Section 26.3.4, “Switching Profiles”](#) (page 515).

26.3.4 Switching Profiles

There are two ways to switch profiles. You can either select a new profile at boot or switch profiles in the running system.

To select a profile at boot, proceed as follows:

- 1 In the boot screen, press `F2` to enter the *Other Options* menu.
- 2 Press `F3` to access the list of profiles available.
- 3 Use the arrow keys to select the appropriate profile and hit `Enter`.

The system boots into the configuration selected.

To switch profiles in a running system, proceed as follows:

- 1 Make sure that you are allowed to switch profiles as a non-`root` user. If you are not allowed to do so, refer to [Section 26.3.2, “Configuring Basic SCPM Settings”](#) (page 512).
- 2 Left-click the Profile Chooser panel applet.
- 3 Select the desired profile in the menu that opens using the arrow keys and hit `Enter`. SCPM runs a check for modified resources and prompts you for a confirmation of the switch. If changes have been made to the system configuration before the switch, select whether to keep them or discard them when switching to another profile.

26.3.5 Editing a Profile

To adjust existing profiles to a changed environment, for example, if you want to change the printer configuration of your home network, proceed as follows:

- 1 Switch to the profile to adjust as described in [Section 26.3.4, “Switching Profiles”](#) (page 515). In this example, you would choose the `home` profile.
- 2 Change the resources that need to be adjusted using the appropriate YaST module. In this example, run the YaST printer configuration.
- 3 After the configuration changes have been applied and you request a profile switch, SCPM asks whether these changes should be permanently applied to the formerly active profile.

TIP: Forcing a Profile Update

If you want to force an update of the active profile, click the profile in the profile selection menu of the Profile Chooser panel applet. This triggers a reload of your profile and you are asked whether to apply the configuration changes or discard them.

26.3.6 Configuring Resource Groups

SCPM comes with a set of predefined resource groups that are included in any profile by default. However, some scenarios require the inclusion of additional resources and resource groups.

To change the resource configuration, proceed as follows:

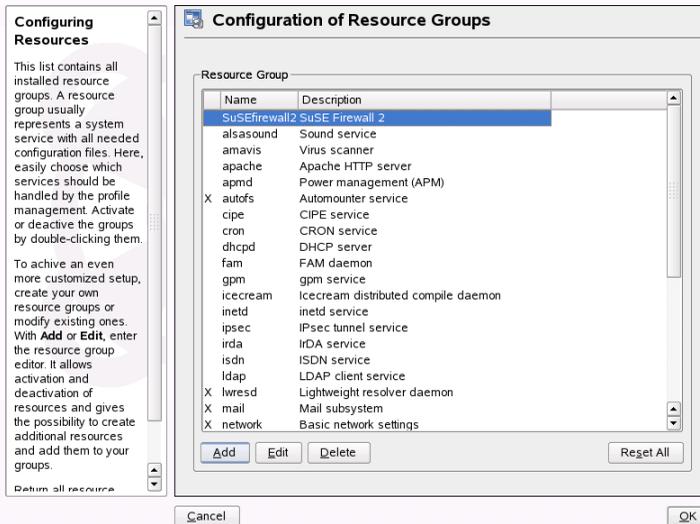
- 1 Start YaST from the main menu and start the YaST Profile Manager module.
- 2 In the *System Configuration Profile Management* dialog, click *Configure* in the *Resource Groups* part of the dialog.

All resource groups available on your system are listed as shown in [Figure 26.3, “Configuring Resource Groups”](#) (page 517).

- 3 To add or edit a resource group:

- a Set or edit *Resource Group* and *Description*.
 - b Enter the appropriate resources (resources, services, or both) and delete those that are not needed. To reset the status of the selected resources—discard any changes made to them and return to the initial configuration values—choose *Reset Group*.
 - c Click *OK* to leave the resource configuration.
- 4 Click *OK* to save your changes to the active profile.

Figure 26.3 *Configuring Resource Groups*



26.4 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.

26.4.1 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.

26.4.2 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`.

Rename or delete profiles 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.

26.4.3 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 `switch` 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.

26.4.4 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`

Run prior to stopping services when leaving the profile

`poststop`

Run after stopping services when leaving the profile

`prestart`

Run prior to starting services when activating the profile

`poststart`

Run 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`). 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`.

26.5 Troubleshooting

This section covers frequent problems encountered with SCPM. Learn how they can arise and how you can solve these issues.

26.5.1 SCPM and NetworkManager

NetworkManager and SCPM share functionality. Both integrate a machine into an existing network, hiding this transaction from the user. NetworkManager works dynamically and adapts to any new environment. SCPM is used to restore defined system setups.

Using NetworkManager and SCPM in parallel does not work properly, because NetworkManager does not provide configurations that can be restored by SCPM. SCPM works exceedingly well for anyone who needs reproducible setups. Any private user constantly switching networks should consider using NetworkManager if network setup is the only component that needs to be adjusted. If you want to use SCPM to manage your system configuration but NetworkManager to manage networking, remove the network resource from SCPM. If you want to use SCPM for network configuration management, disable NetworkManager.

26.5.2 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.

26.6 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`.

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.

27.1 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. There must be a swap partition at least as big as the RAM to write all the active data. 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 it. 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, like PCI devices that can be put into a special power saving mode, can be deactivated with ACPI (at least theoretically) or permanently disabled 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.

27.2 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 Enterprise 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 of `apm=parameter`. *parameter* is one of:

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 provides many other features.

27.3 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 27.3.4, “Troubleshooting”](#) (page 532) for more information about troubleshooting ACPI problems.

27.3.1 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 `acpid`. 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`. One possibility for easy access to those values is the `powersave` command, which acts as a front-end for the Powersave daemon. 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 27.3.3, “ACPI Tools”](#) (page 532). The charge level at which a battery event (such as warning, low and critical) is triggered can be specified in `alarm`.

`/proc/acpi/button`

This directory contains information about various switches, like the laptop lid and buttons.

`/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 the system 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 its various front-ends, like `powersave`, `kpowersave`, and `wmpowersave`. See [Section 27.3.3, “ACPI Tools”](#) (page 532).

27.3.2 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. There are two main approaches to performing CPU frequency scaling—by the kernel itself or by a userspace application. Therefore, there are different kernel governors that can be set below `/sys/devices/system/cpu/cpu*/cpufreq/`.

userspace governor

If the userspace governor is set, the kernel gives the control of CPU frequency scaling to a userspace application, usually a daemon. In SUSE Linux Enterprise distributions, this daemon is the `powersaved` package. When this implementation is used, the CPU frequency is adjusted in regard to the current system load. By default, one of the kernel implementations is used. However, on some

hardware or in regard to specific processors or drivers, the userspace implementation is still the only working solution.

ondemand governor

This is the kernel implementation of a dynamic CPU frequency policy and should work on most systems. As soon as there is a high system load, the CPU frequency is immediately increased. It is lowered on a low system load.

conservative governor

This governor is similar to the ondemand implementation, except that a more conservative policy is used. The load of the system must be high for a specific amount of time before the CPU frequency is increased.

powersave governor

The cpu frequency is statically set to the lowest possible.

performance governor

The cpu frequency is statically set to the highest possible.

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 the kernel ondemand governor or 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 Enterprise these technologies are controlled by the powersave daemon. The configuration is explained in [Section 27.5, “The powersave Package”](#) (page 535).

27.3.3 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 `pmttools`).

27.3.4 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 27.5.4, “Troubleshooting”](#) (page 541).

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)

27.4 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 27.6, “The YaST Power Management Module”](#) (page 543).

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.

27.5 The powersave Package

The `powersave` package cares about all the previously-mentioned power saving functions. Due to the increasing demand for lower energy consumption in general, some of its features are also important on workstations and servers, such as `suspend`, `standby`, or `cpu frequency scaling`.

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, except `acpid` that acts as a multiplexer for `acpi` events, 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.

27.5.1 Configuring the powersave Package

The configuration of `powersave` is distributed to several files. Every configuration option listed there contains additional documentation about its functionality.

`/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 (usually a Bash script) in `/usr/lib/powersave/scripts/`. Predefined internal actions are:

- `ignore`
- `throttle`
- `dethrottle`
- `suspend_to_disk`
- `suspend_to_ram`
- `standby`
- `do_suspend_to_disk`
- `do_suspend_to_ram`
- `do_standby`
- `notify`
- `screen_saver`
- `reread_cpu_capabilities`

`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:

`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.

`set_disk_settings`

Executes the disk settings made in `/etc/sysconfig/powersave/disk`.

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 a pop-up window in X or a message on the console. Subsequently, the event `EVENT_GLOBAL_SUSPEND2DISK` is generated, resulting in the execution of the mentioned actions and a secure system suspend mode. The internal action `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 and whether the userspace or the kernel implementation should be used.

`/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 27.5.4, “Troubleshooting”](#) (page 541) 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/disk`

This configuration file controls the actions and settings made regarding the hard disk.

`/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.

27.5.2 Configuring APM and ACPI

Suspend and Standby

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. This sleep mode is enabled by default and should work on all systems.

Suspend to RAM (ACPI S3, APM suspend)

Saves the states of all devices to the main memory. Only the main memory continues consuming power. This sleep mode is disabled by default because still causes problems on some systems. However, support has been extended greatly.

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 Enterprise):

```
EVENT_GLOBAL_SUSPEND2DISK=
    "prepare_suspend_to_disk screen_saver do_suspend_to_disk"
EVENT_GLOBAL_SUSPEND2RAM=
    "prepare_suspend_to_ram screen_saver do_suspend_to_ram"
EVENT_GLOBAL_STANDBY=
    "prepare_standby screen_saver 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=12
BATTERY_LOW=7
BATTERY_CRITICAL=2
```

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 27.5.1, “Configuring the powersave Package”](#) (page 535).

```
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 27.6, “The YaST Power Management Module”](#) (page 543).

27.5.3 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.

```
EVENT_OTHER="ignore"
```

This event happens if an unknown event is encountered by the daemon. Unknown events include ACPI hotkeys on some machines.

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.

27.5.4 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.

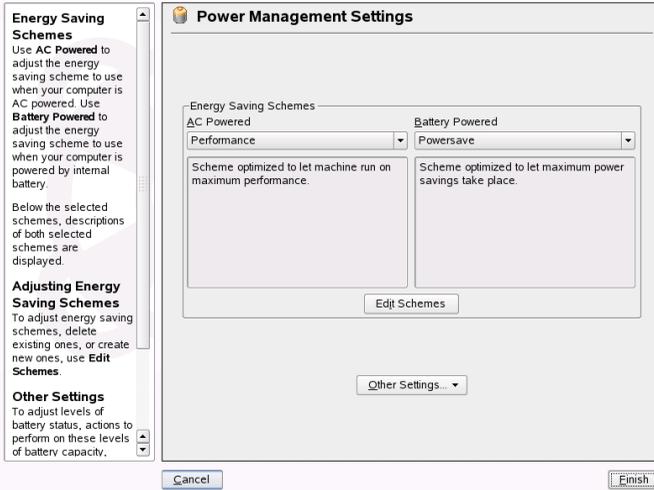
27.5.5 For More Information

- [/usr/share/doc/packages/powersave](#)—Local Powersave daemon documentation
- <http://powersave.sourceforge.net>—Most recent Powersave daemon documentation
- http://www.opensuse.org/Projects_Powersave—Project page in the openSUSE wiki

27.6 The YaST Power Management Module

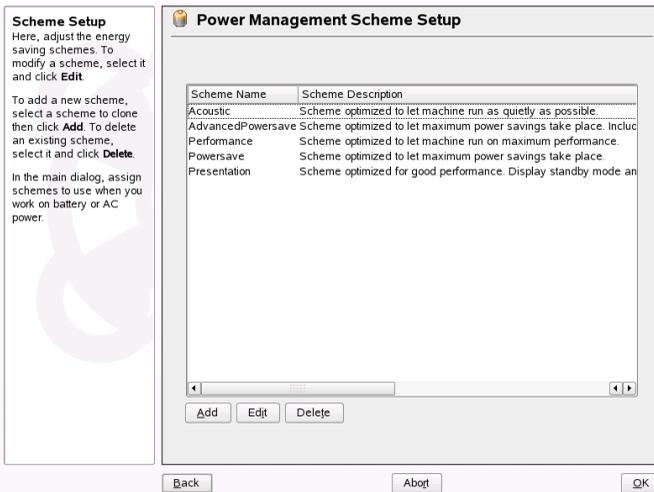
The YaST power management module can configure all power management settings already described. When started from the YaST Control Center with *System* → *Power Management*, the first dialog of the module opens (see [Figure 27.1](#), “Scheme Selection” (page 544)).

Figure 27.1 *Scheme Selection*



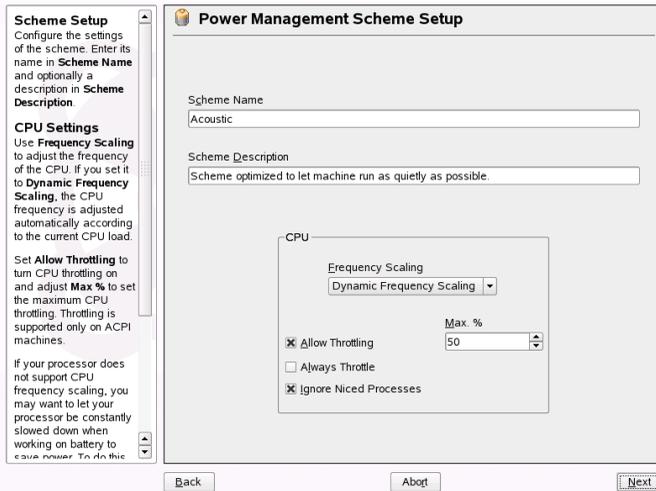
In this dialog, select the schemes to use for battery operation and AC operation. To add or modify the schemes, click *Edit Schemes*, which opens an overview of the existing schemes like that shown in [Figure 27.2, “Overview of Existing Schemes”](#) (page 544).

Figure 27.2 *Overview of Existing Schemes*



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 27.3, “Configuring a Scheme”](#) (page 545).

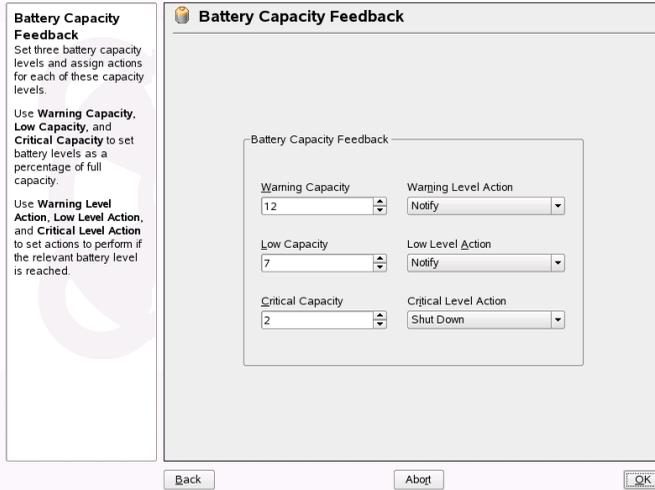
Figure 27.3 *Configuring a Scheme*



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 and whether processes with low priority (*niced* processes) should be ignored when adjusting the CPU frequency. 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/powersave_manual.html#Thermal` to learn how you can use the fan and passive cooling methods.

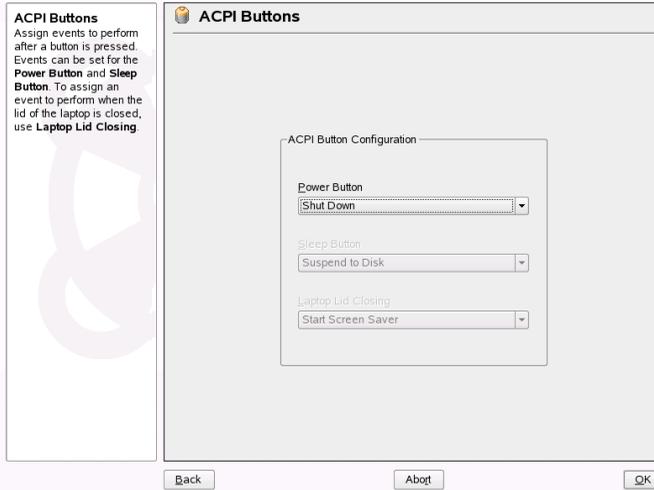
Global power management settings can also be made from the initial dialog using *Battery Warning*, *ACPI Settings*, or *Suspend Permissions*. Access these controls by clicking *Other Settings* and selecting the appropriate item from the menu. Click *Battery Warning* to access the dialog for the battery charge level, shown in [Figure 27.4, “Battery Charge Level”](#) (page 546).

Figure 27.4 *Battery Charge Level*



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 27.5 *ACPI Settings*



Access the dialog for configuring the ACPI buttons using *ACPI Settings*. It is shown in [Figure 27.5, “ACPI Settings”](#) (page 547). 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.

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.

28.1 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 28.1 *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.

28.1.1 Hardware

802.11 cards are not supported by SUSE® Linux Enterprise. 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 Enterprise 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`.

28.1.2 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 558)), 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 553)) 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 en-

terprises. 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.

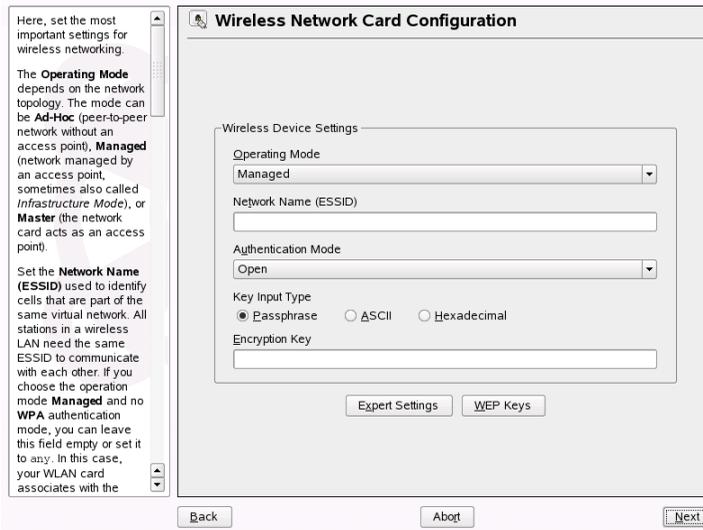
CCMP (defined in IEEE 802.11i)

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.

28.1.3 Configuration with YaST

To configure your wireless network card, start the YaST *Network Card* module. Here you can also choose whether to use YaST or NetworkManager for managing your network card. If you select YaST, select the device type *Wireless* in *Network Address Setup* and click *Next*. In *Wireless Network Card Configuration*, shown in [Figure 28.1](#), “YaST: Configuring the Wireless Network Card” (page 554), make the basic settings for the WLAN operation:

Figure 28.1 YaST: Configuring the Wireless Network Card



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 553) and [Section “Security”](#) (page 558) 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.

Shared Key

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 *WEP 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 ac-

ording 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 *Identity*, *Client Certificate*, *Client Key*, and *Server Certificate*. TTLS and PEAP require *Identity* and *Password*. *Server Certificate* and *Anonymous Identity* are 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 *Details* to enter the advanced authentication dialog for your WPA-EAP setup. Select the authentication method for the second stage of EAP-TTLS or EAP-PEAP communication. If you selected TTLS in the previous dialog, choose *any*, MD5, GTC, CHAP, PAP, MSCHAPv1, or MSCHAPv2. If you selected PEAP, choose *any*, MD5, GTC, or MSCHAPv2. *PEAP version* can be used to force the use of a certain PEAP implementation if the automatically-determined setting does not work for you.

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.

28.1.4 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.

28.1.5 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 558)). 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.

28.1.6 Troubleshooting

If your WLAN card fails to respond, check if you have downloaded the needed firmware. Refer to [Section 28.1.1, “Hardware”](#) (page 550). 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 features an article on this subject at http://en.opensuse.org/SDB:Name_Resolution_Does_Not_Work_with_Several_Concurrent_DHCP_Clients.

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 Enterprise 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`.

28.1.7 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.hp1.hp.com/personal/Jean_Tourrilhes/Linux/Wireless.html.

28.2 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 10 and 100 meters.

28.2.1 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. If you need to debug problems with Bluetooth connections, install the package `bluez-hcidump`.

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.

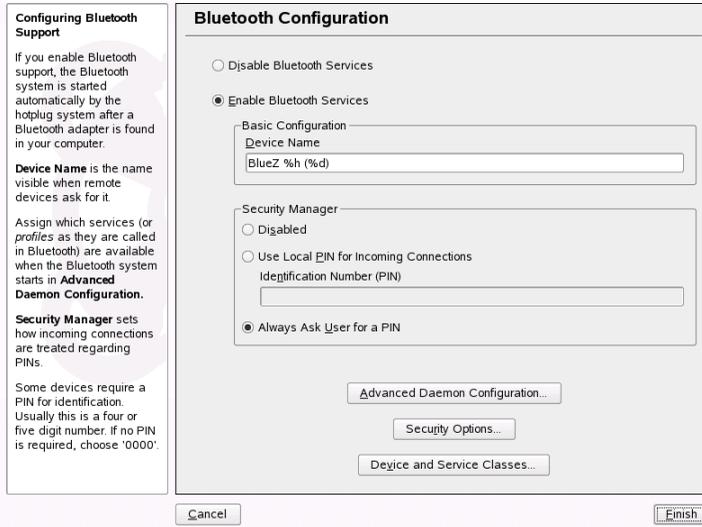
28.2.2 Configuration

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 28.2, “YaST Bluetooth Configuration”](#) (page 562), 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 28.2 *YaST Bluetooth Configuration*



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 561). 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.

28.2.3 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, ac-

tivate both `hcid` and `sdpd` can with `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

Use `hcitool` 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.

Use `hcitool name device-address` 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 modify these settings. For example, `hciconfig hci0 name TEST` sets the name to `TEST`.

sdptool

Use `sdptool` to check which services are made available by a specific device. The command `sdptool browse device_address` returns all services of a device. Use `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.

28.2.4 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 whether to 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.

28.2.5 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, establish a connection 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*, do this with the following two commands:

```
ip addr add 192.168.1.3/24 dev bnep0
ip link set bnep0 up
```

On *H2*, use the following commands:

```
ip addr add 192.168.1.4/24 dev bnep0
ip link set bnep0 up
```

Now *H1* can be accessed from *H2* at 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 Enterprise Desktop. 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. You need a special service daemon running on the laptop to get the data from the phone. If the package `kbluetooth` is installed, you do not need to start a special daemon. If `kbluetooth` is not installed, use the `opd` daemon from the `bluez-utils` package. 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.

If you use `kbluetooth`, you are prompted for a directory when the photograph is received on the laptop.

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.

28.2.6 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 566))?

The setup described in [Section “Network Connection between Two Hosts”](#) (page 566) may not work for several reasons. For example, one of the two computers may not support SSH. 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 567))? 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. Verify that `opd` is active and that you have write access to the specified directory.

Does the scenario described in [Section “Data Transfer from a Mobile Phone to the Computer”](#) (page 567) 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`.

If you have installed the `bluez-hcidump` package, you can use `hcidump -X` to check what is sent between the devices. Sometimes the output helps give a hint where the problem is, but be aware of the fact that it is only partly in “clear text.”

28.2.7 For More Information

Some additional (last-minute) documentation can be found in `/usr/share/doc/packages/bluez-utils/` (German and English versions available).

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>

28.3 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.

Find information about IrDA 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/>.

28.3.1 Software

The necessary kernel modules are included in the kernel package. The package `irda` provides the necessary helper applications for supporting the infrared interface. Find the documentation at `/usr/share/doc/packages/irda/README` after the installation of the package.

28.3.2 Configuration

The IrDA system service is not started automatically when the system is booted. Use the YaST IrDA module for 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 28.3.4, “Troubleshooting”](#) (page 572). 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.

If preferred, configure manually in the file `/etc/sysconfig/irda`. This file contains only one variable, `IRDA_PORT`, which determines the interface to use in SIR mode.

28.3.3 Usage

Data can be sent to the device file `/dev/irllpt0` for printing. The device file `/dev/irllpt0` 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 *Add* → *Directly Connected Printers*. Select *IrDA Printer* and click *Next* to configure the printer device. Usually, `ir1p0` is the right connection. Click *Finish* to apply your settings. Details about operating printers in Linux are available in [Chapter 19, Printer Operation](#) (page 421).

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.”

28.3.4 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 28.1, “Output of irdadump”](#) (page 572) appears regularly when a Canon BJC-80 printer is in visible range of the computer:

Example 28.1 *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 IV. Services

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 29.1, “Several Protocols in the TCP/IP Protocol Family”](#) (page 577) 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>.

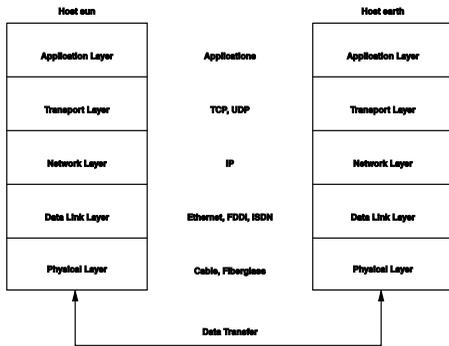
Table 29.1 *Several Protocols in the TCP/IP Protocol Family*

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 29.1, “Simplified Layer Model for TCP/IP”](#) (page 579), 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 29.1 *Simplified Layer Model for TCP/IP*



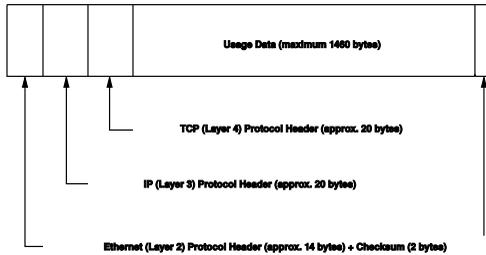
The diagram provides one or two examples for each layer. The layers are ordered according to *abstraction levels*. The lowest layer is very close to the hardware. The uppermost layer, however, is almost a complete abstraction from the hardware. Every layer has its own special function. The special functions 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 29.2, “TCP/IP Ethernet Packet”](#) (page 580). The proof sum is

located at the end of the packet, not at the beginning. This simplifies things for the network hardware.

Figure 29.2 *TCP/IP Ethernet Packet*



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.

29.1 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 29.2, “IPv6—The Next Generation Internet”](#) (page 583).

29.1.1 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 29.1, “Writing IP Addresses”](#) (page 581).

Example 29.1 *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.

29.1.2 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 29.2, “Linking IP Addresses to the Netmask”](#) (page 582). 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 29.2, “Linking IP Addresses to the Netmask”](#) (page 582) the first net with 24 bits could also be written as 192.168.0.0/24.

Example 29.2 *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 29.2 *Specific Addresses*

Address Type	Description
Base Network Address	This is the netmask AND any address in the network, as shown in Example 29.2, “Linking IP Addresses to the Netmask”

Address Type	Description
	(page 582) under <i>Result</i> . 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.
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 29.3, “Private IP Address Domains”](#) (page 583).

Table 29.3 *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

29.2 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.

29.2.1 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 29.2.2, “Address Types and Structure”](#) (page 586).

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 29.2.3, “Coexistence of IPv4 and IPv6”](#) (page 590). 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*).

29.2.2 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 29.3, “Sample IPv6 Address”](#) (page 587), where all three lines represent the same address.

Example 29.3 *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 29.4, “IPv6 Address Specifying the Prefix Length”](#) (page 587), 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 29.4 *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 29.4, “Various IPv6 Prefixes”](#) (page 587).

Table 29.4 *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.

Prefix (hex)	Definition
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).
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 29.2.3, “Coexistence of IPv4 and IPv6”](#) (page 590)) 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 auto-

matically 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.

29.2.3 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 29.2.2, “Address Types and Structure”](#) (page 586)).

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.

29.2.4 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.

29.2.5 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).

29.3 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 protocol `whois` is closely related to DNS. With this program, quickly find out who is responsible for any given domain.

29.4 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 29.6, “Configuring a Network Connection Manually”](#) (page 612).

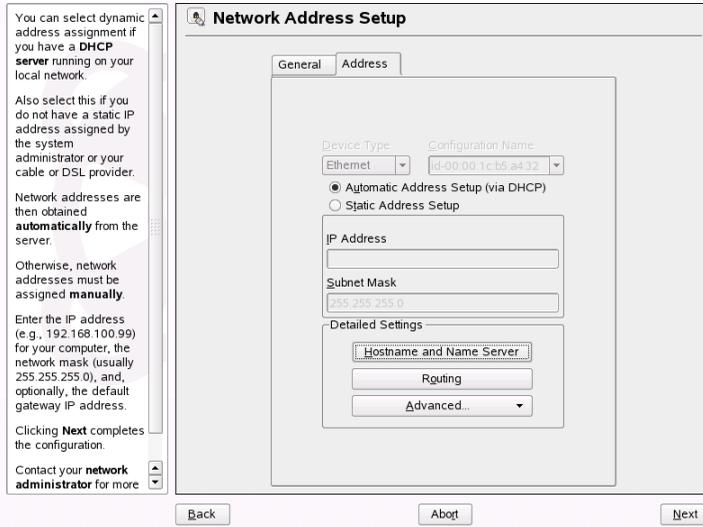
During installation, YaST can be used to configure automatically all interfaces that have been detected. Additional hardware can be configured any time after installation in the installed system. The following sections describe the network configuration for all types of network connections supported by SUSE Linux Enterprise.

29.4.1 Configuring the Network Card with YaST

To configure your network wired or wireless card in YaST, select *Network Devices* → *Network Card*. After starting the module, YaST displays a general network configuration dialog. Choose whether to use YaST or NetworkManager to manage all your network devices. If you want to configure your network in the traditional way with the YaST, check *Traditional Method with ifup* and click *Next*. To use NetworkManager, check *User Controlled with NetworkManager* and click *Next*. Find detailed information about NetworkManager in [Section 29.5, “Managing Network Connections with NetworkManager”](#) (page 609).

The upper part of the next dialog shows a list with all the network cards available for configuration. Any card properly detected is listed with its name. To change the configuration of the selected device, click *Edit*. Devices that could not be detected can be configured using *Add* as described in [Section “Configuring an Undetected Network Card”](#) (page 601).

Figure 29.3 *Configuring a Network Card*



Changing the Configuration of a Network Card

To change the configuration of a network card, select a card from the list of the detected cards in the YaST network card configuration module and click *Edit*. The *Network Address Setup* dialog appears in which to adjust the card configuration using the *Address* and *General* tabs. For information about wireless card configuration, see [Section 28.1.3, “Configuration with YaST”](#) (page 554).

Use the *Address* tab to adjust the following aspects of your network card configuration:

Type of Address Setup and IP Address Information

Depending on the purpose of your SUSE Linux Enterprise machine, decide whether a static or dynamic address should be used. Most client setups in corporate networks or home networks work fine with dynamic setup through DHCP. Servers or routers, however, should use static IP addresses. For details, refer to [Section “Configuring IP Addresses”](#) (page 596).

Hostname and DNS

To enable your SUSE Linux Enterprise machine to integrate into a network, set hostname and name service information. If DHCP is used for the network setup,

these settings are made automatically. For details, refer to [Section “Configuring Hostname and DNS”](#) (page 598).

Routing

To ensure that network traffic from your machine takes the correct path, set a default gateway and, if needed, a route. If DHCP is used, these settings are made automatically. For details, refer to [Section “Configuring Routing”](#) (page 598).

Advanced Options

Some hardware needs additional options to be passed to the kernel to make it work. Access the dialog to enter these options with *Hardware Details*. Refer to [Section “Adding Special Hardware Options”](#) (page 599) for details.

The DHCP setup that is run by default applies to most use cases. To fine-tune the DHCP configuration, use *DHCP Options*. Refer to [Section “Configuring IP Addresses”](#) (page 596) for details.

If your machine should have multiple IP addresses, set them up using *Additional Addresses*. See [Section “Configuring IP Addresses”](#) (page 596).

Use the *General* tab to adjust the following items:

Firewall Zone

Determine whether your network interface should be protected by a firewall. For details, refer to [Section “Configuring the Firewall”](#) (page 600).

Device Activation

Depending on which applications or scripts you use to control your network devices, set the appropriate start mode. For details, refer to [Section “Starting the Device”](#) (page 599).

MTU

Set the maximum transfer rate (MTU) of your interface. Normally, you can safely leave this setting empty and run with the defaults. Only change this value if your setup explicitly requires it.

Configuring IP Addresses

When possible, wired network cards available during installation are automatically configured to use automatic address setup, DHCP.

DHCP 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 in *DHCP Client Options*. Find this dialog from the *Address* tab by selecting *Advanced* → *DHCP Options*. Specify whether the DHCP server should always honor broadcast requests and any identifier to use. If you have a virtual host setup where different hosts communicate through the same interface, an identifier is necessary to distinguish them.

DHCP is a good choice for client configuration but it is not ideal for server configuration. To set a static IP address, proceed as follows:

- 1 Select a card from the list of detected cards in the YaST network card configuration module and click *Edit*.
- 2 In the *Address* tab, choose *Static Address Setup*.
- 3 Enter *IP Address* and *Subnet Mask*.
- 4 Click *Next*.
- 5 To activate the configuration, click *Next* again.

One network device can have multiple IP addresses, called aliases. To set an alias for your network card, proceed as follows:

- 1 Select a card from the list of detected cards in the YaST network card configuration module and click *Edit*.
- 2 In the *Address* tab, choose *Advanced* → *Additional Addresses*.
- 3 Click *Add*.
- 4 Enter *Alias Name*, *IP Address*, and *Netmask*.
- 5 Click *OK*.
- 6 Click *OK* again.
- 7 Click *Next*.
- 8 To activate the configuration, click *Next* again.

Configuring Hostname and DNS

If you did not change the network configuration during installation and the wired card was available, a hostname was automatically generated for your computer and DHCP was activated. The same applies to the name service information your host needs to integrate into a network environment. If DHCP is used for network address setup, the list of domain name servers is automatically filled with the appropriate data. If a static setup is preferred, set these values manually.

To change the name of your computer and adjust the name server search list, proceed as follows:

- 1 Select a card from the list of detected cards in the YaST network card configuration module and click *Edit*.
- 2 In the *Address* tab, click *Hostname and Name Server*.
- 3 To disable DHCP-driven host name configuration, deselect *Change Hostname via DHCP*.
- 4 Enter *Hostname* and, if it is needed, *Domain Name*.
- 5 To disable DHCP driven updates of the name server list, deselect *Update Name Servers and Search List via DHCP*.
- 6 Enter the name servers and domain search list.
- 7 Click *OK*.
- 8 Click *Next*.
- 9 To activate the configuration, click *Next* again.

Configuring Routing

To make your machine communicate with other machines and other networks, routing information must be given to make network traffic take the correct path. If DHCP is used, this information is automatically provided. If a static setup is used, this data must be added manually.

- 1 Select a card from the list of detected cards in the YaST network card configuration module and click *Edit*.
- 2 In the *Address* tab, click *Routing*.
- 3 Enter the IP of the *Default Gateway*.
- 4 Click *OK*.
- 5 Click *Next*.
- 6 To activate the configuration, click *Next* again.

Adding Special Hardware Options

Sometimes a module of a network card needs special parameters to work correctly. To set them with YaST, proceed as follows:

- 1 Select a card from the list of detected cards in the YaST network card configuration module and click *Edit*.
- 2 In the *Address* tab, click *Advanced* → *Hardware Details*.
- 3 In *Options*, enter the parameters for your network card. If two cards are configured that use the same module, these parameters are used for both.
- 4 Click *OK*.
- 5 Click *Next*.
- 6 To activate configuration, click *Next* again.

Starting the Device

You can configure your device to start during boot, on cable connection, on card detection, manually, or never. To change device start-up, proceed as follows:

- 1 Select a card from the list of detected cards in the YaST network card configuration module and click *Edit*.

- 2 In the *General* tab, select the desired entry from *Device Activation*.
- 3 Click *Next*.
- 4 To activate the configuration, click *Next* again.

Configuring the Firewall

Without having to enter the detailed firewall setup as described in [Section 38.4.1, “Configuring the Firewall with YaST”](#) (page 714), you can determine the basic firewall setup for your device as part of the device setup. Proceed as follows:

- 1 Select a card from the list of detected cards in the YaST network card configuration module and click *Edit*.
- 2 Enter the *General* tab of the network configuration dialog.
- 3 Determine the firewall zone to which your interface should be assigned. The following options are available:

Firewall Disabled

The firewall is not run at all. Only use this option if your machine part of a greater network that is protected by an outer firewall.

Internal Zone (Unprotected)

The firewall is run, but does not enforce any rules to protect this interface. Only use this option, if your machine part is part of a greater network that is protected by an outer firewall.

Demilitarized Zone

A demilitarized zone is an additional line of defense in front of an internal network and the (hostile) Internet. Hosts assigned to this zone can be reached from the internal network and from the Internet, but cannot access the internal network.

External Zone

The firewall is run on this interface and fully protects it against other (presumably hostile) network traffic. This is the default option.

- 4 Click *Next*.

- 5 Activate the configuration by clicking *Next* again.

Configuring an Undetected Network Card

It may happen that your card is not detected correctly. In this case, the card is not included in the list of the detected cards. If you are sure that your system includes a driver for your card, you can configure it manually. To configure an undetected network card, proceed as follows:

- 1 Click *Add*.
- 2 Set the *Device Type* of the interface from the available options, *Configuration Name*, and *Module Name*. If the network card is a PCMCIA or USB device, activate the respective check box and exit this dialog with *Next*. Otherwise, select your network card model from *Select from List*. YaST then automatically selects the appropriate kernel module for the card.

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 kernel module as well as the options needed to initialize the hardware.

- 3 Click *Next*.
- 4 In the *Address* tab, set the device type of the interface, the configuration name, and IP address. To use a static address, choose *Static Address Setup* then complete *IP Address* and *Subnet Mask*. Here, you can also select to configure the hostname, name server, and routing details (see [Section “Configuring Hostname and DNS”](#) (page 598) and [Section “Configuring Routing”](#) (page 598)).

If you selected *Wireless* as the device type of the interface, configure the wireless connection in the next dialog. Detailed information about wireless device configuration is available in [Section 28.1, “Wireless LAN”](#) (page 549).

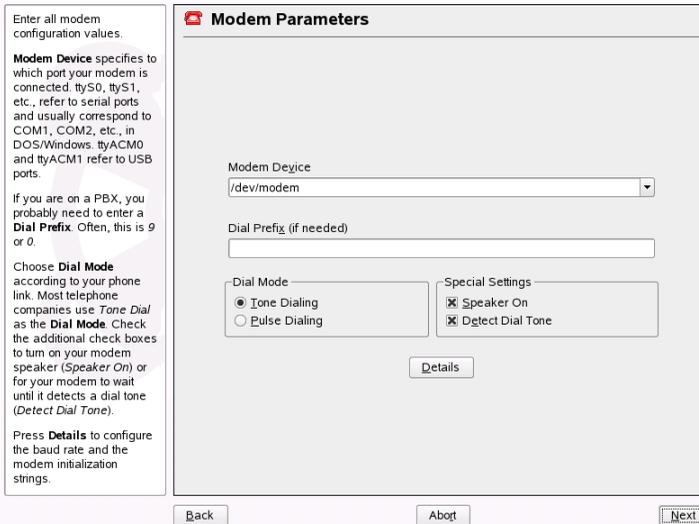
- 5 In the *General* tab, set the *Firewall Zone* and *Device Activation*. With *User Controlled*, grant connection control to ordinary users.
- 6 Click *Next*.
- 7 To activate the new network configuration, click *Next* again.

Information about the conventions for configuration names is available in the `getcfg(8)` man page.

29.4.2 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 29.4 *Modem Configuration*



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 detected automatically 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

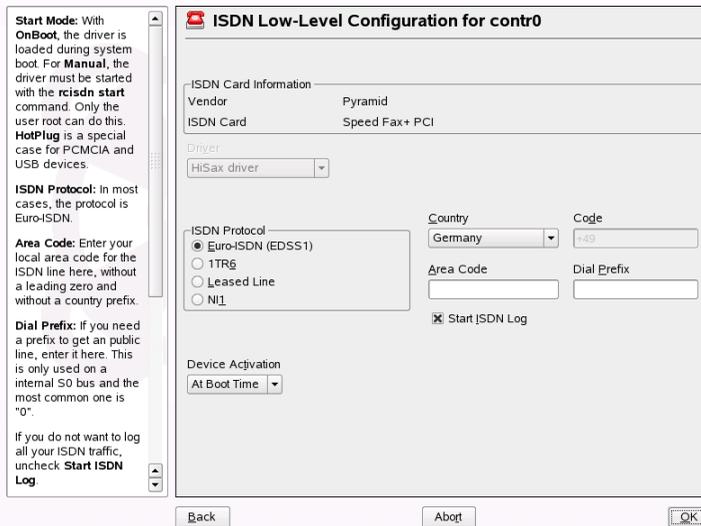
IP address and the remote IP address. Ask your ISP for this information. Leave *Default Route* enabled and close the dialog by selecting *OK*.

Selecting *Next* returns to the original dialog, which displays a summary of the modem configuration. Close this dialog with *Finish*.

29.4.3 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 29.5 *ISDN Configuration*

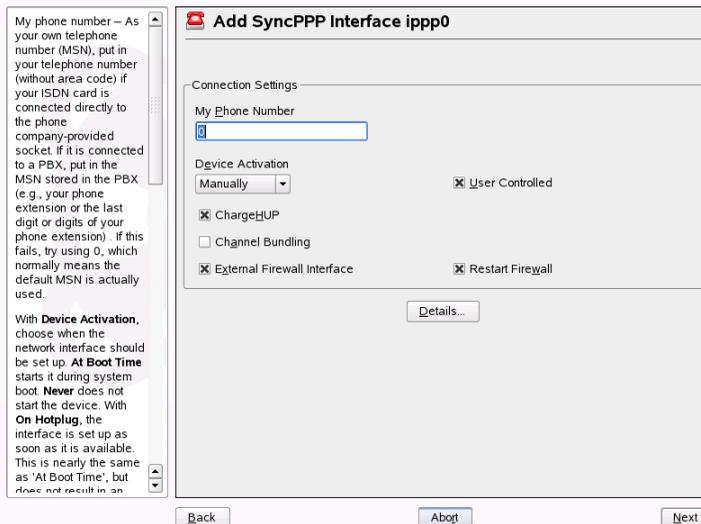


In the next dialog, shown in [Figure 29.5, “ISDN Configuration”](#) (page 604), select the protocol to use. The default is *Euro-ISDN (EDSS1)*, but for older or larger exchanges, select *ITR6*. If you are in the US, select *NII*. Select your country in the relevant field. The corresponding country code 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 29.6 *ISDN Interface Configuration*



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*.

29.4.4 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.

For further information about the configuration of cable modems, read the Support Database article on the topic, which is available online at http://en.opensuse.org/SDB:Setting_Up_an_Internet_Connection_via_Cable_Modem_with_SuSE_Linux_8.0_or_Higher.

29.4.5 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)

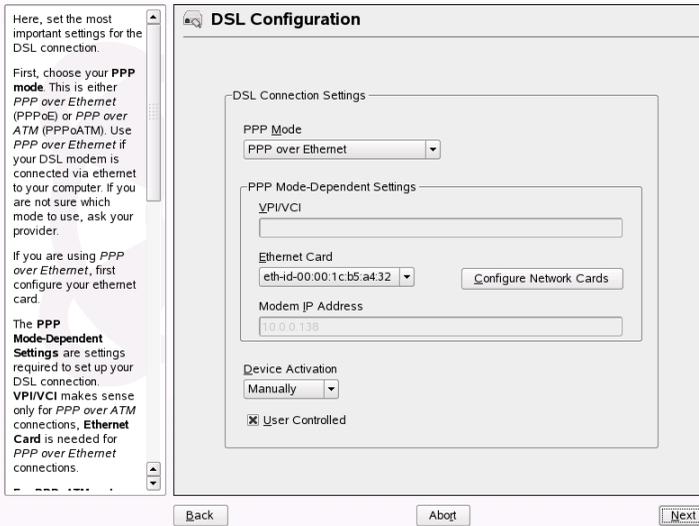
- Point-to-Point Tunneling Protocol (PPTP)—Austria

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 29.4.1, “Configuring the Network Card with YaST”](#) (page 594)). 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 29.7 *DSL Configuration*



To begin the DSL configuration (see [Figure 29.7, “DSL Configuration”](#) (page 608)), 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.

29.5 Managing Network Connections with NetworkManager

NetworkManager is the ideal solution for a mobile workstation. With NetworkManager, you do not need to worry about configuring network interfaces and switching between networks when you are moving. NetworkManager can automatically connect to known WLAN networks. If you have two or more connection possibilities, it can connect to the faster one.

NOTE: NetworkManager and SCPM

Do not use NetworkManager together with SCPM when SCPM profiles also change network settings. If you want to use SCPM and NetworkManager at the same time, disable the network resource in SCPM configuration.

NetworkManager is not a suitable solution in the following cases:

- You want to use more than one provider for dial-up for one interface.
- Your computer is a router for your network.
- Your computer provides network services for other computers in your network, for example, it is a DHCP or DNS server.

NOTE: NetworkManager and Network Devices Configured with YaST

If you switch your system from traditional configuration with YaST to NetworkManager controlled, NetworkManager adopts configurations from YaST.

29.5.1 Controlling NetworkManager

To start NetworkManager, enable NetworkManager in the YaST module of your network device. Because NetworkManager does not need standard network configuration, the YaST configuration becomes inactive. NetworkManager automatically chooses the best network available, but it can automatically connect only to a known network. For the first connection to a network, use the NetworkManager applet. If the network requires additional information, like its name, password or encryption key, NetworkManager prompts for it.

Both KDE and GNOME have their own applets for NetworkManager. An appropriate applet should start automatically with the desktop environment. The applet is then shown as an icon in the system tray. Functions of both applets are similar, but their interfaces are different. They can also be used in other graphical environments with standard system tray support.

The KNetworkManager Applet

KNetworkManager is a KDE applet for controlling NetworkManager. If it is not running start it with the command `knetworkmanager`. When it is running, a blue Earth icon is shown in the system tray. Right-clicking the icon opens the KNetworkManager menu with various commands for managing network connections.

The menu contains available network connections, both wired and wireless devices. If you hold the mouse cursor over them, details are displayed. The currently used connection is checked in the menu. The signal strength of wireless networks is also shown in the menu. Encrypted wireless networks are marked with a blue lock icon. To connect to an encrypted network, choose it from the menu. In the dialog that opens, choose the type of *Encryption* the network uses and enter the appropriate *Passphrase* or *Key*.

To connect to a network that does not broadcast its service set identifier (ESSID) and therefore cannot be detected automatically, choose *Connect to Other Wireless Network*. In the dialog that opens, enter the ESSID and set encryption parameters if necessary.

To access dial-up connections, choose *Dial-Up Connections*. When dial-up connections have already been defined, start the connection by clicking the connection to use. *Configure Dial-Up Connections* opens YaST where you can define a new dial-up connection.

To disable any active network connection, choose *Options* → *Switch to Offline mode* from the KNetworkManager menu. To reenablen the connection, choose *Options* → *Switch to Online mode*. To disable wireless network connections, choose *Options* → *Disable Wireless* from the KNetworkManager menu. To reenablen wireless connections, choose *Options* → *Enable Wireless*. Enabling networking takes a few seconds.

The GNOME NetworkManager Applet

GNOME also has its own applet for NetworkManager. If it is not running, start it with the command `nm-applet`. When it is running, an icon is shown in the system tray. The appearance of the icon depends on state of the network connection. If you are not sure what the icon means, hold the mouse cursor over it until an explanation appears.

Left-click the applet icon to show a menu with available networks. The currently used connection is checked in the menu. The signal strength of wireless networks is also shown in the menu. Encrypted wireless networks are marked with a shield icon. To connect to an encrypted network, choose it from the menu. In the dialog that opens,

choose the type of *Encryption* the network uses and enter the appropriate *Passphrase* or *Key*.

To connect to a network that does not broadcast its service set identifier (ESSID) and therefore cannot be detected automatically, left-click the icon and choose *Connect to Other Wireless Network*. In the dialog that opens, enter the ESSID and set encryption parameters if necessary.

To disable networking, right-click the applet icon and uncheck *Enable Networking*. To disable wireless networking, right-click the applet icon and uncheck *Enable Wireless*.

To get information about the current connection (including interface used, IP address, and hardware address), right-click the applet icon and choose *Connection information* from the menu.

29.5.2 For More Information

More information about NetworkManager and D-BUS can be found on the following Web sites and directories:

- <http://www.gnome.org/projects/NetworkManager/>—NetworkManager project page
- <http://www.freedesktop.org/Software/dbus>—D-BUS project page
- `/usr/share/doc/packages/NetworkManager`

29.6 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 29.5, “Manual Network Configuration Scripts” (page 615) summarizes the most important scripts involved in the network configuration. Where possible, the scripts are distinguished by hardware and interface.

Table 29.5 *Manual Network Configuration Scripts*

Configura- tion Stage	Command	Function
Hardware	<code>hw{up,down,status}</code>	The <code>hw*</code> 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 <code>hwup</code> .
Interface	<code>getcfg</code>	<code>getcfg</code> can be used to query the interface name associated with a configuration name or a hardware description. More information is available in the manual page of <code>getcfg</code> .
Interface	<code>if{up,down,status}</code>	The <code>if*</code> scripts start existing network interfaces or return the status of the specified interface. More information is available in the manual page of <code>ifup</code> .

More information about hotplug and persistent device names is available in [Chapter 20, *Dynamic Kernel Device Management with udev*](#) (page 445).

29.6.1 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 29.5](#), “`/etc/resolv.conf`” (page 617) shows what `/etc/resolv.conf` could look like.

Example 29.5 `/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`), `ippd` (`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 29.6](#), “`/etc/hosts`” (page 618), 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 29.6 `/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 29.7](#), “`/etc/networks`” (page 618).

Example 29.7 `/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 29.6](#), “Parameters for `/etc/host.conf`” (page 619) shows the parameters available. A sample `/etc/host.conf` is shown in [Example 29.8](#), “`/etc/host.conf`” (page 619).

Table 29.6 *Parameters for `/etc/host.conf`*

<code>order hosts, bind</code>	Specifies in which order the services are accessed for the name resolution. Available arguments are (separated by blank spaces or commas): <code>hosts</code> : Searches the <code>/etc/hosts</code> file <code>bind</code> : Accesses a name server <code>nis</code> : Uses NIS
<code>multi on/off</code>	Defines if a host entered in <code>/etc/hosts</code> can have multiple IP addresses.
<code>nospoof on</code> <code>spoofalert on/off</code>	These parameters influence the name server <i>spoofing</i> , but, apart from that, do not exert any influence on the network configuration.
<code>trim domainname</code>	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 <code>/etc/hosts</code> file, but should still be recognized with the attached domain names.

Example 29.8 *`/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 29.9](#), “`/etc/nsswitch.conf`” (page 620). 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.

Example 29.9 `/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 29.7](#), “Databases Available via `/etc/nsswitch.conf`” (page 620). In addition, `automount`, `bootparams`, `netmasks`, and `publickey` are expected in the near future. The configuration options for NSS databases are listed in [Table 29.8](#), “Configuration Options for NSS “Databases”” (page 621).

Table 29.7 *Databases Available via `/etc/nsswitch.conf`*

<code>aliases</code>	Mail aliases implemented by <code>sendmail</code> ; see man 5 <code>aliases</code> .
<code>ethers</code>	Ethernet addresses.
<code>group</code>	For user groups, used by <code>getgrent</code> . See also the man page for <code>group</code> .

hosts	For hostnames and IP addresses, used by <code>gethostbyname</code> and similar functions.
netgroup	Valid host and user lists in the network for the purpose of controlling access permissions; see the <code>netgroup(5)</code> man page.
networks	Network names and addresses, used by <code>getnetent</code> .
passwd	User passwords, used by <code>getpwent</code> ; see the <code>passwd(5)</code> man page.
protocols	Network protocols, used by <code>getprotoent</code> ; see the <code>protocols(5)</code> man page.
rpc	Remote procedure call names and addresses, used by <code>getrpcbyname</code> and similar functions.
services	Network services, used by <code>getservent</code> .
shadow	Shadow passwords of users, used by <code>getspnam</code> ; see the <code>shadow(5)</code> man page.

Table 29.8 *Configuration Options for NSS “Databases”*

files	directly access files, for example, <code>/etc/aliases</code>
db	access via a database
nis, nisplus	NIS, see also Chapter 32, Using NIS (page 641)
dns	can only be used as an extension for <code>hosts</code> and <code>networks</code>
compat	can only be used as an extension for <code>passwd</code> , <code>shadow</code> , and <code>group</code>

`/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.

29.6.2 Testing the Configuration

Before you write your configuration to the configuration files, you can test it. To set up a test configuration, use the `ip` command. To test the connection, use the `ping` command. Older configuration tools, `ifconfig` and `route`, are also available.

The commands `ip`, `ifconfig`, and `route` change the network configuration directly without saving it in the configuration file. Unless you enter your configuration in the correct configuration files, the changed network configuration is lost on reboot.

Configuring a Network Interface with `ip`

`ip` is a tool to show and configure routing, network devices, policy routing, and tunnels. It was designed as a replacement for the older tools `ifconfig` and `route`.

`ip` is very complex tool. Its common syntax is `ip options object command`. You can work with the following objects:

link

This object represents a network device.

address

This object represents the IP address of device.

neighbour

This object represents a ARP or NDISC cache entry.

route

This object represents the routing table entry.

rule

This object represents a rule in the routing policy database.

maddress

This object represents a multicast address.

mroute

This object represents a multicast routing cache entry.

tunnel

This object represents a tunnel over IP.

If no command is given, the default command is used, usually `list`.

Change the state of a device with the command `ip link set device_name command`. For example, to deactivate device `eth0`, enter `ip link set eth0 down`. To activate it again, use `ip link set eth0 up`.

After activating a device, you can configure it. To set the IP address, use `ip addr add ip_address + dev device_name`. For example, to set the address of the interface `eth0` to `192.168.12.154/30` with standard broadcast (option `brd`), enter `ip addr add 192.168.12.154/30 brd + dev eth0`.

To have a working connection, you must also configure the default gateway. To set gateway for your system, enter `ip route get gateway_ip_address`. To translate one IP address to another, use `nat:ip route add nat ip_address via other_ip_address`.

To display all devices, use `ip link ls`. To display only running interfaces, use `ip link ls up`. To print interface statistics for a device, enter `ip -s link ls device_name`. To view addresses of your devices, enter `ip addr`. In the output of the `ip addr`, also find information about MAC addresses of your devices. To show all routes, use `ip route show`.

For more information about using `ip`, enter `ip help` or see the `ip(8)` man page. The `help` option is also available for all `ip` objects. If, for example, you want to read help for `ip addr`, enter `ip addr help`. Find the `ip` manual in `/usr/share/doc/packages/iproute2/ip-cref.pdf`.

Testing a Connection with ping

The `ping` command is the standard tool for testing whether a TCP/IP connection works. It uses the ICMP protocol to send a small data packet, `ECHO_REQUEST` datagram, 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.

`ping` does more than test only the function of the connection between two computers: it also provides some basic information about the quality of the connection. In [Example 29.10](#), “[Output of the Command ping](#)” (page 625), you can see an example of the `ping` output. The second-to-last line contains information about number of transmitted packets, packet loss, and total time of `ping` running.

As the destination, you can use a hostname or IP address, for example, `ping example.com` or `ping 130.57.5.75`. The program sends packets until you press `Ctrl` + `C`.

If you only need to check the functionality of the connection, you can limit the number of the packets with the `-c` option. For example to limit `ping` to three packets, enter `ping -c 3 192.168.0.`

Example 29.10 *Output of the Command ping*

```
ping -c 3 example.com
  PING example.com (130.57.5.75) 56(84) bytes of data.
  64 bytes from example.com (130.57.5.75): icmp_seq=1 ttl=49 time=188
ms
  64 bytes from example.com (130.57.5.75): icmp_seq=2 ttl=49 time=184
ms
  64 bytes from example.com (130.57.5.75): icmp_seq=3 ttl=49 time=183
ms

  --- example.com ping statistics ---
  3 packets transmitted, 3 received, 0% packet loss, time 2007ms
  rtt min/avg/max/mdev = 183.417/185.447/188.259/2.052 ms
```

The default interval between two packets is one second. To change the interval, ping provides option `-i`. For example to increase ping interval to ten seconds, enter `ping -i 10 192.168.0`.

In a system with multiple network devices, it is sometimes useful to send the ping through a specific interface address. To do so, use the `-I` option with the name of the selected device, for example, `ping -I wlan1 192.168.0`.

For more options and information about using ping, enter `ping -h` or see the ping (8) man page.

Configuring the Network with ifconfig

`ifconfig` is a traditional network configuration tool. In contrast to `ip`, you can use it only for interface configuration. If you want to configure routing, use `route`.

NOTE: ifconfig and ip

The program `ifconfig` is obsolete. Use `ip` instead.

Without arguments, `ifconfig` displays the status of the currently active interfaces. As you can see in [Example 29.11, “Output of the ifconfig Command”](#) (page 626), `ifconfig` has very well-arranged and detailed output. The output also contains information about the MAC address of your device, the value of `HWaddr`, in the first line.

Example 29.11 Output of the ifconfig Command

```
eth0      Link encap:Ethernet  HWaddr 00:08:74:98:ED:51
          inet6 addr: fe80::208:74ff:fe98:ed51/64 Scope:Link
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:634735 errors:0 dropped:0 overruns:4 frame:0
          TX packets:154779 errors:0 dropped:0 overruns:0 carrier:1
          collisions:0 txqueuelen:1000
          RX bytes:162531992 (155.0 Mb)  TX bytes:49575995 (47.2 Mb)
          Interrupt:11 Base address:0xec80

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:8559 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8559 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:533234 (520.7 Kb)  TX bytes:533234 (520.7 Kb)

wlan1     Link encap:Ethernet  HWaddr 00:0E:2E:52:3B:1D
          inet addr:192.168.2.4  Bcast:192.168.2.255  Mask:255.255.255.0
          inet6 addr: fe80::20e:2eff:fe52:3b1d/64 Scope:Link
          UP BROADCAST NOTRAILERS RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:50828 errors:0 dropped:0 overruns:0 frame:0
          TX packets:43770 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:45978185 (43.8 Mb)  TX bytes:7526693 (7.1 Mb)
```

For more options and information about using `ifconfig`, enter `ifconfig -h` or see the `ifconfig (8)` man page.

Configuring Routing with route

`route` is a program for manipulating the IP routing table. You can use it to view your routing configuration and add or remove of routes.

NOTE: route and ip

The program `route` is obsolete. Use `ip` instead.

`route` is especially useful if you need quick and comprehensible information about your routing configuration to determine problems with routing. To view your current routing configuration, enter `route -n` as `root`.

Example 29.12 Output of the route -n Command

```
route -n
Kernel IP routing table
Destination      Gateway          Genmask         Flags   MSS Window  irtt
Iface
 10.20.0.0        *                255.255.248.0   U           0 0        0
eth0
 link-local       *                255.255.0.0     U           0 0        0
eth0
 loopback         *                255.0.0.0       U           0 0        0 lo

 default          styx.exam.com    0.0.0.0         UG          0 0        0
eth0
```

For more options and information about using route, enter `route -h` or see the `route (8) man` page.

29.6.3 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 29.9, “Some Start-Up Scripts for Network Programs”](#) (page 627).

Table 29.9 *Some Start-Up Scripts for Network Programs*

<code>/etc/init.d/network</code>	This script handles the configuration of the network interfaces. The hardware must already have been initialized by <code>/etc/init.d/coldplug</code> (via <code>hotplug</code>). If the <code>network</code> service was not started, no network interfaces are implemented when they are inserted via <code>hotplug</code> .
<code>/etc/init.d/inetd</code>	Starts <code>xinetd</code> . <code>xinetd</code> can be used to make server services available on the system. For example, it can start <code>vsftpd</code> whenever an FTP connection is initiated.
<code>/etc/init.d/portmap</code>	Starts the portmapper needed for the RPC server, such as an NFS server.

<code>/etc/init.d/nfsserver</code>	Starts the NFS server.
<code>/etc/init.d/sendmail</code>	Controls the sendmail process.
<code>/etc/init.d/ypserv</code>	Starts the NIS server.
<code>/etc/init.d/ypbind</code>	Starts the NIS client.

29.7 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 `ippd` 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 `ippd` 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.

29.7.1 Configuring smpppd

The connections provided by `smpppd` are automatically configured by YaST. The actual dial-up programs `KInternet` and `cinetnet` 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.

29.7.2 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 basically 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.

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 Enterprise 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, file server, or print server on your SUSE Linux Enterprise.

30.1 Registering Your Own Services

Many applications under SUSE Linux Enterprise 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.

30.2 SLP Front-Ends in SUSE Linux Enterprise

SUSE Linux Enterprise 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. `slp` 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.

30.3 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 Enterprise, 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.

30.4 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 Enterprise 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*.

Time Synchronization with NTP

31

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.

31.1 Configuring an NTP Client with YaST

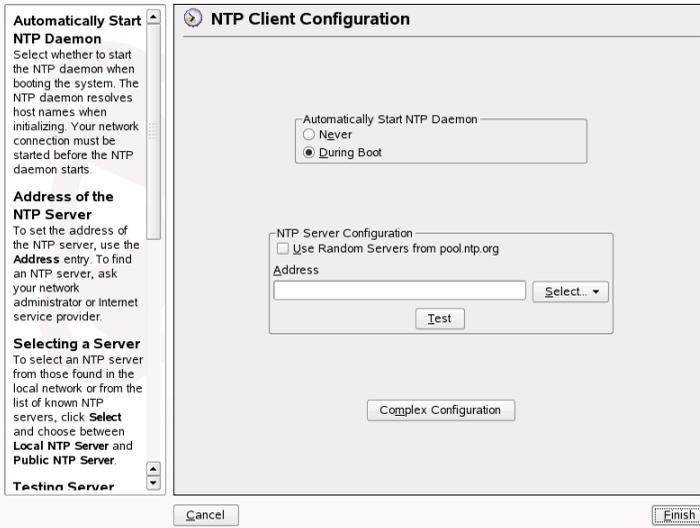
`xntp` is preset to use the local computer clock as a time reference. Using the (BIOS) clock, however, only serves as a fallback for the case that no time source of greater precision is available. SUSE® Linux Enterprise facilitates the configuration of an NTP client with YaST. Use the quick or complex configuration for clients that do not run the

SuSEfirewall because they are part of a protected intranet. Both are described in the following.

31.1.1 Quick NTP Client Configuration

The quick 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 specify the *NTP Server Configuration*. Either click *Use Random Servers from pool.ntp.org* if you cannot use a local time server or click *Select* to access a second dialog in which to select a suitable time server for your network.

Figure 31.1 *YaST: Configuring an NTP Client*



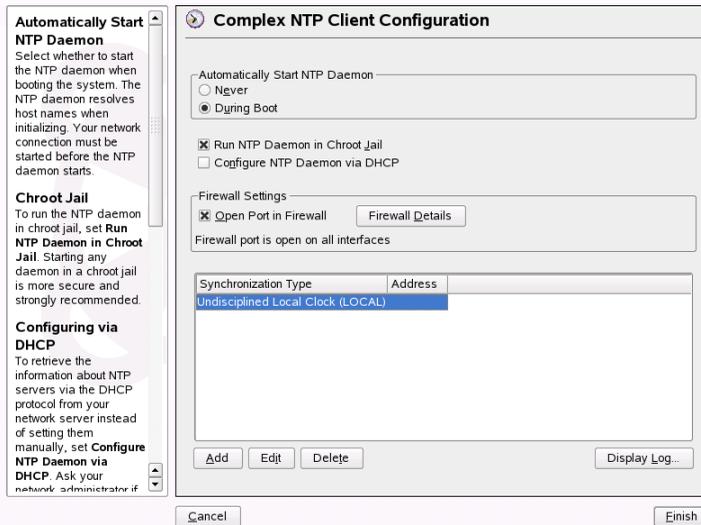
In the detailed server selection dialog, determine whether to implement time synchronization using a time server from your local network (*Local NTP Server*) or an Internet-based time server that takes care of your time zone (*Public NTP Server*). For a local time server, click *Lookup* to start an SLP 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*.

31.1.2 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 31.1](#), “[YaST: Configuring an NTP Client](#)” (page 636), after selecting the start-up mode as described in the quick configuration.

Figure 31.2 *YaST: Complex NTP Client Configuration*



In *Complex NTP Client Configuration*, determine whether `xntpd` should be started in a chroot jail. By default, *Run NTP Daemon in Chroot Jail* is activated. 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 31.1.1, “Quick NTP Client Configuration”](#) (page 636)). 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.

31.2 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 `rcntpd 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`.

31.3 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.

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 36, *Sharing File Systems with NFS*](#) (page 681).

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.”

32.1 Configuring NIS Clients

Use the module *NIS Client* to configure a workstation to use NIS. Select whether the host has a static IP address or receives one issued by DHCP. DHCP can also provide the NIS domain and the NIS server. If a static IP address is used, specify the NIS domain and the NIS server manually. See [Figure 32.1, “Setting Domain and Address of a NIS Server”](#) (page 642). *Find* makes YaST search for an active NIS server in your whole network. Depending on the size of your local network, this may be a time-consuming process. *Broadcast* asks for a NIS server in the local network after the specified servers fail to respond.

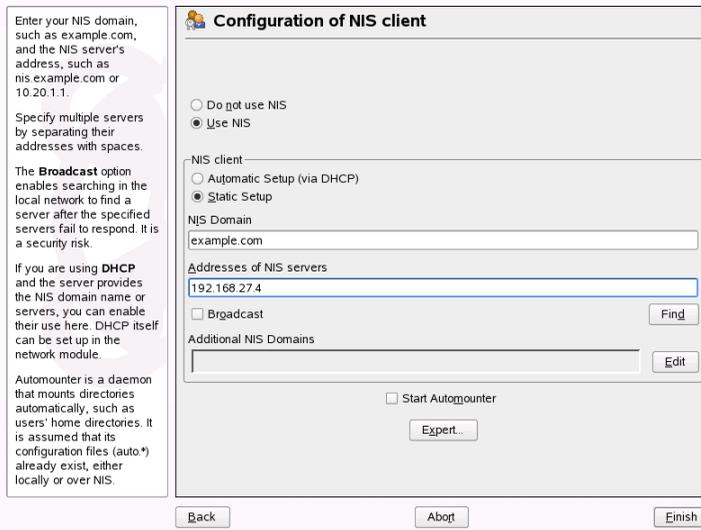
You can also specify multiple servers by entering their addresses in *Addresses of NIS servers* and separating them by spaces.

Depending on your local installation, you may also want to activate the automounter. This option also installs additional software if required.

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 32.1 *Setting Domain and Address of a NIS Server*



Configuring eDirectory Authentication

33

You can use Novell® Linux User Management (LUM) to configure SUSE® Linux Enterprise Desktop workstations on your network so that users can log in to them using their Novell eDirectory™ usernames and passwords instead of their local Linux workstation usernames and passwords. Using LUM and eDirectory to manage user login information eliminates the need to create local users in the `/etc/passwd` and `/etc/shadow` files on each SUSE Linux Enterprise Desktop workstation. It also simplifies user account management by consolidating user accounts into a central point of administration.

You can use eDirectory tools and technologies to manage access to Linux resources on the network. After authenticating, users have the rights and privileges specified in eDirectory. These are the same rights and privileges that are typically stored in a local account or redirected to other authentication methods, such as NIS. The user account information stored in eDirectory lets users access file and printer resources on your network.

Users can log in to SUSE Linux Enterprise Desktop workstations using access methods such as `login`, `ftp`, `ssh`, `su`, `rsh`, `rlogin`, `xdm` (KDE), and `gdm` (GNOME). They only need to enter their familiar eDirectory username and password. They do not have to remember their full context—LUM searches out the correct user in eDirectory.

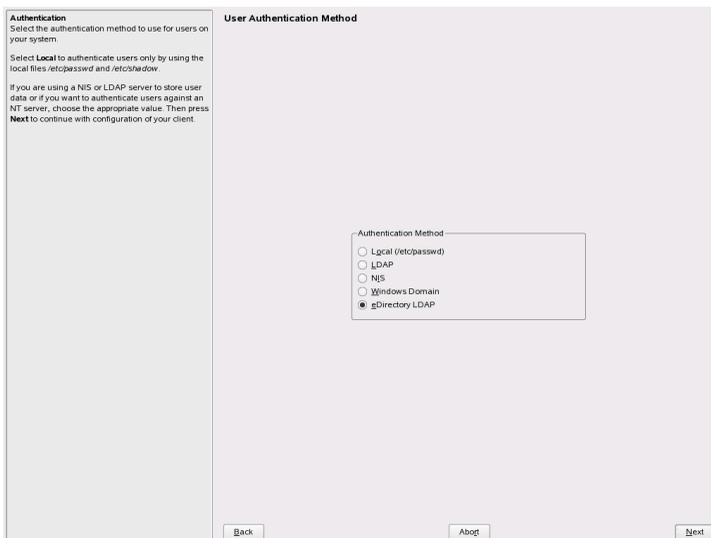
This chapter guides you through the steps required to set up a SUSE Linux Enterprise Desktop workstation to use eDirectory authentication, which includes configuring the SUSE Linux Enterprise Desktop workstation for eDirectory authentication and enabling users on the eDirectory server. For more detailed information on LUM and on configuring your eDirectory 8.6.x, 8.7.x, or 8.8.x server to use LUM, see the Novell Linux

33.1 Setting Up Workstations to Use eDirectory Authentication

Before users can use their eDirectory usernames and passwords to log in, the SUSE Linux Enterprise Desktop workstation must be configured with Linux User Management components. You can set up eDirectory Authentication during the SUSE Linux Enterprise Desktop installation, or you can use YaST to set it up anytime after installation.

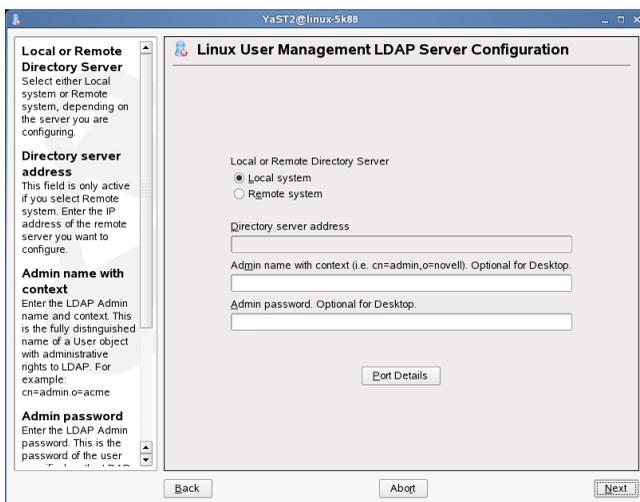
To install and configure LUM during the SUSE Linux Enterprise Desktop installation, select *eDirectory LDAP* as the authentication method on the User Authenticating Method screen, then complete [Step 3](#) (page 645) through [Step 10](#) (page 647) below. If it is not already installed, you will be prompted to install the `yast2-linux-user-mgmt` package.

Figure 33.1 *User Authentication Method Screen in the SUSE Linux Enterprise Desktop Installation*



To install and configure LUM on an already running workstation:

- 1 On the workstation, launch the YaST Control Center.
GNOME: Click *Computer* → *More Applications* → *YaST Control Center*.
KDE: Click *the menu button* → *System* → *YaST (Control Center)*.
- 2 Click *Security and Users* → *Linux User Management*.
- 3 Specify whether eDirectory is running on the computer itself (*Local System*) or on another computer on the network (*Remote System*).



- 4 If eDirectory is running on a remote system, specify the remote system's IP address.
- 5 (Optional) Specify the eDirectory admin name, context, and password, then click *Next*.

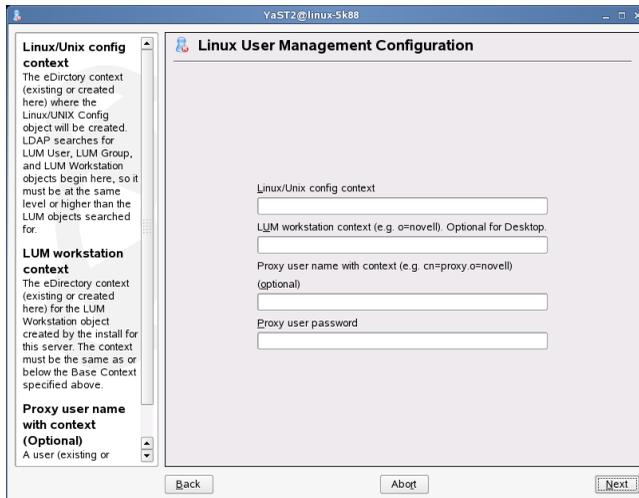
The admin name and context must be entered in LDAP syntax which uses a comma instead of a period (for example, cn=admin,o=novell).

IMPORTANT

If you don't have rights to create objects in the eDirectory tree, leave these fields blank. You will need to contact your eDirectory administrator, give them the host name of your client, and ask them to create a LUM Workstation object with your host name. You should also ask them where you can get a copy of the CA certificate for the LDAP server. You will need to place this certificate in the `/var/lib/novell-lum` directory.

The name of the CA certificate matches the name of the “preferred-server” entry in the `/etc/nam.conf` file and has a `.der` extension. You can type `namconfig get preferred-server` to get the name. For example, if `namconfig get preferred-server` returns `server.xyz.com`, your certificate file name will be `.server.xyz.com.der`.

6 Specify the location of the Linux/UNIX Config object.



The Linux/UNIX Config object stores a list of the locations (contexts) where Linux/UNIX Workstation objects reside on the network. It also controls the range of numbers to be assigned as UIDs and GIDs when User and Group objects are created. This object is created when LUM is configured on the

eDirectory server, and is usually located in an upper container of the eDirectory tree (for example, o=novell). Contact your eDirectory administrator for the context.

For more information, see “Understanding eDirectory Objects and Linux” [<http://www.novell.com/documentation/oes/lumadgd/data/bx3sbv9.html>] in the *Novell Linux User Management Technology Guide* .

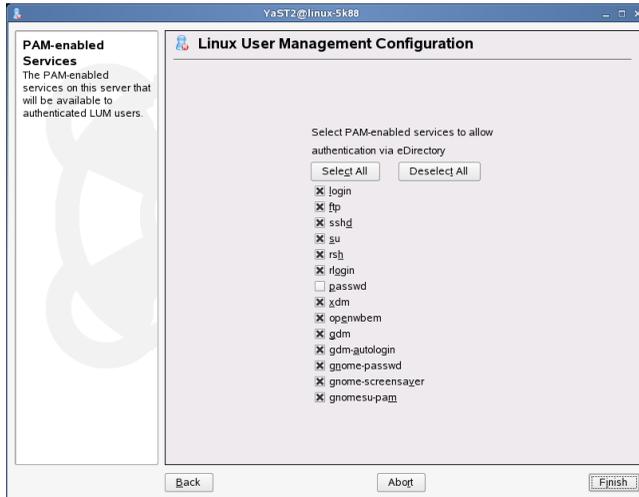
7 (Optional) Specify the location of the LUM Workstation object.

The LUM Workstation object represents the actual computer a user logs in to. If you have rights to create objects in the eDirectory tree (that is, you were able to specify the eDirectory admin name, context, and password in [Step 5](#) (page 645)), this object is automatically created as part of the workstation configuration and is usually placed in an Organization (O) or Organizational Unit (OU) container in the eDirectory tree. You can also create a LUM Workstation object by clicking *Linux User Management* → *Create Linux Workstation Object* in iManager.

8 (Optional) If you have disabled anonymous binds to the LDAP server, specify a proxy user name, context, and password that has rights to the LDAP tree.

9 Click *Next* to continue.

10 Select which login access methods should use eDirectory for authentication.



11 Click *Finish*.

Installing and configuring LUM technology sets up the SUSE Linux Enterprise Desktop workstation to validate login requests against user account information stored in eDirectory. Before users can log in, they must have eDirectory user accounts created with iManager and extended for LUM, and their User object must be associated with the workstation they will log in to. See [Section 33.2, “Using iManager to Enable Users for eDirectory Authentication”](#) (page 648) for more information.

33.2 Using iManager to Enable Users for eDirectory Authentication

When Linux User Management components are properly installed, you can use eDirectory and iManager to specify which users can access SUSE Linux Enterprise Desktop computers on the network. iManager is the browser-based utility for managing eDirectory objects. It runs in a network browser such as Mozilla* Firefox*, Netscape* Navigator*, or Internet Explorer.

When you create user or group accounts in iManager, you are prompted to “LUM enable” the User object or Group object. You can also use iManager to enable existing User or Group objects for Linux.

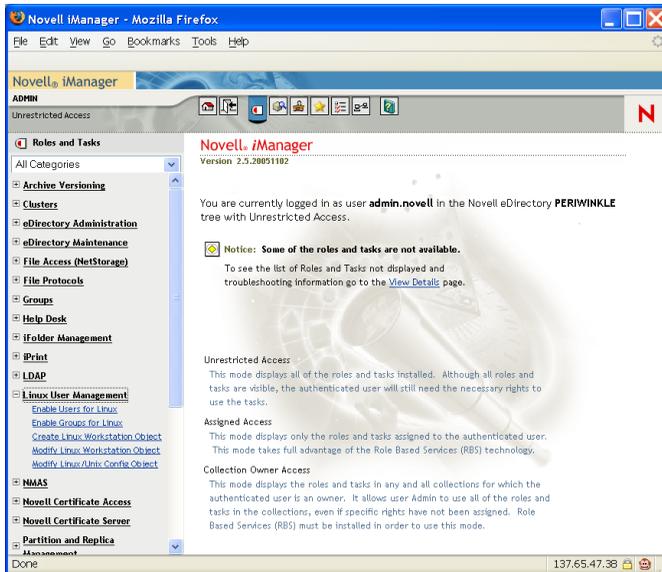
Each time you configure a SUSE Linux Enterprise Desktop workstation for eDirectory authentication, eDirectory users that are LUM enabled must be associated with a workstation before they can log in from that workstation.

- 1 Launch iManager by entering the following in the Address field of a network browser:

```
http://target_server/nps/iManager
```

where *target_server* is the IP address or domain name of the eDirectory server. You are prompted to provide the full context of the admin user (for example, admin.novell) and password.

- 2 Make sure you are in the Roles and Tasks view by clicking  on the top button bar, then select *Linux User Management* in the navigation panel on the left.



- 3 Click *Enable Users for Linux*, select the User object you want to enable, then click *Next*.

When an eDirectory User object is extended to hold Linux user-login properties, it is said to be *LUM enabled* or *enabled for Linux*. When enabled for Linux, a user can simply access the Linux computer using Telnet, SSH, or other supported methods (see [Step 10](#) (page 647)) and enter their username and password. The access request is redirected to find the appropriate username and login information stored in eDirectory.

When extended for Linux, the eDirectory User object holds Linux-related properties, such as user ID, primary group ID, primary group name, location of home directory, and preferred shell.

- 4 Assign the user to a group, then click *Next*.

The group and its corresponding group ID (GID) are assigned as the user's primary GID. If the selected user account already has a primary GID, this group's GID is assigned to the user as secondary. You can choose any of the following ways to assign the user to a group:

- **An Existing eDirectory Group:** If the Group object has not yet been enabled for Linux, its properties are extended to include Linux login attributes. You can click the Object Selector icon to browse the tree for an existing group.
- **An Existing Linux-Enabled Group:** This option lets you select an existing eDirectory Group object, but if you use the Object Selector to browse, you can view and select only those Group objects already extended with Linux login attributes.
- **Create a New Linux-Enabled Group:** This option lets you create a new eDirectory Group object. When created, the Group object is extended to include Linux login attributes.

- 5 Select the workstations that the users in the group should have access to, then click *Next*.
- 6 Click *Finish* to apply the changes, then click *OK*.

Users should now be able to use their eDirectory user login credentials to log in to their SUSE Linux Enterprise Desktop workstations.

33.3 Turning Off LUM and eDirectory Authentication

There might be times when you want to turn off a workstation's ability to accept logins from eDirectory. You can permanently turn off this ability by removing the LUM software from the workstation. You can temporarily disable eDirectory authentication by stopping the `named` daemon.

To stop `named`, open a shell window and enter `rcnamed stop`.

To turn on eDirectory authentication and LUM, open a shell window and enter `rcnamed start`.

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, such as 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 used to administer 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.

34.1 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. This means it is not suitable as a centralized data administration tool in heterogeneous networks.

Unlike NIS, the LDAP service is not restricted to pure Unix networks. Windows servers (from 2000) support LDAP as a directory 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 more easily.

34.2 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

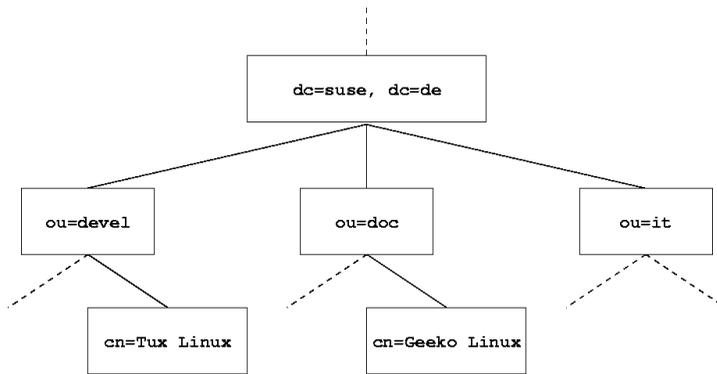
These objects can themselves contain other objects. Such object classes are `root` (the root element of the directory tree, which does not really exist), `c` (country), `ou` (organizational unit), and `dc` (domain component). This model is comparable to the directories (folders) in a file system.

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 34.1, “Structure of an LDAP Directory”](#) (page 656).

Figure 34.1 Structure of an LDAP Directory



The complete diagram is 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 types of objects that should be stored in the DIT are globally determined following a *schema*. 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 schema, 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 34.1, “Commonly Used Object Classes and Attributes”](#) (page 657) 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 34.1 Commonly Used Object Classes and Attributes

Object Class	Meaning	Example Entry	Required Attributes
dcObject	<i>domainComponent</i> (name components of the domain)	suse	dc
organizationalUnit	<i>organizationalUnit</i> (organizational unit)	doc	ou
inetOrgPerson	<i>inetOrgPerson</i> (person-related data for the intranet or Internet)	Geeko Linux	sn and cn

[Example 34.1, “Excerpt from schema.core”](#) (page 657) shows an excerpt from a scheme directive with explanations (line numbering for explanatory reasons).

Example 34.1 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 a 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`.

34.3 Configuring an LDAP Client with YaST

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 and installs the necessary files.

34.3.1 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 34.2, “pam_unix2.conf Adapted to LDAP”](#) (page 659)).

Example 34.2 *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. Find more about the workings of `nsswitch.conf` in [Section 29.6.1, “Configuration Files”](#) (page 615). The following lines must be present in `nsswitch.conf` for user administration and authentication with LDAP. See [Example 34.3, “Adaptations in nsswitch.conf”](#) (page 659).

Example 34.3 *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`.

34.3.2 Configuring the LDAP Client

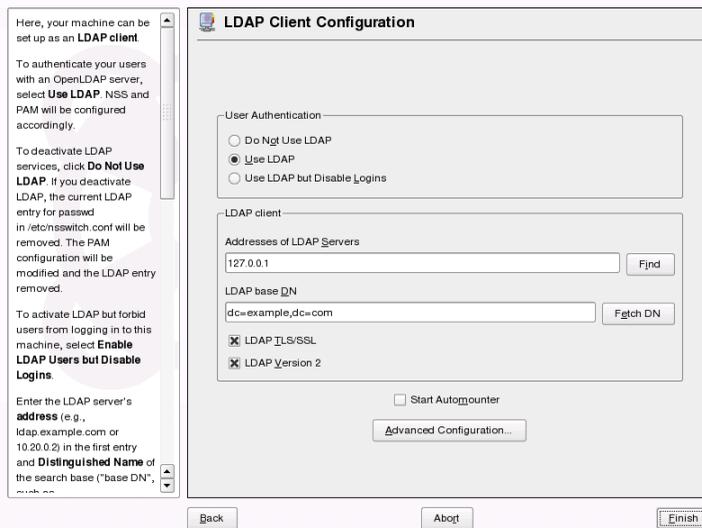
After the initial adjustments of `nss_ldap`, `pam_ldap`, `/etc/passwd`, and `/etc/group` have been taken care of by YaST, you can simply connect your client to the server and let YaST manage users over LDAP. This basic setup is described in [Section “Basic Configuration”](#) (page 660).

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 663).

Basic Configuration

The basic LDAP client configuration dialog ([Figure 34.2, “YaST: Configuration of the LDAP Client”](#) (page 660)) 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.

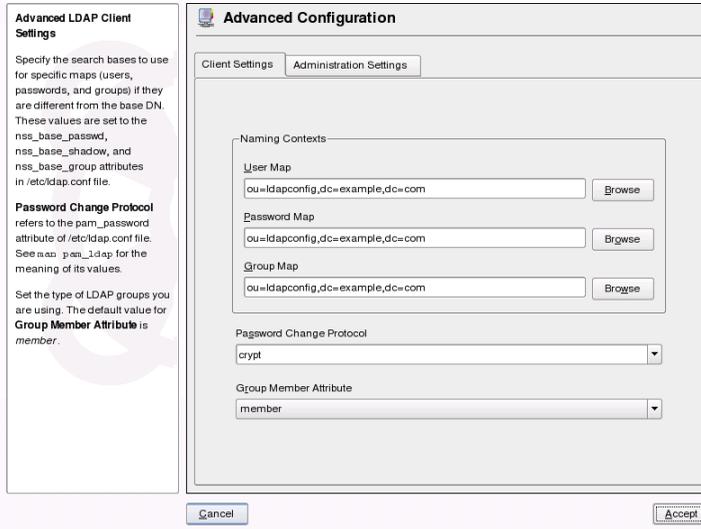
Figure 34.2 *YaST: Configuration of the LDAP Client*



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. To retrieve the base DN automatically, click *Fetch DN*. YaST then checks for any LDAP database on the server address specified above. Choose the appropriate base DN from the search results given by YaST.
- 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 34.3 *YaST: Advanced Configuration*



To modify data on the server as administrator, click *Advanced Configuration*. The following dialog is split in two tabs. See [Figure 34.3, “YaST: Advanced Configuration”](#) (page 662).

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:

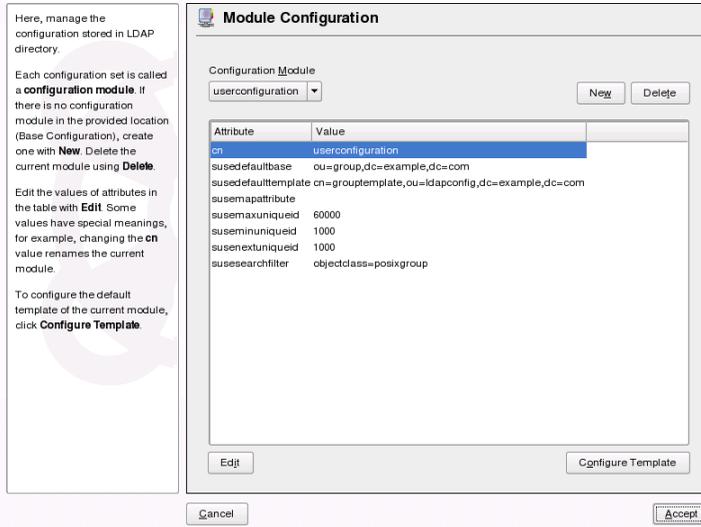
- a** Set the base for storing your user management data via *Configuration Base DN*.
- b** Enter the appropriate value for *Administrator DN*. This DN must be identical with the `rootdn` value specified in `/etc/openldap/slapd.conf` to enable this particular user to manipulate data stored on the LDAP server. Enter the full DN (such as `cn=admin,dc=suse,dc=de`) or activate *Append Base DN* to have the base DN added automatically when you enter `cn=admin`.
- 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 663).

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 34.4 *YaST: Module Configuration*



The dialog for module configuration (Figure 34.4, “YaST: Module Configuration” (page 664)) allows the creation of new modules, selection and modification of existing configuration modules, and design and modification of templates for such modules.

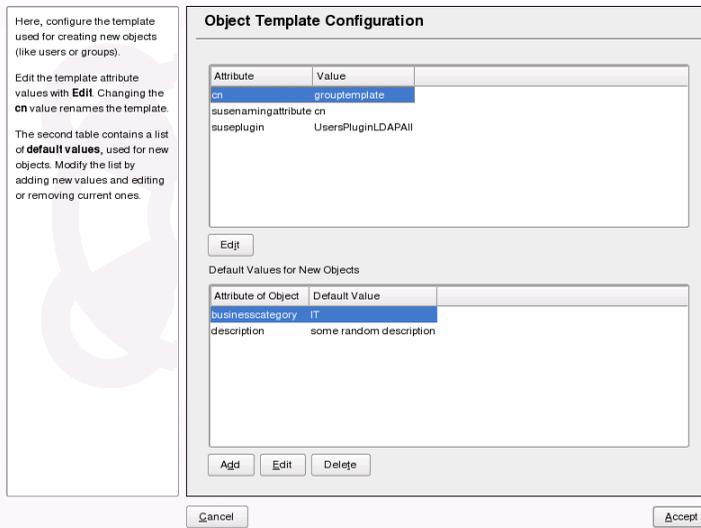
To create a new configuration module, proceed as follows:

- 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 *Accept*, 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 34.5 *YaST: Configuration of an Object Template*



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`.

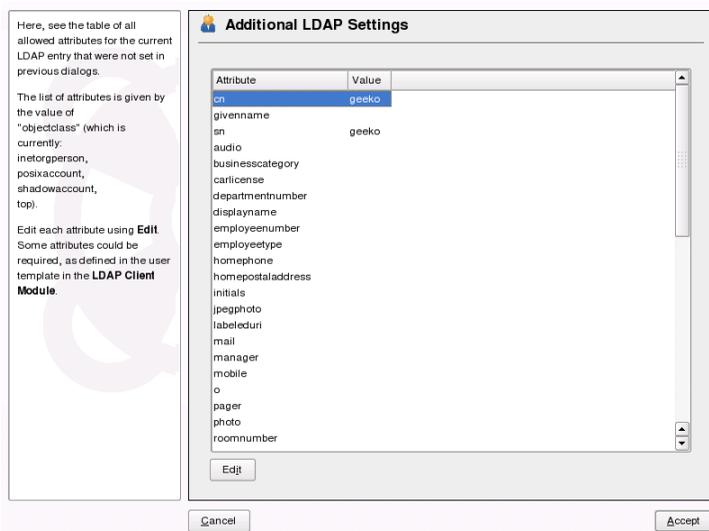
Once all modules and templates are configured correctly and ready to run, new groups and users can be registered in the usual way with YaST.

34.4 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 663).
 - 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 34.6, “YaST: Additional LDAP Settings”](#) (page 667)).
- 4 Click *Accept* to apply your settings and leave the user configuration.

Figure 34.6 *YaST: Additional LDAP Settings*



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*.

34.5 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.

Samba

35

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. Configure Samba with YaST, SWAT (a Web interface), or the configuration file.

35.1 Terminology

The following are some terms used in Samba documentation and in the YaST module.

SMB protocol

Samba uses the SMB (server message block) protocol 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.

CIFS protocol

CIFS (common Internet file system) protocol is another protocol supported by Samba. CIFS defines a standard remote file system access protocol for use over the network, enabling groups of users to work together and share documents across the network.

NetBIOS

NetBIOS is a software interface (API) designed for communication between machines. Here, a name service is provided. It enables machines connected to the

network 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 as long as 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.

Samba server

Samba server is a server that provides SMB/CIFS services and NetBIOS over IP naming services to clients. For Linux, there are two daemons for Samba server: `smnd` for SMB/CIFS services and `nmbd` for naming services.

Samba client

Samba client is a system that uses Samba services from a Samba server over the SMB protocol. 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. You do not need run any daemon for Samba client.

Shares

SMB servers provide hardware space to their clients by means of shares. Shares are printers and directories with their 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.

35.2 Starting and Stopping Samba

You can start or stop the Samba server automatically during boot or manually. Starting and stopping policy is a part of the YaST Samba server configuration described in [Section 35.3.1, “Configuring a Samba Server with YaST”](#) (page 671).

To stop or start running Samba services with YaST, use *System* → *System Services (Runlevel)*. From a command line, stop services required for Samba with `rcsmb stop` && `rcnmb stop` and start them with `rcnmb start` && `rcsmb start`.

35.3 Configuring a Samba Server

A samba server in SUSE® Linux Enterprise 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.

35.3.1 Configuring a Samba Server with YaST

To configure a Samba server, start YaST and select *Network Services* → *Samba Server*. When starting the module for the first time, the *Samba Server Installation* dialog starts, prompting you to make just a few basic decisions concerning administration of the server then at the end of the configuration prompts for the password of Samba root. For later starts, the *Samba Server Configuration* dialog appears.

The *Samba Server Installation* dialog consists of two steps:

Workgroup or Domain Name

Select an existing name from *Workgroup or Domain Name* or enter a new one and click *Next*.

Samba Server Type

In the next step, specify whether your server should act as PDC and click *Next*.

You can change all settings from *Samba Server Installation* later in the *Samba Server Configuration* dialog with the *Identity* tab.

Advanced Samba Configuration with YaST

During first start of Samba server module the *Samba Server Configuration* dialog appears directly after *Samba Server Installation* dialog. Use it to adjust your Samba server configuration.

After editing your configuration, click *Finish* to close the configuration.

Starting the Server

In the *Start Up* tab, configure the start of the Samba server. To start the service every time your system boots, select *During Boot*. To activate manual start, choose *Manually*. More information about starting a Samba server is provided in [Section 35.2, “Starting and Stopping Samba”](#) (page 671).

In this tab, you can also open ports in your firewall. To do so, select *Open Port in Firewall*. If you have multiple network interfaces, select the network interface for Samba services by clicking *Firewall Details*, selecting the interfaces, and clicking *OK*.

Shares

In the *Shares* tab, determine the Samba shares to activate. There are some predefined shares, like homes and printers. Use *Toggle Status* to switch between *Active* and *Inactive*. Click *Add* to add new shares and *Delete* to delete the selected share.

Identity

In the *Identity* tab, you can determine the domain with which the host is associated (*Base Settings*) and whether to use an alternative hostname in the network (*NetBIOS Host Name*). To set expert global settings or set user authentication, click *Advanced Settings*.

35.3.2 Web Administration with SWAT

An alternative tool for Samba server administration is SWAT (Samba Web Administration Tool). It provides a simple Web interface with which to configure the Samba server. To use SWAT, open <http://localhost:901> in a Web browser and log in as user `root`. If you do not have a special Samba root account, use the system `root` account.

NOTE: Activating SWAT

After Samba server installation, SWAT is not activated. To activate it, open *Network Services* → *Network Services (xinetd)* in YaST, enable the network services configuration, select *swat* from the table, and click *Toggle Status (On or Off)*.

35.3.3 Configuring the Server Manually

If you intend to use Samba as a server, install `samba`. 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.

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 workgroup. 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.

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 35.1 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 network. 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 35.2 *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`.

Security Levels

To improve 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 selection of share, user, or 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`.

35.4 Configuring Clients

Clients can only access the Samba server via TCP/IP. NetBEUI and NetBIOS via IPX cannot be used with Samba.

35.4.1 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 *Windows Domain Membership*. 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.

35.4.2 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.

35.5 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. In a Windows-based network, this task is handled by a primary domain controller (PDC). You can use a Windows NT server configured as PDC, but this task can also be done with the help of a Samba server. The entries that must be made in the `[global]` section of `smb.conf` are shown in [Example 35.3, “Global Section in smb.conf”](#) (page 678).

Example 35.3 *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 35.4 *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 35.5 *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`.

35.6 For More Information

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 documentation is installed for more online documentation and examples. Find a commented example configuration (`smb.conf.SuSE`) in the `examples` subdirectory.

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. You can find Samba HOWTO Collection in `/usr/share/doc/packages/samba/Samba-HOWTO-Collection.pdf` after installing the package `samba-doc`.

Find detailed information about LDAP and migration from Windows NT or 2000 in `/usr/share/doc/packages/samba/examples/LDAP/smbldap-tools-*/doc`, where `*` is your `smbldap-tools` version.

36

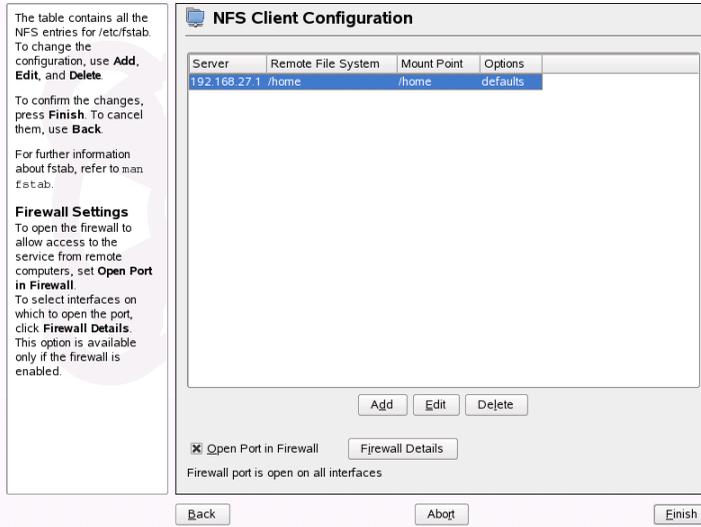
Sharing File Systems with NFS

As mentioned in [Chapter 32, Using NIS](#) (page 641), 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.

36.1 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 36.1, “NFS Client Configuration with YaST”](#) (page 682).

Figure 36.1 NFS Client Configuration with YaST



36.2 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 `reportmap 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-pathlocal-path
```

If user directories from the machine `sun`, for example, should be imported, use the following command:

```
mount sun:/home /home
```

36.3 For More Information

Information about configuring an NFS server is available in `/usr/share/doc/packages/nfs-utils/README` and the documents listed there. The detailed

technical documentation is available online at <http://nfs.sourceforge.net/>

.

File Synchronization

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.

37.1 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.

37.1.1 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.

37.1.2 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.

37.1.3 Subversion

In contrast to CVS, which “evolved,” Subversion (SVN) 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.

37.1.4 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.

37.1.5 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.

37.2 Determining Factors for Selecting a Program

There are some important factors to consider when deciding which program to use.

37.2.1 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.

37.2.2 Portability

Subversion, CVS, and unison are also available for many other operating systems, including various Unix and Windows systems.

37.2.3 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.

37.2.4 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.

37.2.5 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 `cv 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 `cv update` are ignored due to the large number of files.

37.2.6 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.

37.2.7 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.

37.2.8 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.

37.2.9 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.

37.2.10 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.

37.2.11 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 37.1 *Features of the File Synchronization Tools: -- = very poor, - = poor or not available, o = medium, + = good, ++ = excellent, x = available*

	unison	CVS/SVN	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/SVN	rsync	mailsync
History	-	x/x	-	-
Hard Disk Space	o	--	o	+
GUI	+	o/o	-	-
Difficulty	+	o/o	+	o
Attacks	+(ssh)	+/(ssh)	+(ssh)	+(SSL)
Data Loss	+	++/++	+	+

37.3 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.

37.3.1 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 37.1.2, “CVS”](#) (page 686)).

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.

37.3.2 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 37.1 *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`.

37.3.3 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`.

37.4 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.

37.4.1 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 CVSROOT=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
```

37.4.2 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.

37.4.3 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/>

37.5 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 37.1.3, “Subversion”](#) (page 687).

37.5.1 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 different types of repository. The Berkeley Database or FSFS (a repository that uses the file system directly) is commonly used. Do not 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 37.5.3, “For More Information”](#) (page 700).

37.5.2 Usage and Operation

Use the command `svn` (similar to `cvcs`) to access a Subversion repository. With `svn help`, obtain the description of a parameter of a command:

```
checkout (co): Check out a working copy from a repository.
usage: checkout URL[@REV]... [PATH]
```

```
    If specified, REV determines in which revision the URL is first
    looked up.
```

```
    If PATH is omitted, the basename of the URL will be used as
    the destination. If multiple URLs are given each will be checked
    out into a sub-directory of PATH, with the name of the sub-directory
    being the basename of the URL.
```

```
...
```

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.

37.5.3 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/svn-book.html> after installation of the package `subversion-doc` and is also available online at <http://svnbook.red-bean.com/svnbook/index.html>.

37.6 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.

37.6.1 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.

37.6.2 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/>.

37.7 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

37.7.1 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.

37.7.2 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`.

37.7.3 For More Information

The README in `/usr/share/doc/packages/maillsync/`, which is included in `maillsync`, provides additional information. In this connection, RFC 2076 “Common Internet Message Headers” is of special interest.

Part V. Security

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.

38.1 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.

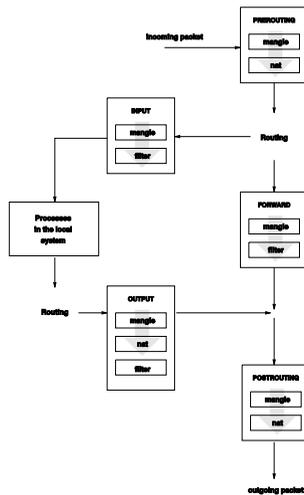
nat

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 38.1 *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 38.1, “[iptables: A Packet's Possible Paths](#)” (page 710) 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.

38.2 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 29.1.2, “Netmasks and Routing”](#) (page 581)) 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 Enterprise 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.

38.3 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 Enterprise. 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`.

38.4 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 29.1.2, “Netmasks and Routing”](#) (page 581)), 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 38.4.1, “Configuring the Firewall with YaST”](#) (page 714)). 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`.

38.4.1 Configuring the Firewall 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*.

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.

38.4.2 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 `ipp0`. 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 (020405B40402080A061AFEB0000000001030300)
```

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.

38.5 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.

SSH: Secure Network Operations

39

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 Enterprise is OpenSSH.

39.1 The OpenSSH Package

SUSE Linux Enterprise 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 Enterprise system is only possible with the OpenSSH utilities and only if the firewall permits access.

39.2 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"
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`.

39.3 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` includ-

ing 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.

39.4 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.

39.5 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. Version 2 of the SSH protocol is used by default.

Override this to use version 1 of the protocol with the `-1` switch. To continue using version 1 after a system 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.

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`.

39.6 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`.

39.7 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`.

Network Authentication—Kerberos

40

An open network provides no means to ensure that a workstation can identify its users properly except the usual password mechanisms. In common installations, the user must enter the password each time a service inside the network is accessed. Kerberos provides an authentication method with which a user registers once then is trusted in the complete network for the rest of the session. To have a secure network, the following requirements must be met:

- Have all users prove their identity for each desired service and make sure that no one can take the identity of someone else.
- Make sure that each network server also proves its identity. Otherwise an attacker might be able to impersonate the server and obtain sensitive information transmitted to the server. This concept is called *mutual authentication*, because the client authenticates to the server and vice versa.

Kerberos helps you meet these requirements by providing strongly encrypted authentication. The following shows how this is achieved. Only the basic principles of Kerberos are discussed here. For detailed technical instruction, refer to the documentation provided with your implementation of Kerberos.

40.1 Kerberos Terminology

The following glossary defines some Kerberos terminology.

credential

Users or clients need to present some kind of credentials that authorize them to request services. Kerberos knows two kinds of credentials—tickets and authenticators.

ticket

A ticket is a per-server credential used by a client to authenticate at a server from which it is requesting a service. It contains the name of the server, the client's name, the client's Internet address, a time stamp, a lifetime, and a random session key.

All this data is encrypted using the server's key.

authenticator

Combined with the ticket, an authenticator is used to prove that the client presenting a ticket is really the one it claims to be. An authenticator is built of the client's name, the workstation's IP address, and the current workstation's time all encrypted with the session key only known to the client and the server from which it is requesting a service. An authenticator can only be used once, unlike a ticket. A client can build an authenticator itself.

principal

A Kerberos principal is a unique entity (a user or service) to which it can assign a ticket. A principal consists of the following components:

- **Primary**—the first part of the principal, which can be the same as your username in the case of a user.
- **Instance**—some optional information characterizing the primary. This string is separated from the primary by a /.
- **Realm**—this specifies your Kerberos realm. Normally, your realm is your domain name in uppercase letters.

mutual authentication

Kerberos ensures that both client and server can be sure of each others identity. They share a session key, which they can use to communicate securely.

session key

Session keys are temporary private keys generated by Kerberos. They are known to the client and used to encrypt the communication between the client and the server for which it requested and received a ticket.

replay

Almost all messages sent in a network can be eavesdropped, stolen, and resent. In the Kerberos context, this would be most dangerous if an attacker manages to obtain your request for a service containing your ticket and authenticator. He could then try to resend it (*replay*) to impersonate you. However, Kerberos implements several mechanisms to deal with that problem.

server or service

Service is used to refer to a specific action to perform. The process behind this action is referred to as a *server*.

40.2 How Kerberos Works

Kerberos is often called a third party trusted authentication service, which means all its clients trust Kerberos's judgment of another client's identity. Kerberos keeps a database of all its users and their private keys.

To ensure Kerberos is worth all the trust put in it, run both the authentication and ticket-granting server on a dedicated machine. Make sure that only the administrator can access this machine physically and over the network. Reduce the (networking) services run on it to the absolute minimum—do not even run `sshd`.

40.2.1 First Contact

Your first contact with Kerberos is quite similar to any login procedure at a normal networking system. Enter your username. This piece of information and the name of the ticket-granting service are sent to the authentication server (Kerberos). If the authentication server knows about your existence, it generates a random session key for further use between your client and the ticket-granting server. Now the authentication server prepares a ticket for the ticket-granting server. The ticket contains the following information—all encrypted with a session key only the authentication server and the ticket-granting server know:

- The names both of the client and the ticket-granting server
- The current time
- A lifetime assigned to this ticket

- The client's IP address
- The newly-generated session key

This ticket is then sent back to the client together with the session key, again in encrypted form, but this time the private key of the client is used. This private key is only known to Kerberos and the client, because it is derived from your user password. Now that the client has received this response, you are prompted for your password. This password is converted into the key that can decrypt the package sent by the authentication server. The package is “unwrapped” and password and key are erased from the workstation's memory. As long as the lifetime given to the ticket used to obtain other tickets does not expire, your workstation can prove your identity.

40.2.2 Requesting a Service

To request a service from any server in the network, the client application needs to prove its identity to the server. Therefore, the application generates an authenticator. An authenticator consists of the following components:

- The client's principal
- The client's IP address
- The current time
- A checksum (chosen by the client)

All this information is encrypted using the session key that the client has already received for this special server. The authenticator and the ticket for the server are sent to the server. The server uses its copy of the session key to decrypt the authenticator, which gives it all information needed about the client requesting its service to compare it to that contained in the ticket. The server checks if the ticket and the authenticator originate from the same client.

Without any security measures implemented on the server side, this stage of the process would be an ideal target for replay attacks. Someone could try to resend a request stolen off the net some time before. To prevent this, the server does not accept any request with a time stamp and ticket received previously. In addition to that, a request with a time stamp differing too much from the time the request is received is ignored.

40.2.3 Mutual Authentication

Kerberos authentication can be used in both directions. It is not only a question of the client being the one it claims to be. The server should also be able to authenticate itself to the client requesting its service. Therefore, it sends some kind of authenticator itself. It adds one to the checksum it received in the client's authenticator and encrypts it with the session key, which is shared between it and the client. The client takes this response as a proof of the server's authenticity and they both start cooperating.

40.2.4 Ticket Granting—Contacting All Servers

Tickets are designed to be used for one server at a time. This implies that you have to get a new ticket each time you request another service. Kerberos implements a mechanism to obtain tickets for individual servers. This service is called the “ticket-granting service”. The ticket-granting service is a service just like any other service mentioned before, so uses the same access protocols that have already been outlined. Any time an application needs a ticket that has not already been requested, it contacts the ticket-granting server. This request consists of the following components:

- The requested principal
- The ticket-granting ticket
- An authenticator

Like any other server, the ticket-granting server now checks the ticket-granting ticket and the authenticator. If they are considered valid, the ticket-granting server builds a new session key to be used between the original client and the new server. Then the ticket for the new server is built, containing the following information:

- The client's principal
- The server's principal
- The current time
- The client's IP address

- The newly-generated session key

The new ticket is assigned a lifetime, which is the lesser of the remaining lifetime of the ticket-granting ticket and the default for the service. The client receives this ticket and the session key, which are sent by the ticket-granting service, but this time the answer is encrypted with the session key that came with the original ticket-granting ticket. The client can decrypt the response without requiring the user's password when a new service is contacted. Kerberos can thus acquire ticket after ticket for the client without bothering the user more than once at login time.

40.2.5 Compatibility to Windows 2000

Windows 2000 contains a Microsoft implementation of Kerberos 5. Because SUSE® Linux Enterprise Server uses the MIT implementation of Kerberos 5, find useful information and guidance in the MIT documentation. See [Section 40.4, “For More Information”](#) (page 731).

40.3 Users' View of Kerberos

Ideally, a user's one and only contact with Kerberos happens during login at the workstation. The login process includes obtaining a ticket-granting ticket. At logout, a user's Kerberos tickets are automatically destroyed, which makes it difficult for anyone else to impersonate this user. The automatic expiration of tickets can lead to a somewhat awkward situation when a user's login session lasts longer than the maximum lifespan given to the ticket-granting ticket (a reasonable setting is 10 hours). However, the user can get a new ticket-granting ticket by running `kinit`. Enter the password again and Kerberos obtains access to desired services without additional authentication. To get a list of all the tickets silently acquired for you by Kerberos, run `klist`.

Here is a short list of some applications that use Kerberos authentication. These applications can be found under `/usr/lib/mit/bin` or `/usr/lib/mit/sbin`. They all have the full functionality of their common UNIX and Linux brothers plus the additional bonus of transparent authentication managed by Kerberos:

- `telnet`, `telnetd`
- `rlogin`

- rsh, rcp, rshd
- ftp, ftpd
- ksu

You no longer have to enter your password for using these applications because Kerberos has already proven your identity. ssh, if compiled with Kerberos support, can even forward all the tickets acquired for one workstation to another one. If you use ssh to log in to another workstation, ssh makes sure that the encrypted contents of the tickets are adjusted to the new situation. Simply copying tickets between workstations is not sufficient because the ticket contains workstation-specific information (the IP address). XDM, GDM, and KDM offer Kerberos support, too. Read more about the Kerberos network applications in *Kerberos V5 UNIX User's Guide* at <http://web.mit.edu/kerberos>

40.4 For More Information

The official site of the MIT Kerberos is <http://web.mit.edu/kerberos>. There, find links to any other relevant resource concerning Kerberos, including Kerberos installation, user, and administration guides.

The paper at <ftp://athena-dist.mit.edu/pub/kerberos/doc/usenix.PS> gives quite an extensive insight to the basic principles of Kerberos without being too difficult to read. It also provides a lot of opportunities for further investigation and reading about Kerberos.

The official Kerberos FAQ is available at <http://www.nrl.navy.mil/CCS/people/kenh/kerberos-faq.html>. The book *Kerberos—A Network Authentication System* by Brian Tung (ISBN 0-201-37924-4) offers extensive information.

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. For laptops or removable media, such as external hard disks or USB sticks, that are prone to being lost or stolen, it is also very useful to encrypt partitions (or parts of your file system) that hold confidential data.

There are several ways to protect your data by means of encryption:

Encrypting a Hard Disk Partition

You can create an encrypted partition with YaST during installation or in an already installed system. See [Section 41.1.1, “Creating an Encrypted Partition during Installation”](#) (page 734) and [Section 41.1.2, “Creating an Encrypted Partition on a Running System”](#) (page 735) for the details. This option can also be used for removable media, such as external hard disks, as described in [Section 41.1.4, “Encrypting the Content of Removable Media”](#) (page 736).

Creating an Encrypted File as Container

You can at any time create an encrypted file on your hard disk or on a removable medium with YaST. The encrypted file can then be used to *store* other files or folders. For more information, refer to [Section 41.1.3, “Creating an Encrypted File as a Container”](#) (page 736).

Encrypting Single Files

If you only have a small number of files that hold sensitive or confidential data, you can encrypt them individually and protect them with a password using the `vi`

editor. Refer to [Section 41.2, “Using vi to Encrypt Single Files”](#) (page 736) for more information.

WARNING: Encrypted Media Is Limited Protection

Be aware that with the methods described in this chapter, you cannot protect your running system from being compromised. After the encrypted media is successfully mounted, everybody with appropriate permissions has access to it. However, encrypted media is useful for cases such as loss or theft of your computer or to prevent unauthorized individuals from reading your confidential data.

41.1 Setting Up a Crypto File System with YaST

Use YaST to encrypt partitions or parts of your file system during installation or in an already installed system. However, encrypting a partition in an already installed system is more difficult because you have to resize and change existing partitions. In such cases, it may be more convenient to create an encrypted file of a defined size in which to *store* other files or parts of your file system. 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.

41.1.1 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 7.5.6, “Partitioner”](#) (page 150), offers the options needed for creating an encrypted partition. To create a

new encrypted partition, click *Create*. 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 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.

When you are installing your system on a machine where several partitions already exist, you can also decide to encrypt an existing partition during installation. In this case follow the description in [Section 41.1.2, “Creating an Encrypted Partition on a Running System”](#) (page 735) and be aware that this action destroys all data on the existing partition to encrypt.

41.1.2 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. However, encrypting an existing partition destroys all data on it and requires resize and restructuring of existing partitions.

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 as in [Section 41.1.1, “Creating an Encrypted Partition during Installation”](#) (page 734).

41.1.3 Creating an Encrypted File as a Container

Instead of using a partition, it is possible to create an encrypted file of a certain size that can then hold other files or folders containing confidential data. Such container files 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 container 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.

41.1.4 Encrypting the Content of Removable Media

YaST treats removable media like external hard disks or USB flash drives the same as any other hard disk. Container 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.

41.2 Using vi to Encrypt Single 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.

Confining Privileges with AppArmor

42

Many security vulnerabilities result from bugs in *trusted* programs. A trusted program runs with privilege that some attacker would like to have. The program fails to keep that trust if there is a bug in the program that allows the attacker to acquire that privilege.

Novell® AppArmor is an application security solution designed specifically to provide least privilege confinement to suspect programs. AppArmor allows the administrator to specify the domain of activities the program can perform by developing a security *profile* for that application—a listing of files that the program may access and the operations the program may perform.

Effective hardening of a computer system requires minimizing the number of programs that mediate privilege then securing the programs as much as possible. With Novell AppArmor, you only need to profile the programs that are exposed to attack in your environment, which drastically reduces the amount of work required to harden your computer. AppArmor profiles enforce policies to make sure that programs do what they are supposed to do, but nothing else.

Administrators only need to care about the applications that are vulnerable to attacks and generate profiles for these. Hardening a system thus comes down to building and maintaining the AppArmor profile set and monitoring any policy violations or exceptions logged by AppArmor's reporting facility.

Building AppArmor profiles to confine an application is very straightforward and intuitive. AppArmor ships with several tools that assist in profile creation. It does not require you to do any programming or script handling. The only task that is required from the administrator is to determine a policy of strictest access and execute permissions for each application that needs to be hardened.

Updates or modifications to the application profiles are only required if the software configuration or the desired range of activities changes. AppArmor offers intuitive tools to handle profile updates or modifications.

Users should not notice AppArmor at all. It runs “behind the scenes” and does not require any user interaction. Performance is not affected noticeably by AppArmor. If some activity of the application is not covered by an AppArmor profile or if some activity of the application is prevented by AppArmor, the administrator needs to adjust the profile of this application to cover this kind of behavior.

This guide outlines the basic tasks that need to be performed with AppArmor to effectively harden a system. For more in-depth information, refer to *Novell AppArmor 2.0 Administration Guide*.

42.1 Installing Novell AppArmor

Novell AppArmor is installed and running by default on any installation of SUSE® Linux Enterprise regardless of what patterns are installed. The packages listed below are needed for a fully functional instance of AppArmor

- `apparmor-parser`
- `libapparmor`
- `apparmor-docs`
- `yast2-apparmor`
- `apparmor-profiles`
- `apparmor-utils`

42.2 Enabling and Disabling Novell AppArmor

Novell AppArmor is configured to run by default on any fresh installation of SUSE Linux Enterprise. There are two ways of toggling the status of AppArmor:

Using YaST System Services (Runlevel)

Disable or enable AppArmor by removing or adding its boot script to the sequence of scripts executed on system boot. Status changes are applied at the next system boot.

Using Novell AppArmor Control Panel

Toggle the status of Novell AppArmor in a running system by switching it off or on using the YaST Novell AppArmor Control Panel. Changes made here are applied instantaneously. The Control Panel triggers a stop or start event for AppArmor and removes or adds its boot script in the system's boot sequence.

To disable AppArmor permanently by removing it from the sequence of scripts executed on system boot, proceed as follows:

- 1 Log in as `root` and start YaST.
- 2 Select *System* → *System Services (Runlevel)*.
- 3 Select *Expert Mode*.
- 4 Select `boot . apparmor` and click *Set/Reset* → *Disable the service*.
- 5 Exit the YaST Runlevel tool with *Finish*.

AppArmor will not be initialized on the next system boot and stays inactive until you explicitly reenables it. Reenabling a service using the YaST Runlevel tool is similar to disabling it.

Toggle the status of AppArmor in a running system by using the AppArmor Control Panel. These changes take effect as soon as you apply them and survive a reboot of the system. To toggle AppArmor's status, proceed as follows:

- 1 Log in as `root` and start YaST.
- 2 Select *Novell AppArmor* → *AppArmor Control Panel*.
- 3 Select *Enable AppArmor* → *Configure*.
- 4 Click *Enable* and *OK* to enable AppArmor or *Disable* and *OK* to disable AppArmor.

- 5 Exit the AppArmor Control Panel with *Done*.

42.3 Getting Started with Profiling Applications

Prepare a successful deployment of Novell AppArmor on your system by carefully considering the following items:

- 1 Determine the applications to profile. Read more on this in [Section 42.3.1, “Choosing the Applications to Profile”](#) (page 742).
- 2 Build the needed profiles as roughly outlined in [Section 42.3.2, “Building and Modifying Profiles”](#) (page 743). Check the results and adjust the profiles when necessary.
- 3 Keep track of what is happening on your system by running AppArmor reports and dealing with security events. Refer to [Section 42.3.3, “Configuring Novell AppArmor Event Notification and Reports”](#) (page 746).
- 4 Update your profiles whenever your environment changes or you need to react to security events logged by AppArmor's reporting tool. Refer to [Section 42.3.4, “Updating Your Profiles”](#) (page 747).

42.3.1 Choosing the Applications to Profile

You only need to protect the programs that are exposed to attacks in your particular setup, so only use profiles for those applications you really run. Use the following list to determine the most likely candidates:

Network Agents

Programs (servers and clients) have open network ports and network agents are server programs that respond to those network ports. User clients, such as mail clients and Web browsers, also have open network ports and mediate privilege. Any attack on a user's Web browser or e-mail client allows the attacker to steal private data from the user. In addition to that, the attacker might use the access gained through that attack as leverage to get more privileges on the system.

Web Applications

CGI Perl scripts, PHP pages, and more complex Web applications can be started through a Web browser.

Cron Jobs

Programs that the cron daemon periodically run read input from a variety of sources.

To find out which processes are currently running with open network ports and might need a profile to confine them, run `aa-unconfined` as `root`.

Example 42.1 *Output of `aa-unconfined`*

```
19848 /usr/sbin/cupsd not confined
19887 /usr/sbin/sshd not confined
19947 /usr/lib/postfix/master not confined
29205 /usr/sbin/sshd confined by '/usr/sbin/sshd (enforce)'
```

Each of the processes in the above example labeled `not confined` might need a custom profile to confine it. Those labeled `confined by` are already protected by AppArmor.

TIP: For More Information

For more information about choosing the the right applications to profile, refer to Chapter 2, *Selecting Programs to Immunize* (↑Novell AppArmor 2.0 Administration Guide).

42.3.2 Building and Modifying Profiles

Novell AppArmor on SUSE Linux Enterprise ships with a preconfigured set of profiles for the most important applications. In addition to that, you can use AppArmor to create your own profiles for any application you want.

There are two ways of managing profiles. One is to use the graphical front-end provided by the YaST Novell AppArmor modules and the other is to use the command line tools provided by the AppArmor suite itself. Both methods basically work the same way.

Running `aa-unconfined` as described in [Section 42.3.1, “Choosing the Applications to Profile”](#) (page 742) identifies a list of applications that may need a profile to run in a safe mode.

For each application, perform the following steps to create a profile:

- 1 As `root`, let AppArmor create a rough outline of the application's profile by running `aa-genprof programname`

or

Outline the basic profile by running *YaST* → *Novell AppArmor* → *Add Profile Wizard* and specifying the complete path of the application to profile.

A basic profile is outlined and AppArmor is put into learning mode, which means that it logs any activity of the program you are executing but does not yet restrict it.

- 2 Run the full range of the application's actions to let AppArmor get a very specific picture of its activities.
- 3 Let AppArmor analyze the log files generated in [Step 2](#) (page 744) by running typing `[5]` in `aa-genprof`.

or

Analyze the logs by clicking *Scan system log for AppArmor events* in the *Add Profile Wizard* and following the instructions given in the wizard until the profile is completed.

AppArmor scans the logs it recorded during the application's run and asks you to set the access rights for each event that was logged. Either set them for each file or use globbing.

- 4 Once all access permissions are set, your profile is set to enforce mode. The profile is applied and AppArmor restricts the application according to the profile just created.

If you started `aa-genprof` on an application that had an existing profile that was in complain mode, this profile remains in learning mode upon exit of this learning cycle. For more information about changing the mode of a profile, refer to Section “aa-complain—Entering Complain or Learning Mode” (Chapter 3, *Building Novell AppArmor Profiles*, ↑Novell AppArmor 2.0 Administration Guide) and Section “aa-enforce—Entering Enforce Mode” (Chapter 3, *Building Novell AppArmor Profiles*, ↑Novell AppArmor 2.0 Administration Guide).

Test your profile settings by performing every task you need with the application you just confined. Normally, the confined program runs smoothly and you do not notice AppArmor activities at all. However, if you notice certain misbehavior with your application, check the system logs and see if AppArmor is too closely constricting your application. Depending on the log mechanism used on your system, there are several places to look for AppArmor log entries:

`/var/log/audit.log`

If the `audit` package is installed and `auditd` is running, AppArmor events are logged as follows:

```
type=APPARMOR msg=audit(1140325305.502:1407): REJECTING w access to
/usr/lib/firefox/update.test (firefox-bin(9469) profile
/usr/lib/firefox/firefox-bin active /usr/lib/firefox/firefox-bin)
```

`/var/log/messages`

If `auditd` is not used, AppArmor events are logged in the standard system log under `/var/log/messages`. An example entry would look like the following:

```
Feb 22 18:29:14 dhcp-81 klogd: audit(1140661749.146:3): REJECTING w access
to /dev/console (mdnsd(3239) profile /usr/sbin/mdnsd active
/usr/sbin/mdnsd)
```

`dmesg`

If `auditd` is not running, AppArmor events can also be checked using the `dmesg` command:

```
audit(1140661749.146:3): REJECTING w access to /dev/console (mdnsd(3239)
profile /usr/sbin/mdnsd active /usr/sbin/mdnsd)
```

To adjust the profile, analyze the log messages relating this application again as described in [Step 3](#) (page 744). Determine the access rights or restrictions when prompted.

TIP: For More Information

For more information about profile building and modification, refer to Chapter 3, *Building Novell AppArmor Profiles* (↑Novell AppArmor 2.0 Administration Guide).

42.3.3 Configuring Novell AppArmor Event Notification and Reports

Set up event notification in Novell AppArmor so you can review security events. Event Notification is an Novell AppArmor feature that informs a specified e-mail recipient when systemic Novell AppArmor activity occurs under the chosen severity level. This feature is currently available in the YaST interface.

To set up event notification in YaST, proceed as follows:

- 1 Make sure that a mail server is running on your system to deliver the event notifications.
- 2 Log in as `root` and start YaST. Then select *Novell AppArmor* → *AppArmor Control Panel*).
- 3 In *Enable Security Event Notification*, select *Configure*.
- 4 For each record type (*Terse*, *Summary*, and *Verbose*), set a report frequency, enter the e-mail address that should receive the reports, and determine the severity of events to log. To include unknown events in the event reports, check *Include Unknown Severity Events*.

NOTE: Selecting Events to Log

Unless you are familiar with AppArmor's event categorization, choose to be notified about events for all security levels.

- 5 Leave this dialog with *OK* → *Finish* to apply your settings.

Using Novell AppArmor reports, you can read important Novell AppArmor security events reported in the log files without manually sifting through the cumbersome messages only useful to the `aa-logprof` tool. You can decrease the size of the report by filtering by date range or program name.

To configure the AppArmor reports, proceed as follows:

- 1 Log in as `root` and start YaST. Select *Novell AppArmor* → *AppArmor Reports*.

- 2 Select the type of report to examine or configure from *Executive Security Summary*, *Applications Audit*, and *Security Incident Report*.
- 3 Edit the report generation frequency, e-mail address, export format, and location of the reports by selecting *Edit* and providing the requested data.
- 4 To run a report of the selected type, click *Run Now*.
- 5 Browse through the archived reports of a given type by selecting *View Archive* and specifying the report type.

or

Delete unneeded reports or add new ones.

TIP: For More Information

For more information about configuring event notification in Novell AppArmor, refer to Section 4.2, “Setting Up Event Notification” (Chapter 4, *Managing Profiled Applications*, ↑Novell AppArmor 2.0 Administration Guide). Find more information about report configuration in Section 4.3, “Reports” (Chapter 4, *Managing Profiled Applications*, ↑Novell AppArmor 2.0 Administration Guide).

42.3.4 Updating Your Profiles

Software and system configurations change over time. As a result of that, your profile setup for AppArmor might need some fine-tuning from time to time. AppArmor checks your system log for policy violations or other AppArmor events and lets you adjust your profile set accordingly. Any application behavior that is outside of any profile definition can also be addressed using the *Update Profile Wizard*.

To update your profile set, proceed as follows:

- 1 Log in as `root` and start YaST.
- 2 Start *Novell AppArmor* → *Update Profile Wizard*.
- 3 Adjust access or execute rights to any resource or for any executable that has been logged when prompted.

- 4 Leave YaST after you answer all questions. Your changes are applied to the respective profiles.

TIP: For More Information

For more information about updating your profiles from the system logs, refer to Section 3.3.5, “Updating Profiles from Log Entries” (Chapter 3, *Building Novell AppArmor Profiles*, ↑Novell AppArmor 2.0 Administration Guide).

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.

43.1 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.

43.1.1 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.

43.1.2 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.

43.1.3 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 17, The Boot Loader](#) (page 387)). 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.

43.1.4 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 Enterprise 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`).

43.1.5 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 43.1.4, “File Permissions”](#) (page 752)).

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 43.1.4, “File Permissions”](#) (page 752)).

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.

43.1.6 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.

43.1.7 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.

43.1.8 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 [Chapter 39, SSH: Secure Network Operations](#) (page 719).

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.

43.1.9 Buffer Overflows and Format String Bugs

As discussed in [Section 43.1.5, “Buffer Overflows and Format String Bugs”](#) (page 753), 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 Enterprise 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.

43.1.10 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.

43.1.11 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.

43.1.12 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.

43.1.13 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.

43.2 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 Enterprise. 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.

43.3 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>.

Part VI. Troubleshooting

Help and Documentation

SUSE® Linux Enterprise comes with various sources of information and documentation. The SUSE Help Center provides central access to the most important documentation resources on your system in searchable form. These resources include online help for installed applications, manual pages, info pages, databases on hardware and software topics, and all manuals delivered with your product.

44.1 Using the SUSE Help Center

When you start the SUSE Help Center for the first time from the main menu (*SuSE Help Center*) or with the command `susehelp` in the shell, a window as shown in [Figure 44.1, “The Main Window of the SUSE Help Center”](#) (page 766) is displayed. The dialog window consists of three main areas:

Menu Bar and Toolbar

The menu bar provides the main editing, navigation, and configuration options. *File* contains the option for printing the currently displayed content. Under *Edit*, access the search function. *Go* contains all navigation possibilities: *Table of Contents* (home page of the Help Center), *Back*, *Forward*, and *Last Search Result*. With *Settings* → *Build Search Index*, generate a search index for all selected information sources. The toolbar contains three navigation icons (forward, back, home) and a printer icon for printing the current contents.

Navigation Area with Tabs

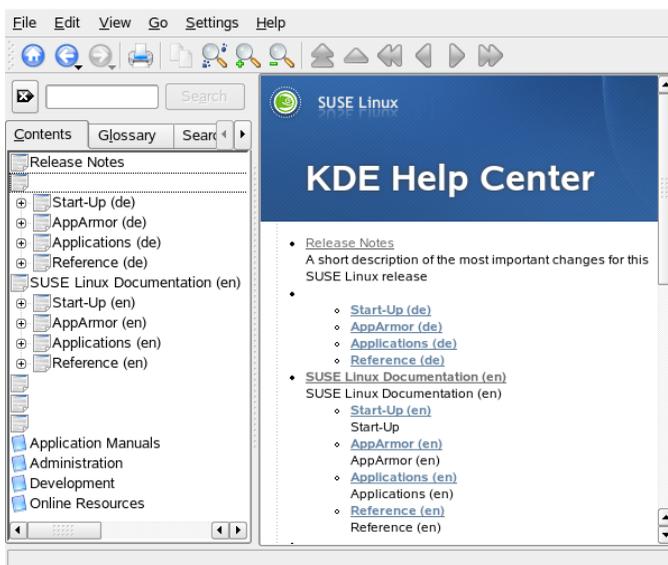
The navigation area in the left part of the window provides an input field for a quick search in selected information sources. Details regarding the search and the

configuration of the search function in the *Search* tab are presented in [Section 44.1.2, “The Search Function”](#) (page 767). The *Contents* tab presents a tree view of all available and currently installed information sources. Click the book icons to open and browse the individual categories.

View Window

The view window always displays the currently selected contents, such as online manuals, search results, or Web pages.

Figure 44.1 *The Main Window of the SUSE Help Center*



NOTE: Language Selects View

The documentation available in the SUSE Help Center depends on the current language. Changing your language changes the tree view.

44.1.1 Contents

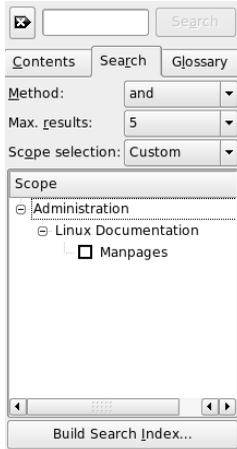
The SUSE Help Center provides access to useful information from various sources. It contains special documentation for SUSE Linux Enterprise (*Start-Up* and *Reference*), all available information sources for your workstation environment, online help for the

installed programs, and help texts for other applications. Furthermore, the SUSE Help Center provides access to SUSE's online databases that cover special hardware and software issues for SUSE Linux Enterprise. All these sources can be searched comfortably once a search index has been generated.

44.1.2 The Search Function

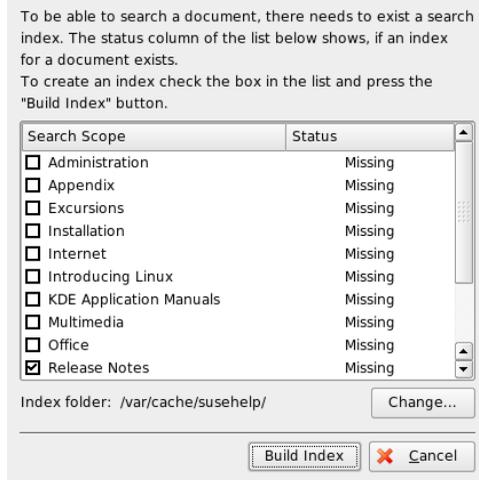
To search all installed information sources of SUSE Linux Enterprise, generate a search index and set a number of search parameters. To do this, use the *Search* tab, shown in [Figure 44.2, “Configuring the Search Function”](#) (page 767).

Figure 44.2 *Configuring the Search Function*



If no search index has been generated, the system automatically prompts you to do so when you click the *Search* tab or enter a search string then click *Search*. In the window for generating the search index, shown in [Figure 44.3, “Generating a Search Index”](#) (page 768), use the check boxes to determine the information sources to index. The index is generated when you exit the dialog with *Build Index*.

Figure 44.3 *Generating a Search Index*



To limit the search base and the hit list as precisely as possible, use the three drop-down menus to determine the number of displayed hits and the selection area of sources to search. The following options are available for determining the selection area:

Default

A predefined selection of sources is searched.

All

All sources are searched.

None

No sources selected for the search.

Custom

Determine the sources to search by activating the respective check boxes in the overview.

When you have completed the search configuration, click *Search*. The relevant items are then displayed in the view window and can easily be navigated with mouse clicks.

44.2 Man Pages

Man pages are an essential part of any Linux system. They explain the usage of a command and all available options and parameters. Man pages are sorted in categories as shown in [Table 44.1, “Man Pages—Categories and Descriptions”](#) (page 769) (taken from the man page for man itself).

Table 44.1 *Man Pages—Categories and Descriptions*

Number	Description
1	Executable programs or shell commands
2	System calls (functions provided by the kernel)
3	Library calls (functions within program libraries)
4	Special files (usually found in /dev)
5	File formats and conventions (/etc/fstab)
6	Games
7	Miscellaneous (including macro packages and conventions), for example, man(7), groff(7)
8	System administration commands (usually only for root)
9	Kernel routines (nonstandard)

Generally, man pages are delivered with the associated command. They can be browsed in the help center or directly in a shell. To display a man page in a shell, use the `man` command. For example, to display the man page for `ls` enter `man ls`. Each man page consists of several parts labeled *NAME*, *SYNOPSIS*, *DESCRIPTION*, *SEE ALSO*, *LICENSING*, and *AUTHOR*. There may be additional sections available depending on the type of command. With `Q`, exit the man page viewer.

Another possibility to display a man page is to use Konqueror. Start Konqueror and type, for example, `man : /ls`. If there are different categories for a command, Konqueror displays them as links.

44.3 Info Pages

Info pages are another important source of information on your system. Usually they are more verbose than man pages. You can browse an info page with an info viewer and display the different sections, called “nodes.” Use the command `info` for this task. For example, to view the info page of `info` itself, type `info info` in the shell.

For more convenience, use the Help Center or Konqueror. Start Konqueror and type `info : /` to view the top level. To display the info page for `grep`, type `info : /grep`.

44.4 The Linux Documentation Project

The Linux Documentation Project (TLDP) is run by a team of volunteers who write Linux and Linux-related documentation (see <http://www.tldp.org>). The set of documents contains tutorials for beginners, but is mainly focused on experienced users and professional system administrators. TLDP publishes HOWTOs, FAQs, and guides (handbooks) under a free license.

44.4.1 HOWTOs

HOWTOs are usually a short, informal, step-by-step guide to accomplishing a specific task. It is written by experts for nonexperts in a procedural manner. For example, how to configure a DHCP server. HOWTOs can be found in the package `howto` and are installed under `/usr/share/doc/howto`

44.4.2 Frequently Asked Questions

FAQs (frequently asked questions) are a series of questions and answers. They originate from Usenet newsgroups where the purpose was to reduce continuous reposting of the same basic questions.

44.5 Wikipedia: The Free Online Encyclopedia

Wikipedia is “a multilingual encyclopedia designed to be read and edited by anyone” (see <http://en.wikipedia.org>). The content of Wikipedia is created by its users and is published under a free license (GFDL). Any visitors can edit articles, which gives the danger of vandalism, but this does not repel visitors. With over four hundred thousand articles, find an answer for nearly every topic.

44.6 Guides and Books

A broad range of guides and books are available for Linux topics.

44.6.1 SUSE Books

SUSE provides detailed and informative books. We provide HTML and PDF versions of our books in different languages. The PDF file is available on the DVD in the directory `docu`. For HTML, install the package `suselinux-manual_LANG` (replace `LANG` with your preferred language.) After the installation, find them in the SUSE Help Center.

44.6.2 Other Manuals

The SUSE help center offers additional manuals and guides for various topics or programs. More can be found at <http://www.tldp.org/guides.html>. They range from *Bash Guide for Beginners* to *Linux Filesystem Hierarchy* to *Linux Administrator's Security Guide*. Generally, guides are more detailed and exhaustive than a HOWTO or FAQ. They are usually written by experts for experts. Some of these books are old but still valid. Install books and guides with YaST.

44.7 Package Documentation

If you install a package in your system, a directory `/usr/share/doc/packages/packagename` is created. You can find files from the package maintainer as well as additional information from SUSE. Sometimes there are also examples, configuration files, additional scripts, or other things available. Usually you can find the following files, but they are not standard and sometimes not all files are available.

AUTHORS

The list of the main developers of this package and usually their tasks.

BUGS

Known bugs or malfunctions of this package. Usually also a link to a Bugzilla Web page where you can search all bugs.

CHANGES, ChangeLog

Summary of changes from version to version. Usually interesting for developers, because it is very detailed.

COPYING, LICENSE

Licensing information.

FAQ

Question and answers collected from mailing lists or newsgroups.

INSTALL

Procedures for installing this package in your system. Normally you do not need it, because you have the package installed already.

README, README.*

General information such as how to use it and what you can do with this package.

TODD

Things that are not implemented yet, but probably will be in the future.

MANIFEST

List of files with a brief summary.

NEWS

Description of what is new in this version.

44.8 Usenet

Created in 1979 before the rise of the Internet, Usenet is one of the oldest computer networks and still in active use. The format and transmission of Usenet articles is very similar to e-mail, but is developed for a many-to-many communication.

Usenet is organized into seven topical categories: `comp.*` for computer-related discussions, `misc.*` for miscellaneous topics, `news.*` for newsgroup-related matters, `rec.*` for recreation and entertainment, `sci.*` for science-related discussions, `soc.*` for social discussions, and `talk.*` for various controversial topics. The top levels are split in subgroups. For instance, `comp.os.linux.hardware` is a newsgroup for Linux-specific hardware issues.

Before you can post an article, have your client connect to a news server and subscribe to a specific newsgroup. News clients include Knode or Evolution. Each news server communicates to other news servers and exchanges articles with them. Not all newsgroups may be available on your news server.

Interesting newsgroups for Linux users are `comp.os.linux.apps`, `comp.os.linux.questions`, and `comp.os.linux.hardware`. If you cannot find a specific newsgroup, go to <http://www.linux.org/docs/usenetlinux.html>. Follow the general Usenet rules available online at <http://www.faqs.org/faqs/usenet/posting-rules/part1/>.

44.9 Standards and Specifications

There are various sources that provide information about standards or specifications.

<http://www.linuxbase.org>

The Free Standards Group is an independent nonprofit organization that promotes the distribution of free software and open source software. The organization endeavors to achieve this by defining distribution-independent standards. The maintenance of several standards, such as the important LSB (Linux Standard Base), is supervised by this organization.

<http://www.w3.org>

The World Wide Web Consortium (W3C) is certainly one of the best-known standards organizations. It was founded in October 1994 by Tim Berners-Lee and

concentrates on standardizing Web technologies. W3C promotes the dissemination of open, license-free, and manufacturer-independent specifications, such as HTML, XHTML, and XML. These Web standards are developed in a four-stage process in *working groups* and are presented to the public as *W3C recommendations* (REC).

<http://www.oasis-open.org>

OASIS (Organization for the Advancement of Structured Information Standards) is an international consortium specializing in the development of standards for Web security, e-business, business transactions, logistics, and interoperability between various markets.

<http://www.ietf.org>

The Internet Engineering Task Force (IETF) is an internationally active cooperative of researchers, network designers, suppliers, and users. It concentrates on the development of Internet architecture and the smooth operation of the Internet by means of protocols.

Every IETF standard is published as an RFC (Request for Comments) and is available free-of-charge. There are six types of RFC: proposed standards, draft standards, Internet standards, experimental protocols, information documents, and historic standards. Only the first three (proposed, draft, and full) are IETF standards in the narrower sense (see <http://www.ietf.org/rfc/rfc1796.txt>).

<http://www.ieee.org>

The Institute of Electrical and Electronics Engineers (IEEE) is an organization that draws up standards in the areas of information technology, telecommunication, medicine and health care, transport, and others. IEEE standards are subject to a fee.

<http://www.iso.org>

The ISO Committee (International Organization for Standards) is the world's largest developer of standards and maintains a network of national standardization institutes in over 140 countries. ISO standards are subject to a fee.

<http://www.din.de>, <http://www.din.com>

The Deutsches Institut für Normung (DIN) is a registered technical and scientific association. It was founded in 1917. According to DIN, the organization is “the institution responsible for standards in Germany and represents German interests in worldwide and European standards organizations.”

The association brings together manufacturers, consumers, trade professionals, service companies, scientists and others who have an interest in the establishment of standards. The standards are subject to a fee and can be ordered using the DIN home page.

Common Problems and Their Solutions

45

This chapter offers a range of common problems that can arise with an intention of covering as many of the various types of potential problems as possible. That way, even if your precise situation is not listed here, there might be one similar enough to offer hints as to the solution.

45.1 Finding Information

Linux logs things in a fair amount of detail. There are several places to look when you have problems with an SLED system, most of which are standard to Linux systems in general and some of which are peculiar to SUSE and SLED systems.

The following is a list of the most commonly checked log files and what they typically contain.

Log File	Description
<code>/var/log/boot.msg</code>	Messages from the kernel during the boot process.
<code>/var/log/mail.*</code>	Messages from the mail system.
<code>/var/log/messages</code>	Ongoing messages from the kernel and system log daemon when running.

Log File	Description
<code>/var/log/SaX.log</code>	Hardware messages from the SaX display and KVM system.
<code>/home/user/.xsession-errors</code>	Messages from the desktop applications currently running. Replace <i>user</i> with the actual username.
<code>/var/log/warn</code>	All messages from the kernel and system log daemon assigned WARNING level or higher.
<code>/var/log/wtmp</code>	Binary file containing user login records for the current machine session. View it with <code>last</code> .
<code>/var/log/Xorg.*.log</code>	Various start-up and runtime logs from the X Window system. It is useful for debugging failed X start-ups.
<code>/var/log/YaST2/</code>	Directory containing YaST's actions and their results.
<code>/var/log/samba/</code>	Directory containing Samba server and client log messages.

Linux comes with a number of tools for system analysis and monitoring. See [Chapter 13, *System Monitoring Utilities*](#) (page 311) for a selection of the most important ones used in system diagnostics.

Each scenario included in the following begins with a header describing the problem followed by a paragraph or two offering suggested solutions, available references for more detailed solutions, and cross-references to other scenarios that might be related.

45.2 Installation Problems

Installation problems are situations when a machine fails to install. It may fail entirely or it may not be able to start the graphical installer. This section highlights some of the

typical problems you might run into and offers possible solutions or workarounds for this kind of situations.

45.2.1 No Bootable CD-ROM Drive Available

If your computer does not contain a bootable CD or DVD-ROM drive or if the one you have is not supported by Linux, there are several options for installing your machine without a need for a built-in CD or DVD drive:

Booting from a Floppy Disk

Create a boot floppy and boot from floppy disk instead of CD or DVD.

Using an External Boot Device

If it is supported by the machine's BIOS and the installation kernel, boot for installation from external CD or DVD drives.

Network Boot via PXE

If a machine lacks a CD or DVD drive, but provides a working ethernet connection, perform a completely network-based installation. See [Section 4.1.3, “Remote Installation via VNC—PXE Boot and Wake on LAN”](#) (page 63) and [Section 4.1.6, “Remote Installation via SSH—PXE Boot and Wake on LAN”](#) (page 67) for details.

Booting from a Floppy Disk (SYSLINUX)

On some older computers, there is no bootable CD-ROM drive available, but a floppy disk drive. To install on such a system, create boot disks and boot your system with them. See [Section 7.5.3, “Boot and Rescue Disks”](#) (page 149) for directions for creating boot disks with YaST.

The boot disks include the loader SYSLINUX and the program linuxrc. SYSLINUX enables the selection of a kernel during the boot procedure and the specification of any parameters needed for the hardware used. The program linuxrc supports the loading of kernel modules for your hardware and subsequently starts the installation.

When booting from a boot disk, the boot procedure is initiated by the boot loader SYSLINUX (package `syslinux`). When the system is booted, SYSLINUX runs a minimum hardware detection that mainly consists of the following steps:

1. The program checks if the BIOS provides VESA 2.0–compliant framebuffer support and boots the kernel accordingly.
2. The monitor data (DDC info) is read.
3. The first block of the first hard disk (MBR) is read to map BIOS IDs to Linux device names during the boot loader configuration. The program attempts to read the block by means of the the lba32 functions of the BIOS to determine if the BIOS supports these functions.

If you keep `[Shift]` pressed when SYSLINUX starts, all these steps are skipped. For troubleshooting purposes, insert the line

```
verbose 1
```

in `syslinux.cfg` for the boot loader to display which action is currently being performed.

If the machine does not boot from the floppy disk, you may need to change the boot sequence in the BIOS to `A, C, CDROM`.

External Boot Devices

Most CD-ROM drives are supported. If problems arise when booting from the CD-ROM drive, try booting CD 2 of the CD set.

If the system does not have a CD-ROM or floppy disk, it is still possible that an external CD-ROM, connected with USB, FireWire, or SCSI, can be used to boot the system. This depends largely on the interaction of the BIOS and the hardware used. Sometimes a BIOS update may help if you encounter problems.

45.2.2 Booting from Installation Media Fails

There are two possible reasons for a machine not to boot for installation:

CD or DVD-ROM Drive Unable to Read the Boot Image

Your CD-ROM drive might not be able to read the boot image on CD 1. In this case, use CD 2 to boot the system. CD 2 contains a conventional 2.88 MB boot image that can be read even by unsupported drives and allows you to perform the installation over the network as described in [Chapter 4, Remote Installation](#) (page 59).

Incorrect Boot Sequence in BIOS

The BIOS boot sequence must have CD-ROM set as the first entry for booting. Otherwise the machine would try to boot from another medium, typically the hard disk. Guidance for changing the BIOS boot sequence can be found the documentation provided with your motherboard or in the following paragraphs.

The BIOS is the software that enables the very basic functions of a computer. Motherboard vendors provide a BIOS specifically made for their hardware. Normally, the BIOS setup can only be accessed at a specific time—when the machine is booting. During this initialization phase, the machine performs a number of diagnostic hardware tests. One of them is a memory check, indicated by a memory counter. When the counter appears, look for a line, usually below the counter or somewhere at the bottom, mentioning the key to press to access the BIOS setup. Usually the key to press is **Del**, **F1**, or **Esc**. Press this key until the BIOS setup screen appears.

Procedure 45.1 *Changing the BIOS Boot Sequence*

- 1 Enter the BIOS using the proper key as announced by the boot routines and wait for the BIOS screen to appear.
- 2 To change the boot sequence in an AWARD BIOS, look for the *BIOS FEATURES SETUP* entry. Other manufacturers may have a different name for this, such as *ADVANCED CMOS SETUP*. When you have found the entry, select it and confirm with **Enter**.
- 3 In the screen that opens, look for a subentry called *BOOT SEQUENCE*. The boot sequence is often set to something like *C, A* or *A, C*. In the former case, the machine first searches the hard disk (C) then the floppy drive (A) to find a bootable medium. Change the settings by pressing **PgUp** or **PgDown** until the sequence is *A, CDROM, C*.
- 4 Leave the BIOS setup screen by pressing **Esc**. To save the changes, select *SAVE & EXIT SETUP* or press **F10**. To confirm that your settings should be saved, press **Y**.

Procedure 45.2 *Changing the Boot Sequence in a SCSI BIOS (Adaptec Host Adapter)*

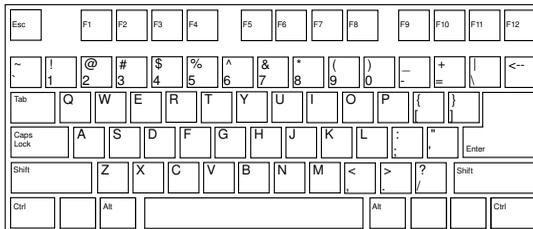
- 1 Open the setup by pressing **Ctrl** + **A**.
- 2 Select *Disk Utilities*, which displays the connected hardware components.

Make note of the SCSI ID of your CD-ROM drive.

- 3 Exit the menu with **[Esc]**.
- 4 Open *Configure Adapter Settings*. Under *Additional Options*, select *Boot Device Options* and press **[Enter]**.
- 5 Enter the ID of the CD-ROM drive and press **[Enter]** again.
- 6 Press **[Esc]** twice to return to the start screen of the SCSI BIOS.
- 7 Exit this screen and confirm with *Yes* to boot the computer.

Regardless of what language and keyboard layout your final installation will be using, most BIOS configurations use the US keyboard layout as depicted in the following figure:

Figure 45.1 *US Keyboard Layout*



45.2.3 Fails to Boot

Some hardware types, mainly fairly old or very recent ones, fail to install. In many cases, this might happen because support for this type of hardware is missing from the installation kernel or due to certain functionalities included in this kernel, such as ACPI, that still cause problems on some hardware.

If your system fails to install using the standard *Installation* mode from the first installation boot screen, try the following:

- 1 With the first CD or DVD still in the CD-ROM drive, reboot the machine with `Ctrl` + `Alt` + `Del` or using the hardware reset button.
- 2 When the boot screen appears, use the arrow keys of your keyboard to navigate to *Installation--ACPI Disabled* and press `Enter` to launch the boot and installation process. This option disables the support for ACPI power management techniques.
- 3 Proceed with the installation as described in [Chapter 3, *Installation with YaST*](#) (page 33).

If this fails, proceed as above, but choose *Installation--Safe Settings* instead. This option disables ACPI and DMA support. Most hardware should boot with this option.

If both of these options fail, use the boot options prompt to pass any additional parameters needed to support this type of hardware to the installation kernel. For more information about the parameters available as boot options, refer to the kernel documentation located in `/usr/src/linux/Documentation/kernel-parameters.txt`.

TIP: Obtaining Kernel Documentation

Install the `kernel-source` package to view the kernel documentation.

There are various other ACPI-related kernel parameters that can be entered at the boot prompt prior to booting for installation:

`acpi=off`

This parameter disables the complete ACPI subsystem on your computer. This may be useful if your computer cannot handle ACPI at all or if you think ACPI in your computer causes trouble.

`acpi=force`

Always enable ACPI even if your computer has an old BIOS dated before the year 2000. This parameter also enables ACPI if it is set in addition to `acpi=off`.

`acpi=noirq`

Do not use ACPI for IRQ routing.

`acpi=ht`

Run only enough ACPI to enable hyper-threading.

```
acpi=strict
```

Be less tolerant of platforms that are not strictly ACPI specification compliant.

```
pci=noacpi
```

Disable PCI IRQ routing of the new ACPI system.

Once you have determined the right parameter combination, YaST automatically writes them to the boot loader configuration to make sure that the system boots properly next time.

If unexplainable errors occur when the kernel is loaded or during the installation, select *Memory Test* in the boot menu to check the memory. If *Memory Test* returns an error, it is usually a hardware error.

45.2.4 Fails to Launch Graphical Installer

After you insert the first CD or DVD into your drive and reboot your machine, the installation screen comes up, but after you select *Installation*, the graphical installer does not start.

There are several ways to deal with this situation:

- Try to select another screen resolution for the installation dialogs.
- Select *Text Mode* for installation.
- Do a remote installation via VNC using the graphical installer.

To change to another screen resolution for installation, proceed as follows:

- 1 Boot for installation.
- 2 Press **F3** twice to open a menu from which to select a lower resolution for installation purposes.
- 3 Select *Installation* and proceed with the installation as described in [Chapter 3, Installation with YaST](#) (page 33).

To perform an installation in text mode, proceed as follows:

- 1 Boot for installation.
- 2 Press `[F3]` twice and select *Text Mode*.
- 3 Select *Installation* and proceed with the installation as described in [Chapter 3, *Installation with YaST*](#) (page 33).

To perform a VNC installation, proceed as follows:

- 1 Boot for installation.
- 2 Enter the following text at the boot options prompt:

```
vnc=1 vncpassword=some_password
```

Replace *some_password* with the password to use for installation.

- 3 Select *Installation* then press `[Enter]` to start the installation and select DHCP for network configuration when prompted to do so.

Instead of starting right into the graphical installation routine, the system continues to run in text mode then halts, displaying a message containing the IP address and port number at which the installer can be reached via a browser interface or a VNC viewer application.

- 4 If using a browser to access the installer, launch the browser and enter the address information provided by the installation routines on the future SUSE® Linux Enterprise machine and hit `[Enter]`:

```
http://ip_address_of_machine:5801
```

A dialog opens in the browser window prompting you for the VNC password. Enter it and proceed with the installation as described in [Chapter 3, *Installation with YaST*](#) (page 33).

IMPORTANT

Installation via VNC works with any browser under any operating system, provided Java support is enabled.

If you use any kind of VNC viewer on your preferred operating system, enter the IP address and password when prompted to do so. A window opens, displaying the installation dialogs. Proceed with the installation as usual.

45.2.5 Only Minimalistic Boot Screen Started

You inserted the first CD or DVD into the drive, the BIOS routines are finished, but the system does not start with the graphical boot screen. Instead it launches a very minimalistic text-based interface. This might happen on any machine not providing sufficient graphics memory for rendering a graphical boot screen.

Although the text boot screen looks minimalistic, it provides nearly the same functionality as the graphical one:

Boot Options

Unlike the graphical interface, the different boot options cannot be selected using the cursor keys of your keyboard. The boot menu of the text mode boot screen offers some keywords to enter at the boot prompt. These keywords map to the options offered in the graphical version. Enter your choice and hit `[Enter]` to launch the boot process.

Custom Boot Options

After selecting a boot option, enter the appropriate keyword at the boot prompt or enter some custom boot options as described in [Section 45.2.3, “Fails to Boot”](#) (page 782). To launch the installation process, press `[Enter]`.

Screen Resolutions

Use the F keys to determine the screen resolution for installation. If you need to boot in text mode, choose `[F3]`.

45.3 Boot Problems

Boot problems are situations when your system does not boot properly (does not boot to the expected runlevel and login screen).

45.3.1 Fails to Load the GRUB Boot Loader

If the hardware is functioning properly, it is possible that the boot loader has become corrupted and Linux cannot start on the machine. In this case, it is necessary to reinstall the boot loader. To reinstall the boot loader, proceed as follows:

- 1 Insert the installation media into the drive.
- 2 Reboot the machine.
- 3 Select *Installation* from the boot menu.
- 4 Select a language.
- 5 Accept the license agreement.
- 6 In the *Installation Mode* screen, select *Other* and set the installation mode to *Repair Installed System*.
- 7 Once in the YaST System Repair module, select *Expert Tools* then select *Install New Boot Loader*.
- 8 Restore the original settings and reinstall the boot loader.
- 9 Leave YaST System Repair and reboot the system.

Other reasons for the machine not booting may be BIOS-related:

BIOS Settings

Check your BIOS for references to your hard drive. GRUB might simply not be started if the hard drive itself cannot be found with the current BIOS settings.

BIOS Boot Order

Check whether your system's boot order includes the hard disk. If the hard disk option was not enabled, your system might install properly, but fail to boot when access to the hard disk is required.

45.3.2 No Login or Prompt Appears

This behavior typically occurs after a failed kernel upgrade and it is known as a *kernel panic* because of the type of error on the system console that sometimes can be seen at the final stage of the process. If, in fact, the machine has just been rebooted following a software update, the immediate goal is to reboot it using the old, proven version of the Linux kernel and associated files. This can be done in the GRUB boot loader screen during the boot process as follows:

- 1 Reboot the computer using the reset button.
- 2 When the GRUB boot screen becomes visible, select *Linux--Failsafe* then press `Enter`. The machine should boot using the prior version of the kernel and its associated files.
- 3 After the boot process has completed, remove the newly installed kernel and, if necessary, manually modify `/boot/grub/menu.lst` to indicate the older kernel as the default option. For some detailed information about the syntax used in this configuration file, refer to [Chapter 17, The Boot Loader](#) (page 387).

Updating this file might not be necessary because automated update tools usually modify it for you during the rollback process.

- 4 Reboot.

If this does not fix the problem because the *Linux--Failsafe* option does not boot the computer properly, boot the computer using the installation media. After the machine has booted, continue with [Step 3](#) (page 788) and [Step 4](#) (page 788).

45.3.3 No Graphical Login

If the machine comes up, but does not boot into the graphical login manager, anticipate problems either with the choice of the default runlevel or the configuration of the X Window System. To check the runlevel configuration, log in as the `root` user and check whether the machine is configured to boot into runlevel 5 (graphical desktop). A quick way to check this is to examine the contents of `/etc/inittab`, as follows:

```
nld-machine:~ # grep "id:" /etc/inittab
id:5:initdefault:
nld-machine:~ #
```

The returned line indicates that the machine's default runlevel (`initdefault`) is set to 5 and that it should boot to the graphical desktop. If the runlevel is set to any other number, use the YaST Runlevel Editor module to set it to 5.

IMPORTANT

Do not edit the runlevel configuration manually. Otherwise `SuSEconfig` (run by YaST) will overwrite these changes on its next run. If you need to make manual changes here, disable future `SuSEconfig` changes by setting `CHECK_INITTAB` in `/etc/sysconfig/suseconfig` to `no`.

If the runlevel is set to 5, you might have corruption problems with your desktop or X Windows software. Examine the log files at `/var/log/Xorg.*.log` for detailed messages from the X server as it attempted to start. If the desktop fails during start, it might log error messages to `/var/log/messages`. If these error messages hint at a configuration problem in the X server, try to fix these issues. If the graphical system still does not come up, consider reinstalling the graphical desktop.

One quick test: the `startx` command should force the X Window System to start with the configured defaults if the user is currently logged in on the console. If that does not work, it should log errors to the console. For more information about the X Window system configuration, refer to [Chapter 22, *The X Window System*](#) (page 465).

45.4 Login Problems

Login problems are those where your machine does, in fact, boot to the expected welcome screen or login prompt, but refuses to accept the username and password or accepts them but then does not behave properly (fails to start the graphic desktop, produces errors, drops to a command line, etc.).

45.4.1 Valid Username and Password Combinations Fail

This usually occurs when the system is configured to use network authentication or directory services and, for some reason, is unable to retrieve results from its configured servers. The `root` user, as the only local user, is the only user that can still log in to

these machines. The following are some common reasons why a machine might appear functional but be unable to process logins correctly:

- The network is not working. For further directions on this, turn to [Section 45.5, “Network Problems”](#) (page 796).
- DNS is not working at the moment (which prevents GNOME or KDE from working and the system from making validated requests to secure servers). One indication that this is the case is that the machine takes an extremely long time to respond to any action. Find more information about this topic in [Section 45.5, “Network Problems”](#) (page 796).
- If the system is configured to use Kerberos, the system's local time might have drifted past the accepted variance with the Kerberos server time (this is typically 300 seconds). If NTP (network time protocol) is not working properly or local NTP servers are not working, Kerberos authentication ceases to function because it depends on common clock synchronization across the network.
- The system's authentication configuration is misconfigured. Check the PAM configuration files involved for any typographical errors or misordering of directives. For additional background information about PAM and the syntax of the configuration files involved, refer to [Chapter 23, Authentication with PAM](#) (page 483).

In all cases that do not involve external network problems, the solution is to reboot the system into single-user mode and repair the configuration before booting again into operating mode and attempting to log in again. To boot into single-user mode:

- 1** Reboot the system. The boot screen appears, offering a prompt.
- 2** Enter `1` at the boot prompt to make the system boot into single-user mode.
- 3** Enter the username and password for `root`.
- 4** Make all the necessary changes.
- 5** Boot into the full multiuser and network mode by entering `telinit 5` at the command line.

45.4.2 Valid Username and Password Not Accepted

This is by far the most common problem users encounter, because there are many reasons this can occur. Depending on whether you use local user management and authentication or network authentication, login failures occur for different reasons.

Local user management can fail for the following reasons:

- The user might have entered the wrong password.
- The user's home directory containing the desktop configuration files is corrupted or write protected.
- There might be problems with the X Window System authenticating this particular user, especially if the user's home directory has been used with another Linux distribution prior to installing the current one.

To locate the reason for a local login failure, proceed as follows:

- 1** Check whether the user remembered his password correctly before you start debugging the whole authentication mechanism. If the user might not remember his password correctly, use the YaST User Management module to change the user's password.
- 2** Log in as `root` and check `/var/log/messages` for error messages of the login process and of PAM.
- 3** Try to log in from a console (using `Ctrl` + `Alt` + `F1`). If this is successful, the blame cannot be put on PAM, because it is possible to authenticate this user on this machine. Try to locate any problems with the X Window System or the desktop (GNOME or KDE). For more information, refer to [Section 45.4.3, “Login Successful but GNOME Desktop Fails”](#) (page 794) and [Section 45.4.4, “Login Successful but KDE Desktop Fails”](#) (page 795).
- 4** If the user's home directory has been used with another Linux distribution, remove the `Xauthority` file in the user's home. Use a console login via `Ctrl` + `Alt` + `F1` and run `rm .Xauthority` as this user. This should eliminate X authentication problems for this user. Try a graphical login again.

- 5 If graphical login still fails, do a console login with `Ctrl` + `Alt` + `F1`. Try to start an X session on another display—the first one (: 0) is already in use:

```
startx -- :1
```

This should bring up a graphical screen and your desktop. If it does not, check the log files of the X Window System (`/var/log/Xorg.displaynumber.log`) or the log file for your desktop applications (`.xsession-errors` in the user's home directory) for any irregularities.

- 6 If the desktop could not start because of corrupt configuration files, proceed with [Section 45.4.3, “Login Successful but GNOME Desktop Fails”](#) (page 794) or [Section 45.4.4, “Login Successful but KDE Desktop Fails”](#) (page 795).

The following are some common reasons why network authentication for a particular user might fail on a specific machine:

- The user might have entered the wrong password.
- The username exists in the machine's local authentication files and is also provided by a network authentication system, causing conflicts.
- The home directory exists but is corrupt or unavailable. Perhaps it is write protected or is on a server that is inaccessible at the moment.
- The user does not have permission to log in to that particular host in the authentication system.
- The machine has changed hostnames, for whatever reason, and the user does not have permission to log in to that host.
- The machine cannot reach the authentication server or directory server that contains that user's information.
- There might be problems with the X Window System authenticating this particular user, especially if the user's home has been used with another Linux distribution prior to installing the current one.

To locate the cause of the login failures with network authentication, proceed as follows:

- 1 Check whether the user remembered his password correctly before you start debugging the whole authentication mechanism.

- 2 Determine the directory server the machine relies on for authentication and make sure that it is up and running and properly communicating with the other machines.
- 3 Determine that the user's username and password work on other machines to make sure that his authentication data exists and is properly distributed.
- 4 See if another user can log in to the misbehaving machine. If another user can log in without difficulty or if `root` can log in, log in and examine the `/var/log/messages` file. Locate the time stamps that correspond to the login attempts and determine if PAM has produced any error messages.
- 5 Try to log in from a console (using `Ctrl` + `Alt` + `F1`). If this is successful, the blame cannot be put on PAM or the directory server on which the user's home is hosted, because it is possible to authenticate this user on this machine. Try to locate any problems with the X Window System or the desktop (GNOME or KDE). For more information, refer to [Section 45.4.3, “Login Successful but GNOME Desktop Fails”](#) (page 794) and [Section 45.4.4, “Login Successful but KDE Desktop Fails”](#) (page 795).
- 6 If the user's home directory has been used with another Linux distribution, remove the `Xauthority` file in the user's home. Use a console login via `Ctrl` + `Alt` + `F1` and run `rm .Xauthority` as this user. This should eliminate X authentication problems for this user. Try a graphical login again.
- 7 If graphical login still fails, do a console login with `Ctrl` + `Alt` + `F1`. Try to start an X session on another display—the first one (`:0`) is already in use:

```
startx -- :1
```

This should bring up a graphical screen and your desktop. If it does not, check the log files of the X Window System (`/var/log/Xorg.displaynumber.log`) or the log file for your desktop applications (`.xsession-errors` in the user's home directory) for any irregularities.

- 8 If the desktop could not start because of corrupt configuration files, proceed with [Section 45.4.3, “Login Successful but GNOME Desktop Fails”](#) (page 794) or [Section 45.4.4, “Login Successful but KDE Desktop Fails”](#) (page 795).

45.4.3 Login Successful but GNOME Desktop Fails

If this is true for a particular user, it is likely that the user's GNOME configuration files have become corrupted. Some symptoms might include the keyboard failing to work, the screen geometry becoming distorted, or even the screen coming up as a bare gray field. The important distinction is that if another user logs in, the machine works normally. If this is the case, it is likely that the problem can be fixed relatively quickly by simply moving the user's GNOME configuration directory to a new location, which causes GNOME to initialize a new one. Although the user is forced to reconfigure GNOME, no data is lost.

1 Log in as `root`.

2 `cd` to the user's home directory.

3 Move the user's GNOME configuration directories to a temporary location:

```
mv ~/.gconf ~/.gconf-ORIG-RECOVER
mv ~/.gnome2 ~/.gnome2-ORIG-RECOVER
```

4 Log out.

5 Have the user log in, but do not allow him to run any applications.

6 Recover the user's individual application configuration data (including the Evolution e-mail client data) by copying the `~/.gconf-ORIG-RECOVER/apps/` directory back into the new `~/.gconf` directory as follows:

```
cp -a ~/.gconf-ORIG-RECOVER/apps ~/.gconf/
```

If this causes the login problems, attempt to recover only the critical application data and force the user to reconfigure the remainder of the applications.

45.4.4 Login Successful but KDE Desktop Fails

There are several reasons why a KDE desktop would not allow users to login. Corrupted cache data can cause login problems as well as corrupt KDE desktop configuration files.

Cache data is used at desktop start-up to increase performance. If this data is corrupted, start-up is slowed down or fails entirely. Removing them forces the desktop start-up routines to start from scratch. This takes more time than a normal start-up, but data is intact after this and the user can login.

To remove the cache files of the KDE desktop, issue the following command as `root`:

```
rm -rf /tmp/kde-user /tmp/socket-user
```

Replace `user` with the actual username. Removing these two directories just removes the corrupted cache files. No real data is harmed using this procedure.

Corrupted desktop configuration files can always be replaced with the initial configuration files. If you want to recover the user's adjustments, carefully copy them back from their temporary location after the configuration has been restored using the default configuration values.

To replace a corrupted desktop configuration with the initial configuration values, proceed as follows:

- 1 Log in as `root`.

- 2 Enter the user's home directory:

```
cd /home/user
```

- 3 Move the KDE configuration directory and the `.skel` files to a temporary location:

```
mv .kde .kde-ORIG-RECOVER
mv .skel .skel-ORIG-RECOVER
```

- 4 Log out.

- 5 Let the user log in to this machine.
- 6 After the desktop has started successfully, copy the user's own configurations back into place:

```
cp -a .kde-ORIG-RECOVER/share .kde/share
```

IMPORTANT

If the user's own adjustments caused the login to fail and continue to do so, repeat the procedure as described above, but do not copy the `.kde/share` directory.

45.5 Network Problems

Many problems of your system may be network-related, even though they do not seem to be at first. For example, the reason for a system not allowing users to log in might be a network problem of some kind. This section introduces a simple check list you can apply to identify the cause of any network problem encountered.

When checking the network connection of your machine, proceed as follows:

- 1 If using an ethernet connection, check the hardware first. Make sure that your network cable is properly plugged into your computer. The control lights next to your ethernet connector, if available, should both be active.

If the connection fails, check whether your network cable works with another machine. If it does, your network card causes the failure. If hubs or switches are included in your network setup, suspect them to be the culprits as well.

- 2 If using a wireless connection, check whether the wireless link can be established by other machines. If this is not the case, contact the wireless network's administrator.
- 3 Once you have checked your basic network connectivity, try to find out which service is not responding. Gather the address information of all network servers needed in your setup. Either look them up in the appropriate YaST module or ask your system administrator. The following list gives some of the typical network servers involved in a setup together with the symptoms of an outage.

DNS (Name Service)

A broken or malfunctioning name service affects the network's functioning in many ways. If the local machine relies on any network servers for authentication and these servers cannot be found due to name resolution issues, users would not even be able to log in. Machines in the network managed by a broken name server would not be able to “see” each other and communicate.

NTP (Time Service)

A malfunctioning or completely broken NTP service could affect Kerberos authentication and X server functionality.

NFS (File Service)

If any application needed data stored in an NFS mounted directory, it would not be able to start or function properly if this service was down or misconfigured. In a worst case scenario, a user's personal desktop configuration would not come up if his home directory containing the `.gconf` or `.kde` subdirectories could not be found due to an outage of the NFS server.

Samba (File Service)

If any application needed data stored in a directory on a Samba server, it would not be able to start or function properly if this service was down.

NIS (User Management)

If your SUSE Linux Enterprise system relied on a NIS server to provide the user data, users would not be able to log in to this machine if the NIS service was down.

LDAP (User Management)

If your SUSE Linux Enterprise system relied on an LDAP server to provide the user data, users would not be able to log in to this machine if the LDAP service was down.

Kerberos (Authentication)

Authentication would not work and login to any machine would fail.

CUPS (Network Printing)

Users would not be able to print.

- 4 Check whether the network servers are running and whether your network setup allows you to establish a connection:

IMPORTANT

The debugging procedure described below only applies to a simple network server/client setup that does not involve any internal routing. It assumes both server and client are members of the same subnet without the need for additional routing.

- a Use `ping hostname` (replace *hostname* with the hostname of the server) to check whether each one of them is up and responding to the network. If this command is successful, it tells you that the host you were looking for is up and running and that the name service for your network is configured correctly.

If `ping` fails with `destination host unreachable`, either your system or the desired server is not properly configured or down. Check whether your system is reachable by running `ping your_hostname` from another machine. If you can reach your machine from another machine, it is the server that is not running at all or not configured correctly.

If `ping` fails with `unknown host`, the name service is not configured correctly or the hostname used was incorrect. Use `ping -n ipaddress` to try to connect to this host without name service. If this is successful, check the spelling of the hostname and for a misconfigured name service in your network. For further checks on this matter, refer to [Step 4.b](#) (page 798). If `ping` still fails, either your network card is not configured correctly or your network hardware is faulty. Refer to [Step 4.c](#) (page 799) for information about this.

- b Use `host hostname` to check whether the hostname of the server you are trying to connect to is properly translated into an IP address and vice versa. If this command returns the IP address of this host, the name service is up and running. If the `host` command fails, check all network configuration files relating to name and address resolution on your host:

```
/etc/resolv.conf
```

This file is used to keep track of the name server and domain you are currently using. It can be modified manually or automatically adjusted by YaST or DHCP. Automatic adjustment is preferable. However, make sure that this file has the following structure and all network addresses and domain names are correct:

```
search fully_qualified_domain_name
nameserver ipaddress_of_nameserver
```

This file can contain more than one name server address, but at least one of them must be correct to provide name resolution to your host. If needed, adjust this file using the YaST DNS and Hostname module.

If your network connection is handled via DHCP, enable DHCP to change hostname and name service information by selecting *Change Hostname via DHCP* and *Update Name Servers and Search List via DHCP* in the YaST DNS and Hostname module.

```
/etc/nsswitch.conf
```

This file tells Linux where to look for name service information. It should look like this:

```
...
hosts: files dns
networks: files dns
...
```

The `dns` entry is vital. It tells Linux to use an external name server. Normally, these entries are automatically made by YaST, but it never hurts to check.

If all the relevant entries on the host are correct, let your system administrator check the DNS server configuration for the correct zone information. If you have made sure that the DNS configuration of your host and the DNS server are correct, proceed with checking the configuration of your network and network device.

- c If your system cannot establish a connection to a network server and you have excluded name service problems from the list of possible culprits, check the configuration of your network card.

Use the command `ifconfig network_device` (executed as `root`) to check whether this device was properly configured. Make sure that both `inet address` and `Mask` are configured correctly. An error in the IP address or a missing bit in your network mask would render your network configuration unusable. If necessary, perform this check on the server as well.

- d** If the name service and network hardware are properly configured and running, but some external network connections still get long time-outs or fail entirely, use `traceroute fully_qualified_domain_name` (executed as `root`) to track the network route these requests are taking. This command lists any gateway (hop) a request from your machine passes on its way to its destination. It lists the response time of each hop and whether this hop is reachable at all. Use a combination of `traceroute` and `ping` to track down the culprit and let the administrators know.

Once you have identified the cause of your network trouble, you can resolve it yourself (if the problem is located on your machine) or let the system administrators of your network know about your findings so they can reconfigure the services or repair the necessary systems.

45.6 Data Problems

Data problems are when the machine might or might not boot properly but, in either case, it is clear that there is data corruption on the system and that the system needs to be recovered. These situations call for a backup of your critical data, enabling you to recover a system state from before your system failed. SUSE Linux Enterprise offers dedicated YaST modules for system backup and restoration as well as a rescue system that can be used to recover a corrupted system from the outside.

45.6.1 Backing Up Critical Data

System backups can be easily managed using the YaST System Backup module:

- 1** As `root`, start YaST and select *System* → *System Backup*.
- 2** Create a backup profile holding all details needed for the backup, filename of the archive file, scope, and type of the backup:
 - a** Select *Profile Management* → *Add*.
 - b** Enter a name for the archive.

- c** Enter the path to the location of the backup if you want to keep a local backup. For your backup to be archived on a network server (via NFS), enter the IP address or name of the server and the directory that should hold your archive.
- d** Determine the archive type and click *Next*.
- e** Determine the backup options to use, such as whether files not belonging to any package should be backed up and whether a list of files should be displayed prior to creating the archive. Also determine whether changed files should be identified using the time-consuming MD5 mechanism.

Use *Expert* to enter a dialog for the backup of entire hard disk areas. Currently, this option only applies to the Ext2 file system.

- f** Finally, set the search constraints to exclude certain system areas from the backup area that do not need to be backed up, such as lock files or cache files. Add, edit, or delete items until your needs are met and leave with *OK*.

- 3** Once you have finished the profile settings, you can start the backup right away with *Create Backup* or configure automatic backup. It is also possible to create other profiles tailored for various other purposes.

To configure automatic backup for a given profile, proceed as follows:

- 1** Select *Automatic Backup* from the *Profile Management* menu.
- 2** Select *Start Backup Automatically*.
- 3** Determine the backup frequency. Choose *daily*, *weekly*, or *monthly*.
- 4** Determine the backup start time. These settings depend on the backup frequency selected.
- 5** Decide whether to keep old backups and how many should be kept. To receive an automatically generated status message of the backup process, check *Send Summary Mail to User root*.
- 6** Click *OK* to apply your settings and have the first backup start at the time specified.

45.6.2 Restoring a System Backup

Use the YaST System Restoration module to restore the system configuration from a backup. Restore the entire backup or select specific components that were corrupted and need to be reset to their old state.

- 1 Start *YaST* → *System* → *System Restoration*.
- 2 Enter the location of the backup file. This could be a local file, a network mounted file, or a file on a removable device, such as a floppy or a CD. Then click *Next*.

The following dialog displays a summary of the archive properties, such as the filename, date of creation, type of backup, and optional comments.

- 3 Review the archived content by clicking *Archive Content*. Clicking *OK* returns you to the *Archive Properties* dialog.
- 4 *Expert Options* opens a dialog in which to fine-tune the restore process. Return to the *Archive Properties* dialog by clicking *OK*.
- 5 Click *Next* to open the view of packages to restore. Press *Accept* to restore all files in the archive or use the various *Select All*, *Deselect All*, and *Select Files* buttons to fine-tune your selection. Only use the *Restore RPM Database* option if the RPM database is corrupted or deleted and this file is included in the backup.
- 6 After you click *Accept*, the backup is restored. Click *Finish* to leave the module after the restore process is completed.

45.6.3 Recovering a Corrupted System

There are several reasons why a system could fail to come up and run properly. A corrupted file system after a system crash, corrupted configuration files, or a corrupted boot loader configuration are the most common ones.

SUSE Linux Enterprise offers two different methods to cope with this kind of situation. You can either use the YaST System Repair functionality or boot the rescue system. The following sections cover both flavors of system repair.

Using YaST System Repair

Before launching the YaST System Repair module, determine in which mode to run it to best fit your needs. Depending on the severeness and cause of your system failure and your expertise, there are three different modes to choose from:

Automatic Repair

If your system failed due to an unknown cause and you basically do not know which part of the system is to blame for the failure, use *Automatic Repair*. An extensive automated check will be performed on all components of your installed system. For a detailed description of this procedure, refer to [Section “Automatic Repair”](#) (page 803).

Customized Repair

If your system failed and you already know which component is to blame, you can cut the lengthy system check with *Automatic Repair* short by limiting the scope of the system analysis to those components. For example, if the system messages prior to the failure seem to indicate an error with the package database, you can limit the analysis and repair procedure to checking and restoring this aspect of your system. For a detailed description of this procedure, refer to [Section “Customized Repair”](#) (page 805).

Expert Tools

If you already have a clear idea of what component failed and how this should be fixed, you can skip the analysis runs and directly apply the tools necessary for the repair of the respective component. For details, refer to [Section “Expert Tools”](#) (page 806).

Choose one of the repair modes as described above and proceed with the system repair as outlined in the following sections.

Automatic Repair

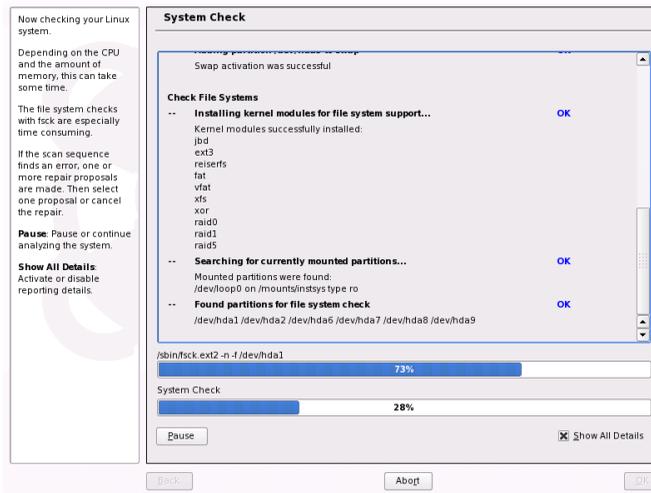
To start the automatic repair mode of YaST System Repair, proceed as follows:

- 1 Boot the system with the original installation medium used for the initial installation (as outlined in [Chapter 3, *Installation with YaST*](#) (page 33)).
- 2 In *System Analysis*, select *Other* → *Repair Installed System*.

3 Select *Automatic Repair*.

YaST now launches an extensive analysis of the installed system. The progress of the procedure is displayed at the bottom of the screen with two progress bars. The upper bar shows the progress of the currently running test. The lower bar shows the overall progress of the analysis. The log window in the top section tracks the currently running test and its result. See [Figure 45.2, “Automatic Repair Mode”](#) (page 804). The following main test runs are performed with every run. They contain, in turn, a number of individual subtests.

Figure 45.2 *Automatic Repair Mode*



Partition Tables of All Hard Disks

Checks the validity and coherence of the partition tables of all detected hard disks.

Swap Partitions

The swap partitions of the installed system are detected, tested, and offered for activation where applicable. The offer should be accepted for the sake of a higher system repair speed.

File Systems

All detected file systems are subjected to a file system–specific check.

Entries in the File `/etc/fstab`

The entries in the file are checked for completeness and consistency. All valid partitions are mounted.

Boot Loader Configuration

The boot loader configuration of the installed system (GRUB or LILO) is checked for completeness and coherence. Boot and root devices are examined and the availability of the `initrd` modules is checked.

Package Database

This checks whether all packages necessary for the operation of a minimal installation are present. While it is optionally possible also to analyze the base packages, this takes a long time because of their vast number.

- 4 Whenever an error is encountered, the procedure stops and a dialog opens outlining the details and possible solutions.

Read the screen messages carefully before accepting the proposed fix. If you decide to decline a proposed solution, your system remains unchanged.

- 5 After the repair process has been terminated successfully, click *OK* and *Finish* and remove the installation media. The system automatically reboots.

Customized Repair

To launch the *Customized Repair* mode and selectively check certain components of your installed system, proceed as follows:

- 1 Boot the system with the original installation medium used for the initial installation (as outlined in [Chapter 3, *Installation with YaST*](#) (page 33)).
- 2 In *System Analysis*, select *Other* → *Repair Installed System*.
- 3 Select *Customized Repair*.

Choosing *Customized Repair* shows a list of test runs that are all marked for execution at first. The total range of tests matches that of automatic repair. If you already know where no damage is present, unmark the corresponding tests. Clicking *Next* starts a narrower test procedure that probably has a significantly shorter running time.

Not all test groups can be applied individually. The analysis of the `fstab` entries is always bound to an examination of the file systems, including existing swap partitions. YaST automatically resolves such dependencies by selecting the smallest number of necessary test runs.

- 4 Whenever an error is encountered, the procedure stops and a dialog opens outlining the details and possible solutions.

Read the screen messages carefully before accepting the proposed fix. If you decide to decline a proposed solution, your system remains unchanged.

- 5 After the repair process has been terminated successfully, click *OK* and *Finish* and remove the installation media. The system automatically reboots.

Expert Tools

If you are knowledgeable with SUSE® Linux Enterprise and already have a very clear idea of what needs to be repaired in your system, directly apply the tools skipping the system analysis.

To make use of the *Expert Tools* feature of the YaST System Repair module, proceed as follows:

- 1 Boot the system with the original installation medium used for the initial installation (as outlined in [Chapter 3, *Installation with YaST*](#) (page 33)).
- 2 In *System Analysis*, select *Other* → *Repair Installed System*.
- 3 Select *Expert Tools* and choose one or more repair options.
- 4 After the repair process has been terminated successfully, click *OK* and *Finish* and remove the installation media. The system automatically reboots.

Expert tools provides the following options to repair your faulty system:

Install New Boot Loader

This starts the YaST boot loader configuration module. Find details in [Section 17.3, “Configuring the Boot Loader with YaST”](#) (page 397).

Start Partitioning Tool

This starts the expert partitioning tool in YaST. Find details in [Section 7.5.6, “Partitioner”](#) (page 150).

Repair File System

This checks the file systems of your installed system. You are first offered a selection of all detected partitions and can then choose the ones to check.

Recover Lost Partitions

It is possible to attempt to reconstruct damaged partition tables. A list of detected hard disks is presented first for selection. Clicking *OK* starts the examination. This can take a while depending on the processing power and size of the hard disk.

IMPORTANT: Reconstructing a Partition Table

The reconstruction of a partition table is tricky. YaST attempts to recognize lost partitions by analyzing the data sectors of the hard disk. The lost partitions are added to the rebuilt partition table when recognized. This is, however, not successful in all imaginable cases.

Save System Settings to Floppy

This option saves important system files to a floppy disk. If one of these files become damaged, it can be restored from disk.

Verify Installed Software

This checks the consistency of the package database and the availability of the most important packages. Any damaged installed packages can be reinstalled with this tool.

Using the Rescue System

Your Linux system contains a rescue system. The rescue system is a small Linux system that can be loaded into a RAM disk and mounted as root file system, allowing you to access your Linux partitions from the outside. Using the rescue system, you can recover or modify any important aspect of your system:

- Manipulate any type of configuration file.
- Check the file system for defects and start automatic repair processes.

- Access the installed system in a “change root” environment
- Check, modify, and reinstall the boot loader configuration
- Resize partitions using the parted command. Find more information about this tool at the Web site of GNU Parted (<http://www.gnu.org/software/parted/parted.html>).

The rescue system can be loaded from various sources and locations. The simplest option is to boot the rescue system from the original installation CD or DVD:

- 1 Insert the installation medium into your CD or DVD drive.
- 2 Reboot the system.
- 3 At the boot screen, choose the *Rescue System* option.
- 4 Enter `root` at the `Rescue :` prompt. A password is not required.

If your hardware setup does not include a CD or DVD drive, you can boot the rescue system from a network source (including the SUSE FTP server). The following example applies to a remote boot scenario—if using another boot medium, such as a floppy disk, modify the `info` file accordingly and boot as you would for a normal installation.

- 1 Enter the configuration of your PXE boot setup and replace `install=protocol://instsource` with `rescue=protocol://instsource`. As with a normal installation, `protocol` stands for any of the supported network protocols (NFS, HTTP, FTP, etc.) and `instsource` for the path to your network installation source.
- 2 Boot the system using Wake on LAN.
- 3 Enter `root` at the `Rescue :` prompt. A password is not required.

Once you have entered the rescue system, you can make use of the virtual consoles that can be reached with `[Alt] + [F1]` to `[Alt] + [F6]`.

A shell and many other useful utilities, such as the `mount` program, are available in the `/bin` directory. The `sbin` directory contains important file and network utilities for reviewing and repairing the file system. This directory also contains the most important binaries for system maintenance, such as `fdisk`, `mkfs`, `mkswap`, `mount`, `mount`, `init`, and

shutdown, and ifconfig, ip, route, and netstat for maintaining the network. The directory `/usr/bin` contains the vi editor, find, less, and ssh.

To see the system messages, either use the command `dmesg` or view the file `/var/log/messages`.

Checking and Manipulating Configuration Files

As an example for a configuration that might be fixed using the rescue system, imagine you have a broken configuration file that prevents the system from booting properly. You can fix this using the rescue system.

To manipulate a configuration file, proceed as follows:

- 1 Start the rescue system using one of the methods described above.
- 2 To mount a root file system located under `/dev/sda6` to the rescue system, use the following command:

```
mount /dev/sda6 /mnt
```

All directories of the system are now located under `/mnt`

- 3 Change the directory to the mounted root file system:

```
cd /mnt
```

- 4 Open the problematic configuration file in the vi editor. Adjust and save the configuration.

- 5 Unmount the root file system from the rescue system:

```
umount /mnt
```

- 6 Reboot the machine.

Repairing and Checking File Systems

Generally, file systems cannot be repaired on a running system. If you encounter serious problems, you may not even be able to mount your root file system and the system boot may end with a `kernel panic`. In this case, the only way is to repair the system from the outside. It is strongly recommended to use the YaST System Repair for this

task (see [Section “Using YaST System Repair”](#) (page 803) for details). However, if you need to do a manual file system check or repair, boot the rescue system. It contains the utilities to check and repair the `ext2`, `ext3`, `reiserfs`, `xfs`, `jfs`, `dosfs`, and `vfat` file systems.

Accessing the Installed System

If you need to access the installed system from the rescue system to, for example, modify the boot loader configuration, or to execute a hardware configuration utility, you need to do this in a “change root” environment.

To set up a “change root” environment based on the installed system, proceed as follows:

- 1 First mount the root partition from the installed system and the device file system:

```
mount /dev/sda6 /mnt
mount --bind /dev /mnt/dev
```

- 2 Now you can “change root” into the new environment:

```
chroot /mnt
```

- 3 Then mount `/proc` and `/sys`:

```
mount /proc
mount /sys
```

- 4 Finally, mount the remaining partitions from the installed system:

```
mount -a
```

- 5 Now you have access to the installed system. Before rebooting the system, unmount the partitions with `umount -a` and leave the “change root” environment with `exit`.

WARNING: Limitations

Although you have full access to the files and applications of the installed system, there are some limitations. The kernel that is running is the one that was booted with the rescue system. It only supports essential hardware and it is not possible to add kernel modules from the installed system unless the kernel

versions are exactly the same (which is unlikely). So you cannot access a sound card, for example. It is also not possible to start a graphical user interface.

Also note that you leave the “change root” environment when you switch the console with `Alt + F1` to `Alt + F6`.

Modifying and Reinstalling the Boot Loader

Sometimes a system cannot boot because the boot loader configuration is corrupted. The start-up routines cannot, for example, translate physical drives to the actual locations in the Linux file system without a working boot loader.

To check the boot loader configuration and reinstall the boot loader, proceed as follows:

- 1 Perform the necessary steps to access the installed system as described in [Section “Accessing the Installed System”](#) (page 810).
- 2 Check whether the following files are correctly configured according to the GRUB configuration principles outlined in [Chapter 17, *The Boot Loader*](#) (page 387).

- `/etc/grub.conf`
- `/boot/grub/device.map`
- `/boot/grub/menu.lst`

Apply fixes to the device mapping (`device.map`) or the location of the root partition and configuration files, if necessary.

- 3 Reinstall the boot loader using the following command sequence:

```
grub --batch < /etc/grub.conf
```

- 4 Unmount the partitions, log out from the “change root” environment, and reboot the system:

```
umount -a
exit
reboot
```


Index

Symbols

- 64-bit Linux, 367
 - kernel specifications, 370
 - runtime support, 367
 - software development, 368

A

- access permissions (see permissions)
- ACLs, 299–310
 - access, 301, 304
 - check algorithm, 309
 - default, 302, 307
 - definitions, 301
 - effects, 307
 - handling, 302
 - masks, 306
 - permission bits, 303
 - structure, 302
 - support, 310
- ACPI
 - disabling, 36
- add-on products, 134
- authentication
 - Kerberos, 195
 - PAM, 483–490
- AutoYaST, 167
 - cloning system, 58

B

- backups, 138
 - creating with YaST, 148
 - restoring, 148
- Bash, 334–345
 - .bashrc, 408
 - .profile, 408
 - commands, 334

- features, 341
- pipes, 343
- profile, 407
- wild cards, 342

BIOS

- boot sequence, 781

Bluetooth, 497, 559

- hciconfig, 565
- hcitool, 565
- network, 563
- opd, 567
- pand, 566
- sdptool, 566

booting, 371

- boot sectors, 387–388
- CD, from, 780
- configuring, 49
 - YaST, 397–402
- floppy disks, from, 779
- graphic, 403
- GRUB, 387–406
- initramfs, 373
- initrd, 373
- log, 168

bzip2, 345

C

cards

- graphics, 471
- network, 594
- radio, 146
- sound, 147
- TV, 146

cat, 355

cd, 351

CDs

- booting from, 780
 - checking, 140
- cellular phones, 499

- chgrp, 349, 352
- chmod, 348, 352
- chown, 349, 351
- CJK, 415
- clear, 360
- commands, 349–360
 - bzip2, 345
 - cat, 355
 - cd, 351
 - chgrp, 349, 352
 - chmod, 348, 352
 - chown, 349, 351
 - clear, 360
 - cp, 350
 - date, 358
 - df, 357
 - diff, 355
 - du, 357
 - file, 354
 - find, 354
 - fonts-config, 473
 - free, 357, 412
 - getfacl, 305
 - grep, 355
 - grub, 388
 - gzip, 345, 353
 - halt, 360
 - help, 336
 - ifconfig, 625
 - ip, 622
 - kill, 358
 - killall, 358
 - less, 355
 - ln, 351
 - locate, 354
 - lp, 431
 - ls, 350
 - man, 349
 - mkdir, 351
 - mount, 356
 - mv, 350
 - nslookup, 359
 - passwd, 359
 - ping, 358, 624
 - ps, 358
 - reboot, 360
 - rm, 351
 - rmdir, 351
 - route, 626
 - scp, 720
 - setfacl, 305
 - sftp, 721
 - slptool, 632
 - smbpasswd, 678
 - ssh, 720
 - ssh-agent, 723
 - ssh-keygen, 722
 - su, 360
 - tar, 344, 353
 - telnet, 359
 - top, 358
 - umount, 356
 - updatedb, 354
- configuration files, 615
 - .bashrc, 408, 411
 - .emacs, 413
 - .mailsync, 702
 - .profile, 408
 - .xsession, 723
 - acpi, 527
 - asound.conf, 147
 - crontab, 408
 - csh.cshrc, 417
 - dhcp, 616
 - fstab, 153, 356
 - group, 186
 - grub.conf, 395
 - host.conf, 619
 - HOSTNAME, 622
 - hosts, 161, 593, 618

- ifcfg-*, 616
- inittab, 375, 377, 414
- inputrc, 415
- irda, 571
- kernel, 373
- language, 415, 417
- logrotate.conf, 409
- menu.lst, 390
- modprobe.d/sound, 147
- network, 616
- networks, 618
- nscd.conf, 622
- nsswitch.conf, 620, 659
- pam_unix2.conf, 658
- passwd, 186
- permissions, 759
- powersave, 527
- powersave.conf, 203
- profile, 407, 411, 417
- resolv.conf, 412, 617
- routes, 616
- samba, 672
- services, 672
- smb.conf, 673, 679
- smppd.conf, 629
- smpppd-c.conf, 629
- sshd_config, 724
- suseconfig, 386
- sysconfig, 156, 383–386
- termcap, 415
- wireless, 616
- XF86Config, 198
- xorg.conf, 198, 467
 - Device, 471
 - Monitor, 472
 - Screen, 469
- configuring, 383
 - answering machines, 159
 - Bluetooth, 141
 - cable modem, 607
 - DNS, 161
 - DSL, 158, 607
 - e-mail, 160
 - fax systems, 159
 - firewalls, 167
 - graphics cards, 141, 176
 - groups, 165
 - GRUB, 388, 395
 - hard disk controllers, 142
 - hard disks
 - DMA, 142
 - hardware, 140–148
 - IPv6, 591
 - IrDA, 571
 - ISDN, 158, 604
 - languages, 157
 - modems, 158, 602
 - monitor, 141, 176
 - network cards, 158
 - networks, 158–163, 594
 - manually, 612–628
 - NFS, 161
 - NTP, 161
 - PAM, 201
 - power management, 155
 - powertweak, 156
 - printing, 424–427
 - radio, 146
 - routing, 163, 616
 - Samba, 671–677
 - clients, 163, 677
 - scanner, 144
 - security, 163–167
 - software, 127–138
 - sound cards, 147
 - SSH, 719
 - system, 125–170
 - system services, 162
 - T-DSL, 609
 - time zone, 157

- TV, 146
- users, 164
- wireless cards, 158
- consoles
 - assigning, 414
 - graphical, 403
 - switching, 414
- core files, 411
- cp, 350
- cpuspeed, 535
- cron, 408
- CVS, 686, 694–696

D

- data security, 498
- date, 358
- df, 357
- diff, 355
- digital cameras, 498
- directories
 - changing, 351
 - creating, 351
 - deleting, 351
 - paths, 339
 - structure, 337
- disks
 - boot, 149, 402
 - required space, 40
 - rescue, 149
- DNS, 593
 - configuring, 161
 - domains, 617
 - mail exchanger, 593
 - name servers, 617
 - NIC, 593
 - security and, 758
 - top level domain, 593
- documentation (see help)
- DOS

- sharing files, 669
- drivers, 135
- drives
 - mounting, 356
 - unmounting, 356
- du, 357

E

- e-mail
 - configuring, 160
 - synchronizing, 496, 687
 - mailsync, 702–705
- editors
 - Emacs, 413–414
 - vi, 360
- Emacs, 413–414
 - .emacs, 413
 - default.el, 413
- encoding
 - ISO-8859-1, 417
- encrypting, 733–736
 - creating partitions, 734
 - files, 736–737
 - files with vi, 736
 - partitions, 734–736
 - removable media, 736
 - YaST, with, 734
- error messages
 - bad interpreter, 154
 - permission denied, 154
- Evolution, 500

F

- file, 354
- file systems, 453–463
 - ACLs, 299–310
 - cryptofs, 733
 - encrypting, 733
 - Ext2, 455–456

- Ext3, 456–457
- FAT, 43
- LFS, 461
- limitations, 461
- NTFS, 44–45
- Reiser4, 458
- ReiserFS, 454–455
- repairing, 809
- selecting, 454
- supported, 460–461
- terms, 453
- XFS, 458–459

files

- archiving, 344, 353
- comparing, 355
- compressing, 344, 353
- copying, 350
- deleting, 351
- encrypting, 736
- finding, 410
- moving, 350
- paths, 339
- searching contents, 355
- searching for, 354
- synchronizing, 685–705
 - CVS, 686, 694–696
 - mailsync, 687, 702–705
 - rsync, 687
 - Subversion, 687
 - Unison, 686, 692–694
- uncompressing, 345
- viewing, 343, 354

find, 354

Firefox

- URL open command, 205

firewalls, 167, 709

- packet filters, 709, 712
- SuSEfirewall2, 709, 713

Firewire (IEEE1394)

- hard disks, 498

- flash drives, 498
- fonts, 473
 - CID-keyed, 478
 - TrueType, 473
 - X11 core, 473
 - Xft, 474
- free, 357

G

GNOME

- shell, 334

graphics

- 3D, 478–481
 - 3Ddiag, 480
 - diagnosis, 479
 - drivers, 478
 - installation support for, 480
 - SaX, 479
 - support for, 478
 - testing, 480
 - troubleshooting, 480
- cards
 - 3D, 478–481
 - drivers, 471
- GLIDE, 478–481
- OpenGL, 478–481
 - drivers, 478
 - testing, 480

grep, 355

groups

- managing, 165

GRUB, 387–406

- boot menu, 390
- boot password, 396
- boot sectors, 388
- booting, 388
- commands, 388–397
- device names, 391
- device.map, 389, 394

- GRUB Geom Error, 404
- grub.conf, 389, 395
- JFS and GRUB, 404
- limitations, 388
- Master Boot Record (MBR), 387
- menu editor, 393
- menu.lst, 389–390
- partition names, 391
- troubleshooting, 404
- uninstalling, 402
- gunzip, 345
- gzip, 345, 353

H

- halt, 360
- hard disks
 - DMA, 142
- hardware
 - Bluetooth, 141
 - graphics cards, 141, 176
 - hard disk controllers, 142
 - information, 142
 - infrared, 141
 - ISDN, 604
 - monitor, 141, 176
- hciconfig, 565
- hcitool, 565
- help, 765–768
 - books, 771
 - FAQs, 770
 - guides, 771
 - HOWTOs, 770
 - info pages, 413, 770
 - Linux documentation (TLDP), 770
 - man pages, 349, 413, 769
 - manuals, 771
 - package documentation, 772
 - specifications, 773
 - standards, 773

- SUSE books, 771
- SUSE Help Center, 765
- Usenet, 773
- Wikipedia, 771
- X, 472
- hostnames, 161

I

- I18N, 415
- inetd, 162
- info pages, 413
- init, 375
 - adding scripts, 380
 - inittab, 375
 - scripts, 378–382
- installation support
 - 3D graphics cards and, 480
- installing
 - directory, into, 139
 - GRUB, 388
 - manually, 201
 - Xen, 139
 - YaST, with, 33–58
- internationalization, 415
- Internet
 - cinternet, 629
 - dial-up, 628–630
 - DSL, 607
 - ISDN, 604
 - KInternet, 629
 - qinternet, 629
 - smpppd, 628–630
 - TDSL, 609
- IP addresses, 581
 - classes, 581
 - IPv6, 583
 - configuring, 591
 - masquerading, 711
 - private, 583

- IrDA, 497, 570–573
 - configuring, 571
 - starting, 571
 - stopping, 571
 - troubleshooting, 572

J

- joystick
 - configuring, 143

K

- KDE
 - configuring as administrator, 271–283
 - KIOSK, 271–283
 - KIOSK Admin Tool, 271
 - profiles, 271
 - shell, 334
- Kerberos, 725–731
 - authenticators, 726
 - credentials, 726
 - principals, 726
 - session key, 726
 - ticket-granting service, 729
 - tickets, 726, 729
- kernels
 - caches, 412
 - limits, 462
- keyboard
 - Asian characters, 415
 - configuring, 143
 - layout, 414
 - mapping, 414
 - compose, 415
 - multikey, 415
 - X Keyboard Extension, 415
 - XKB, 415
- kill, 358
- killall, 358
- KIOSK, 271–283

- kiosktool, 271
- Kontakt, 500
- KPilot, 500
- KPowersave, 495
- KSysguard, 495

L

- L10N, 415
- languages, 138, 157
- laptops, 491–498
 - hardware, 491
 - IrDA, 570–573
 - NetworkManager, 494
 - PCMCIA, 491
 - power management, 491, 523–535
 - SCPM, 492, 509
 - SLP, 494
- LDAP, 653–668
 - administering groups, 666
 - administering users, 666
 - directory tree, 655
 - YaST
 - client, 658
 - modules, 660
 - templates, 660
- less, 343, 355
- LFS, 461
- license agreement, 38
- Lightweight Directory Access Protocol (see LDAP)
- Linux
 - networks and, 577
 - sharing files with another OS, 669
 - uninstalling, 402
- linuxrc
 - manual installation, 201
- ln, 351
- localization, 415
- locate, 354, 410

- log files, 166, 409
 - boot.msg, 168, 527
 - messages, 168, 717
 - Unison, 694
 - XFree86, 480
- logging
 - login attempts, 166
- Logical Volume Manager (see LVM)
- logrotate, 409
- ls, 335, 350
- LVM
 - YaST, 113

M

- man pages, 349, 413
- masquerading, 711
 - configuring with SuSEfirewall2, 713
- Master Boot Record (see MBR)
- MBR, 387–388
- memory
 - RAM, 412
- mkdir, 351
- mobility, 491–500
 - cellular phones, 499
 - data security, 498
 - digital cameras, 498
 - external hard disks, 498
 - Firewire (IEEE1394), 498
 - laptops, 491
 - PDA's, 499
 - USB, 498
- modems
 - cable, 607
 - YaST, 602
- more, 343
- mount, 356
- mouse
 - configuring, 144
- mv, 350

N

- NAT (see masquerading)
- NetBIOS, 669
- Network File System (see NFS)
- Network Information Service (see NIS)
- NetworkManager, 494
- networks, 577
 - authentication
 - Kerberos, 725–731
 - base network address, 582
 - Bluetooth, 497, 563
 - broadcast address, 583
 - configuration files, 615–622
 - configuring, 158–163, 594–609, 612–628
 - IPv6, 591
 - DNS, 593
 - IrDA, 497
 - localhost, 583
 - netmasks, 581
 - routing, 163, 581
 - SLP, 631
 - TCP/IP, 577
 - wireless, 497
 - WLAN, 497
 - YaST, 594
 - alias, 597
 - gateway, 598
 - hostname, 598
 - IP address, 596
 - starting, 599
- NFS, 681
 - clients, 161, 681
 - importing, 682
 - mounting, 682
- NIS, 641–642
 - clients, 161, 641
- notebooks (see laptops)
- nslookup, 359

NSS, 620
databases, 620

NTP
client, 161

O

opd, 567
OpenLDAP (see LDAP)
OpenSSH (see SSH)
OS/2
sharing files, 669

P

packet filters (see firewalls)
PAM, 483–490
configuring, 201
pand, 566
partitions
creating, 39, 150–151
encrypting, 734
fstab, 153
LVM, 152
parameters, 152
partition table, 387
RAID, 152
resizing Windows, 42
swap, 152
types, 40
passwd, 359
passwords
changing, 359
paths, 339
absolute, 339
relative, 339
PCI device
drivers, 154
PCMCIA, 491, 501
IrDA, 570–573
PDAs, 499

permissions, 346
ACLs, 299–310
changing, 348, 352
directories, 347
file permissions, 410
file systems, 346
files, 346
viewing, 347
ping, 358, 624
Pluggable Authentication Modules (see PAM)
PostgreSQL
updating, 186
power management, 491, 523–543
ACPI, 523, 526–533, 538
APM, 523, 525–526, 538
battery monitor, 524
charge level, 539
cpufrequency, 535
cpuspeed, 535
hibernation, 524
powersave, 535
standby, 524
suspend, 524
YaST, 543
powersave, 535
configuring, 535
printing, 421, 424–427
applications, from, 431
command line, 431
configuring with YaST, 424
connection, 425
CUPS, 431
drivers, 426
GDI printers, 437
Ghostscript driver, 426
IrDA, 571
kprinter, 431
network, 438
port, 425

- PPD file, 426
- queues, 426
- Samba, 670
- test page, 426
- troubleshooting
 - network, 438
- xpp, 431
- private branch exchange, 605
- processes, 358
 - killing, 358
 - overview, 358
- protocols
 - CIFS, 669
 - IPv6, 583
 - LDAP, 653
 - SLP, 631
 - SMB, 669
- proxies, 162
- ps, 358

R

- RAID
 - YaST, 119
- reboot, 360
- registering
 - command line, 140
 - Internet, without, 140
 - YaST, 140
- release notes, 168
- repairing systems, 803
- rescue system, 807
 - starting from CD, 808
 - starting from network source, 808
- RFCs, 577
- rm, 351
- rmdir, 351
- routing, 163, 581, 616–617
 - masquerading, 711
 - netmasks, 581

- routes, 616
 - static, 616
- RPM
 - security, 760
- rsync, 687, 700
- runlevels, 156, 375–378
 - changing, 377–378
 - editing in YaST, 382

S

- Samba, 669–679
 - CIFS, 669
 - clients, 163, 670, 677–678
 - configuring, 671–677
 - installing, 671
 - login, 678
 - names, 669
 - permissions, 676
 - printers, 670
 - printing, 678
 - security, 676–677
 - server, 670
 - servers, 163, 671–677
 - shares, 670, 674
 - SMB, 669
 - starting, 671
 - stopping, 671
 - swat, 672
 - TCP/IP and, 669
- SaX2
 - display device, 178
 - display settings, 176
 - dual head, 178
 - graphics card, 177
 - graphics tablet, 181
 - keyboard settings, 181
 - mouse settings, 180
 - multihead, 179
 - remote access (VNC), 182

- resolution and color depth, 178
- touchscreen, 182
- scanning
 - configuring, 144
 - troubleshooting, 145
- SCPM, 156, 509
 - advanced settings, 520
 - laptops, 492
 - managing profiles, 518
 - resource groups, 518
 - starting, 518
 - switching profiles, 519
- screen
 - resolution, 471
- scripts
 - init.d, 375, 378–382, 627
 - boot, 379
 - boot.local, 380
 - boot.setup, 380
 - halt, 380
 - network, 627
 - nfsserver, 628
 - portmap, 627
 - rc, 377–378, 380
 - sendmail, 628
 - xinetd, 627
 - ypbind, 628
 - ypserv, 628
 - irda, 571
 - mkinitrd, 373
 - modify_resolvconf, 412, 617
 - SuSEconfig, 383–386
 - disabling, 386
- sdptool, 566
- security, 749–761
 - attacks, 757–758
 - booting, 750, 752
 - bugs and, 753, 756
 - configuring, 163–167
 - DNS, 758

- encrypted file system, 498
- engineering, 750
- firewalls, 167, 709
- intrusion detection, 201
- local, 751–754
- network, 755–758
- passwords, 751–752
- permissions, 752–753
- reporting problems, 761
- RPM signatures, 760
- Samba, 676
- serial terminals, 750
- SSH, 719–724
- tcpd, 761
- telnet, 719
- tips and tricks, 758
- viruses, 754
- worms, 758
- X and, 755
- Service Location Protocol (see SLP)
- shells, 333–364
 - Bash, 334
 - commands, 349–360
 - pipes, 343
 - wild cards, 342
- SLP, 494, 631
 - browser, 633
 - Konqueror, 633
 - registering services, 631
 - slptool, 632
- SMB (see Samba)
- smpd, 669
- soft RAID (see RAID)
- software
 - development
 - SUSE SDK, 135
 - drivers, 135
 - installing, 127–134
 - removing, 127–134
- sound

- configuring in YaST, 147
- fonts, 148
- MIDI, 148
- mixers, 200
- SSH, 719–724
 - authentication mechanisms, 722
 - daemon, 721
 - key pairs, 721, 723
 - scp, 720
 - sftp, 721
 - ssh, 720
 - ssh-agent, 723–724
 - ssh-keygen, 723
 - sshd, 721
 - X and, 723
- su, 360
- Subversion, 687, 697
- SUSE books, 771
- SUSE SDK, 135
- SVN (Subversion), 687
- synchronizing data, 497
 - e-mail, 496
 - Evolution, 500
 - Kontakt, 500
 - KPilot, 500
- system
 - configuring, 125–170
 - languages, 157
 - limiting resource use, 411
 - localizing, 415
 - rebooting, 360
 - rescuing, 807
 - security, 165
 - services, 162
 - shutdown, 360
 - updating, 137
- system monitoring, 495
 - KPowersave, 495
 - KSysguard, 495

T

- tar, 344, 353
- TCP/IP, 577
 - ICMP, 578
 - IGMP, 578
 - layer model, 578
 - packets, 579–580
 - TCP, 577
 - UDP, 578
- telnet, 359
- time zones, 157
- TLDP, 770
- top, 358
- Tripwire
 - replaced by AIDE, 201
- TV
 - card configuration, 146

U

- ulimit, 411
 - options, 411
- umount, 356
- uninstalling
 - GRUB, 402
 - Linux, 402
- updatedb, 354
- updating
 - online, 136–137
 - command line, 173
 - passwd and group, 186
 - patch CD, 137
 - problems, 186
 - sound mixers, 200
 - YaST, 186
- USB
 - flash drives, 498
 - hard disks, 498
- users
 - /etc/passwd, 486, 659

managing, 164

V

variables

environment, 416

virtual memory, 152

VNC

administration, 162

W

whois, 594

wild cards, 354

Windows

sharing files, 669

wireless connections

Bluetooth, 559

WLAN, 497

X

X

character sets, 473

CID-keyed fonts, 478

display device, 178

display settings, 176

drivers, 471

dual head, 178

font systems, 473

fonts, 473

graphics card, 177

graphics tablet, 181

help, 472

keyboard settings, 181

mouse settings, 180

multihead, 179

optimizing, 467–473

remote access (VNC), 182

resolution and color depth, 178

SaX2, 467

security, 755

SSH and, 723

touchscreen, 182

TrueType fonts, 473

virtual screen, 471

X11 core fonts, 473

xf86config, 467

xft, 473

Xft, 474

X Keyboard Extension (see keyboard, XKB)

X Window System (see X)

X.Org, 467

Xen

installation into directory, 139

Xft, 474

xinetd, 162

XKB (see keyboard, XKB)

xorg.conf

color depth, 470

Depth, 470

Device, 470

Display, 470

Files, 468

InputDevice, 468

Modeline, 470

modelines, 468

Modes, 468, 470

Monitor, 468, 470

ServerFlags, 468

Y

YaST

3D, 479

add-on, 134

autoinstallation, 167

profiles, 167

AutoYaST, 167

backups, 138, 148

Bluetooth, 141

- boot configuration, 397
 - default system, 400
 - security, 401
 - time-out, 400
- boot loader
 - disk order, 401
 - location, 399
 - password, 401
 - type, 398
- boot mode, 49
- cable modem, 607
- configuring, 125–170
- Control Center, 126
- disk creation, 149
- disk space, 40
- DMA, 142
- DNS, 161
- driver CDs, 170
- DSL, 607
- e-mail, 160
- firewall, 167
- graphics cards, 141, 176
- group management, 165
- GRUB, 398
- hard disk controllers, 142
- hardware, 140–148
 - detection, 48
 - information, 142
- hostname, 50, 161
- infrared, 141
- installation into directory, 139
- installation mode, 38
- installation scope, 46
- installation settings, 39
- installation sources, 135
- installation summary, 39
- installing with, 33–58
- ISDN, 604
- joystick, 143
- Kerberos client, 161
- keyboard, 48, 143
- languages, 37, 126, 138, 157
- LDAP, 161
 - clients, 55, 658
- LILO, 398
- LVM, 113, 150
- media check, 140
- modems, 602
- monitor, 141, 176
- ncurses, 170
- network card, 594
- network configuration, 51, 158–163
- NFS clients, 161
- NIS clients, 56, 641
- Novell AppArmor, 163
- NTP client, 161
- online update, 136–137
- package dependencies, 47
- partitioning, 39, 150
- PCI device drivers, 154
- power management, 155, 543
- powertweak, 156
- printing, 424–427
- profile manager, 156
- radio cards, 146
- RAID, 119
- registering, 140
- release notes, 168
- repairing systems, 803
- root password, 50
- routing, 163
- runlevels, 382
- safe settings, 36
- Samba
 - clients, 56, 163, 677
- scanner, 144
- SCPM, 156
- security, 163–167
- sendmail, 160
- SLP browser, 633

- software, 127–138
- software updates, 53
- sound cards, 147
- starting, 33, 125
- support query, 168
- sysconfig editor, 156, 384
- system analysis, 38
- system security, 165
- system start-up, 33
- T-DSL, 609
- text mode, 170–176
 - modules, 173
- time zone, 38, 157
- TV cards, 146
- updating, 137, 186
- user management, 164
- Xen, 139
- YP (see NIS)

