Start-Up

openSUSE Leap 15.1

Start-Up

openSUSE Leap 15.1

Publication Date: May 27, 2019

SUSE LLC
10 Canal Park Drive
Suite 200
Cambridge MA 02141
USA
https://www.suse.com/documentation

Contents

About This Guide

This manual will see you through your initial contact with openSUSE® Leap. Check out the various parts of this manual to learn how to install, use and enjoy your system.

Installation

Guides you through the installation process and the basic configuration of your system. The Quick Start section shows a quick walk through the installation using default values. The second part of this chapter provides details for every installation step.

Administration

Introduces YaST, the central tool for installation and configuration of your system. Learn how to initially set up your system and how to modify key components of your system.

Managing and Updating Software

Understand how to install or remove software with either YaST or using the command line, how to use the 1-Click Install feature, and how to keep your system up-to-date.

The Bash Shell

Learn how to work with the bash shell, the default command line interpreter on openSUSE Leap. Get to know the most commonly used Linux commands and understand basic concepts of a Linux system.

Help and Troubleshooting

Provides an overview of where to find help and additional documentation in case you need more information or want to perform specific tasks with your system. Also find a compilation of the most frequent problems and annoyances and learn how to solve these problems on your own.

1 Available Documentation

Note: Online Documentation and Latest Updates

Documentation for our products is available at http://doc.opensuse.org/, where you can also find the latest updates, and browse or download the documentation in various formats.

In addition, the product documentation is usually available in your installed system under `/usr/share/doc/manual`.

The following documentation is available for this product:

Start-Up

This manual will see you through your initial contact with openSUSE® Leap. Check out the various parts of this manual to learn how to install, use and enjoy your system.

***Book* "Reference"**

Covers system administration tasks like maintaining, monitoring and customizing an initially installed system.

***Book* "Virtualization Guide"**

Describes virtualization technology in general, and introduces libvirt—the unified interface to virtualization—and detailed information on specific hypervisors.

***Book* "AutoYaST Guide"**

AutoYaST is a system for unattended mass deployment of openSUSE Leap systems using an AutoYaST profile containing installation and configuration data. The manual guides you through the basic steps of auto-installation: preparation, installation, and configuration.

***Book* "Security Guide"**

Introduces basic concepts of system security, covering both local and network security aspects. Shows how to use the product inherent security software like AppArmor or the auditing system that reliably collects information about any security-relevant events.

***Book* "System Analysis and Tuning Guide"**

An administrator's guide for problem detection, resolution and optimization. Find how to inspect and optimize your system by means of monitoring tools and how to efficiently manage resources. Also contains an overview of common problems and solutions and of additional help and documentation resources.

***Book* "GNOME User Guide"**

Introduces the GNOME desktop of openSUSE Leap. It guides you through using and configuring the desktop and helps you perform key tasks. It is intended mainly for end users who want to make efficient use of GNOME as their default desktop.

2 Feedback

Several feedback channels are available:

Bug Reports

To report bugs for openSUSE Leap, go to https://bugzilla.opensuse.org/, log in, and click *New*.

Mail

For feedback on the documentation of this product, you can also send a mail to `doc-team@suse.com`. Make sure to include the document title, the product version and the publication date of the documentation. To report errors or suggest enhancements, provide a concise description of the problem and refer to the respective section number and page (or URL).

3 Documentation Conventions

The following notices and typographical conventions are used in this documentation:

- `/etc/passwd`: directory names and file names
- *`PLACEHOLDER`*: replace *`PLACEHOLDER`* with the actual value
- `PATH`: the environment variable PATH
- **`ls`**, `--help`: commands, options, and parameters
- `user`: users or groups
- `package name` : name of a package
- 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*, ↑Another Manual): This is a reference to a chapter in another manual.
- Commands that must be run with `root` privileges. Often you can also prefix these commands with the **`sudo`** command to run them as non-privileged user.

```
root # command
tux > sudo command
```

- Commands that can be run by non-privileged users.

```
tux > command
```

- Notices

Warning: Warning Notice

Vital information you must be aware of before proceeding. Warns you about security issues, potential loss of data, damage to hardware, or physical hazards.

Important: Important Notice

Important information you should be aware of before proceeding.

Note: Note Notice

Additional information, for example about differences in software versions.

Tip: Tip Notice

Helpful information, like a guideline or a piece of practical advice.

4 About the Making of This Documentation

This documentation is written in GeekoDoc (https://github.com/openSUSE/geekodoc), a subset of DocBook 5 (http://www.docbook.org). The XML source files were validated by **jing** (see https://code.google.com/p/jing-trang/), processed by **xsltproc**, and converted into XSL-FO using a customized version of Norman Walsh's stylesheets. The final PDF is formatted through FOP from Apache Software Foundation (https://xmlgraphics.apache.org/fop). The open source tools and the environment used to build this documentation are provided by the DocBook Authoring and Publishing Suite (DAPS). The project's home page can be found at https://github.com/openSUSE/daps.

The XML source code of this documentation can be found at https://github.com/SUSE/doc-sle.

5 Source Code

The source code of openSUSE Leap is publicly available. Refer to http://en.opensuse.org/Source_code for download links and more information.

6 Acknowledgments

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

I Installation

1 *Installation Quick Start*

Use the following procedures to install a new version of openSUSE® Leap 15.1. This document gives a quick overview on how to run through a default installation of openSUSE Leap on the x86_64 architecture.

1.1 Welcome to openSUSE Leap

For more detailed installation instructions see *Chapter 3, Installation Steps*.

1.1.1 Minimum System Requirements

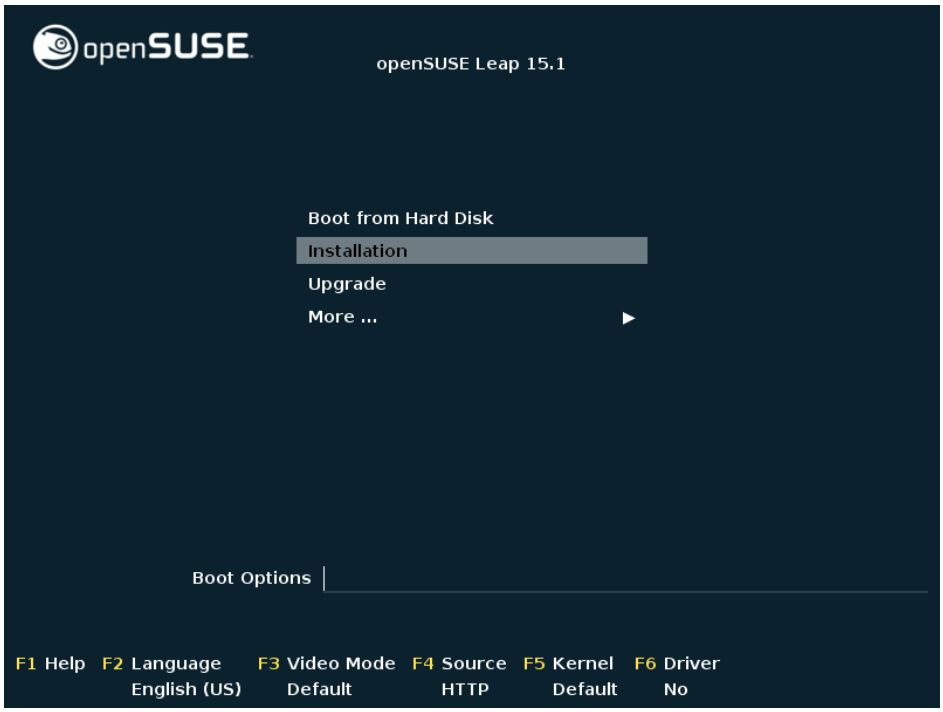
- any AMD64/Intel* EM64T processor (32-bit processors are not supported)
- 1 GB physical RAM (4 GB or more strongly recommended)
- 10 GB available disk space for a minimal installation, 16 GB for a graphical desktop (more is recommended). In case you plan to use Btrfs snapshots a minimum of 40 GB for the root partition is recommended.
- Supports most modern sound and graphics cards, 1024 x 768 display resolution (higher recommended)

1.1.2 Installing openSUSE Leap

Use these instructions if there is no existing Linux system on your machine, or if you want to replace an existing Linux system.

1.1.2.1 Booting the Installation System

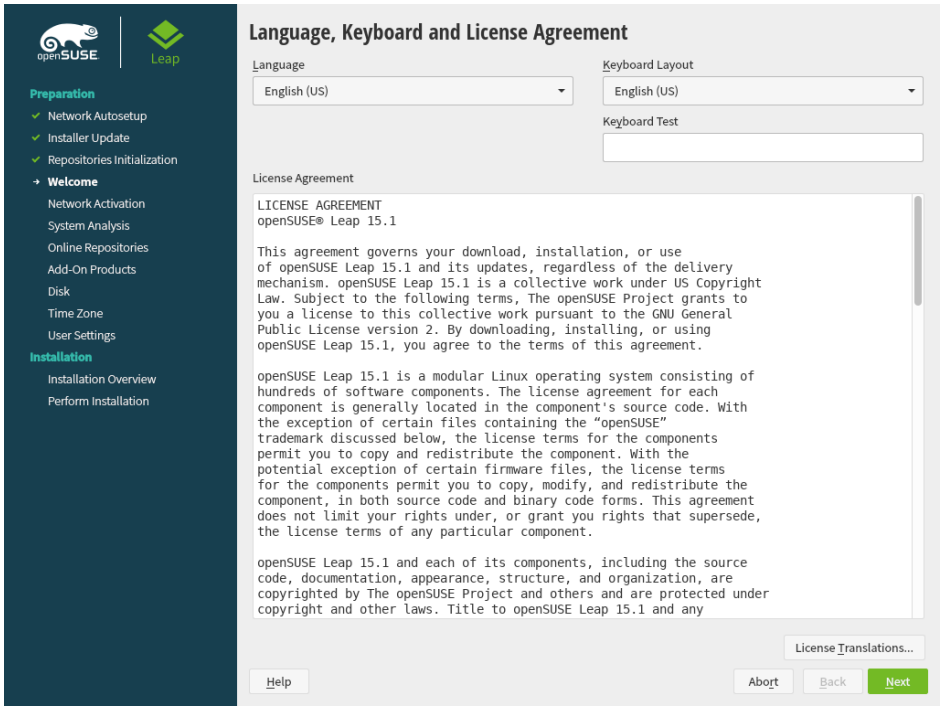
Insert a DVD or a bootable USB stick containing the installation image for openSUSE Leap, then reboot the computer to start the installation program. On machines with a traditional BIOS you will see the graphical boot screen shown below. On machines equipped with UEFI, a slightly different boot screen is used. Secure boot on UEFI machines is supported.



On BIOS machines, use F2 to change the language for the installer. A corresponding keyboard layout is chosen automatically. See *Section 2.2.1, "The Boot Screen on Machines Equipped with Traditional BIOS"* or *Section 2.2.2, "The Boot Screen on Machines Equipped with UEFI"* for more information about changing boot parameters. On UEFI machines adjust the language and keyboard settings in the next step.

Select *Installation* on the boot screen, then press Enter. This boots the system and loads the openSUSE Leap installer.

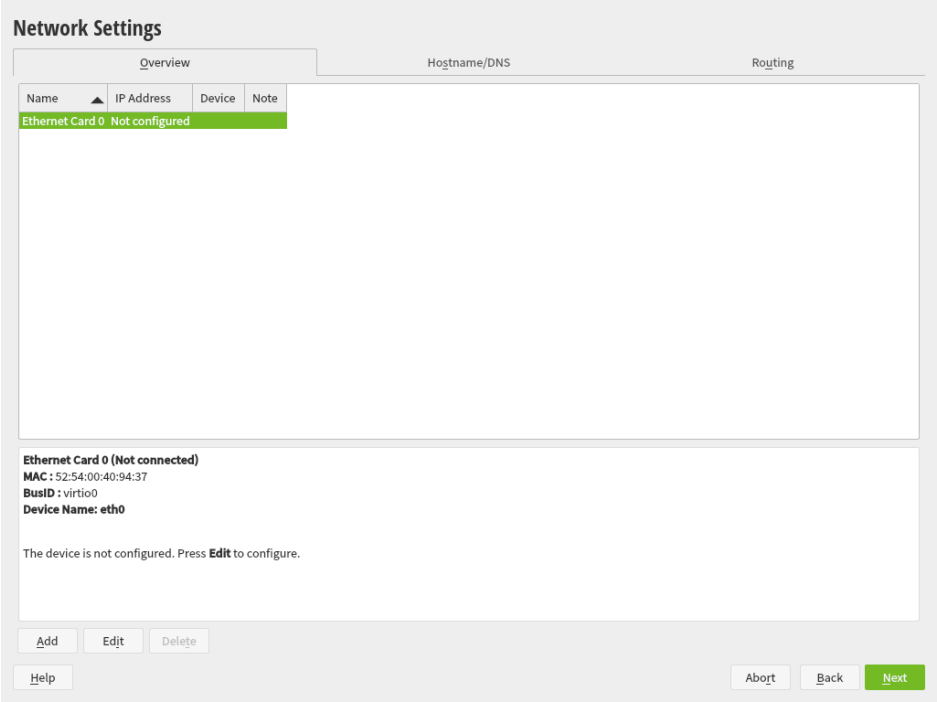
1.1.2.2 Language, Keyboard and License Agreement



On systems with a traditional BIOS the *Language* and *Keyboard Layout* settings are initialized with the language you chose at the boot screen. If you did not change the default, or are using a UEFI machine it will be English (US). Change the settings here, if necessary. Use the *Keyboard Test* text box to test the layout.

Read the License Agreement. It is presented in the language you have chosen. Other *License Translations* are available. Proceed with *Next*.

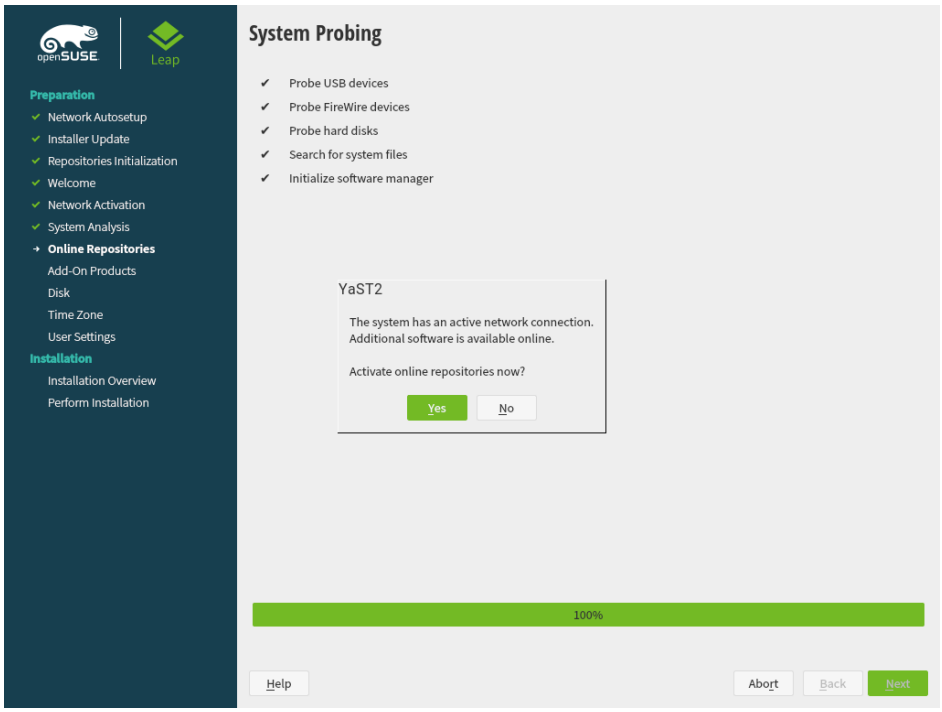
1.1.2.3 Network Settings



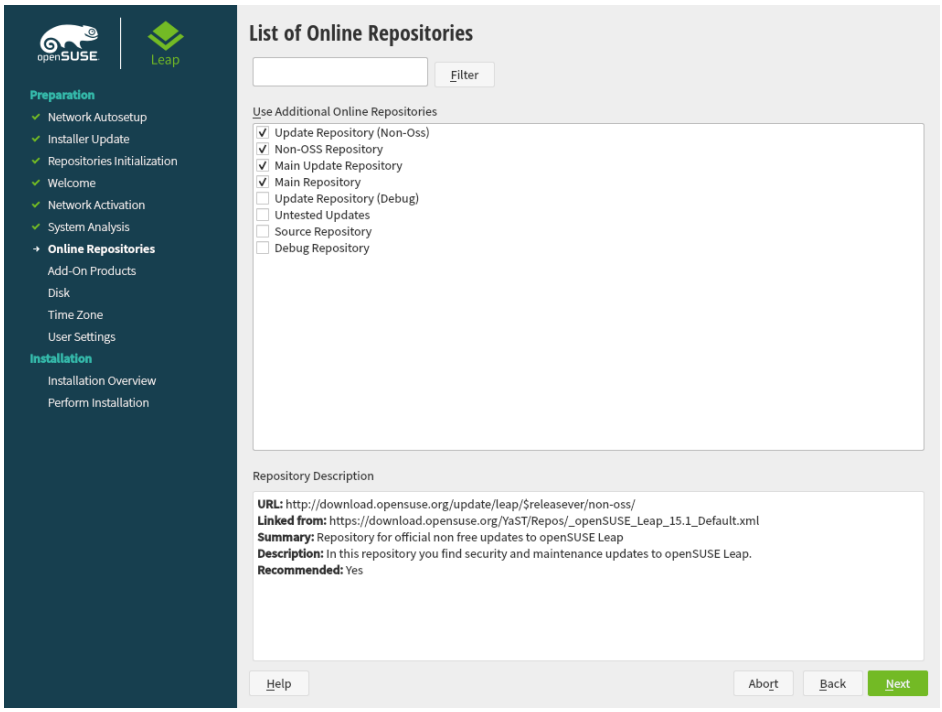
If the network can not be configured automatically, the *Network Settings* dialog opens. Choose a network interface from the list and configure it with *Edit*. Alternatively, *Add* an interface manually. See *Section 3.4, "Network Settings"* and *Book "*Reference*", Chapter 13 "Basic Networking", Section 13.4 "Configuring a Network Connection with YaST"* for more information. If you prefer to do an installation without network access, skip this step without making any changes and proceed with *Next*.

1.1.2.4 Online Repositories

A system analysis is performed, where the installer probes for storage devices, and tries to find other installed systems. If a network connection with Internet access is available, you will be asked to activate the online repositories. Answer with *Yes* to proceed. In case you do not have Internet access, this step will be skipped.



The online repositories are official openSUSE package sources. They not only offer additional packages not included on the installation media, but also the update repositories containing security and bug fixes. Using the default selection is recommended. Add at least the *Main Update Repository*, because it makes sure the system is installed with the latest security patches.

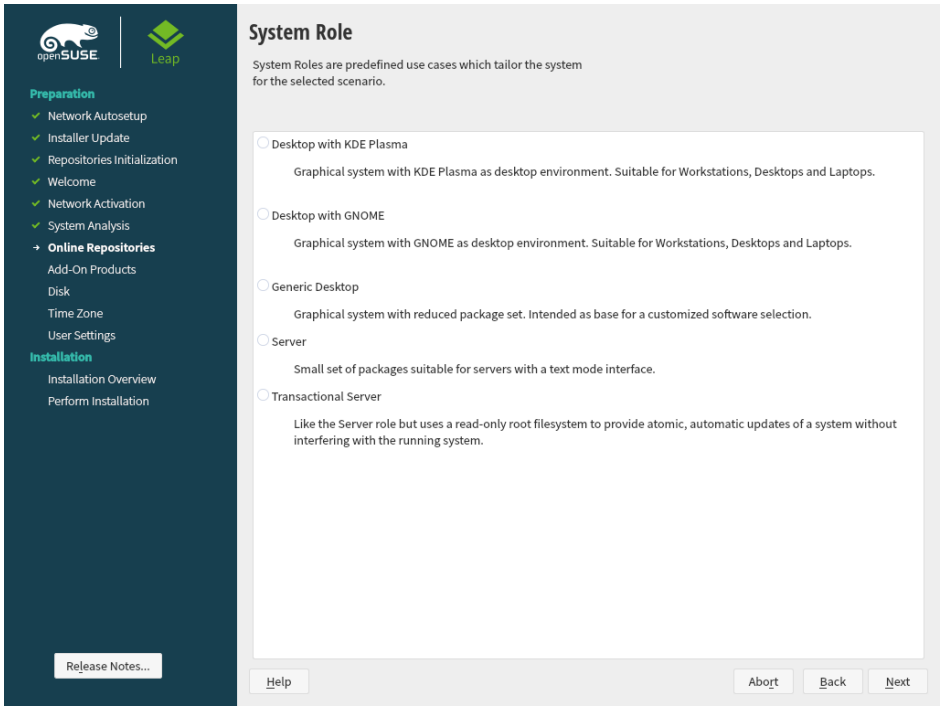


You have the following choices:

- The *Main Repository (OSS)* contains open source software (OSS). Compared to the DVD installation media, it contains many additional software packages, among them many additional desktop systems.
- The *Main Update Repository* contains security updates and fixes for packages from the *Main Repository (OSS)* and the DVD installation media. Choosing this repository is recommended for all installation scenarios.
- The *Main Repository (Non-OSS)* contains packages with a proprietary software license. Choosing it is not required for installing a custom desktop system.
- Choosing *Main Update Repository (Non-OSS)* is recommended when also having chosen the *Main Repository (Non-OSS)*. It contains the respective updates and security fixes.
- All other repositories are intended for experienced users and developers. Click on a repository name to get more information.

Confirm your selection with *Next*. Depending on your choice, you need to confirm one or more license agreements. Do so by choosing *Next* until you proceed to the *System Role* screen. Now choose *Next* to proceed.

1.1.2.5 System Role



Choose a general software and system configuration with this step by selecting a desktop or server configuration.

For a desktop installation, choose between *Desktop with KDE Plasma*, *Desktop with GNOME* and *Generic Desktop*. KDE is slightly similar to Windows, GNOME offers an alternative, innovative environment. In case you prefer an alterntive to the KDE or GNOME desktops, choose *Generic Desktop*. You will be able to choose between the XFCE, LXDE, MATE and others later in the installation process by selecting *Software* in the *Installation Settings dialog*.

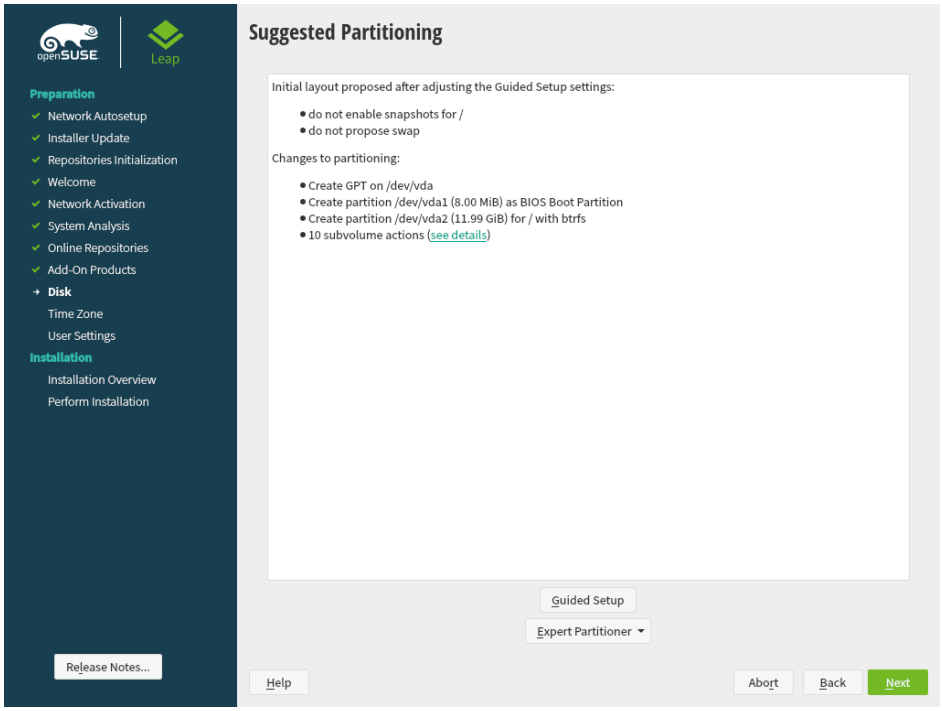
If setting up a server, you probably do not need a graphical user interface. Choose *Server (Text Mode)* in this case. Alternatively, set up a server system with a read-only root partition and transactional updates by choosing *Transactional Server*. This selection also is a prerequisite for setting up openSUSE Kubic. See https://kubic.opensuse.org/blog/2018-04-04-transactionalupdates/ for more information on transactional updates.

You can also manually choose the software configuration for your system. Select *Custom* and then *Next* to get to the *Software Selection and System Tasks* dialog. Choose one or more patterns for installation. By clicking *Details*, you can select individual packages.

Tip: Release Notes

From this point on, the Release Notes can be viewed from any screen during the installation process by selecting *Release Notes*.

1.1.2.6 Suggested Partitioning



Define a partition setup for openSUSE Leap in this step. Review the partition setup proposed by the system. If necessary, change it. You have the following options:

Guided Setup

Starts a wizard which lets you refine the partitioning proposal. Options available here depend on your system setup. In case it contains more than a single hard disk, you may choose which disk(s) to use and where to place the root partition. If the disk(s) already contain partitions, decide whether to remove or resize them.

In subsequent steps you may also add LVM support and disk encryption. You can change the file system for the root partition and decide whether to have a separate home partition or not.

Expert Partitioner

Opens the *Expert Partitioner* described in *Book "Reference", Chapter 5 "Expert Partitioner", Section 5.1 "Using the Expert Partitioner"*. This gives you full control over the partitioning setup and lets you create a custom setup. This option is intended for experts.

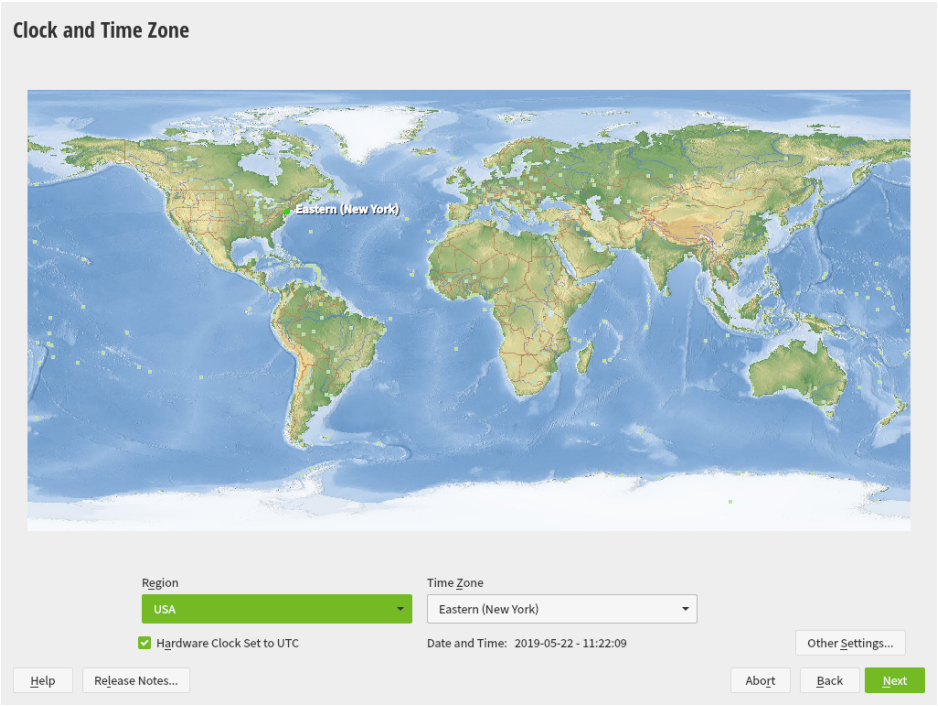
Note: Separate Home Partition

The default proposal no longer suggests to create a separate partition for `/home`. The `/home` directory contains the user's data and personal configuration files. Placing it on a separate directory makes it easier to rebuild the system in the future, or allows to share it with different Linux installations on the same machine.

In case you want to change the proposal to create a separate partition for `/home`, choose *Guided Setup* and click *Next* until you reach the *Filesystem Options* screen. Check *Propose Separate Home Partition*. By default it will be formatted with *XFS*, but you can choose to use a different file system. Close the dialog by clicking *Next* again.

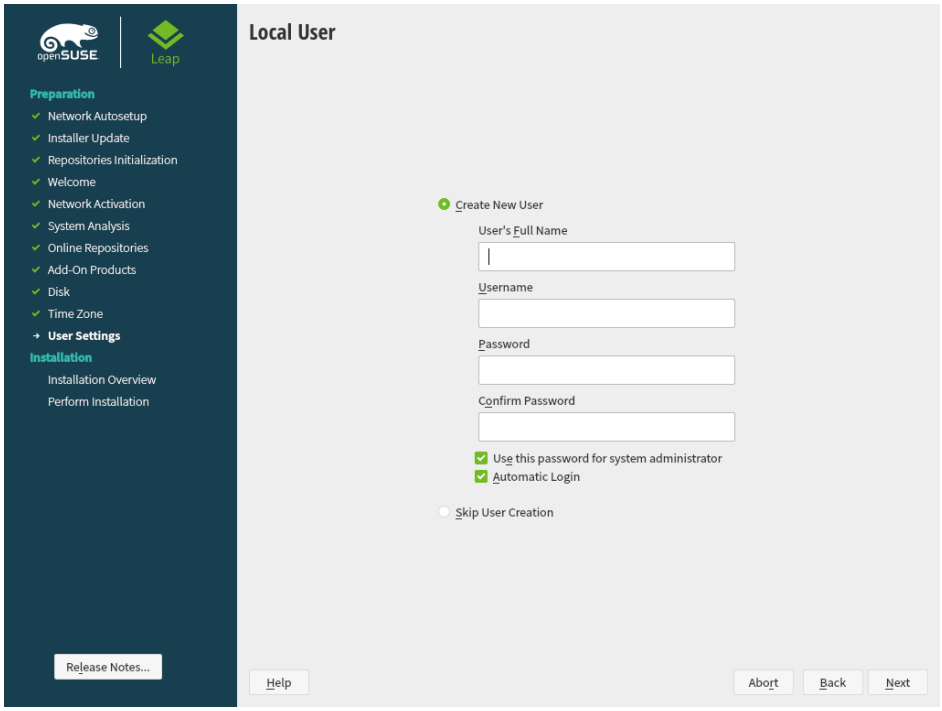
To accept the proposed setup without any changes, choose *Next* to proceed.

1.1.2.7 Clock and Time Zone



Select the clock and time zone to use in your system. To manually adjust the time or to configure an NTP server for time synchronization, choose *Other Settings*. See *Section 3.8, "Clock and Time Zone"* for detailed information. Proceed with *Next*.

1.1.2.8 Local User



To create a local user, type the first and last name in the *User's Full Name* field, the login name in the *Username* field, and the password in the *Password* field.

The password should be at least eight characters long and should contain both uppercase and lowercase letters and numbers. The maximum length for passwords is 72 characters, and passwords are case-sensitive.

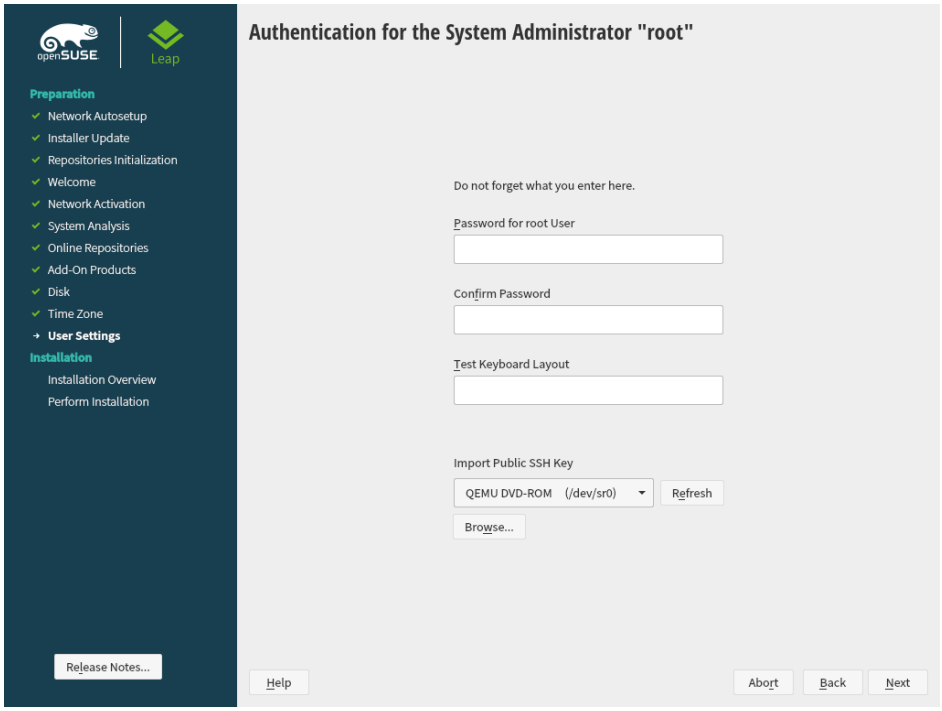
For security reasons it is also strongly recommended *not* to enable the *Automatic Login*. You should also *not Use this Password for the System Administrator* but rather provide a separate `root` password in the next installation step.

If you install on a system where a previous Linux installation was found, you may *Import User Data from a Previous Installation*. Click *Choose User* for a list of available user accounts. Select one or more user.

In an environment where users are centrally managed (for example by NIS or LDAP) you may want to skip the creation of local users. Select *Skip User Creation* in this case.

Proceed with *Next*.

1.1.2.9 Authentication for the System Administrator “root”



Provide a password for the system administrator account (called the `root` user).

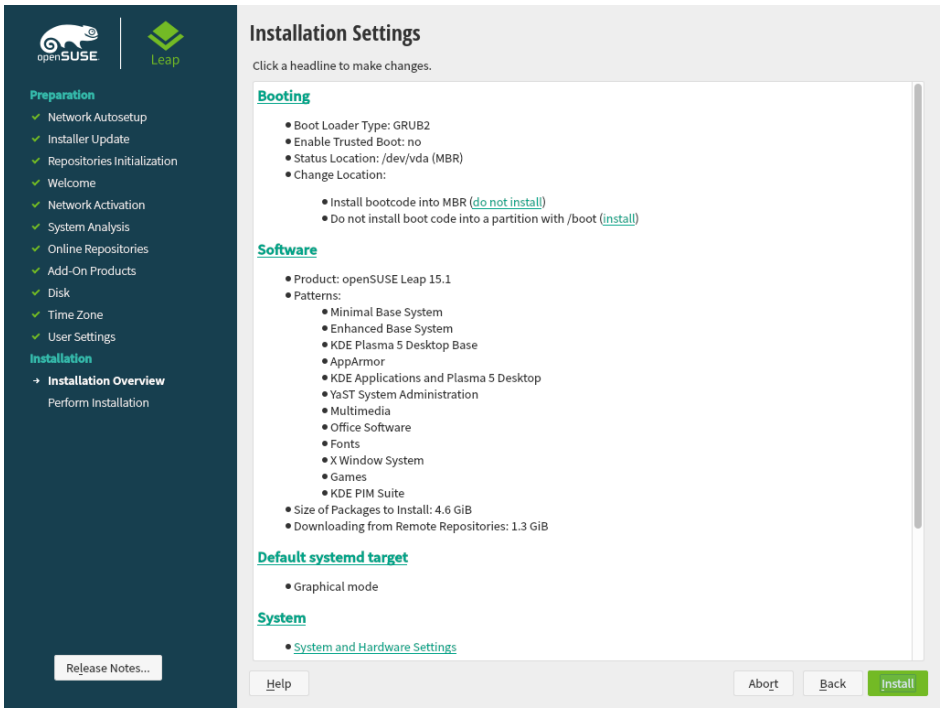
You should never forget the `root` password! After you entered it here, the password cannot be retrieved. See *Section 3.10, “Authentication for the System Administrator “root”” for more information.*
Proceed with *Next*.

Tip: Passwords and Keyboard Layout

It is recommended to only use characters that are available on an English keyboard. In case of a system error or when you need to start your system in rescue mode a localized keyboard might not be available.

In case you would like to enable password-less authentication via SSH login, you can import a key via *Import Public SSH Key*. If you want to completely disable `root` login via password, upload a key only and do not provide a root password. A login as system administrator will only be possible via SSH using the respective keyin this case.

1.1.2.10 Installation Settings



Use the *Installation Settings* screen to review and—if necessary—change several proposed installation settings. The current configuration is listed for each setting. To change it, click the headline. Some settings, such as firewall or SSH can directly be changed by clicking the respective links.

Tip: Remote System Access

Changes you can make in the *Installation Settings*, can also be made later at any time from the installed system. However, if you need remote access directly after the installation, you should adjust the *Firewall and SSH* settings by opening the SSH port and enabling the SSH server.

Booting

This section shows the boot loader configuration. Changing the defaults is only recommended if really needed. Refer to *Book "*Reference*", Chapter 12 "The Boot Loader GRUB 2"* for details.

Software

The default scope of software includes the base system and X Window with the selected desktop. Clicking *Software* opens the *Software Selection and System Tasks* screen, where you can change the software selection by selecting or deselecting patterns. Each pattern contains several software packages needed for specific functions (for example, Web and LAMP server or a print server). For a more detailed selection based on software packages to install, select *Details* to switch to the YaST *Software Manager*. See *Chapter 10, Installing or Removing Software* for more information.

Default Systemd Target

If you have chosen to install a desktop system, the system boots into the *graphical* target, with network, multiuser and display manager support. If you have not installed a desktop, the system boots into a login shell (*Text Mode*).

System

View detailed hardware information by clicking *System*. In the resulting screen you can also change *Kernel Settings*—see *Section 3.11.6, "System"* for more information.

Security

The *CPU Mitigations* refer to kernel boot command line parameters for software mitigations that have been deployed to prevent CPU side-channel attacks. Click the highlighted entry to choose a different option. For details, see *Book "*Reference*", Chapter 12 "The Boot Loader GRUB 2" CPU Mitigations*.

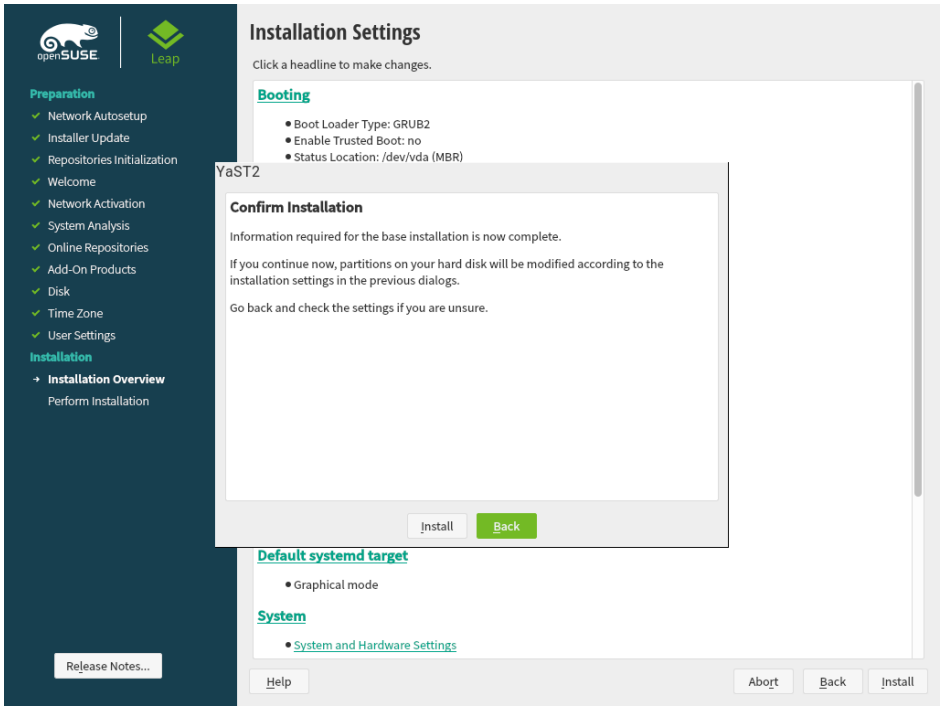
By default, the Firewall is enabled with all network interfaces configured for the public zone. See *Book "*Security Guide*", Chapter 16 "Masquerading and Firewalls", Section 16.4 "*`firewalld`*"* for configuration details.

The SSH service is disabled by default, its port (22) is closed. Therefore logging in from remote is not possible by default. Click *enable* and *open* to toggle these settings.

Network Configuration

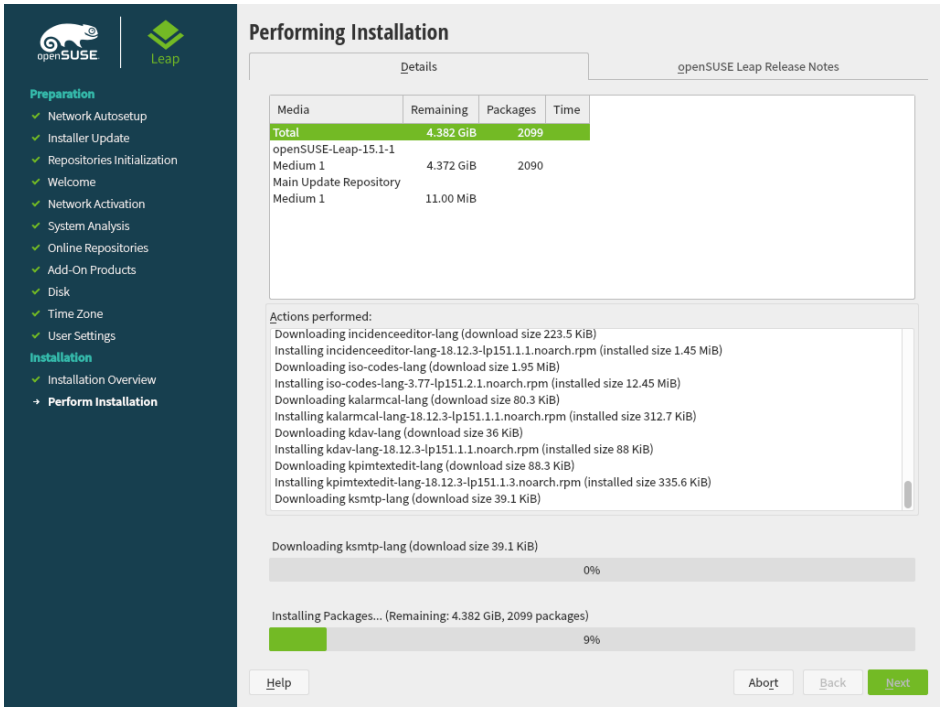
Displays the current network configuration. Click *Network Configuration* to change the settings. For details, see *Book "*Reference*", Chapter 13 "Basic Networking", Section 13.4 "Configuring a Network Connection with YaST"*.

1.1.2.11 Start the Installation



After you have finalized the system configuration on the *Installation Settings* screen, click *Install*. Depending on your software selection you may need to agree to license agreements before the installation confirmation screen pops up. Up to this point no changes have been made to your system. After you click *Install* a second time, the installation process starts.

1.1.2.12 The Installation Process



During the installation, the progress is shown in detail on the *Details* tab. The *openSUSE Leap Release Notes* tab shows important information; reading them is recommended.

After the installation routine has finished, the computer is rebooted into the installed system. Log in and start YaST to fine-tune the system. If you are not using a graphical desktop or are working from remote, refer to *Book "Reference", Chapter 1 "YaST in Text Mode"* for information on using YaST from a terminal.

2 Boot Parameters

openSUSE Leap allows setting several parameters during boot, for example choosing the source of the installation data or setting the network configuration.

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

The way the system is started for the installation depends on the architecture—system start-up is different for PC (AMD64/Intel 64) or mainframe, for example. If you install openSUSE Leap as a VM Guest on a KVM or Xen hypervisor, follow the instructions for the AMD64/Intel 64 architecture.

Note: Boot Options and Boot Parameters

The terms *Boot Parameters* and *Boot Options* are often used interchangeably. In this documentation, we mostly use the term *Boot Parameters*.

2.1 Using the Default Boot Parameters

The boot parameters are described in detail in *Chapter 3, Installation Steps*. Generally, 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 *Chapter 4, Troubleshooting*.

The menu bar at the bottom of the screen offers some advanced functionality needed in some setups. Using the function keys (`F1` ... `F12`), you can specify additional options to pass to the installation routines without having to know the detailed syntax of these parameters (see *Chapter 2, Boot Parameters*). A detailed description of the available function keys is available in *Section 2.2.1, “The Boot Screen on Machines Equipped with Traditional BIOS”*.

2.2 PC (AMD64/Intel 64/ARM AArch64)

This section describes changing the boot parameters for AMD64, Intel 64, and ARM AArch64.

2.2.1 The Boot Screen on Machines Equipped with Traditional BIOS

The boot screen displays several options for the installation procedure. *Boot from Hard Disk* boots the installed system and is selected by default, because the CD is often left in the drive. Select one of the other options with the arrow keys and press Enter to boot it. The relevant options are:

Installation

The normal installation mode. All modern hardware functions are enabled. In case the installation fails, see F5 *Kernel* for boot parameters that disable potentially problematic functions.

Upgrade

Perform a system upgrade. For more information refer to *Chapter 13, Upgrading the System and System Changes*.

More > Rescue System

Starts a minimal Linux system without a graphical user interface. For more information, see *Section 17.5.2, "Using the Rescue System"*. This option is not available on live CDs.

More > Boot Linux System

Boot a Linux system that is already installed. You will be asked from which partition to boot the system.

More > Check Installation Media

This option is only available when you install from media created from downloaded ISOs. In this case it is recommended to check the integrity of the installation medium. This option starts the installation system before automatically checking the media. In case the check was successful, the normal installation routine starts. If a corrupt media is detected, the installation routine aborts. Replace the broken medium and restart the installation process.

More > Memory Test

Tests your system RAM using repeated read and write cycles. Terminate the test by rebooting. For more information, see *Section 4.4, "Boot Failure"*.

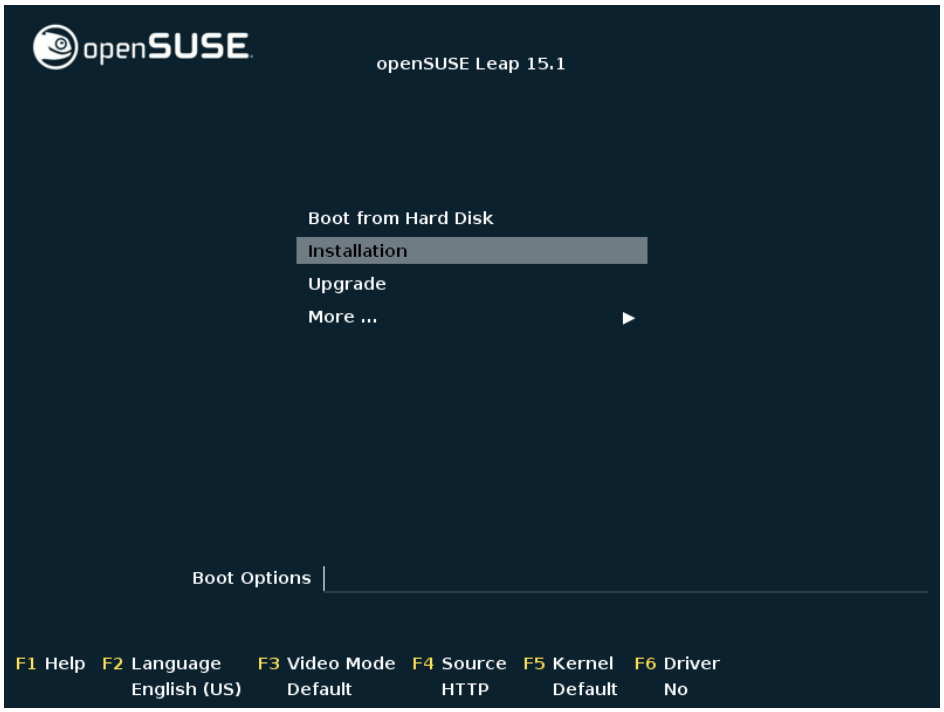


FIGURE 2.1: THE BOOT SCREEN ON MACHINES WITH A TRADITIONAL BIOS

Use the function keys shown at the bottom of the screen to change the language, screen resolution, installation source or to add an additional driver from your hardware vendor:

F1 *Help*

Get context-sensitive help for the active element of the boot screen. Use the arrow keys to navigate, Enter to follow a link, and Esc to leave the help screen.

F2 *Language*

Select the display language and a corresponding keyboard layout for the installation. The default language is English (US).

F3 *Video Mode*

Select various graphical display modes for the installation. By *Default* the video resolution is automatically determined using KMS ("Kernel Mode Setting"). If this setting does not work on your system, choose *No KMS* and, optionally, specify `vga=ask` on the boot command line to get prompted for the video resolution. Choose *Text Mode* if the graphical installation causes problems.

F4 *Source*

Normally, the installation is performed from the inserted installation medium. Here, select other sources, like FTP or NFS servers. If the installation is deployed on a network with an SLP server, select an installation source available on the server with this option.

F5 ***Kernel***

If you encounter problems with the regular installation, this menu offers to disable a few potentially problematic functions. If your hardware does not support ACPI (advanced configuration and power interface) select *No ACPI* to install without ACPI support. *No local APIC* disables support for APIC (Advanced Programmable Interrupt Controllers) which may cause problems with some hardware. *Safe Settings* boots the system with the DMA mode (for CD/DVD-ROM drives) and power management functions disabled.

If you are not sure, try the following options first: *Installation—ACPI Disabled* or *Installation —Safe Settings*. Experts can also use the command line (*Boot Options*) to enter or change kernel parameters.

F6 ***Driver***

Press this key to notify the system that you have an optional driver update for openSUSE Leap. With *File* or *URL*, load drivers directly before the installation starts. If you select *Yes*, you are prompted to insert the update disk at the appropriate point in the installation process.

2.2.2 The Boot Screen on Machines Equipped with UEFI

UEFI (Unified Extensible Firmware Interface) is a new industry standard which replaces and extends the traditional BIOS. The latest UEFI implementations contain the “Secure Boot” extension, which prevents booting malicious code by only allowing signed boot loaders to be executed. See *Book “*Reference*”, Chapter 14 “UEFI (Unified Extensible Firmware Interface)”* for more information.

The boot manager GRUB 2, used to boot machines with a traditional BIOS, does not support UEFI, therefore GRUB 2 is replaced with GRUB 2 for EFI. If Secure Boot is enabled, YaST will automatically select GRUB 2 for EFI for installation. From an administrative and user perspective, both boot manager implementations behave the same and are called `GRUB 2` in the following.

Tip: Using Additional Drivers with Secure Boot

When installing with Secure Boot enabled, you cannot load drivers that are not shipped with openSUSE Leap. This is also true of drivers shipped via SolidDriver, because their signing key is not trusted by default.

To load drivers not shipped with openSUSE Leap, do either of the following:

- Before the installation, add the needed keys to the firmware database via firmware/system management tools.
- Use a bootable ISO that will enroll the needed keys in the MOK list on the first boot.

For more information, see *Book "Reference", Chapter 14 "UEFI (Unified Extensible Firmware Interface)", Section 14.1 "Secure Boot"*.

The boot screen displays several options for the installation procedure. Change the selected option with the arrow keys and press Enter to boot it. The relevant options are:

Installation

The normal installation mode. All modern hardware functions are enabled. In case the installation fails, see F5 *Kernel* for boot parameters that disable potentially problematic functions.

Upgrade

Perform a system upgrade. For more information refer to *Chapter 13, Upgrading the System and System Changes*.

More › Rescue System

Starts a minimal Linux system without a graphical user interface. For more information, see *Section 17.5.2, "Using the Rescue System"*. This option is not available on Live CDs.

More › Boot Linux System

Boot a Linux system that is already installed. You will be asked from which partition to boot the system.

More › Check Installation Media

This option is only available when you install from media created from downloaded ISOs. In this case it is recommended to check the integrity of the installation medium. This option starts the installation system before automatically checking the media. In case the check was successful, the normal installation routine starts. If a corrupt media is detected, the installation routine aborts.

FIGURE 2.2: THE BOOT SCREEN ON MACHINES WITH UEFI

GRUB 2 for EFI on openSUSE Leap does not support a boot prompt or function keys for adding boot parameters. By default, the installation will be started with American English and the boot media as the installation source. A DHCP lookup will be performed to configure the network. To change these defaults or to add boot parameters you need to edit the respective boot entry. Highlight it using the arrow keys and press E. See the on-screen help for editing hints (note that only an English keyboard is available now). The *Installation* entry will look similar to the following:

```
setparams 'Installation'

   set gfxpayload=keep
   echo 'Loading kernel ...'
   linuxefi /boot/x86_64/loader/linux splash=silent
   echo 'Loading initial ramdisk ...'
   initrdefi /boot/x86_64/loader/initrd
```

Add space-separated parameters to the end of the line starting with `linuxefi`. To boot the edited entry, press F10. If you access the machine via serial console, press Esc–0. A complete list of parameters is available at http://en.opensuse.org/Linuxrc.

2.3 List of Important Boot Parameters

This section contains a selection of important boot parameters.

2.3.1 General Boot Parameters

`autoyast=` *URL*

The `autoyast` parameter specifies the location of the `autoinst.xml` control file for automatic installation.

`manual=<0|1>`

The `manual` parameter controls whether the other parameters are only default values that still must be acknowledged by the user. Set this parameter to `0` if all values should be accepted and no questions asked. Setting `autoyast` implies setting `manual` to `0`.

`Info=` *URL*

Specifies a location for a file from which to read additional options.

`upgrade=<0|1>`

To upgrade openSUSE Leap, specify `Upgrade=1`.

`dud=` *URL*

Load driver updates from *URL*.

Set `dud=ftp://ftp.example.com/PATH_TO_DRIVER` or `dud=http://www.example.com/PATH_TO_DRIVER` to load drivers from a URL. When `dud=1` you will be asked for the URL during boot.

`language=` *LANGUAGE*

Set the installation language. Some supported values are `cs_CZ`, `de_DE`, `es_ES`, `fr_FR`, `ja_JP`, `pt_BR`, `pt_PT`, `ru_RU`, `zh_CN`, and `zh_TW`.

`acpi=off`

Disable ACPI support.

`noapic`

No logical APIC.

`nomodeset`

Disable KMS.

`textmode=1`

Start installer in text mode.

`console=` *`SERIAL_DEVICE`*`[,`*`MODE`*`]`

`SERIAL_DEVICE` can be an actual serial or parallel device (for example `ttyS0`) or a virtual terminal (for example `tty1`). *`MODE`* is the baud rate, parity and stop bit (for example `9600n8`). The default for this setting is set by the mainboard firmware. If you do not see output on your monitor, try setting `console=tty1`. It is possible to define multiple devices.

2.3.2 Configuring the Network Interface

Important: Configuring the Network Interface

The settings discussed in this section apply only to the network interface used during installation. Configure additional network interfaces in the installed system by following the instructions given in *Book "*Reference*", Chapter 13 "Basic Networking", Section 13.6 "Configuring a Network Connection Manually"*.

The network will only be configured if it is required during the installation. To force the network to be configured, use the `netsetup` parameter.

`netsetup=`*`VALUE`*

`netsetup=dhcp` forces a configuration via DHCP. Set `netsetup=-dhcp` when configuring the network with the boot parameters `hostip`, `gateway` and `nameserver`. With the option `netsetup=hostip,netmask,gateway,nameserver` the installer asks for the network settings during boot.

`ifcfg=`*`INTERFACE`*`[.`*`VLAN`*`]=`*`SETTINGS`*

`INTERFACE` can be `*` to match all interfaces or, for example, `eth*` to match all interfaces that start with `eth`. It is also possible to use MAC addresses as values.
Optionally, a VLAN can be set behind the interface name, separated by a period.

If *SETTINGS* is `dhcp`, all matching interfaces will be configured with DHCP. It is possible to set static parameters. With static parameters, only the first matching interface will be configured. The syntax for the static configuration is:

```
ifcfg=*="IPS_NETMASK,GATEWAYS,NAMESERVERS,DOMAINS"
```

Each comma separated value can in turn contain a list of space character separated values. *IPS_NETMASK* is in the *CIDR notation*, for example `10.0.0.1/24`. The quotes are only needed when using space character separated lists. Example with two name servers:

```
ifcfg=*="10.0.0.10/24,10.0.0.1,10.0.0.1 10.0.0.2,example.com"
```

`hostname=host.example.com`

Enter the fully qualified host name.

`domain=example.com`

Domain search path for DNS. Allows you to use short host names instead of fully qualified ones.

`hostip=192.168.1.2[/24]`

Enter the IP address of the interface to configure. The IP can contain the subnet mask, for example `hostip=192.168.1.2/24`. This setting is only evaluated if the network is required during the installation.

`gateway=192.168.1.3`

Specify the gateway to use. This setting is only evaluated if the network is required during the installation.

`nameserver=192.168.1.4`

Specify the DNS server in charge. This setting is only evaluated if the network is required during the installation.

`domain=example.com`

Domain search path. This setting is only evaluated if the network is required during the installation.

2.3.3 Specifying the Installation Source

If you are not using the DVD for installation, specify an alternative installation source.

`install=SOURCE`

Specify the location of the installation source to use. Possible protocols are `cd`, `hd`, `slp`, `nfs`, `smb` (Samba/CIFS), `ftp`, `tftp` `http`, and `https`. Not all source types are available on all platforms.
The default option is `cd`.
If an `ftp`, `tftp` or `smb` URL is given, specify the user name and password with the URL. These parameters are optional and anonymous or guest login is assumed if they are not given. Example:

```
install=ftp://USER:PASSWORD@SERVER/DIRECTORY/DVD1/
```

To install over an encrypted connection, use an `https` URL. If the certificate cannot be verified, use the `sslcerts=0` boot parameter to disable certificate checking.
In case of a Samba or CIFS installation, you can also specify the domain that should be used:

```
install=smb://WORKDOMAIN;USER:PASSWORD@SERVER/DIRECTORY/DVD1/
```

To use `cd`, `hd` or `slp`, set them as the following example:

```
install=cd:/
install=hd:/?device=sda/PATH_TO_ISO
install=slp:/
```

2.3.4 Specifying Remote Access

Only one of the different remote control methods should be specified at a time. The different methods are: SSH, VNC, remote X server.

`display_ip=` *`IP_ADDRESS`*

`Display_IP` causes the installing system to try to connect to an X server at the given address.

> **Important: X Authentication Mechanism**
>
> The direct installation with the X Window System relies on a primitive authentication mechanism based on host names. This mechanism is disabled on current openSUSE Leap versions. Installation with SSH or VNC is preferred.

`vnc=1`

Enables a VNC server during the installation.

`vncpassword=` *`PASSWORD`*

Sets the password for the VNC server.

`ssh=1`

`ssh` enables SSH installation.

`ssh.password=` *`PASSWORD`*

Specifies an SSH password for the root user during installation.

2.4 Advanced Setups

To configure access to a local RMT or `supportconfig` server for the installation, you can specify boot parameters to set up these services during installation. The same applies if you need IPv6 support during the installation.

2.4.1 Using IPv6 for the Installation

By default you can only assign IPv4 network addresses to your machine. To enable IPv6 during installation, enter one of the following parameters at the boot prompt:

Accept IPv4 and IPv6

```
ipv6=1
```

Accept IPv6 only

```
ipv6only=1
```

2.4.2 Using a Proxy for the Installation

In networks enforcing the usage of a proxy server for accessing remote Web sites, registration during installation is only possible when configuring a proxy server.

To use a proxy during the installation, press F4 on the boot screen and set the required parameters in the *HTTP Proxy* dialog. Alternatively provide the kernel parameter `proxy` at the boot prompt:

```
proxy=http://USER:PASSWORD@proxy.example.com:PORT
```

Specifying *USER* and *PASSWORD* is optional—if the server allows anonymous access, the following data is sufficient: `http://proxy.example.com:`*PORT*.

2.4.3 Enabling SELinux Support

Enabling SELinux upon installation start-up enables you to configure it after the installation has been finished without having to reboot. Use the following parameters:

```
security=selinux selinux=1
```

2.4.4 Enabling the Installer Self-Update

During installation and upgrade, YaST can update itself as described in *Section 3.2, "Installer Self-Update"* to solve potential bugs discovered after release. The `self_update` parameter can be used to modify the behavior of this feature.

To enable the installer self-update, set the parameter to `1`:

```
self_update=1
```

To use a user-defined repository, specify a URL:

```
self_update=https://updates.example.com/
```

2.4.5 Scale User Interface for High DPI

If your screen uses a very high DPI, use the boot parameter `QT_AUTO_SCREEN_SCALE_FACTOR`. This scales font and user interface elements to the screen DPI.

```
QT_AUTO_SCREEN_SCALE_FACTOR=1
```

2.4.6 Using CPU Mitigations

The boot parameter `mitigations` lets you control mitigation options for side-channel attacks on affected CPUs. Its possible values are:

`auto`. Enables all mitigations required for your CPU model, but does not protect against cross-CPU thread attacks. This setting may impact performance to some degree, depending on the workload.

`nosmt`. Provides the full set of available security mitigations. Enables all mitigations required for your CPU model. In addition, it disables Simultaneous Multithreading (SMT) to avoid side-channel attacks across multiple CPU threads. This setting may further impact performance, depending on the workload.

`off`. Disables all mitigations. Side-channel attacks against your CPU are possible, depending on the CPU model. This setting has no impact on performance.

Each value comes with a set of specific parameters, depending on the CPU architecture, the kernel version, and on the vulnerabilities that need to be mitigated. Refer to the kernel documentation for details.

2.5 More Information

You can find more information about boot parameters in the openSUSE wiki at https://en.opensuse.org/SDB:Linuxrc#Parameter_Reference.

3 Installation Steps

This chapter describes the procedure in which the data for openSUSE Leap is copied to the target device. Some basic configuration parameters for the newly installed system are set during the procedure. A graphical user interface will guide you through the installation. The text mode installation has the same steps and only looks different. For information about performing non-interactive automated installations, see *Book "AutoYaST Guide"*.

If you are a first-time user of openSUSE Leap, you should follow the default YaST proposals in most parts, but you can also adjust the settings as described here to fine-tune your system according to your preferences. Help for each installation step is provided by clicking *Help*.

Tip: Installation Without a Mouse

If the installer does not detect your mouse correctly, use [→|] for navigation, arrow keys to scroll, and [Enter] to confirm a selection. Various buttons or selection fields contain a letter with an underscore. Use [Alt]–[Letter] to select a button or a selection directly instead of navigating there with [→|].

3.1 Overview

This section provides an overview of all installation steps. Each step contains a link to a more detailed description.

1. Before the installation starts, the installer can update itself. For details, see *Section 3.2, "Installer Self-Update"*.
2. The actual installation starts with choosing the language and accepting the license agreement. For details, see *Section 3.3, " Language, Keyboard, and License Agreement "*.
3. Configure the network. This is only required when you need network access during the installation and the automatic network configuration via DHCP failed. If the automatic network configuration succeeded, this step is skipped. For details, see *Section 3.4, "Network Settings"*.

4. Configure the online repositories. By adding official openSUSE repositories, you get access to more software and get the latest security updates already during installation. For details, see *Section 3.5, "Online Repositories"*. This step is optional and can be skipped.

5. Select a desktop or a role for your system. Among other things, this defines the default list of packages to install and makes a suggestion for partitioning the hard disks. For details, see *Section 3.6, "System Role"*.

6. Partition the hard disks of your system. For details, see *Section 3.7, "Partitioning"*.

7. Choose a time zone. For details, see *Section 3.8, "Clock and Time Zone"*.

8. Create a user. For details, see *Section 3.9, "Create New User"*.

9. Optionally, set a different password for the system administrator `root`. For details, see *Section 3.10, "Authentication for the System Administrator "root""*.

10. In a final step, the installer presents an overview of all settings. If required, you can change them. For details, see *Section 3.11, "Installation Settings"*.

11. The installer copies all required data and informs you about the progress. For details, see *Section 3.12, "Performing the Installation"*.

3.2 Installer Self-Update

During the installation and upgrade process, YaST can update itself to solve bugs in the installer that were discovered after the release. This functionality is enabled by default; to disable it, set the boot parameter `self_update` to `0`. For more information, see *Section 2.4.4, "Enabling the Installer Self-Update"*.

Important: Networking during Self-Update

To download installer updates, YaST needs network access. By default, it tries to use DHCP on all network interfaces. If there is a DHCP server in the network, it will work automatically.

If you need a static IP setup, you can use the `ifcfg` boot argument. For more details, see the linuxrc documentation at https://en.opensuse.org/Linuxrc.

Tip: Language Selection

The installer self-update is executed before the language selection step. This means that progress and errors which happen during this process are displayed in English by default.

To use another language for this part of the installer, use the `language` boot parameter if available for your architecture, for example, `language=de_DE`. On machines equipped with a traditional BIOS, alternatively, press F2 in the boot menu and select the language from the list.

Although this feature was designed to run without user intervention, it is worth knowing how it works. If you are not interested, you can jump directly to *Section 3.3, "Language, Keyboard, and License Agreement"* and skip the rest of this section.

3.2.1 Self-Update Process

The process can be broken down into two different parts:

1. Determine the update repository location.
2. Download and apply the updates to the installation system.

3.2.1.1 Determining the Update Repository Location

Installer Self-Updates are distributed as regular RPM packages via a dedicated repository, so the first step is to find out the repository URL.

Important: Installer Self-Update Repository Only

No matter which of the following options you use, only the installer self-update repository URL is expected, for example:

```
self_update=https://www.example.com/my_installer_updates/
```

Do not supply any other repository URL—for example the URL of the software update repository.

YaST will try the following sources of information:

1. The `self_update` boot parameter. (For more details, see *Section 2.4.4, "Enabling the Installer Self-Update"*.) If you specify a URL, it will take precedence over any other method.
2. The `/general/self_update_url` profile element in case you are using AutoYaST.
3. If none of the previous attempts worked, the fallback URL (defined in the installation media) will be used.

3.2.1.2 Downloading and Applying the Updates

When the updates repository is determined, YaST will check whether an update is available. If so, all the updates will be downloaded and applied to the installation system.

Finally, YaST will be restarted to load the new version and the welcome screen will be shown. If no updates were available, the installation will continue without restarting YaST.

Note: Update Integrity

Update signatures will be checked to ensure integrity and authorship. If a signature is missing or invalid, you will be asked whether you want to apply the update.

3.2.1.3 Temporary Self-Update Add-on Repository

Some packages distributed in the self-update repository provide additional data for the installer, like the installation defaults, system role definitions and similar. If the installer finds such packages in the self-update repository, a local temporary repository is created, to which those packages are copied. They are used during the installation process, but at the end of the installation, the temporary local repository is removed. Its packages are *not* installed onto the target system.

This additional repository is not displayed in the list of add-on products, but during installation it may still be visible as `SelfUpdate0` repository in the package management.

3.2.2 Custom Self-Update Repositories

YaST can use a user-defined repository instead of the official one by specifying a URL through the `self_update` boot parameter. However, the following points should be considered:

- Only HTTP/HTTPS and FTP repositories are supported.
- Only RPM-MD repositories are supported (required by RMT).
- Packages are not installed in the usual way: They are uncompressed only and no scripts are executed.
- No dependency checks are performed. Packages are installed in alphabetical order.
- Files from the packages override the files from the original installation media. This means that the update packages might not need to contain all files, only files that have changed. Unchanged files are omitted to save memory and download bandwidth.

Note: Only One Repository

Currently, it is not possible to use more than one repository as source for installer self-updates.

3.3 Language, Keyboard, and License Agreement

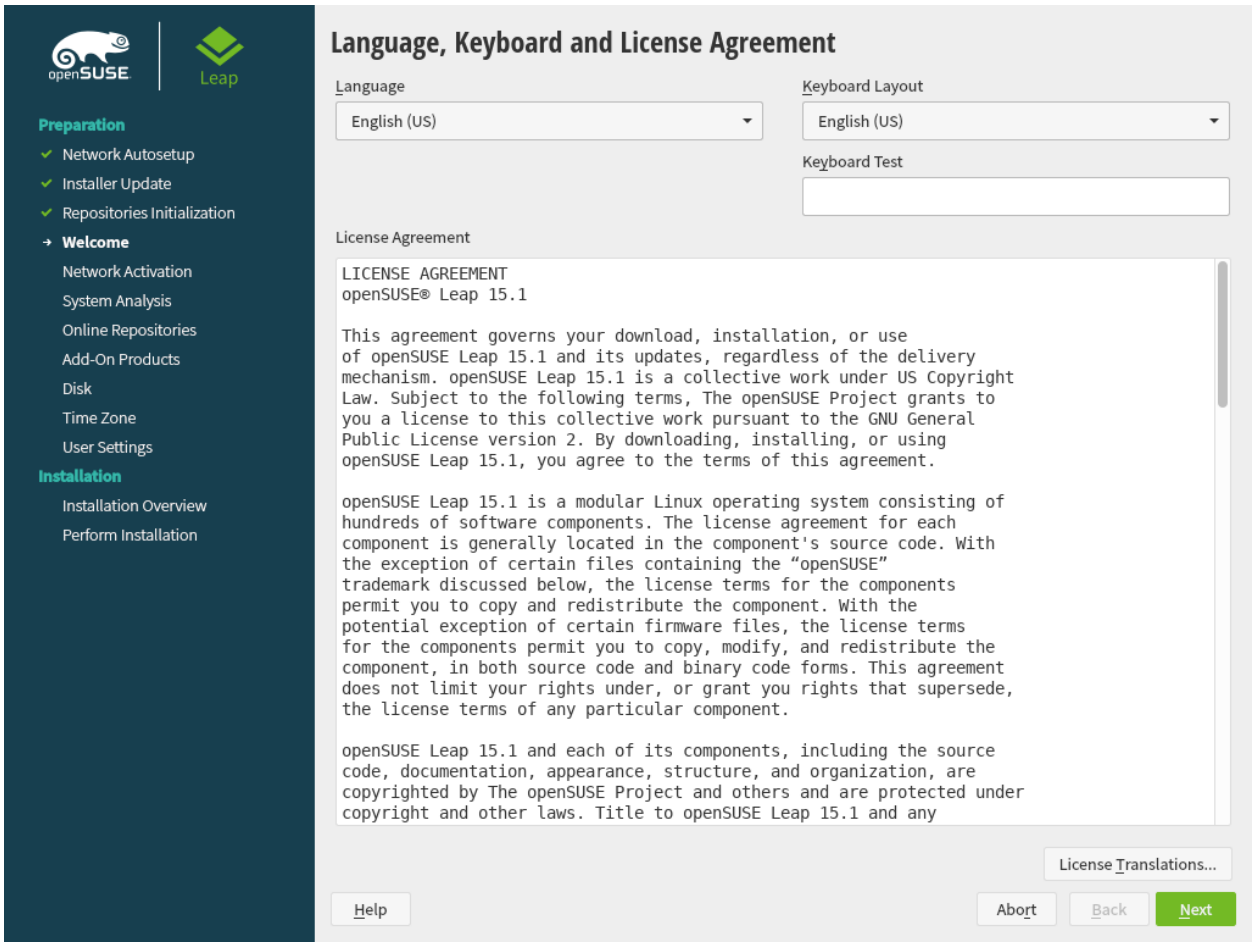


FIGURE 3.1: LANGUAGE, KEYBOARD, AND LICENSE AGREEMENT

The *Language* and *Keyboard Layout* settings are initialized with the language you chose on the boot screen. If you did not change the default, it will be English (US). Change the settings here, if necessary.

Changing the language will automatically preselect a corresponding keyboard layout. Override this proposal by selecting a different keyboard layout from the drop-down box. Use the *Keyboard Test* text box to test the layout. The language selected here is also used to assume a time zone for the system clock. This setting can be modified later in the installed system as described in *Chapter 6, Changing Language and Country Settings with YaST*.

Read the license agreement. It is presented in the language you have. Translations are available via the *License Language* drop-down box. Proceed with *Next* if you agree to the terms and conditions. If you do not agree, click *Abort* to terminate the installation.

3.4 Network Settings

After booting into the installation, the installation routine is set up. During this setup, an attempt to configure at least one network interface with DHCP is made. In case this attempt has failed, the *Network Settings* dialog launches now.

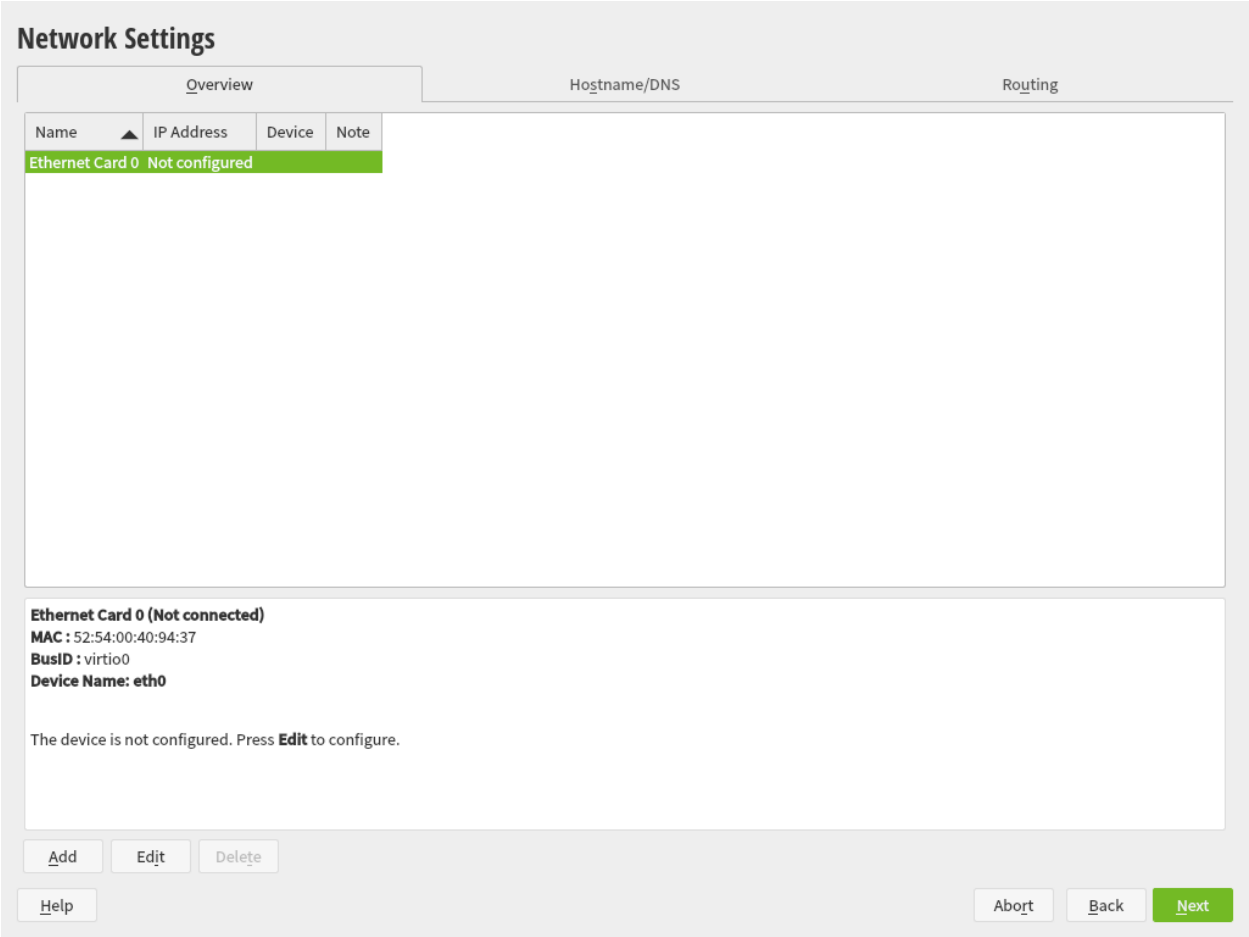


FIGURE 3.2: NETWORK SETTINGS

Choose a network interface from the list and click *Edit* to change its settings. Use the tabs to configure DNS and routing. See *Book "Reference", Chapter 13 "Basic Networking", Section 13.4 "Configuring a Network Connection with YaST"* for more details.

In case DHCP was successfully configured during installation setup, you can also access this dialog by clicking *Network Configuration* at the the *Installation Settings* step. It lets you change the automatically provided settings.

Note: Network Configuration with Boot Parameters

If at least one network interface has been configured via boot parameters (see *Section 2.3.2, "Configuring the Network Interface"*), automatic DHCP configuration is disabled and the boot parameter configuration is imported and used.

Tip: Accessing Network Storage or Local RAID

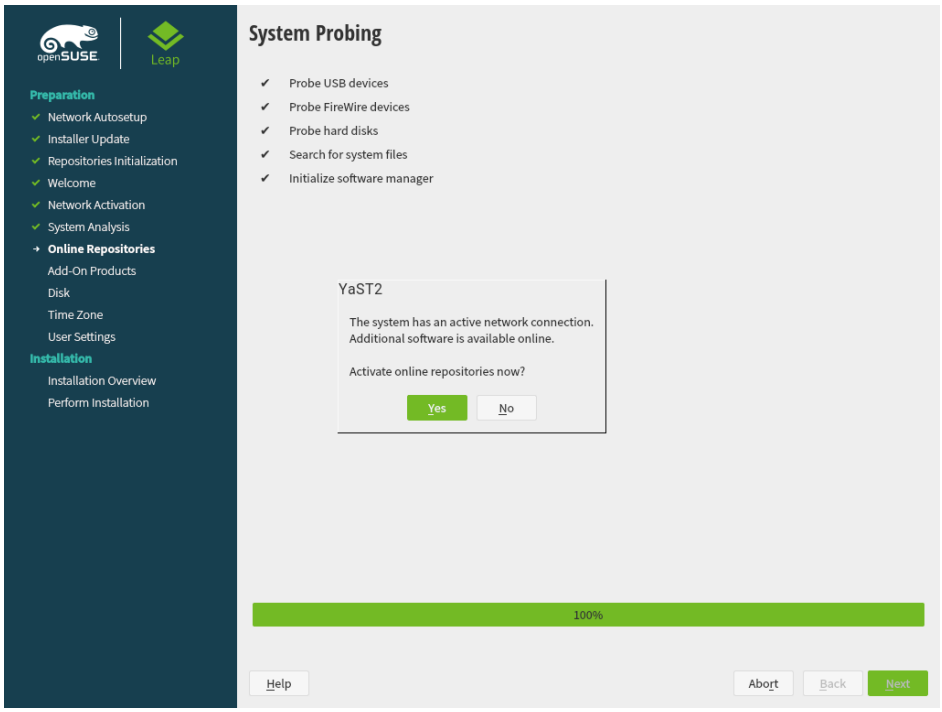
To access a SAN or a local RAID during the installation, you can use the libstorage command line client for this purpose:

1. Switch to a console with Ctrl–Alt–F2.
2. Install the libstoragemgmt extension by running **extend libstoragemgmt**.
3. Now you have access to the **lsmcli** command. For more information, run **lsmcli --help**.
4. To return to the installer, press Alt–F7

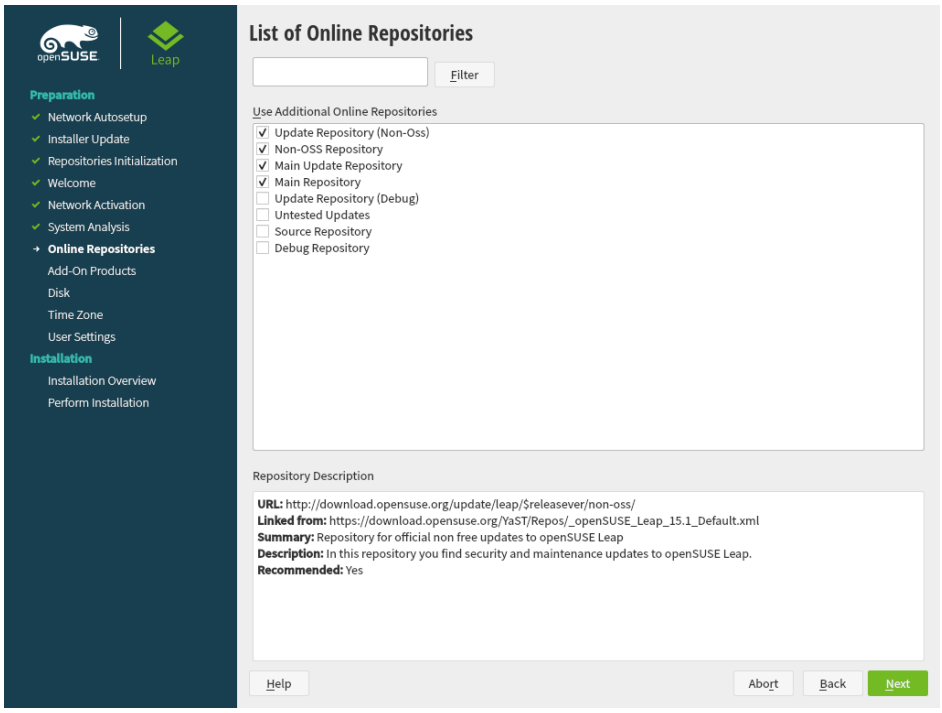
Supported are Netapp Ontap, all SMI-S compatible SAN providers, and LSI MegaRAID.

3.5 Online Repositories

A system analysis is performed, where the installer probes for storage devices, and tries to find other installed systems. If a network connection with Internet access is available, you will be asked to activate the online repositories. Answer with *Yes* to proceed. In case you do not have Internet access, this step will be skipped.



The online repositories are official openSUSE package sources. They not only offer additional packages not included on the installation media, but also the update repositories containing security and bug fixes. Using the default selection is recommended. Add at least the *Main Update Repository*, because it makes sure the system is installed with the latest security patches.



You have the following choices:

- The *Main Repository (OSS)* contains open source software (OSS). Compared to the DVD installation media, it contains many additional software packages, among them many additional desktop systems.
- The *Main Update Repository* contains security updates and fixes for packages from the *Main Repository (OSS)* and the DVD installation media. Choosing this repository is recommended for all installation scenarios.
- The *Main Repository (Non-OSS)* contains packages with a proprietary software license. Choosing it is not required for installing a custom desktop system.
- Choosing *Main Update Repository (Non-OSS)* is recommended when also having chosen the *Main Repository (Non-OSS)*. It contains the respective updates and security fixes.
- All other repositories are intended for experienced users and developers. Click on a repository name to get more information.

Confirm your selection with *Next*. Depending on your choice, you need to confirm one or more license agreements. Do so by choosing *Next* until you proceed to the *System Role* screen. Now choose *Next* to proceed.

3.6 System Role

openSUSE Leap supports a broad range of features. To simplify the installation, the installer offers predefined use cases which adjust the system to be installed so it is tailored for the selected scenario. Currently this affects the package set and the suggested partitioning scheme.

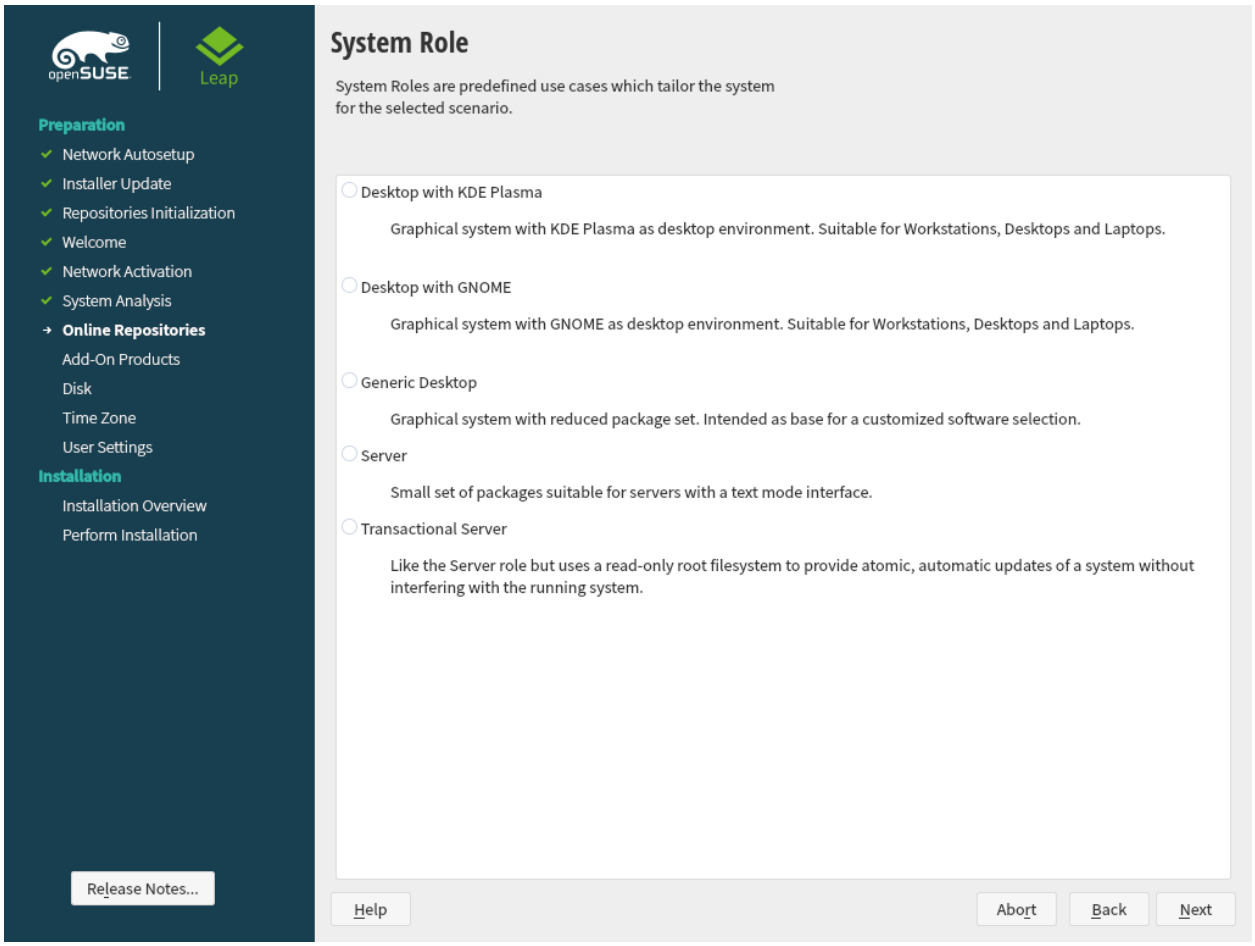


FIGURE 3.3: SYSTEM ROLE

Choose the *System Role* that meets your requirements best. The availability of system roles depends on your selection of modules and extensions. The following system roles are available with the default selection:

Desktop with KDE Plasma

A powerful desktop environment with a complete PIM suite (mail, calendar, tasks, notes, and feeds), widgets runnning on the desktop and many more features. If you are familiar with Windows, KDE is the recommended choice. For more information see https://kde.org/.

Desktop with GNOME

A desktop environment offering an alternative, innovative user experience. GNOME was designed with usability and productivity in mind. For more information see https://www.gnome.org/.

Generic Desktop

In case you prefer an alterntive to the KDE or GNOME desktops, choose this option. You will be able to choose between the following alternatives later in the installation process by selecting *Software* in the *Installation Settings dialog*:

Enlightenment (https://www.enlightenment.org/)
LXDE (https://lxde.org/)
LXQT (https://lxqt.org/)
MATE (https://mate-desktop.org/)
XFCE (https://xfce.org/)
Note that when installing from the DVD all these desktop systems except XFCE are only available when having enabled the MAIN Repository (OSS) in the *Online Repositories* step. You can still enable this repository now by using the *Back* button until you reach the welcome screen. From there, choose *Next* and then agree to add online repositories.

Server

If setting up a server, you probably do not need a graphical user interface and desktop applications such as an office suite. This option gives you a reduced set of packages suitable for servers.

Transactional Server

Similar to the server role, but with a read-only root partition and transactional updates. This selection also is a prerequisite for setting up openSUSE Kubic. See https://kubic.opensuse.org/blog/2018-04-04-transactionalupdates/ for more information on transactional updates.

3.7 Partitioning

3.7.1 Important Information

Warning: Read this Section Carefully

Read this section carefully before continuing with *Section 3.7.2, "Suggested Partitioning"*.

Custom Partitioning on UEFI Machines

A UEFI machine *requires* an EFI system partition that must be mounted to `/boot/efi`. This partition must be formatted with the `FAT32` file system.

If an EFI system partition is already present on your system (for example from a previous Windows installation) use it by mounting it to `/boot/efi` without formatting it.

If no EFI system partition is present on your UEFI machine, make sure to create it. The EFI system partition must be a physical partition or RAID 1. Other RAID levels, LVM and other technologies are not supported. It needs to be formatted with the FAT32 file system.

Custom Partitioning and Snapper

If the root partition is larger than 16 GB, openSUSE Leap by default enables file system snapshots.

openSUSE Leap uses Snapper together with Btrfs for this feature. Btrfs needs to be set up with snapshots enabled for the root partition.

If the disk is smaller than 16 GB, all Snapper features and automatic snapshots are disabled to prevent the system partition `/` from running out of space.

Being able to create system snapshots that enable rollbacks requires important system directories to be mounted on a single partition, for example `/usr` and `/var`. Only directories that are excluded from snapshots may reside on separate partitions, for example `/usr/local`, `/var/log`, and `/tmp`.

For details, see *Book "Reference", Chapter 3 "System Recovery and Snapshot Management with Snapper"*.

Important: Btrfs Snapshots and Root Partition Size

Snapshots occupy space on their partition. As a rule of thumb, the older a snapshot is, or the bigger the changeset they cover is, the bigger the snapshot. Plus, the more snapshots you keep, the more diskspace you need.

To prevent the root partition running full with snapshot data, you need to make sure it is big enough. In case you do frequent updates or other installations, consider at least 30 GB for the root partition. If you plan to keep snapshots activated for a system upgrade (to be able to roll back), you should consider 40 GB or more.

Btrfs Data Volumes

Using Btrfs for data volumes is supported on openSUSE Leap 15.1. For applications that require Btrfs as a data volume, consider creating a separate file system with quota groups disabled. This is already the default for non-root file systems.

Btrfs on an Encrypted Root Partition

The default partitioning setup suggests the root partition as Btrfs. To encrypt the root partition, make sure to use the GPT partition table type instead of the MSDOS type. Otherwise the GRUB2 boot loader may not have enough space for the second stage loader.

Supported Software RAID Volumes

Installing to and booting from existing software RAID volumes is supported for Disk Data Format (DDF) volumes and Intel Matrix Storage Manager (IMSM) volumes. IMSM is also known by the following names:

- Intel Rapid Storage Technology
- Intel Matrix Storage Technology
- Intel Application Accelerator / Intel Application Accelerator RAID Edition

Mount Points for FCoE and iSCSI Devices

FCoE and iSCSI devices will appear asynchronously during the boot process. While the initrd guarantees that those devices are set up correctly for the root file system, there are no such guarantees for any other file systems or mount points like `/usr`. Hence any system mount points like `/usr` or `/var` are not supported. To use those devices, ensure correct synchronization of the respective services and devices.

Handling of Windows Partitions in Proposals

In case the disk selected for the suggested partitioning proposal contains a large Windows FAT or NTFS partition, it will automatically be resized to make room for the openSUSE Leap installation. To avoid data loss it is strongly recommended to

- make sure the partition is not fragmented (run a defragmentation program from Windows prior to the openSUSE Leap installation)
- double-check the suggested size for the Windows partition is big enough
- back up your data prior to the openSUSE Leap installation

To adjust the proposed size of the Windows partition, use the *Expert Partitioner*.

Proposal with a Separate Home Partition

The default proposal no longer suggests to create a separate partition for `/home`. The `/home` directory contains the user's data and personal configuration files. Placing it on a separate directory makes it easier to rebuild the system in the future, or allows to share it with different Linux installations on the same machine.

In case you want to change the proposal to create a separate partition for `/home`, choose *Guided Setup* and click *Next* until you reach the *Filesystem Options* screen. Check *Propose Separate Home Partition*. By default it will be formatted with *XFS*, but you can choose to use a different file system. Close the dialog by clicking *Next* again.

3.7.2 Suggested Partitioning

Define a partition setup for openSUSE Leap in this step.

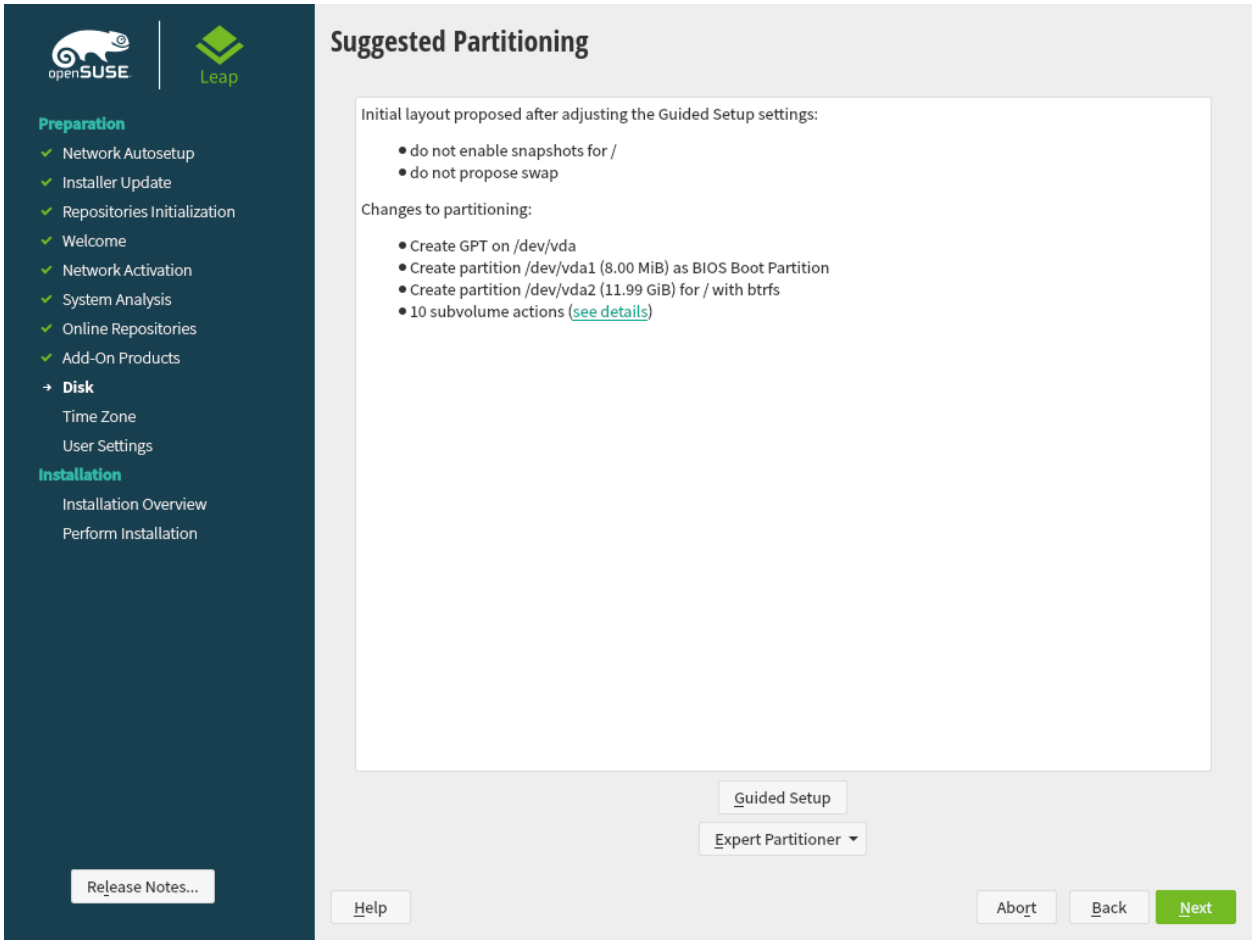


FIGURE 3.4: SUGGESTED PARTITIONING

The installer creates a proposal for one of the available disks containing a root partition formatted with Btrfs and a swap partition. If one or more swap partitions have been detected on the available hard disks, these partitions will be used. You have several options to proceed:

Next

To accept the proposal without any changes, click *Next* to proceed with the installation workflow.

Guided Setup

To adjust the proposal, choose *Guided Setup*. First, choose which hard disks and partitions to use. In the *Partitioning Scheme* screen, you can enable Logical Volume Management (LVM) and activate disk encryption. Afterwards specify the *Filesystem Options*. You can adjust the file system for the root partition and create a separate home and swap partitions. If you plan to suspend your machine, make sure to create a separate swap partition and check *Enlarge to RAM Size for Suspend*. If the root file system format is Btrfs, you can also enable or disable Btrfs snapshots here.

Expert Partitioner

To create a custom partition setup click *Expert Partitioner*. Select either *Start with Current Proposal* if you want start with the suggested disk layout, or *Start with Existing Partitions* to ignore the suggested layout and start with the existing layout on the disk. You can *Add*, *Edit*, *Resize*, or *Delete* partitions.

You can also set up Logical Volumes (LVM), configure software RAID and device mapping (DM), encrypt Partitions, mount NFS shares and manage tmpfs volumes with the Expert Partitioner. To fine-tune settings such as the subvolume and snapshot handling for each Btrfs partition, choose *Btrfs*. For more information about custom partitioning and configuring advanced features, refer to *Book "*Reference*", Chapter 5 "Expert Partitioner", Section 5.1 "Using the Expert Partitioner"*.

3.8 Clock and Time Zone

In this dialog, select your region and time zone. Both are preselected according to the installation language.

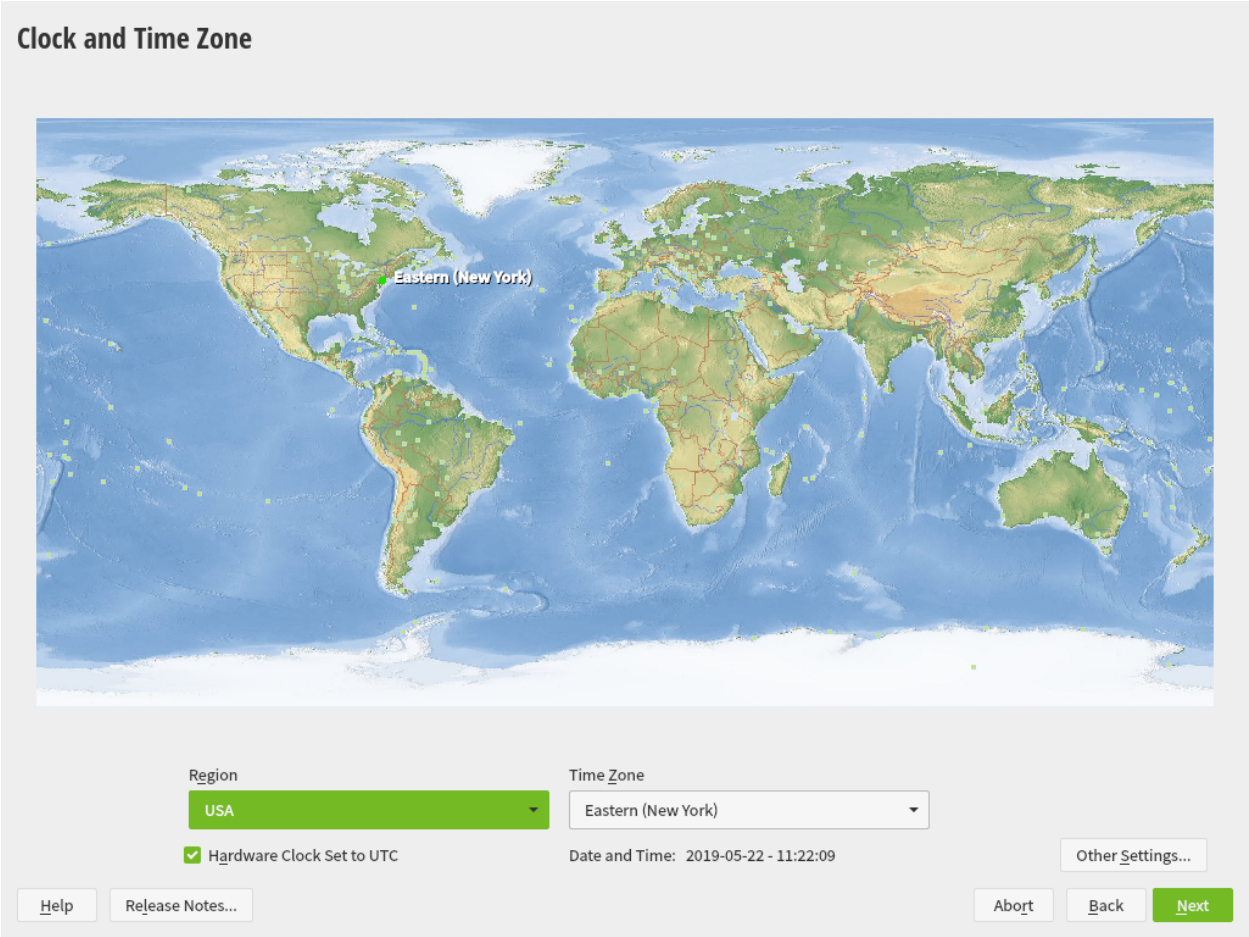


FIGURE 3.5: CLOCK AND TIME ZONE

To change the preselected values, either use the map or the drop-down boxes for *Region* and *Time Zone*. When using the map, point the cursor at the rough direction of your region and left-click to zoom. Now choose your country or region by left-clicking. Right-click to return to the world map.

To set up the clock, choose whether the *Hardware Clock is Set to UTC*. If you run another operating system on your machine, such as Microsoft Windows, it is likely your system uses local time instead. If you run Linux on your machine, set the hardware clock to UTC and have the switch from standard time to daylight saving time performed automatically.

Important: Set the Hardware Clock to UTC

The switch from standard time to daylight saving time (and vice versa) can only be performed automatically when the hardware clock (CMOS clock) is set to UTC. This also applies if you use automatic time synchronization with NTP, because automatic synchronization will only be performed if the time difference between the hardware and system clock is less than 15 minutes.

Since a wrong system time can cause serious problems (missed backups, dropped mail messages, mount failures on remote file systems, etc.), it is strongly recommended to *always* set the hardware clock to UTC.

If a network is already configured, you can configure time synchronization with an NTP server. Click *Other Settings* to either alter the NTP settings or to *Manually* set the time. See *Book "Reference", Chapter 18 "Time Synchronization with NTP"* for more information on configuring the NTP service. When finished, click *Accept* to continue the installation.

If running without NTP configured, consider setting `SYSTOHC=no` (`sysconfig` variable) to avoid saving unsynchronized time into the hardware clock.

3.9 Create New User

Create a local user in this step.

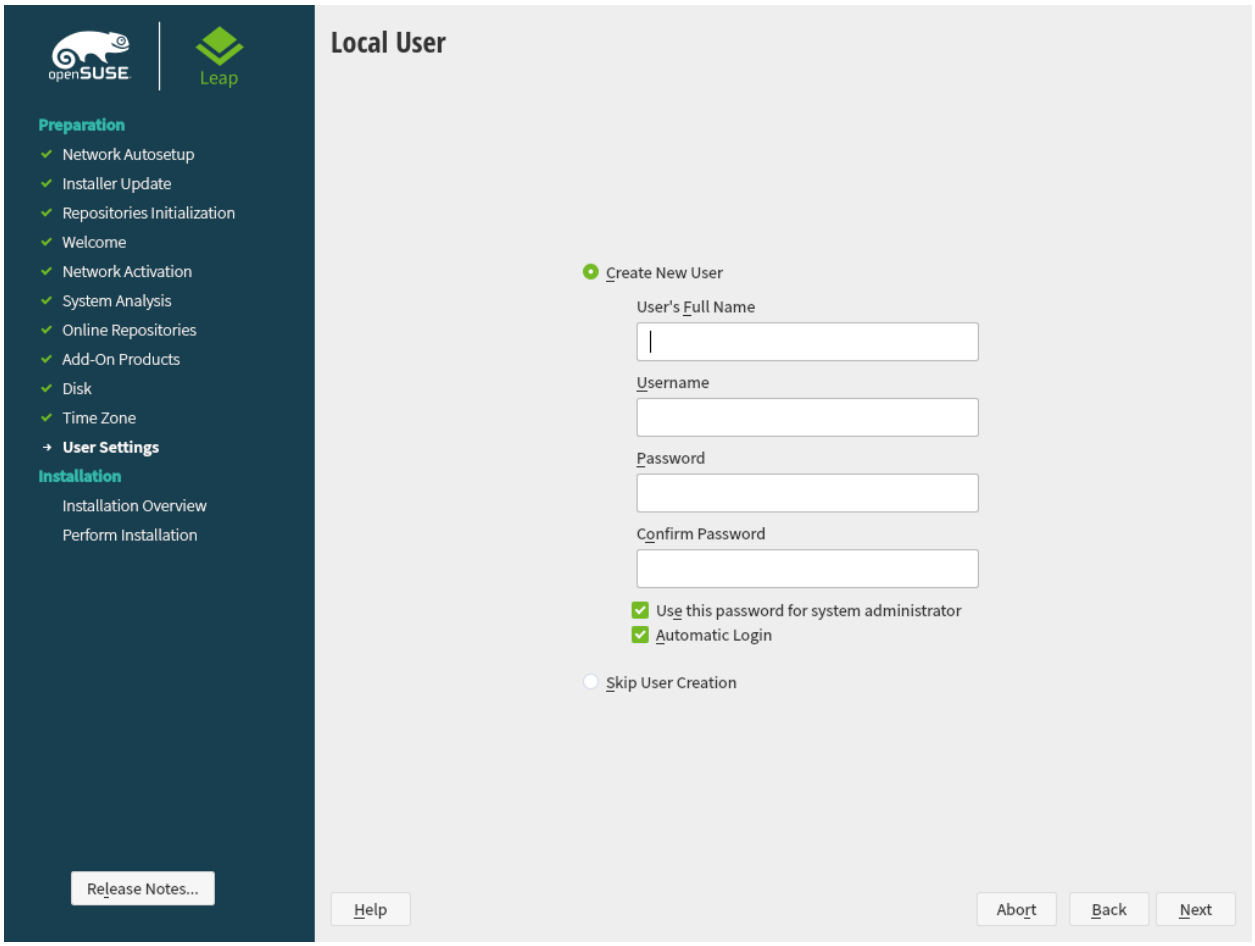


FIGURE 3.6: CREATE NEW USER

After entering the first name and last name, either accept the proposal or specify a new *User name* that will be used to log in. Only use lowercase letters (a-z), digits (0-9) and the characters _. (dot), _- (hyphen) and __ (underscore). Special characters, umlauts and accented characters are not allowed.

Finally, enter a password for the user. Re-enter it for confirmation (to ensure that you did not type something else by mistake). To provide effective security, a password should be at least six characters long and consist of uppercase and lowercase letters, numbers and special characters (7-bit ASCII). Umlauts or accented characters are not allowed. Passwords you enter are checked for weakness. When entering a password that is easy to guess (such as a dictionary word or a name) you will see a warning. It is a good security practice to use strong passwords.

Important: User Name and Password

Remember both your user name and the password because they are needed each time you log in to the system.

If you install openSUSE Leap on a machine with one or more existing Linux installations, YaST allows you to import user data such as user names and passwords. Select *Import User Data from a Previous Installation* and then *Choose Users* for import.

If you do not want to configure any local users (for example when setting up a client on a network with centralized user authentication), skip this step by choosing *Next* and confirming the warning. Network user authentication can be configured at any time later in the installed system; refer to *Chapter 5, Managing Users with YaST* for instructions.

Two additional options are available:

Use this Password for System Administrator

If checked, the same password you have entered for the user will be used for the system administrator `root`. This option is suitable for stand-alone workstations or machines in a home network that are administrated by a single user. When not checked, you are prompted for a system administrator password in the next step of the installation workflow (see *Section 3.10, "Authentication for the System Administrator "root""*).

Automatic Login

This option automatically logs the current user in to 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. If you store sensitive data on your system, you should not enable this option if the computer can also be accessed by others.

In an environment where users are centrally managed (for example by NIS or LDAP) you may want to skip the creation of local users. Select *Skip User Creation* in this case.

3.10 Authentication for the System Administrator “root”

If you have not chosen *Use this Password for System Administrator* in the previous step, you will be prompted to enter a password for the System Administrator `root` or provide a public SSH key. Otherwise this configuration step is skipped.

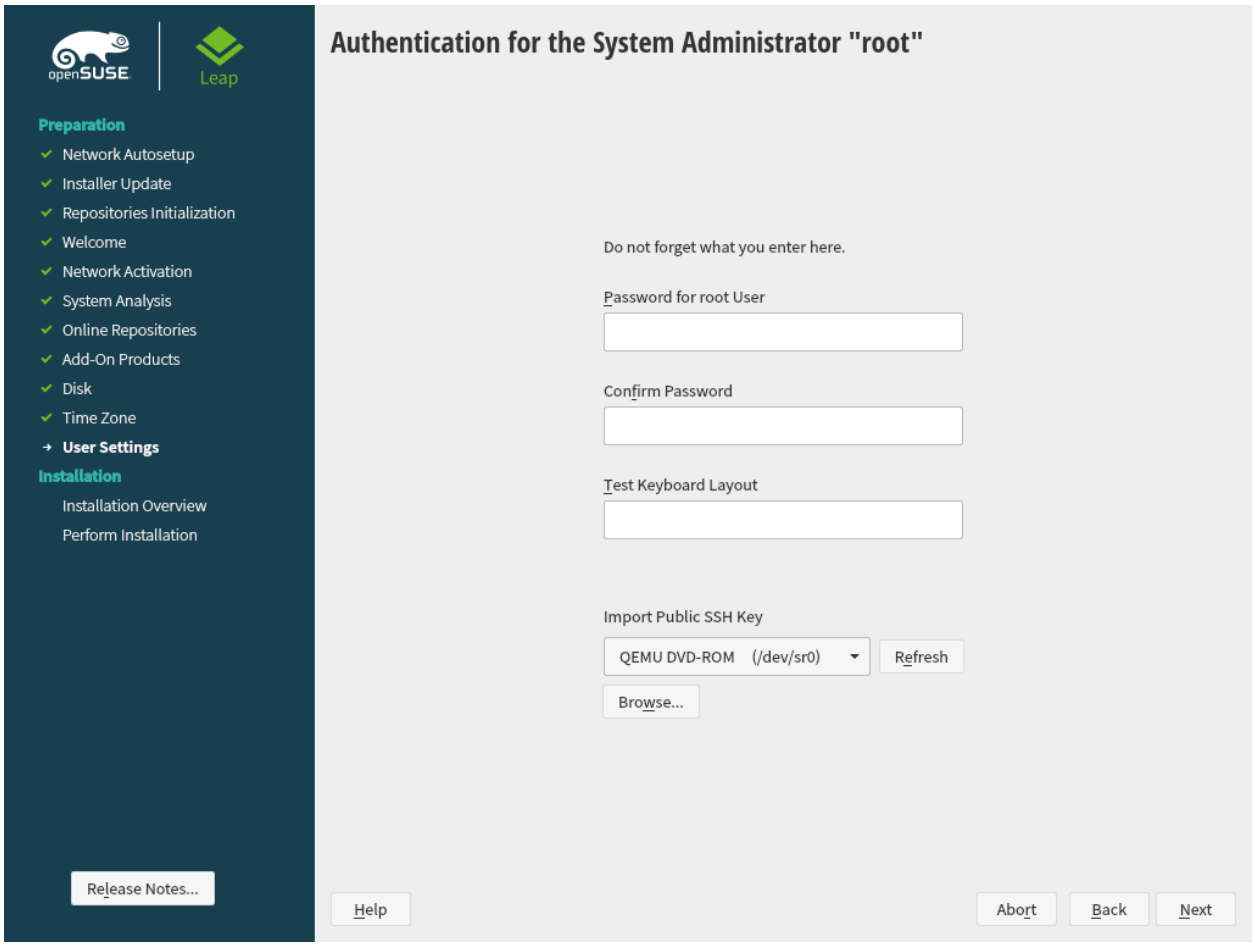


FIGURE 3.7: AUTHENTICATION FOR THE SYSTEM ADMINISTRATOR `root`

`root` is the name of the superuser, or the administrator of the system. Unlike regular users, `root` has unlimited rights to 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 administration, maintenance, and repair. Logging in as `root` for daily work is rather risky: a single mistake could lead to irretrievable loss of system files.

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

Tip: Passwords and Keyboard Layout

It is recommended to only use characters that are available on an English keyboard. In case of a system error or when you need to start your system in rescue mode a localized keyboard might not be available.

The `root` password can be changed any time later in the installed system. To do so run YaST and start *Security and Users* › *User and Group Management*.

Important: The `root` User

The user `root` has all the permissions needed to make changes to the system. To carry out such tasks, the `root` password is required. You cannot carry out any administrative tasks without this password.

In some situations it is preferred to access the system remotely via SSH using a public key. This screen allows you to select a public key from a medium.

The following procedure describes how to add a public SSH key from a USB stick. It works the same way with CD/DVD-ROM or from an existing partition. Proceed as follows:

PROCEDURE 3.1: ADDING A PUBLIC SSH KEY FOR USER `root`

1. Insert into your computer the USB storage device containing the public SSH key. The public SSH key has the file extension `.pub`.
2. Click *Refresh*. You should see the device in the list selector under *Import Public Key*.
3. Click *Browse* and select the public SSH key.
4. Proceed with *Next*.
5. In the *Installation Settings* summary, make sure to check under *Firewall and SSH* the SSH port. Click *open* so it reads *SSH port will be open*.

After the installation is finished, you can log in through SSH using the provided public SSH key.

3.11 Installation Settings

On the last step before the real installation takes place, you can alter installation settings suggested by the installer. To modify the suggestions, click the respective headline. After having made changes to a particular setting, you are always returned to the Installation Settings window, which is updated accordingly.

If you have added an SSH key for your `root` as mentioned in *Procedure 3.1*, make sure to open the SSH port in the *Firewall and SSH* settings.

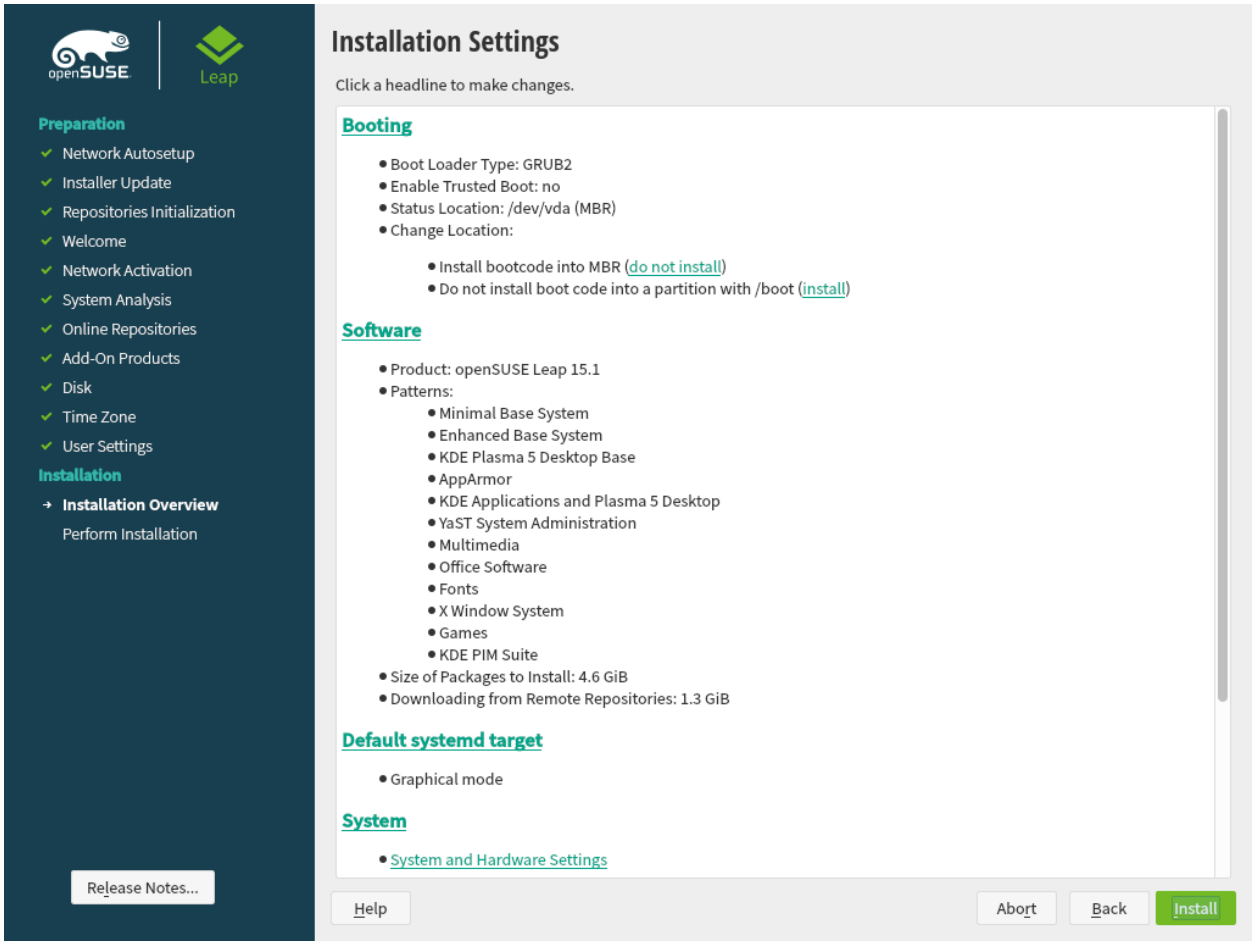


FIGURE 3.8: INSTALLATION SETTINGS

3.11.1 *Software*

openSUSE Leap contains several software patterns for various application purposes. The available choice of patterns and packages depends on your selection of modules and extensions.

Click *Software* to open the *Software Selection and System Tasks* screen where you can modify the pattern selection according to your needs. Select a pattern from the list and see a description in the right-hand part of the window.

Each pattern contains several software packages needed for specific functions (for example Multimedia or Office software). If you chose *Generic Desktop* in the *System Role* dialog choose a desktop environment from the list of available *Graphical Environments*. For a more detailed selection based on software packages to install, select *Details* to switch to the YaST Software Manager.

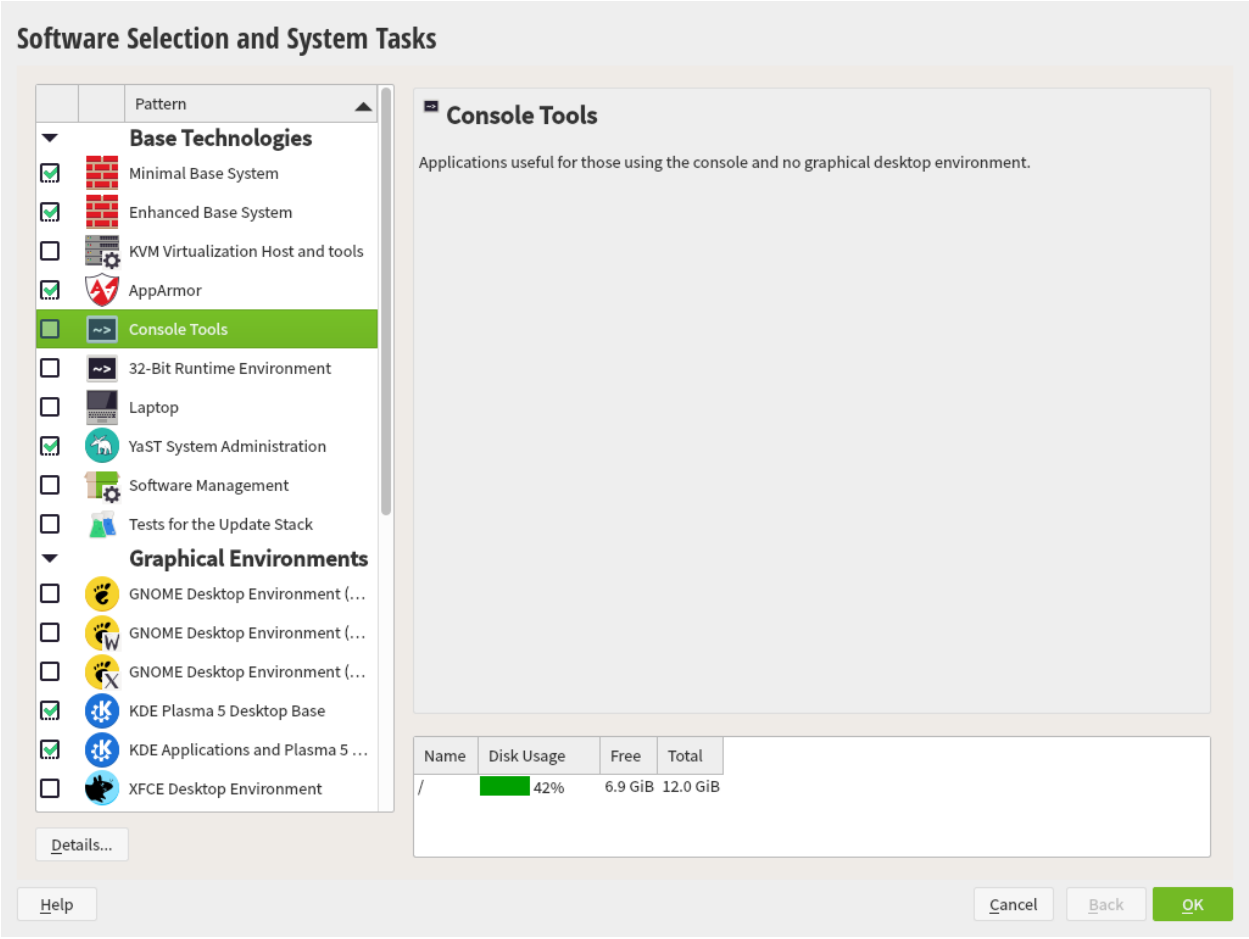


FIGURE 3.9: SOFTWARE SELECTION AND SYSTEM TASKS

You can also install additional software packages or remove software packages from your system at any later time with the YaST Software Manager. For more information, refer to *Chapter 10, Installing or Removing Software*.

By default, openSUSE Leap uses the Wayland display server protocol.

Tip: Adding Secondary Languages

The language you selected with the first step of the installation will be used as the primary (default) language for the system. You can add secondary languages from within the *Software* dialog by choosing *Details* › *View* › *Languages*.

3.11.2 *Booting*

The installer proposes a boot configuration for your system. Other operating systems found on your computer, such as Microsoft Windows or other Linux installations, will automatically be detected and added to the boot loader. However, openSUSE Leap will be booted by default. Normally, you can leave these settings unchanged. If you need a custom setup, modify the proposal according to your needs. For information, see *Book "*Reference*", Chapter 12 "The Boot Loader GRUB 2", Section 12.3 "Configuring the Boot Loader with YaST"*.

Important: Software RAID 1

Booting a configuration where `/boot` resides on a software RAID 1 device is supported, but it requires to install the boot loader into the MBR (*Boot Loader Location* › *Boot from Master Boot Record*). Having `/boot` on software RAID devices with a level other than RAID 1 is not supported.

3.11.3 *Firewall and SSH*

By default `firewalld` is enabled on all configured network interfaces. To globally disable the firewall for this computer, click *Disable* (not recommended).

Note: Firewall Settings

If the firewall is activated, all interfaces are configured to be in the "External Zone", where all ports are closed by default, ensuring maximum security. The only port you can open during the installation is port 22 (SSH), to allow remote access. All other services requiring network access (such as FTP, Samba, Web server, etc.) will only work after having adjusted the firewall settings. Refer to *Book "*Security Guide*", Chapter 16 "Masquerading and Firewalls"* for more information.

To enable remote access via the secure shell (SSH), make sure the `SSH service` is enabled and the `SSH port` is open.

Tip: Existing SSH Host Keys

If you install openSUSE Leap on a machine with existing Linux installations, the installation routine imports an SSH host key. It chooses the host key with the most recent access time by default.

If you are performing a remote administration over VNC, you can also specify whether the machine should be accessible via VNC after the installation. Note that enabling VNC also requires you to set the *Default systemd Target* to *graphical*.

3.11.4 *Default systemd Target*

openSUSE Leap can boot into two different targets (formerly known as “runlevels”). The *graphical* target starts a display manager, whereas the *multi-user* target starts the command line interface.

The default target is *graphical*. In case you have not installed the *X Window System* patterns, you need to change it to *multi-user*. If the system should be accessible via VNC, you need to choose *graphical*.

3.11.5 *Import SSH Host Keys and Configuration*

If an existing Linux installation on your computer was detected, YaST will import the most recent SSH host key found in `/etc/ssh` by default, optionally including other files in the directory as well. This makes it possible to reuse the SSH identity of the existing installation, avoiding the `REMOTE HOST IDENTIFICATION HAS CHANGED` warning on the first connection. Note that this item is not shown in the installation summary if YaST has not discovered any other installations. You have the following choices:

I would like to import SSH keys from a previous install:

Select this option to import the SSH host key and optionally the configuration of an installed system. You can select the installation to import from in the option list below.

Import SSH Configuration

Enable this to copy other files in `/etc/ssh` to the installed system in addition to the host keys.

3.11.6 *System*

This screen lists all the hardware information the installer could obtain about your computer. When opened for the first time, the hardware detection is started. Depending on your system, this may take some time. Select any item in the list and click *Details* to see detailed information about the selected item. Use *Save to File* to save a detailed list to either the local file system or a removable device.

Advanced users can also change the *PCI ID Setup* and kernel settings by choosing *Kernel Settings*. A screen with two tabs opens:

PCI ID Setup

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. You can add PCI IDs to a device driver here. Only advanced users should attempt to do so.

To add an ID, click *Add* and select whether to *Manually* enter the data, or whether to choose from a list. Enter the required data. The *SysFS Dir* is the directory name from `/sys/bus/pci/drivers`—if empty, the *driver* name is used as the directory name. Existing entries can be managed with *Edit* and *Delete*.

Kernel Settings

Change the *Global I/O Scheduler* here. If *Not Configured* is chosen, the default setting for the respective architecture will be used. This setting can also be changed at any time later from the installed system. Refer to *Book "*System Analysis and Tuning Guide*", Chapter 12 "Tuning I/O Performance"* for details on I/O tuning.

Also activate the *Enable SysRq Keys* here. These keys will let you issue basic commands (such as rebooting the system or writing kernel dumps) in case the system crashes. Enabling these keys is recommended when doing kernel development. Refer to https://www.kernel.org/doc/html/latest/admin-guide/sysrq.html for details.

3.12 Performing the Installation

After configuring all installation settings, click *Install* in the Installation Settings window to start the installation. Some software may require a license confirmation. If your software selection includes such software, license confirmation dialogs are displayed. Click *Accept* to install the software package. When not agreeing to the license, click *I Disagree* and the software package will not be installed. In the dialog that follows, confirm with *Install* again.

The installation usually takes between 15 and 30 minutes, depending on the system performance and the selected software scope. After having prepared the hard disk and having saved and restored the user settings, the software installation starts. Choose *Details* to switch to the installation log or *Release Notes* to read important up-to-date information that was not available when the manuals were printed.

After the software installation has completed, the system reboots into the new installation where you can log in. To customize the system configuration or to install additional software packages, start YaST.

4 Troubleshooting

This section highlights some typical problems you may run into during installation and offers possible solutions or workarounds.

4.1 Checking Media

If you encounter any problems using the openSUSE Leap installation media, check the integrity of your installation media. Boot from the media and choose*More* › *Check Installation Media* from the boot menu. A minimal system boots and lets you choose which device to check. Select the respective device and confirm with *OK* to perform the check.

In a running system, start YaST and choose *Software* › *Media Check*. Insert the medium click *Start Check*. Checking may take several minutes.

If errors are detected during the check, do not use this medium for installation. Media problems may, for example, occur when having burned the medium on DVD yourself. Burning the media at a low speed (4x) helps to avoid problems.

4.2 No Bootable DVD Drive Available

If your computer does not contain a built-in bootable DVD drive there are several alternatives. This is also an option if your drive is not supported by openSUSE Leap.

Using External DVD Device

Linux supports most existing DVD drives. If the system has no DVD drive, it is still possible that an external DVD drive, connected through USB, FireWire, or SCSI, can be used to boot the system. Sometimes a BIOS update may help if you encounter problems.

Network Boot via PXE

If a machine lacks a DVD drive, but provides a working Ethernet connection, perform a completely network-based installation.

USB Flash Drive

You can use a USB flash drive if your machine lacks a DVD drive and network connection.

4.3 Booting from Installation Media Fails

One reason a machine does not boot the installation media can be an incorrect boot sequence setting in BIOS. The BIOS boot sequence must have DVD drive 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 in the documentation provided with your mainboard, 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 several 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 one of Del, F1, or Esc. Press this key until the BIOS setup screen appears.

PROCEDURE 4.1: 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* or *BOOT ORDER*. Change the settings by pressing Page ↑ or Page ↓ until the DVD drive is listed first.
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 4.2: CHANGING THE BOOT SEQUENCE IN AN SCSI BIOS (ADAPTEC HOST ADAPTER)

1. Open the setup by pressing Ctrl–A.
2. Select *Disk Utilities*. The connected hardware components are now displayed.
   Make note of the SCSI ID of your DVD 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 DVD 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 shown in the following figure:

FIGURE 4.1: US KEYBOARD LAYOUT

4.4 Boot Failure

Some hardware types, mainly very old or very recent ones, fail to boot. Reasons can be missing support for hardware in the installation kernel or drivers causing problems on some specific hardware.

If your system fails to install using the standard *Installation* mode from the first installation boot screen, try the following:

1. With the DVD still in the drive, reboot the machine with Ctrl–Alt–Del or using the hardware reset button.

2. When the boot screen appears, press F5, use the arrow keys of your keyboard to navigate to *No ACPI* 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 Steps*.

If this fails, proceed as above, but choose *Safe Settings* instead. This option disables ACPI and DMA support. Most hardware will boot with this option.

If both of these options fail, use the boot parameters 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 parameters, 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 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 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.

`pnpacpi=off`

This option is for serial or parallel problems when your BIOS setup contains wrong interrupts or ports.

`notsc`

Disable the time stamp counter. This option can be used to work around timing problems on your systems. It is a recent feature, so if you see regressions on your machine, especially time related or even total hangs, this option is worth a try.

`nohz=off`

Disable the nohz feature. If your machine hangs, this option may help. Otherwise it is of no use.

When 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 inexplicable 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.

4.5 Fails to Launch Graphical Installer

After you insert the medium 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.

PROCEDURE 4.3: CHANGE SCREEN RESOLUTION FOR INSTALLATION

1. Boot for installation.
2. Press F3 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 Steps*.

PROCEDURE 4.4: INSTALLATION IN TEXT MODE

1. Boot for installation.
2. Press F3 and select *Text Mode*.
3. Select *Installation* and proceed with the installation as described in *Chapter 3, Installation Steps*.

PROCEDURE 4.5: VNC INSTALLATION

1. Boot for installation.
2. Enter the following text at the boot parameters prompt:

   ```
   vnc=1 vncpassword=SOME_PASSWORD
   ```

   Replace *SOME_PASSWORD* with the password to use for VNC installation.
3. Select *Installation* then press Enter to start the installation.
   Instead of starting right into the graphical installation routine, the system continues to run in a text mode. The system 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 openSUSE Leap machine and press 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 Steps*.

   Important: Cross-platform Support

   Installation via VNC works with any browser under any operating system, provided Java support is enabled.

   Provide the IP address and password to your VNC viewer when prompted. A window opens, displaying the installation dialogs. Proceed with the installation as usual.

4.6 Only Minimalist Boot Screen Started

You inserted the medium into the drive, the BIOS routines are finished, but the system does not start with the graphical boot screen. Instead it launches a very minimalist text-based interface. This may happen on any machine not providing sufficient graphics memory for rendering a graphical boot screen.

Although the text boot screen looks minimalist, it provides nearly the same functionality as the graphical one:

Boot Options

Unlike the graphical interface, the different boot parameters 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 press `Enter` to launch the boot process.

Custom Boot Options

After selecting a boot parameter, enter the appropriate keyword at the boot prompt or enter some custom boot parameters as described in *Section 4.4, "Boot Failure"*. To launch the installation process, press `Enter`.

Screen Resolutions

Use the function keys (`F1` ... `F12`) to determine the screen resolution for installation. If you need to boot in text mode, choose `F3`.

II Administration

5 Managing Users with YaST

During installation, you could have created a local user for your system. With the YaST module *User and Group Management* you can add users or edit existing ones. It also lets you configure your system to authenticate users with a network server.

5.1 User and Group Administration Dialog

To administer users or groups, start YaST and click *Security and Users* › *User and Group Management*. Alternatively, start the *User and Group Administration* dialog directly by running `sudo yast2 users &` from a command line.

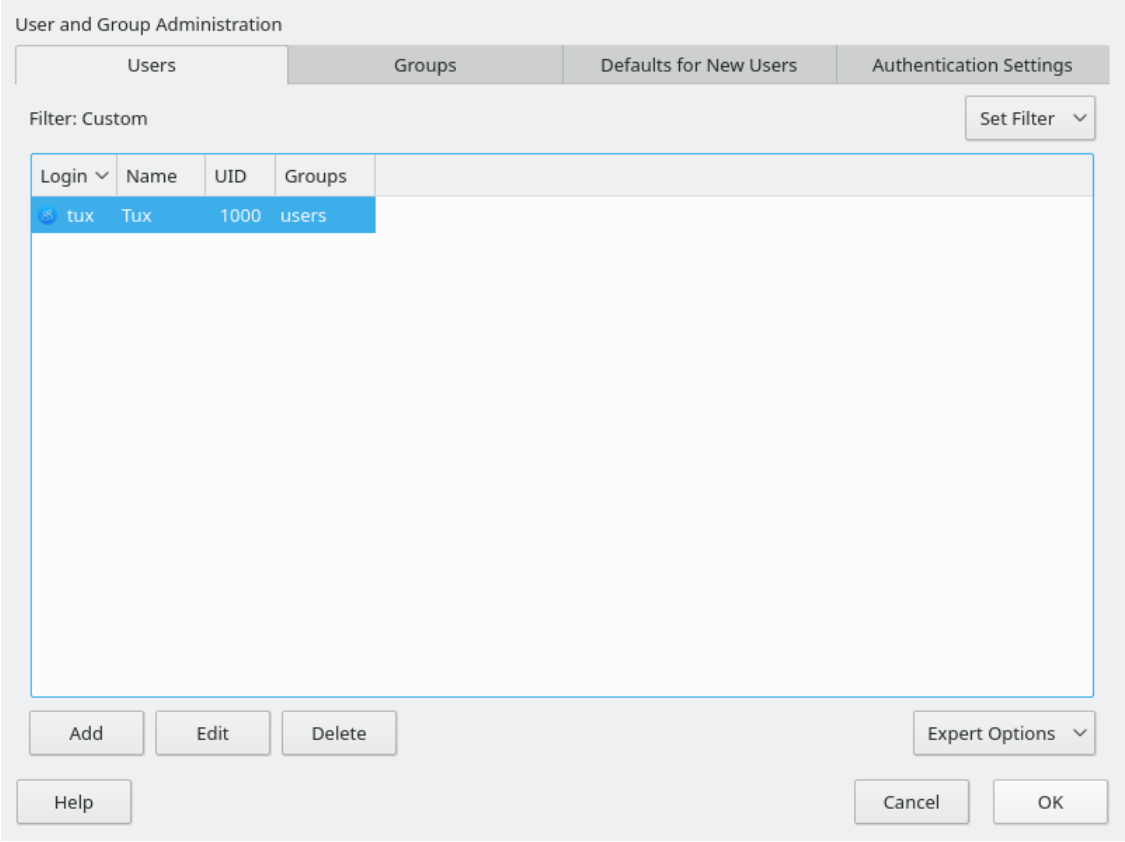


FIGURE 5.1: YAST USER AND GROUP ADMINISTRATION

Every user is assigned a system-wide user ID (UID). Apart from the users which can log in to your machine, there are also several *system users* for internal use only. Each user is assigned to one or more groups. Similar to *system users*, there are also *system groups* for internal use.

Depending on the set of users you choose to view and modify with, the dialog (local users, network users, system users), the main window shows several tabs. These allow you to execute the following tasks:

Managing User Accounts

From the *Users* tab create, modify, delete or temporarily disable user accounts as described in *Section 5.2, "Managing User Accounts"*. Learn about advanced options like enforcing password policies, using encrypted home directories, or managing disk quotas in *Section 5.3, "Additional Options for User Accounts"*.

Changing Default Settings

Local users accounts are created according to the settings defined on the *Defaults for New Users* tab. Learn how to change the default group assignment, or the default path and access permissions for home directories in *Section 5.4, "Changing Default Settings for Local Users"*.

Assigning Users to Groups

Learn how to change the group assignment for individual users in *Section 5.5, "Assigning Users to Groups"*.

Managing Groups

From the *Groups* tab, you can add, modify or delete existing groups. Refer to *Section 5.6, "Managing Groups"* for information on how to do this.

Changing the User Authentication Method

When your machine is connected to a network that provides user authentication methods like NIS or LDAP, you can choose between several authentication methods on the *Authentication Settings* tab. For more information, refer to *Section 5.7, "Changing the User Authentication Method"*.

For user and group management, the dialog provides similar functionality. You can easily switch between the user and group administration view by choosing the appropriate tab at the top of the dialog.

Filter options allow you to define the set of users or groups you want to modify: On the *Users* or *Group* tab, click *Set Filter* to view and edit users or groups. They are listed according to certain categories, such as *Local Users* or *LDAP Users*, if applicable. With *Set Filter* › *Customize Filter* you can also set up and use a custom filter.

Depending on the filter you choose, not all of the following options and functions will be available from the dialog.

5.2 Managing User Accounts

YaST offers to create, modify, delete or temporarily disable user accounts. Do not modify user accounts unless you are an experienced user or administrator.

Note: Changing User IDs of Existing Users

File ownership is bound to the user ID, not to the user name. After a user ID change, the files in the user's home directory are automatically adjusted to reflect this change. However, after an ID change, the user no longer owns the files they created elsewhere in the file system unless the file ownership for those files are manually modified.

In the following, learn how to set up default user accounts. For further options, refer to *Section 5.3, "Additional Options for User Accounts"*.

PROCEDURE 5.1: ADDING OR MODIFYING USER ACCOUNTS

1. Open the YaST *User and Group Administration* dialog and click the *Users* tab.
2. With *Set Filter* define the set of users you want to manage. The dialog lists users in the system and the groups the users belong to.
3. To modify options for an existing user, select an entry and click *Edit*.
   To create a new user account, click *Add*.
4. Enter the appropriate user data on the first tab, such as *Username* (which is used for login) and *Password*. This data is sufficient to create a new user. If you click *OK* now, the system will automatically assign a user ID and set all other values according to the default.
5. Activate *Receive System Mail* if you want any kind of system notifications to be delivered to this user's mailbox. This creates a mail alias for `root` and the user can read the system mail without having to first log in as `root`.
   The mails sent by system services are stored in the local mailbox `/var/spool/mail/` *`USERNAME`*, where *`USERNAME`* is the login name of the selected user. To read e-mails, you can use the **`mail`** command.
6. To adjust further details such as the user ID or the path to the user's home directory, do so on the *Details* tab.
   If you need to relocate the home directory of an existing user, enter the path to the new home directory there and move the contents of the current home directory with *Move to New Location*. Otherwise, a new home directory is created without any of the existing data.

7. To force users to regularly change their password or set other password options, switch to *Password Settings* and adjust the options. For more details, refer to *Section 5.3.2, "Enforcing Password Policies"*.

8. If all options are set according to your wishes, click *OK*.

9. Click *OK* to close the administration dialog and to save the changes. A newly added user can now log in to the system using the login name and password you created.
   Alternatively, to save all changes without exiting the *User and Group Administration* dialog, click *Expert Options* › *Write Changes Now*.

Tip: Matching User IDs

It is useful to match the (local) user ID to the ID in the network. For example, a new (local) user on a laptop should be integrated into a network environment with the same user ID. This ensures that the file ownership of the files the user creates "offline" is the same as if they had created them directly on the network.

PROCEDURE 5.2: DISABLING OR DELETING USER ACCOUNTS

1. Open the YaST *User and Group Administration* dialog and click the *Users* tab.

2. To temporarily disable a user account without deleting it, select the user from the list and click *Edit*. Activate *Disable User Login*. The user cannot log in to your machine until you enable the account again.

3. To delete a user account, select the user from the list and click *Delete*. Choose if you also want to delete the user's home directory or to retain the data.

5.3 Additional Options for User Accounts

In addition to the settings for a default user account, openSUSE® Leap offers further options. For example, options to enforce password policies, use encrypted home directories or define disk quotas for users and groups.

5.3.1 Automatic Login and Passwordless Login

If you use the GNOME desktop environment you can configure *Auto Login* for a certain user and *Passwordless Login* for all users. Auto login causes a user to become automatically logged in to the desktop environment on boot. This functionality can only be activated for one user at a time. Login without password allows all users to log in to the system after they have entered their user name in the login manager.

Warning: Security Risk

Enabling *Auto Login* or *Passwordless Login* on a machine that can be accessed by more than one person is a security risk. Without the need to authenticate, any user can gain access to your system and your data. If your system contains confidential data, do not use this functionality.

to activate auto login or login without password, access these functions in the YaST *User and Group Administration* with *Expert Options* › *Login Settings*.

5.3.2 Enforcing Password Policies

On any system with multiple users, it is a good idea to enforce at least basic password security policies. Users should change their passwords regularly and use strong passwords that cannot easily be exploited. For local users, proceed as follows:

PROCEDURE 5.3: CONFIGURING PASSWORD SETTINGS

1. Open the YaST *User and Group Administration* dialog and select the *Users* tab.
2. Select the user for which to change the password options and click *Edit*.
3. Switch to the *Password Settings* tab. The user's last password change is displayed on the tab.
4. To make the user change their password at next login, activate *Force Password Change*.
5. To enforce password rotation, set a *Maximum Number of Days for the Same Password* and a *Minimum Number of Days for the Same Password*.
6. To remind the user to change their password before it expires, set the number of *Days before Password Expiration to Issue Warning*.

7. To restrict the period of time the user can log in after their password has expired, change the value in *Days after Password Expires with Usable Login*.
8. You can also specify a certain expiration date for the complete account. Enter the *Expiration Date* in *YYYY-MM-DD* format. Note that this setting is not password-related but rather applies to the account itself.
9. For more information about the options and about the default values, click *Help*.
10. Apply your changes with *OK*.

5.3.3 Managing Quotas

To prevent system capacities from being exhausted without notification, system administrators can set up quotas for users or groups. Quotas can be defined for one or more file systems and restrict the amount of disk space that can be used and the number of inodes (index nodes) that can be created there. Inodes are data structures on a file system that store basic information about a regular file, directory, or other file system object. They store all attributes of a file system object (like user and group ownership, read, write, or execute permissions), except file name and contents.

openSUSE Leap allows usage of `soft` and `hard` quotas. Additionally, grace intervals can be defined that allow users or groups to temporarily violate their quotas by certain amounts.

Soft Quota

Defines a warning level at which users are informed that they are nearing their limit. Administrators will urge the users to clean up and reduce their data on the partition. The soft quota limit is usually lower than the hard quota limit.

Hard Quota

Defines the limit at which write requests are denied. When the hard quota is reached, no more data can be stored and applications may crash.

Grace Period

Defines the time between the overflow of the soft quota and a warning being issued. Usually set to a rather low value of one or several hours.

PROCEDURE 5.4: ENABLING QUOTA SUPPORT FOR A PARTITION

To configure quotas for certain users and groups, you need to enable quota support for the respective partition in the YaST Expert Partitioner first.

1. In YaST, select *System* › *Partitioner* and click *Yes* to proceed.
2. In the *Expert Partitioner*, select the partition for which to enable quotas and click *Edit*.
3. Click *Fstab Options* and activate *Enable Quota Support*. If the `quota` package is not already installed, it will be installed when you confirm the respective message with *Yes*.
4. Confirm your changes and leave the *Expert Partitioner*.
5. Make sure the service `quotaon` is running by entering the following command:

   ```
   tux > sudo systemctl status quotaon
   ```

   It should be marked as being `active`. If this is not the case, start it with the command **`systemctl start quotaon`**.

PROCEDURE 5.5: SETTING UP QUOTAS FOR USERS OR GROUPS

Now you can define soft or hard quotas for specific users or groups and set time periods as grace intervals.

1. In the YaST *User and Group Administration*, select the user or the group you want to set the quotas for and click *Edit*.
2. On the *Plug-Ins* tab, select the *Manage User Quota* entry and click *Launch* to open the *Quota Configuration* dialog.
3. From *File System*, select the partition to which the quota should apply.
4. Below *Size Limits*, restrict the amount of disk space. Enter the number of 1 KB blocks the user or group may have on this partition. Specify a *Soft Limit* and a *Hard Limit* value.
5. Additionally, you can restrict the number of inodes the user or group may have on the partition. Below *Inodes Limits*, enter a *Soft Limit* and *Hard Limit*.
6. You can only define grace intervals if the user or group has already exceeded the soft limit specified for size or inodes. Otherwise, the time-related text boxes are not activated. Specify the time period for which the user or group is allowed to exceed the limits set above.
7. Confirm your settings with *OK*.
8. Click *OK* to close the administration dialog and save the changes.
   Alternatively, to save all changes without exiting the *User and Group Administration* dialog, click *Expert Options* › *Write Changes Now*.

openSUSE Leap also ships command line tools like `repquota` or `warnquota`. System administrators can use these tools to control the disk usage or send e-mail notifications to users exceeding their quota. Using **`quota_nld`**, administrators can also forward kernel messages about exceeded quotas to D-BUS. For more information, refer to the `repquota`, the `warnquota` and the **`quota_nld`** man page.

5.4 Changing Default Settings for Local Users

When creating new local users, several default settings are used by YaST. These include, for example, the primary group and the secondary groups the user belongs to, or the access permissions of the user's home directory. You can change these default settings to meet your requirements:

1. Open the YaST *User and Group Administration* dialog and select the *Defaults for New Users* tab.
2. To change the primary group the new users should automatically belong to, select another group from *Default Group*.
3. To modify the secondary groups for new users, add or change groups in *Secondary Groups*. The group names must be separated by commas.
4. If you do not want to use `/home/USERNAME` as default path for new users' home directories, modify the *Path Prefix for Home Directory*.
5. To change the default permission modes for newly created home directories, adjust the umask value in *Umask for Home Directory*. For more information about umask, refer to *Book "*Security Guide*", Chapter 10 "Access Control Lists in Linux"* and to the **`umask`** man page.
6. For information about the individual options, click *Help*.
7. Apply your changes with *OK*.

5.5 Assigning Users to Groups

Local users are assigned to several groups according to the default settings which you can access from the *User and Group Administration* dialog on the *Defaults for New Users* tab. In the following, learn how to modify an individual user's group assignment. If you need to change the default group assignments for new users, refer to *Section 5.4, "Changing Default Settings for Local Users"*.

PROCEDURE 5.6: CHANGING A USER'S GROUP ASSIGNMENT

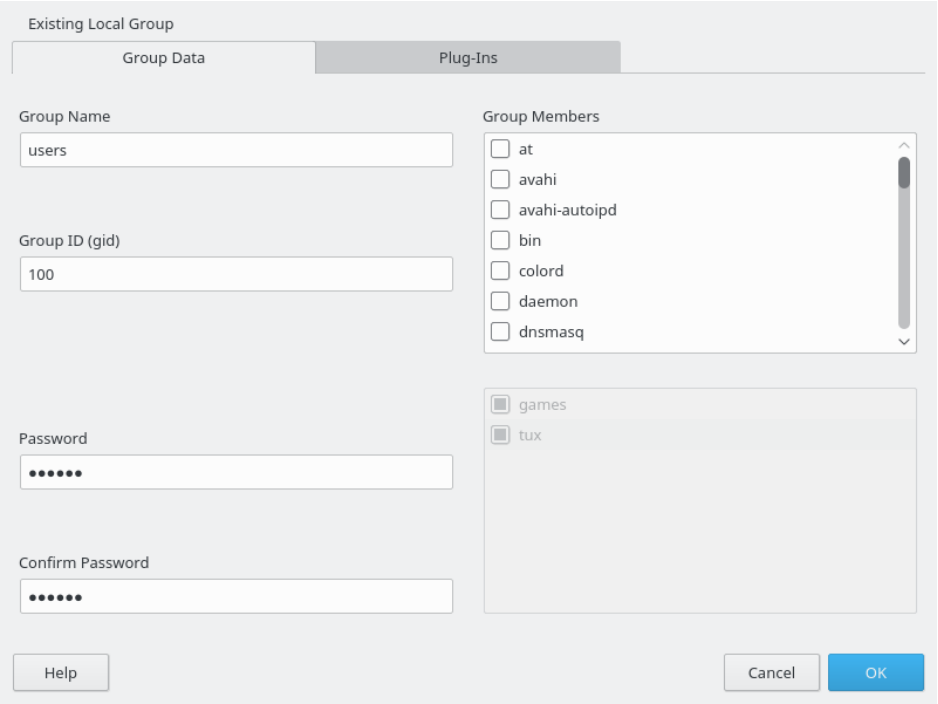
1. Open the YaST *User and Group Administration* dialog and click the *Users* tab. It lists users and the groups the users belong to.
2. Click *Edit* and switch to the *Details* tab.
3. To change the primary group the user belongs to, click *Default Group* and select the group from the list.
4. To assign the user additional secondary groups, activate the corresponding check boxes in the *Additional Groups* list.
5. Click *OK* to apply your changes.
6. Click *OK* to close the administration dialog and save the changes.
   Alternatively, to save all changes without exiting the *User and Group Administration* dialog, click *Expert Options* › *Write Changes Now*.

5.6 Managing Groups

With YaST you can also easily add, modify or delete groups.

PROCEDURE 5.7: CREATING AND MODIFYING GROUPS

1. Open the YaST *User and Group Management* dialog and click the *Groups* tab.
2. With *Set Filter* define the set of groups you want to manage. The dialog lists groups in the system.
3. To create a new group, click *Add*.
4. To modify an existing group, select the group and click *Edit*.
5. In the following dialog, enter or change the data. The list on the right shows an overview of all available users and system users which can be members of the group.



6. To add existing users to a new group select them from the list of possible *Group Members* by checking the corresponding box. To remove them from the group deactivate the box.

7. Click *OK* to apply your changes.

8. Click *OK* to close the administration dialog and save the changes.
   Alternatively, to save all changes without exiting the *User and Group Administration* dialog, click *Expert Options* › *Write Changes Now*.

To delete a group, it must not contain any group members. To delete a group, select it from the list and click *Delete*. Click *OK* to close the administration dialog and save the changes. Alternatively, to save all changes without exiting the *User and Group Administration* dialog, click *Expert Options* › *Write Changes Now*.

5.7 Changing the User Authentication Method

When your machine is connected to a network, you can change the authentication method. The following options are available:

NIS

Users are administered centrally on a NIS server for all systems in the network. For details, see *Book "*Security Guide*", Chapter 3 "Using NIS"*.

SSSD

The *System Security Services Daemon* (SSSD) can locally cache user data and then allow users to use the data, even if the real directory service is (temporarily) unreachable. For details, see *Book "*Security Guide*", Chapter 4 "Setting Up Authentication Servers and Clients Using YaST", Section 4.3 "SSSD"*.

Samba

SMB authentication is often used in mixed Linux and Windows networks. For details, see *Book "*Reference*", Chapter 21 "Samba"* and *Book "*Security Guide*", Chapter 7 "Active Directory Support"*.

To change the authentication method, proceed as follows:

1. Open the *User and Group Administration* dialog in YaST.
2. Click the *Authentication Settings* tab to show an overview of the available authentication methods and the current settings.
3. To change the authentication method, click *Configure* and select the authentication method you want to modify. This takes you directly to the client configuration modules in YaST. For information about the configuration of the appropriate client, refer to the following sections:

   NIS: *Book "*Security Guide*", Chapter 3 "Using NIS", Section 3.2 "Configuring NIS Clients"*

   LDAP: *Book "*Security Guide*", Chapter 4 "Setting Up Authentication Servers and Clients Using YaST", Section 4.2 "Configuring an Authentication Client with YaST"*

   Samba: *Book "*Reference*", Chapter 21 "Samba", Section 21.5.1 "Configuring a Samba Client with YaST"*

   SSSD: *Book "*Security Guide*", Chapter 4 "Setting Up Authentication Servers and Clients Using YaST", Section 4.3 "SSSD"*
4. After accepting the configuration, return to the *User and Group Administration* overview.
5. Click *OK* to close the administration dialog.

5.8 Default System Users

By default, openSUSE Leap creates user names which cannot be deleted. These users are typically defined in the Linux Standard Base. The following list provides the common user names and their purpose:

COMMON USER NAMES INSTALLED BY DEFAULT

`bin`,
`daemon`
: Legacy user, included for compatibility with legacy applications. New applications should no longer use this user name.

`gdm`
: Used by GNOME Display Manager (GDM) to provide graphical logins and manage local and remote displays.

`lp`
: Used by the Printer daemon for Common Unix Printing System (CUPS).

`mail`
: User reserved for mailer programs like **`sendmail`** or **`postfix`**.

`man`
: Used by man to access man pages.

`messagebus`
: Used to access D-Bus (desktop bus), a software bus for inter-process communication. Daemon is `dbus-daemon`.

`nobody`
: User that owns no files and is in no privileged groups. Nowadays, its use is limited as it is recommended by Linux Standard Base to provide a separate user account for each daemon.

`nscd`
: Used by the Name Service Caching Daemon. This daemon is a lookup service to improve performance with NIS and LDAP. Daemon is `nscd`.

`polkitd`
: Used by the PolicyKit Authorization Framework which defines and handles authorization requests for unprivileged processes. Daemon is `polkitd`.

`postfix`
: Used by the Postfix mailer.

`pulse`
: Used by the Pulseaudio sound server.

`root`
: Used by the system administrator, providing all appropriate privileges.

`rpc`
: Used by the **`rpcbind`** command, an RPC port mapper.

`rtkit`
: Used by the `rtkit` package providing a D-Bus system service for real time scheduling mode.

`salt`
: User for parallel remote execution provided by Salt. Daemon is named `salt-master`.

`scard`
: User for communication with smart cards and readers. Daemon is named `pcscd`.

`srvGeoClue`
: Used by the GeoClue D-Bus service to provide location information.

`sshd`
: Used by the Secure Shell daemon (SSH) to ensure secured and encrypted communication over an insecure network.

`statd`
: Used by the Network Status Monitor protocol (NSM), implemented in the `rpc.statd` daemon, to listen for reboot notifications.

`systemd-coredump`
: Used by the **`/usr/lib/systemd/systemd-coredump`** command to acquire, save and process core dumps.

`systemd-network`
: Used by the **`/usr/lib/systemd/systemd-networkd`** command to manage networks.

`systemd-timesync`
: Used by the **`/usr/lib/systemd/systemd-timesyncd`** command to synchronize the local system clock with a remote Network Time Protocol (NTP) server.

6 Changing Language and Country Settings with YaST

Working in different countries or having to work in a multilingual environment requires your computer to be set up to support this. openSUSE® Leap can handle different `locales` in parallel. A locale is a set of parameters that defines the language and country settings reflected in the user interface.

The main system language was selected during installation and keyboard and time zone settings were adjusted. However, you can install additional languages on your system and determine which of the installed languages should be the default.

For those tasks, use the YaST language module as described in *Section 6.1, "Changing the System Language"*. Install secondary languages to get optional localization if you need to start applications or desktops in languages other than the primary one.

Apart from that, the YaST timezone module allows you to adjust your country and timezone settings accordingly. It also lets you synchronize your system clock against a time server. For details, refer to *Section 6.2, "Changing the Country and Time Settings"*.

6.1 Changing the System Language

Depending on how you use your desktop and whether you want to switch the entire system to another language or only the desktop environment itself, there are several ways to do this:

Changing the System Language Globally

Proceed as described in *Section 6.1.1, "Modifying System Languages with YaST"* and *Section 6.1.2, "Switching the Default System Language"* to install additional localized packages with YaST and to set the default language. Changes are effective after the next login. To ensure that the entire system reflects the change, reboot the system or close and restart all running services, applications, and programs.

Changing the Language for the Desktop Only

Provided you have previously installed the desired language packages for your desktop environment with YaST as described below, you can switch the language of your desktop using the desktop's control center. Refer to *Book "GNOME User Guide", Chapter 3 "Customizing Your Settings", Section 3.2 "Configuring Language Settings"* for details. After the X server has been restarted, your entire desktop reflects your new choice of language. Applications not belonging to your desktop framework are not affected by this change and may still appear in the language that was set in YaST.

Temporarily Switching Languages for One Application Only

You can also run a single application in another language (that has already been installed with YaST). To do so, start it from the command line by specifying the language code as described in *Section 6.1.3, "Switching Languages for Standard X and GNOME Applications"*.

6.1.1 Modifying System Languages with YaST

YaST knows two different language categories:

Primary Language

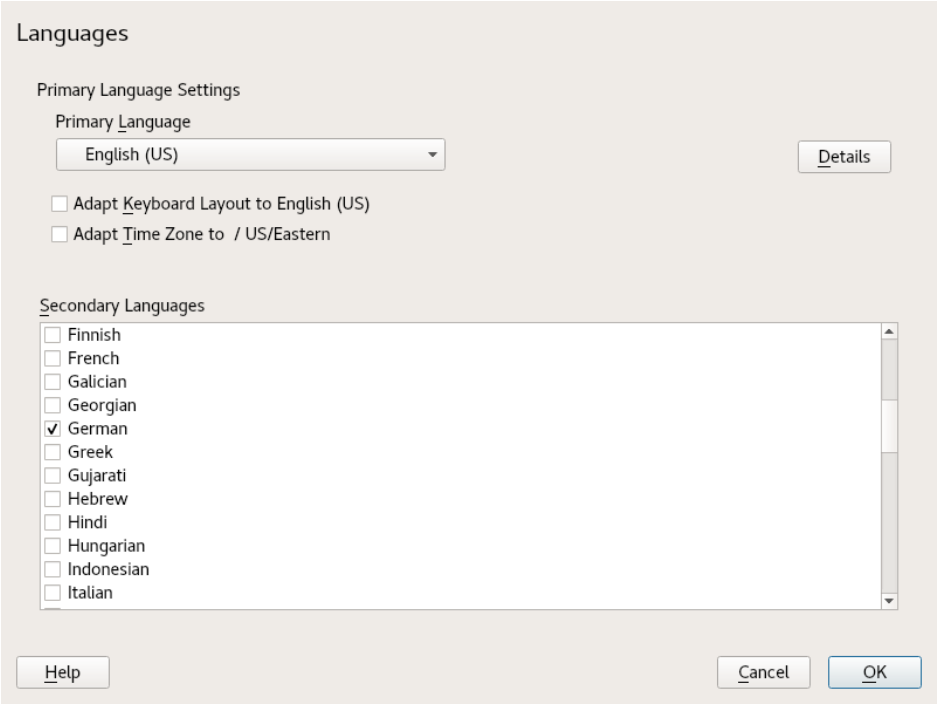
The primary language set in YaST applies to the entire system, including YaST and the desktop environment. This language is used whenever available unless you manually specify another language.

Secondary Languages

Install secondary languages to make your system multilingual. Languages installed as secondary languages can be selected manually for a specific situation. For example, use a secondary language to start an application in a certain language to do word processing in this language.

Before installing additional languages, determine which of them should be the default system language (primary language).

To access the YaST language module, start YaST and click *System* › *Language*. Alternatively, start the *Languages* dialog directly by running `sudo yast2 language &` from a command line.



PROCEDURE 6.1: INSTALLING ADDITIONAL LANGUAGES

When installing additional languages, YaST also allows you to set different locale settings for the user `root`, see *Step 4*. The option *Locale Settings for User root* determines how the locale variables (`LC_*`) in the file `/etc/sysconfig/language` are set for `root`. You can set them to the same locale as for normal users. Alternatively, you can keep it unaffected by any language changes, or only set the variable `RC_LC_CTYPE` to the same values as for the normal users. The `RC_LC_CTYPE` variable sets the localization for language-specific function calls.

1. To add languages in the YaST language module, select the *Secondary Languages* you want to install.
2. To make a language the default language, set it as *Primary Language*.
3. Additionally, adapt the keyboard to the new primary language and adjust the time zone, if appropriate.

Tip: Advanced Settings

For advanced keyboard or time zone settings, select *Hardware* › *System Keyboard Layout* or *System* › *Date and Time* in YaST to start the respective dialogs. For more information, refer to *Section 7.1, "Setting Up Your System Keyboard Layout"* and *Section 6.2, "Changing the Country and Time Settings"*.

4. To change language settings specific to the user `root`, click *Details*.

   a. Set *Locale Settings for User root* to the desired value. For more information, click *Help*.

   b. Decide if you want to *Use UTF-8 Encoding* for `root` or not.

5. If your locale was not included in the list of primary languages available, try specifying it with *Detailed Locale Setting*. However, some localization may be incomplete.

6. Confirm your changes in the dialogs with *OK*. If you have selected secondary languages, YaST installs the localized software packages for the additional languages.

The system is now multilingual. However, to start an application in a language other than the primary one, you need to set the desired language explicitly as explained in *Section 6.1.3, "Switching Languages for Standard X and GNOME Applications"*.

6.1.2 Switching the Default System Language

To globally change the default language of a system, use the following procedure:

1. Start the YaST language module.

2. Select the desired new system language as *Primary Language*.

Important: Deleting Former System Languages

If you switch to a different primary language, the localized software packages for the former primary language will be removed from the system. To switch the default system language but keep the former primary language as additional language, add it as *Secondary Language* by enabling the respective check box.

3. Adjust the keyboard and time zone options as desired.

4. Confirm your changes with *OK*.

5. After YaST has applied the changes, restart current X sessions (for example, by logging out and logging in again) to make YaST and the desktop applications reflect your new language settings.

6.1.3 Switching Languages for Standard X and GNOME Applications

After you have installed the respective language with YaST, you can run a single application in another language.

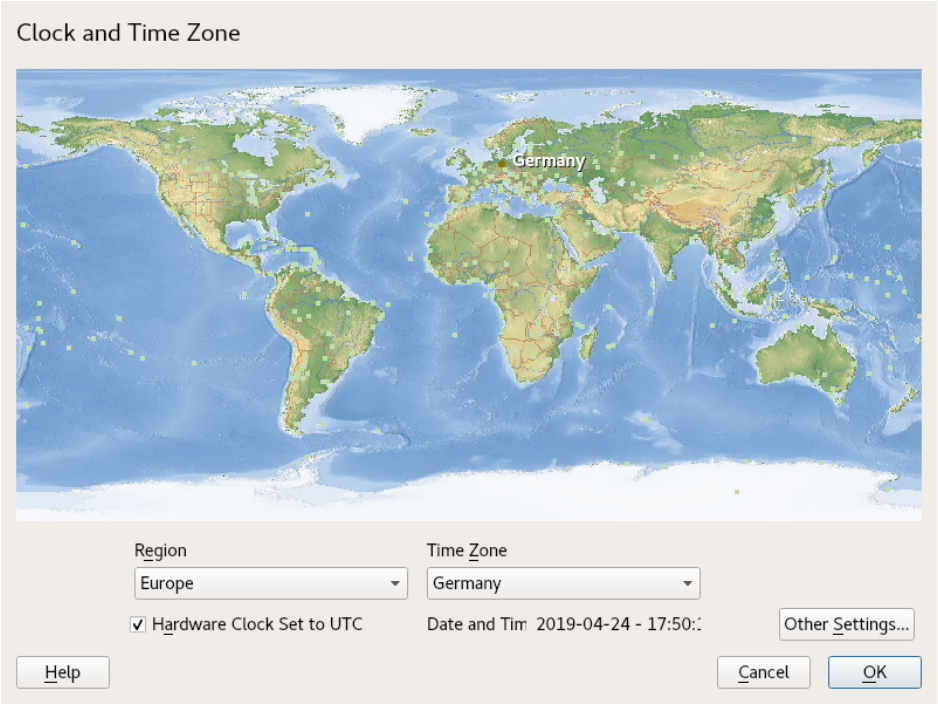
Start the application from the command line by using the following command:

```
LANG=LANGUAGE application
```

For example, to start f-spot in German, run **`LANG=de_DE f-spot`**. For other languages, use the appropriate language code. Get a list of all language codes available with the **`locale -av`** command.

6.2 Changing the Country and Time Settings

Using the YaST date and time module, adjust your system date, clock and time zone information to the area you are working in. To access the YaST module, start YaST and click *System* › *Date and Time*. Alternatively, start the *Clock and Time Zone* dialog directly by running **`sudo yast2 timezone &`** from a command line.



First, select a general region, such as *Europe*. Choose an appropriate country that matches the one you are working in, for example, *Germany*.

Depending on which operating systems run on your workstation, adjust the hardware clock settings accordingly:

- If you run another operating system on your machine, such as Microsoft Windows*, it is likely your system does not use UTC, but local time. In this case, deactivate *Hardware Clock Set To UTC*.
- If you only run Linux on your machine, set the hardware clock to UTC and have the switch from standard time to daylight saving time performed automatically.

Important: Set the Hardware Clock to UTC

The switch from standard time to daylight saving time (and vice versa) can only be performed automatically when the hardware clock (CMOS clock) is set to UTC. This also applies if you use automatic time synchronization with NTP, because automatic synchronization will only be performed if the time difference between the hardware and system clock is less than 15 minutes.

Since a wrong system time can cause serious problems (missed backups, dropped mail messages, mount failures on remote file systems, etc.) it is strongly recommended to *always* set the hardware clock to UTC.

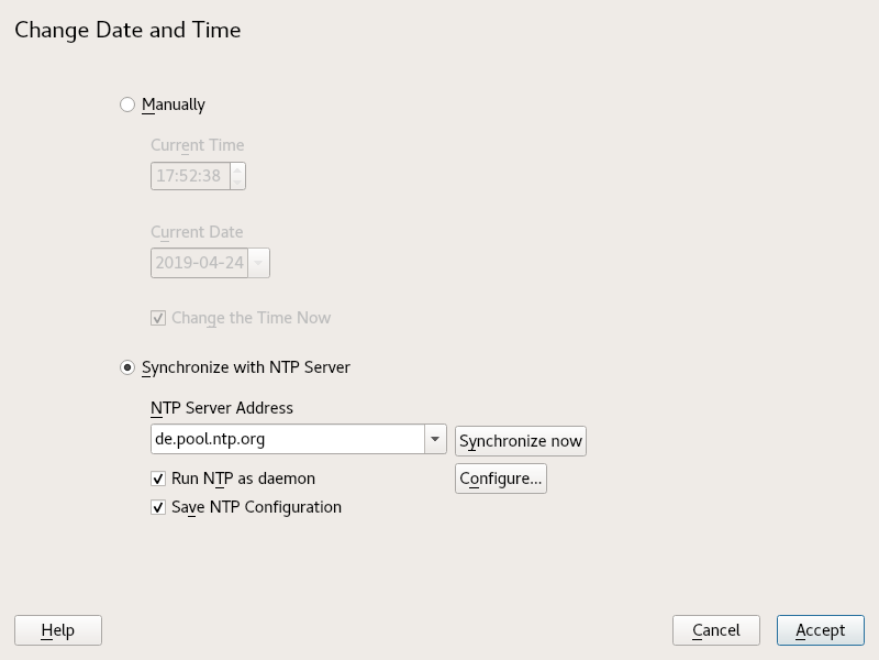
You can change the date and time manually or opt for synchronizing your machine against an NTP server, either permanently or only for adjusting your hardware clock.

PROCEDURE 6.2: MANUALLY ADJUSTING TIME AND DATE

1. In the YaST timezone module, click *Other Settings* to set date and time.
2. Select *Manually* and enter date and time values.
3. Confirm your changes.

PROCEDURE 6.3: SETTING DATE AND TIME WITH NTP SERVER

1. Click *Other Settings* to set date and time.
2. Select *Synchronize with NTP Server*.
3. Enter the address of an NTP server, if not already populated.



4. Click *Synchronize Now* to get your system time set correctly.
5. To use NTP permanently, enable *Save NTP Configuration*.

6. With the *Configure* button, you can open the advanced NTP configuration. For details, see *Book "*Reference*", Chapter 18 "Time Synchronization with NTP", Section 18.1 "Configuring an NTP Client with YaST"*.

7. Confirm your changes.

7 Setting Up Hardware Components with YaST

YaST allows you to configure hardware items such as audio hardware, your system keyboard layout or printers.

Note: Graphics Card, Monitor, Mouse and Keyboard Settings

Graphics card, monitor, mouse and keyboard can be configured with GNOME tools. See *Book "*GNOME User Guide*", Chapter 3 "Customizing Your Settings"* for details.

7.1 Setting Up Your System Keyboard Layout

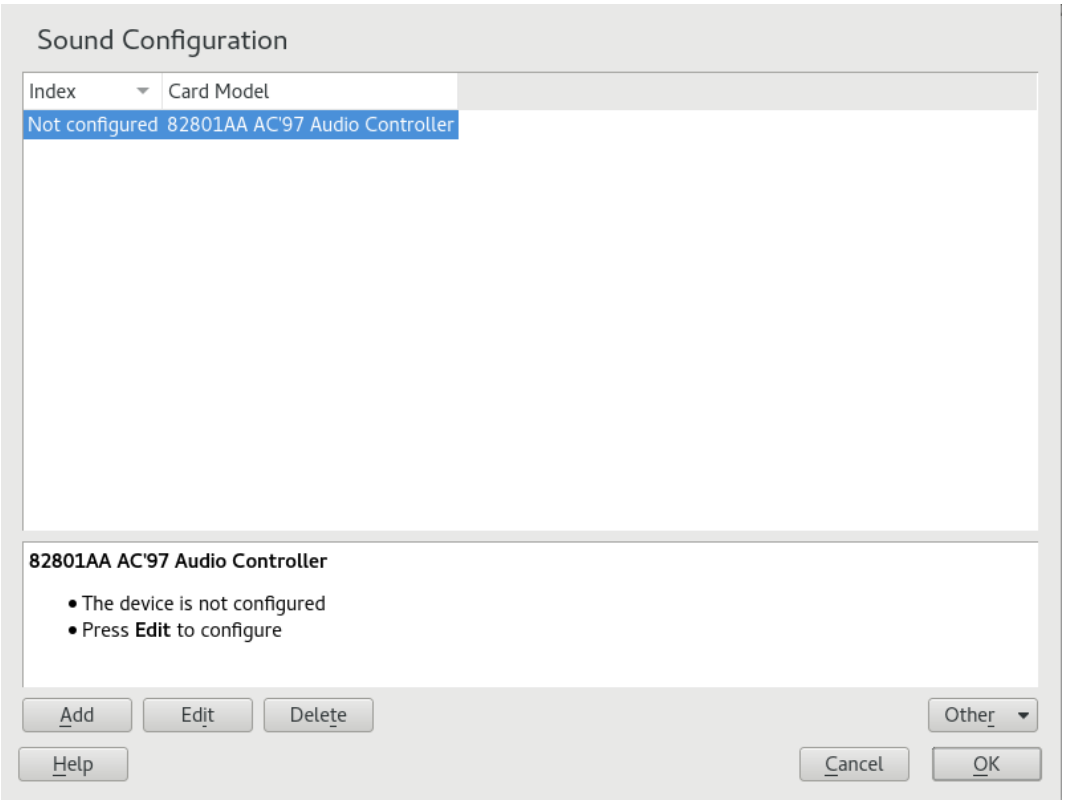
The YaST *System Keyboard Layout* module lets you define the default keyboard layout for the system (also used for the console). Users can modify the keyboard layout in their individual X sessions, using the desktop's tools.

1. Start the YaST *System Keyboard Configuration* dialog by clicking *Hardware* › *System Keyboard Layout* in YaST. Alternatively, start the module from the command line with `sudo yast2 keyboard`.
2. Select the desired *Keyboard Layout* from the list.
3. Optionally, you can also define the keyboard repeat rate or keyboard delay rate in the *Expert Settings*.
4. Try the selected settings in the *Test* text box.
5. If the result is as expected, confirm your changes and close the dialog. The settings are written to `/etc/sysconfig/keyboard`.

7.2 Setting Up Sound Cards

YaST detects most sound cards automatically and configures them with the appropriate values. To change the default settings, or to set up a sound card that could not be configured automatically, use the YaST sound module. There, you can also set up additional sound cards or switch their order.

To start the sound module, start YaST and click *Hardware* › *Sound*. Alternatively, start the *Sound Configuration* dialog directly by running `yast2 sound &` as user `root` from a command line.



The dialog shows all sound cards that were detected.

PROCEDURE 7.1: CONFIGURING SOUND CARDS

If you have added a new sound card or YaST could not automatically configure an existing sound card, follow the steps below. For configuring a new sound card, you need to know your sound card vendor and model. If in doubt, refer to your sound card documentation for the required information. For a reference list of sound cards supported by ALSA with their corresponding sound modules, see http://www.alsa-project.org/main/index.php/Matrix:Main.

During configuration, you can choose between the following setup options:

Quick Automatic Setup

You are not required to go through any of the further configuration steps—the sound card is configured automatically. You can set the volume or any options you want to change later.

Normal Setup

Allows you to adjust the output volume and play a test sound during the configuration.

Advanced setup with possibility to change options

For experts only. Allows you to customize all parameters of the sound card.

Important: Advanced Configuration

Only use this option if you know exactly what you are doing. Otherwise leave the parameters untouched and use the normal or the automatic setup options.

1. Start the YaST sound module.
2. To configure a detected, but *Not Configured* sound card, select the respective entry from the list and click *Edit*.
   To configure a new sound card, click *Add*. Select your sound card vendor and model and click *Next*.
3. Choose one of the setup options and click *Next*.
4. If you have chosen *Normal Setup*, you can now *Test* your sound configuration and make adjustments to the volume. You should start at about ten percent volume to avoid damage to your hearing or the speakers.
5. If all options are set according to your wishes, click *Next*.
   The *Sound Configuration* dialog shows the newly configured or modified sound card.
6. To remove a sound card configuration that you no longer need, select the respective entry and click *Delete*.
7. Click *OK* to save the changes and leave the YaST sound module.

PROCEDURE 7.2: MODIFYING SOUND CARD CONFIGURATIONS

1. To change the configuration of an individual sound card (for experts only!), select the sound card entry in the *Sound Configuration* dialog and click *Edit*.
   This takes you to the *Sound Card Advanced Options* where you can fine-tune several parameters. For more information, click *Help*.
2. To adjust the volume of an already configured sound card or to test the sound card, select the sound card entry in the *Sound Configuration* dialog and click *Other*. Select the respective menu item.

Note: YaST Mixer

The YaST mixer settings provide only basic options. They are intended for troubleshooting (for example, if the test sound is not audible). Access the YaST mixer settings from *Other* › *Volume*. For everyday use and fine-tuning of sound options, use the mixer applet provided by your desktop or the **`alsasound`** command line tool.

3. For playback of MIDI files, select *Other* › *Start Sequencer*.
4. When a supported sound card is detected, you can install SoundFonts for playback of MIDI files:
    a. Insert the original driver CD-ROM into your CD or DVD drive.
    b. Select *Other* › *Install SoundFonts* to copy SF2 SoundFonts™ to your hard disk. The SoundFonts are saved in the directory `/usr/share/sfbank/creative/`.
5. If you have configured more than one sound card in your system you can adjust the order of your sound cards. To set a sound card as primary device, select the sound card in the *Sound Configuration* and click *Other* › *Set as the Primary Card*. The sound device with index `0` is the default device and thus used by the system and the applications.
6. By default, openSUSE Leap uses the PulseAudio sound system. This is an abstraction layer that helps to mix multiple audio streams, bypassing any restrictions the hardware may have. To enable or disable the PulseAudio sound system, click *Other* › *PulseAudio Configuration*. If enabled, PulseAudio daemon is used to play sounds. Disable *PulseAudio Support* to use something else system-wide.

The volume and configuration of all sound cards are saved when you click *OK* and leave the YaST sound module. The mixer settings are saved to the file `/etc/asound.state`. The ALSA configuration data is appended to the end of the file `/etc/modprobe.d/sound` and written to `/etc/sysconfig/sound`.

7.3 Setting Up a Printer

YaST can be used to configure a local printer connected to your machine via USB and to set up printing with network printers. It is also possible to share printers over the network. Further information about printing (general information, technical details, and troubleshooting) is available in *Chapter 8, Printer Operation*.

In YaST, click *Hardware* › *Printer* to start the printer module. By default it opens in the *Printer Configurations* view, displaying a list of all printers that are available and configured. This is especially useful when having access to a lot of printers via the network. From here you can also *Print a Test Page* and configure printers.

Note: Starting CUPS

To print from your system, CUPS must be running. In case it is not running, you are asked to start it. Answer with *Yes*, or you cannot configure printing. In case CUPS is not started at boot time, you will also be asked to enable this feature. It is recommended to say *Yes*, otherwise CUPS would need to be started manually after each reboot.

7.3.1 Configuring Printers

Usually a USB printer is automatically detected. There are two possible reasons it is not automatically detected:

- The USB printer is switched off.
- Communication between printer and computer is not possible. Check the cable and the plugs to make sure that the printer is properly connected. If this is the case, the problem may not be printer-related, but rather a USB-related problem.

Configuring a printer is a three-step process: specify the connection type, choose a driver, and name the print queue for this setup.

For many printer models, several drivers are available. When configuring the printer, YaST defaults to those marked `recommended` as a general rule. Normally it is not necessary to change the driver. However, if you want a color printer to print only in black and white, you can use a driver that does not support color printing. If you experience performance problems with a PostScript printer when printing graphics, try to switch from a PostScript driver to a PCL driver (provided your printer understands PCL).

If no driver for your printer is listed, try to select a generic driver with an appropriate standard language from the list. Refer to your printer's documentation to find out which language (the set of commands controlling the printer) your printer understands. If this does not work, refer to *Section 7.3.1.1, "Adding Drivers with YaST"* for another possible solution.

A printer is never used directly, but always through a print queue. This ensures that simultaneous jobs can be queued and processed one after the other. Each print queue is assigned to a specific driver, and a printer can have multiple queues. This makes it possible to set up a second queue on a color printer that prints black and white only, for example. Refer to *Section 8.1, "The CUPS Workflow"* for more information about print queues.

PROCEDURE 7.3: ADDING A NEW PRINTER

1. Start the YaST printer module with *Hardware* › *Printer*.
2. In the *Printer Configurations* screen click *Add*.
3. If your printer is already listed under `Specify the Connection`, proceed with the next step. Otherwise, try to *Detect More* or start the *Connection Wizard*.
4. In the text box under `Find and Assign a Driver` enter the vendor name and the model name and click *Search for*.
5. Choose a driver that matches your printer. It is recommended to choose the driver listed first. If no suitable driver is displayed:
    a. Check your search term.
    b. Broaden your search by clicking *Find More*.
    c. Add a driver as described in *Section 7.3.1.1, "Adding Drivers with YaST"*.
6. Specify the `Default paper size`.
7. In the *Set Arbitrary Name* field, enter a unique name for the print queue.
8. The printer is now configured with the default settings and ready to use. Click *OK* to return to the *Printer Configurations* view. The newly configured printer is now visible in the list of printers.

7.3.1.1 Adding Drivers with YaST

Not all printer drivers available for openSUSE Leap are installed by default. If no suitable driver is available in the *Find and Assign a Driver* dialog when adding a new printer install a driver package containing drivers for your printers:

PROCEDURE 7.4: INSTALLING ADDITIONAL DRIVER PACKAGES

1. Start the YaST printer module with *Hardware* › *Printer*.
2. In the *Printer Configurations* screen, click *Add*.
3. In the `Find and Assign a Driver` section, click *Driver Packages*.
4. Choose one or more suitable driver packages from the list. Do *not* specify the path to a printer description file.
5. Choose *OK* and confirm the package installation.
6. To directly use these drivers, proceed as described in *Procedure 7.3, "Adding a New Printer"*.

PostScript printers do not need printer driver software. PostScript printers need only a PostScript Printer Description (PPD) file which matches the particular model. PPD files are provided by the printer manufacturer.

If no suitable PPD file is available in the *Find and Assign a Driver* dialog when adding a PostScript printer, install a PPD file for your printer:

Several sources for PPD files are available. It is recommended to first try additional driver packages that are shipped with openSUSE Leap but not installed by default (see below for installation instructions). If these packages do not contain suitable drivers for your printer, get PPD files directly from your printer vendor or from the driver CD of a PostScript printer. For details, see *Section 8.8.2, "No Suitable PPD File Available for a PostScript Printer"*. Alternatively, find PPD files at http://www.linuxfoundation.org/collaborate/workgroups/openprinting/database/databaseintro, the "OpenPrinting.org printer database". When downloading PPD files from OpenPrinting, keep in mind that it always shows the latest Linux support status, which is not necessarily met by openSUSE Leap.

PROCEDURE 7.5: ADDING A PPD FILE FOR POSTSCRIPT PRINTERS

1. Start the YaST printer module with *Hardware* › *Printer*.
2. In the *Printer Configurations* screen, click *Add*.

3. In the `Find and Assign a Driver` section, click *Driver Packages*.
4. Enter the full path to the PPD file into the text box under `Make a Printer Description File Available`.
5. Click *OK* to return to the `Add New Printer Configuration` screen.
6. To directly use this PPD file, proceed as described in *Procedure 7.3, "Adding a New Printer"*.

7.3.1.2 Editing a Local Printer Configuration

By editing an existing configuration for a printer you can change basic settings such as connection type and driver. It is also possible to adjust the default settings for paper size, resolution, media source, etc. You can change identifiers of the printer by altering the printer description or location.

1. Start the YaST printer module with *Hardware* › *Printer*.
2. In the *Printer Configurations* screen, choose a local printer configuration from the list and click *Edit*.
3. Change the connection type or the driver as described in *Procedure 7.3, "Adding a New Printer"*. This should only be necessary in case you have problems with the current configuration.
4. Optionally, make this printer the default by checking *Default Printer*.
5. Adjust the default settings by clicking *All Options for the Current Driver*. To change a setting, expand the list of options by clicking the relative `+` sign. Change the default by clicking an option. Apply your changes with *OK*.

7.3.2 Configuring Printing via the Network with YaST

Network printers are not detected automatically. They must be configured manually using the YaST printer module. Depending on your network setup, you can print to a print server (CUPS, LPD, SMB, or IPX) or directly to a network printer (preferably via TCP). Access the configuration view for network printing by choosing *Printing via Network* from the left pane in the YaST printer module.

7.3.2.1 Using CUPS

In a Linux environment CUPS is usually used to print via the network. The simplest setup is to only print via a single CUPS server which can directly be accessed by all clients. Printing via more than one CUPS server requires a running local CUPS daemon that communicates with the remote CUPS servers.

Important: Browsing Network Print Queues

CUPS servers announce their print queues over the network either via the traditional CUPS browsing protocol or via Bonjour/DNS-SD. Clients need to browse these lists, so users can select specific printers to send their print jobs to. To browse network print queues, the service `cups-browsed` provided by the package `cups-filters-cups-browsed` must run on all clients that print via CUPS servers. `cups-browsed` is started automatically when configuring network printing with YaST.

In case browsing does not work after having started `cups-browsed`, the CUPS server(s) probably announce the network print queues via Bonjour/DNS-SD. In this case you need to additionally install the package `avahi` and start the associated service with **`sudo systemctl start avahi-daemon`** on all clients.

PROCEDURE 7.6: PRINTING VIA A SINGLE CUPS SERVER

1. Start the YaST printer module with *Hardware* › *Printer*.
2. From the left pane, launch the *Print via Network* screen.
3. Check *Do All Your Printing Directly via One Single CUPS Server* and specify the name or IP address of the server.
4. Click *Test Server* to make sure you have chosen the correct name or IP address.
5. Click *OK* to return to the *Printer Configurations* screen. All printers available via the CUPS server are now listed.

PROCEDURE 7.7: PRINTING VIA MULTIPLE CUPS SERVERS

1. Start the YaST printer module with *Hardware* › *Printer*.
2. From the left pane, launch the *Print via Network* screen.
3. Check *Accept Printer Announcements from CUPS Servers*.

4. Under `General Settings` specify which servers to use. You may accept connections from all networks available or from specific hosts. If you choose the latter option, you need to specify the host names or IP addresses.

5. Confirm by clicking *OK* and then *Yes* when asked to start a local CUPS server. After the server has started YaST will return to the *Printer Configurations* screen. Click *Refresh list* to see the printers detected so far. Click this button again, in case more printers are available.

7.3.2.2 Using Print Servers other than CUPS

If your network offers print services via print servers other than CUPS, start the YaST printer module with *Hardware* › *Printer* and launch the *Print via Network* screen from the left pane. Start the *Connection Wizard* and choose the appropriate *Connection Type*. Ask your network administrator for details on configuring a network printer in your environment.

7.3.3 Sharing Printers over the Network

Printers managed by a local CUPS daemon can be shared over the network and so turn your machine into a CUPS server. Usually you share a printer by enabling so-called “browsing mode” in CUPS. If browsing is enabled, the local print queues are made available on the network for listening to remote CUPS daemons. It is also possible to set up a dedicated CUPS server that manages all print queues and can directly be accessed by remote clients. In this case it is not necessary to enable browsing.

PROCEDURE 7.8: SHARING PRINTERS

1. Start the YaST printer module with *Hardware* › *Printer*.

2. Launch the *Share Printers* screen from the left pane.

3. Select *Allow Remote Access*. Also check *For computers within the local network* and enable browsing mode by also checking *Publish printers by default within the local network*.

4. Click *OK* to restart the CUPS server and to return to the *Printer Configurations* screen.

5. Regarding CUPS and firewall settings, see http://en.opensuse.org/SDB:CUPS_and_SANE_Firewall_settings.

7.4 Setting Up a Scanner

You can configure a USB or SCSI scanner with YaST. The `sane-backends` package contains hardware drivers and other essentials needed to use a scanner. If you own an HP All-In-One device, see *Section 7.4.1, "Configuring an HP All-In-One Device"*, instructions on how to configure a network scanner are available at *Section 7.4.3, "Scanning over the Network"*.

PROCEDURE 7.9: CONFIGURING A USB OR SCSI SCANNER

1. Connect your USB or SCSI scanner to your computer and turn it on.
2. Start YaST and select *Hardware* › *Scanner*. YaST builds the scanner database and tries to detect your scanner model automatically.
   If a USB or SCSI scanner is not properly detected, try *Other* › *Restart Detection*.
3. To activate the scanner select it from the list of detected scanners and click *Edit*.
4. Choose your model form the list and click *Next* and *Finish*.
5. Use *Other* › *Test* to make sure you have chosen the correct driver.
6. Leave the configuration screen with *OK*.

7.4.1 Configuring an HP All-In-One Device

An HP All-In-One device can be configured with YaST even if it is made available via the network. If you own a USB HP All-In-One device, start configuring as described in *Procedure 7.9, "Configuring a USB or SCSI Scanner"*. If it is detected properly and the *Test* succeeds, it is ready to use.

If your USB device is not properly detected, or your HP All-In-One device is connected to the network, run the HP Device Manager:

1. Start YaST and select *Hardware* › *Scanner*. YaST loads the scanner database.
2. Start the HP Device Manager with *Other* › *Run hp-setup* and follow the on-screen instructions. After having finished the HP Device Manager, the YaST scanner module automatically restarts the auto detection.
3. Test it by choosing *Other* › *Test*.
4. Leave the configuration screen with *OK*.

7.4.2 Sharing a Scanner over the Network

openSUSE Leap allows the sharing of a scanner over the network. To do so, configure your scanner as follows:

1. Configure the scanner as described in *Section 7.4, "Setting Up a Scanner"*.
2. Choose *Other* › *Scanning via Network*.
3. Enter the host names of the clients (separated by a comma) that should be allowed to use the scanner under *Server Settings* › *Permitted Clients for saned* and leave the configuration dialog with *OK*.

7.4.3 Scanning over the Network

To use a scanner that is shared over the network, proceed as follows:

1. Start YaST and select *Hardware* › *Scanner*.
2. Open the network scanner configuration menu by *Other* › *Scanning via Network*.
3. Enter the host name of the machine the scanner is connected to under *Client Settings* › *Servers Used for the net Metadriver*
4. Leave with *OK*. The network scanner is now listed in the Scanner Configuration window and is ready to use.

8 Printer Operation

openSUSE® Leap supports printing with many types of printers, including remote network printers. Printers can be configured manually or with YaST. For configuration instructions, refer to *Section 7.3, "Setting Up a Printer"*. Both graphical and command line utilities are available for starting and managing print jobs. If your printer does not work as expected, refer to *Section 8.8, "Troubleshooting"*.

CUPS (Common Unix Printing System) is the standard print system in openSUSE Leap.

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 (USB, Ethernet, or Wi-Fi) 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. 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.

Currently PostScript is being replaced by PDF as the standard print job format. PostScript + PDF printers that can directly print PDF (in addition to PostScript) already exist. For traditional PostScript printers PDF needs to be converted to PostScript in the printing workflow.

Standard Printers (Languages Like PCL and ESC/P)

In the case of known printer languages, the print system can convert PostScript jobs to the respective printer language with Ghostscript. This processing stage is called 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 an adequate print result. Linux may not be able to address some special printer functions. Except for HP and Epson, there are currently no printer manufacturers who develop Linux drivers and make them available to Linux distributors under an open source license.

Proprietary Printers (Also Called GDI Printers)

These printers do not support any of the common printer languages. They use their own undocumented printer languages, which are subject to change when a new edition of a model is released. Usually only Windows drivers are available for these printers. See *Section 8.8.1, "Printers without Standard Printer Language Support"* 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.linuxfoundation.org/OpenPrinting/

The OpenPrinting home page with the printer database. The database shows 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 openSUSE Leap version was released. Thus, the databases may not necessarily indicate the correct status, but only provide an approximation.

http://pages.cs.wisc.edu/~ghost/

The Ghostscript Web page.

`/usr/share/doc/packages/ghostscript/catalog.devices`

List of built-in Ghostscript drivers.

8.1 The CUPS Workflow

The user creates a print job. The print job consists of the data to print plus information for the spooler. This includes the name of the printer or the name of the print queue, and optionally, information for the filter, such as printer-specific options.

At least one dedicated print 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 generated by the application that is printing (usually PostScript or PDF, but also ASCII, 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. This requires a printer driver suitable for your printer. The back-end receives the printer-specific data from the filter then passes it to the printer.

8.2 Methods and Protocols for Connecting Printers

There are various possibilities for connecting a printer to the system. The configuration of CUPS does not distinguish between a local printer and a printer connected to the system over the network. For more information about the printer connection, read the article *CUPS in a Nutshell* at http://en.opensuse.org/SDB:CUPS_in_a_Nutshell.

Warning: Changing Cable Connections in a Running System

When connecting the printer to the machine, do not forget that only USB devices can be plugged in or unplugged during operation. To avoid damaging your system or printer, shut down the system before changing any connections that are not USB.

8.3 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.

To configure a PostScript printer, the best approach is to get a suitable PPD file. Many PPD files are available in the packages `manufacturer-PPDs` and `OpenPrintingPPDs-postscript`. See *Section 8.7.3, "PPD Files in Various Packages"* and *Section 8.8.2, "No Suitable PPD File Available for a PostScript Printer"*.

New PPD files can be stored in the directory `/usr/share/cups/model/` or added to the print system with YaST as described in *Section 7.3.1.1, "Adding Drivers with YaST"*. Subsequently, the PPD file can be selected during the printer setup.

Be careful if a printer manufacturer wants you to install entire software packages. This kind of installation may result in the loss of the support provided by openSUSE Leap. Also, 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.

8.4 Network Printers

A network printer can support various protocols, some even concurrently. Although most of the supported protocols are standardized, some manufacturers modify the standard. Manufacturers then provide drivers for only a few operating 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 need to experiment with various options to achieve a functional configuration.

CUPS supports the `socket`, `LPD`, `IPP` and `smb` protocols.

socket

Socket refers to a connection in which the plain print data is sent directly to a TCP socket. Some socket port numbers that are commonly used are `9100` or `35`. The device URI (uniform resource identifier) syntax is: socket://*`IP.OF.THE.PRINTER`*:*`PORT`*, for example: `socket://192.168.2.202:9100/`.

LPD (Line Printer Daemon)

The LPD protocol is described in RFC 1179. Under this protocol, some job-related data, such as the ID of the print queue, is sent before the actual print data is sent. Therefore, a print queue must be specified when configuring the LPD protocol. The implementations of diverse printer manufacturers are flexible enough to accept any name as the print queue. If necessary, the printer manual should indicate what name to use. LPT, LPT1, LP1 or similar names are often used. The port number for an LPD service is `515`. An example device URI is `lpd://192.168.2.202/LPT1`.

IPP (Internet Printing Protocol)

IPP is 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. The name of the print queue is necessary to configure IPP correctly. The port number for IPP is `631`. Example device URIs are `ipp://192.168.2.202/ps` and `ipp://192.168.2.202/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/smb.example.com/printer`, `smb://user:password@smb.example.com/printer`, and `smb://smb.example.com/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 ascertain the protocol. **`nmap`** checks a host for open ports. For example:

```
tux > nmap -p 35,137-139,515,631,9100-10000 IP.OF.THE.PRINTER
```

8.5 Configuring CUPS with Command Line Tools

CUPS can be configured with command line tools like **`lpinfo`**, **`lpadmin`** and **`lpoptions`**. You need a device URI consisting of a back-end, such as USB, and parameters. To determine valid device URIs on your system use the command **`lpinfo -v | grep ":/"`**:

```
tux > sudo lpinfo -v | grep ":/"
direct usb://ACME/FunPrinter%20XL
network socket://192.168.2.253
```

With **`lpadmin`** the CUPS server administrator can add, remove or manage print queues. To add a print queue, use the following syntax:

```
tux > sudo lpadmin -p QUEUE -v DEVICE-URI -P PPD-FILE -E
```

Then the device (`-v`) is available as *`QUEUE`* (`-p`), using the specified PPD file (`-P`). This means that you must know the PPD file and the device URI 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:

```
tux > sudo lpadmin -p ps -v usb://ACME/FunPrinter%20XL -P \
/usr/share/cups/model/Postscript.ppd.gz -E
```

The following example configures a network printer:

```
tux > sudo lpadmin -p ps -v socket://192.168.2.202:9100/ -P \
/usr/share/cups/model/Postscript-level1.ppd.gz -E
```

For more options of **lpadmin**, see the man page of `lpadmin(8)`.

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:

   ```
   tux > sudo lpoptions -p QUEUE -l
   ```

   Example:

   ```
   Resolution/Output Resolution: 150dpi *300dpi 600dpi
   ```

   The activated default option is identified by a preceding asterisk (`*`).

2. Change the option with **lpadmin**:

   ```
   tux > sudo lpadmin -p QUEUE -o Resolution=600dpi
   ```

3. Check the new setting:

   ```
   tux > sudo lpoptions -p QUEUE -l

   Resolution/Output Resolution: 150dpi 300dpi *600dpi
   ```

When a normal user runs **lpoptions**, the settings are written to `~/.cups/lpoptions`. However, `root` settings are written to `/etc/cups/lpoptions`.

8.6 Printing from the Command Line

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*.

8.7 Special Features in openSUSE Leap

Several CUPS features have been adapted for openSUSE Leap. Some of the most important changes are covered here.

8.7.1 CUPS and Firewall

After completing a default installation of openSUSE Leap, `firewalld` is active and the network interfaces are configured to be in the `public` zone, which blocks incoming traffic.

When `firewalld` is active, you may need to configure it to allow clients to browse network printers by allowing `mdns` and `ipp` through the internal network zone. The public zone should never expose printer queues.

(More information about the `firewalld` configuration is available in *Book "*Security Guide*", Chapter 16 "Masquerading and Firewalls", Section 16.4 "*`firewalld`*"* and at http://en.opensuse.org/SDB:CUPS_and_SANE_Firewall_settings.)

8.7.1.1 CUPS Client

Normally, a CUPS client runs on a regular workstation located in a trusted network environment behind a firewall. In this case it is recommended to configure the network interface to be in the `Internal Zone`, so the workstation is reachable from within the network.

8.7.1.2 CUPS Server

If the CUPS server is part of a trusted network environment protected by a firewall, the network interface should be configured to be in the `Internal Zone` of the firewall. It is not recommended to set up a CUPS server in an untrusted network environment unless you ensure that it is protected by special firewall rules and secure settings in the CUPS configuration.

8.7.2 Browsing for Network Printers

CUPS servers regularly announce the availability and status information of shared printers over the network. Clients can access this information to display a list of available printers in printing dialogs, for example. This is called "browsing".

CUPS servers announce their print queues over the network either via the traditional CUPS browsing protocol, or via Bonjour/DNS-SD. To enable browsing network print queues, the service `cups-browsed` needs to run on all clients that print via CUPS servers. `cups-browsed` is not started by default. To start it for the active session, use **`sudo systemctl start cups-browsed`**. To ensure it is automatically started after booting, enable it with **`sudo systemctl enable cups-browsed`** on all clients.

In case browsing does not work after having started `cups-browsed`, the CUPS server(s) probably announce the network print queues via Bonjour/DNS-SD. In this case you need to additionally install the package `avahi` and start the associated service with **`sudo systemctl start avahi-daemon`** on all clients.

See *Section 8.7.1, "CUPS and Firewall"* for information on allowing printer browsing through `firewalld`.

8.7.3 PPD Files in Various Packages

The YaST printer configuration sets up the queues for CUPS using the PPD files installed in `/usr/share/cups/model`. 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. For this purpose, the YaST printer configuration generates a database from the vendor and model information extracted from the PPD files.

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. For example, if you have PostScript printers the PPD files can be copied directly to `/usr/share/cups/model` (if they do not already exist in the `manufacturer-PPDs` or `OpenPrintingPPDs-postscript` packages) to achieve an optimum configuration for your printers.

Additional PPD files are provided by the following packages:

- `gutenprint`: the Gutenprint driver and its matching PPDs
- `splix`: the SpliX driver and its matching PPDs
- `OpenPrintingPPDs-ghostscript`: PPDs for Ghostscript built-in drivers
- `OpenPrintingPPDs-hpijs`: PPDs for the HPIJS driver for non-HP printers

8.8 Troubleshooting

The following sections cover some of the most frequently encountered printer hardware and software problems and ways to solve or circumvent these problems. Among the topics covered are GDI printers, PPD files and port configuration. Common network printer problems, defective printouts, and queue handling are also addressed.

8.8.1 Printers without Standard Printer Language Support

These printers do not support any common printer language and can only be addressed with special proprietary control sequences. Therefore they can 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. Usually the manufacturer delivers drivers only for Windows, and since the Windows driver uses the GDI interface these printers are also called *GDI printers*. The actual problem is not the programming interface, but that these printers can only be addressed with the proprietary printer language of the respective printer model.

Some GDI printers can be switched to operate either in GDI mode or in one of the standard printer languages. See the manual of the printer whether this is possible. Some models require special Windows software to do the switch (note that the Windows printer driver may always switch the printer back into GDI mode when printing from Windows). For other GDI printers there are extension modules for a standard printer language available.

Some manufacturers provide proprietary drivers for their printers. The disadvantage of proprietary printer drivers is that there is no guarantee that these work with the installed print system or 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 printer which supports a standard printer language (preferably PostScript). 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 because of new developments in the print system.

8.8.2 No Suitable PPD File Available for a PostScript Printer

If the `manufacturer-PPDs` or `OpenPrintingPPDs-postscript` packages do not contain a 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.

8.8.3 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`*:

```
tux > 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.

Provided that the respective **`lpd`** is active and the host accepts queries, run the following command as `root` to query a status report for *`QUEUE`* on remote *`HOST`*:

```
root # 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 shown in *Example 8.1, "Error Message from* **`lpd`***"*, the problem is caused by the remote **`lpd`**.

EXAMPLE 8.1: ERROR MESSAGE FROM `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`

A CUPS network server can broadcast its queues by default every 30 seconds on UDP port `631`. Accordingly, the following command can be used to test whether there is a broadcasting CUPS network server in the network. Make sure to stop your local CUPS daemon before executing the command.

```
tux > netcat -u -l -p 631 & PID=$! ; sleep 40 ; kill $PID
```

If a broadcasting CUPS network server exists, the output appears as shown in *Example 8.2, "Broadcast from the CUPS Network Server"*.

EXAMPLE 8.2: BROADCAST FROM THE CUPS NETWORK SERVER

```
ipp://192.168.2.202:631/printers/queue
```

The following command can be used to test if a TCP connection can be established to **`cupsd`** (port `631`) on *`HOST`*:

```
tux > 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.

```
tux > echo -en "\r" \
| lp -d queue -h HOST
```

Troubleshooting a Network Printer or Print Server Machine

Spoolers running in a print server machine sometimes cause problems when they need to deal with multiple print jobs. Since this is caused by the spooler in the print server machine, there no way to resolve this issue. As a work-around, circumvent the spooler in the print server machine by addressing the printer connected to the print server machine directly with the TCP socket. See *Section 8.4, "Network Printers"*.

In this way, the print server machine 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 machine. If the printer is connected to the print server machine and turned on, this TCP port can usually be determined with the **nmap** utility from the `nmap` package some time after the print server machine is powered up. For example, **nmap** *IP-address* may deliver the following output for a print server machine:

```
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 machine can be addressed via TCP socket on port `9100`. By default, **nmap** only checks several 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 man page of **nmap**.

Enter a command like

```
tux > 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.

8.8.4 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 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 cannot print the printer-specific data, select a PPD file that is more suitable for the printer.

8.8.5 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 determines how many unsuccessful attempts are appropriate until the data transfer is reported as impossible. As further attempts would be in vain, **`cupsd`** disables printing for the respective queue. After eliminating the cause of the problem, the system administrator must re-enable printing with the command **`cupsenable`**.

8.8.6 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`** on the server 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. As a print job is usually forwarded immediately, it cannot be deleted with the job number on the client host This is because the client **`cupsd`** regards the print job as completed when it has been forwarded to the server **`cupsd`**.

To delete the print job on the server, use a command such as **`lpstat -h cups.example.com -o`** to determine the job number on the server. This assumes that the server has not already completed the print job (that is, sent it completely to the printer). Use the obtained job number to delete the print job on the server as follows:

```
tux > cancel -h cups.example.com QUEUE-JOBNUMBER
```

8.8.7 Defective Print Jobs and Data Transfer Errors

If you switch the printer off or shut down the computer during the printing process, print jobs remain in the queue. Printing resumes when the computer (or the printer) is switched back on. Defective print jobs must be removed from the queue with **`cancel`**.

If a print job is corrupted or an error occurs in the communication between the host and the printer, the printer cannot process the data correctly and prints numerous sheets of paper with unintelligible characters. To fix the problem, 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 cups.example.com -o`** to check which queue is currently printing. Delete the print job with **`cancel`** *QUEUE*-*JOBNUMBER* or **`cancel -h cups.example.com`** *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.
4. Reset the printer completely by switching it off for some time. Then insert the paper and turn on the printer.

8.8.8 Debugging CUPS

Use the following generic procedure to locate problems in CUPS:

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.

8.8.9 For More Information

In-depth information about printing on openSUSE Leap is presented in the openSUSE Support Database at http://en.opensuse.org/Portal:Printing.

9 Accessing File Systems with FUSE

FUSE is the acronym for *file system in user space*. This means you can configure and mount a file system as an unprivileged user. Normally, you need to be `root` for this task. FUSE alone is a kernel module. Combined with plug-ins, it allows you to extend FUSE to access almost all file systems like remote SSH connections, ISO images, and more.

9.1 Configuring FUSE

Before you can use FUSE, you need to install the package `fuse`. Depending which file system you want to use, you need additional plug-ins available as separate packages. For an overview, see *Section 9.5, "Available FUSE Plug-ins"*.

Generally you do not need to configure FUSE. However, it is a good idea to create a directory where all your mount points are combined. For example, you can create a directory `~/mounts` and insert your subdirectories for your different file systems there.

9.2 Mounting an NTFS Partition

NTFS, the *New Technology File System*, is the default file system of Windows. Since under normal circumstances the unprivileged user cannot mount NTFS block devices using the external FUSE library, the process of mounting a Windows partition described below requires root privileges.

1. Become `root` and install the package `ntfs-3g`.
2. Create a directory that is to be used as a mount point, for example `~/mounts/windows`.
3. Find out which Windows partition you need. Use YaST and start the partitioner module to see which partition belongs to Windows, but do not modify anything. Alternatively, become `root` and execute `/sbin/fdisk -l`. Look for partitions with a partition type of `HPFS/NTFS`.
4. Mount the partition in read-write mode. Replace the placeholder *`DEVICE`* with your respective Windows partition:

```
tux > ntfs-3g /dev/DEVICE MOUNT POINT
```

To use your Windows partition in read-only mode, append `-o ro`:

```
tux > ntfs-3g /dev/DEVICE MOUNT POINT -o ro
```

The command **`ntfs-3g`** uses the current user (UID) and group (GID) to mount the given device. If you want to set the write permissions to a different user, use the command **`id`** `USER` to get the output of the UID and GID values. Set it with:

```
root # id tux
uid=1000(tux) gid=100(users) groups=100(users),16(dialout),33(video)
ntfs-3g /dev/DEVICE MOUNT POINT -o uid=1000,gid=100
```

Find additional options in the man page.

To unmount the resource, run **`fusermount -u`** *MOUNT POINT*.

9.3 Mounting Remote File System with SSHFS

SSH, the secure shell network protocol, can be used to exchange data between two computers using a secure channel. To establish an SSH connection through FUSE, proceed as follows:

1. Install the package `sshfs`.
2. Create a directory that is to be used as a mount point. A good idea is to use `~/mounts/`*HOST*. Replace *HOST* with the name of your remote computer.
3. Mount the remote file system:

   ```
   root # sshfs USER@HOST MOUNT POINT
   ```

4. Enter your password for the remote computer.

To unmount the resource, run **`fusermount -u`** *MOUNT POINT*.

9.4 Mounting an ISO File System

To look into an ISO image, you can mount it with the `fuseiso` package:

1. Install the package `fuseiso`.

2. Create a directory that is to be used as a mount point, for example `~/mounts/iso`.

3. Mount the ISO image:

```
root # fuseiso ISO_IMAGE MOUNT POINT
```

You can only read content from the ISO image, but you can not write back. To unmount the resource, use **`fusermount -u`** *`MOUNT POINT`*.

9.5 Available FUSE Plug-ins

FUSE is dependent on plug-ins. The following table lists common plug-ins.

TABLE 9.1: AVAILABLE FUSE PLUG-INS

`curlftpfs`	mount FTP servers
`encfs`	mount encrypted file systems
`fuseiso`	mounts CD-ROM images with ISO9660 file systems in them
`fusepod`	mount iPods
`fusesmb`	mount browseable Samba clients or Windows shares
`gphotofs`	mount supported digital cameras through gPhoto
`ntfs-3g`	mount NTFS volumes (with read and write support)
`obexfs`	mount Bluetooth devices
`sshfs`	file system client based on SSH file transfer protocol
`wdfs`	mount WebDAV file systems

9.6 For More Information

See the home page http://fuse.sourceforge.net of FUSE for more information.

III Managing and Updating Software

10 Installing or Removing Software

Use YaST's software management module to search for software components you want to add or remove. YaST resolves all dependencies for you. To install packages not shipped with the installation media, add software repositories to your setup and let YaST manage them. Keep your system up-to-date by managing software updates with the update applet.

Change the software collection of your system with the YaST Software Manager. This YaST module is available in two flavors: a graphical variant for X Window and a text-based variant to be used on the command line. The graphical flavor is described here—for details on the text-based YaST, see *Book "Reference", Chapter 1 "YaST in Text Mode"*.

Note: Confirmation and Review of Changes

When installing, updating or removing packages, any changes in the Software Manager are only applied after clicking *Accept* or *Apply*. YaST maintains a list with all actions, allowing you to review and modify your changes before applying them to the system.

10.1 Definition of Terms

The following terms are important for understanding installing and removing software in openSUSE Leap.

Repository

A local or remote directory containing packages, plus additional information about these packages (package metadata).

(Repository) Alias/Repository Name

A short name for a repository (called `Alias` within Zypper and *Repository Name* within YaST). It can be chosen by the user when adding a repository and must be unique.

Repository Description Files

Each repository provides files describing content of the repository (package names, versions, etc.). These repository description files are downloaded to a local cache that is used by YaST.

Product

Represents a whole product, for example openSUSE® Leap.

Pattern

A pattern is an installable group of packages dedicated to a certain purpose. For example, the `Laptop` pattern contains all packages that are needed in a mobile computing environment. Patterns define package dependencies (such as required or recommended packages) and come with a preselection of packages marked for installation. This ensures that the most important packages needed for a certain purpose are available on your system after installation of the pattern. If necessary, you can manually select or deselect packages within a pattern.

Package

A package is a compressed file in `rpm` format that contains the files for a particular program.

Patch

A patch consists of one or more packages and may be applied by means of delta RPMs. It may also introduce dependencies to packages that are not installed yet.

Resolvable

A generic term for product, pattern, package or patch. The most commonly used type of resolvable is a package or a patch.

Delta RPM

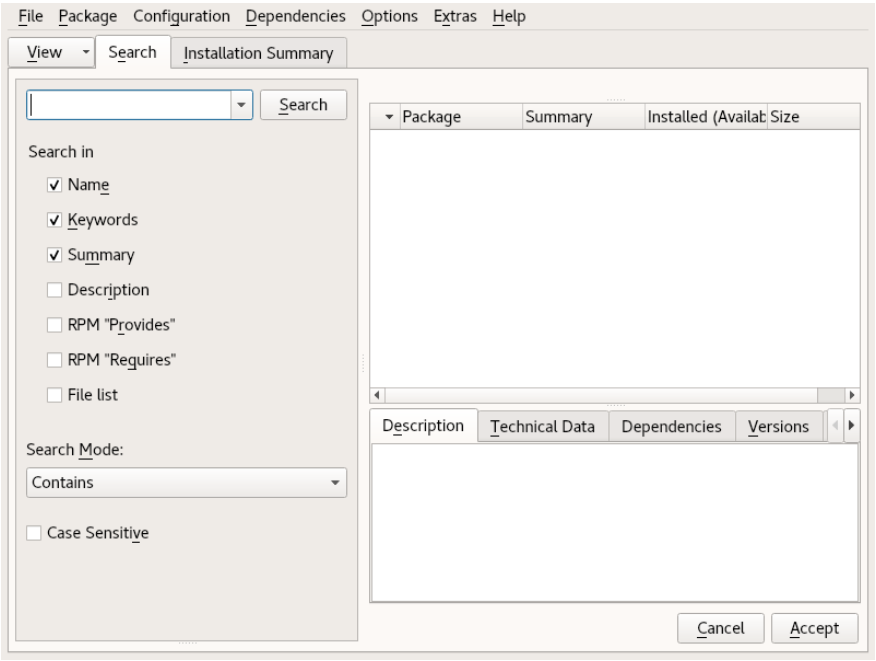
A delta RPM consists only of the binary diff between two defined versions of a package, and therefore has the smallest download size. Before being installed, the full RPM package is rebuilt on the local machine.

Package Dependencies

Certain packages are dependent on other packages, such as shared libraries. In other terms, a package may `require` other packages—if the required packages are not available, the package cannot be installed. In addition to dependencies (package requirements) that must be fulfilled, some packages `recommend` other packages. These recommended packages are only installed if they are actually available, otherwise they are ignored and the package recommending them is installed nevertheless.

10.2 Using the YaST Software Manager

Start the software manager from the *YaST Control Center* by choosing *Software* › *Software Management*.



10.2.1 Views for Searching Packages or Patterns

The YaST software manager can install packages or patterns from all currently enabled repositories. It offers different views and filters to make it easier to find the software you are searching for. The *Search* view is the default view of the window. To change view, click *View* and select one of the following entries from the drop-down box. The selected view opens in a new tab.

Patterns

Lists all patterns available for installation on your system.

Package Groups

Lists all packages sorted by groups such as *Graphics*, *Programming*, or *Security*.

Languages

A filter to list all packages needed to add a new system language.

Repositories

A filter to list packages by repository. To select more than one repository, hold the Ctrl key while clicking repository names. The "pseudo repository" *@System* lists all packages currently installed.

Services

Shows which packages belong to a certain module or extension. Select an entry (for example, `Basesystem` or `High Availability`) to display a list of packages that belong to this module or extension.

Search

Lets you search for a package according to certain criteria. Enter a search term and press Enter. Refine your search by specifying where to *Search In* and by changing the *Search Mode*. For example, if you do not know the package name but only the name of the application that you are searching for, try including the package *Description* in the search process.

Installation Summary

If you have already selected packages for installation, update or removal, this view shows the changes that will be applied to your system when you click *Accept*. To filter for packages with a certain status in this view, activate or deactivate the respective check boxes. Press Shift–F1 for details on the status flags.

Tip: Finding Packages Not Belonging to an Active Repository

To list all packages that do not belong to an active repository, choose *View* › *Repositories* › *@System* and then choose *Secondary Filter* › *Unmaintained Packages*. This is useful, for example, if you have deleted a repository and want to make sure no packages from that repository remain installed.

10.2.2 Installing and Removing Packages or Patterns

Certain packages are dependent on other packages, such as shared libraries. On the other hand, some packages cannot coexist with others on the system. If possible, YaST automatically resolves these dependencies or conflicts. If your choice results in a dependency conflict that cannot be automatically solved, you need to solve it manually as described in *Section 10.2.4, "Package Dependencies"*.

Note: Removal of Packages

When removing any packages, by default YaST only removes the selected packages. If you want YaST to also remove any other packages that become unneeded after removal of the specified package, select *Options* › *Cleanup when deleting packages* from the main menu.

1. Search for packages as described in *Section 10.2.1, "Views for Searching Packages or Patterns"*.
2. The packages found are listed in the right pane. To install a package or remove it, right-click it and choose *Install* or *Delete*. If the relevant option is not available, check the package status indicated by the symbol in front of the package name—press Shift–F1 for help.

Tip: Applying an Action to All Packages Listed

To apply an action to all packages listed in the right pane, go to the main menu and choose an action from *Package* › *All in This List*.

3. To install a pattern, right-click the pattern name and choose *Install*.
4. It is not possible to remove a pattern. Instead, select the packages of a pattern you want to remove and mark them for removal.
5. To select more packages, repeat the steps mentioned above.
6. Before applying your changes, you can review or modify them by clicking *View* › *Installation Summary*. By default, all packages that will change status, are listed.
7. To revert the status for a package, right-click the package and select one of the following entries: *Keep* if the package was scheduled to be deleted or updated, or *Do Not Install* if it was scheduled for installation. To abandon all changes and quit the Software Manager, click *Cancel* and *Abandon*.
8. When you are finished, click *Accept* to apply your changes.
9. In case YaST found dependencies on other packages, a list of packages that have additionally been chosen for installation, update or removal is presented. Click *Continue* to accept them.
   After all selected packages are installed, updated or removed, the YaST Software Manager automatically terminates.

Note: Installing Source Packages

Installing source packages with YaST Software Manager is not possible at the moment. Use the command line tool **zypper** for this purpose. For more information, see *Book "Reference", Chapter 2 "Managing Software with Command Line Tools", Section 2.1.2.5 "Installing or Downloading Source Packages"*.

10.2.3 Updating Packages

Instead of updating individual packages, you can also update all installed packages or all packages from a certain repository. When mass updating packages, the following aspects are generally considered:

- priorities of the repositories that provide the package,
- architecture of the package (for example, AMD64/Intel 64),
- version number of the package,
- package vendor.

Which of the aspects has the highest importance for choosing the update candidates depends on the respective update option you choose.

1. To update all installed packages to the latest version, choose *Package* › *All Packages* › *Update if Newer Version Available* from the main menu.
   All repositories are checked for possible update candidates, using the following policy: YaST first tries to restrict the search to packages with the same architecture and vendor like the installed one. If the search is positive, the "best" update candidate from those is selected according to the process below. However, if no comparable package of the same vendor can be found, the search is expanded to all packages with the same architecture. If still no comparable package can be found, all packages are considered and the "best" update candidate is selected according to the following criteria:

   1. Repository priority: Prefer the package from the repository with the highest priority.
   2. If more than one package results from this selection, choose the one with the "best" architecture (best choice: matching the architecture of the installed one).

If the resulting package has a higher version number than the installed one, the installed package will be updated and replaced with the selected update candidate.
This option tries to avoid changes in architecture and vendor for the installed packages, but under certain circumstances, they are tolerated.

Note: Update Unconditionally

If you choose *Package* › *All Packages* › *Update Unconditionally* instead, the same criteria apply but any candidate package found is installed unconditionally. Thus, choosing this option might actually lead to downgrading some packages.

2. To make sure that the packages for a mass update derive from a certain repository:

   a. Choose the repository from which to update as described in *Section 10.2.1, "Views for Searching Packages or Patterns"* .

   b. On the right hand side of the window, click *Switch system packages to the versions in this repository*. This explicitly allows YaST to change the package vendor when replacing the packages.
   When you proceed with *Accept*, all installed packages will be replaced by packages deriving from this repository, if available. This may lead to changes in vendor and architecture and even to downgrading some packages.

   c. To refrain from this, click *Cancel switching system packages to the versions in this repository*. Note that you can only cancel this until you click the *Accept* button.

3. Before applying your changes, you can review or modify them by clicking *View* › *Installation Summary*. By default, all packages that will change status, are listed.

4. If all options are set according to your wishes, confirm your changes with *Accept* to start the mass update.

10.2.4 Package Dependencies

Most packages are dependent on other packages. If a package, for example, uses a shared library, it is dependent on the package providing this library. On the other hand, some packages cannot coexist, causing a conflict (for example, you can only install one mail transfer agent: sendmail or postfix). When installing or removing software, the Software Manager makes sure no dependencies or conflicts remain unsolved to ensure system integrity.

In case there exists only one solution to resolve a dependency or a conflict, it is resolved automatically. Multiple solutions always cause a conflict which needs to be resolved manually. If solving a conflict involves a vendor or architecture change, it also needs to be solved manually. When clicking *Accept* to apply any changes in the Software Manager, you get an overview of all actions triggered by the automatic resolver which you need to confirm.

By default, dependencies are automatically checked. A check is performed every time you change a package status (for example, by marking a package for installation or removal). This is generally useful, but can become exhausting when manually resolving a dependency conflict. To disable this function, go to the main menu and deactivate *Dependencies* › *Autocheck*. Manually perform a dependency check with *Dependencies* › *Check Now*. A consistency check is always performed when you confirm your selection with *Accept*.

To review a package's dependencies, right-click it and choose *Show Solver Information*. A map showing the dependencies opens. Packages that are already installed are displayed in a green frame.

Note: Manually Solving Package Conflicts

Unless you are very experienced, follow the suggestions YaST makes when handling package conflicts, otherwise you may not be able to resolve them. Keep in mind that every change you make, potentially triggers other conflicts, so you can easily end up with a steadily increasing number of conflicts. In case this happens, *Cancel* the Software Manager, *Abandon* all your changes and start again.

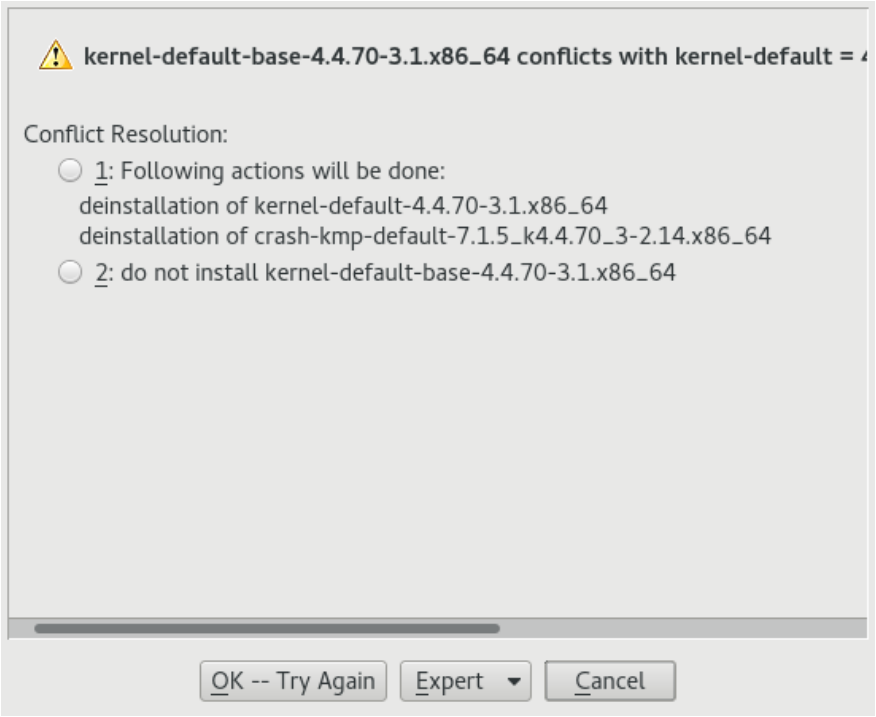


FIGURE 10.1: CONFLICT MANAGEMENT OF THE SOFTWARE MANAGER

10.2.5 Handling of Package Recommendations

In addition to the hard dependencies required to run a program (for example a certain library), a package can also have weak dependencies, that add for example extra functionality or translations. These weak dependencies are called package recommendations.

The way package recommendations are handled has slightly changed starting with openSUSE Leap 42.1. Nothing has changed when installing a new package—recommended packages are still installed by default.

Prior to openSUSE Leap 42.1, missing recommendations for already installed packages were installed automatically. Now these packages will no longer be installed automatically. To switch to the old default, set `PKGMGR_REEVALUATE_RECOMMENDED="yes"` in `/etc/sysconfig/yast2`. To install all missing recommendations for already installed packages, start *YaST* › *Software Manager* and choose *Extras* › *Install All Matching Recommended Packages*.

To disable the installation of recommended packages when installing new packages, deactivate *Dependencies* › *Install Recommended Packages* in the YaST Software Manager. If using the command line tool Zypper to install packages, use the option `--no-recommends`.

10.3 Managing Software Repositories and Services

To install third-party software, add software repositories to your system. By default, the product repositories such as openSUSE Leap-DVD 15.1 and a matching update repository are automatically configured. Depending on the initially selected product, an additional repository containing translations, dictionaries, etc. might also be configured.

To manage repositories, start YaST and select *Software* › *Software Repositories*. The *Configured Software Repositories* dialog opens. Here, you can also manage subscriptions to so-called *Services* by changing the *View* at the right corner of the dialog to *All Services*. A Service in this context is a *Repository Index Service* (RIS) that can offer one or more software repositories. Such a Service can be changed dynamically by its administrator or vendor.

Each repository provides files describing content of the repository (package names, versions, etc.). These repository description files are downloaded to a local cache that is used by YaST. To ensure their integrity, software repositories can be signed with the GPG Key of the repository maintainer. Whenever you add a new repository, YaST offers the ability to import its key.

Warning: Trusting External Software Sources

Before adding external software repositories to your list of repositories, make sure this repository can be trusted. SUSE is not responsible for any problems arising from software installed from third-party software repositories.

10.3.1 Adding Software Repositories

You can either add repositories from DVD/CD, removable mass storage devices (such as flash disks), a local directory, an ISO image or a network source.

To add repositories from the *Configured Software Repositories* dialog in YaST proceed as follows:

1. Click *Add*.

2. Select one of the options listed in the dialog:

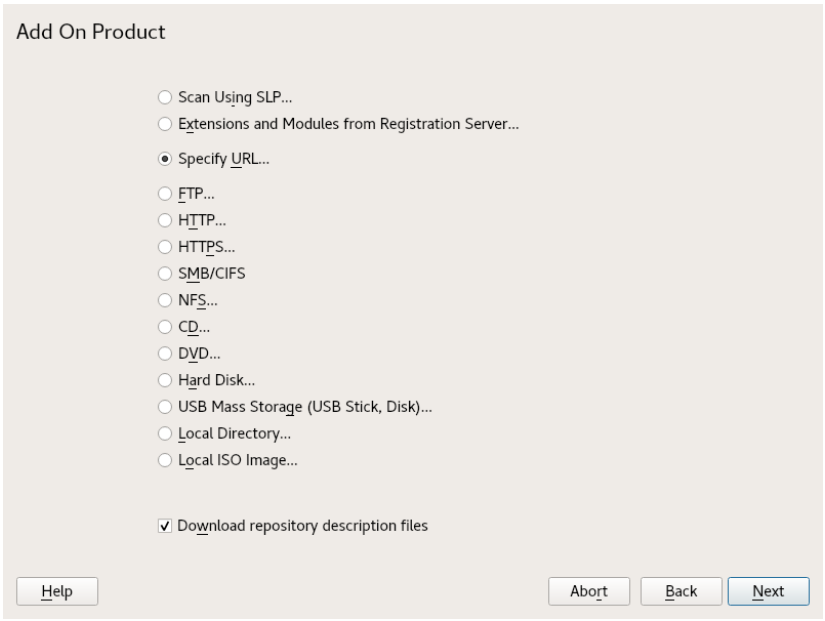


FIGURE 10.2: ADDING A SOFTWARE REPOSITORY

- To scan your network for installation servers announcing their services via SLP, select *Scan Using SLP* and click *Next*.
- To add a repository from a removable medium, choose the relevant option and insert the medium or connect the USB device to the machine, respectively. Click *Next* to start the installation.
- For the majority of repositories, you will be asked to specify the path (or URL) to the media after selecting the respective option and clicking *Next*. Specifying a *Repository Name* is optional. If none is specified, YaST will use the product name or the URL as repository name.

The option *Download Repository Description Files* is activated by default. If you deactivate the option, YaST will automatically download the files later, if needed.

3. Depending on the repository you have added, you may be prompted to import the repository's GPG key or asked to agree to a license.
After confirming these messages, YaST will download and parse the metadata. It will add the repository to the list of *Configured Repositories*.

4. If needed, adjust the repository *Properties* as described in *Section 10.3.2, "Managing Repository Properties"*.

5. Confirm your changes with *OK* to close the configuration dialog.

6. After having successfully added the repository, the software manager starts and you can install packages from this repository. For details, refer to *Chapter 10, Installing or Removing Software*.

10.3.2 Managing Repository Properties

The *Configured Software Repositories* overview of the *Software Repositories* lets you change the following repository properties:

Status

The repository status can either be *Enabled* or *Disabled*. You can only install packages from repositories that are enabled. To turn a repository off temporarily, select it and deactivate *Enable*. You can also double-click a repository name to toggle its status. To remove a repository completely, click *Delete*.

Refresh

When refreshing a repository, its content description (package names, versions, etc.) is downloaded to a local cache that is used by YaST. It is sufficient to do this once for static repositories such as CDs or DVDs, whereas repositories whose content changes often should be refreshed frequently. The easiest way to keep a repository's cache up-to-date is to choose *Automatically Refresh*. To do a manual refresh click *Refresh* and select one of the options.

Keep Downloaded Packages

Packages from remote repositories are downloaded before being installed. By default, they are deleted upon a successful installation. Activating *Keep Downloaded Packages* prevents the deletion of downloaded packages. The download location is configured in `/etc/zypp/zypp.conf`, by default it is `/var/cache/zypp/packages`.

Priority

The *Priority* of a repository is a value between `1` and `200`, with `1` being the highest priority and `200` the lowest priority. Any new repositories that are added with YaST get a priority of `99` by default. If you do not care about a priority value for a certain repository, you can also set the value to `0` to apply the default priority to that repository (`99`). If a package is available in more than one repository, then the repository with the highest priority takes precedence. This is useful to avoid downloading packages unnecessarily from the Internet by giving a local repository (for example, a DVD) a higher priority.

Important: Priority Compared to Version

The repository with the highest priority takes precedence in any case. Therefore, make sure that the update repository always has the highest priority, otherwise you might install an outdated version that will not be updated until the next online update.

Name and URL

To change a repository name or its URL, select it from the list with a single-click and then click *Edit*.

10.3.3 Managing Repository Keys

To ensure their integrity, software repositories can be signed with the GPG Key of the repository maintainer. Whenever you add a new repository, YaST offers to import its key. Verify it as you would do with any other GPG key and make sure it does not change. If you detect a key change, something might be wrong with the repository. Disable the repository as an installation source until you know the cause of the key change.

To manage all imported keys, click *GPG Keys* in the *Configured Software Repositories* dialog. Select an entry with the mouse to show the key properties at the bottom of the window. *Add*, *Edit* or *Delete* keys with a click on the respective buttons.

10.4 The GNOME Package Updater

SUSE offers a continuous stream of software security patches and updates for your product. They can be installed using tools available with your desktop or by running the *YaST Online Update* module. This section describes how to update the system from the GNOME desktop using the *Package Updater*.

Contrary to the YaST Online Update module, the GNOME *Package Updater* not only offers to install patches from the update repositories, but also new versions of packages that are already installed. (Patches fix security issues or malfunctions; the functionality and version number is usually not changed. New versions of a package increase the version number and usually add functionality or introduce major changes).

Whenever new patches or package updates are available, GNOME shows a notification in the notification area or on the lock screen.

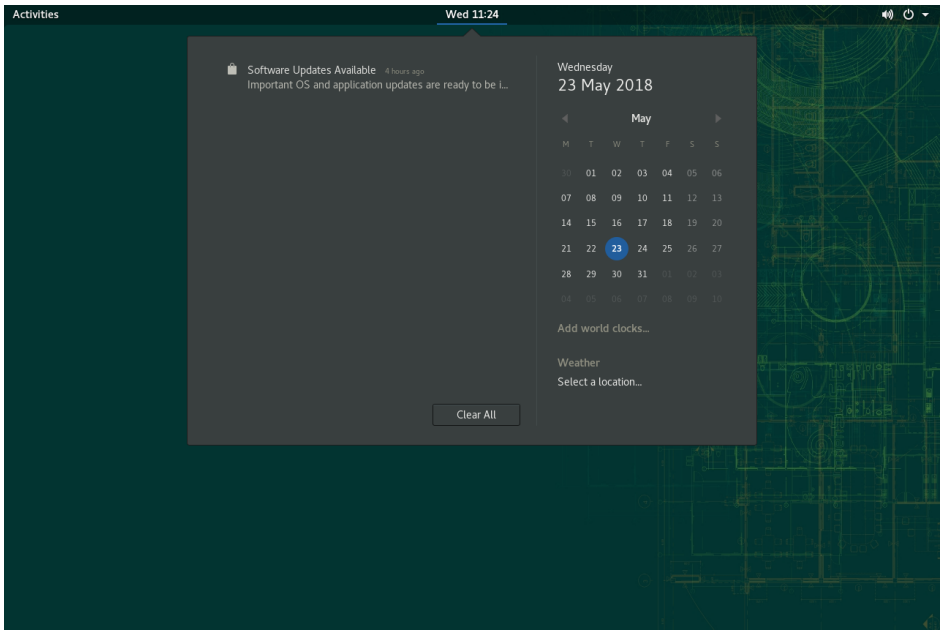
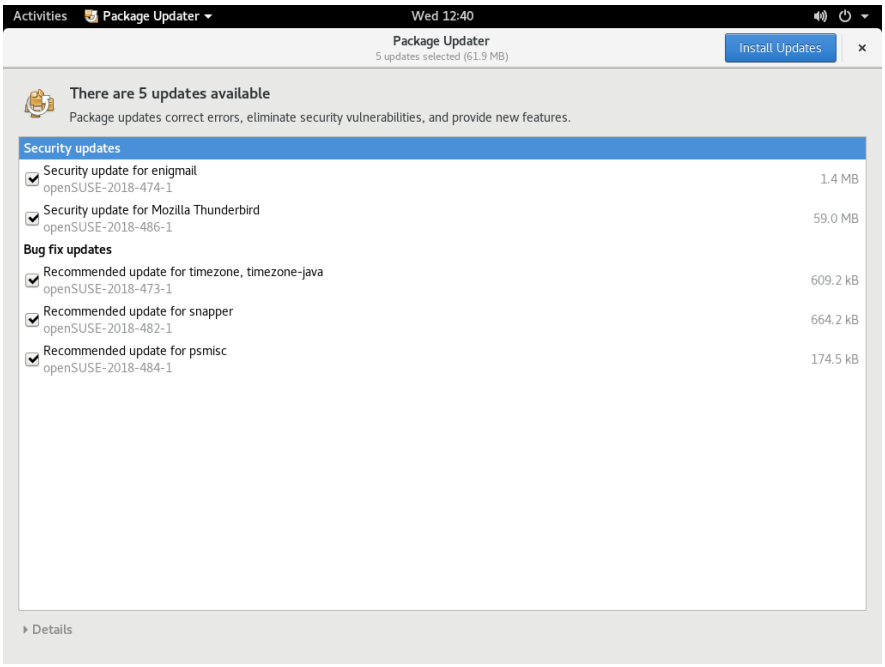


FIGURE 10.3: UPDATE NOTIFICATION ON GNOME DESKTOP

To configure the notification settings for the *Package Updater*, start GNOME *Settings* and choose *Notifications* › *Package Update*.

PROCEDURE 10.1: INSTALLING PATCHES AND UPDATES WITH THE GNOME PACKAGE UPDATER

1. To install the patches and updates, click the notification message. This opens the GNOME *Package Updater*. Alternatively, open the updater from *Activities* by typing `package U` and choosing *Package Updater*.



2. Updates are sorted into four categories:

   Security Updates (Patches)

   Fix severe security hazards and should always be installed.

   Recommended Updates (Patches)

   Fix issues that could compromise your computer. Installing them is strongly recommended.

   Optional Updates (Patches)

   Fix non-security relevant issues or provide enhancements.

   Other Updates

   New versions of packages that are installed.

   All available updates are preselected for installation. If you do not want to install all updates, deselect unwanted updates first. It is strongly recommended to always install all security and recommended updates.

   To get detailed information on an update, click its title and then *Details*. The information will be displayed in a box beneath the package list.

3. Click *Install Updates* to start the installation.

4. Some updates may require to restart the machine or to log out. Check the message that is displayed after the installation for instructions.

10.5 Updating Packages with GNOME Software

In addition to the GNOME *Package Updater*, GNOME provides *GNOME Software* which has the following functionality:

- Install, update, and remove software delivered as an RPM via PackageKit
- Install, update, and remove software delivered as a Flatpak
- Install, update, and remove GNOME shell extensions (https://extensions.gnome.org)
- Update firmware for hardware devices using *Linux Vendor Firmware Service* (LVFS, https://fwupd.org)

In addition to this, *GNOME Software* provides screenshots, ratings and reviews for software.

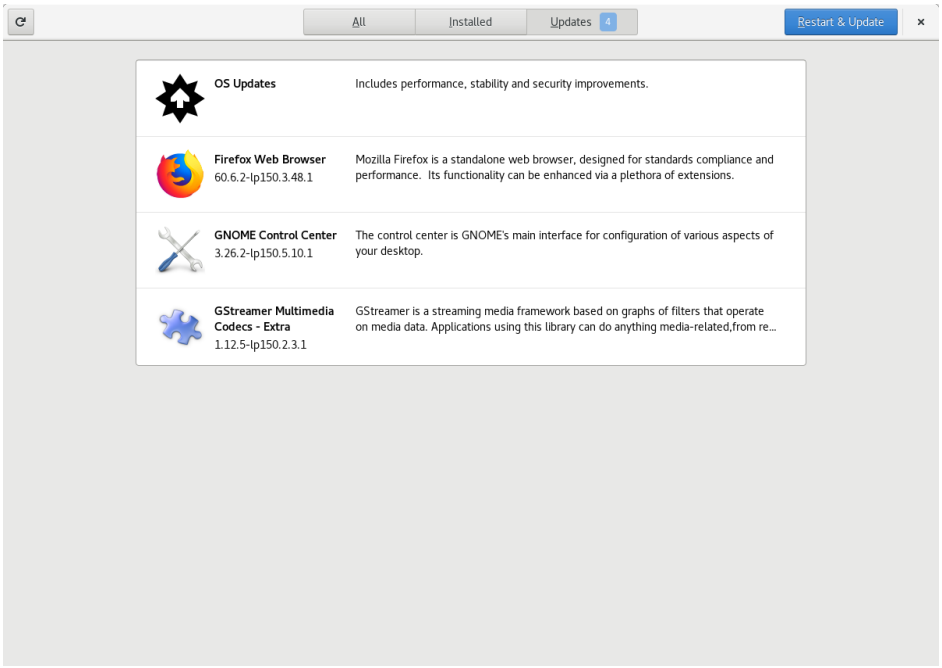


FIGURE 10.4: GNOME SOFTWARE—*UPDATES* VIEW

GNOME Software has the following differences to other tools provided on openSUSE Leap:

- Unlike YaST or Zypper, for installing software packaged as an RPM, *GNOME Software* is restricted to software that provides AppStream metadata. This includes most desktop applications.
- While the GNOME *Package Updater* updates packages within the running system (forcing you to restart the respective applications), *GNOME Software* downloads the updates but only applies them at the next reboot of the system.

11 Installing Add-On Products

Add-on products are system extensions. You can install a third party add-on product or a special system extension of openSUSE® Leap (for example, a CD with support for additional languages or a CD with binary drivers). To install a new add-on, start YaST and select *Software* › *Add-On Products*. You can select various types of product media, like CD, FTP, USB mass storage devices (such as USB flash drives or disks) or a local directory. You can also work directly with ISO files. To add an add-on as ISO file media, select *Local ISO Image* then enter the *Path to ISO Image*. The *Repository Name* is arbitrary.

11.1 Add-Ons

To install a new add-on, proceed as follows:

1. In YaST select *Software* › *Add-On Products* to see an overview of already installed add-on products.
2. To install a new add-on product, click *Add*.
3. From the list of available *Media Types* specify the type matching your repository.
4. To add a repository from a removable medium, choose the relevant option and insert the medium or connect the USB device to the machine, respectively.
5. You can choose to *Download Repository Description Files* now. If the option is deselected, YaST will automatically download the files later, if needed. Click *Next* to proceed.
6. When adding a repository from the network, enter the data you are prompted for. Continue with *Next*.
7. Depending on the repository you have added, you may be asked if you want to import the GPG key with which it is signed or asked to agree to a license.
   After confirming these messages, YaST will download and parse the metadata and add the repository to the list of *Configured Repositories.*.
8. If needed, adjust the repository *Properties* as described in *Section 10.3.2, "Managing Repository Properties"* or confirm your changes with *OK* to close the configuration dialog.

9. After having successfully added the repository for the add-on media, the software manager starts and you can install packages. Refer to *Chapter 10, Installing or Removing Software* for details.

11.2 Binary Drivers

Some hardware needs binary-only drivers to function properly. 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*.

12 YaST Online Update

SUSE offers a continuous stream of software security updates for your product. By default, the update applet is used to keep your system up-to-date. Refer to *Section 10.4, "The GNOME Package Updater"* for further information on the update applet. This chapter covers the alternative tool for updating software packages: YaST Online Update.

The current patches for openSUSE® Leap are available from an update software repository, which is automatically configured during the installation. Alternatively, you can manually add an update repository from a source you trust. To add or remove repositories, start the Repository Manager with *Software* › *Software Repositories* in YaST. Learn more about the Repository Manager in *Section 10.3, "Managing Software Repositories and Services"*.

SUSE provides updates with different relevance levels:

Security Updates

Fix severe security hazards and should always be installed.

Recommended Updates

Fix issues that could compromise your computer.

Optional Updates

Fix non-security relevant issues or provide enhancements.

12.1 The Online Update Dialog

To open the YaST *Online Update* dialog, start YaST and select *Software* › *Online Update*. Alternatively, start it from the command line with **`yast2 online_update`**.

The *Online Update* window consists of four sections.

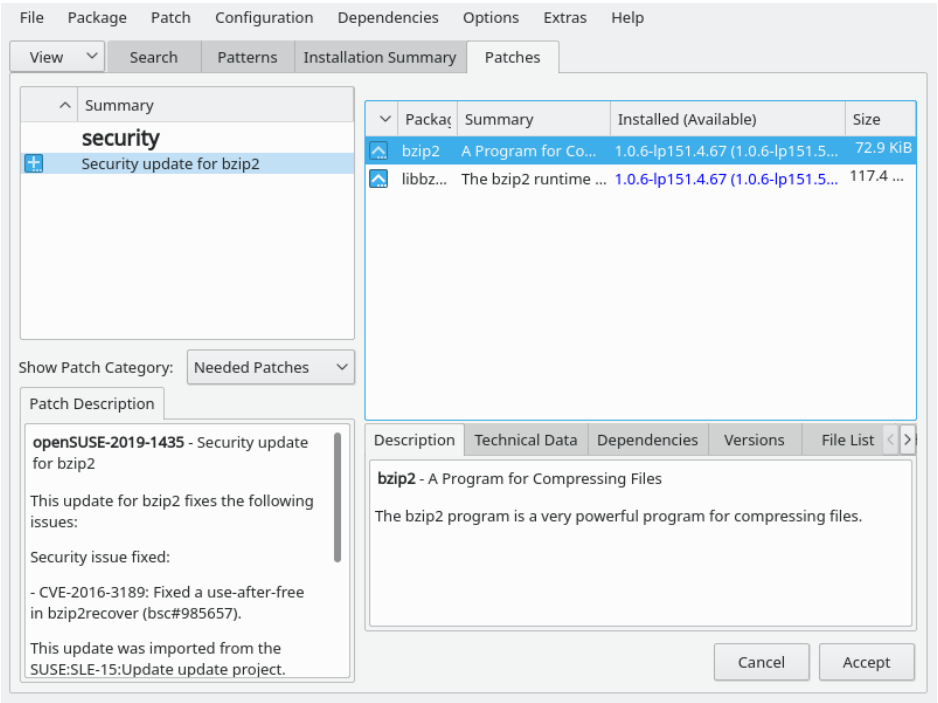


FIGURE 12.1: YAST ONLINE UPDATE

The *Summary* section on the left lists the available patches for openSUSE Leap. The patches are sorted by security relevance: `security`, `recommended`, and `optional`. You can change the view of the *Summary* section by selecting one of the following options from *Show Patch Category*:

***Needed Patches* (default view)**

Non-installed patches that apply to packages installed on your system.

Unneeded Patches

Patches that either apply to packages not installed on your system, or patches that have requirements which have already have been fulfilled (because the relevant packages have already been updated from another source).

All Patches

All patches available for openSUSE Leap.

Each list entry in the *Summary* section consists of a symbol and the patch name. For an overview of the possible symbols and their meaning, press `Shift`–`F1`. Actions required by `Security` and `Recommended` patches are automatically preset. These actions are *Autoinstall*, *Autoupdate* and *Autodelete*.

If you install an up-to-date package from a repository other than the update repository, the requirements of a patch for this package may be fulfilled with this installation. In this case a check mark is displayed in front of the patch summary. The patch will be visible in the list until you mark it for installation. This will in fact not install the patch (because the package already is up-to-date), but mark the patch as having been installed.

Select an entry in the *Summary* section to view a short *Patch Description* at the bottom left corner of the dialog. The upper right section lists the packages included in the selected patch (a patch can consist of several packages). Click an entry in the upper right section to view details about the respective package that is included in the patch.

12.2 Installing Patches

The YaST Online Update dialog allows you to either install all available patches at once or manually select the desired patches. You may also revert patches that have been applied to the system.

By default, all new patches (except `optional` ones) that are currently available for your system are already marked for installation. They will be applied automatically once you click *Accept* or *Apply*. If one or multiple patches require a system reboot, you will be notified about this before the patch installation starts. You can then either decide to continue with the installation of the selected patches, skip the installation of all patches that need rebooting and install the rest, or go back to the manual patch selection.

PROCEDURE 12.1: APPLYING PATCHES WITH YAST ONLINE UPDATE

1. Start YaST and select *Software* › *Online Update*.
2. To automatically apply all new patches (except `optional` ones) that are currently available for your system, click *Apply* or *Accept*.
3. First modify the selection of patches that you want to apply:
    a. Use the respective filters and views that the interface provides. For details, refer to *Section 12.1, “The Online Update Dialog”*.
    b. Select or deselect patches according to your needs and wishes by right-clicking the patch and choosing the respective action from the context menu.

Important: Always Apply Security Updates

Do not deselect any `security`-related patches without a very good reason. These patches fix severe security hazards and prevent your system from being exploited.

c. Most patches include updates for several packages. To change actions for single packages, right-click a package in the package view and choose an action.

d. To confirm your selection and apply the selected patches, proceed with *Apply* or *Accept*.

4. After the installation is complete, click *Finish* to leave the YaST *Online Update*. Your system is now up-to-date.

12.3 Automatic Online Update

YaST also offers the possibility to set up an automatic update with daily, weekly or monthly schedule. To use the respective module, you need to install the `yast2-online-update-configuration` package first.

By default, updates are downloaded as delta RPMs. Since rebuilding RPM packages from delta RPMs is a memory- and processor-intensive task, certain setups or hardware configurations might require you to disable the use of delta RPMs for the sake of performance.

Some patches, such as kernel updates or packages requiring license agreements, require user interaction, which would cause the automatic update procedure to stop. You can configure to skip patches that require user interaction.

PROCEDURE 12.2: CONFIGURING THE AUTOMATIC ONLINE UPDATE

1. After installation, start YaST and select *Software* › *Online Update Configuration*. Alternatively, start the module with **`yast2 online_update_configuration`** from the command line.

2. Activate *Automatic Online Update*.

3. Choose the update interval: *Daily*, *Weekly*, or *Monthly*.

4. To automatically accept any license agreements, activate *Agree with Licenses*.

5. Sometimes patches may require the attention of the administrator, for example when restarting critical services. For example, this might be an update for Docker Open Source Engine that requires all containers to be restarted. Before these patches are installed, the user is informed about the consequences and is asked to confirm the installation of the patch. Such patches are called “Interactive Patches”.
   When installing patches automatically, it is assumed that you have accepted the installation of interactive patches. If you rather prefer to review these patches before they get installed, select *Skip Interactive Patches*. In this case, interactive patches will be skipped during automated patching. Make sure to periodically run a manual online update, to check whether interactive patches are waiting to be installed.
6. To automatically install all packages recommended by updated packages, activate *Include Recommended Packages*.
7. To disable the use of delta RPMs (for performance reasons), deactivate *Use Delta RPMs*.
8. To filter the patches by category (such as security or recommended), activate *Filter by Category* and add the appropriate patch categories from the list. Only patches of the selected categories will be installed. Others will be skipped.
9. Confirm your configuration with *OK*.

The automatic online update does not automatically restart the system afterward. If there are package updates that require a system reboot, you need to do this manually.

13 Upgrading the System and System Changes

You can upgrade an existing system without completely reinstalling it. There are two types of renewing the system or parts of it: *updating individual software packages* and *upgrading the entire system*. Updating individual packages is covered in *Chapter 10, Installing or Removing Software* and *Chapter 12, YaST Online Update*. Two ways to upgrade the system are discussed in the following sections— see *Section 13.1.3, "Upgrading with YaST"* and *Section 13.1.4, "Distribution Upgrade with Zypper"*.

13.1 Upgrading the System

Important: openSUSE Leap 15.1 is only available as 64-bit version

openSUSE Leap 15.1 is only available as 64-bit version. Upgrading 32-bit installations to 64-bit is not supported. Please follow the instructions in *Chapter 1, Installation Quick Start* and *Chapter 3, Installation Steps* to install openSUSE Leap on your computer or consider switching to openSUSE Tumbleweed (https://en.opensuse.org/Portal:Tumbleweed).

The release notes are bundled in the installer, and you may also read them online at openSUSE Leap Release Notes (https://doc.opensuse.org/release-notes/).

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 you update and repartition your system. There is no general rule regarding how much space each partition should have. Space requirements depend on your particular partitioning profile, the software selected, and the version numbers of the system.

13.1.1 Preparations

Before upgrading, copy the old configuration files to a separate medium (such as removable hard disk or USB flash drive) to secure the data. This primarily applies to files stored in `/etc` as well as some of the directories and files in `/var`. 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 13.1, "List with* **`df -h`***"*, the root partition to write down is `/dev/sda3` (mounted as `/`).

EXAMPLE 13.1: LIST WITH **`df -h`**

```
Filesystem     Size  Used Avail Use% Mounted on
/dev/sda3       74G   22G   53G  29% /
udev           252M  124K  252M   1% /dev
/dev/sda5      116G  5.8G  111G   5% /home
/dev/sda1       39G  1.6G   37G   4% /windows/C
/dev/sda2      4.6G  2.6G  2.1G  57% /windows/D
```

13.1.2 Possible Problems

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

13.1.2.1 Checking passwd and group in /etc

Before upgrading 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` to eliminate any reported errors.

13.1.2.2 Shut Down Virtual Machines

If your machine serves as a VM Host Server for KVM or Xen, make sure to properly shut down all running VM Guests prior to the update. Otherwise you may not be able to access the guests after the update.

13.1.2.3 PostgreSQL

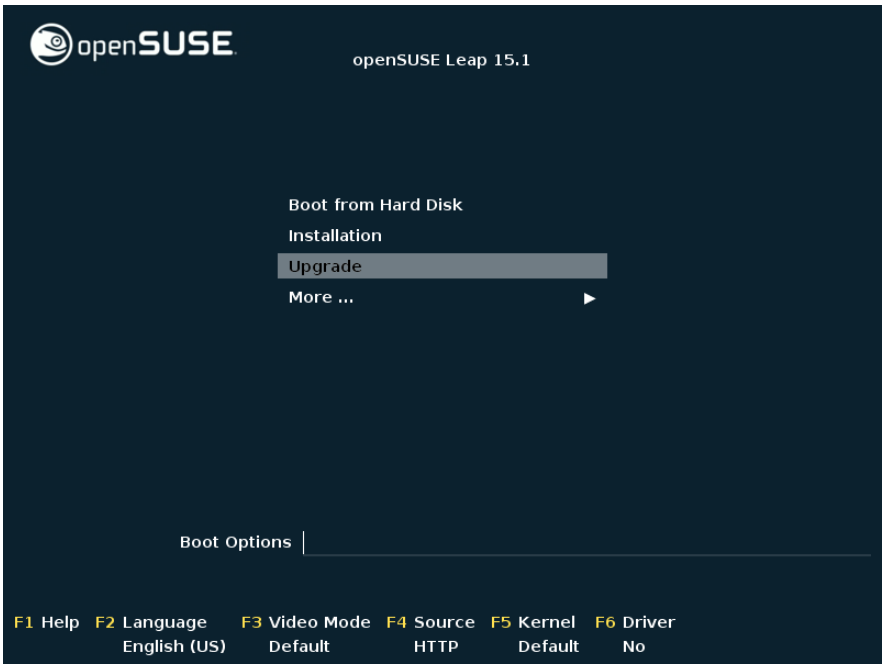
Before updating PostgreSQL (`postgres`), dump the databases. See the manual page of **`pg_dump`**. This is only necessary if you actually used PostgreSQL prior to your update.

13.1.3 Upgrading with YaST

Following the preparation procedure outlined in *Section 13.1.1, "Preparations"*, you can now upgrade your system:

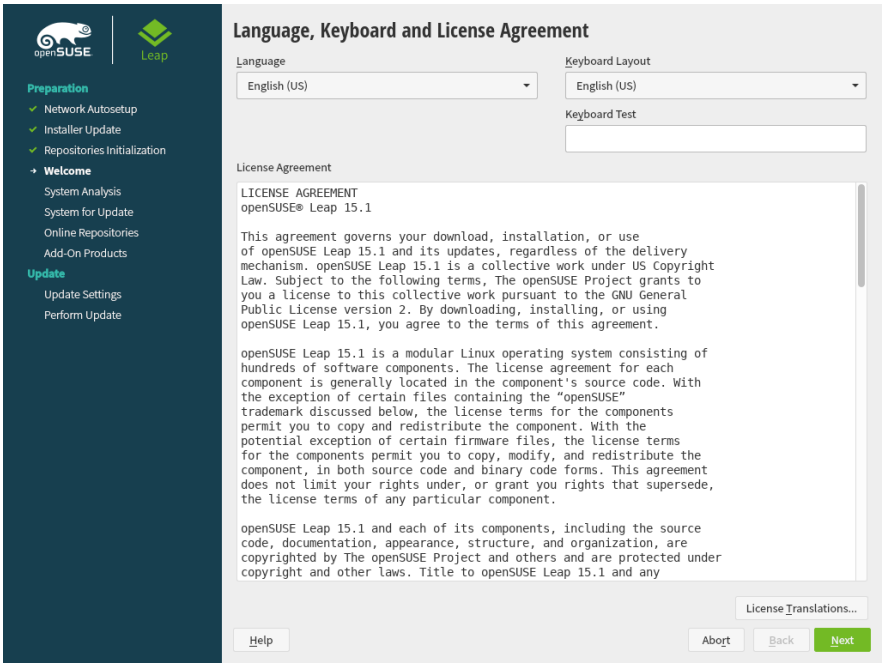
1. Insert the openSUSE Leap DVD into the drive, then reboot the computer to start the installation program. On machines with a traditional BIOS you will see the graphical boot screen shown below. On machines equipped with UEFI, a slightly different boot screen is used. Secure boot on UEFI machines is supported.

   Use F2 to change the language for the installer. A corresponding keyboard layout is chosen automatically. See *Section 2.2.1, "The Boot Screen on Machines Equipped with Traditional BIOS"* or *Section 2.2.2, "The Boot Screen on Machines Equipped with UEFI"* for more information about changing boot parameters.

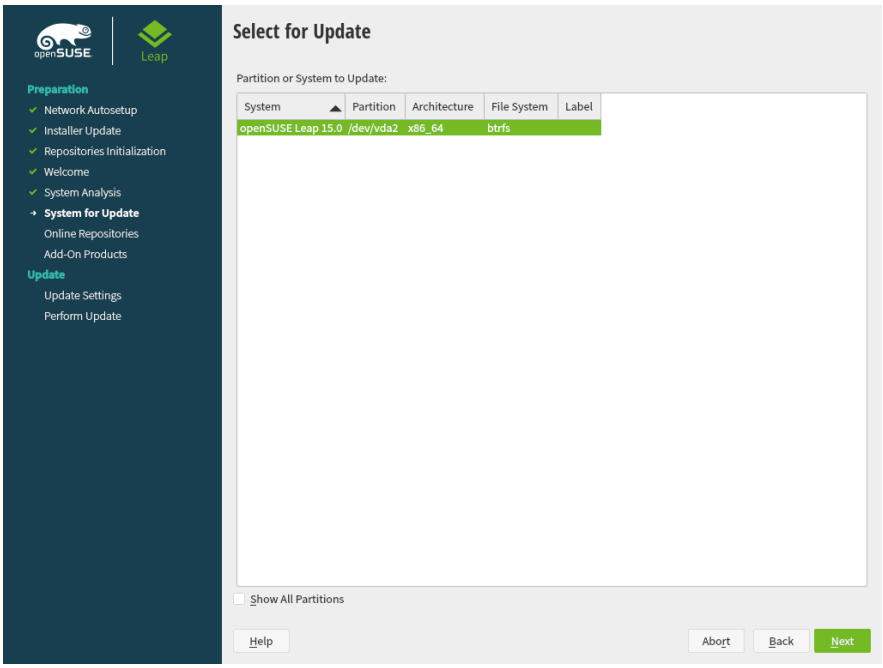


2. Select *Upgrade* on the boot screen, then press Enter. This boots the system and loads the openSUSE Leap installer. Do not select *Installation*.

3. The *Language* and *Keyboard Layout* are initialized with the language settings you have chosen on the boot screen. Change them here, if necessary.
   Read the License Agreement. It is presented in the language you have chosen on the boot screen. *License Translations* are available. Proceed with *Next*.



4. YaST determines if 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/sda3` was selected in the example in *Section 13.1.1, "Preparations"*). YaST reads the old `fstab` on this partition to analyze and mount the file systems listed there.



Tip: Release Notes

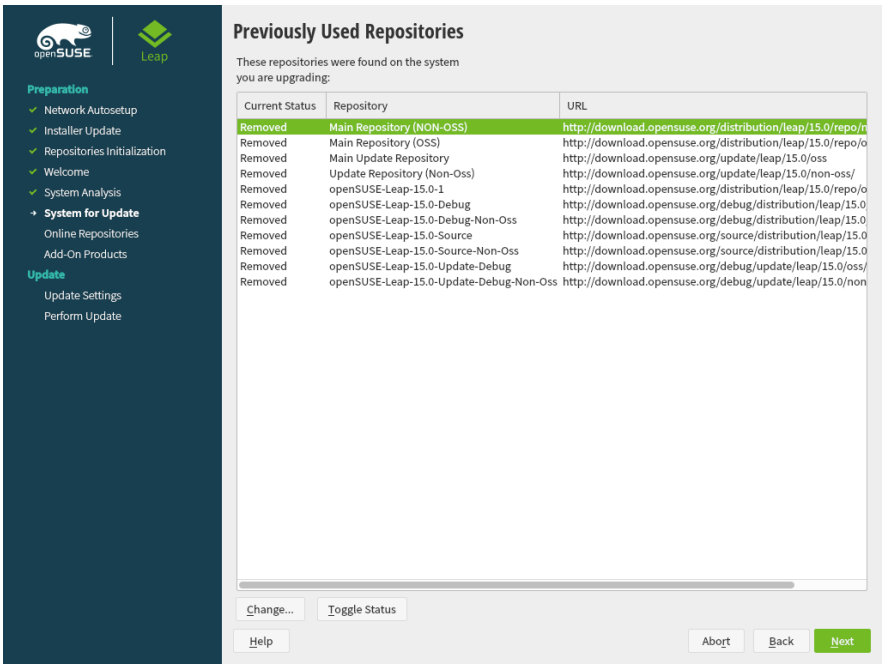
From this point on, the Release Notes can be viewed from any screen during the installation process by selecting *Release Notes*.

5. YaST shows a list of *Previously Used Repositories*. By default all repositories will get removed. If you had not added any custom repositories, do not change the settings. The packages for the upgrade will be installed from DVD and you can optionally enable the default online repositories can be chosen in the next step.
   If you have had added custom repositories, for example from the openSUSE Build Service, you have two choices:

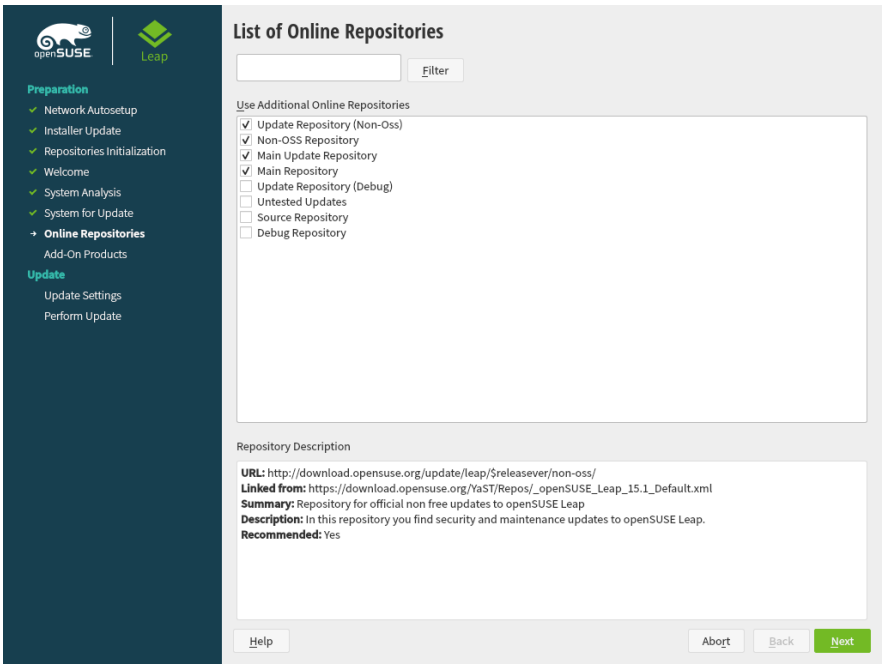
   - Leave the repository in state `Removed`. Software that was installed from this repository will get removed during the upgrade. Use this method if no version of the repository that matches the new openSUSE Leap version, is available.

   - Update and enable the repository. Use this method if a version that matches the new openSUSE Leap version is available for the repository. Change it's URL by clicking the repository in the list and then *Change*. Enable the repository afterwards by clicking *Toggle Status* until it is set to *Enable*.

Do not use repositories matching the previous version unless you are absolutely sure they will also work with the new openSUSE version. If not, the system may be unstable or not work at all.



6. In case an Internet connection is available, you may now activate optional online repositories. Please enable all repositories you had enable before to ensure all packages get upgraded correctly. Enabling the update repositories is strongly recommended—this will ensure that you get the latest package versions available, including ll security updates and fixes.



After having proceeded with *Next*, you need to confirm the license agreement for the online repositories with *Next*.

7. Use the *Installation Settings* screen to review and—if necessary—change several proposed installation settings. The current configuration is listed for each setting. To change it, click the headline.

   System

   View detailed hardware information by clicking *System*. In the resulting screen you can also change *Kernel Settings*—see *Section 3.11.6, "System"* for more information.

   Update Options

   By default, YaST will update perform full *Update with Installation of New Software and Features* based on a selection of patterns. Each pattern contains several software packages needed for specific functions (for example, Web and LAMP server or a print server).

   Here you can change the package selection or change the *Update Mode* to *Only Update Installed Packages*.

   Packages

You can further tweak the package selection on the *Packages* screen. Here you can not only select patterns but also list their contents and search for individual packages. See *Chapter 10, Installing or Removing Software* for more information.

If you intend to enhance your system, it is recommended to finish the upgrade first and then install additional software.

Backup

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

Language

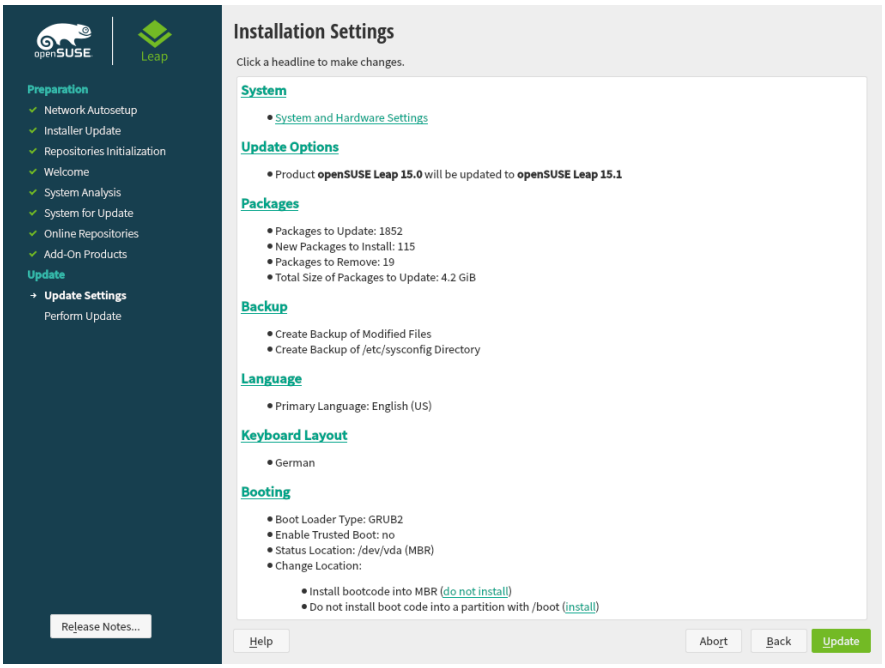
This section allows you to change the *Primary Language*primary language and configure additional *Secontry Languages.*. Optionally, you can adjust the keyboard layout and timezone to the selected primary language.

Keyboard Layout

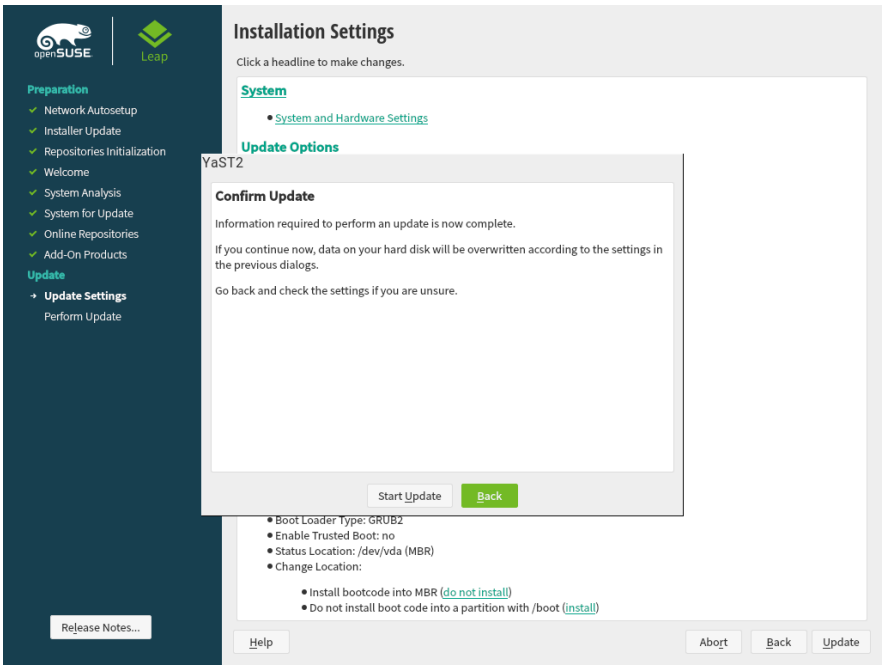
Here you can change the keyboard layout and adjust additional *Expert Keyboard Settings*.

Booting

This section shows the boot loader configuration. Changing the defaults is only recommended if really needed. Refer to *Book "*Reference*", Chapter 12 "The Boot Loader GRUB 2"* for details.



8. After you have finalized the system configuration on the *Installation Settings* screen, click *Update*. Depending on your software selection you may need to agree to license agreements before the installation confirmation screen pops up. Up to this point no changes have been made to your system. After you click *Update* a second time, the upgrade process starts.



Once the basic upgrade installation is finished, YaST reboots the system. Finally, YaST updates the remaining software, if any and displays the release notes, if wanted.

13.1.4 Distribution Upgrade with Zypper

With the `zypper` command line utility you can upgrade to the next version of the distribution. Most importantly, you can initiate the system upgrade process from within the running system.

This feature is attractive for advanced users who want to run remote upgrades or upgrades on many similarly configured systems.

13.1.4.1 Preparing the Upgrade with Zypper

To avoid unexpected errors during the upgrade process using `zypper`, minimize risky constellations.

- Quit as many applications and stop unneeded services as possible and log out all regular users.
- Disable third party repositories before starting the upgrade, or lower the priority of these repositories to make sure packages from the default system repositories will get preference. Enable them again after the upgrade and edit their version string to match the version number of the distribution of the upgraded now running system.

13.1.4.2 The Upgrade Procedure

Warning: Check Your System Backup

Before actually starting the upgrade procedure, check that your system backup is up-to-date and restorable. This is especially important because you need to enter many of the following steps manually.

The program `zypper` supports long and short command names. For example, you can abbreviate `zypper install` as `zypper in`. In the following text, the short variants are used.

1. Run the online update to make sure the software management stack is up-to-date. For more information, see *Chapter 12, YaST Online Update*.

2. Configure the repositories you want to use as update sources. Getting this right is crucial. Either use YaST (see *Section 10.3, "Managing Software Repositories and Services"*) or **zypper** (see *Book "Reference", Chapter 2 "Managing Software with Command Line Tools", Section 2.1 "Using Zypper"*). The name of the repositories used in the following steps may vary depending on your customizations.
   To view your current repositories enter:

```
tux > zypper lr -u
```

   a. Increase the version number of the system repositories from 42.3 to 15.0 `leap/`. Add the new repositories with commands such as:

```
server=http://download.example.org
tux > sudo zypper ar $server/distribution/leap/15.0/repo/oss/ Leap-15-OSS
tux > sudo zypper ar $server/update/leap/15.0/oss/ Leap-15-Update
```

      And remove the old repositories:

```
zypper rr Leap-42.3-OSS
zypper rr Leap-42.3-Update
```

      If necessary, repeat these steps for other repositories to ensure a clean upgrade path for all your packages.

   b. Disable third party repositories or other Open Build Service repositories, because **zypper dup** is guaranteed to work with the default repositories only (replace *REPO-ALIAS* with the name of the repository you want to disable):

```
tux > sudo zypper mr -d REPO-ALIAS
```

      Alternatively, you can lower the priority of these repositories.

Note: Handling of Unresolved Dependencies

zypper dup will remove all packages having unresolved dependencies, but it keeps packages of disabled repositories as long as their dependencies are satisfied.

`zypper dup` ensures that all installed packages come from one of the available repositories. It does not consider the version or architecture, but prevents changing the vendor of the installed packages by default, using the `--no-allow-vendor-change` option. Packages that are no longer available in the repositories are considered orphaned. Such packages get uninstalled if their dependencies cannot be satisfied. If they can be satisfied, such packages stay installed.

c. Once done, check your repository configuration with:

```
tux > zypper lr -d
```

3. Refresh local metadata and repository contents with `zypper ref`.

4. Update Zypper and the package management itself with `zypper patch --updatestack-only`.

5. Run the actual distribution upgrade with `zypper dup`. You are asked to confirm the license of openSUSE Leap and of some packages—depending on the set of installed packages.

6. Reboot the system with `shutdown -r now`.

13.1.5 Updating Individual Packages

Regardless of your overall updated environment, you can always update individual packages. From this point on, however, it is your responsibility to ensure that your system remains consistent.

Use the YaST software management tool to update packages as described in *Chapter 10, Installing or Removing Software*. Select components from the YaST package selection list according to your needs. If a newer version of a package exists, the version numbers of the installed and the available versions are listed in blue color in the *Installed (Available)* column. If you select a package essential for the overall operation of the system, YaST issues a warning. Such packages should be updated only in the update mode. For example, many packages contain *shared libraries*. Updating these programs and applications in the running system may lead to system instability.

13.2 Additional Information

Problems and special issues of the various versions are published online as they are identified. See the links listed below. Important updates of individual packages can be accessed using the YaST Online Update. For more information, see *Chapter 12, YaST Online Update*.

Refer to the `Product highlights` (http://en.opensuse.org/Product_highlights) and the `Bugs` article in the openSUSE wiki at http://en.opensuse.org/openSUSE:Most_annoying_bugs for information about recent changes and issues.

IV The Bash Shell

14 Shell Basics

When working with Linux, you can communicate with the system almost without ever requiring a command line interpreter (the shell). After 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 operating system. The graphical user interface in Linux is initially configured during installation and used by desktop environments such as KDE or GNOME.

Nevertheless, it is useful to have some basic knowledge of working with a shell because you might encounter situations where the graphical user interface is not available. For example, if some problem with the X Window System occurs. If you are not familiar with a shell, you might feel a bit uncomfortable at first when entering commands, but the more you get used to it, the more you will realize that the command line is often the quickest and easiest way to perform some daily tasks.

For Unix or Linux, several shells are available which differ slightly in behavior and in the commands they accept. The default shell in openSUSE® Leap is Bash (GNU Bourne-Again Shell).

The following sections will guide you through your first steps with the Bash shell and will show you how to complete some basic tasks via the command line. If you are interested in learning more or rather feel like a shell “power user” already, refer to *Chapter 15, Bash and Bash Scripts*.

14.1 Starting a Shell

Basically, there are two different ways to start a shell from the graphical user interface which usually shows after you have booted your computer:

- you can leave the graphical user interface or
- you can start a terminal window *within* the graphical user interface.

While the first option is always available, you can only make use of the second option when you are already logged in to a desktop such as KDE or GNOME. Whichever way you choose, there is always a way back and you can switch back and forth between the shell and the graphical user interface.

If you want to give it a try, press Ctrl–Alt–F2 to leave the graphical user interface. The graphical user interface disappears and you are taken to a shell which prompts you to log in. Type your username and press Enter. Then type your password and press Enter. The prompt now changes and shows some useful information as in the following example:

```
  ❶    ❷    ❸
tux@linux:~>
```

❶ Your login.

❷ The hostname of your computer.

❸ Path to the current directory. Directly after login, the current directory usually is your home directory, indicated by the ~ symbol (tilde) .

When you are logged in at a remote computer the information provided by the prompt always shows you which system you are currently working on.

When the cursor is located behind this prompt, you can pass commands directly to your computer system. For example, you can now enter **ls** -l to list the contents of the current directory in a detailed format. If this is enough for your first encounter with the shell and you want to go back to the graphical user interface, you should log out from your shell session first. To do so, type **exit** and press Enter. Then press Alt–F7 to switch back to the graphical user interface. You will find your desktop and the applications running on it unchanged.

When you are already logged in to the GNOME or the KDE desktop and want to start a terminal window within the desktop, press Alt–F2 and enter **konsole** (for KDE) or **gnome-terminal** (for GNOME). This opens a terminal window on your desktop. As you are already logged in to your desktop, the prompt shows information about your system as described above. You can now enter commands and execute tasks just like in any shell which runs parallel to your desktop. To switch to another application on the desktop just click on the corresponding application window or select it from the taskbar of your panel. To close the terminal window press Alt–F4.

14.2 Entering Commands

As soon as the prompt appears on the shell it is ready to receive and execute commands. A command can consist of several elements. The first element is the actual command, followed by parameters or options. You can type a command and edit it by using the following keys: ←, →, Home, End, <– (Backspace), Del, and Space. You can correct typing errors or add options. The command is not executed until 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 —or if you explicitly ask for them by executing a command with a certain option.

Also keep this in mind for commands to delete objects. Before entering a command like `rm` (without any option) for removing a file, you should know if you really want to get rid of the object: it will be deleted irretrievably, without confirmation.

14.2.1 Using Commands without Options

In *Section 14.6.1, "Permissions for User, Group and Others"* you already got to know one of the most basic commands: `ls`, which used to list the contents of a directory. This command can be used with or without options. Entering the plain `ls` command shows the contents of the current directory:

```
tux > ls
bin Desktop Documents public_html tux.txt
tux >
```

Files in Linux may have a file extension or a suffix, 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 in the Bash shell give you a hint: directories are usually shown in blue, files in black.

14.2.2 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 carry out specific tasks. Options are separated from the command with a blank and are usually prefixed with a hyphen. The `ls -l` command shows the contents of the same directory in full detail (long listing format):

```
tux > ls -l
drwxr-xr-x 1 tux users     48 2015-06-23 16:08 bin
drwx---r-- 1 tux users  53279 2015-06-21 13:16 Desktop
```

```
drwx------ 1 tux users    280 2015-06-23 16:08 Documents
drwxr-xr-x 1 tux users  70733 2015-06-21 09:35 public_html
-rw-r--r-- 1 tux users  47896 2015-06-21 09:46 tux.txt
tux >
```

This output shows the following information about each object:

```
drwxr-xr-x❶ 1❷ tux❸ users❹ 48❺ 2006-06-23 16:08❻ bin❼
```

❶ Type of object and access permissions. For further information, refer to *Section 14.6.1, "Permissions for User, Group and Others"*.

❷ Number of hard links to this file.

❸ Owner of the file or directory. For further information, refer to *Section 14.6.1, "Permissions for User, Group and Others"*.

❹ Group assigned to the file or directory. For further information, refer to *Section 14.6.1, "Permissions for User, Group and Others"*.

❺ File size in bytes.

❻ Date and time of the last change.

❼ Name of the object.

Usually, you can combine several options by prefixing only the first option with a hyphen and then write the others consecutively without a blank. For example, if you want to see all files in a directory in long listing format, you can combine the two options `-l` and `-a` (show all files) for the **`ls`** command. Executing **`ls`** `-la` shows also hidden files in the directory, indicated by a dot in front (for example, `.hiddenfile`).

The list of contents you get with **`ls`** is sorted alphabetically by filenames. But like in a graphical file manager, you can also sort the output of **`ls`** `-l` according to various criteria such as date, file extension or file size:

- For date and time, use **`ls`** `-lt` (displays newest first).
- For extensions, use **`ls`** `-lx` (displays files with no extension first).
- For file size, use **`ls`** `-lS` (displays largest first).

To revert the order of sorting, add `-r` as an option to your **`ls`** command. For example, **`ls`** `-lr` gives you the contents list sorted in reverse alphabetical order, **`ls`** `-ltr` shows the oldest files first. There are lots of other useful options for **`ls`**. In the following section you will learn how to investigate them.

14.2.3 Bash Shortcut Keys

After having entered several commands, your shell will begin to fill up with all sorts of commands and the corresponding outputs. In the following table, find some useful shortcut keys for navigating and editing in the shell.

Shortcut Key	Function
Ctrl – L	Clears the screen and moves the current line to the top of the page.
Ctrl – C	Aborts the command which is currently being executed.
Shift – Page ↑	Scrolls upwards.
Shift – Page ↓	Scrolls downwards.
Ctrl – U	Deletes from cursor position to start of line.
Ctrl – K	Deletes from cursor position to the end of line.
Ctrl – D	Closes the shell session.
↑, ↓	Browses in the history of executed commands.

14.3 Getting Help

If you remember the name of command but are not sure about the options or the syntax of the command, choose one of the following possibilities:

`--help` / `-h` option

If you only want to look up the options of a certain command, try entering the command followed by a space and `--help`. This `--help` option exists for many commands. For example, `ls --help` displays all the options for the `ls` command.

Manual Pages

To learn more about the various commands, you can also use the manual pages. Manual pages also give a short description of what the command does. They can be accessed with **man** followed by the name of the command, for example, **man ls**.
Man pages are displayed directly in the shell. To navigate them, use the following keys:

- Move up and down with Page ↑ and Page ↓
- Move between the beginning and the end of a document with Home and End
- Quit the man page viewer by pressing Q

For more information about the **man** command, use **man man**.

Info Pages

Info pages usually provide even more information about commands. To view the info page for a certain command, enter **info** followed by the name of the command (for example, **info ls**).
Info pages are displayed directly in the shell. To navigate them, use the following keys:

- Use Space to move forward a section (*node*). Use <– to move backward a section.
- Move up and down with Page ↑ and Page ↓
- Quit the info page viewer by pressing Q

Note that man pages and info pages do not exist for all commands. Sometimes both are available (usually for key commands), sometimes only a man page or an info page exists, and sometimes neither of them are available.

14.4 Working with Files and Directories

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:

Absolute Path

The entire path from the root directory (`/`) to the relevant file or directory. For example, the absolute path to a text file named `file.txt` in your `Documents` directory might be:

```
/home/tux/Documents/file.txt
```

Relative Path

The path from the current working directory to the relevant file or directory. If your current working directory is `/home/tux`, the relative path `file.txt` in your `Documents` directory is:

```
Documents/file.txt
```

However, if your working directory is `/home/tux/Music` instead, you need to move up a level to `/home/tux` (with `..`) before you can go further down:

```
../Documents/file.txt
```

Paths contain file names, directories or both, separated by slashes. Absolute paths always start with a slash. Relative paths do not have a slash at the beginning, but can have one or two dots.

When entering commands, you can choose either way to specify a path, depending on your preferences or the amount of typing, both will lead to the same result. To change directories, use the **`cd`** command and specify the path to the directory.

Note: Handling Blanks in Filenames or Directory Names

If a filename or the name of a directory contains a space, either escape the space using a back slash (`\`) in front of the blank or enclose the filename in single quotes. Otherwise Bash interprets a filename like `My Documents` as the names of two files or directories, `My` and `Documents` in this case.

When specifying paths, the following “shortcuts” can save you a lot of typing:

- The tilde symbol (`~`) is a shortcut for home directories. For example, to list the contents of your home directory, use **`ls`** `~`. To list the contents of another user's home directory, enter **`ls`** *`~USERNAME`* (or course, this will only work if you have permission to view the contents, see *Section 14.6, “File Access Permissions”*). For example, entering **`ls ~tux`** would list the contents of the home directory of a user named `tux`. You can use the tilde symbol as shortcut for home directories also if you are working in a network environment where your home directory may not be called `/home` but can be mapped to any directory in the file system.

From anywhere in the file system, you can reach your home directory by entering `cd ~` or by simply entering `cd` without any options.

- When using relative paths, refer to the current directory with a dot (`.`). This is mainly useful for commands such as `cp` or `mv` by which you can copy or move files and directories.
- The next higher level in the tree is represented by two dots (`..`). In order to switch to the parent directory of your current directory, enter `cd ..`, to go up two levels from the current directory enter `cd ../..` etc.

To apply your knowledge, find some examples below. They address basic tasks you may want to execute with files or folders using Bash.

14.4.1 Examples for Working with Files and Directories

Suppose you want to copy a file located somewhere in your home directory to a subdirectory of `/tmp` that you need to create first.

PROCEDURE 14.1: CREATING AND CHANGING DIRECTORIES

From your home directory create a subdirectory in `/tmp`:

1. Enter

   ```
   tux > mkdir /tmp/test
   ```

   `mkdir` stands for "make directory". This command creates a new directory named `test` in the `/tmp` directory. In this case, you are using an absolute path to create the `test` directory.

2. To check what happened, now enter

   ```
   tux > ls -l /tmp
   ```

   The new directory `test` should appear in the list of contents of the `/tmp` directory.

3. Switch to the newly created directory with

   ```
   tux > cd /tmp/test
   ```

PROCEDURE 14.2: CREATING AND COPYING FILES

Now create a new file in a subdirectory of your home directory and copy it to `/tmp/test`. Use a relative path for this task.

Important: Overwriting of Existing Files

Before copying, moving or renaming a file, check if your target directory already contains a file with the same name. If yes, consider changing one of the filenames or use **cp** or **mv** with options like `-i`, which will prompt before overwriting an existing file. Otherwise Bash will overwrite the existing file without confirmation.

1. To list the contents of your home directory, enter

   ```
   tux > ls -l ~
   ```

   It should contain a subdirectory called `Documents` by default. If not, create this subdirectory with the **mkdir** command you already know:

   ```
   tux > mkdir ~/Documents
   ```

2. To create a new, empty file named `myfile.txt` in the `Documents` directory, enter

   ```
   tux > touch ~/Documents/myfile.txt
   ```

   Usually, the **touch** command updates the modification and access date for an existing file. If you use **touch** with a filename which does not exist in your target directory, it creates a new file.

3. Enter

   ```
   tux > ls -l ~/Documents
   ```

   The new file should appear in the list of contents.

4. To copy the newly created file, enter

   ```
   tux > cp ~/Documents/myfile.txt .
   ```

   Do not forget the dot at the end.

   This command tells Bash to go to your home directory and to copy `myfile.txt` from the `Documents` subdirectory to the current directory, `/tmp/test`, without changing the name of the file.

5. Check the result by entering

   ```
   tux > ls -l
   ```

The file `myfile.txt` should appear in the list of contents for `/tmp/test`.

PROCEDURE 14.3: RENAMING AND REMOVING FILES OR DIRECTORIES

Now suppose you want to rename `myfile.txt` into `tuxfile.txt`. Finally you decide to remove the renamed file and the `test` subdirectory.

1. To rename the file, enter

```
tux > mv myfile.txt tuxfile.txt
```

2. To check what happened, enter

```
tux > ls -l
```

Instead of `myfile.txt`, `tuxfile.txt` should appear in the list of contents.
`mv` stands for `move` and is used with two options: the first option specifies the source, the second option specifies the target of the operation. You can use `mv` either

- to rename a file or a directory,
- to move a file or directory to a new location or
- to do both in one step.

3. Coming to the conclusion that you do not need the file any longer, you can delete it by entering

```
tux > rm tuxfile.txt
```

Bash deletes the file without any confirmation.

4. Move up one level with `cd ..` and check with

```
tux > ls -l test
```

if the `test` directory is empty now.

5. If yes, you can remove the `test` directory by entering

```
tux > rmdir test
```

14.5 Becoming Root

`root`, also called the superuser, has privileges which authorize them to access all parts of the system and to execute administrative tasks. They have the unrestricted capacity to make changes to the system and they have unlimited access to all files. Therefore, performing some administrative tasks or running certain programs such as YaST requires `root` permissions.

14.5.1 Using **su**

In order to temporarily become `root` in a shell, proceed as follows:

1. Enter **`su`**. You are prompted for the `root` password.
2. Enter the password. If you mistyped the `root` password, the shell displays a message. In this case, you have to re-enter **`su`** before retyping the password. If your password is correct, a hash symbol `#` appears at the end of the prompt, signaling that you are acting as `root` now.
3. Execute your task. For example, transfer ownership of a file to a new user which only `root` is allowed to do:

   ```
   tux > chown wilber kde_quick.xml
   ```
4. After having completed your tasks as `root`, switch back to your normal user account. To do so, enter

   ```
   tux > exit
   ```

   The hash symbol disappears and you are acting as “normal” user again.

14.5.2 Using **sudo**

Alternatively, you can also use **`sudo`** (superuser “do”) to execute some tasks which normally are for `root`s only. With sudo, administrators can grant certain users `root` privileges for some commands. Depending on the system configuration, users can then run `root` commands by entering their normal password only. Due to a timestamp function, users are only granted a “ticket” for a restricted period of time after having entered their password. The ticket usually expires after a few minutes. In openSUSE, sudo requires the `root` password by default (if not configured otherwise by your system administrator).

For users, sudo is convenient as it prevents you from switching accounts twice (to `root` and back again). To change the ownership of a file using sudo, only one command is necessary instead of three:

```
tux > sudo chown wilber kde_quick.xml
```

After you have entered the password which you are prompted for, the command is executed. If you enter a second `root` command shortly after that, you are not prompted for the password again, because your ticket is still valid. After a certain amount of time, the ticket automatically expires and the password is required again. This also prevents unauthorized persons from gaining `root` privileges in case a user forgets to switch back to their normal user account again and leaves a `root` shell open.

14.6 File Access Permissions

In Linux, objects such as files or folders or processes generally belong to the user who created or initiated them. There are some exceptions to this rule. For more information about the exceptions, refer to *Book "Security Guide", Chapter 10 "Access Control Lists in Linux"*. The group which is associated with a file or a folder depends on the primary group the user belongs to when creating the object.

When you create a new file or directory, initial access permissions for this object are set according to a predefined scheme. As an owner of a file or directory, you can change the access permissions for this object. For example, you can protect files holding sensitive data against read access by other users and you can authorize the members of your group or other users to write, read, or execute several of your files where appropriate. As `root`, you can also change the ownership of files or folders.

14.6.1 Permissions for User, Group and Others

Three permission sets are defined for each file object on a Linux system. These sets include the read, write, and execute permissions for each of three types of users—the owner, the group, and other users.

The following example shows the output of an **`ls`** `-l` command in a shell. This command lists the contents of a directory and shows the details for each file and folder in that directory.

EXAMPLE 14.1: ACCESS PERMISSIONS FOR FILES AND FOLDERS

```
-rw-r----- 1 tux users      0 2015-06-23 16:08 checklist.txt
-rw-r--r-- 1 tux users  53279 2015-06-21 13:16 gnome_quick.xml
-rw-rw---- 1 tux users      0 2015-06-23 16:08 index.htm
-rw-r--r-- 1 tux users  70733 2015-06-21 09:35 kde-start.xml
-rw-r--r-- 1 tux users  47896 2015-06-21 09:46 kde_quick.xml
drwxr-xr-x 2 tux users     48 2015-06-23 16:09 local
-rwxr--r-- 1 tux users 624398 2015-06-23 15:43 tux.sh
```

As shown in the third column, all objects belong to user `tux`. They are assigned to the group `users` which is the primary group the user `tux` belongs to. To retrieve the access permissions the first column of the list must be examined more closely. Let's have a look at the file `kde-start.xml`:

Type	User Permissions	Group Permissions	Permissions for Others
`-`	`rw-`	`r--`	`r--`

The first column of the list consists of one leading character followed by nine characters grouped in three blocks. The leading character indicates the file type of the object: in this case, the hyphen (`-`) shows that `kde-start.xml` is a file. If you find the character `d` instead, this shows that the object is a directory, like `local` in *Example 14.1, "Access Permissions For Files and Folders"*.

The next three blocks show the access permissions for the owner, the group and other users (from left to right). Each block follows the same pattern: the first position shows read permissions (`r`), the next position shows write permissions (`w`), the last one shows execute permission (`x`). A lack of either permission is indicated by `-`. In our example, the owner of `kde-start.xml` has read and write access to the file but cannot execute it. The `users` group can read the file but cannot write or execute it. The same holds true for the other users as shown in the third block of characters.

14.6.2 Files and Folders

Access permissions have a slightly different impact depending on the type of object they apply to: file or directory. The following table shows the details:

TABLE 14.1: ACCESS PERMISSIONS FOR FILES AND DIRECTORIES

Access Permission	File	Folder
Read (r)	Users can open and read the file.	Users can view the contents of the directory. Without this permission, users cannot list the contents of this directory with `ls -l`, for example. However, if they only have execute permission for the directory, they can nevertheless access certain files in this directory if they know of their existence.
Write (w)	Users can change the file: They can add or drop data and can even delete the contents of the file. However, this does not include the permission to remove the file completely from the directory as long as they do not have write permissions for the directory where the file is located.	Users can create, rename or delete files in the directory.
Execute (x)	Users can execute the file. This permission is only relevant for files like programs or shell scripts, not for text files. If the operating	Users can change into the directory and execute files there. If they do not have read access to that directory they cannot list

Access Permission	File	Folder
	system can execute the file directly, users do not need read permission to execute the file. However, if the file must me interpreted like a shell script or a perl program, additional read permission is needed.	the files but can access them nevertheless if they know of their existence.

Note that access to a certain file is always dependent on the correct combination of access permissions for the file itself *and* the directory it is located in.

14.6.3 Modifying File Permissions

In Linux, objects such as files or folder or processes generally belong to the user who created or initiated them. The group which is associated with a file or a folder depends on the primary group the user belongs to when creating the object. When you create a new file or directory, initial access permissions for this object are set according to a predefined scheme. For further details refer to *Section 14.6, "File Access Permissions"*.

As the owner of a file or directory (and, of course, as `root`), you can change the access permissions to this object.

To change object attributes like access permissions of a file or folder, use the **`chmod`** command followed by the following parameters:

- the users for which to change the permissions,
- the type of access permission you want to remove, set or add and
- the files or folders for which you want to change permissions separated by spaces.

The users for which you can change file access permissions fall into the following categories: the owner of the file (user, `u`), the group that own the file (group, `g`) and the other users (others, `o`). You can add, remove or set one or more of the following permissions: read, write or execute.

As `root`, you can also change the ownership of a file: with the command **`chown`** (change owner) you can transfer ownership to a new user.

14.6.3.1 Examples for Changing Access Permissions and Ownership

The following example shows the output of an `ls -l` command in a shell.

EXAMPLE 14.2: ACCESS PERMISSIONS FOR FILES AND FOLDERS

```
-rw-r----- 1 tux users      0 2015-06-23 16:08 checklist.txt
-rw-r--r-- 1 tux users  53279 2015-06-21 13:16 gnome_quick.xml
-rw-rw---- 1 tux users      0 2015-06-23 16:08 index.htm
-rw-r--r-- 1 tux users  70733 2015-06-21 09:35 kde-start.xml
-rw-r--r-- 1 tux users  47896 2015-06-21 09:46 kde_quick.xml
drwxr-xr-x 2 tux users     48 2015-06-23 16:09 local
-r-xr-xr-x 1 tux users 624398 2015-06-23 15:43 tux.jpg
```

In the example above, user `tux` owns the file `kde-start.xml` and has read and write access to the file but cannot execute it. The `users` group can read the file but cannot write or execute it. The same holds true for the other users as shown by the third block of characters.

PROCEDURE 14.4: CHANGING ACCESS PERMISSIONS

Suppose you are `tux` and want to modify the access permissions to your files:

1. If you want to grant the `users` group also write access to `kde-start.xml`, enter

   ```
   tux > chmod g+w kde-start.xml
   ```

2. To grant the `users` group and other users write access to `kde-start.xml`, enter

   ```
   tux > chmod go+w kde-start.xml
   ```

3. To remove write access for all users, enter

   ```
   tux > chmod -w kde-start.xml
   ```

   If you do not specify any kind of users, the changes apply to all users— the owner of the file, the owning group and the others. Now even the owner `tux` does not have write access to the file without first reestablishing write permissions.

4. To prohibit the `users` group and others to change into the directory `local`, enter

   ```
   tux > chmod go-x local
   ```

5. To grant others write permissions for two files, for `kde_quick.xml` and `gnome_quick.xml`, enter

   ```
   tux > chmod o+w  kde_quick.xml gnome_quick.xml
   ```

PROCEDURE 14.5: CHANGING OWNERSHIP

Suppose you are `tux` and want to transfer the ownership of the file `kde_quick.xml` to an other user, named `wilber`. In this case, proceed as follows:

1. Enter the username and password for `root`.
2. Enter

```
root # chown wilber kde_quick.xml
```

3. Check what happened with

```
tux > ls -l kde_quick.xml
```

You should get the following output:

```
-rw-r--r-- 1 wilber users  47896 2006-06-21 09:46 kde_quick.xml
```

4. If the ownership is set according to your wishes, switch back to your normal user account.

14.7 Time-Saving Features of Bash

Entering commands in Bash can involve a lot of typing. This section introduces some features that can save you both time and typing.

History

By default, Bash "remembers" commands you have entered. This feature is called *history*. You can browse through commands that have been entered before, select one you want to repeat and then execute it again. To do so, press ↑ repeatedly until the desired command appears at the prompt. To move forward through the list of previously entered commands, press ↓. For easier repetition of a certain command from Bash history, just type the first letter of the command you want to repeat and press Page ↑.

You can now edit the selected command (for example, change the name of a file or a path), before you execute the command by pressing Enter. To edit the command line, move the cursor to the desired position using the arrow keys and start typing.

You can also search for a certain command in the history. Press Ctrl–R to start an incremental search function. showing the following prompt:

```
tux > (reverse-i-search)`':
```

Just type one or several letters from the command you are searching for. Each character you enter narrows down the search. The corresponding search result is shown on the right side of the colon whereas your input appears on the left of the colon. To accept a search result, press Esc. The prompt now changes to its normal appearance and shows the command you chose. You can now edit the command or directly execute it by pressing Enter.

Completion

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 →| (Tabulator). 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 it becomes ambiguous again. 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 →|, 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

You can replace one or more characters in a filename with a wild card for pathname expansion. Wild cards are characters that can stand for other characters. There are three different types of these in Bash:

Wild Card	Function
?	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*.

14.7.1 Examples For Using History, Completion and Wildcards

The following examples illustrate how to make use of these convenient features of Bash.

PROCEDURE 14.6: USING HISTORY AND COMPLETION

If you already did the example *Section 14.4.1, "Examples for Working with Files and Directories"*, your shell buffer should be filled with commands which you can retrieve using the history function.

1. Press ↑ repeatedly until **cd ~** appears.
2. Press Enter to execute the command and to switch to your home directory.
   By default, your home directory contains two subdirectories starting with the same letter, `Documents` and `Desktop`.
3. Type **cd D** and press →|.
   Nothing happens since Bash cannot identify to which one of the subdirectories you want to change.
4. Press →| again to see the list of possible choices:

```
tux > cd D
Desktop/ Documents/ Downloads/
tux > cd D
```

5. The prompt still shows your initial input. Type the next character of the subdirectory you want to go to and press →| again.
   Bash now completes the path.
6. You can now execute the command with Enter.

PROCEDURE 14.7: USING WILDCARDS

Now suppose that your home directory contains several files with various file extensions. It also holds several versions of one file which you saved under different filenames `myfile1.txt`, `myfile2.txt` etc. You want to search for certain files according to their properties.

1. First, create some test files in your home directory:
   a. Use the **touch** command to create several (empty) files with different file extensions, for example `.pdf`, `.xml` and `.jpg`.

You can do this consecutively (do not forget to use the Bash history function) or with only one **touch** command: simply add several filenames separated by a space.

b. Create at least two files that have the same file extension, for example `.html`.

c. To create several "versions" of one file, enter

```
tux > touch myfile{1..5}.txt
```

This command creates five consecutively numbered files: `myfile1.txt`, ..., `myfile5.txt`.

d. List the contents of the directory. It should look similar to this:

```
tux > ls -l
-rw-r--r-- 1 tux users   0 2006-07-14 13:34 foo.xml
-rw-r--r-- 1 tux users   0 2006-07-14 13:47 home.html
-rw-r--r-- 1 tux users   0 2006-07-14 13:47 index.html
-rw-r--r-- 1 tux users   0 2006-07-14 13:47 toc.html
-rw-r--r-- 1 tux users   0 2006-07-14 13:34 manual.pdf
-rw-r--r-- 1 tux users   0 2006-07-14 13:49 myfile1.txt
-rw-r--r-- 1 tux users   0 2006-07-14 13:49 myfile2.txt
-rw-r--r-- 1 tux users   0 2006-07-14 13:49 myfile3.txt
-rw-r--r-- 1 tux users   0 2006-07-14 13:49 myfile4.txt
-rw-r--r-- 1 tux users   0 2006-07-14 13:49 myfile5.txt
-rw-r--r-- 1 tux users   0 2006-07-14 13:32 tux.png
```

2. With wild cards, select certain subsets of the files according to various criteria:

a. To list all files with the `.html` extension, enter

```
tux > ls -l *.html
```

b. To list all "versions" of `myfile.txt`, enter

```
tux > ls -l myfile?.txt
```

Note that you can only use the `?` wild card here because the numbering of the files is single-digit. As soon as you have a file named `myfile10.txt` you must to use the `*` wild card to view all versions of `myfile.txt` (or add another question mark, so your string looks like **`myfile??.txt`**).

c. To remove, for example, version 1-3 and version 5 of `myfile.txt`, enter

```
tux > rm myfile[1-3,5].txt
```

d. Check the result with

```
tux > ls -l
```

Of all `myfile.txt` versions only `myfile4.txt` should be left.

You can also combine several wild cards in one command. In the example above, **`rm myfile[1-3,5].*`** would lead to the same result as **`rm myfile[1-3,5].txt`** because there are only files with the extension `.txt` available.

Note: Using Wildcards in **`rm`** Commands

Wildcards in a **`rm`** command can be very useful but also dangerous: you might delete more files from your directory than intended. To see which files would be affected by the **`rm`**, run your wildcard string with **`ls`** instead of **`rm`** first.

14.8 Editing Texts

In order to edit files from the command line, you will need to know the vi editor. vi is a default editor which can be found on nearly every UNIX/Linux system. It can run several operating modes in which the keys you press have different functions. This does not make it very easy for beginners, but you should know at least the most basic operations with vi. There may be situations where no other editor than vi is available.

Basically, vi makes use of three operating modes:

***command* mode**

In this mode, vi accepts certain key combinations as commands. Simple tasks such as searching words or deleting a line can be executed.

***insert* mode**

In this mode, you can write normal text.

***extended* mode**

In this mode, also known as colon mode (as you have to enter a colon to switch to this mode), vi can execute also more complex tasks such as searching and replacing text.

In the following (very simple) example, you will learn how to open and edit a file with vi, how to save your changes and quit vi.

14.8.1 Example: Editing with vi

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.

1. To create and open a new file with vi, enter

   ```
   tux > vi textfile.txt
   ```

   By default, vi opens in *command* mode in which you cannot enter text.

2. Press `I` to switch to insert mode. The bottom line changes and indicates that you now can insert text.

3. Write some sentences. If you want to insert a new line, first press `Esc` to switch back to command mode. Press `O` to insert a new line and to switch to insert mode again.

4. In the insert mode, you can edit the text with the arrow keys and with `Del`.

5. To leave vi, press `Esc` to switch to command mode again. Then press `:` which takes you to the extended mode. The bottom line now shows a colon.

6. To leave vi and save your changes, type **`wq`** (`w` for `write`; `q` for `quit`) and press `Enter`. If you want to save the file under a different name, type **`w`** *`FILENAME`* and press `Enter`. To leave vi without saving, type **`q!`** instead and press `Enter`.

14.9 Searching for Files or Contents

Bash offers you several commands to search for files and to search for the contents of files:

`find`

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.

grep

The **grep** command finds a specific search string in the specified text files. If the search string is found, the command displays the line in which `searchstring` was found, along with the filename. If desired, use wild cards to specify filenames.

14.9.1 Examples for Searching

- To search your home directory for all occurrences of filenames that contain the file extension `.txt`, use:

```
tux > find ~ -name '*.txt' -print
```

- To search a directory (in this case, your home directory) for all occurrences of files which contain, for example, the word `music`, use:

```
tux > grep music ~/*
```

grep is case-sensitive by default. Hence, with the command above you will not find any files containing `Music`.To ignore case, use the `-i` option.

- To use a search string which consists of more than one word, enclose the string in double quotation marks, for example:

```
tux > grep "music is great" ~/*
```

14.10 Viewing Text Files

When searching for the contents of a file with **grep**, the output gives you the line in which the `searchstring` was found along with the filename. Often this contextual information is still not enough information to decide whether you want to open and edit this file. Bash offers you several commands to have a quick look at the contents of a text file directly in the shell, without opening an editor.

head

With **head** you can view the first lines of a text file. If you do not specify the command any further, **head** shows the first 10 lines of a text file.

tail

The **tail** command is the counterpart of **head**. If you use **tail** without any further options it displays the last 10 lines of a text file. This can be very useful to view log files of your system, where the most recent messages or log entries are usually found at the end of the file.

less

With **less**, display the whole contents of a text file. To move up and down half a page use Page ↑ and Page ↓. Use Space to scroll down one page. Home takes you to the beginning, and End to the end of the document. To end the viewing mode, press Q.

more

Instead of less, you can also use the older program more. It has basically the same function—however, it is less convenient because it does not allow you to scroll backward. Use Space to move forward. When you reach the end of the document, the viewer closes automatically.

cat

The **cat** command displays the contents of a file, printing the entire contents to the screen without interruption. As **cat** does not allow you to scroll it is not very useful as viewer but it is rather often used in combination with other commands.

14.11 Redirection and Pipes

Sometimes it would be useful if you could write the output of a command to a file for further editing or if you could combine several commands, using the output of one command as the input for the next one. The shell offers this function by means of redirection or pipes.

Normally, the standard output in the shell is your screen (or an open shell window) and the standard input is the keyboard. With certain symbols you can redirect the input or the output to another object, such as a file or another command.

Redirection

With > you can forward the output of a command to a file (output redirection), with < you can use a file as input for a command (input redirection).

Pipe

By means of a pipe symbol `|` you can also redirect the output: with a pipe, you can combine several commands, using the output of one command as input for the next command. In contrast to the other redirection symbols `>` and `<`, the use of the pipe is not constrained to files.

14.11.1 Examples for Redirection and Pipe

1. To write the output of a command like **`ls`** to a file, enter

```
tux > ls -l > filelist.txt
```

   This creates a file named `filelist.txt` that contains the list of contents of your current directory as generated by the **`ls`** command.
   However, if a file named `filelist.txt` already exists, this command overwrites the existing file. To prevent this, use `>>` instead of `>`. Entering

```
tux > ls -l >> filelist.txt
```

   simply appends the output of the **`ls`** command to an already existing file named `filelist.txt`. If the file does not exist, it is created.

2. Redirections also works the other way round. Instead of using the standard input from the keyboard for a command, you can use a file as input:

```
tux > sort < filelist.txt
```

   This will force the **`sort`** command to get its input from the contents of `filelist.txt`. The result is shown on the screen. Of course, you can also write the result into another file, using a combination of redirections:

```
tux > sort < filelist.txt > sorted_filelist.txt
```

3. If a command generates a lengthy output, like **`ls`** `-l` may do, it may be useful to pipe the output to a viewer like `less` to be able to scroll through the pages. To do so, enter

```
tux > ls -l | less
```

   The list of contents of the current directory is shown in `less`.

The pipe is also often used in combination with the **grep** command in order to search for a certain string in the output of another command. For example, if you want to view a list of files in a directory which are owned by the user `tux`, enter

```
tux > ls -l | grep tux
```

14.12 Starting Programs and Handling Processes

As you have seen in *Section 14.8, "Editing Texts"*, programs can be started from the shell. Applications with a graphical user interface need the X Window System and can only be started from a terminal window within a graphical user interface. For example, if you want to open a file named `vacation.pdf` in your home directory from a terminal window in KDE or GNOME, simply run **okular ~/vacation.pdf** (or **evince ~/vacation.pdf**) to start a PDF viewer displaying your file.

When looking at the terminal window again you will realize that the command line is blocked as long as the PDF viewer is open, meaning that your prompt is not available. To change this, press Ctrl–Z to suspend the process and enter **bg** to send the process to the background.

Now you can still have a look at `vacation.pdf` while your prompt is available for further commands. An easier way to achieve this is by sending a process to the background directly when starting it. To do so, add an ampersand at the end of the command:

```
tux > okular ~/vacation.pdf &
```

If you have started several background processes (also named jobs) from the same shell, the **jobs** command gives you an overview of the jobs. It also shows the job number in brackets and their status:

```
tux > jobs
[1]   Running        okular book.opensuse.startup-xep.pdf &
[2]-  Running        okular book.opensuse.reference-xep.pdf &
[3]+  Stopped        man jobs
```

To bring a job to the foreground again, enter **fg** *JOB_NUMBER*.

Whereas **job** only shows the background processes started from a specific shell, the **ps** command (run without options) shows a list of all your processes—those you started. Find an example output below:

```
tux > ps
```

```
PID TTY          TIME CMD
15500 pts/1    00:00:00 bash
28214 pts/1    00:00:00 okular
30187 pts/1    00:00:00 kwrite
30280 pts/1    00:00:00 ps
```

In case a program cannot be terminated in the normal way, use the **kill** command to stop the process (or processes) belonging to that program. To do so, specify the process ID (PID) shown by the output of **ps**. For example, to shut down the KWrite editor in the example above, enter

```
tux > kill 30187
```

This sends a *TERM* signal that instructs the program to shut itself down.

Alternatively, if the program or process you want to terminate is a background job and is shown by the **jobs** command, you can also use the **kill** command in combination with the job number to terminate this process. When identifying the job with the job number, you must prefix the number with a percent character (`%`):

```
tux > kill %JOB_NUMBER
```

If **kill** does not help—as is sometimes the case for "runaway" programs—try

```
tux > kill -9 PID
```

This sends a *KILL* signal instead of a *TERM* signal, usually bringing the specified process to an end.

This section is intended to introduce the most basic set of commands for handling jobs and processes. Find an overview for system administrators in *Book "*System Analysis and Tuning Guide*", Chapter 2 "System Monitoring Utilities", Section 2.3 "Processes"*.

14.13 Archives and Data Compression

On Linux, there are two types of commands that make data easier to transfer:

- Archivers, which create a big file out of several smaller ones. The most commonly used archiver is **tar**, another example is **cpio**.
- Compressors, which losslessly make a file smaller. The most commonly used compressors are **gzip** and **bzip2**.

When combining these two types of commands, their effect is comparable to the compressed archive files that are prevalent on other operating systems, for example, `ZIP` or `RAR`.

To pack the `test` directory with all its files and subdirectories into an archive named `testarchive.tar`, do the following:

PROCEDURE 14.8: ARCHIVING FILES

1. Open a shell.
2. Use **`cd`** to change to your home directory where the `test` directory is located.
3. Commpress the file with:

   ```
   tux > 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.
   The `test` directory with all its files and directories has remained unchanged on your hard disk.
4. View the contents of the archive file with:

   ```
   tux > tar -tf testarchive.tar
   ```
5. To unpack the archive, use:

   ```
   tux > tar -xvf testarchive.tar
   ```

   If files in your current directory are named the same as the files in the archive, they will be overwritten without warning.

To compress files, use **`gzip`** or, for better compression, **`bzip2`**.

PROCEDURE 14.9: COMPRESSING A FILE

1. For this example, reuse the archive `testarchive.tar` from *Procedure 14.8, "Archiving Files"*. To compress the archive, use:

   ```
   tux > gzip testarchive.tar
   ```

   With **`ls`**, now see that the file `testarchive.tar` is no longer there and that the file `testarchive.tar.gz` has been created instead.

As an alternative, use **bzip2 testarchive.tar** which works analogously but provides somewhat better compression.

2. Now decompress and unarchive the file again:

   - This can be done in two steps by first decompressing and then unarchiving the file:

     ```
     tux > gzip --decompress testarchive.tar.gz
     tux > tar -xvf testarchive.tar
     ```

   - You can also decompress and unarchive in one step:

     ```
     tux > tar -xvf testarchive.tar
     ```

   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.

14.14 Important Linux Commands

This section provides an overview of the most important Linux commands. 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.

Adjust the parameters 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**.

14.14.1 File Commands

The following section lists the most important commands for file management. It covers everything from general file administration to the manipulation of file system ACLs.

14.14.1.1 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 USER_NAME*[:*GROUP*] *FILES*

Transfers ownership of a file to the user with the specified user name.

`-R`

Changes files and directories in all subdirectories

chgrp *OPTIONS GROUP_NAME FILES*

Transfers the group ownership of a given `file` to the group with the specified group name. The file owner can change group ownership only 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 several 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`.

`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.

14.14.1.2 Commands to Access File Contents

`file` *OPTIONS FILES*

In Linux, files can have a file extensions but do not need to have one. The **`file`** determines the file type of a given file. With the output of **`file`**, you can then choose an appropriate application with which to open the file.

-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 Page ↑ and Page ↓ or a full screen page down with Space. Jump to the beginning or end of a file using Home and End. Press Q to quit the program.

grep *OPTIONS SEARCH_STRING 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 *SEARCH_STRING* was found along with the file name.

`-i`

Ignores case

`-H`

Only displays the names of the relevant 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 FILE_1 FILE_2*

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 to 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

14.14.1.3 File Systems

mount *OPTIONS DEVICE MOUNT_POINT*

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` *`FILE_SYSTEM`*

Specify the file system: For Linux hard disks, this is commonly `ext4`, `xfs`, or `btrfs`. 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 needs to 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 MOUNT_POINT`*

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 relevant drive.

14.14.2 System Commands

The following section lists a few of the most important commands needed for retrieving system information and controlling processes and the network.

14.14.2.1 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 *Book "Reference", Chapter 15 "Special System Features", Section 15.1.7 "The* **`free`** *Command"* 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.

14.14.2.2 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 PROCESS_NAME*

This command is similar to **kill**, but uses the process name (instead of the process ID) as an argument, ending all processes with that name.

14.14.2.3 Network

ping *OPTIONS HOSTNAME_OR_IP_ADDRESS*

The **ping** command is the standard tool for testing the basic functionality of TCP/IP networks. It sends a small data packet to the destination host, requesting an immediate reply. If this works, **ping** displays a message to that effect, which indicates that the network link is 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)

host *OPTIONS HOSTNAME SERVER*

The domain name system resolves domain names to IP addresses. With this tool, send queries to name servers (DNS servers).

ssh *OPTIONS* [*USER@*]*HOSTNAME* *COMMAND*

SSH is actually an Internet protocol that enables you to work on remote hosts across a network. SSH is also the name of a Linux program that uses this protocol to enable operations on remote computers.

14.14.2.4 Miscellaneous

passwd *OPTIONS* *USER_NAME*

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* *USER_NAME*

The **su** command makes it possible to log in under a different user name from a running session. Specify a user name 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 user name, you are prompted for the `root` password and change to the superuser (`root`). Use **su -** to start a login shell for a 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.

14.14.3 For More Information

There are many more commands than listed in this chapter. For information about other commands or more detailed information, also see the publication *Linux in a Nutshell* by O'Reilly.

15 Bash and Bash Scripts

Today, many people use computers with a graphical user interface (GUI) like GNOME. Although GUIs offer many features, they're limited when performing automated task execution. Shells complement GUIs well, and this chapter gives an overview of some aspects of shells, in this case the Bash shell.

15.1 What is “The Shell”?

Traditionally, *the* shell is Bash (Bourne again Shell). When this chapter speaks about “the shell” it means Bash. There are more shells available (ash, csh, ksh, zsh, …), each employing different features and characteristics. If you need further information about other shells, search for *shell* in YaST.

15.1.1 Knowing the Bash Configuration Files

A shell can be invoked as an:

1. **Interactive login shell.** This is used when logging in to a machine, invoking Bash with the `--login` option or when logging in to a remote machine with SSH.
2. **“Ordinary” interactive shell.** This is normally the case when starting xterm, konsole, gnome-terminal or similar tools.
3. **Non-interactive shell.** This is used when invoking a shell script at the command line.

Depending on the type of shell you use, different configuration files will be read. The following tables show the login and non-login shell configuration files.

TABLE 15.1: BASH CONFIGURATION FILES FOR LOGIN SHELLS

File	Description
`/etc/profile`	Do not modify this file, otherwise your modifications may be destroyed during your next update!

File	Description
`/etc/profile.local`	Use this file if you extend `/etc/profile`
`/etc/profile.d/`	Contains system-wide configuration files for specific programs
`~/.profile`	Insert user specific configuration for login shells here

Note that the login shell also sources the configuration files listed under *Table 15.2, "Bash Configuration Files for Non-Login Shells"*.

TABLE 15.2: BASH CONFIGURATION FILES FOR NON-LOGIN SHELLS

`/etc/bash.bashrc`	Do not modify this file, otherwise your modifications may be destroyed during your next update!
`/etc/bash.bashrc.local`	Use this file to insert your system-wide modifications for Bash only
`~/.bashrc`	Insert user specific configuration here

Additionally, Bash uses some more files:

TABLE 15.3: SPECIAL FILES FOR BASH

File	Description
`~/.bash_history`	Contains a list of all commands you have typed
`~/.bash_logout`	Executed when logging out
`~/.alias`	User defined aliases of frequently used commands. See **`man 1 alias`** for more details about defining aliases.

15.1.2 The Directory Structure

The following table provides a short overview of the most important higher-level directories that you find on a Linux system. Find more detailed information about the directories and important subdirectories in the following list.

TABLE 15.4: OVERVIEW OF A STANDARD DIRECTORY TREE

Directory	Contents
`/`	Root directory—the starting point of the directory tree.
`/bin`	Essential binary files, such as commands that are needed by both the system administrator and normal users. Usually also contains the shells, such as Bash.
`/boot`	Static files of the boot loader.
`/dev`	Files needed to access host-specific devices.
`/etc`	Host-specific system configuration files.
`/home`	Holds the home directories of all users who have accounts on the system. However, `root`'s home directory is not located in `/home` but in `/root`.
`/lib`	Essential shared libraries and kernel modules.
`/media`	Mount points for removable media.
`/mnt`	Mount point for temporarily mounting a file system.
`/opt`	Add-on application software packages.
`/root`	Home directory for the superuser `root`.
`/sbin`	Essential system binaries.
`/srv`	Data for services provided by the system.
`/tmp`	Temporary files.
`/usr`	Secondary hierarchy with read-only data.

Directory	Contents
`/var`	Variable data such as log files.
`/windows`	Only available if you have both Microsoft Windows* and Linux installed on your system. Contains the Windows data.

The following list provides more detailed information and gives some examples of which files and subdirectories can be found in the directories:

`/bin`
: Contains the basic shell commands that may be used both by `root` and by other users. These commands include **`ls`**, **`mkdir`**, **`cp`**, **`mv`**, **`rm`** and **`rmdir`**. `/bin` also contains Bash, the default shell in openSUSE Leap.

`/boot`
: Contains data required for booting, such as the boot loader, the kernel, and other data that is used before the kernel begins executing user-mode programs.

`/dev`
: Holds device files that represent hardware components.

`/etc`
: Contains local configuration files that control the operation of programs like the X Window System. The `/etc/init.d` subdirectory contains LSB init scripts that can be executed during the boot process.

`/home/`*`USERNAME`*
: Holds the private data of every user who has an account on the system. The files located here can only be modified by their owner or by the system administrator. By default, your e-mail directory and personal desktop configuration are located here in the form of hidden files and directories, such as `.gconf/` and `.config`.

Note: Home Directory in a Network Environment

If you are working in a network environment, your home directory may be mapped to a directory in the file system other than `/home`.

`/lib`

Contains the essential shared libraries needed to boot the system and to run the commands in the root file system. The Windows equivalent for shared libraries are DLL files.

`/media`

Contains mount points for removable media, such as CD-ROMs, flash disks, and digital cameras (if they use USB). `/media` generally holds any type of drive except the hard disk of your system. When your removable medium has been inserted or connected to the system and has been mounted, you can access it from here.

`/mnt`

This directory provides a mount point for a temporarily mounted file system. `root` may mount file systems here.

`/opt`

Reserved for the installation of third-party software. Optional software and larger add-on program packages can be found here.

`/root`

Home directory for the `root` user. The personal data of `root` is located here.

`/run`

A tmpfs directory used by `systemd` and various components. `/var/run` is a symbolic link to `/run`.

`/sbin`

As the `s` indicates, this directory holds utilities for the superuser. `/sbin` contains the binaries essential for booting, restoring and recovering the system in addition to the binaries in `/bin`.

`/srv`

Holds data for services provided by the system, such as FTP and HTTP.

`/tmp`

This directory is used by programs that require temporary storage of files.

Important: Cleaning up `/tmp` at Boot Time

Data stored in `/tmp` is not guaranteed to survive a system reboot. It depends, for example, on settings made in `/etc/tmpfiles.d/tmp.conf`.

`/usr`

`/usr` has nothing to do with users, but is the acronym for Unix system resources. The data in `/usr` is static, read-only data that can be shared among various hosts compliant with the `Filesystem Hierarchy Standard` (FHS). This directory contains all application programs including the graphical desktops such as GNOME and establishes a secondary hierarchy in the file system. `/usr` holds several subdirectories, such as `/usr/bin`, `/usr/sbin`, `/usr/local`, and `/usr/share/doc`.

`/usr/bin`

Contains generally accessible programs.

`/usr/sbin`

Contains programs reserved for the system administrator, such as repair functions.

`/usr/local`

In this directory the system administrator can install local, distribution-independent extensions.

`/usr/share/doc`

Holds various documentation files and the release notes for your system. In the `manual` subdirectory find an online version of this manual. If more than one language is installed, this directory may contain versions of the manuals for different languages.

Under `packages` find the documentation included in the software packages installed on your system. For every package, a subdirectory `/usr/share/doc/packages/`*`PACKAGENAME`* is created that often holds README files for the package and sometimes examples, configuration files or additional scripts.

If HOWTOs are installed on your system `/usr/share/doc` also holds the `howto` subdirectory in which to find additional documentation on many tasks related to the setup and operation of Linux software.

`/var`

Whereas `/usr` holds static, read-only data, `/var` is for data which is written during system operation and thus is variable data, such as log files or spooling data. For an overview of the most important log files you can find under `/var/log/`, refer to *Table 17.1, "Log Files"*.

`/windows`

Only available if you have both Microsoft Windows and Linux installed on your system. Contains the Windows data available on the Windows partition of your system. Whether you can edit the data in this directory depends on the file system your Windows partition

uses. If it is FAT32, you can open and edit the files in this directory. For NTFS, openSUSE Leap also includes write access support. However, the driver for the NTFS-3g file system has limited functionality.

15.2 Writing Shell Scripts

Shell scripts provide a convenient way to perform a wide range of tasks: collecting data, searching for a word or phrase in a text and other useful things. The following example shows a small shell script that prints a text:

EXAMPLE 15.1: A SHELL SCRIPT PRINTING A TEXT

```
#!/bin/sh ❶
# Output the following line: ❷
echo "Hello World" ❸
```

❶ The first line begins with the *Shebang* characters (`#!`) which indicate that this file is a script. The interpreter, specified after the *Shebang*, executes the script. In this case, the specified interpreter is **`/bin/sh`**.

❷ The second line is a comment beginning with the hash sign. We recommend that you comment difficult lines. With proper commenting, you can remember the purpose and function of the line. Also, other readers will hopefully understand your script. Commenting is considered good practice in the development community.

❸ The third line uses the built-in command **`echo`** to print the corresponding text.

Before you can run this script, there are a few prerequisites:

1. Every script should contain a Shebang line (as in the example above). If the line is missing, you need to call the interpreter manually.

2. You can save the script wherever you want. However, it is a good idea to save it in a directory where the shell can find it. The search path in a shell is determined by the environment variable `PATH`. Usually a normal user does not have write access to `/usr/bin`. Therefore it is recommended to save your scripts in the users' directory `~/bin/`. The above example gets the name `hello.sh`.

3. The script needs executable permissions. Set the permissions with the following command:

   ```
   tux > chmod +x ~/bin/hello.sh
   ```

If you have fulfilled all of the above prerequisites, you can execute the script in the following ways:

1. **As Absolute Path.** The script can be executed with an absolute path. In our case, it is `~/bin/hello.sh`.
2. **Everywhere.** If the `PATH` environment variable contains the directory where the script is located, you can execute the script with `hello.sh`.

15.3 Redirecting Command Events

Each command can use three channels, either for input or output:

- **Standard Output.** This is the default output channel. Whenever a command prints something, it uses the standard output channel.
- **Standard Input.** If a command needs input from users or other commands, it uses this channel.
- **Standard Error.** Commands use this channel for error reporting.

To redirect these channels, there are the following possibilities:

`Command > File`

Saves the output of the command into a file, an existing file will be deleted. For example, the `ls` command writes its output into the file `listing.txt`:

```
tux > ls > listing.txt
```

`Command >> File`

Appends the output of the command to a file. For example, the `ls` command appends its output to the file `listing.txt`:

```
tux > ls >> listing.txt
```

`Command < File`

Reads the file as input for the given command. For example, the `read` command reads in the content of the file into the variable:

```
tux > read a < foo
```

`Command1 | Command2`

Redirects the output of the left command as input for the right command. For example, the **cat** command outputs the content of the `/proc/cpuinfo` file. This output is used by **grep** to filter only those lines which contain `cpu`:

```
tux > cat /proc/cpuinfo | grep cpu
```

Every channel has a *file descriptor*: 0 (zero) for standard input, 1 for standard output and 2 for standard error. It is allowed to insert this file descriptor before a `<` or `>` character. For example, the following line searches for a file starting with `foo`, but suppresses its errors by redirecting it to `/dev/null`:

```
tux > find / -name "foo*" 2>/dev/null
```

15.4 Using Aliases

An alias is a shortcut definition of one or more commands. The syntax for an alias is:

```
alias NAME=DEFINITION
```

For example, the following line defines an alias **lt** that outputs a long listing (option `-l`), sorts it by modification time (`-t`), and prints it in reverse sorted order (`-r`):

```
tux > alias lt='ls -ltr'
```

To view all alias definitions, use **alias**. Remove your alias with **unalias** and the corresponding alias name.

15.5 Using Variables in Bash

A shell variable can be global or local. Global variables, or environment variables, can be accessed in all shells. In contrast, local variables are visible in the current shell only.

To view all environment variables, use the **printenv** command. If you need to know the value of a variable, insert the name of your variable as an argument:

```
tux > printenv PATH
```

A variable, be it global or local, can also be viewed with **`echo`**:

```
tux > echo $PATH
```

To set a local variable, use a variable name followed by the equal sign, followed by the value:

```
tux > PROJECT="SLED"
```

Do not insert spaces around the equal sign, otherwise you get an error. To set an environment variable, use **`export`**:

```
tux > export NAME="tux"
```

To remove a variable, use **`unset`**:

```
tux > unset NAME
```

The following table contains some common environment variables which can be used in you shell scripts:

TABLE 15.5: USEFUL ENVIRONMENT VARIABLES

`HOME`	the home directory of the current user
`HOST`	the current host name
`LANG`	when a tool is localized, it uses the language from this environment variable. English can also be set to `C`
`PATH`	the search path of the shell, a list of directories separated by colon
`PS1`	specifies the normal prompt printed before each command
`PS2`	specifies the secondary prompt printed when you execute a multi-line command
`PWD`	current working directory
`USER`	the current user

15.5.1 Using Argument Variables

For example, if you have the script **foo.sh** you can execute it like this:

```
tux > foo.sh "Tux Penguin" 2000
```

To access all the arguments which are passed to your script, you need positional parameters. These are `$1` for the first argument, `$2` for the second, and so on. You can have up to nine parameters. To get the script name, use `$0`.

The following script **foo.sh** prints all arguments from 1 to 4:

```
#!/bin/sh
echo \"$1\" \"$2\" \"$3\" \"$4\"
```

If you execute this script with the above arguments, you get:

```
"Tux Penguin" "2000" "" ""
```

15.5.2 Using Variable Substitution

Variable substitutions apply a pattern to the content of a variable either from the left or right side. The following list contains the possible syntax forms:

`${VAR#pattern}`

removes the shortest possible match from the left:

```
tux > file=/home/tux/book/book.tar.bz2
tux > echo ${file#*/}
home/tux/book/book.tar.bz2
```

`${VAR##pattern}`

removes the longest possible match from the left:

```
tux > file=/home/tux/book/book.tar.bz2
tux > echo ${file##*/}
book.tar.bz2
```

`${VAR%pattern}`

removes the shortest possible match from the right:

```
tux > file=/home/tux/book/book.tar.bz2
```

```
tux > echo ${file%.*}
/home/tux/book/book.tar
```

`${VAR%%pattern}`

removes the longest possible match from the right:

```
tux > file=/home/tux/book/book.tar.bz2
tux > echo ${file%%.*}
/home/tux/book/book
```

`${VAR/pattern_1/pattern_2}`

substitutes the content of *VAR* from the *PATTERN_1* with *PATTERN_2*:

```
tux > file=/home/tux/book/book.tar.bz2
tux > echo ${file/tux/wilber}
/home/wilber/book/book.tar.bz2
```

15.6 Grouping and Combining Commands

Shells allow you to concatenate and group commands for conditional execution. Each command returns an exit code which determines the success or failure of its operation. If it is 0 (zero) the command was successful, everything else marks an error which is specific to the command.

The following list shows, how commands can be grouped:

`Command1 ; Command2`

executes the commands in sequential order. The exit code is not checked. The following line displays the content of the file with **cat** and then prints its file properties with **ls** regardless of their exit codes:

```
tux > cat filelist.txt ; ls -l filelist.txt
```

`Command1 && Command2`

runs the right command, if the left command was successful (logical AND). The following line displays the content of the file and prints its file properties only, when the previous command was successful (compare it with the previous entry in this list):

```
tux > cat filelist.txt && ls -l filelist.txt
```

`Command1 || Command2`

runs the right command, when the left command has failed (logical OR). The following line creates only a directory in `/home/wilber/bar` when the creation of the directory in `/home/tux/foo` has failed:

```
tux > mkdir /home/tux/foo || mkdir /home/wilber/bar
```

`funcname(){ ... }`

creates a shell function. You can use the positional parameters to access its arguments. The following line defines the function `hello` to print a short message:

```
tux > hello() { echo "Hello $1"; }
```

You can call this function like this:

```
tux > hello Tux
```

which prints:

```
Hello Tux
```

15.7 Working with Common Flow Constructs

To control the flow of your script, a shell has **`while`**, **`if`**, **`for`** and **`case`** constructs.

15.7.1 The if Control Command

The **`if`** command is used to check expressions. For example, the following code tests whether the current user is Tux:

```
if test $USER = "tux"; then
  echo "Hello Tux."
else
  echo "You are not Tux."
fi
```

The test expression can be as complex or simple as possible. The following expression checks if the file `foo.txt` exists:

```
if test -e /tmp/foo.txt ; then
  echo "Found foo.txt"
```

```
fi
```

The test expression can also be abbreviated in square brackets:

```
if [ -e /tmp/foo.txt ] ; then
  echo "Found foo.txt"
fi
```

Find more useful expressions at https://bash.cyberciti.biz/guide/If..else..fi.

15.7.2 Creating Loops with the `for` Command

The `for` loop allows you to execute commands to a list of entries. For example, the following code prints some information about PNG files in the current directory:

```
for i in *.png; do
 ls -l $i
done
```

15.8 For More Information

Important information about Bash is provided in the man pages `man bash`. More about this topic can be found in the following list:

- http://tldp.org/LDP/Bash-Beginners-Guide/html/index.html—Bash Guide for Beginners
- http://tldp.org/HOWTO/Bash-Prog-Intro-HOWTO.html—BASH Programming - Introduction HOW-TO
- http://tldp.org/LDP/abs/html/index.html—Advanced Bash-Scripting Guide
- http://www.grymoire.com/Unix/Sh.html—Sh - the Bourne Shell

V Help and Troubleshooting

16 Help and Documentation

openSUSE® Leap comes with various sources of information and documentation, many of which are already integrated into your installed system.

Documentation in `/usr/share/doc`

This traditional help directory holds various documentation files and release notes for your system. It contains also information of installed packages in the subdirectory `packages`. Find more detailed information in *Section 16.1, "Documentation Directory"*.

Man Pages and Info Pages for Shell Commands

When working with the shell, you do not need to know the options of the commands by heart. Traditionally, the shell provides integrated help by means of man pages and info pages. Read more in *Section 16.2, "Man Pages"* and *Section 16.3, "Info Pages"*.

Desktop Help Center

The help center of the GNOME desktop (Help) provides central access to the most important documentation resources on your system in searchable form. These resources include online help for installed applications, man pages, info pages, and the SUSE manuals delivered with your product.

Separate Help Packages for Some Applications

When installing new software with YaST, the software documentation is usually installed automatically and appears in the help center of your desktop. However, some applications, such as GIMP, may have different online help packages that can be installed separately with YaST and do not integrate into the help centers.

16.1 Documentation Directory

The traditional directory to find documentation on your installed Linux system is `/usr/share/doc`. Usually, the directory contains information about the packages installed on your system, plus release notes, manuals, and more.

Note: Contents Depends on Installed Packages

In the Linux world, many manuals and other kinds of documentation are available in the form of packages, like software. How much and which information you find in `/usr/share/docs` also depends on the (documentation) packages installed. If you cannot find the subdirectories mentioned here, check if the respective packages are installed on your system and add them with YaST, if needed.

16.1.1 SUSE Manuals

We provide HTML and PDF versions of our books in different languages. In the `manual` subdirectory, find HTML versions of most of the SUSE manuals available for your product. For an overview of all documentation available for your product refer to the preface of the manuals.

If more than one language is installed, `/usr/share/doc/manual` may contain different language versions of the manuals. The HTML versions of the SUSE manuals are also available in the help center of both desktops. For information on where to find the PDF and HTML versions of the books on your installation media, refer to the openSUSE Leap Release Notes. They are available on your installed system under `/usr/share/doc/release-notes/` or online at your product-specific Web page at https://doc.opensuse.org/release-notes/.

16.1.2 Package Documentation

Under `packages`, find the documentation that is included in the software packages installed on your system. For every package, a subdirectory `/usr/share/doc/packages/`*`PACKAGENAME`* is created. It often contains README files for the package and sometimes examples, configuration files, or additional scripts. The following list introduces typical files to be found under `/usr/share/doc/packages`. None of these entries are mandatory and many packages might only include a few of them.

`AUTHORS`
: List of the main developers.

`BUGS`
: Known bugs or malfunctions. Might also contain 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`

How to install this package on your system. As the package is already installed by the time you get to read this file, you can safely ignore the contents of this file.

`README`, `README.*`

General information on the software. For example, for what purpose and how to use it.

`TODO`

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.

16.2 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 can be accessed with **`man`** followed by the name of the command, for example, **`man ls`**.

Man pages are displayed directly in the shell. To navigate them, move up and down with Page ↑ and Page ↓. 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`**. Man pages are sorted in categories as shown in *Table 16.1, "Man Pages—Categories and Descriptions"* (taken from the man page for man itself).

TABLE 16.1: 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)

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.

16.3 Info Pages

Info pages are another important source of information on your system. Usually, they are more detailed than man pages. They consist of more than command line options and contain sometimes whole tutorials or reference documentation. To view the info page for a certain command, enter **`info`** followed by the name of the command, for example, **`info ls`** . You can browse an info page with a viewer directly in the shell and display the different sections, called "nodes". Use Space to move forward and <- to move backward. Within a node, you can also

browse with Page ↑ and Page ↓ but only Space and <– will take you also to the previous or subsequent node. Press Q to end the viewing mode. Not every command comes with an info page and vice versa.

16.4 Online Resources

In addition to the online versions of the SUSE manuals installed under `/usr/share/doc`, you can also access the product-specific manuals and documentation on the Web. For an overview of all documentation available for openSUSE Leap check out your product-specific documentation Web page at http:/doc.opensuse.org/.

If you are searching for additional product-related information, you can also refer to the following Web sites:

SUSE Forums

There are several forums where you can dive in on discussions about SUSE products. See http://forums.opensuse.org/ for a list.

GNOME Documentation

Documentation for GNOME users, administrators and developers is available at http://library.gnome.org/.

The Linux Documentation Project

The Linux Documentation Project (TLDP) is run by a team of volunteers who write Linux-related documentation (see http://www.tldp.org). It is probably the most comprehensive documentation resource for Linux. 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. Parts of the documentation from TLDP are also available on openSUSE Leap.

You can also try general-purpose search engines. For example, use the search terms `Linux CD-RW help` or `OpenOffice file conversion problem` if you have trouble with burning CDs or LibreOffice file conversion.

17 Common Problems and Their Solutions

This chapter describes a range of potential problems and their solutions. Even if your situation is not precisely listed here, there may be one similar enough to offer hints to the solution of your problem.

17.1 Finding and Gathering Information

Linux reports things in a very detailed way. There are several places to look when you encounter problems with your system, most of which are standard to Linux systems in general, and some are relevant to openSUSE Leap systems. Most log files can be viewed with YaST (*Miscellaneous* › *Start-Up Log*).

YaST offers the possibility to collect all system information needed by the support team. Use *Other* › *Support* and select the problem category. When all information is gathered, attach it to your support request.

A list of the most frequently checked log files follows with the description of their typical purpose. Paths containing `~` refer to the current user's home directory.

TABLE 17.1: LOG FILES

Log File	Description
`~/.xsession-errors`	Messages from the desktop applications currently running.
`/var/log/apparmor/`	Log files from AppArmor, see *Book "*Security Guide" for detailed information.
`/var/log/audit/audit.log`	Log file from Audit to track any access to files, directories, or resources of your system, and trace system calls. See *Book "*Security Guide" for detailed information.
`/var/log/mail.*`	Messages from the mail system.
`/var/log/NetworkManager`	Log file from NetworkManager to collect problems with network connectivity

Log File	Description
`/var/log/samba/`	Directory containing Samba server and client log messages.
`/var/log/warn`	All messages from the kernel and system log daemon with the “warning” level or higher.
`/var/log/wtmp`	Binary file containing user login records for the current machine session. View it with **`last`**.
`/var/log/Xorg.*.log`	Various start-up and runtime log files from the X Window System. It is useful for debugging failed X start-ups.
`/var/log/YaST2/`	Directory containing YaST's actions and their results.
`/var/log/zypper.log`	Log file of Zypper.

Apart from log files, your machine also supplies you with information about the running system. See *Table 17.2: System Information With the `/proc` File System*

TABLE 17.2: SYSTEM INFORMATION WITH THE `/proc` FILE SYSTEM

File	Description
`/proc/cpuinfo`	Contains processor information, including its type, make, model, and performance.
`/proc/dma`	Shows which DMA channels are currently being used.
`/proc/interrupts`	Shows which interrupts are in use, and how many of each have been in use.
`/proc/iomem`	Displays the status of I/O (input/output) memory.

File	Description
`/proc/ioports`	Shows which I/O ports are in use at the moment.
`/proc/meminfo`	Displays memory status.
`/proc/modules`	Displays the individual modules.
`/proc/mounts`	Displays devices currently mounted.
`/proc/partitions`	Shows the partitioning of all hard disks.
`/proc/version`	Displays the current version of Linux.

Apart from the `/proc` file system, the Linux kernel exports information with the `sysfs` module, an in-memory file system. This module represents kernel objects, their attributes and relationships. For more information about `sysfs`, see the context of udev in *Book "*Reference*", Chapter 16 "Dynamic Kernel Device Management with* `udev`*"*. *Table 17.3* contains an overview of the most common directories under `/sys`.

TABLE 17.3: SYSTEM INFORMATION WITH THE `/sys` FILE SYSTEM

File	Description
`/sys/block`	Contains subdirectories for each block device discovered in the system. Generally, these are mostly disk type devices.
`/sys/bus`	Contains subdirectories for each physical bus type.
`/sys/class`	Contains subdirectories grouped together as a functional types of devices (like graphics, net, printer, etc.)
`/sys/device`	Contains the global device hierarchy.

Linux comes with several tools for system analysis and monitoring. See *Book "*System Analysis and Tuning Guide*", Chapter 2 "System Monitoring Utilities"* for a selection of the most important ones used in system diagnostics.

Each of the following scenarios 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 are related.

17.2 Boot Problems

Boot problems are situations when your system does not boot properly (does not boot to the expected target and login screen).

17.2.1 The GRUB 2 Boot Loader Fails to Load

If the hardware is functioning properly, it is possible that the boot loader is corrupted and Linux cannot start on the machine. In this case, it is necessary to repair the boot loader. To do so, you need to start the Rescue System as described in *Section 17.5.2, "Using the Rescue System"* and follow the instructions in *Section 17.5.2.4, "Modifying and Re-installing the Boot Loader"*.

Alternatively, you can use the Rescue System to fix the boot loader as follows. Boot your machine from the installation media. In the boot screen, choose *More* › *Boot Linux System*. Select the disk containing the installed system and kernel with the default kernel options.

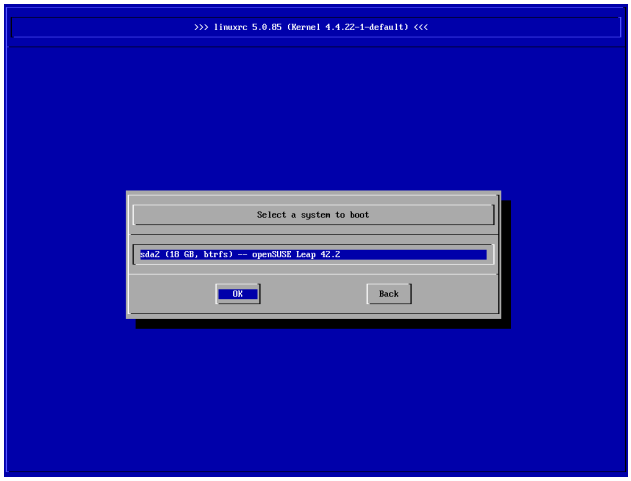


FIGURE 17.1: SELECT DISK

When the system is booted, start YaST and switch to *System* › *Boot Loader*. Make sure that the *Write generic Boot Code to MRB* option is enabled, and click *OK*. This fixes the corrupted boot loader by overwriting it, or installs the boot loader if it is missing.

Other reasons for the machine not booting may be BIOS-related:

BIOS Settings

Check your BIOS for references to your hard disk. GRUB 2 may simply not be started if the hard disk 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 may install properly, but fails to boot when access to the hard disk is required.

17.2.2 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 2 boot loader screen during the boot process as follows:

1. Reboot the computer using the reset button, or switch it off and on again.
2. When the GRUB 2 boot screen becomes visible, select the *Advanced Options* entry and choose the previous kernel from the menu. The machine will 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, set the default boot entry to the old kernel using the YaST *Boot Loader* module. For more information refer to *Book "*Reference*", Chapter 12 "The Boot Loader GRUB 2", Section 12.3 "Configuring the Boot Loader with YaST"*. However, doing this is probably not necessary because automated update tools normally modify it for you during the rollback process.
4. Reboot.

If this does not fix the problem, boot the computer using the installation media. After the machine has booted, continue with *Step 3*.

17.2.3 No Graphical Login

If the machine starts, but does not boot into the graphical login manager, anticipate problems either with the choice of the default systemd target or the configuration of the X Window System. To check the current systemd default target run the command **`sudo systemctl get-default`**. If the value returned is *not* `graphical.target`, run the command **`sudo systemctl isolate graphical.target`**. If the graphical login screen starts, log in and start *YaST* › *System* › *Services Manager* and set the *Default System Target* to *Graphical Interface*. From now on the system should boot into the graphical login screen.

If the graphical login screen does not start even if having booted or switched to the graphical target, your desktop or X Window software is probably misconfigured or corrupted. 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 may log error messages to the system journal that can be queried with the command **`journalctl`** (see *Book "*Reference*", Chapter 11 "***`journalctl`***: Query the* `systemd` *Journal"* for more information). 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.

17.2.4 Root Btrfs Partition Cannot Be Mounted

If a `btrfs` root partition becomes corrupted, try the following options:

- Mount the partition with the `-o recovery` option.
- If that fails, run **`btrfs-zero-log`** on your root partition.

17.2.5 Force Checking Root Partitions

If the root partition becomes corrupted, use the parameter `forcefsck` on the boot prompt. This passes the option `-f` (force) to the **`fsck`** command.

17.3 Login Problems

Login problems occur when your machine does boot to the expected welcome screen or login prompt, but refuses to accept the user name 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.).

17.3.1 Valid User Name and Password Combinations Fail

This usually occurs when the system is configured to use network authentication or directory services and, for some reason, cannot 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 a machine appears functional but cannot process logins correctly:

- The network is not working. For further directions on this, turn to *Section 17.4, "Network Problems"*.
- DNS is not working at the moment (which prevents GNOME 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 17.4, "Network Problems"*.
- If the system is configured to use Kerberos, the system's local time may 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 *Book "Security Guide", Chapter 2 "Authentication with PAM"*.
- The home partition is encrypted. Find more information about this topic in *Section 17.3.3, "Login to Encrypted Home Partition Fails"*.

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. Press `Esc` to exit the splash screen and get to the GRUB 2 text-based menu.
3. Press `B` to enter the GRUB 2 editor.
4. Add the following parameter to the line containing the kernel parameters:

   ```
   systemd.unit=rescue.target
   ```

5. Press `F10`.
6. Enter the user name and password for `root`.
7. Make all the necessary changes.
8. Boot into the full multiuser and network mode by entering **`systemctl isolate graphical.target`** at the command line.

17.3.2 Valid User Name 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 may have entered the wrong password.
- The user's home directory containing the desktop configuration files is corrupted or write protected.
- There may 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 their password correctly before you start debugging the whole authentication mechanism. If the user may have not have remembered their password correctly, use the YaST User Management module to change the user's password. Pay attention to the Caps Lock key and unlock it, if necessary.

2. Log in as `root` and check the system journal with **`journalctl -e`** 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 GNOME desktop. For more information, refer to *Section 17.3.4, "Login Successful but GNOME Desktop Fails"*.

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 graphical login again.

5. If the desktop could not start because of corrupt configuration files, proceed with *Section 17.3.4, "Login Successful but GNOME Desktop Fails"*.

In the following, common reasons a network authentication for a particular user may fail on a specific machine are listed:

- The user may have entered the wrong password.
- The user name 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 host names, 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 may 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 their password correctly before you start debugging the whole authentication mechanism.
2. Determine the directory server which 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 user name and password work on other machines to make sure that their 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 system journal with **`journalctl -e`** > 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 problem is not with 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 GNOME desktop. For more information, refer to *Section 17.3.4, "Login Successful but GNOME Desktop Fails"*.
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 graphical login again.
7. If the desktop could not start because of corrupt configuration files, proceed with *Section 17.3.4, "Login Successful but GNOME Desktop Fails"*.

17.3.3 Login to Encrypted Home Partition Fails

It is recommended to use an encrypted home partition for laptops. If you cannot log in to your laptop, the reason is usually simple: your partition could not be unlocked.

During the boot time, you need to enter the passphrase to unlock your encrypted partition. If you do not enter it, the boot process continues, leaving the partition locked.

To unlock your encrypted partition, proceed as follows:

1. Switch to the text console with Ctrl–Alt–F1.
2. Become `root`.
3. Restart the unlocking process again with:

```
root # systemctl restart home.mount
```

4. Enter your passphrase to unlock your encrypted partition.
5. Exit the text console and switch back to the login screen with Alt–F7.
6. Log in as usual.

17.3.4 Login Successful but GNOME Desktop Fails

If this is the case, it is likely that your GNOME configuration files have become corrupted. Some symptoms may 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. It is then 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. Switch to a text console by pressing Ctrl–Alt–F1.
2. Log in with your user name.
3. Move the user's GNOME configuration directories to a temporary location:

```
tux > mv .gconf  .gconf-ORIG-RECOVER
tux > mv .gnome2 .gnome2-ORIG-RECOVER
```

4. Log out.
5. Log in again, but do not run any applications.

6. Recover your 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:

```
tux > cp -a .gconf-ORIG-RECOVER/apps .gconf/
```

If this causes the login problems, attempt to recover only the critical application data and reconfigure the remainder of the applications.

17.4 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 may be a network problem of some kind. This section introduces a simple checklist you can apply to identify the cause of any network problem encountered.

PROCEDURE 17.1: HOW TO IDENTIFY NETWORK PROBLEMS

When checking the network connection of your machine, proceed as follows:

1. If you use an Ethernet connection, check the hardware first. Make sure that your network cable is properly plugged into your computer and router (or hub, etc.). The control lights next to your Ethernet connector are normally 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, they may be faulty, as well.

2. If using a wireless connection, check whether the wireless link can be established by other machines. If not, contact the wireless network's administrator.

3. When 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 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 functionality in many ways. If the local machine relies on any network servers for authentication and these servers cannot be found because of 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 needs data stored in an NFS mounted directory, it cannot start or function properly if this service was down or misconfigured. In the worst case scenario, a user's personal desktop configuration would not come up if their home directory containing the `.gconf` subdirectory could not be found because of a faulty NFS server.

Samba (File Service)

If any application needs data stored in a directory on a faulty Samba server, it cannot start or function properly.

NIS (User Management)

If your openSUSE Leap system relies on a faulty NIS server to provide the user data, users cannot log in to this machine.

LDAP (User Management)

If your openSUSE Leap system relies on a faulty LDAP server to provide the user data, users cannot log in to this machine.

Kerberos (Authentication)

Authentication will not work and login to any machine fails.

CUPS (Network Printing)

Users cannot print.

4. Check whether the network servers are running and whether your network setup allows you to establish a connection:

Important: Limitations

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`** *`IP_ADDRESS/HOSTNAME`* (replace with the host name or IP address 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`** *`IP address`* or *`YOUR_HOSTNAME`* from another machine. If you can reach your machine from another machine, it is the server that is not running or not configured correctly.
If ping fails with `unknown host`, the name service is not configured correctly or the host name used was incorrect. For further checks on this matter, refer to *Step 4.b*. If ping still fails, either your network card is not configured correctly or your network hardware is faulty.

b. Use **`host`** *`HOSTNAME`* to check whether the host name 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:

`/var/run/netconfig/resolv.conf`

This file is used to keep track of the name server and domain you are currently using. It is a symbolic link to `/run/netconfig/resolv.conf` and is usually automatically adjusted by YaST or DHCP. 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 Network Settings module (Hostname/DNS tab).
If your network connection is handled via DHCP, enable DHCP to change host name and name service information by selecting *Set Hostname via DHCP* (can be set globally for any interface or per interface) and *Update Name Servers and Search List via DHCP* in the YaST Network Settings module (Hostname/DNS tab).

`/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 managed by YaST, but it would be prudent 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. For detailed information about DNS, refer to *Book "Reference", Chapter 19 "The Domain Name System"*. 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 **`ip addr show`** *`NETWORK_DEVICE`* to check whether this device was properly configured. Make sure that the `inet address` with the netmask (*`/MASK`*) is 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) that 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. Use a combination of traceroute and ping to track down the culprit and let the administrators know.

When 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.

17.4.1 NetworkManager Problems

If you have a problem with network connectivity, narrow it down as described in *Procedure 17.1, "How to Identify Network Problems"*. If NetworkManager seems to be the culprit, proceed as follows to get logs providing hints on why NetworkManager fails:

1. Open a shell and log in as `root`.
2. Restart the NetworkManager:

```
tux > sudo systemctl restart NetworkManager
```

3. Open a Web page, for example, http://www.opensuse.org as normal user to see, if you can connect.
4. Collect any information about the state of NetworkManager in `/var/log/NetworkManager`.

For more information about NetworkManager, refer to *Book "Reference", Chapter 28 "Using NetworkManager"*.

17.5 Data Problems

Data problems are when the machine may or may 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 the system state from before your system failed.

17.5.1 Managing Partition Images

Sometimes you need to perform a backup from an entire partition or even hard disk. Linux comes with the **dd** tool which can create an exact copy of your disk. Combined with **gzip** you save some space.

PROCEDURE 17.2: BACKING UP AND RESTORING HARD DISKS

1. Start a Shell as user `root`.
2. Select your source device. Typically this is something like `/dev/sda` (labeled as *SOURCE*).
3. Decide where you want to store your image (labeled as *BACKUP_PATH*). It must be different from your source device. In other words: if you make a backup from `/dev/sda`, your image file must not to be stored under `/dev/sda`.
4. Run the commands to create a compressed image file:

   ```
   root # dd if=/dev/SOURCE | gzip > /BACKUP_PATH/image.gz
   ```

5. Restore the hard disk with the following commands:

   ```
   root # gzip -dc /BACKUP_PATH/image.gz | dd of=/dev/SOURCE
   ```

If you only need to back up a partition, replace the *SOURCE* placeholder with your respective partition. In this case, your image file can lie on the same hard disk, but on a different partition.

17.5.2 Using the Rescue System

There are several reasons a system could fail to come up and run properly. A corrupted file system following a system crash, corrupted configuration files, or a corrupted boot loader configuration are the most common ones.

To help you to resolve these situations, openSUSE Leap contains a rescue system that you can boot. 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 re-install the boot loader configuration.
- Recover from a badly installed device driver or unusable kernel.
- Resize partitions using the parted command. Find more information about this tool at the GNU Parted Web site 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 medium.

1. Insert the installation medium into your DVD drive.
2. Reboot the system.
3. At the boot screen, press F4 and choose *DVD-ROM*. Then choose *Rescue System* from the main menu.
4. Enter `root` at the `Rescue:` prompt. A password is not required.

If your hardware setup does not include a DVD drive, you can boot the rescue system from a network source. The following example applies to a remote boot scenario—if using another boot medium, such as a DVD, modify the `info` file accordingly and boot as you would for a normal installation.

1. Enter the configuration of your PXE boot setup and add the lines `install=`*`PROTOCOL`*`://`*`INSTSOURCE`* and `rescue=1`. If you need to start the repair system, use `repair=1` instead. 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.

When you have entered the rescue system, you can use the virtual consoles that can be reached with Alt–F1 to Alt–F6.

A shell and 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`**, and **`shutdown`**, **`ip`** and **`ss`** 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 system log with **`journalctl`**.

17.5.2.1 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:

   ```
   tux > sudo mount /dev/sda6 /mnt
   ```

   All directories of the system are now located under `/mnt`
3. Change the directory to the mounted root file system:

   ```
   tux > sudo 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:

   ```
   tux > sudo umount /mnt
   ```
6. Reboot the machine.

17.5.2.2 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. The system contains the utilities to check and repair the `btrfs`, `ext2`, `ext3`, `ext4`, `xfs`, `dosfs`, and `vfat` file systems. Look for the command **`fsck.`***`FILESYSTEM`*. For example, if you need a file system check for `btrfs`, use **`fsck.btrfs`**.

17.5.2.3 Accessing the Installed System

If you need to access the installed system from the rescue system, you need to do this in a *change root* environment. For example, to modify the boot loader configuration, or to execute a hardware configuration utility.

To set up a change root environment based on the installed system, proceed as follows:

1. 

   ### Tip: Import LVM Volume Groups

   If you are using an LVM setup (refer to *Book "*Reference*", Chapter 5 "Expert Partitioner", Section 5.2 "LVM Configuration"* for more general details), import all existing volume groups to be able to find and mount the device(s):

   ```
   rootvgimport -a
   ```

   Run **`lsblk`** to check which node corresponds to the root partition. It is `/dev/sda2` in our example:

   ```
   tux > lsblk
   NAME        MAJ:MIN RM   SIZE RO TYPE  MOUNTPOINT
   sda           8:0    0 149,1G  0 disk
   ├─sda1        8:1    0     2G  0 part  [SWAP]
   ├─sda2        8:2    0    20G  0 part  /
   └─sda3        8:3    0   127G  0 part
     └─cr_home 254:0    0   127G  0 crypt /home
   ```

2. Mount the root partition from the installed system:

   ```
   tux > sudo mount /dev/sda2 /mnt
   ```

3. Mount `/proc`, `/dev`, and `/sys` partitions:

```
tux > sudo mount -t proc none /mnt/proc
tux > sudo mount --rbind /dev /mnt/dev
tux > sudo mount --rbind /sys /mnt/sys
```

4. Now you can "change root" into the new environment, keeping the `bash` shell:

```
tux > chroot /mnt /bin/bash
```

5. Finally, mount the remaining partitions from the installed system:

```
tux > mount -a
```

6. 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, not with the change root environment. It only supports essential hardware and it is not possible to add kernel modules from the installed system unless the kernel versions are identical. Always check the version of the currently running (rescue) kernel with **`uname -r`** and then find out if a matching subdirectory exists in the `/lib/modules` directory in the change root environment. If yes, you can use the installed modules, otherwise you need to supply their correct versions on other media, such as a flash disk. Most often the rescue kernel version differs from the installed one — then you cannot simply 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.

17.5.2.4 Modifying and Re-installing 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 re-install the boot loader, proceed as follows:

1. Perform the necessary steps to access the installed system as described in *Section 17.5.2.3, "Accessing the Installed System"*.
2. Check that the GRUB 2 boot loader is installed on the system. If not, install the package `grub2` and run

   ```
   tux > sudo grub2-install /dev/sda
   ```
3. Check whether the following files are correctly configured according to the GRUB 2 configuration principles outlined in *Book "Reference", Chapter 12 "The Boot Loader GRUB 2"* and apply fixes if necessary.

   - `/etc/default/grub`
   - `/boot/grub2/device.map` (optional file, only present if created manually)
   - `/boot/grub2/grub.cfg` (this file is generated, do not edit)
   - `/etc/sysconfig/bootloader`
4. Re-install the boot loader using the following command sequence:

   ```
   tux > sudo grub2-mkconfig -o /boot/grub2/grub.cfg
   ```
5. Unmount the partitions, log out from the "change root" environment, and reboot the system:

   ```
   tux > umount -a
   exit
   reboot
   ```

17.5.2.5 Fixing Kernel Installation

A kernel update may introduce a new bug which can impact the operation of your system. For example a driver for a piece of hardware in your system may be faulty, which prevents you from accessing and using it. In this case, revert to the last working kernel (if available on the system) or install the original kernel from the installation media.

Tip: How to Keep Last Kernels after Update

To prevent failures to boot after a faulty kernel update, use the kernel multiversion feature and tell `libzypp` which kernels you want to keep after the update.

For example to always keep the last two kernels and the currently running one, add

```
multiversion.kernels = latest,latest-1,running
```

to the `/etc/zypp/zypp.conf` file. See *Book "Reference", Chapter 6 "Installing Multiple Kernel Versions"* for more information.

A similar case is when you need to re-install or update a broken driver for a device not supported by openSUSE Leap. For example when a hardware vendor uses a specific device, such as a hardware RAID controller, which needs a binary driver to be recognized by the operating system. The vendor typically releases a Driver Update Disk (DUD) with the fixed or updated version of the required driver.

In both cases you need to access the installed system in the rescue mode and fix the kernel related problem, otherwise the system may fail to boot correctly:

1. Boot from the openSUSE Leap installation media.
2. If you are recovering after a faulty kernel update, skip this step. If you need to use a driver update disk (DUD), press `F6` to load the driver update after the boot menu appears, and choose the path or URL to the driver update and confirm with *Yes*.
3. Choose *Rescue System* from the boot menu and press `Enter`. If you chose to use DUD, you will be asked to specify where the driver update is stored.
4. Enter `root` at the `Rescue:` prompt. A password is not required.
5. Manually mount the target system and "change root" into the new environment. For more information, see *Section 17.5.2.3, "Accessing the Installed System"*.
6. If using DUD, install/re-install/update the faulty device driver package. Always make sure the installed kernel version exactly matches the version of the driver you are installing.
   If fixing faulty kernel update installation, you can install the original kernel from the installation media with the following procedure.
   a. Identify your DVD device with **`hwinfo --cdrom`** and mount it with **`mount /dev/sr0 /mnt`**.

b. Navigate to the directory where your kernel files are stored on the DVD, for example `cd /mnt/suse/x86_64/`.

c. Install required `kernel-*`, `kernel-*-base`, and `kernel-*-extra` packages of your flavor with the `rpm -i` command.

7. Update configuration files and reinitialize the boot loader if needed. For more information, see *Section 17.5.2.4, "Modifying and Re-installing the Boot Loader"*.

8. Remove any bootable media from the system drive and reboot.

A GNU Licenses

This appendix contains the GNU Free Documentation License version 1.2.

GNU Free Documentation License

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or non-commercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or non-commercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

```
Copyright (c) YEAR YOUR NAME.
Permission is granted to copy, distribute
 and/or modify this document
under the terms of the GNU Free
 Documentation License, Version 1.2
or any later version published by the Free
 Software Foundation;
with no Invariant Sections, no Front-Cover
 Texts, and no Back-Cover Texts.
A copy of the license is included in the
 section entitled “GNU
Free Documentation License”.
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the “with...Texts.” line with this:

```
with the Invariant Sections being LIST
 THEIR TITLES, with the
Front-Cover Texts being LIST, and with the
 Back-Cover Texts being LIST.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.