



SUSE Linux Enterprise Micro 5.1

Deployment Guide

Deployment Guide

SUSE Linux Enterprise Micro 5.1

This guide details how to install single or multiple systems, and how to exploit the product-inherent capabilities for a deployment infrastructure.

Publication Date: May 02, 2024

<https://documentation.suse.com> 

Copyright © 2006–2024 SUSE LLC and contributors. All rights reserved.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or (at your option) version 1.3; with the Invariant Section being this copyright notice and license. A copy of the license version 1.2 is included in the section entitled “GNU Free Documentation License”.

For SUSE trademarks, see <https://www.suse.com/company/legal/>. All third-party trademarks are the property of their respective owners. Trademark symbols (®, ™ etc.) denote trademarks of SUSE and its affiliates. Asterisks (*) denote third-party trademarks.

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

Contents

Preface ix

- 1 Available documentation ix
- 2 Improving the documentation x
- 3 Documentation conventions xi
- 4 Support xiii
 - Support statement for SUSE Linux Enterprise Micro xiii • Technology previews xiv

I INSTALLATION PREPARATION 1

1 Planning for SUSE Linux Enterprise Micro 2

- 1.1 Considerations for deployment of SUSE Linux Enterprise Micro 2
- 1.2 Deployment of SUSE Linux Enterprise Micro 2
- 1.3 Running SUSE Linux Enterprise Micro 3
- 1.4 Registering SUSE Linux Enterprise Micro 3

2 Installation on AMD64 and Intel 64 5

- 2.1 Hardware requirements 5
- 2.2 Installation considerations 6
 - Installation on hardware or virtual machine 6 • Installation target 6
- 2.3 Controlling the installation 6
- 2.4 Booting the installation system 7
- 2.5 Dealing with boot and installation problems 9
 - Problems booting 9 • Problems installing 10 • Redirecting the boot source to the installation medium 10

3 Installation on Arm AArch64 11

- 3.1 Hardware requirements 11
- 3.2 Installation considerations 12
 - Installation on hardware or virtual machine 12 • Installation target 12
- 3.3 Controlling the installation 13
- 3.4 Booting the installation system 13
- 3.5 Dealing with boot and installation problems 15
 - Problems booting 15 • Problems installing 16 • Redirecting the boot source to the boot DVD 16
- 3.6 Raspberry Pi 16
 - Boot process 17 • Lack of a real-time clock 19 • Deploying an appliance image 20 • Installation from USB media 20 • Installation from network 21 • More information 21

4 Installation on IBM Z and LinuxONE 23

- 4.1 System requirements 23
 - Hardware 23 • MicroCode Level, APARs, and fixes 25 • Software 26
- 4.2 Preparing for installation 26
 - Making the installation data available 27 • Installation types 33 • Preparing the IPL of the SUSE Linux Enterprise Micro installation system 34 • IPLing the SUSE Linux Enterprise Micro installation system 38 • Network configuration 43 • Connecting to the SUSE Linux Enterprise Micro installation system 46 • The SUSE Linux Enterprise Micro boot procedure on IBM Z 49
- 4.3 Secure boot 49
- 4.4 The parmfile—automating the system configuration 50
 - General parameters 51 • Configuring the network interface 52 • Specifying the installation source and YaST interface 54 • I/O device auto-configuration on IBM Z systems 55 • Example parmfiles 57
- 4.5 Using the vt220 terminal emulator 58

4.6	More information	58
	General documents about Linux on IBM Z	58 • Technical issues of Linux on IBM Z
	IBM Z	59 • Advanced configurations for Linux on IBM Z
	IBM Z	59
II	RAW IMAGE DEPLOYMENT	61
5	Description of raw images	62
5.1	Default partitioning	62
5.2	First boot detection	63
6	Deploying raw images	64
6.1	Registration	65
6.2	Managing extensions	66
	Activating SUSE Linux Enterprise Live Patching	66 • Deactivating SUSE Linux Enterprise Live Patching
	SUSE Linux Enterprise Live Patching	67
7	Configuring with Ignition	68
7.1	About Ignition	68
7.2	config.ign	68
	Configuration examples	69 • Converting YAML fcc file to JSON ign
		77
8	Configuring with Combustion	79
8.1	About Combustion	79
8.2	The script configuration file	80
	Network configuration	80 • Partitioning
	81 • Setting a password for root	82 • Adding SSH keys
	82 • Enabling services	82 • Installing packages
	82	
III	MANUAL INSTALLATION	84
9	Boot parameters	85
9.1	Using the default boot parameters	85

9.2	PC (AMD64/Intel 64/Arm AArch64)	85
	The boot screen on machines equipped with traditional BIOS	86
	The boot screen on machines equipped with UEFI	88
9.3	List of important boot parameters	90
	General boot parameters	90
	Configuring the network interface	91
	Specifying the installation source	93
	Specifying remote access	94
9.4	Advanced setups	95
	Using IPv6 for the installation	95
	Using a proxy for the installation	95
	Enabling SELinux support	96
	Scale user interface for high DPI	96
	Using CPU mitigations	96
9.5	More information	97
10	Installation steps	98
10.1	Overview	98
10.2	Network settings	99
10.3	Language, Keyboard, and License Agreement	101
10.4	IBM Z: disk activation	102
	Configuring DASD disks	103
	Configuring zFCP disks	104
10.5	Registration	105
	Loading registration codes from USB storage	107
10.6	Extension and Module Selection	108
10.7	NTP Configuration	109
10.8	Authentication for the System Administrator root	110
10.9	Installation Settings	111
	Partitioning	112
	Software	124
	Timezone	125
	Network Configuration	126
	Booting	127
	Kdump	128
	System	129
	Security	131
11	Remote installation	133
11.1	Overview	133

- 11.2 Scenarios for remote installation 134
 - Installation from source media via VNC 134 • Installation from network via VNC 135 • Installation from source media via SSH 136 • Installation from network via SSH 137
- 11.3 Monitoring installation via VNC 138
 - Preparing for VNC installation 138 • Connecting to the installation program 139
- 11.4 Monitoring installation via SSH 140
 - Preparing for SSH installation 140 • Connecting to the installation program 140
- 11.5 Monitoring installation via serial console 141
- 12 Troubleshooting 142**
 - 12.1 Checking media 142
 - 12.2 No bootable drive available 142
 - 12.3 Booting from installation media fails 143
 - 12.4 Boot failure 144
 - 12.5 Fails to launch graphical installer 146
 - 12.6 Only minimalist boot screen started 147
- A GNU licenses 149**

Preface

1 Available documentation

Online documentation

Our documentation is available online at <https://documentation.suse.com>. Browse or download the documentation in various formats.



Note: Latest updates

The latest updates are usually available in the English-language version of this documentation.

SUSE Knowledgebase

If you have run into an issue, also check out the Technical Information Documents (TIDs) that are available online at <https://www.suse.com/support/kb/>. Search the SUSE Knowledgebase for known solutions driven by customer need.

Release notes

For release notes, see <https://www.suse.com/releasesnotes/>.

In your system

For offline use, the release notes are also available under `/usr/share/doc/release-notes` on your system. The documentation for individual packages is available at `/usr/share/doc/packages`.

Many commands are also described in their *manual pages*. To view them, run `man`, followed by a specific command name. If the `man` command is not installed on your system, install it with `sudo zypper install man`.

2 Improving the documentation

Your feedback and contributions to this documentation are welcome. The following channels for giving feedback are available:

Service requests and support

For services and support options available for your product, see <https://www.suse.com/support/>.

To open a service request, you need a SUSE subscription registered at SUSE Customer Center. Go to <https://scc.suse.com/support/requests>, log in, and click *Create New*.

Bug reports

Report issues with the documentation at <https://bugzilla.suse.com/>.

To simplify this process, click the *Report an issue* icon next to a headline in the HTML version of this document. This preselects the right product and category in Bugzilla and adds a link to the current section. You can start typing your bug report right away.

A Bugzilla account is required.

Contributions

To contribute to this documentation, click the *Edit source document* icon next to a headline in the HTML version of this document. This will take you to the source code on GitHub, where you can open a pull request.

A GitHub account is required.



Note: *Edit source document* only available for English

The *Edit source document* icons are only available for the English version of each document. For all other languages, use the *Report an issue* icons instead.

For more information about the documentation environment used for this documentation, see the repository's README.

Mail

You can also report errors and send feedback concerning the documentation to doc-team@suse.com. Include the document title, the product version, and the publication date of the document. Additionally, include the relevant section number and title (or provide the URL) and provide a concise description of the problem.

3 Documentation conventions

The following notices and typographic conventions are used in this document:

- `/etc/passwd`: Directory names and file names
- `PLACEHOLDER`: Replace `PLACEHOLDER` with the actual value
- `PATH`: An environment variable
- `ls`, `--help`: Commands, options, and parameters
- `user`: The name of a user or group
- `package_name`: The name of a software 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
- `AMD/Intel` This paragraph is only relevant for the AMD64/Intel 64 architectures. The arrows mark the beginning and the end of the text block. `<`
- `IBM Z, POWER` This paragraph is only relevant for the architectures `IBM Z` and `POWER`. The arrows mark the beginning and the end of the text block. `<`
- *Chapter 1, “Example chapter”*: A cross-reference to another chapter in this guide.
- Commands that must be run with `root` privileges. You can also prefix these commands with the `sudo` command to run them as a non-privileged user:

```
# command
> sudo command
```

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

```
> command
```

- Commands can be split into two or multiple lines by a backslash character (`\`) at the end of a line. The backslash informs the shell that the command invocation will continue after the line's end:

```
> echo a b \
```

```
c d
```

- A code block that shows both the command (preceded by a prompt) and the respective output returned by the shell:

```
> command  
output
```

- 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.

- Compact Notices



Additional information, for example about differences in software versions.



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

4 Support

Find the support statement for SUSE Linux Enterprise Micro and general information about technology previews below. For details about the product lifecycle, see <https://www.suse.com/lifecycle>.

If you are entitled to support, find details on how to collect information for a support ticket at <https://documentation.suse.com/sles-15/html/SLES-all/cha-adm-support.html>.

4.1 Support statement for SUSE Linux Enterprise Micro

To receive support, you need an appropriate subscription with SUSE. To view the specific support offers available to you, go to <https://www.suse.com/support/> and select your product.

The support levels are defined as follows:

L1

Problem determination, which means technical support designed to provide compatibility information, usage support, ongoing maintenance, information gathering and basic troubleshooting using available documentation.

L2

Problem isolation, which means technical support designed to analyze data, reproduce customer problems, isolate a problem area and provide a resolution for problems not resolved by Level 1 or prepare for Level 3.

L3

Problem resolution, which means technical support designed to resolve problems by engaging engineering to resolve product defects which have been identified by Level 2 Support.

For contracted customers and partners, SUSE Linux Enterprise Micro is delivered with L3 support for all packages, except for the following:

- Technology previews.
- Sound, graphics, fonts, and artwork.
- Packages that require an additional customer contract.
- Packages with names ending in `-devel` (containing header files and similar developer resources) will only be supported together with their main packages.


SUSE will only support the usage of original packages. That is, packages that are unchanged and not recompiled.

4.2 Technology previews

Technology previews are packages, stacks, or features delivered by SUSE to provide glimpses into upcoming innovations. Technology previews are included for your convenience to give you a chance to test new technologies within your environment. We would appreciate your feedback. If you test a technology preview, please contact your SUSE representative and let them know about your experience and use cases. Your input is helpful for future development.

Technology previews have the following limitations:

- Technology previews are still in development. Therefore, they may be functionally incomplete, unstable, or otherwise *not* suitable for production use.
- Technology previews are *not* supported.
- Technology previews may only be available for specific hardware architectures.
- Details and functionality of technology previews are subject to change. As a result, upgrading to subsequent releases of a technology preview may be impossible and require a fresh installation.
- SUSE may discover that a preview does not meet customer or market needs, or does not comply with enterprise standards. Technology previews can be removed from a product at any time. SUSE does not commit to providing a supported version of such technologies in the future.

For an overview of technology previews shipped with your product, see the release notes at <https://www.suse.com/releasenotes> .

I Installation preparation

- 1 Planning for SUSE Linux Enterprise Micro 2
- 2 Installation on AMD64 and Intel 64 5
- 3 Installation on Arm AArch64 11
- 4 Installation on IBM Z and LinuxONE 23

1 Planning for SUSE Linux Enterprise Micro

This chapter describes some basic considerations before installing SUSE Linux Enterprise Micro.

1.1 Considerations for deployment of SUSE Linux Enterprise Micro

The implementation of an operating system either in an existing IT environment or as a completely new roll out must be carefully prepared. At the beginning of the planning process, you should try to define the project goals and necessary features. This must always be done individually for each project, but the questions to answer should include the following:

- How many installations should be done? Depending on this, the best deployment methods differ.
- Will the system be in a hostile environment? Have a look at [to get an overview of consequences](#).
- How will you get regular updates? All patches are provided online for registered users. Find the registration and patch support database at <http://download.suse.com/>.
- Do you need help for your local installation? SUSE provides training, support, and consulting for all topics pertaining to SUSE Linux Enterprise Micro. Find more information about this at <https://www.suse.com/products/server/>.
- Do you need third-party products? Make sure that the required product is also supported on the desired platform. SUSE can provide help to support software on different platforms when needed.

1.2 Deployment of SUSE Linux Enterprise Micro

To make sure that your system will run flawlessly, always try to use certified hardware. The hardware certification process is an ongoing process and the database of certified hardware is updated regularly. Find the search form for certified hardware at <https://www.suse.com/yesssearch/Search.jsp>.

In case you intend to install only several nodes of SLE Micro, you can choose the manual installation or you can directly deploy raw images. For a large-scale deployment it is recommended to use the automatic installation by using AutoYaST, SUSE Manager, or wherever you can easily copy the raw images to the desired machines, you can also use the raw images for a large-scale deployment.

SUSE Linux Enterprise Micro provides you with a broad variety of services.

In addition to the plain software installation, you should consider training the end users of the systems and help desk staff.



Note: Terminology

In the following sections, the system to hold your new SUSE Linux Enterprise Micro installation is called *target system* or *installation target*. The term *repository* (previously called “installation source”) is used for all sources of installation data. This includes physical media, such as CD, DVD, or USB flash drive, and network servers distributing the installation data in your network.

1.3 Running SUSE Linux Enterprise Micro

The SUSE Linux Enterprise Micro operating system is a well-tested and stable system. Unfortunately, this does not prevent hardware failures or other causes for downtime or data loss. For any serious computing task where data loss could occur, a regular backup should be done.

For optimal security and data safety, you should make regular updates of all the operated machines. If you have a mission critical server, you should run a second identical (pre-production) machine that you can use to test all changes. This also gives you the possibility of switching machines in the case of hardware failure.

1.4 Registering SUSE Linux Enterprise Micro

To get technical support and product updates, you need to register and activate your SUSE product with the SUSE Customer Center. We recommend to register during the installation, since this will enable you to install the system with the latest updates and patches available. However, if you are offline or want to skip the registration step, you can register at any time later from the installed system.

In case your organization does not provide a local registration server, registering SUSE Linux Enterprise requires a SUSE Customer Center account. In case you do not have one yet, go to the SUSE Customer Center home page (<https://scc.suse.com/>) to create one.

During the manual installation you will be asked to enter your registration code. For details, refer to [Section 10.5, "Registration"](#). If you deploy SLE Micro raw images, you need to register your system after the installation, for details refer to [Section 6.1, "Registration"](#).

If you deploy your instances automatically using AutoYaST, you can register the system during the installation by providing the respective information in the AutoYaST control file. For details, see *Book "AutoYaST Guide", Chapter 4 "Configuration and installation options", Section 4.3 "System registration and extension selection"*.

2 Installation on AMD64 and Intel 64

This chapter describes the steps necessary to prepare for the installation of SUSE Linux Enterprise Micro on AMD64 and Intel 64 computers. It introduces the steps required to prepare for various installation methods. The list of hardware requirements provides an overview of systems supported by SUSE Linux Enterprise Micro. Find information about available installation methods and several commonly known problems. Also learn how to control the installation, provide installation media, and boot with regular methods.

2.1 Hardware requirements

The SUSE® Linux Enterprise Micro operating system can be deployed on a wide range of hardware. It is impossible to list all the different combinations of hardware SUSE Linux Enterprise Server supports. However, to provide you with a guide to help you during the planning phase, the minimum requirements are presented here.

If you want to be sure that a given computer configuration will work, find out which platforms have been certified by SUSE. Find a list at <https://www.suse.com/yessearch/>.

The Intel 64 and AMD64 architectures support the simple migration of x86 software to 64 bits. Like the x86 architecture, they constitute a value-for-money alternative.

CPU

All CPUs available on the market to date are supported.

Maximum number of CPUs

The maximum number of CPUs supported by software design is 8192 for Intel 64 and AMD64. If you plan to use such a large system, verify with our hardware system certification Web page for supported devices, see <https://www.suse.com/yessearch/>.

Memory requirements

SLE Micro requires at least 1GB RAM. Bear in mind that this is a minimal value for the operation system, the actual memory size depends on the workload.

Hard disk requirements

The minimum hard disk space is 12GB. But the recommended value is 20GB of hard disk space. Adjust the value according to workloads of your containers.

2.2 Installation considerations

This section encompasses many factors that need to be considered before installing SUSE Linux Enterprise Micro on AMD64 and Intel 64 hardware.

2.2.1 Installation on hardware or virtual machine

SUSE Linux Enterprise Micro is normally installed as an independent operating system. With virtualization it is also possible to run multiple instances of SUSE Linux Enterprise Server on the same hardware. However, the installation of the VM Host Server is performed like a typical installation with some additional packages.

2.2.2 Installation target

Most installations are to a local hard disk. Therefore, it is necessary for the hard disk controllers to be available to the installation system. If a special controller (like a RAID controller) needs an extra kernel module, provide a kernel module update disk to the installation system.

Other installation targets may be various types of block devices that provide sufficient disk space and speed to run an operating system. This includes network block devices like iSCSI or SAN. It is also possible to install on network file systems that offer the standard Unix permissions. However, it may be problematic to boot these, because they must be supported by the initramfs before the actual system can start. Such installations can be useful when you need to start the same system in different locations or you plan to use virtualization features like domain migration.

2.3 Controlling the installation

Control the installation in one of several ways. Boot the setup with one of the options listed in [Section 2.4, "Booting the installation system"](#). To enable the different control methods refer to [Section 9.3.4, "Specifying remote access"](#). For information about how to use each remote control method, refer to [Chapter 11, Remote installation](#).

A brief overview of the different methods:

Local with monitor and keyboard

This is the method most frequently used to install SUSE Linux Enterprise Micro. This also requires the smallest preparation effort but requires a lot of direct interaction.

Remote via SSH

You can control the installation via SSH either in text mode or use X-forwarding for a graphical installation. For details refer to [Section 11.4, “Monitoring installation via SSH”](#).

Remote via serial console

For this installation method you need a second computer connected by a *null modem* cable to the computer on which to install SUSE Linux Enterprise Micro. The installation then proceeds in text mode. For details refer to [Section 11.5, “Monitoring installation via serial console”](#).

Remote via VNC

Use this method if you want a graphical installation without direct access to the target machine. For details refer to [Section 11.3, “Monitoring installation via VNC”](#).

2.4 Booting the installation system

This section gives an overview of the steps required for the complete installation of SUSE® Linux Enterprise Micro.

In case of the manual installation, an overview of booting the installation system is provided by the following procedure:

1. Prepare the installation media.

USB Flash Drive

In case of the manual installation from an ISO, creating a bootable flash disk is the easiest way to start the installation. You need to copy the ISO to the device using the **dd**. The flash disk must not be mounted, and all data on the device will be erased.

```
# dd if=PATH_TO_ISO_IMAGE of=USB_STORAGE_DEVICE bs=4M
```

Network booting

If the target computer's firmware supports it, you can boot the computer from the network and install from a server. This booting method requires a boot server that provides the needed boot images over the network. The exact protocol depends on your hardware. Commonly you need several services, such as TFTP and DHCP or PXE boot.

It is possible to install from many common network protocols, such as NFS, HTTP, FTP, or SMB. For more information on how to perform such an installation, refer to [Chapter 11, Remote installation](#).

2. Configure the target system firmware to boot the medium you chose. Refer to the documentation of your hardware vendor about how to configure the correct boot order.
3. Set the boot parameters required for your installation control method. An overview of the different methods is provided in [Section 2.3, "Controlling the installation"](#). A list of boot parameters is available in [Chapter 9, Boot parameters](#).
4. Perform the installation as described in [Chapter 10, Installation steps](#). The system needs to restart after the installation is finished.
5. Optional: Change the boot order of the system to directly boot from the medium to which SUSE Linux Enterprise Micro has been installed. If the system boots from the installation medium, the first boot parameter will be to boot the installed system.

For the deployment of raw images, the procedure is the following:

1. Prepare the raw image, for details refer to [Procedure 6.1, "Preparing the raw disk image"](#).
2. Prepare the configuration media, for details refer to [Procedure 6.2, "Preparing the configuration device"](#).
3. Configure the target system firmware to boot the medium where you copied the raw image. Refer to the documentation of your hardware vendor about how to configure the correct boot order.
4. Perform the installation as described in [Chapter 6, Deploying raw images](#). The system needs to restart after the installation is finished.

2.5 Dealing with boot and installation problems

Prior to delivery, SUSE® Linux Enterprise Micro is subjected to an extensive test program. Despite this, problems occasionally occur during boot or installation.

2.5.1 Problems booting

Boot problems may prevent the YaST installer from starting on your system. Another symptom is when your system does not boot after the installation has been completed.

Installed system boots, not media

Change your computer's firmware or BIOS so that the boot sequence is correct. To do this, consult the manual for your hardware.

The computer hangs

Change the console on your computer so that the kernel outputs are visible. Be sure to check the last outputs. This is normally done by pressing `Ctrl – Alt – F10`. If you cannot resolve the problem, consult the SUSE Linux Enterprise Micro support staff. To log all system messages at boot time, use a serial connection as described in [Section 2.3, “Controlling the installation”](#).

Boot disk

The boot disk is a useful interim solution if you have difficulties setting the other configurations or if you want to postpone the decision regarding the final boot mechanism.

Virus warning after installation

There are BIOS variants that check the structure of the boot sector (MBR) and erroneously display a virus warning after the installation of GRUB 2. Solve this problem by entering the BIOS and looking for corresponding adjustable settings. For example, switch off *virus protection*. You can switch this option back on again later. It is unnecessary, however, if Linux is the only operating system you use.

2.5.2 Problems installing

If an unexpected problem occurs during installation, information is needed to determine the cause of the problem. Use the following directions to help with troubleshooting:

- Check the outputs on the various consoles. You can switch consoles with the key combination `Ctrl - Alt - Fn`. For example, obtain a shell in which to execute various commands by pressing `Ctrl - Alt - F2`.
- Try launching the installation with “Safe Settings” (press `F5` on the installation screen and choose *Safe Settings*). If the installation works without problems in this case, there is an incompatibility that causes either ACPI or APIC to fail. In some cases, a BIOS or firmware update fixes this problem.
- Check the system messages on a console in the installation system by entering the command `dmesg -T`.

2.5.3 Redirecting the boot source to the installation medium

To simplify the installation process and avoid accidental installations, the default setting on the installation medium for SUSE Linux Enterprise Micro is that your system is booted from the first hard disk. At this point, an installed boot loader normally takes over control of the system. This means that the boot medium can stay in the drive during the installation. To start the installation, choose one of the installation possibilities in the boot menu of the media.

3 Installation on Arm AArch64

This chapter describes the steps necessary to prepare for the installation of SUSE Linux Enterprise Micro on Arm AArch64 computers. It introduces the steps required to prepare for various installation methods. The list of hardware requirements provides an overview of systems supported by SUSE Linux Enterprise Server. Find information about available installation methods and several common known problems. Also learn how to control the installation, provide installation media, and boot with regular methods.

3.1 Hardware requirements

The SUSE® Linux Enterprise Micro operating system can be deployed on a wide range of hardware. It is impossible to list all the different combinations of hardware SUSE Linux Enterprise Micro supports. However, to provide you with a guide to help you during the planning phase, the minimum requirements are presented here.

If you want to be sure that a given computer configuration will work, find out which platforms have been certified by SUSE. Find a list at <https://www.suse.com/yessearch/>.

CPU

The minimum requirement is a CPU that supports the Armv8-A instruction set architecture (ISA), for example, Arm Cortex-A53 or Cortex-A57. Refer to <https://www.arm.com/products/processors/cortex-a/> for a list of available Armv8-A processors.

CPUs with the Armv8-R (realtime) and Armv8-M (microcontroller) ISA are currently not supported.

Maximum number of CPUs

The maximum number of CPUs supported by software design is 256. If you plan to use such a large system, check our hardware system certification Web page for supported devices, see <https://www.suse.com/yessearch/>.

Memory requirements

A minimum of 1024 MB of memory is required for a minimal installation. On machines with more than two processors, add 512 MB per CPU. For remote installations via HTTP or FTP, add another 150 MB. Note that these values are only valid for the installation of the

operating system—the actual memory requirement in production depends on the system's workload. For systems running the GNOME desktop environment, a minimum of 2048 MB of memory is required and 4096 MB is recommended.

Hard disk requirements

The disk requirements depend largely on workload of your containers. Commonly, you need more space than the installation software itself needs to have a system that works properly. The minimum required value is 12 GB. The recommended value is 20 GB.

Boot methods

The computer can be booted from a USB disk or a network. A special boot server is required to boot over the network. This can be set up with SUSE Linux Enterprise Server.

3.2 Installation considerations

This section encompasses many factors that need to be considered before installing SUSE Linux Enterprise Micro on Arm AArch64 hardware.

3.2.1 Installation on hardware or virtual machine

SUSE Linux Enterprise Micro is normally installed as an independent operating system. With virtualization it is also possible to run multiple instances of SLE Micro on the same hardware. However, the installation of the VM Host Server is performed like a typical installation with some additional packages.

3.2.2 Installation target

Most installations are to a local hard disk. Therefore, it is necessary for the hard disk controllers to be available to the installation system. If a special controller (like a RAID controller) needs an extra kernel module, provide a kernel module update disk to the installation system.

Other installation targets may be various types of block devices that provide sufficient disk space and speed to run an operating system. This includes network block devices like iSCSI or SAN. It is also possible to install on network file systems that offer the standard Unix permissions. However, it may be problematic to boot these, because they must be supported by

the `initramfs` before the actual system can start. Such installations can be useful when you need to start the same system in different locations or you plan to use virtualization features like domain migration.

3.3 Controlling the installation

Control the installation in one of several ways. Boot the setup with one of the options listed in [Section 3.4, “Booting the installation system”](#). To enable the different control methods refer to [Section 9.3.4, “Specifying remote access”](#). For information about how to use each remote control method, refer to [Chapter 11, Remote installation](#).

A brief overview of the different methods:

Local with monitor and keyboard

This is the method most frequently used to install SUSE Linux Enterprise Micro. This also requires the smallest preparation effort but requires a lot of direct interaction.

Remote via SSH

You can control the installation via SSH either in text mode or use X-forwarding for a graphical installation. For details refer to [Section 11.4, “Monitoring installation via SSH”](#).

Remote via serial console

For this installation method you need a second computer connected by a *null modem* cable to the computer on which to install SUSE Linux Enterprise Micro. The installation then proceeds in text mode. For details refer to [Section 11.5, “Monitoring installation via serial console”](#).

Remote via VNC

Use this method if you want a graphical installation without direct access to the target machine. For details refer to [Section 11.3, “Monitoring installation via VNC”](#).

3.4 Booting the installation system

This section gives an overview of the steps required for the complete installation of SUSE® Linux Enterprise Micro.

For a full description of how to install and configure the system with YaST, refer to [Part III, “Manual installation”](#).

In case of the manual installation, an overview of booting the installation system is provided by the following procedure:

1. Prepare the installation media.

USB Flash Drive

In case of the manual installation from an ISO, creating a bootable flash disk is the easiest way to start the installation. You need to copy the ISO to the device using the `dd`. The flash disk must not be mounted, and all data on the device will be erased.

```
# dd if=PATH_TO_ISO_IMAGE of=USB_STORAGE_DEVICE bs=4M
```

If you are deploying raw images, you need to prepare the configuration device. For details refer to [Chapter 6, Deploying raw images](#).

Network booting

If the target computer's firmware supports it, you can boot the computer from the network and install from a server. This booting method requires a boot server that provides the needed boot images over the network. The exact protocol depends on your hardware. Commonly you need several services, such as TFTP and DHCP or PXE boot.

It is possible to install from many common network protocols, such as NFS, HTTP, FTP, or SMB. For more information on how to perform such an installation, refer to [Chapter 11, Remote installation](#).

2. Configure the target system firmware to boot the medium you chose. Refer to the documentation of your hardware vendor about how to configure the correct boot order.
3. Set the boot parameters required for your installation control method. An overview of the different methods is provided in [Section 3.3, "Controlling the installation"](#). A list of boot parameters is available in [Chapter 9, Boot parameters](#).
4. Perform the installation as described in [Chapter 10, Installation steps](#). The system needs to restart after the installation is finished.
5. Optional: Change the boot order of the system to directly boot from the medium to which SUSE Linux Enterprise Micro has been installed. If the system boots from the installation medium, the first boot parameter will be to boot the installed system.

For the deployment of raw images, the procedure is the following:

1. Prepare the raw image, for details refer to [Procedure 6.1, "Preparing the raw disk image"](#).
2. Prepare the configuration media, for details refer to [Procedure 6.2, "Preparing the configuration device."](#)
3. Configure the target system firmware to boot the medium where you copied the raw image. Refer to the documentation of your hardware vendor about how to configure the correct boot order.
4. Perform the installation as described in [Chapter 6, Deploying raw images](#). The system needs to restart after the installation is finished.

3.5 Dealing with boot and installation problems

Although SUSE® Linux Enterprise Micro undergoes an extensive test program, problems may occasionally occur during boot or installation.

3.5.1 Problems booting

Boot problems may prevent the YaST installer from starting on your system. Another symptom is failure to boot after the installation has been completed.

Installed system boots, not media

Change your computer's firmware so that the boot sequence is correct. To do this, consult the manual for your hardware.

The computer hangs

Change the console on your computer so that the kernel outputs are visible. Be sure to check the last few lines of output. This is normally done by pressing **Ctrl – Alt – F10**. If you cannot resolve the problem, consult the SUSE Linux Enterprise Micro support staff. To log all system messages at boot time, use a serial connection as described in [Section 2.3, "Controlling the installation"](#).

Boot disk

The boot disk is a useful interim solution for boot issues. If you have difficulties setting the other configurations, or if you want to postpone the decision regarding the final boot mechanism, use a boot disk.

3.5.2 Problems installing

If an unexpected problem occurs during installation, information is needed to determine the cause of the problem. Use the following directions to help with troubleshooting:

- Check the outputs on the various consoles. You can switch consoles with the key combination `Ctrl – Alt – Fn` . For example, obtain a shell in which to execute various commands by pressing `Ctrl – Alt – F2` .
- Try launching the installation with “Safe Settings” (press `F5` on the installation screen and choose *Safe Settings*). If the installation works without problems in this case, there is an incompatibility that causes either ACPI or APIC to fail. In some cases, a firmware update fixes this problem.
- Check the system messages on a console in the installation system by entering the command `dmesg -T` .

3.5.3 Redirecting the boot source to the boot DVD

To simplify the installation process and avoid accidental installations, the default setting on the installation DVD for SUSE Linux Enterprise Micro is that your system is booted from the first hard disk. At this point, an installed boot loader normally takes over control of the system. This means that the boot DVD can stay in the drive during an installation. To start the installation, choose one of the installation possibilities in the boot menu of the media.

3.6 Raspberry Pi

SUSE® Linux Enterprise Server is the first enterprise Linux distribution to support the inexpensive Raspberry Pi* single-board computer. SUSE Linux Enterprise Micro 5.1 supports the following models:

- Raspberry Pi 3 Model A +
- Raspberry Pi 3 Model B
- Raspberry Pi 3 Model B +
- Raspberry Pi 4 Model B
- Raspberry Pi Compute Module 3

- Raspberry Pi Compute Module 3 +
- Raspberry Pi Compute Module 4

The Raspberry Pi differs from more conventional server machines in several ways. First and foremost, it does not come with a boot loader capable of loading operating systems. SUSE Linux Enterprise Micro therefore ships additional boot loader software to fill that gap.

3.6.1 Boot process

The primary processor on the Raspberry Pi's System-on-Chip (SoC) is the Broadcom VideoCore Graphics Processing Unit (GPU), not the Arm Central Processing Unit (CPU). It is the GPU which starts initializing the hardware from a first-stage boot loader in the on-chip Boot Read-Only Memory (Boot ROM). Only a few configuration options can influence the Boot ROM; see [Section 3.6.1.2, "OTP memory"](#).

The Raspberry Pi 3 hardware does not have any built-in firmware. Instead, its second-stage boot loader firmware `bootcode.bin` is loaded from the boot medium every time the machine is powered on. It in turn loads the third-stage boot loader `start.elf`.

The Raspberry Pi 4 hardware has a small Electrically Erasable Programmable Read-Only Memory (EEPROM) for the second-stage boot loader. Apart from that, its boot sequence is similar to that of the Raspberry Pi 3, loading the third-stage boot loader `start4.elf` from the boot medium.



Warning: EEPROM update on Raspberry Pi 4

An update of the second-stage boot loader can be performed by booting from a specially prepared microSD card.

Only insert boot media that you trust, and verify that no file called `recovery.bin` is unintentionally present.

If an `armstub8.bin` file is present, it will be loaded as a fourth-stage boot loader at AArch64 Exception Level 3 (EL3). Otherwise, a minimal integrated stub will be used.



Note: EL3 security considerations

Code loaded for EL3 (often called BL31) will reside in memory, and Linux may attempt hypercalls into EL3 throughout its runtime.

Verify that your boot media have no `armstub8.bin` file unintentionally present. SUSE Linux Enterprise Micro 5.1 does not include it.

Beware that the Raspberry Pi's SoC does not provide TrustZone secure memory. Both the OS on the CPU and any software on the GPU may access its RAM. It is therefore unsuited for cryptographic EL0-s applications. SUSE Linux Enterprise Micro does not provide an EL1-s Trusted Execution Environment (TEE) for that reason.

SUSE Linux Enterprise Micro for the Raspberry Pi is configured to load a fifth-stage boot loader called `Das U-Boot`.

3.6.1.1 `Config.txt`

There is no non-volatile memory to hold configuration information. This means there are no conventional settings to adjust for boot device order, time and date, and so on.

Instead, the boot loader reads a configuration file `config.txt` from the boot medium. The `config.txt` provided by SUSE should not be modified. It allows the user to optionally provide an `extraconfig.txt` file, which can override any setting from `config.txt` if needed. This permits SUSE Linux Enterprise Micro to update the `config.txt` file when needed, without overwriting any user settings.

3.6.1.2 `OTP memory`

The SoC also has a very small amount of One-Time Programmable Memory (OTP memory). This can be used to configure some settings, such as whether the Boot ROM should attempt to boot from USB devices or over Ethernet.

This OTP memory is described on the Raspberry Pi Foundation Web site: <https://www.raspberrypi.org/documentation/hardware/raspberrypi/otpbits.md>



Warning: One-time programmable only

Configuration settings written into OTP memory cannot be reversed.

The most common use case for OTP memory will be enabling USB boot on Raspberry Pi 3 Model B or Compute Module 3.

3.6.1.3 Enabling USB boot mode for Raspberry Pi 3 Model B

To permanently allow booting from connected USB mass storage devices on Raspberry Pi 3 Model B, and from its on-board USB Ethernet, prepare a microSD card as described in [Section 3.6.3, “Deploying an appliance image”](#). Before unmounting or ejecting the card and booting from it, add to its FAT partition a text file `extraconfig.txt` ([Section 3.6.1.1, “Config.txt”](#)) with the following setting:

```
program_usb_boot_mode=1
```

Then continue to boot from the modified microSD card as usual. Once you see output from the U-Boot or GRUB boot loaders or the Linux kernel, you can remove power and then the microSD card. Your device should now be able to boot from USB ([Section 3.6.4, “Installation from USB media”](#)).

Note that once USB boot mode has been enabled for Raspberry Pi 3 Model B, USB boot mode cannot be disabled again ([Section 3.6.1.2, “OTP memory”](#)).

For more details, refer to the Raspberry Pi Foundation Web site: <https://www.raspberrypi.org/documentation/hardware/raspberrypi/bootmodes/msd.md> 

For the Raspberry Pi Compute Module 3, the setting required is the same, but the deployment of the modified image is a little more complicated.

3.6.2 Lack of a real-time clock

There is no battery-backed Real-Time Clock (RTC) on the Raspberry Pi itself.



Note: Time synchronization

The lack of a Real-Time Clock means that Raspberry Pi devices need to be configured to fetch the time from a network server by Network Time Protocol.

However, base boards for the Raspberry Pi Compute Modules may feature an RTC.

It is also possible to connect an RTC via the GPIO connector, using Hardware Attached on Top (HATs) or other expansion boards.

Either way, check whether the respective RTC chipset is supported by SUSE Linux Enterprise Micro. The connected RTC will need to be described to the operating system via a Device Tree Overlay ([Section 3.6.1.1, “Config.txt”](#)). For example, the MyPi base board might use:

```
dtparam=i2c1=on
```

3.6.3 Deploying an appliance image

The most common method to deploy an operating system onto Raspberry Pi hardware is to copy a pre-installed system image onto a boot medium, usually a microSD card. This is the simplest and easiest method.

SUSE provides a preconfigured bootable image of SUSE Linux Enterprise Micro for Raspberry Pi hardware. This comes with the Btrfs file system, with compression enabled to improve performance and reduce wear on microSD media.

A microSD card with a minimum size of 8 GB is recommended. Faster cards will give better system performance. On the first boot, the operating system automatically expands the file system to fill the card. This means that the first boot will be substantially slower than subsequent boots. The process of writing the card image onto microSD media is described in the [Raspberry Pi Quick Start \(https://documentation.suse.com/sles-15/html/SLES-rpi-quick/art-rpiquick.html\)](https://documentation.suse.com/sles-15/html/SLES-rpi-quick/art-rpiquick.html).

3.6.4 Installation from USB media

Some models of Raspberry Pi allow booting from USB mass storage devices. This will then allow deploying SUSE Linux Enterprise Micro on Raspberry Pi similar to server platforms.

Installation can be performed from a removable USB medium, such as a memory stick, onto a microSD card in the machine's internal slot. Alternatively, it can be performed from a removable USB medium onto another USB medium, such as a USB-connected hard disk.



Note: USB bandwidth limitations

Note that the Ethernet controller on the Raspberry Pi 3 is connected to the device's on-board USB 2.0 bus.

Therefore an operating system running from a disk attached via USB must share the total 480 Mbps bandwidth of the USB 2.0 controller. This will limit performance, and could significantly impact network performance.

This limitation does not apply to the Raspberry Pi 4.

Newer models of Raspberry Pi 3 with BCM2837 B0 silicon (silver instead of black chip), including Raspberry Pi 3 Model B+ and Compute Module 3+, allow booting from USB-connected storage devices by default.

On older models, such as Raspberry Pi 3 Model B or Compute Module 3, USB boot can be enabled by booting from a specially prepared microSD card once. See [Section 3.6.1.2, “OTP memory”](#) for instructions.

3.6.5 Installation from network

Because of the hardware's lack of on-board firmware ([Section 3.6.1, “Boot process”](#)), network-booting the Raspberry Pi using PXE is more complex than with more conventional computers.

The process of setting up a PXE boot server for x86 and Arm is described in the SUSE Best Practices document [How to Set Up a Multi-PXE Installation Server](https://documentation.suse.com/sbp/all/html/SBP-Multi-PXE-Install/index.html) (<https://documentation.suse.com/sbp/all/html/SBP-Multi-PXE-Install/index.html>).

The Raspberry Pi Foundation publishes information on how to boot using PXE one Raspberry Pi from another Raspberry Pi: https://www.raspberrypi.org/documentation/hardware/raspberrypi/bootmodes/net_tutorial.md.

3.6.6 More information

For more information, consult the following resources:

SUSE Linux Enterprise Server 15 SP3 Release Notes

For more information about hardware compatibility, supported options and functionality when running on Raspberry Pi hardware, consult the *Boot and Driver Enablement for Raspberry Pi* section of the SUSE Linux Enterprise Server Release Notes:

<https://www.suse.com/releasenotes/aarch64/SUSE-SLES/15-SP3/#aarch64-rpi>

Raspberry Pi Quick Start


<https://documentation.suse.com/sles-15/html/SLES-rpi-quick/art-rpiquick.html>

openSUSE Hardware Compatibility List: Raspberry Pi 3

The openSUSE project also has information about installing and configuring Raspberry Pi hardware. Much of this also applies to SUSE Linux Enterprise.

See https://en.opensuse.org/HCL:Raspberry_Pi3.

Das U-Boot

More information about Das U-Boot boot loader can be found on the project's GitHub page at <https://github.com/u-boot/u-boot> .

4 Installation on IBM Z and LinuxONE

This chapter describes the procedure for preparing the installation of SUSE® Linux Enterprise Micro on IBM Z. It provides all information needed to prepare the installation on the LPAR and z/VM side.

4.1 System requirements

This section provides basic information about the system requirements, level of MicroCode, and software. It also covers the different installation types and how to do an IPL for the first installation. For detailed technical information about IBM Z on SUSE Linux Enterprise Micro, refer to <https://developer.ibm.com/technologies/linux/>.

4.1.1 Hardware

SUSE Linux Enterprise Micro runs on the following platforms:

- IBM zEnterprise EC12 (zEC12) (2827)
- IBM zEnterprise BC12 (zBC12) (2828)
- IBM z13 (2964)
- IBM z13s (2965)
- IBM z Systems z16 A01 (3931)
- IBM LinuxONE Emperor (2964)
- IBM LinuxONE Rockhopper (2965)
- IBM z14 (3906)
- IBM z14 ZR1 (3907)
- IBM z Systems z15 (8561)
- IBM LinuxONE Emperor II (3906)

- IBM LinuxONE Rockhopper II (3907)
- IBM LinuxONE Rockhopper III (8561)
- IBM LinuxONE Emperor 4 (3931)

4.1.1.1 Memory requirements

Different installation methods have different memory requirements during installation. At least 1 GB of memory is recommended for the text-mode installation under z/VM, LPAR, and KVM. Installation in the graphical mode requires at least 1.5 GB of memory.



Note: Memory requirements with remote installation sources

A minimum of 512 MB of memory is required for installation from NFS, FTP, and SMB installation sources, or when VNC is used. Keep in mind that memory requirements also depend on the number of devices visible to the z/VM guest or the LPAR image. Installation with many accessible devices (even if unused for the installation) may require more memory.

4.1.1.2 Disk space requirements

The disk requirements depend largely on the workload of your containers. Minimal requirements for SLE Micro are 12 GB. The recommended value is 20 GB of hard disk space.

4.1.1.3 Network connection

A network connection is needed to communicate with your SUSE Linux Enterprise Micro system. This can be one or several of the following connections or network cards:

- OSA Express Ethernet (including Fast and Gigabit Ethernet)
- HiperSockets or Guest LAN
- 10 GBE, VSWITCH
- RoCE (RDMA over Converged Ethernet)

The following interfaces are still included, but no longer supported:

- CTC (or virtual CTC)
- ESCON
- IP network interface for IUCV

For installations under KVM, make sure the following requirements are met to enable the VM Guest to access the network transparently:

- The virtual network interface is connected to a host network interface.
- The host network interface is connected to a network that the virtual server will join.
- If the host is configured to have a redundant network connection by grouping two independent OSA network ports into a bonded network interface, the identifier for the bonded network interface is `bond0`. If more than one bonded interface exists, it is `bond1`, `bond2`, etc.
- A non-redundant network connection setup requires the identifier of the single network interface. The identifier has the following format: `enccw0.0.NNNN`, where `NNNN` is the device number of the desired network interface.

4.1.2 MicroCode Level, APARs, and fixes

Documentation about restrictions and requirements for this release of SUSE Linux Enterprise Server be found on IBM developerWorks at <https://developer.ibm.com/technologies/linux/>. We recommend to use the highest service level available. Contact IBM support for minimum requirements.

For z/VM, the following versions are supported:

- z/VM 6.4
- z/VM 7.1
- z/VM 7.2
- z/VM 7.3

Since it might be necessary to activate the VM APARs before installing the new MicroCode levels, clarify the order of installation with IBM support.

4.1.3 Software

When installing SUSE Linux Enterprise Micro via non-Linux–based NFS or FTP, you might experience problems with NFS or FTP server software. The Windows* standard FTP server can cause errors, so we recommend performing installation via SMB on these machines.

To connect to the SUSE Linux Enterprise Micro installation system, one of the following methods is required (SSH or VNC are recommended):

SSH with terminal emulation (xterm compatible)

SSH is a standard Unix tool that is present on most Unix or Linux systems. For Windows, you can use the Putty SSH client.

VNC client

For Linux, the `vncviewer` VNC client is included in SUSE Linux Enterprise Micro as part of the `tightvnc` package. For Windows, TightVNC is also available. Download it from <https://www.tightvnc.com/>.

X server

Find a suitable X server implementation on any Linux or Unix workstation. There are many commercial X Window System environments for Windows and macOS*. Some can be downloaded as free trial versions.



Tip: More information

Before installing SUSE Linux Enterprise Micro on IBM Z, consult the `README` file located in the root directory of the first installation medium of SUSE Linux Enterprise Micro.

4.2 Preparing for installation

This chapter explains how to make the data accessible for installation, install SUSE Linux Enterprise Micro using different methods, and prepare and use the IPL of the SUSE Linux Enterprise Micro installation system. The chapter also provides information about network configuration and network installation.

4.2.1 Making the installation data available

This section provides detailed information about making the SUSE Linux Enterprise Micro IBM Z installation data accessible for installation. Depending on your computer and system environment, choose between NFS or FTP installation. If you are running Microsoft Windows workstations in your environment, you can use the Windows network (including the SMB protocol) to install SUSE Linux Enterprise Micro on your IBM Z system.



Tip: IPL from DVD

It is possible to IPL from DVD and use the DVD as the installation medium. This is very convenient if you have restrictions setting up an installation server providing installation media over your network. The prerequisite is an FCP-attached SCSI DVD Drive.



Note: No installation from hard disk

It is not possible to perform installation from a hard disk by putting the content of the DVD to a partition on a DASD.

4.2.1.1 Using a Linux workstation or SUSE Linux Enterprise Micro DVD

You can use a Linux workstation in your computer environment to provide the installation data to the IBM Z installation process by NFS or FTP.



Important: Exporting mounted devices with NFS

Exporting the file system root (`/`) does not automatically export the mounted devices, such as DVD. Therefore, you need to explicitly name the mount point in `/etc/exports`:

```
/media/dvd *(ro)
```

After changing this file, restart the NFS server with the command `sudo systemctl restart nfsserver`.

Setting up an FTP server on a Linux system involves the installation and configuration of server software like `vsftpd`. Downloading the installation data via anonymous login is not supported, therefore you need to configure the FTP server to support user authentication.

4.2.1.1.1 SUSE Linux Enterprise Micro on DVD

The first installation medium of the SUSE Linux Enterprise Micro for IBM Z contains a bootable Linux image for Intel-based workstations and an image for IBM Z.

For Intel-based workstations, boot from this medium. When prompted, choose the desired answer language and keyboard layout and select *Start rescue system*. You need at least 64 MB RAM for this. No disk space is needed, because the entire rescue system resides in the workstation's RAM. This approach requires setting up the networking of the workstation manually.

For IBM Z, IPL your LPAR/VM guest from this medium as described in [Section 4.2.4.1.2, "IPL from FCP-attached SCSI DVD"](#). After entering your network parameters, the installation system treats the medium as the source of installation data. Because IBM Z cannot have an X11-capable terminal attached directly, choose between VNC or SSH installation. Refer to [Section 11.3, "Monitoring installation via VNC"](#) or [Section 11.4, "Monitoring installation via SSH"](#) for more information. SSH also provides a graphical installation by tunneling the X connection through SSH with `ssh -X`.

4.2.1.2 Using a Microsoft Windows workstation

You can use a Microsoft Windows workstation on your network to make the installation media available. The easiest way to do this is to use the SMB protocol. Make sure to activate *SMB over TCP/IP* as this enables the encapsulation of SMB packages into TCP/IP packages. Find details in the Windows online help or other Windows-related documentation that covers networking.

4.2.1.2.1 Using SMB

To make the installation media available with SMB, insert the USB flash drive with SLE-15-SP3-Online-ARCH-GM-media1.iso into the USB port of the Windows workstation. Then create a new share using the USB flash drive's letter and make it available for everyone in the network.

The installation path in YaST can be:

```
smb://DOMAIN;USER:PW@SERVERNAME/SHAREPATH
```

Where the placeholders mean:

DOMAIN

Optional workgroup or active directory domain.

USER,

PW

Optional user name and password of a user who can access this server and its share.

SERVERNAME

The name of the server that hosts the share(s).

SHAREPATH

The path to the share(s).

4.2.1.2.2 With NFS

Refer to the documentation provided with the third party product that enables NFS server services for your Windows workstation. The USB flash drive containing the SLE-15-SP3-Online-ARCH-GM-media1.iso medium must be in the available NFS path.

4.2.1.2.3 Using FTP

Refer to the documentation provided with the third-party product that is enabling FTP server services on your Windows workstation. The USB flash drive containing the SLE-15-SP3-Online-ARCH-GM-media1.iso medium must be in the available FTP path.

The FTP server that is bundled with certain Microsoft Windows releases implements only a subset of the FTP commands, and it is not suitable for providing the installation data. In this case, use a third-party FTP server that offers the required functionality.

4.2.1.2.4 Using an FCP-attached SCSI DVD drive

After you IPLed from the SCSI DVD as described in [Section 4.2.4.1.2, "IPL from FCP-attached SCSI DVD"](#), the installation system uses the DVD as the installation medium. In this case, you do not need the installation media on an FTP, NFS, or SMB server. However, you need the network configuration data for your SUSE Linux Enterprise Micro, because you must set up the network during the installation to perform a graphical installation via VNC or by X.

4.2.1.3 Using a Cobbler server for zPXE

IPLing from the network requires a Cobbler server to provide the kernel, initrd, and the installation data. Preparing the Cobbler server requires the following steps:

- [Section 4.2.1.3.1](#)
- [Section 4.2.1.3.2](#)
- [Section 4.2.1.3.3](#)
- [Section 4.2.1.3.4](#)

4.2.1.3.1 Importing the installation data

Importing the media requires the installation source to be available on the Cobbler server—either from USB flash drive or from a network source. Run the following command to import the data:

```
> sudo cobbler import --path=PATH ❶ --name=IDENTIFIER ❷ --arch=s390x
```

- ❶ Mount point of the installation data.
- ❷ A string identifying the imported product, for example “sles15_s390x”. This string is used as the name for the subdirectory where the installation data is copied to. On a Cobbler server running on SUSE Linux Enterprise this is `/srv/www/cobbler/ks_mirror/IDENTIFIER`. This path may be different if Cobbler runs on another operating system.

4.2.1.3.2 Adding a distribution

Adding a distribution allows Cobbler to provide the kernel and the initrd required to IPL via zPXE. Run the following command on the Cobbler server to add SUSE Linux Enterprise Micro for IBM Z:

```
> sudo cobbler distro add --arch=s390 --breed=suse --name="IDENTIFIER" ❶ \  
--os-version=sles15 ❷ \  
--initrd=/srv/www/cobbler/ks_mirror/IDENTIFIER/boot/s390x/initrd ❸ \  
--kernel=/srv/www/cobbler/ks_mirror/IDENTIFIER/boot/s390x/linux ❹ \  
--kopts="install=http://cobbler.example.com/cobbler/ks_mirror/IDENTIFIER" ❺
```

- ❶ Unique identifier for the distribution, for example “SLES 5.1 IBM Z”.

- ② Operating system identifier. Use `sles15`.
- ③ Path to the `initrd`. The first part of the path (`/srv/www/cobbler/ks_mirror/IDENTIFIER/`) depends on the location of the imported data and the subdirectory name you chose when importing the installation data.
- ④ Path to the kernel. The first part of the path (`/srv/www/cobbler/ks_mirror/IDENTIFIER/`) depends on the location of the imported data and the subdirectory name you chose when importing the installation data.
- ⑤ URL to the installation directory on the Cobbler server.

4.2.1.3.3 Adjusting the profile

Adding a distribution (see [Section 4.2.1.3.2, “Adding a distribution”](#)) automatically generates a profile with the corresponding `IDENTIFIER`. Use the following command to make a few required adjustments:

```
> sudo cobbler distro edit \  
--name=IDENTIFIER ① --os-version=sles10 ② --ksmeta="" ③ \  
--kopts="install=http://cobbler.example.com/cobbler/ks_mirror/IDENTIFIER" ④
```

- ① Identifier for the profile. Use the string specified when added the distribution.
- ② Operating system version. Distribution to which the profile should apply. Use the string specified with `--name=IDENTIFIER` in the importing step.
- ③ Option required for templating Kickstart files. Since it is not used for SUSE, leave it empty.
- ④ Space-separated list of kernel parameters. It must include at least the `install` parameter.

4.2.1.3.4 Adding systems

The last step is to add systems to the Cobbler server. This step must be performed for every IBM Z guest that should boot via zPXE. Guests are identified by their z/VM user ID (in the following example, the ID “linux01”). Note that the ID must be lowercase. To add a system, run the following command:

```
> sudo cobbler system add --name=linux01 --hostname=linux01.example.com \  
--profile=IDENTIFIER --interface=qdio \  
--ip-address=192.168.2.103 --subnet=192.168.2.255 --netmask=255.255.255.0 \  
--name-servers=192.168.1.116 --name-servers-search=example.com \  
--gateway=192.168.2.1 --kopts="KERNEL_OPTIONS"
```

The `--kopts` option allows you to specify the kernel and installation parameters that are usually specified in the `parmfile`. Specify the parameters using the following format: `PARAMETER1=VAL - UE1 PARAMETER2=VALUE2`. The installer prompts for missing parameters. For a fully-automated installation, you need to specify all parameters for networking, DASDs and provide an AutoYaST file. Below is an example for a guest equipped with an OSA interface using the same network parameters as above.

```
--kopts=" \  
AutoYaST=http://192.168.0.5/autoinst.xml \  
Hostname=linux01.example.com \  
Domain=example.com \  
HostIP=192.168.2.103 \  
Gateway=192.168.2.1 \  
Nameserver=192.168.1.116 \  
Searchdns=example.com \  
InstNetDev=osa; \  
Netmask=255.255.255.0 \  
Broadcast=192.168.2.255 \  
OsaInterface=qdio \  
Layer2=0 \  
PortNo=0 \  
ReadChannel=0.0.0700 \  
WriteChannel=0.0.0701 \  
DataChannel=0.0.0702 \  
DASD=600"
```

4.2.1.4 Installing from a USB Flash Drive of the HMC

Installation of SUSE Linux Enterprise Micro on IBM Z servers usually requires a network installation source. If this requirement cannot be fulfilled, SUSE Linux Enterprise Server allows you to use the USB flash drive of the Hardware Management Console (HMC) as an installation source for installation on an LPAR.

To perform installation from the USB flash drive of the HMC, proceed as follows:

- Add `install=hmc:/` to the `parmfile` (see [Section 4.4, “The `parmfile`—automating the system configuration](#)”) or kernel options.
- In the manual-mode installation using `linuxrc`, choose *Start Installation*, then *Installation*, and then *Hardware Management Console*. The installation medium must be in the HMC.

Important: Configure network

Before starting the installation, specify a network configuration in `linuxrc`. You cannot do this via boot parameters, and it is very likely that you will need network access. In `linuxrc`, go to *Start Installation*, then choose *Network Setup*.

Important: Linux system must boot first

Before granting access to the media in the USB flash drive of the HMC, wait until the Linux system is booted. IPLing can disrupt the connection between the HMC and the LPAR. If the first attempt to use the described method fails, you can grant the access and retry the option `HMC`.

Note: Installation repository

The USB flash drive is not kept as an installation repository, as the installation is a one-time procedure. If you need an installation repository, register and use the online repository.

4.2.2 Installation types

This section describes SUSE Linux Enterprise Micro installation steps for each installation mode. When the preparation steps described in the previous chapters have been completed, follow the overview of the desired installation mode.

As described in [Section 4.2.1, “Making the installation data available”](#), there are three different installation modes for Linux on IBM Z: LPAR, z/VM, and KVM guest installation.

PROCEDURE 4.1: OVERVIEW OF AN LPAR INSTALLATION

1. Prepare the devices needed for installation. See [Section 4.2.3.1, “Preparing the IPL of an LPAR installation”](#).
2. IPL the installation system. See [Section 4.2.4.1, “IPLing an LPAR installation”](#).
3. Configure the network. See [Section 4.2.5, “Network configuration”](#).
4. Connect to the SUSE Linux Enterprise Micro installation system. See [Section 4.2.6, “Connecting to the SUSE Linux Enterprise Micro installation system”](#).

5. Start the installation using YaST and IPL the installed system. See *Chapter 10, Installation steps*.

PROCEDURE 4.2: INSTALLATION OVERVIEW OF Z/VM INSTALLATION

1. Prepare the devices needed for installation. See *Section 4.2.3.2.1, "Adding a Linux guest using dirMaint"*.
2. IPL the installation system. See *Section 4.2.4.2, "IPLing a z/VM installation"*.
3. Configure the network. See *Section 4.2.5, "Network configuration"*.
4. Connect to the SUSE Linux Enterprise Micro installation system. See *Section 4.2.6, "Connecting to the SUSE Linux Enterprise Micro installation system"*.
5. Start the installation using YaST and IPL the installed system. See *Chapter 10, Installation steps*.

PROCEDURE 4.3: OVERVIEW OF A KVM GUEST INSTALLATION

1. Create a virtual disk image and write a domain XML file. See *Section 4.2.3.3, "Preparing the IPL of a KVM guest installation"*.
2. Prepare the installation target and IPL the VM Guest. See *Section 4.2.4.3, "IPLing a KVM guest installation"*.
3. *Section 4.2.5.3, "Set up the network and select the installation source"*.
4. Connect to the SUSE Linux Enterprise Micro installation system. See *Section 4.2.6, "Connecting to the SUSE Linux Enterprise Micro installation system"*.
5. Start the installation using YaST and IPL the installed system. See *Chapter 10, Installation steps*.

4.2.3 Preparing the IPL of the SUSE Linux Enterprise Micro installation system

4.2.3.1 Preparing the IPL of an LPAR installation

Configure your IBM Z system to start in ESA/S390 or Linux-only mode with an appropriate activation profile and IOCDs. For further information, refer to the IBM documentation. Continue as described in *Section 4.2.4.1, "IPLing an LPAR installation"*.

4.2.3.2 Preparing the IPL of a z/VM installation

4.2.3.2.1 Adding a Linux guest using dirMaint

The first step is to attach and format one or multiple DASDs in the system to be used by the Linux guest in z/VM. Next, create a new user in z/VM. The example shows the directory for a user `LINUX1` with the password `LINPWD`, 1 GB of memory (extendable up to 2 GB), several minidisks (MDISK), two CPUs, and an OSA QDIO device.



Tip: Assigning memory to z/VM guests

When assigning memory to a z/VM guest, make sure that the memory size is adequate for the preferred installation type. To set the memory size to 1 GB, use the command `CP DEFINE STORAGE 1G`. After the installation has finished, reset the memory size to the desired value.

EXAMPLE 4.1: CONFIGURATION OF A Z/VM DIRECTORY

```
USER LINUX1 LINPWD 1024M 2048M G
*
* _____
* LINUX1
* _____
* This VM Linux guest has two CPUs defined.

CPU 01 CPUID 111111
CPU 02 CPUID 111222
IPL CMS PARM AUTOCR
IUCV ANY
IUCV ALLOW
MACH ESA 10
OPTION MAINTCCW RMCHINFO
SHARE RELATIVE 2000
CONSOLE 01C0 3270 A
SPOOL 000C 2540 READER *
SPOOL 000D 2540 PUNCH A
SPOOL 000E 3203 A
* OSA QDIO DEVICE DEFINITIONS
DEDICATE 9A0 9A0
DEDICATE 9A1 9A1
DEDICATE 9A2 9A2
*
LINK MAINT 0190 0190 RR
```

```
LINK MAINT 019E 019E RR
LINK MAINT 019D 019D RR
* MINIDISK DEFINITIONS
MDISK 201 3390 0001 0050 DASD40 MR ONE4ME TW04ME THR4ME
MDISK 150 3390 0052 0200 DASD40 MR ONE4ME TW04ME THR4ME
MDISK 151 3390 0253 2800 DASD40 MR ONE4ME TW04ME THR4ME
```

This example uses minidisk 201 as the guest's home disk. Minidisk 150 with 200 cylinders is the Linux swap device. Disk 151 with 2800 cylinders holds the Linux installation.

As user `MAINT`, add the guest to the user directory with **DIRM FOR LINUX1 ADD**. Enter the name of the guest (`LINUX1`) and press **F5**. Set up the environment of the user with:

```
DIRM DIRECT
DIRM USER WITHPASS
```

The last command returns a reader file number. This number is needed for the next command:

```
RECEIVE <number> USER DIRECT A (REPL)
```

You can now log in on the guest as user `LINUX1`.

If you do not have the `dirmaint` option available, refer to the IBM documentation on how to set up this user.

Proceed with [Section 4.2.4.2, "IPLing a z/VM installation"](#).

4.2.3.3 Preparing the IPL of a KVM guest installation

A KVM guest installation requires a domain XML file that specifies the virtual machine and at least one virtual disk image for the installation.

4.2.3.3.1 Create a virtual disk image

By default, libvirt searches for disk images in `/var/lib/libvirt/images/` on the VM Host Server. Although images can also be stored anywhere on the file system, it is recommended to store all images in a single location for easier maintainability. To create an image, log in to the KVM host server and run the following command:

```
qemu-img create -f qcow2 /var/lib/libvirt/images/s12lin_qcow2.img 10G
```

This creates a qcow2 image with a size of 10 GB in `/var/lib/libvirt/images/`.

4.2.3.3.2 Write a domain XML file

A domain XML file is used to define the VM Guest. To create the domain XML file, open an empty file `s15-1.xml` with an editor and create a file like in the following example.

EXAMPLE 4.2: EXAMPLE DOMAIN XML FILE

The following example creates a VM Guest with a single CPU, 1 GB RAM, and the virtual disk image created in the previous section ([Section 4.2.3.3.1, "Create a virtual disk image"](#)). It assumes that the virtual server is attached to the host network interface `bond0`. Change the source devices element to match your network setup.

```
<domain type="kvm">
  <name>s15-1</name>
  <description>Guest-System SUSE SLES15</description>
  <memory>1048576</memory>
  <vcpu>1</vcpu>
  <os>
    <type arch="s390x" machine="s390-ccw-virtio">hvm</type>
    <!-- Boot kernel - remove 3 lines after successfull installation -->
    <kernel>/var/lib/libvirt/images/s15-kernel.boot</kernel>
    <initrd>/var/lib/libvirt/images/s15-initrd.boot</initrd>
    <cmdline>linuxrcstderr=/dev/console</cmdline>
  </os>
  <iothreads>1</iothreads>
  <on_poweroff>destroy</on_poweroff>
  <on_reboot>restart</on_reboot>
  <on_crash>preserve</on_crash>
  <devices>
    <emulator>/usr/bin/qemu-system-s390x</emulator>
    <disk type="file" device="disk">
      <driver name="qemu" type="qcow2" cache="none" iothread="1" io="native"/>
      <source file="/var/lib/libvirt/images/s15lin_qcow2.img"/>
      <target dev="vda" bus="virtio"/>
    </disk>
    <interface type="direct">
      <source dev="bond0" mode="bridge"/>
      <model type="virtio"/>
    </interface>
    <console type="pty">
      <target type="sclp"/>
    </console>
  </devices>
</domain>
```

4.2.4 IPLing the SUSE Linux Enterprise Micro installation system

4.2.4.1 IPLing an LPAR installation

There are different ways to IPL SUSE Linux Enterprise Micro into an LPAR. The preferred way is to use the *Load from CD-ROM or server* feature of the SE or HMC.

4.2.4.1.1 IPL from DVD-ROM

Mark the LPAR to install and select *Load from CD-ROM or server*. Leave the field for the file location blank or enter the path to the root directory of the first DVD-ROM and select *Continue*. Keep the default selection in the list of options that appears. *Operating system messages* should now show the kernel boot messages.

4.2.4.1.2 IPL from FCP-attached SCSI DVD

You can use the *Load* procedure by selecting *SCSI* as *Load type* to IPL from SCSI. Enter the WWPN (Worldwide port name) and LUN (Logical unit number) provided by your SCSI bridge or storage (16 digits—do not omit the trailing 0s). The boot program selector must be 2. Use your FCP adapter as *Load address* and perform an IPL.

4.2.4.2 IPLing a z/VM installation

This section describes IPLing the installation system to install SUSE Linux Enterprise Micro for IBM Z on a z/VM system.

4.2.4.2.1 IPL from the z/VM reader

You need a working TCP/IP connection and an FTP client program within your newly-defined z/VM guest to transfer the installation system via FTP. Setting up TCP/IP for z/VM is beyond the scope of this manual. Refer to the appropriate IBM documentation.

Log in as the z/VM Linux guest to IPL. Make the content of the directory `/boot/s390x` of the Unified Installer (medium 1) available via FTP within your network. From this directory, get the files `linux`, `initrd`, `parmfile`, and `sles.exec`. Transfer the files with a fixed block size

of 80 characters. Specify it with the FTP command `locsite fix 80`. It is important to copy `linux` (the Linux kernel) and `initrd` (the installation image) as binary files, so use the `binary` transfer mode. `parmfile` and `sles.exec` need to be transferred in ASCII mode.

The following example shows the required steps. This particular scenario assumes that the required files are accessible from an FTP server at the IP address `192.168.0.3` and the login is `lininst`.

EXAMPLE 4.3: TRANSFERRING THE BINARIES VIA FTP

```
FTP 192.168.0.3
VM TCP/IP FTP Level 530
Connecting to 192.168.0.3, port 21
220 ftpserver FTP server (Version wu-2.4.2-academ[BETA-18](1)
Thu Feb 11 16:09:02 GMT 2010) ready.
USER
lininst
331 Password required for lininst
PASS
*****
230 User lininst logged in.
Command:
binary
200 Type set to I
Command:
locsite fix 80
Command:
get /media/dvd1/boot/s390x/linux sles.linux
200 PORT Command successful
150 Opening BINARY mode data connection for /media/dvd1/boot/s390x/linux
(10664192 bytes)
226 Transfer complete.
10664192 bytes transferred in 13.91 seconds.
Transfer rate 766.70 Kbytes/sec.
Command:
get /media/dvd1/boot/s390x/initrd sles.initrd
200 PORT Command successful
150 Opening BINARY mode data connection for /media/dvd1/boot/s390x/initrd
(21403276 bytes)
226 Transfer complete.
21403276 bytes transferred in 27.916 seconds.
Transfer rate 766.70 Kbytes/sec.
Command:
ascii
200 Type set to A
Command:
get /media/dvd1/boot/s390x/parmfile sles.parmfile
```

```

150 Opening ASCII mode data connection for /media/dvd1/boot/s390x/parmfile
(5 bytes)
226 Transfer complete.
5 bytes transferred in 0.092 seconds.
Transfer rate 0.05 Kbytes/sec.
Command:
get /media/dvd1/boot/s390x/sles.exec sles.exec
150 Opening ASCII mode data connection for /media/dvd1/boot/s390x/sles.exec
(891 bytes)
226 Transfer complete.
891 bytes transferred in 0.097 seconds.
Transfer rate 0.89 Kbytes/sec.
Command:
quit

```

Use the REXX script `sles.exec` you downloaded to IPL the Linux installation system. This script loads the kernel, `parmfile`, and the initial RAM disk into the reader for IPL.

EXAMPLE 4.4: SLES.EXEC

```

/* REXX LOAD EXEC FOR SUSE LINUX S/390 VM GUESTS      */
/* LOADS SUSE LINUX S/390 FILES INTO READER          */
SAY ''
SAY 'LOADING SLES FILES INTO READER...'
'CP CLOSE RDR'
'PURGE RDR ALL'
'SPOOL PUNCH * RDR'
'PUNCH SLES LINUX A (NOH'
'PUNCH SLES PARMFILE A (NOH'
'PUNCH SLES INITRD A (NOH'
'IPL 00C'

```

Using the script, you can IPL the SUSE Linux Enterprise Micro installation system with the command `sles`. The Linux kernel then starts and outputs its boot messages.

To continue the installation, proceed to [Section 4.2.5, “Network configuration”](#).

4.2.4.2.2 IPL from FCP-attached SCSI DVD

To IPL in z/VM, prepare the SCSI IPL process by using the SET LOADDEV parameter:

```
SET LOADDEV PORTNAME 200400E8 00D74E00 LUN 00020000 00000000 BOOT 2
```

After setting the LOADDEV parameter with the appropriate values, IPL your FCP adapter, for example:

```
IPL FC00
```

To continue the installation, proceed with [Section 4.2.5, “Network configuration”](#).

4.2.4.2.3 IPL from a Cobbler server with zPXE

To IPL from a Cobbler server with zPXE, you need to transfer the `zpxe.rexx` script via FTP from the Cobbler server to your z/VM guest. To do this, the z/VM guest needs a working TCP/IP connection and an FTP client program.

Log in as the z/VM Linux guest to IPL and transfer the script with a fixed size of 80 characters in ASCII mode (see [Example 4.3, “Transferring the binaries via FTP”](#) for an example). The `zpxe.rexx` script is available on the Unified Installer DVD at `/boot/s390x/zpxe.rexx` or on a SLE Cobbler server at `/usr/share/doc/packages/s390-tools/zpxe.rexx`.

`zpxe.rexx` is supposed to replace the `PROFILE EXEC` of your guest. Make a backup copy of the existing `PROFILE EXEC` and rename `ZPXE REXX` to `PROFILE EXEC`. Alternatively, call `ZPXE REXX` from the existing `PROFILE EXEC` by adding the `'ZPXE REXX'` line to it.

The last step is to create a configuration file, `ZPXE CONF` that instructs `ZPXE REXX` which Cobbler server to contact and which disk to IPL. Run `xedit zpxe conf a` and create `ZPXE CONF` with the following content (replace the example data accordingly):

```
HOST cobbler.example.com
IPLDISK 600
```

This connects the Cobbler server next time you log in to the z/VM guest. If an installation is scheduled on the Cobbler server, it will be executed. To schedule the installation, run the following command on the Cobbler server:

```
> sudo cobbler system edit --name ID ❶ --netboot-enabled 1 ❷ --profile PROFILENAME ❸
```

- ❶ z/VM user ID.
- ❷ Enable IPLing from the network.
- ❸ Name of an existing profile, see [Section 4.2.1.3.3, “Adjusting the profile”](#).

4.2.4.3 IPLing a KVM guest installation

To start the guest installation, you first need to start the VM Guest defined in [Section 4.2.3.3.1, “Create a virtual disk image”](#). Before you begin, ensure the kernel and `initrd` are available for IPL.

4.2.4.3.1 Preparing the installation source

Kernel and initrd of the installation system need to be copied to the VM Host Server to IPL the VM Guest into the installation system.

1. Log in to the KVM host and make sure you can connect to the remote host or device serving the installation source.
2. Copy the following two files from the installation source to `/var/lib/libvirt/images/`. If the data is served from a remote host, use `ftp`, `sftp`, or `scp` to transfer the files:

`/boot/s390x/initrd`

`/boot/s390x/cd.ikr`

3. Rename the files on the KVM host:

```
> sudo cd /var/lib/libvirt/images/  
> sudo mv initrd s15-initrd.boot  
> sudo mv cd.ikr s15-kernel.boot
```

4.2.4.3.2 IPL the VM Guest

To IPL the VM Guest, log in to the KVM host and run the following command:

```
> virsh create s15-1.xml --console
```

The installation process starts when the VM Guest is up and running, and you should see the following message:

```
Domain s15-1 started  
Connected to domain s15-1  
Escape character is ^]  
Initializing cgroup subsys cpuset  
Initializing cgroup subsys cpu  
Initializing  
cgroup subsys cpuacct  
. .  
Please make sure your installation medium is available.  
Retry?  
0) <-- Back <--  
1) Yes  
2) No
```


Answer 2) No and choose *Installation* on the next step. Proceed as described in [Section 4.2.5.3](#), “Set up the network and select the installation source”.

4.2.5 Network configuration

Wait until the kernel has completed its start-up routines. If you perform the installation in basic mode or in an LPAR, open the *Operating System Messages* on the HMC or SE.

First, choose *Start Installation* in the `linuxrc` main menu. Then choose *Start Installation or Update* to start the installation process. Select *Network* as the installation medium, then select the type of network protocol to use for the installation. [Section 4.2.1](#), “Making the installation data available” describes how to make the installation data available for the various types of network connections. Currently, *FTP*, *HTTP*, *NFS*, and *SMB/CIFS* (Windows file sharing) are supported.

From the list of available devices, choose an OSA or HiperSockets network device for receiving the installation data. Although the list may contain CTC, ESCON, or IUCV devices, they are no longer supported on SUSE Linux Enterprise Micro.

4.2.5.1 Configure a HiperSockets interface

Select a HiperSocket device from the list of network devices. Then enter values for the read, write, and data channels:

EXAMPLE 4.5: SUPPORTED NETWORK CONNECTION TYPES AND DRIVER PARAMETERS

```
Choose the network device.

1) IBM parallel CTC Adapter (0.0.0600)
2) IBM parallel CTC Adapter (0.0.0601)
3) IBM parallel CTC Adapter (0.0.0602)
4) IBM Hipersocket (0.0.0800)
5) IBM Hipersocket (0.0.0801)
6) IBM Hipersocket (0.0.0802)
7) IBM OSA Express Network card (0.0.0700)
8) IBM OSA Express Network card (0.0.0701)
9) IBM OSA Express Network card (0.0.0702)
10) IBM OSA Express Network card (0.0.f400)
11) IBM OSA Express Network card (0.0.f401)
12) IBM OSA Express Network card (0.0.f402)
13) IBM IUCV

> 4
```

```
Device address for read channel. (Enter '+++ ' to abort).
```

```
[0.0.0800]> 0.0.0800
```

```
Device address for write channel. (Enter '+++ ' to abort).
```

```
[0.0.0801]> 0.0.0801
```

```
Device address for data channel. (Enter '+++ ' to abort).
```

```
[0.0.0802]> 0.0.0802
```

4.2.5.2 Configure an OSA express device

Select an OSA Express device from the list of network devices and specify a port number. Enter the values for the read, write and data channels. Choose whether to enable OSI Layer 2 support.

The port number is required for the new 2 port OSA Express 3 Network devices. If you are not using an OSA Express 3 device, enter 0. OSA Express cards can also run in the “OSI layer 2 support” mode or the older more common “layer 3” mode. The card mode affects all systems that share the device, including systems on other LPARs. If in doubt, specify 2 for compatibility with the default mode used by other operating systems such as z/VM and z/OS. Consult with your hardware administrator for further information on these options.

EXAMPLE 4.6: NETWORK DEVICE DRIVER PARAMETERS

```
Choose the network device.
```

```
1) IBM parallel CTC Adapter (0.0.0600)
2) IBM parallel CTC Adapter (0.0.0601)
3) IBM parallel CTC Adapter (0.0.0602)
4) IBM Hipersocket (0.0.0800)
5) IBM Hipersocket (0.0.0801)
6) IBM Hipersocket (0.0.0802)
7) IBM OSA Express Network card (0.0.0700)
8) IBM OSA Express Network card (0.0.0701)
9) IBM OSA Express Network card (0.0.0702)
10) IBM OSA Express Network card (0.0.f400)
11) IBM OSA Express Network card (0.0.f401)
12) IBM OSA Express Network card (0.0.f402)
13) IBM IUCV
```

```
> 7
```

```
Enter the relative port number. (Enter '+++ ' to abort).
```

```
> 0
```

```

Device address for read channel. (Enter '+++' to abort).
[0.0.0700]> 0.0.0700

Device address for write channel. (Enter '+++' to abort).
[0.0.0701]> 0.0.0701

Device address for data channel. (Enter '+++' to abort).
[0.0.0702]> 0.0.0702

Enable OSI Layer 2 support?

0) <-- Back <--
1) Yes
2) No

> 1

MAC address. (Enter '+++' to abort).
> +++

```

4.2.5.3 Set up the network and select the installation source

After all network device parameters have been entered, the respective driver is installed and you see the corresponding kernel messages.

Next, you need to specify whether to use DHCP autoconfiguration for setting up the network interface parameters. Because DHCP only works on a few devices and requires special hardware configuration settings, choose *NO*. Doing this prompts you to specify the following networking parameters:

- The IP address of the system to install
- The corresponding netmask (if not having been specified with the IP address)
- The IP address of a gateway to reach the server
- A list of search domains covered by the domain name server (DNS)
- The IP address of your domain name server

EXAMPLE 4.7: NETWORKING PARAMETERS

```

Automatic configuration via DHCP?

0) <-- Back <--

```

```
1) Yes
2) No

> 2

Enter your IP address with network prefix.

You can enter more than one, separated by space, if necessary.
Leave empty for autoconfig.

Examples: 192.168.5.77/24 2001:db8:75:fff::3/64. (Enter '+++' to abort).
> 192.168.0.20/24

Enter your name server IP address.

You can enter more than one, separated by space, if necessary.
Leave empty if you don't need one.

Examples: 192.168.5.77 2001:db8:75:fff::3. (Enter '+++' to abort).
> 192.168.0.1

Enter your search domains, separated by a space:. (Enter '+++' to abort).
> example.com

Enter the IP address of your name server. Leave empty if you do not need one. (Enter '+++' to abort).
> 192.168.0.1
```

Finally, provide the required information about the installation server, such as the IP address, the directory containing the installation data, and login credentials. The installation system loads when the required information has been provided.

4.2.6 Connecting to the SUSE Linux Enterprise Micro installation system

After loading the installation system, `linuxrc` prompts you to choose what type of display to use to control the installation procedure. The available options include `X11` (X Window System), `VNC` (Virtual Network Computing protocol), `SSH` (text mode or X11 installation via Secure Shell), or `ASCII Console`. The recommended options are `VNC` or `SSH`.

If the `ASCII Console` option is selected, YaST starts in text mode, and you can perform the installation directly within your terminal. Using the `ASCII Console` is only useful when installing into LPAR.



Note: Terminal emulation for ASCII console

To be able to work with YaST in the text mode, it needs to run in a terminal with VT220/Linux emulation (also called ASCII console).

4.2.6.1 Initiating the installation for VNC

To remotely control an installation via VNC, follow these steps:

1. Selecting the VNC option starts the VNC server. A short note in the console displays the IP address and display number for connecting with vncviewer.
2. Enter the IP address and the display number of the SUSE Linux Enterprise Micro installation system when prompted to do so.
3. When prompted, enter the IP address and the display number of the SUSE Linux Enterprise Micro installation system.

```
http://<IP address of installation system>:5801/
```

4. After the connection has been established, install SUSE Linux Enterprise Micro with YaST.

4.2.6.2 Initiating the installation for the X Window system



Important: X authentication mechanism

The direct installation with the X Window System relies on an authentication mechanism based on host names. This mechanism is disabled in current SUSE Linux Enterprise Micro versions. We recommend performing the installation using SSH or VNC.

To remotely control an installation via X forwarding, follow these steps:

1. Make sure that the X server allows the client (the system that is installed) to connect. Set the variable **`DISPLAYMANAGER_XSERVER_TCP_PORT_6000_OPEN="yes"`** in the file `/etc/sysconfig/displaymanager`. Restart the X server and allow client binding to the server using **`xhost CLIENT_IP_ADDRESS`**.
2. When prompted at the installation system, enter the IP address of the machine running the X server.

3. Wait until YaST opens, then start the installation.

4.2.6.3 Initiating the installation for SSH

To connect to an installation system with the name `earth` via SSH, use the `ssh -X earth` command. If your workstation runs on Microsoft Windows, use the Putty tool available from <http://www.chiark.greenend.org.uk/~sgtatham/putty/>. Set *Enable X11 forwarding* in Putty under *Connection > SSH > X11*.

If you use another operating system, execute `ssh -X earth` to connect to an installation system with the name `earth`. X-Forwarding over SSH is supported if you have a local X server available. Otherwise, YaST provides a text interface over ncurses.

When prompted, enter the `root` user name and log in with your password. Enter `yast.ssh` to start YaST. YaST then guides you through the installation.

Important: Fixing YaST over SSH issue

In certain situations, running the GUI version of YaST over SSH with X forwarding may fail with the following error message:

```
XIO: fatal IO error 11 (Resource temporarily unavailable) on X server
"localhost:11.0"
```

In this case you have two options.

- Run YaST with the `QT_XCB_GL_INTEGRATION=none` option, for example:

```
QT_XCB_GL_INTEGRATION=none yast.ssh
QT_XCB_GL_INTEGRATION=none yast2 disk
```

- Run the ncurses version of YaST application by disabling X forwarding or by specifying ncurses as the desired UI. To do the latter, use the `yast2 disk --ncurses` or `YUI_PREFERRED_BACKEND=ncurses yast2 disk` command.

Proceed with the installation procedure as described in [Chapter 10, Installation steps](#).

4.2.7 The SUSE Linux Enterprise Micro boot procedure on IBM Z

On SLES 10 and 11 the boot process was handled by the `zipl` boot loader. To enable booting from Btrfs partitions and supporting system rollbacks with Snapper, the way SUSE Linux Enterprise Micro is booted on IBM Z has changed.

GRUB 2 replaces `zipl` on SUSE Linux Enterprise Micro for IBM Z. GRUB 2 on the AMD64/Intel 64 architecture includes device drivers on the firmware level to access the file system. On the mainframe there is no firmware and adding `ccw` to GRUB 2 would not only be a major undertaking, but would also require a reimplementaion of `zipl` in GRUB 2. Therefore SUSE Linux Enterprise Micro uses a two-stage approach:

Stage one:

A separate partition containing the kernel and an `initrd` is mounted to `/boot/zipl`. This kernel and the `initrd` are loaded via `zipl` using the configuration from `/boot/zipl/config`. This configuration adds the keyword `initgrub` to the kernel command line. When the kernel and `initrd` are loaded, the `initrd` activates the devices required to mount the root file system (see `/boot/zipl/active_devices.txt`). Afterward a GRUB 2 user space program is started, which reads `/boot/grub2/grub.cfg`.

Stage two:

The kernel and the `initrd` specified in `/boot/grub2/grub.cfg` are started via `kexec`. Devices listed in `/boot/zipl/active_devices.txt` that are necessary for starting the on-disk system are then activated. Other devices from that list will be whitelisted, but otherwise ignored. The root file system is mounted and the boot procedure continues like on the other architectures.

4.3 Secure boot

For the secure boot functionality to work on a IBM Z system, the following conditions must be met.

- The machine must be z15 T01, z15 T02, LinuxONE III LT1, LinuxONE III LT2, or a later model.
- You must use an LPAR (secure boot is not supported on z/VM and KVM).
- The LPAR must have secure boot enabled.
- You must use SCSI (FCP) disks (secure boot is not supported on DASD).



Note: Hardware migration

In case you migrate to a different machine (for example, from z13 to z15), ensure that the LPAR on the target machine has the secure boot state of the system on its disk.

Changing the secure boot state must be performed according to the following procedure.

PROCEDURE 4.4: CHANGING SECURE BOOT STATE

1. Enable secure boot in YaST and write the new boot loader.
2. Shut down the system.
3. Change the configuration of the LPAR (enable or disable secure boot).
4. Boot the system.



Note: Secure boot on HMC

The system on the disk configured with the `secure=1` parameter can be booted on z15 HMC as long as the firmware supports the new on-disk format (which is always the case on z15).

4.4 The parmfile—automating the system configuration

The installation process can be partially automated by specifying the essential parameters in the `parmfile`. The `parmfile` contains all the data required for network setup and DASD configuration. In addition to that, it can be used to set up the connection method to the SUSE Linux Enterprise Micro installation system and the YaST instance running there. This reduces user interaction to the actual YaST installation.

The parameters listed in [Section 4.4.1, “General parameters”](#) can be passed to the installation routine as the default values for installation. Note that all IP addresses, server names, and numerical values are examples. Replace them with the actual values of your installation scenario.

The number of lines in the parmfile is limited to 10. You can specify more than one parameter on a line. Parameter names are not case-sensitive. Parameters must be separated by spaces. You may specify the parameters in any order. Always keep the `PARAMETER=value` string together on one line. The length of each line must not exceed 80 characters. For example:

```
Hostname=s390zvm01.suse.de HostIP=10.11.134.65
```



Tip: Using IPv6 during the installation

By default, you can only assign IPv4 network addresses to your machine. To enable IPv6 during installation, specify one of the following parameters at the boot prompt: `ipv6=1` (accept IPv4 and IPv6) or `ipv6only=1` (accept IPv6 only).

Some of the following parameters are required. If they are missing, the automatic process prompts you to specify them.

4.4.1 General parameters

AutoYaST= <URL> Manual=0

The `AutoYaST` parameter specifies the location of the `autoinst.xml` control file for automatic installation. The `Manual` parameter controls if 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` defaults `Manual` to `0`.

DeviceAutoConfig=<0|1|2>

In `linuxrc`, the `DeviceAutoConfig` parameter controls the use of I/O device auto-configuration data for IBM Z systems.

If set to `0`, auto-configuration is disabled. If set to `1`, the existing auto-config data are applied. If set to `2` (the default), a dialog is shown if auto-config data are present. The user is asked whether to apply them.

For more details, see [Section 4.4.4, “I/O device auto-configuration on IBM Z systems”](#).

Info= <URL>

Specifies a location for a file with additional options. This helps to overcome the limitations of 10 lines (and 80 characters per line under z/VM) for the parmfile. Further documentation on the Info file can be found in *Book “AutoYaST Guide”, Chapter 9 “The auto-installation*

process”, Section 9.3.3 “Combining the `linuxrc` info file with the AutoYaST control file”. Since the Info file can typically only be accessed through the network on IBM Z, you cannot use it to specify the options required to set up the network (that is, the options described in Section 4.4.2, “Configuring the network interface”). Other `linuxrc`-specific options, such as those related to debugging, must be specified in the parmfile itself.

Upgrade=<0|1>

To upgrade SUSE Linux Enterprise, specify **Upgrade=1**. A custom parmfile is required for upgrading an existing installation of SUSE Linux Enterprise. Without this parameter, the installation provides no upgrade option.

4.4.2 Configuring the network interface



Important: Configuring the network interface

The settings described in this section apply only to the network interface used during installation.

Hostname=zsystems.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.

Gateway=192.168.1.3

Specify the gateway to use.

Nameserver=192.168.1.4

Specify the DNS server in charge.

InstNetDev=osa

Enter the type of interface to configure. Possible values are osa, hsi, ctc, escon, and iucv (CTC, ESCON, and IUCV are no longer officially supported).

For the ctc interfaces escon and iucv (CTC, ESCON, and IUCV are no longer officially supported), enter the IP address of the peer:

```
Pointopoint=192.168.55.20
```

OsaInterface=<lcs|qdio>

For osa network devices, specify the host interface (qdio or lcs).

Layer2=<0|1>

For osa QDIO Ethernet and hsi devices, specify whether to enable (1) or disable (0) OSI Layer 2 support.

OSAHWAddr=02:00:65:00:01:09

For Layer 2-enabled osa QDIO Ethernet devices. Either specify a MAC address manually or state OSAHWADDR= (with trailing white space) for the system default.

PortNo=<0|1>

For osa network devices, specify the port number (provided the device supports this feature). The default value is 0.

Each of the interfaces requires certain setup options:

- Interfaces ctc and escon (CTC and ESCON are no longer officially supported):

```
ReadChannel=0.0.0600  
WriteChannel=0.0.0601
```

ReadChannel specifies the READ channel to use. WriteChannel specifies the WRITE channel.

- For the ctc interface (no longer officially supported), specify the protocol that should be used for this interface:

```
CTCProtocol=<0/1/2>
```

Valid entries would be:

<u>0</u>	Compatibility mode, also for non-Linux peers other than OS/390 and z/OS (this is the default mode)
<u>1</u>	Extended mode

- Network device type `osa` with interface `lcs`:

```
ReadChannel=0.0.0124
```

`ReadChannel` stands for the channel number used in this setup. A second port number can be derived from this by adding one to `ReadChannel`. `Portnumber` is used to specify the relative port.

- Interface `iucv`:

```
IUCVPeer=PEER
```

Enter the name of the peer machine.

- Network device type `osa` with interface `qdio` for OSA-Express Gigabit Ethernet:

```
ReadChannel=0.0.0700
WriteChannel=0.0.0701
DataChannel=0.0.0702
```

For `ReadChannel`, enter the number of the READ channel. For `WriteChannel`, enter the number of the WRITE channel. `DataChannel` specifies the DATA channel. Make sure that the READ channel has an even device number.

- Interface `hsi` for HiperSockets and VM guest LANs:

```
ReadChannel=0.0.0800
WriteChannel=0.0.0801
DataChannel=0.0.0802
```

For `ReadChannel`, enter the appropriate number for the READ channel. For `WriteChannel` and `DataChannel`, enter the WRITE and DATA channel numbers.

4.4.3 Specifying the installation source and YaST interface

`Install=nfs://server/directory/DVD1/`

Specify the location of the installation source to use. Supported protocols are `nfs`, `smb` (Samba/CIFS), `ftp`, `tftp`, `http`, and `https`.

If an `ftp`, `tftp` or `smb` URL is provided, specify the user name and password. Skip credentials for anonymous or guest login.

```
Install=ftp://USER:PASSWORD@SERVER/DIRECTORY/DVD1/  
Install=tftp://USER:PASSWORD@SERVER/DIRECTORY/DVD1/
```

If you want to perform the installation over an encrypted connection, use an `https` URL. If the certificate cannot be verified, use the `sslcerts=0` boot option to disable certificate checking.

In case of a Samba or CIFS installation, you can also specify the domain:

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

`ssh=1 vnc=1 Display_IP=192.168.42.42`

The installation method depends on which parameter you specify. `ssh` enables SSH installation, `vnc` starts a VNC server on the installing machine, and `Display_IP` causes the installing system to try to connect to an X server at the specified address. Only one of these parameters should be set.



Important: X authentication mechanism

The direct installation with the X Window System relies on an authentication mechanism based on host names. This mechanism is disabled on current SUSE Linux Enterprise Micro versions. We recommend to perform an installation using SSH or VNC is preferred.

To allow a connection between YaST and the remote X server, run `xhost <IP address>` with the address of the installing machine on the remote machine.

For `VNC`, specify a password of six to eight characters to use for installation:

```
VNCPassword=<a password>
```

For `SSH`, specify a password of six to eight characters to use for installation:

```
ssh.password=<a password>
```

4.4.4 I/O device auto-configuration on IBM Z systems

I/O device auto-configuration is a mechanism that allows users to specify IDs and settings of I/O devices that should be automatically enabled in Linux. This information is specified for an LPAR via an HMC running in the DPM (Dynamic Partition Manager) mode.



Note

The I/O device auto-configuration functionality is available on systems with the DPM running. DPM runs by default on LinuxONE machines. For IBM Z, this functionality must be ordered.

In `linuxrc`, the `DeviceAutoConfig` parameter controls the use of I/O device auto-configuration data for IBM Z systems.

DeviceAutoConfig=0

If set to 0, auto-configuration is disabled.

DeviceAutoConfig=1

If set to 1, existing auto-config data are applied.

DeviceAutoConfig=2 (default)

If set to 2 (the default), a dialog is shown if auto-config data are present. The user is asked whether to apply them.

If device auto-config is disabled by the user, the kernel parameter `rd.zdev=no-auto` is added to the boot options of the target system.

To enable I/O auto-configuration using YaST, run the `yast2 system_settings` command, switch to the *Kernel Settings* section, and enable the *Enable I/O device auto-configuration* option.

To disable I/O auto-configuration in an AutoYaST profile, add the following kernel parameter in the `append` section of the global boot loader options. For example:

```
<bootloader>
  <global>
    <append>rd.zdev=no-auto</append>
  </global>
</bootloader>
```

For more context on the AutoYaST boot loader options, see *Book "AutoYaST Guide", Chapter 4 "Configuration and installation options", Section 4.4 "The boot loader"*.

During installation, the status of the auto-configuration setting is displayed in the *Device Settings* section of the *Installation Settings'* screen.

4.4.5 Example parmfiles

The maximum capacity of a parmfile is 860 characters. As a rule of thumb, the parmfile should contain a maximum of 10 lines with no more than 79 characters. When reading a parmfile, all lines are concatenated without adding white spaces, therefore the last character (79) of each line needs to be a `Space` .

To receive potential error messages on the console, use

```
linuxrclog=/dev/console
```

EXAMPLE 4.8: PARMFILE FOR AN INSTALLATION FROM NFS WITH VNC AND AUTOYAST, WITH I/O DEVICE AUTO CONFIGURATION

```
ramdisk_size=131072 root=/dev/ram1 ro init=/linuxrc TERM=dumb
instnetdev=osa osainterface=qdio layer2=1 osahwaddr=
pointopoint=192.168.0.1 hostip=192.168.0.2
nameserver=192.168.0.3 DeviceAutoConfig=1
install=nfs://192.168.0.4/SLES/SLES-12-Server/s390x/DVD1
autoyast=http://192.168.0.5/autoinst.xml
linuxrclog=/dev/console vnc=1 VNCPassword=testing
```

EXAMPLE 4.9: PARMFILE FOR INSTALLATION WITH NFS, SSH, AND HSI AND AUTOYAST WITH NFS

```
ramdisk_size=131072 root=/dev/ram1 ro init=/linuxrc TERM=dumb
AutoYast=nfs://192.168.1.1/autoinst/s390.xml
Hostname=zsystems.example.com HostIP=192.168.1.2
Gateway=192.168.1.3 Nameserver=192.168.1.4
InstNetDev=hsi layer2=0
Netmask=255.255.255.128 Broadcast=192.168.1.255
readchannel=0.0.702c writechannel=0.0.702d datachannel=0.0.702e
install=nfs://192.168.1.5/SLES-12-Server/s390x/DVD1/
ssh=1 ssh.password=testing linuxrclog=/dev/console
```

EXAMPLE 4.10: PARMFILE FOR INSTALLATION IN VLAN

```
ro ramdisk_size=50000 MANUAL=0 PORTNO=1 ReadChannel=0.0.b140
WriteChannel=0.0.b141 DataChannel=0.0.b142
cio_ignore=all,!condev,!0.0.b140-0.0.b142,!0.0.e92c,!0.0.5000,!0.0.5040
HostIP= Gateway= Hostname=zsystems.example.com nameserver=192.168.0.1
Install=ftp://user:password@10.0.0.1/s390x/SLES15.0/INST/ usevnc=1
vncpassword=12345 InstNetDev=osa Layer2=1 OSAInterface=qdio ssl_certs=0
osahwaddr= domain=example.com self_update=0
vlanid=201
```

4.5 Using the vt220 terminal emulator

Recent MicroCode Levels allow the use of an integrated vt220 terminal emulator (ASCII terminal) in addition to the standard line mode terminal. The vt220 terminal is connected to `/dev/ttysclp0`. The line mode terminal is connected to `/dev/ttysclp_line0`. For LPAR installations, the vt220 terminal emulator is activated by default.

To start the ASCII console on HMC, log in to the HMC, and select *Systems Management > Systems > IMAGE_ID*. Select the radio button for the LPAR and select *Recovery > Integrated ASCII Console*.

To redirect the kernel messages at boot time from the system console to the vt220 terminal, add the following entries to the `parameters` line in `/etc/zipl.conf`:

```
console=ttysclp0 console=ttysclp_line0
```

The resulting `parameters` line would look like the following example:

```
parameters = "root=/dev/dasda2 TERM=dumb console=ttysclp0 console=ttysclp_line0"
```

Save the changes in `/etc/zipl.conf`, run `zipl`, and reboot the system.

4.6 More information

Find further technical documentation about IBM Z in the IBM Redbooks (<https://www.redbooks.ibm.com/Redbooks.nsf/domains/zsystems>) or at IBM developerWorks (<https://developer.ibm.com/>). SUSE Linux Enterprise Micro-specific documentation is available from <https://developer.ibm.com/technologies/linux/>.

4.6.1 General documents about Linux on IBM Z

A general coverage of Linux on IBM Z can be found in the following documents:

- Linux on IBM eServer zSeries and S/390: ISP and ASP Solutions (SG24-6299)

These documents might not reflect the current state of Linux, but the principles of Linux deployment outlined there remain accurate.

4.6.2 Technical issues of Linux on IBM Z

Refer to the following documents for technical information about the Linux kernel and application topics. For the most recent versions of the documents, visit (<https://developer.ibm.com/technologies/linux/>).

- Linux on System z Device Drivers, Features, and Commands
- zSeries ELF Application Binary Interface Supplement
- Linux on System z Device Drivers, Using the Dump Tools
- IBM zEnterprise 196 Technical Guide
- IBM zEnterprise EC12 Technical Guide
- IBM z13 Technical Guide
- IBM z14 Technical Guide
- IBM z15 Technical Guide

A Redbook for Linux application development is available at <http://www.redbooks.ibm.com>:

- Linux on IBM eServer zSeries and S/390: Application Development (SG24-6807)

4.6.3 Advanced configurations for Linux on IBM Z

Refer to the following Redbooks, Redpapers, and online resources for more complex IBM Z scenarios:

- Linux on IBM eServer zSeries and S/390: Large Scale Deployment (SG24-6824)
- Linux on IBM eServer zSeries and S/390: Performance Measuring and Tuning (SG24-6926)
- Linux with zSeries and ESS: Essentials (SG24-7025)
- IBM TotalStorage Enterprise Storage Server Implementing ESS Copy Services with IBM eServer zSeries (SG24-5680)
- Linux on IBM zSeries and S/390: High Availability for z/VM and Linux (REDP-0220)
- Saved Segments Planning and Administration
<http://publibz.boulder.ibm.com/epubs/pdf/hcsg4a00.pdf>

- Linux on System z documentation for "Development stream"
- Introducing IBM Secure Execution for Linux, Securing the guest
[https://
www.ibm.com/support/knowledgecenter/linuxonibm/com.ibm.linux.z.lxse/lxse_t_secureexe-
cution.html](https://www.ibm.com/support/knowledgecenter/linuxonibm/com.ibm.linux.z.lxse/lxse_t_secureexecution.html) 

II Raw image deployment

- 5 Description of raw images 62
- 6 Deploying raw images 64
- 7 Configuring with Ignition 68
- 8 Configuring with Combustion 79

5 Description of raw images

SLE Micro raw images are delivered for the AMD64/Intel 64 architecture (an image with RT kernel is available), IBM Z ZSeries (an image of the KVM host, images to be used with FBA or DASD devices) and also AArch64. These images are intended to be configured on the first boot by using either Ignition or Combustion. The boot loader detects the first boot, see more in [Section 5.2, “First boot detection”](#). Each image has default mounted subvolumes as described in [Section 5.1, “Default partitioning”](#). The procedure of deploying these images is described in [Chapter 6, Deploying raw images](#).



Important: Do not use firewall along with Podman

Using firewall along with Podman may cause missing Podman related firewall rules after reloading the `firewalld` service. Therefore, it is recommended to keep the firewall disabled if you intend to use Podman.



Note: SLE Micro as a VM Host Server

SLE Micro can run as a KVM host server—Xen is not supported. However, there are several limitations of SLE Micro running as a VM Host Server, for details refer to [virtualization limits and support \(https://documentation.suse.com/sles/15-SP3/html/SLES-all/cha-virt-support.html\)](https://documentation.suse.com/sles/15-SP3/html/SLES-all/cha-virt-support.html).

5.1 Default partitioning

The raw images are delivered with a default partitioning scheme, which can be changed during the first boot by using Ignition or Combustion. For a procedure to repartition the system, refer to [Section 7.2, “config.ign”](#) or [Section 8.2, “The script configuration file”](#).



Important: Btrfs is mandatory for the root file system

If you intend to perform any changes to the default partitioning scheme, the root file system must be btrfs.

Each image has the following subvolumes:

```
/home
/root
/opt
/srv
/usr/local
/var
```

The images also have mounted subvolumes for booting by default. The specific subvolumes differ according to the architecture.

You can recognize the subvolumes mounted by default by the option `x-initrd.mount` in `/etc/fstab`. Other subvolumes or partitions must be configured either by Ignition or Combustion.

5.2 First boot detection

The configuration runs on the first boot only. To distinguish between the first and subsequent boots, the flag file `/boot/writable/firstboot_happened` is created after the first boot took place. If the file is not present in the file system, the attribute `ignition.firstboot` is passed to the kernel command line, and thus both Ignition and Combustion are triggered to run (in the `initramfs`). After completing the first boot, the `/boot/writable/firstboot_happened` flag file is created.



Note: The flag file is always created

Even though the configuration may not be successful, due to improper or missing configuration files, the `/boot/writable/firstboot_happened` flag file is created.

You may force the first boot configuration on subsequent boot by passing the `ignition.firstboot` attribute to the kernel command line or by deleting the `/boot/writable/firstboot_happened` flag file.

6 Deploying raw images

SUSE Linux Enterprise Micro provides raw images that can be directly deployed to your device storage—a memory card, USB flash drive, or a hard disk. The options for which type of device you can deploy the image to depend on your particular hardware—follow your vendor documentation.

To initially configure the deployed image, you need to prepare the configuration data as described in *Procedure 6.2, “Preparing the configuration device.”* and you need to copy the configuration data to a device (for example USB disk). The device needs to be connected to your host running SLE Micro during its first boot.

To prepare the setup, you need two separate devices. One for the raw disk image, where SLE Micro runs, and another one that serves as a configuration media.

PROCEDURE 6.1: PREPARING THE RAW DISK IMAGE

1. Download the raw image and decompress it:

```
> xz -d DOWNLOADED_IMAGE.raw.xz
```

2. Copy the decompressed image to the device where SLE Micro will run:

```
> dd if=DOWNLOADED_IMAGE.raw of=/dev/sdX
```

The following procedure describes how to prepare the configuration device (usually a USB flash disk).

PROCEDURE 6.2: PREPARING THE CONFIGURATION DEVICE.

1. Format the disk to any file system supported by SLE Micro: Ext3, Ext4, etc.:

```
> sudo mkfs.ext4 /dev/sdY
```

2. Set the device label to either ignition (when either Ignition or Combustion is used) or combustion (when only Combustion is used). For the Ext4 file system:

```
> sudo e2label /dev/sdY ignition
```

You can use any type of configuration storage media that your virtualization system or your hardware supports—ISO image, a USB flash disk, etc.

3. Mount the device:

```
> sudo mount /dev/sdY /mnt
```

4. Create the directory structure as mentioned in *Chapter 7, Configuring with Ignition* or *Chapter 8, Configuring with Combustion*, depending on the configuration tool used:

```
> sudo mkdir -p /mnt/ignition/
```

or:

```
> sudo mkdir -p /mnt/combustion/
```

5. Prior to booting for the first time, prepare all elements of the configuration that will be used by *Ignition* or *Combustion*. To login to your system, you need to provide a password for root or set up passwordless authentication, otherwise the system will not be accessible after the first boot.

After the first boot, you need to register your SUSE Linux Enterprise Micro instance by using the command line tool **SUSEConnect**. For details, refer to *Section 6.1, "Registration"*.

SLE Micro has available extension for live patching. To use this extension, you need to add the extension to your subscription from the installed system. For details, refer to *Section 6.2, "Managing extensions"*.

6.1 Registration

Registering the system is possible from the command line using **SUSEConnect**. For information that goes beyond the scope of this section, refer to the inline documentation with **SUSEConnect --help**

PROCEDURE 6.3: PRODUCT REGISTRATION WITH SUSECONNECT

1. To register SUSE Linux Enterprise Micro with SUSE Customer Center, run **SUSEConnect** as follows:

```
# SUSEConnect -r REGISTRATION_CODE -e EMAIL_ADDRESS
```

To register with a local registration server, additionally provide the URL to the server:

```
# SUSEConnect -r REGISTRATION_CODE -e EMAIL_ADDRESS \
```

```
--url "https://suse_register.example.com/"
```

Replace `REGISTRATION_CODE` with the registration code you received with your copy of SUSE Linux Enterprise Micro. Replace `EMAIL_ADDRESS` with the e-mail address associated with the SUSE account you or your organization uses to manage subscriptions.

2. SUSE Linux Enterprise Micro is now registered.

6.2 Managing extensions

SLE Micro supports the kernel live patching extension. Bear in mind that the extension might require additional subscription.



Note: SUSE Linux Enterprise Live Patching availability

The SUSE Linux Enterprise Live Patching extension is available only for the x86 (except for the real-time kernel) and IBM Z architectures.

As the extension activation or deactivation is performed as a transactional-update and thus creates a new snapshot, you need to restart your system to boot to the new snapshot and apply the changes.

6.2.1 Activating SUSE Linux Enterprise Live Patching

To activate the SUSE Linux Enterprise Live Patching proceed as follows:

1. List available extensions by running:

```
# transactional-update --quiet register -list-extensions
```

2. The output provides you with a command regarding how to activate the live patching extension. You can skip this step if you activated the extension during the manual installation.

```
# transactional-update register -p sle-module-live-patching/15.3/x86_64 \  
-r registration code
```


3. After activating the [SUSE Linux Enterprise Live Patching](#) extension, configure `libzypp` in the `/etc/zypp/zypp.conf` file as follows:

```
multiversion = provides:multiversion(kernel)
```

to keep the current kernel running while patching the system, otherwise you may get dependency conflicts while kernel updates are being applied.

```
multiversion.kernels = latest
```

after applying the live patch, a cleanup of kernels is performed in the new snapshot. If not set, the snapshot keeps the previous kernel and performs kernel updates also on the previous kernel.

4. Additionally, set `LIVEPATCH_KERNEL='always'` in the `/etc/sysconfig/livepatching` file.



Note: Matching version of the kernel-default-livepatch and kernel

To ensure that the live patches will be installed even after the kernel upgrade, install the matching version of the `kernel-default-livepatch` package.

5. Now install the extension by running:

```
# transactional-update pkg install kernel-default-livepatch
```

6. Reboot your system to switch to the new snapshot.

6.2.2 Deactivating SUSE Linux Enterprise Live Patching

To deactivate the extension, run the following command:

```
# transactional-update register -d \  
-p sle-module-live-patching/15.3/x86_64
```

7 Configuring with Ignition

This chapter provides details about the Ignition provisioning tool that is used to set up a machine. Here you'll learn how to provide required configuration files used for the machine definition.

7.1 About Ignition

Ignition is a provisioning tool that enables you to configure a system according to your specification on the first boot. When the system is booted for the first time, Ignition is loaded as part of an `initramfs` and searches for a configuration file within a specific directory (on a USB flash disk or you can provide a URL). All changes are performed before kernel switches from the temporal file system to the real root file system (before the `switch_root` command is issued). Ignition uses a configuration file in the JSON format. The file is called `config.ign`.

7.2 `config.ign`

The `config.ign` is a JSON configuration file that provides prescriptions for Ignition. You can either create the file manually in JSON, or you can use the Fuel Ignition tool (<https://opensuse.github.io/fuel-ignition/>) to generate a basic set of prescriptions. Bear in mind that the Fuel Ignition tool does not provide a full set of options, so you might have to modify the file manually. Alternatively, for the purpose of better human readability, you can create the file `config.fcc` in YAML and transpile the file to JSON. For details, refer to [Section 7.2.2, "Converting YAML fcc file to JSON ign"](#).

When installing on bare metal, the configuration file `config.ign` must reside in the `ignition` subdirectory on the configuration media labeled `ignition`. The directory structure must look as follows:

```
<root directory>
└─ ignition
   └─ config.ign
```

In case you intend to configure a QEMU/KVM virtual machine, provide the path to the `config.ign` as an attribute of the `qemu` command. For example:

```
-fw_cfg name=opt/com.coreos/config,file=PATH_TO_config.ign
```

The `config.ign` contains various data types: objects, strings, integers, Booleans and lists of objects. For a complete specification, refer to [Ignition specification v3.3.0 \(https://coreos.github.io/ignition/configuration-v3_3/\)](https://coreos.github.io/ignition/configuration-v3_3/).

The `version` attribute is mandatory, and in the case of SLE Micro, its value must be set either to `3.3.0` or to any lower version. Otherwise Ignition will fail.

If you want to log in to your system as root, you must at least include a password for `root`. However, it is recommended to establish access via SSH keys. If you want to configure a password, make sure to use a secure one. If you use a randomly generated password, use at least 10 characters. If you create your password manually, use even more than 10 characters and combine uppercase and lowercase letters and numbers.

7.2.1 Configuration examples

This section will provide you with some common examples of the Ignition configuration in both JSON and YAML formats. If you create configuration in the YAML format, you need to transpile the configuration to JSON as described in [Section 7.2.2, “Converting YAML fcc file to JSON ign”](#).



Important: Declaring content outside of default subvolumes

Bear in mind that if you want to create files outside the *default mounted directories*, you need to define the directories by using the `filesystem` attribute.



Note: The `version` attribute is mandatory

Include the version specification in `config.ign` (version 3.3.0 or lower), resp. `config.fcc` (version 1.4.0 or lower).

7.2.1.1 Storage configuration

The `storage` attribute is used to configure partitions, RAID, define file systems, create files, etc. To define partitions, use the `disks` attribute. The `filesystem` attribute is used to format partitions and define mount points of particular partitions. The `files` attribute can be used to create files in the file system. Each of the mentioned attributes is described in following sections.

7.2.1.1.1 The disks attribute

The `disks` attribute is a list of devices that enables you to define partitions on these devices. The `disks` attribute must contain at least one `device`, other attributes are optional. The following example will use a single virtual device and divide the disk into four partitions:

```
{
  "variant": "fcos",
  "version": "3.3.0",
  "storage": {
    "disks": [
      {
        "device": "/dev/vda",
        "wipe_table": true,
        "partitions": [
          {
            "label": "root",
            "number": 1,
            "type_guid": "4F68BCE3-E8CD-4DB1-96E7-FBCAF984B709"
          },
          {
            "label": "boot",
            "number": 2,
            "type_guid": "BC13C2FF-59E6-4262-A352-B275FD6F7172"
          },
          {
            "label": "swap",
            "number": 3,
            "type_guid": "0657FD6D-A4AB-43C4-84E5-0933C84B4F4F"
          },
          {
            "label": "home",
            "number": 4,
            "type_guid": "933AC7E1-2EB4-4F13-B844-0E14E2AEF915"
          }
        ]
      }
    ]
  }
}
```

The same example in the YAML format:

```
variant: fcos
version: 1.4.0
storage:
  disks:
```

```

- device: "/dev/vda"
  wipeTable: true
  partitions:
    - label: root
      number: 1
      typeGuid: 4F68BCE3-E8CD-4DB1-96E7-FBCAF984B709
    - label: boot
      number: 2
      typeGuid: BC13C2FF-59E6-4262-A352-B275FD6F7172
    - label: swap
      number: 3
      typeGuid: 0657FD6D-A4AB-43C4-84E5-0933C84B4F4F
    - label: home
      number: 4
      typeGuid: 933AC7E1-2EB4-4F13-B844-0E14E2AEF915

```

7.2.1.1.2 The raid attribute

The `raid` is a list of RAID arrays. The following attributes of `raid` are mandatory:

level

a level of the particular RAID array (linear, raid0, raid1, raid2, raid3, raid4, raid5, raid6)

devices

a list of devices in the array referenced by their absolute paths

name

a name that will be used for the md device

```

{
  "variant": "fcos",
  "version": "3.3.0",
  "storage": {
    "raid": [
      {
        "name": "system",
        "level": "raid1",
        "devices": [
          "/dev/sda",
          "/dev/sdb"
        ]
      }
    ]
  }
}

```

```
}
```

The same example in the YAML format:

```
variant: fcos
version: 1.4.0
storage:
  - raid: data
    name: system
    level: raid1
    devices: "/dev/sda", "/dev/sdb"
```

7.2.1.1.3 The filesystem attribute

filesystem must contain the following attributes:

device

the absolute path to the device, typically /dev/sda in case of physical disk

format

the file system format (btrfs, ext4, xfs, vfat or swap)



Note

In the case of SLE Micro, the root file system must be formatted to btrfs.

The following example demonstrates using the filesystem attribute. The /opt directory will be mounted to the /dev/sda1 partition, which is formatted to btrfs. The partition table will not be erased.

```
{
  "variant": "fcos",
  "version": "3.3.0",
  "storage": {
    "filesystems": [
      {
        "path": "/opt",
        "device": "/dev/sda1",
        "format": "btrfs",
        "wipe_filesystem": false
      }
    ]
  }
}
```

```
}  
}
```

The same example in the YAML format:

```
variant: fcos  
version: 1.4.0  
storage:  
  filesystems:  
    - path: /opt  
      device: "/dev/sda1"  
      format: btrfs  
      wipe_filesystem: false
```

7.2.1.1.4 The files attribute

You can use the `files` attribute to create any files on your machine. Bear in mind that if you want to create files outside the *default mounted directories*, you need to define the directories by using the `filesystem` attribute.

In the following example, a host name is created by using the `files` attribute. The file `/etc/hostname` will be created with the `slemicro-1` host name.



Note: Different numeral systems in JSON and YAML

Bear in mind that JSON uses the decimal numeral system, so the `mode` value is a decimal notation of the access rights. To use the octal notation, prefer YAML in this case.

```
{  
  "variant": "fcos",  
  "version": "3.3.0",  
  "storage": {  
    "files": [  
      {  
        "path": "/etc/hostname",  
        "mode": 420,  
        "overwrite": true,  
        "contents": {  
          "inline": "slemicro-1"  
        }  
      }  
    ]  
  }  
}
```

```
}  
}
```

The same example in YAML:

```
variant: fcos  
version: 1.4.0  
storage:  
  files:  
    - path: /etc/hostname  
      mode: 0644  
      overwrite: true  
      contents:  
        inline: "slemicro-1"
```

7.2.1.1.5 The `directories` attribute

The `directories` attribute is a list of directories that will be created in the file system. The `directories` attribute must contain at least one `path` attribute.

```
{  
  "variant": "fcos",  
  "version": "3.3.0",  
  "storage": {  
    "directories": [  
      {  
        "path": "/mnt/backup",  
        "user": {  
          "name": "tux"  
        }  
      }  
    ]  
  }  
}
```

The same example in YAML:

```
variant: fcos  
version: 1.4.0  
storage:  
  directories:  
    - path: /mnt/backup  
      user:  
        - name: tux
```


7.2.1.2 Users administration

The `passwd` attribute is used to add users. If you intend to log in to your system, create `root` and set the `root`'s password and/or add the SSH key to the Ignition configuration. You need to hash the `root` password, for example, by using the `openssl` command:

```
openssl passwd -6
```

The command creates a hash of the password you chose. Use this hash as the value of the `password_hash` attribute.

```
variant: fcos
version: 1.0.0
passwd:
  users:
    - name: root
      password_hash: "$6$PfKm6Fv5Wbq0vZ0C
$g4kByYM.D2B5GCsgluuqDNL87oeXiHqctr6INNNmF75WPGgkLn909uVx4iEe3UdbbhaHbTJ1vpZymKWuDIrWI1"
      ssh_authorized_keys:
        - ssh-rsa long...key user@host
```

The `users` attribute must contain at least one `name` attribute. `ssh_authorized_keys` is a list of ssh keys for the user.



Note: Creating other users than root

When you are creating other users than `root`, you need to define `/home` directories for the users, because these directories (usually `/home/USER_NAME`) are not mounted by default. Therefore, declare these directories using the `storage/filesystem` attribute. For example, for the `tux`, the example looks as follows:

```
{
  "ignition": {
    "version": "3.2.0"
  },
  "passwd": {
    "users": [
      {
        "name": "tux",
        "passwordHash": "$2a$10$US9XSqL0qMmGq/
Onh1VjP0wuZREh2.iEt1wD5LI7DKgV24NJU.w06"
      }
    ]
  },
}
```

```

"storage": {
  "filesystems": [
    {
      "device": "/dev/disk/by-label/ROOT",
      "format": "btrfs",
      "mountOptions": [
        "subvol=/@/home"
      ],
      "path": "/home",
      "wipeFilesystem": false
    }
  ]
}

```

The same in YAML:

```

version: 1.2.0
passwd:
  users:
    - name: tux
      passwordHash: $2a$10$US9XSqL0qMmGq/Onh1VjP0wuZREh2.iEt1wD5LI7DKgV24NJU.w06
storage:
  filesystems:
    - device: /dev/disk/by-label/ROOT
      format: btrfs
      mountOptions:
        - subvol=/@/home
      path: /home
      wipeFilesystem: false

```

7.2.1.3 Enabling systemd services

You can enable `systemd` services by specifying them in the `systemd` attribute. The `name` must be the exact name of a service to be enabled (including the suffix).

```

variant: fcos
version: 1.0.0
systemd:
  units:
    - name: sshd.service
      enabled: true

```

```
{
```

```

"ignition": {
  "version": "3.0.0"
},
"systemd": {
  "units": [
    {
      "enabled": true,
      "name": "sshd.service"
    }
  ]
}
}

```

The same example in YAML:

```

variant: fcos
version: 1.0.0
systemd:
  units:
  - name: sshd.service
    enabled: true

```

7.2.2 Converting YAML fcc file to JSON ign

To make the Ignition configuration more human-readable, you can use a two-phase configuration. First, prepare your configuration in YAML as a `fcc` file and then transpile this configuration to JSON. The transpilation can be done by the `butane` tool.

During the transpilation, `butane` also verifies the syntax of the YAML file to catch potential errors in the structure. For the latest version of the `butane` tool, add a repository:

```

> sudo zypper ar -f \
  https://download.opensuse.org/repositories/devel:/kubic:/ignition/DISTRIBUTION/ \
  devel_kubic_ignition

```

where `DISTRIBUTION` is one of the following (depending on your distribution):

- `openSUSE_Tumbleweed`
- `openSUSE_Leap_${release_number}`
- `15.a`

Now you can install the `butane` tool:

```

> sudo zypper in butane

```

Now you can invoke butane by running:

```
> butane -p -o config.ign config.fcc
```

where:

- config.fcc is the path to the YAML configuration file
- config.ign is the path to the output JSON configuration file
- The -p command option adds line breaks to the output file and thus makes it more readable.

8 Configuring with Combustion

This chapter describes Combustion—the tool used to configure your system on first boot according to your configuration.

8.1 About Combustion

Combustion is a dracut module that enables you to configure your system on its first boot. Combustion reads a provided file, called `script` and executes commands in it and thus perform changes to the file system. You can use Combustion to change the default partitions, set users' passwords, create files, install packages, etc.

The Combustion dracut module is invoked after the `ignition.firstboot` argument is passed to the kernel command line. Combustion then reads configuration from `script`. Combustion tries to configure network, if the network flag has been found in `script`. After `/sysroot` is mounted, Combustion tries to activate all mount points in `/etc/fstab` and then call **transactional-update** to apply other changes (like setting `root` password or installing packages).

When using Combustion, you need to label the configuration device with the name `combustion`, create a specific directory structure in that configuration medium, and include a configuration file named `script`. In the root directory of the configuration medium, create a directory called `combustion` and place the `script` into this directory along with other files—SSH key, configuration files, etc. The directory structure then should look as follows:

```
<root directory>
├── combustion
│   ├── script
│   └── other files
```

You can use Combustion to configure your QEMU/KVM virtual machine. In this case, pass the location of the `script` file using the `fw_cfg` parameter of the `qemu` command:

```
-fw_cfg name=opt/org.opensuse.combustion/script,file=/var/combustion-script
```

Combustion can be used along with Ignition. If you intend to do so, label your configuration medium `ignition` and include the `ignition` directory with the `config.ign` to your directory structure as shown below:

```
<root directory>
```

```
└─ combustion
  └─ script
    └─ other files
└─ ignition
  └─ config.ign
```

In this scenario, Ignition runs before Combustion.

8.2 The script configuration file

The `script` configuration file is a set of commands that are executed on your system in a transactional-update shell. This section provides examples for performing various configuration tasks by using Combustion.



Important: Include interpreter declaration

As the `script` file is interpreted by shell, make sure to start the file with the interpreter declaration at the first line, for example for Bash:

```
#!/bin/bash
```

If you want to log in to your system, include at least the `root` password. Though it is recommend to establish the authentication using SSH keys. If you need to use a `root` password, make sure to configure a secure password. In case you use a randomly generated password, use at least 10 characters. If you create your password manually, use even more than 10 characters and combine uppercase and lowercase letters, and numbers.

8.2.1 Network configuration

To configure and use network connection during the first boot, add the following statement to your `script`:

```
# combustion: network
```

Using this statement will pass the `rd.neednet=1` argument to dracut. If you do not use the statement, the system will be configured without any network connection.

8.2.2 Partitioning

SLE Micro raw images are delivered with a default partitioning scheme as described in [Section 5.1, “Default partitioning”](#). You might want to use a different partitioning. The following set of example snippets moves the `/home` to a different partition.



Note: Performing changes outside of directories included in snapshots

The following script performs changes that are not included in snapshots. If the script fails and the snapshot get discarded, some changes remains visible and cannot be reverted (like the changes to the `/dev/vdb` device.)

The following snippet creates a GPT with a single partition on the `/dev/vdb` device:

```
sfdisk /dev/vdb <<EOF
label: gpt
type=linux
EOF

partition=/dev/vdb1
```

The partition is formatted to BTRFS:

```
wipefs --all ${partition}
mkfs.btrfs ${partition}
```

Possible content of `/home` is moved to the new `/home` folder location by the following snippet:

```
mount /home
mount ${partition} /mnt
rsync -aAXP /home/ /mnt/
umount /home /mnt
```

The snippet below removes an old entry in `/etc/fstab` and creates a new entry:

```
awk -i inplace '$2 != "/home"' /etc/fstab
echo "$(blkid -o export ${partition} | grep ^UUID=) /home btrfs defaults 0 0" >>/etc/
fstab
```

8.2.3 Setting a password for root

Before you set the `root` password, generate a hash of the password, e.g. by using the `openssl passwd -6`. To set the password, add the following to your `script`:

```
echo 'root:$5$.wn2BZHLEJ5R3B1C$TAHEchlU.h2tvf0p0ki54NaHpGYKwdNhjaBuSpDotD7' | chpasswd -e
```

8.2.4 Adding SSH keys

The following snippet creates a directory to store the `root`'s SSH key and then copies the public SSH key located on the configuration device to the `authorized_keys` file.

```
mkdir -pm700 /root/.ssh/  
cat id_rsa_new.pub >> /root/.ssh/authorized_keys
```



Note

The SSH service must be enabled in case you need to use remote login via SSH. For details, refer to [Section 8.2.5, "Enabling services"](#).

8.2.5 Enabling services

You may need to enable some services, for example the SSH service. To enable the SSH service add the following line to `script`:

```
systemctl enable sshd.service
```

8.2.6 Installing packages



Important: Network connection and registering your system might be necessary

As some package may require additional subscription, you might need to register your system beforehand. Also available network connection may be needed to install additional packages.

During the first boot configuration, you can install additional packages to your system. For example, you can install the `vim` editor by adding:

```
zypper --non-interactive install vim-small
```



Note

Bear in mind that you will not be able to use `zypper` after the configuration is complete and you boot to the configured system. To perform changes later, you must use the `transactional-update` command to create a changed snapshot. For details, refer to *Article "Administration Guide", Section 2 "Administration using transactional updates"*.

III Manual installation

- 9 Boot parameters **85**
- 10 Installation steps **98**
- 11 Remote installation **133**
- 12 Troubleshooting **142**

9 Boot parameters

SUSE Linux Enterprise Micro 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 SUSE Linux Enterprise Micro 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*.

9.1 Using the default boot parameters

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 12, 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 9, Boot parameters](#)). A detailed description of the available function keys is available in [Section 9.2.1, “The boot screen on machines equipped with traditional BIOS”](#).

9.2 PC (AMD64/Intel 64/Arm AArch64)

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

9.2.1 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.

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 12.4, "Boot failure"](#).

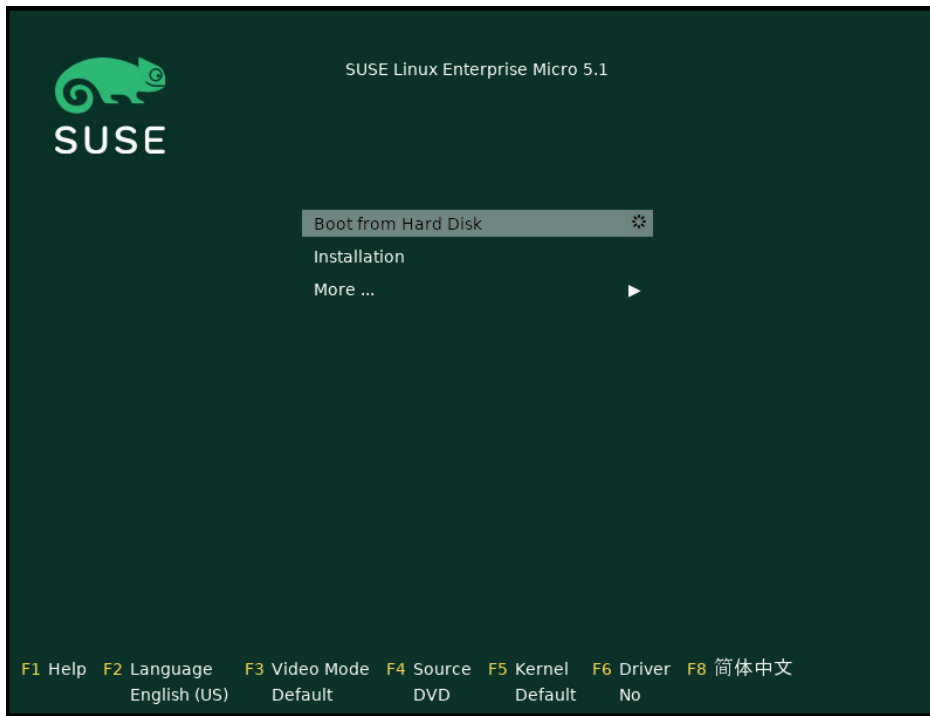


FIGURE 9.1: THE BOOT SCREEN ON MACHINES WITH A TRADITIONAL BIOS

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 SUSE Linux Enterprise Micro. 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.

9.2.2 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.

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 SUSE Linux Enterprise Micro. This is also true of drivers shipped via SolidDriver, because their signing key is not trusted by default.

To load drivers not shipped with SUSE Linux Enterprise Micro, 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.

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.

More > Rescue System

Starts a minimal Linux system without a graphical user interface. For more information, see

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.

The Boot Screen on Machines with UEFI

FIGURE 9.2: THE BOOT SCREEN ON MACHINES WITH UEFI

GRUB 2 for EFI on SUSE Linux Enterprise Micro 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 <https://en.opensuse.org/Linuxrc>.

9.3 List of important boot parameters

This section contains a selection of important boot parameters.

9.3.1 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 SUSE Linux Enterprise Micro, 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.

9.3.2 Configuring the network interface



Important: Configuring the network interface

The settings discussed in this section apply only to the network interface used during installation.

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

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]=[.try,]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. If you add the try option, configuration will stop when the installation repository can be reached via one of the configured interfaces.

Alternatively you use static configuration. With static parameters, only the first matching interface will be configured, unless you add the try option. This will configure all interfaces until the repository can be reached.

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"
```



Tip: Other networking parameters

The ifcfg boot parameter is very powerful and allows you to set almost all networking parameters. In addition to the parameters mentioned above, you can set values for all configuration options (comma separated) from /etc/sysconfig/network/ifcfg.template and /etc/sysconfig/network/config. The following example sets a custom MTU size on an interface otherwise configured via DHCP:

```
ifcfg=eth0=dhcp,MTU=1500
```

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.

9.3.3 Specifying the installation source

If you are not using DVD or USB flash drive 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:/
```

9.3.4 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. For information about how to use the parameters listed in this section, see [Chapter 11, Remote installation](#).

`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 SUSE Linux Enterprise Micro 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.

9.4 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.

9.4.1 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
```

9.4.2 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.

On systems with traditional BIOS, press **F4** on the boot screen and set the required parameters in the *HTTP Proxy* dialog.

On Systems with UEFI BIOS, provide the boot parameter `proxy` at the boot prompt:

1. On the boot screen, press **E** to edit the boot menu.
2. Append the `proxy` paramter to the `linux` line in the following format:

```
proxy=https://proxy.example.com:PORT
```

If the proxy server requires authentication, add the credentials as follows:

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

If the proxy server's SSL certificate cannot be verified, disable certificate checking with the `sslcerts=0` boot parameter.

The outcome will be similar to the following:



```
SUSE Linux Enterprise 15 SP3
setparams 'Installation'
set ifconfig loadkeep
echo loading kernel ...
linux /boot/x86_64/loader/linux splash=silent proxy=https://USER:PASSWORD@proxy.example.com:3128 sslcert
set
echo loading initial ramdisk ...
initrd /boot/x86_64/loader/initrd

Minimal Emacs-like screen editing is supported. TAB lists completions. Press Ctrl+u or F10 to
boot, Ctrl+u or F2 for a covered line or ESC to discard edits and return to the GRUB menu.
```

FIGURE 9.3: GRUB OPTIONS EDITOR

3. Press **F10** to boot with the new proxy setting.

9.4.3 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
```

9.4.4 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
```

9.4.5 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.

9.5 More information

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

10 Installation steps

This chapter describes the procedure in which the data for SUSE Linux Enterprise Micro 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 but looks different. For information about performing non-interactive automated installations, see *Book "AutoYaST Guide"*.

If you are a first-time user of SUSE Linux Enterprise Micro, 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 `→|`.

10.1 Overview

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

1. At first, YaST performs network configuration. For details, refer to [Section 10.2, "Network settings"](#).
2. The actual installation starts with language and keyboard selection and the license agreement. For details, refer to [Section 10.3, "Language, Keyboard, and License Agreement"](#).
3. Accept the license agreement to proceed to the next step.
4. IBM Z machines need to activate disks. For details, see [Section 10.4, "IBM Z: disk activation"](#).
5. Register your system. For details, refer to [Section 10.5, "Registration"](#).

6. Install available extensions. For details, refer to [Section 10.6, “Extension and Module Selection”](#)
7. Configure NTP servers as described in [Section 10.7, “NTP Configuration”](#).
8. Set a password for the system administrator `root`. For details, refer to [Section 10.8, “Authentication for the System Administrator root”](#).
9. The last installation step is an overview of all installation settings. For details, refer to [Section 10.9, “Installation Settings”](#).

10.2 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.

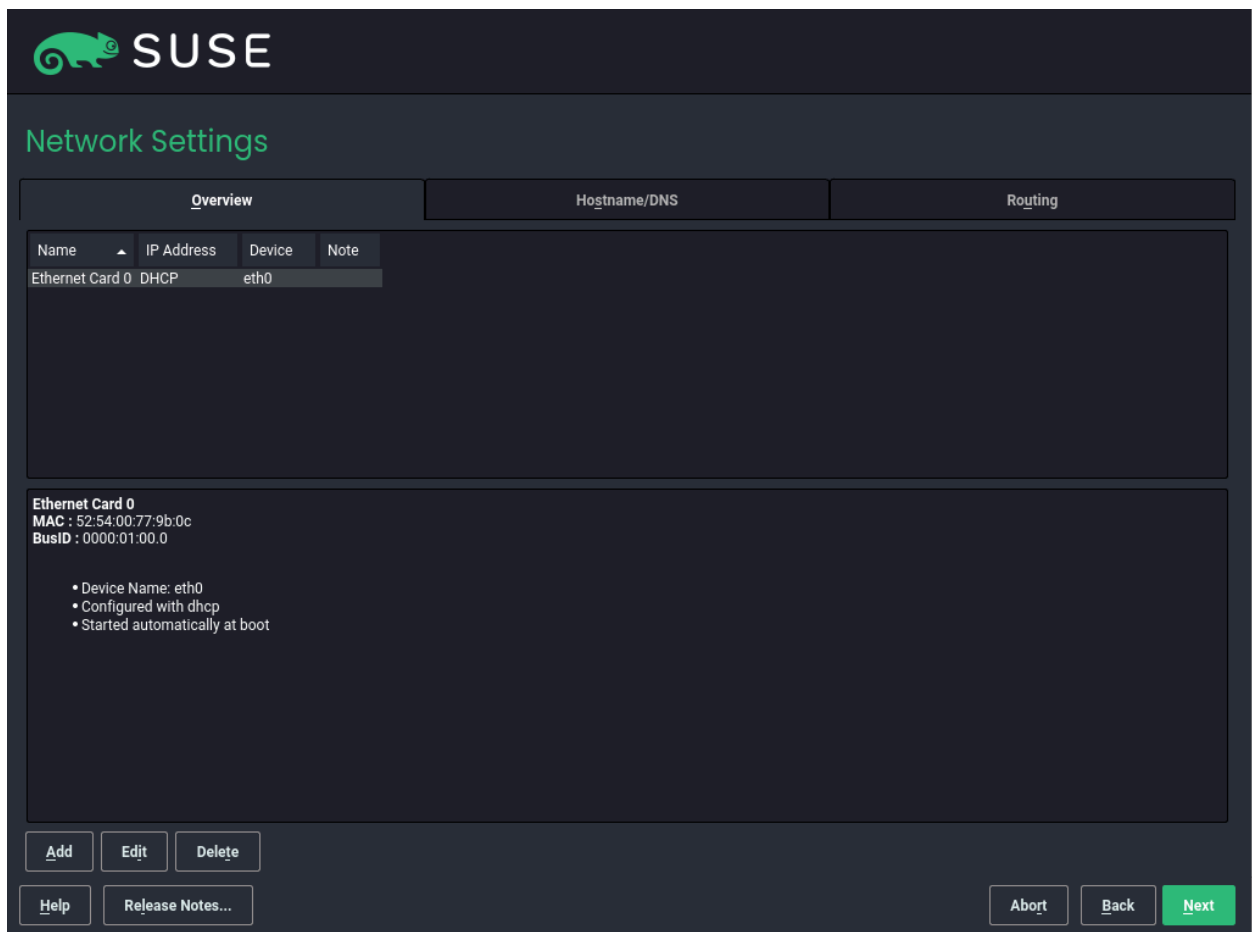


FIGURE 10.1: NETWORK SETTINGS

Choose a network interface from the list and click *Edit* to change its settings. Use the tabs to configure DNS and routing. On IBM Z this dialog does not start automatically. It can be started in the *Disk Activation* step.

In case DHCP was successfully configured during installation setup, you can also access this dialog by clicking *Network Configuration* at the *SUSE Customer Center Registration* and 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 9.3.2, “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.

10.3 Language, Keyboard, and License Agreement

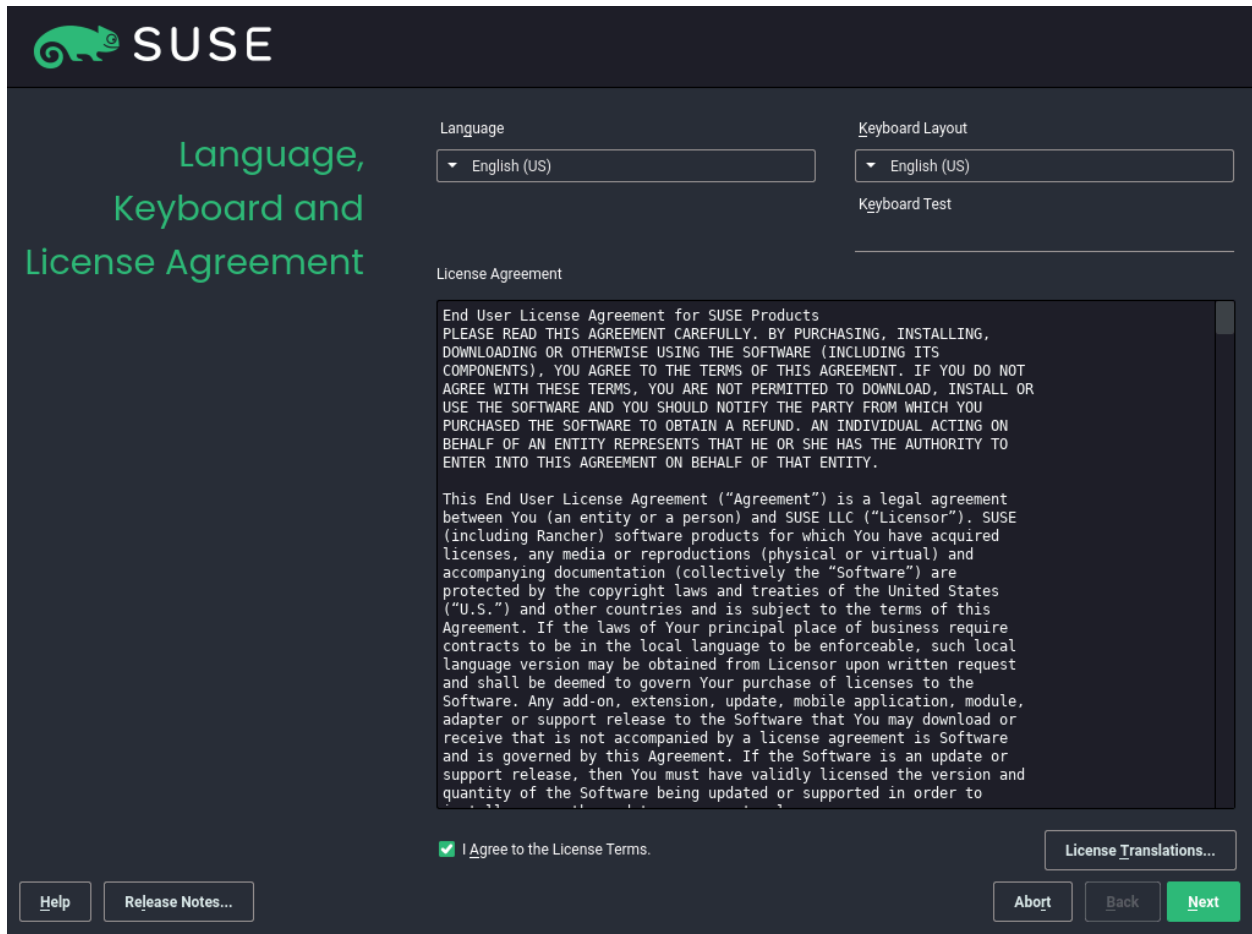


FIGURE 10.2: LICENSE AGREEMENT

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.

By clicking *Release notes* you can access the SLE Micro release notes in English.

Read the License Agreement. It is presented in the language you have chosen on the boot screen. Translations are available via the *License Translations > Language* drop-down box. If you agree to the terms, check *I Agree to the License Terms* and click *Next* to proceed with the installation. If you do not agree to the license agreement, you cannot install SUSE Linux Enterprise Micro. Click *Abort* to terminate the installation.

10.4 IBM Z: disk activation

When installing on IBM Z platforms, the language selection dialog is followed by a dialog to configure the attached hard disks.

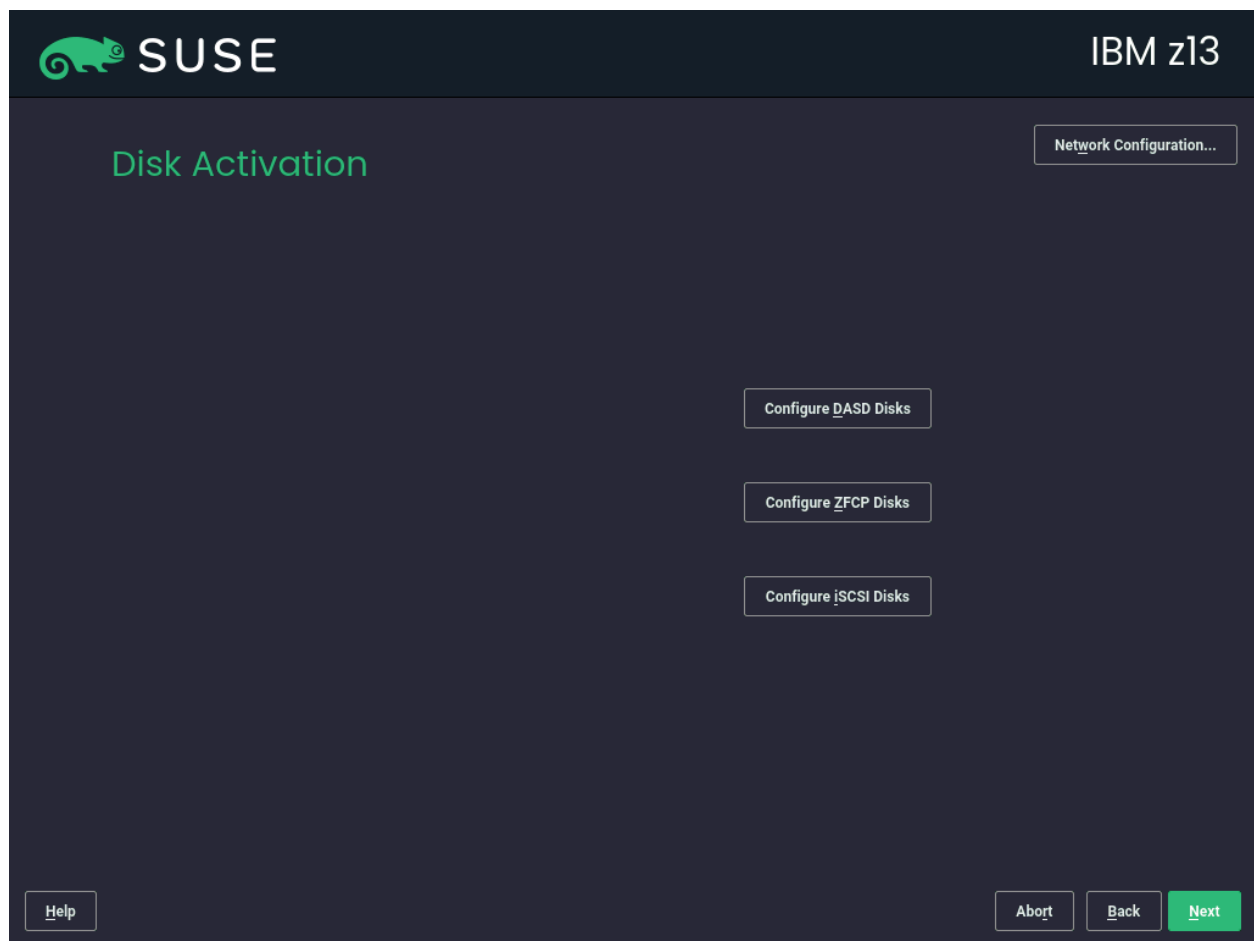


FIGURE 10.3: DISK ACTIVATION

Select DASD, Fibre Channel Attached SCSI Disks (zFCP), or iSCSI for installation of SUSE Linux Enterprise Micro. The DASD and zFCP configuration buttons are only available if the corresponding devices are attached.

You can also change the *Network Configuration* in this screen by launching the *Network Settings* dialog. Choose a network interface from the list and click *Edit* to change its settings. Use the tabs to configure DNS and routing.

10.4.1 Configuring DASD disks

Skip this step if you are not installing on IBM Z hardware.

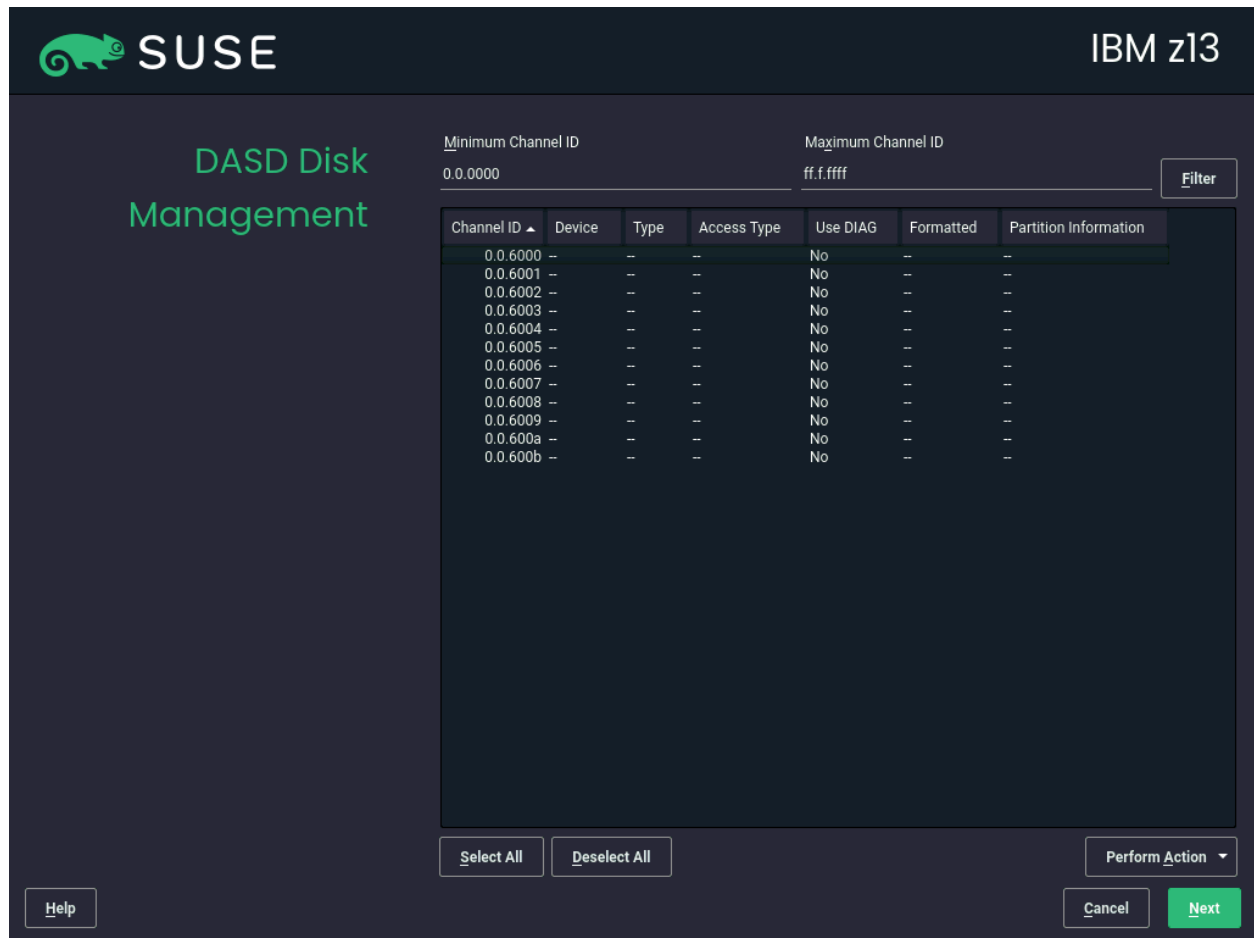


FIGURE 10.4: DASD DISK MANAGEMENT

After selecting *Configure DASD Disks*, an overview lists all available DASDs. To get a clearer picture of the available devices, use the text box located above the list to specify a range of channels to display. To filter the list according to such a range, select *Filter*.

Specify the DASDs to use for the installation by selecting the corresponding entries in the list. Use *Select All* to select all DASDs currently displayed. Activate and make the selected DASDs available for the installation by selecting *Perform Action > Activate*. To format the DASDs, select *Perform Action > Format*.

10.4.2 Configuring zFCP disks

Skip this step if you are not installing on IBM Z hardware.

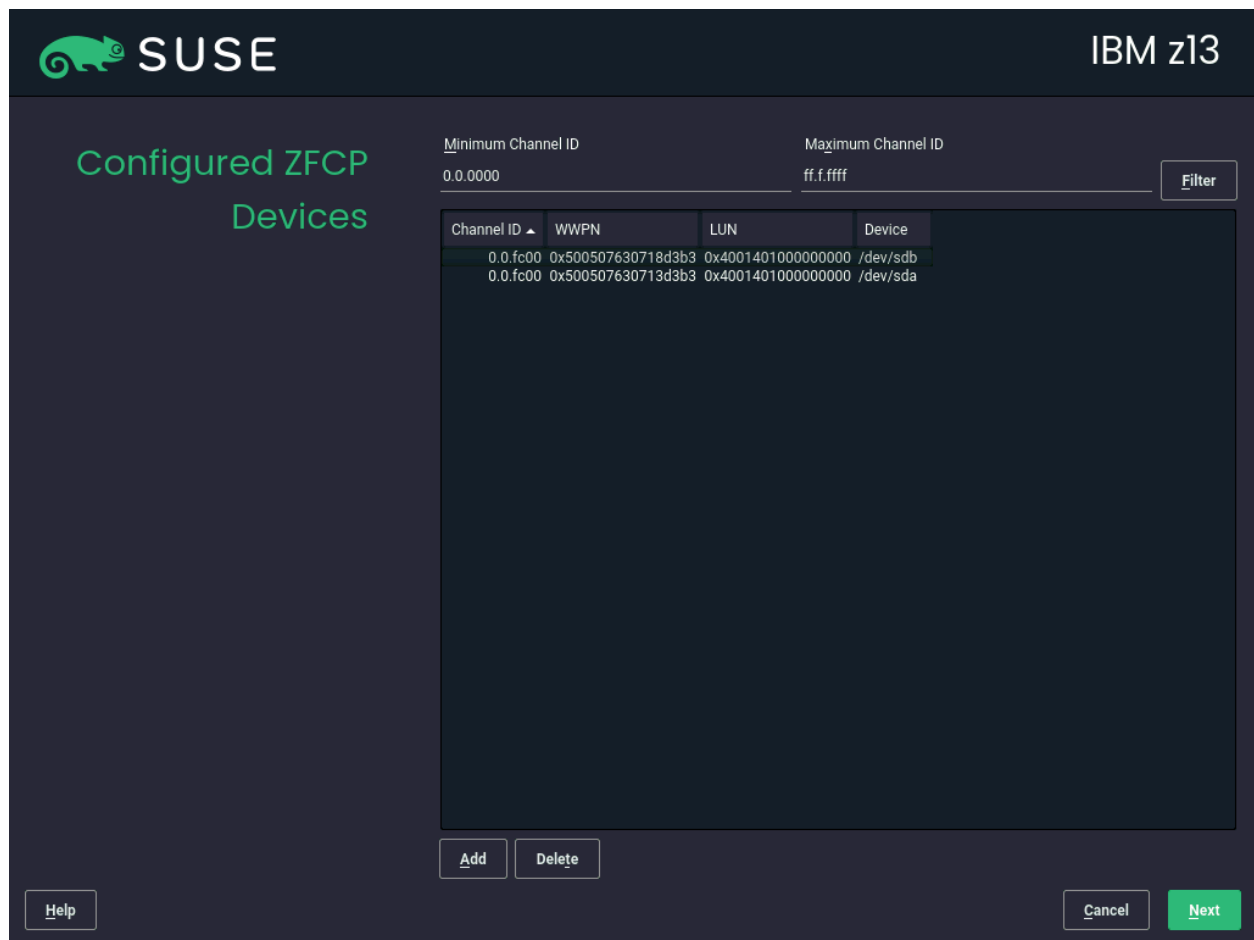


FIGURE 10.5: CONFIGURED ZFCP DEVICES

After selecting *Configure zFCP Disks*, a dialog with a list of the zFCP disks available on the system opens. In this dialog, select *Add* to open another dialog in which to enter zFCP parameters.

To make a zFCP disk available for the SUSE Linux Enterprise Micro installation, choose an available *Channel Number* from the drop-down box. *Get WWPNS* (World Wide Port Number) and *Get LUNs* (Logical Unit Number) return lists with available WWPNS and FCP-LUNs, respectively, to choose from. Automatic LUN scanning only works with NPIV enabled.

When completed, exit the zFCP dialog with *Next* and the general hard disk configuration dialog with *Finish* to continue with the rest of the configuration.

10.5 Registration

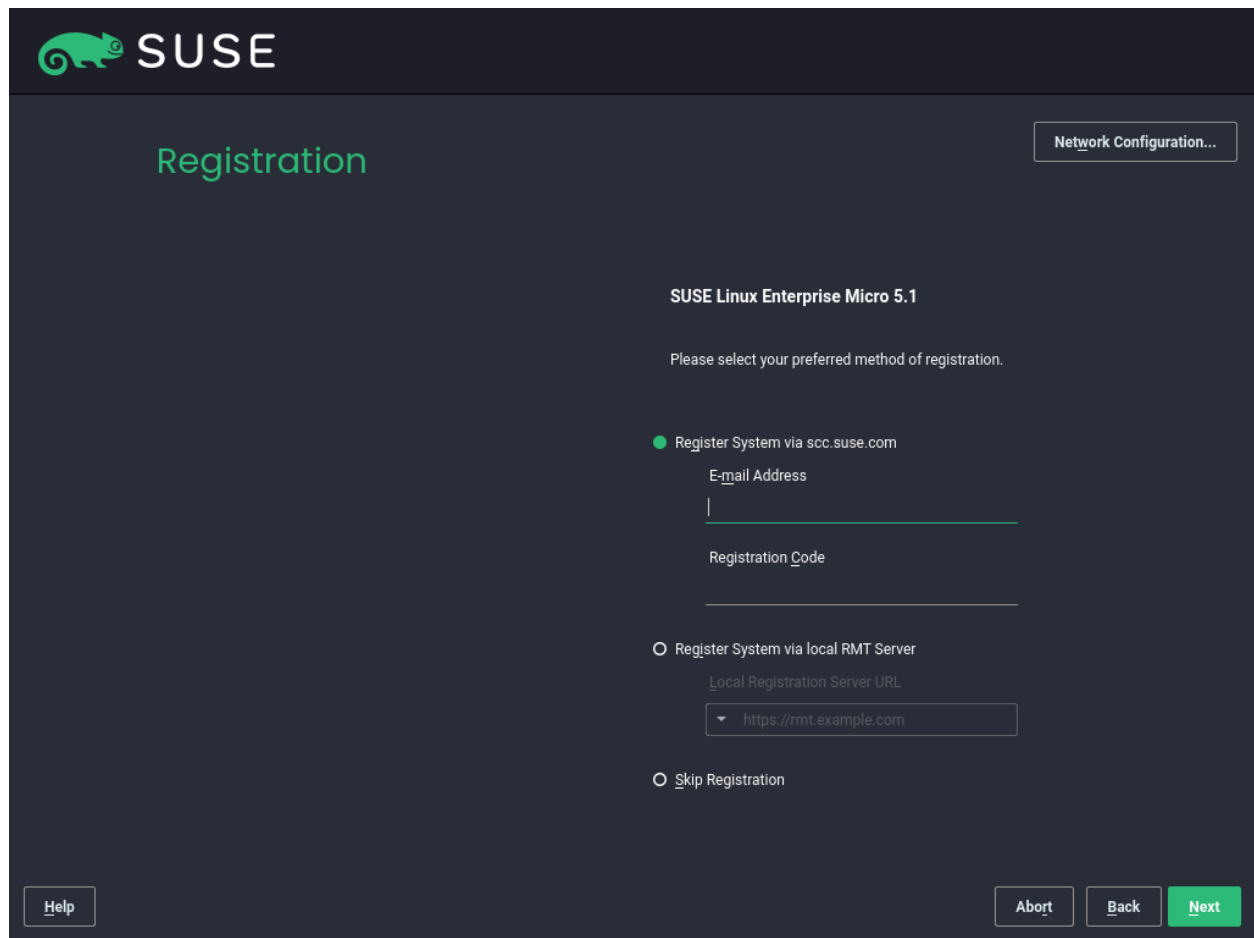


FIGURE 10.6: REGISTRATION

To get technical support and product updates, you need to register and activate SUSE Linux Enterprise Micro with the SUSE Customer Center or a local registration server. Registering your product at this stage also grants you immediate access to the update repository. This enables you to install the system with the latest updates and patches available.

From this dialog, you can switch to the YaST *Network Settings* module by clicking *Network Configuration*. For details, see [Section 10.2, "Network settings"](#).

The dialog offers the following possibilities, each described further:

Register system via scc.suse.com

To register with the SUSE Customer Center, enter the *E-mail Address* associated with your SCC account and the *Registration Code* for SUSE Linux Enterprise Micro. Proceed with *Next*.

Register system via local RMT server

If your organization provides a local registration server, you may alternatively register there. Activate *Register System via local RMT Server* and either choose a URL from the drop-down box or type in an address. Proceed with *Next*.

Skip registration

If you want to skip registration or you are offline, click *Skip Registration*. Accept the warning with *OK* and proceed with *Next*.



Important: Skipping registration

Your system needs to be registered in order to retrieve updates and to be eligible for support. You can register later, after the installation by using **SUSEConnect**, for details, see [Section 6.1, "Registration"](#).



Tip: Installing product patches at installation time

After SUSE Linux Enterprise Micro has been successfully registered, you are asked whether to install the latest available online updates during the installation. If choosing *Yes*, the system will be installed with the most current packages without having to apply the updates after installation. Activating this option is recommended.

If the system was successfully registered during installation, YaST will disable repositories from local installation media such as CD/DVD or flash disks when the installation has been completed. This prevents problems if the installation source is no longer available and ensures that you always get the latest updates from the online repositories.

10.5.1 Loading registration codes from USB storage

To make the registration more convenient, you can also store your registration codes on a USB storage device such as a flash disk. YaST will automatically pre-fill the corresponding text box. This is particularly useful when testing the installation or if you need to register many systems or extensions.

Create a file named `regcodes.txt` or `regcodes.xml` on the USB disk. If both are present, the XML takes precedence.

In that file, identify the product with the name returned by `zypper search --type product` and assign it a registration code as follows:

EXAMPLE 10.1: `regcodes.txt`

```
SLEMicro    cc36aae1
```

EXAMPLE 10.2: `regcodes.xml`

```
<?xml version="1.0"?>
<profile xmlns="http://www.suse.com/1.0/yast2ns"
  xmlns:config="http://www.suse.com/1.0/configs">
  <suse_register>
    <addons config:type="list">
      <addon>
<name>SLEMicro</name>
<reg_code>cc36aae1</reg_code>
      </addon>
    </addons>
  </suse_register>
</profile>
```



Note: Limitations

Currently flash disks are only scanned during installation or upgrade, but not when registering a running system.

10.6 Extension and Module Selection

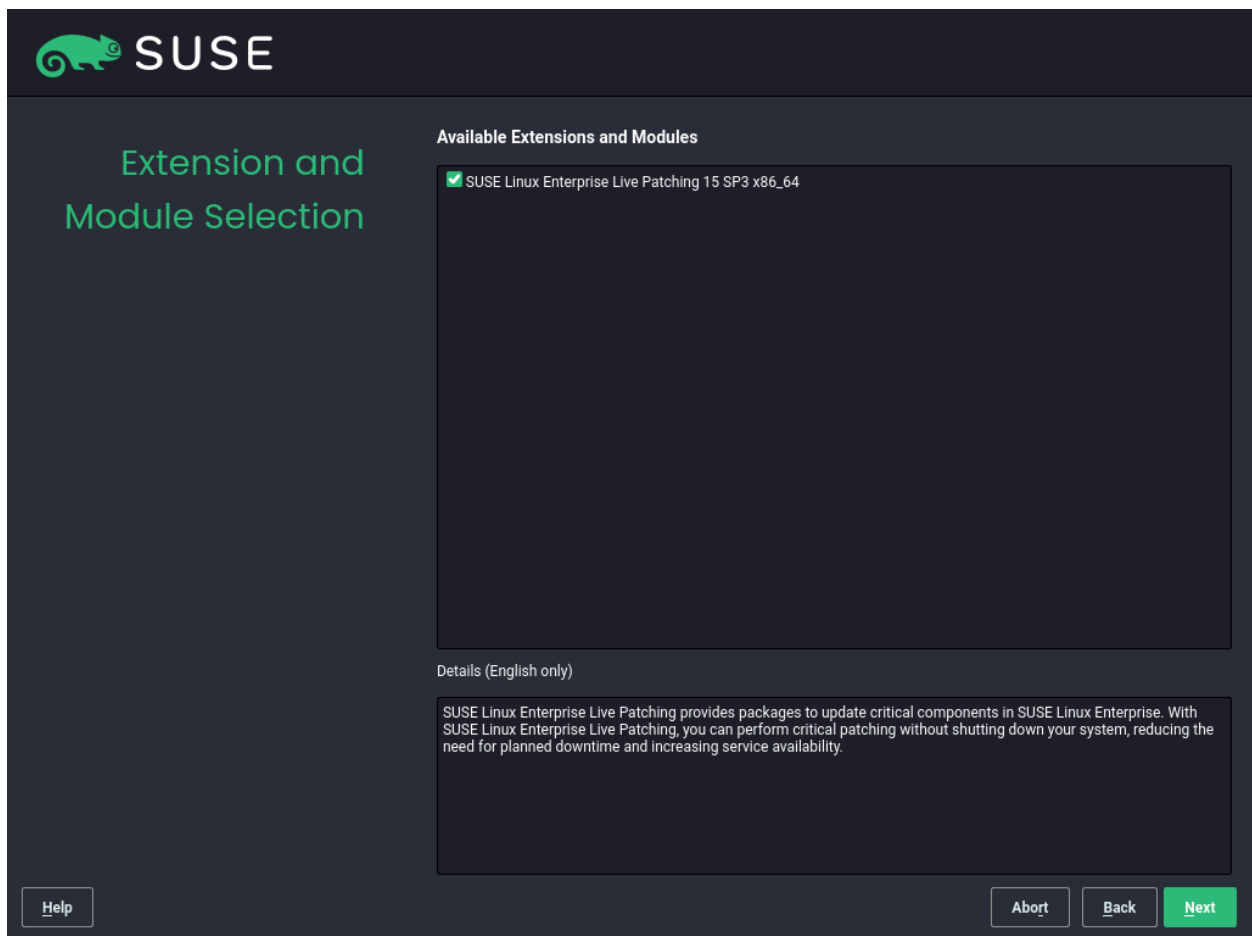


FIGURE 10.7: EXTENSIONS

SLE Micro currently offers the SUSE Linux Enterprise Live Patching extension that enables you to apply critical patches without rebooting your system. Bear in mind that you might need an additional subscription on top of the subscription for SLE Micro.

! Important: Post-installation configuration

If you enable the SUSE Linux Enterprise Live Patching extension, you need to configure your system as described in [Section 6.2.1, “Activating SUSE Linux Enterprise Live Patching”](#).

To enable the module, click the checkbox and then *Next* to proceed.



Note: Availability of the SUSE Linux Enterprise Live Patching extension

The SUSE Linux Enterprise Live Patching extension is available only for the x86 (except for the real-time kernel) and IBM Z architectures.

10.7 NTP Configuration

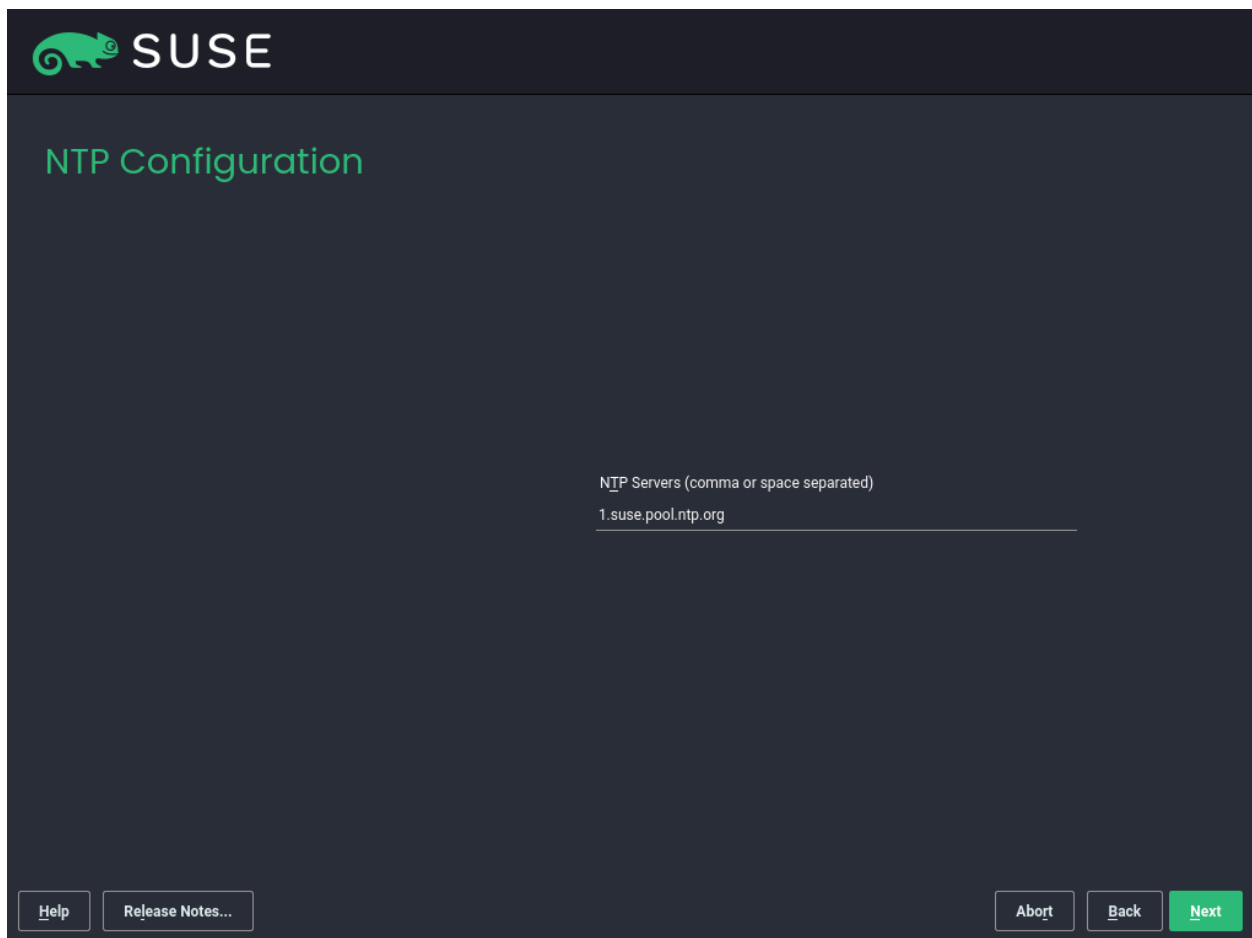


FIGURE 10.8: NTP CONFIGURATION

In order to keep time on your system properly synchronized, configure at least one NTP server. You can enter more NTP servers as a comma or space separated list.

10.8 Authentication for the System Administrator `root`

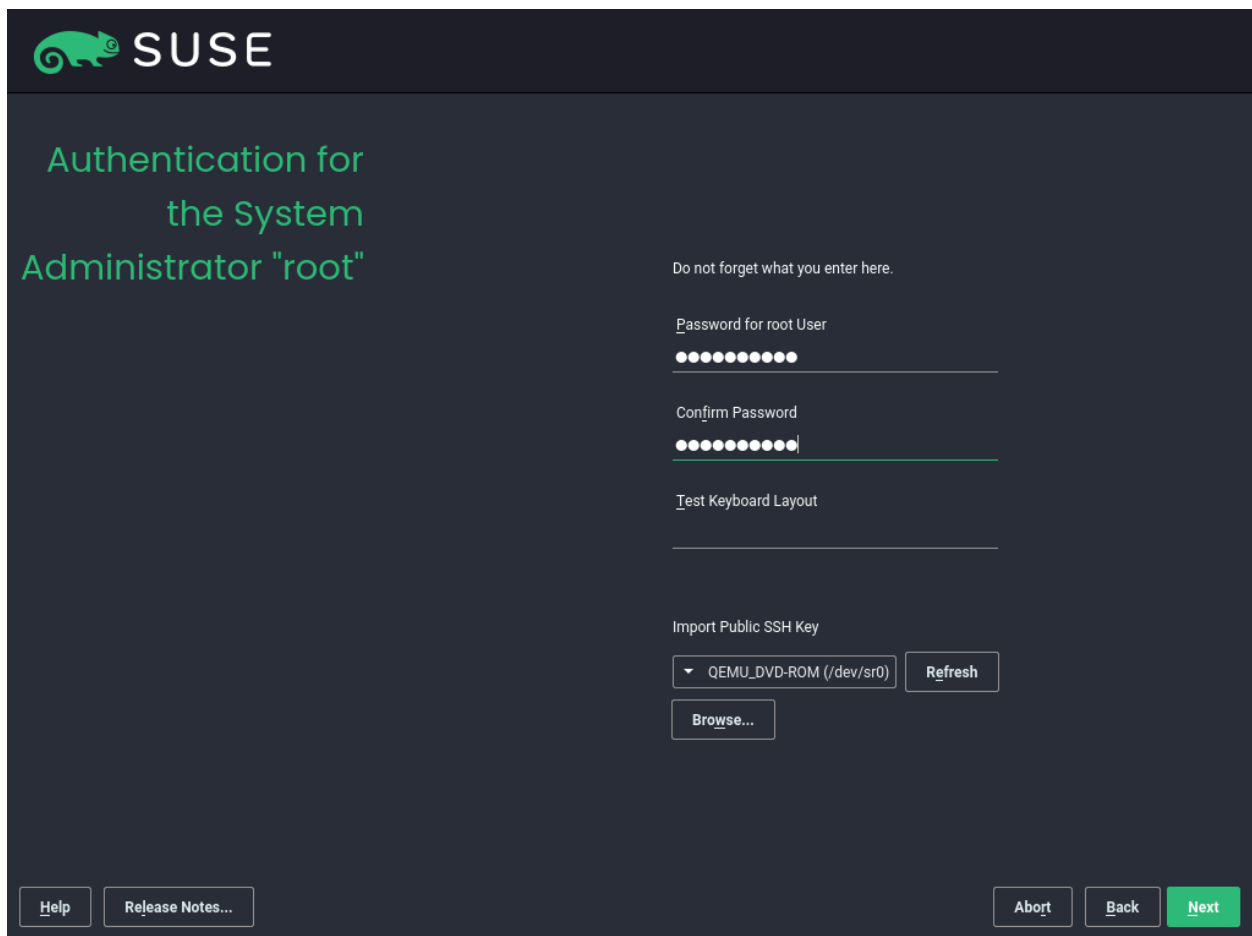


FIGURE 10.9: AUTHENTICATION FOR `root`

Configure a strong password for `root`. If your `root` password is randomly generated, use at least 10 characters. If you set your `root` password manually, use a longer password that includes a combination of uppercase and lowercase letters and numbers. The maximum length for passwords is 72 characters, and passwords are case-sensitive.

If you want to access the system remotely via SSH using a public key, import a key from a removable storage device or an existing partition. To do so click *Browse* and select the public SSH key.

Click *Next* to proceed to the next installation step.

10.9 Installation Settings

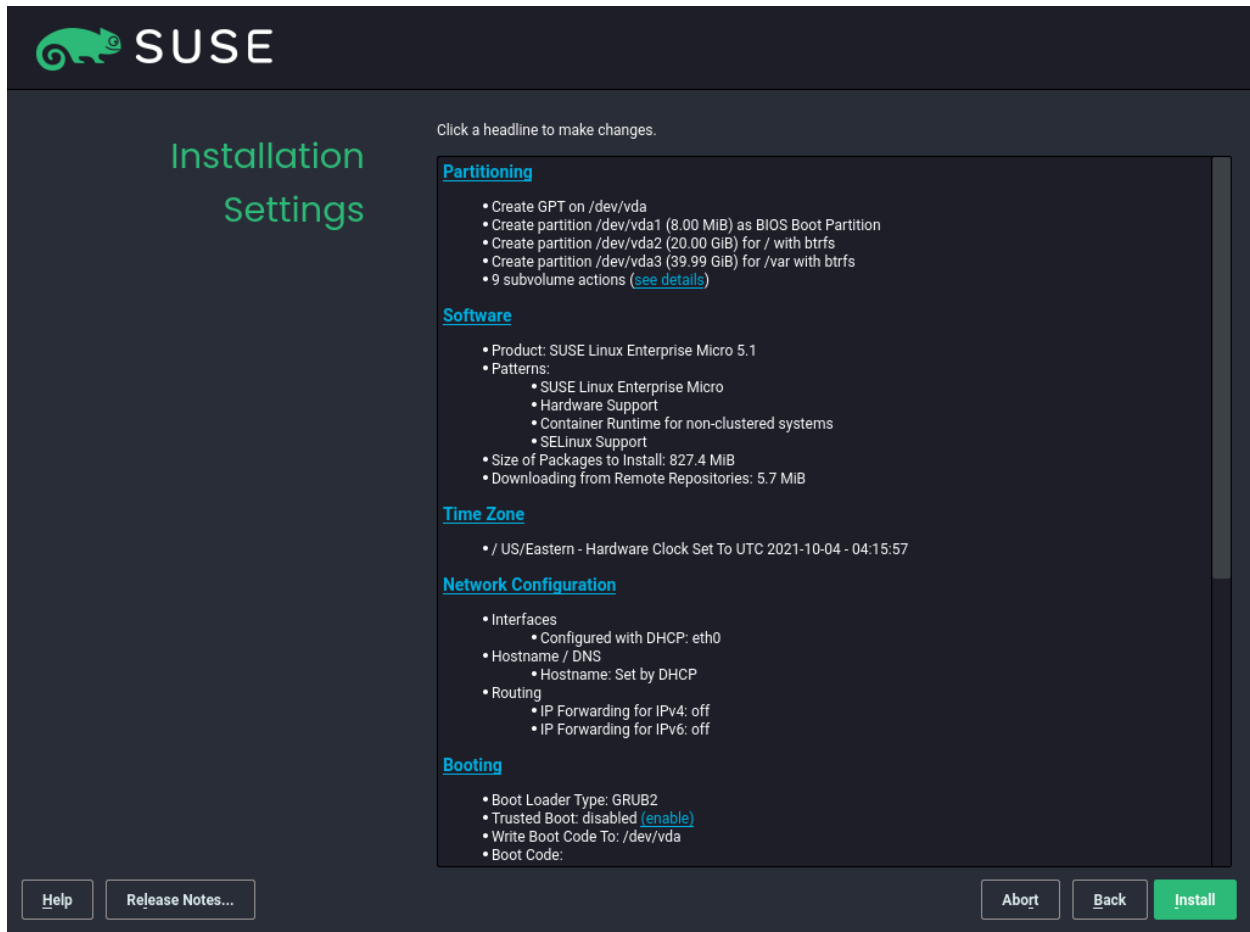


FIGURE 10.10: INSTALLATION SETTINGS

To access a particular setting, click the respective heading. Or some options can be directly changed on the screen by clicking the button next to the option.

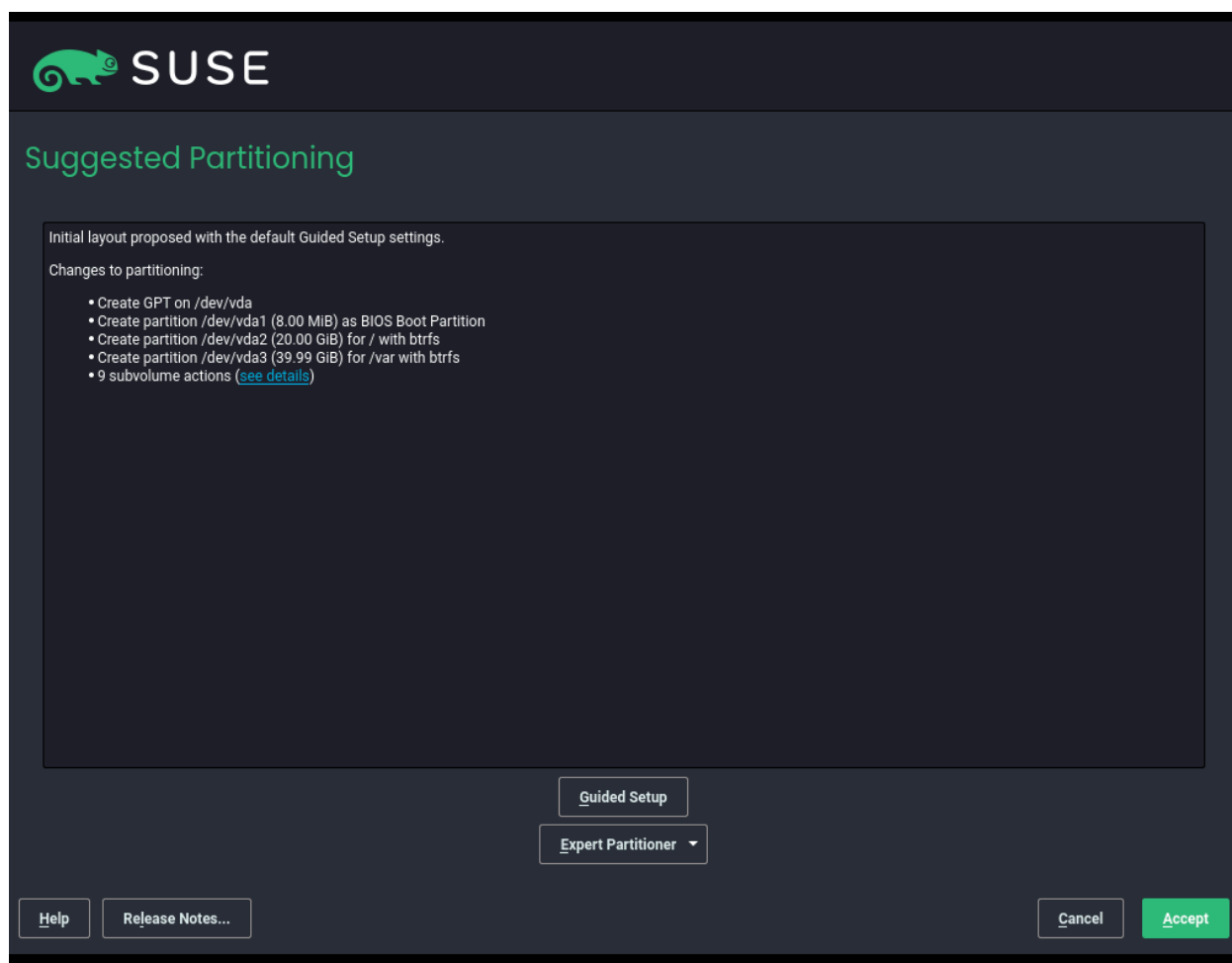


FIGURE 10.11: SUGGESTED PARTITIONING



Warning: The use of Btrfs and snapshots is mandatory.

SLE Micro requires Btrfs on the root partition with snapshots and Snapper enabled. Snapper is enabled by default—do not disable it afterwards.

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 12 GB, SUSE Linux Enterprise Micro by default enables file system snapshots. It is not recommended to use the root partition smaller than 12 GB, because it might cause issues when running SLE Micro.

SUSE Linux Enterprise Micro uses Snapper together with Btrfs for this feature. Btrfs needs to be set up with snapshots enabled for the root partition.

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

The installer will automatically create single snapshots during and immediately after the installation.



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 disk space 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 40 GB for the root partition.

Btrfs data volumes

Using Btrfs for data volumes is supported on SUSE Linux Enterprise Micro 5.1. 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 formatting the root partition to 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.

IBM Z: Using minidisks in z/VM

If SUSE Linux Enterprise Micro is installed on minidisks in z/VM, which reside on the same physical disk, the access path of the minidisks (`/dev/disk/by-id/`) is not unique. This is because it represents the ID of the physical disk. If two or more minidisks are on the same physical disk, they all have the same ID.

To avoid problems when mounting minidisks, always mount them either *by path* or *by UUID*.

IBM Z: LVM root file system

If you configure the system with a root file system on LVM or software RAID array, you must place `/boot` on a separate, non-LVM or non-RAID partition, otherwise the system will fail to boot. The recommended size for such a partition is 500 MB and the recommended file system is Ext4.

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
- Intel Virtual RAID on CPU (Intel VROC, see <https://www.intel.com/content/www/us/en/support/articles/000024498/memory-and-storage/ssd-software.html> for more details)

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.

In case you need to adjust the partitioning scheme, click the *Partitioning* menu to open the *Suggested partitioning* dialog box.

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:

Accept

Click *Accept* to accept the proposal without any changes and return to the *Installation Settings* screen.

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. Afterward 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. For details, refer to [Section 10.9.1.1, "Expert Partitioner"](#).

10.9.1.1 Expert Partitioner

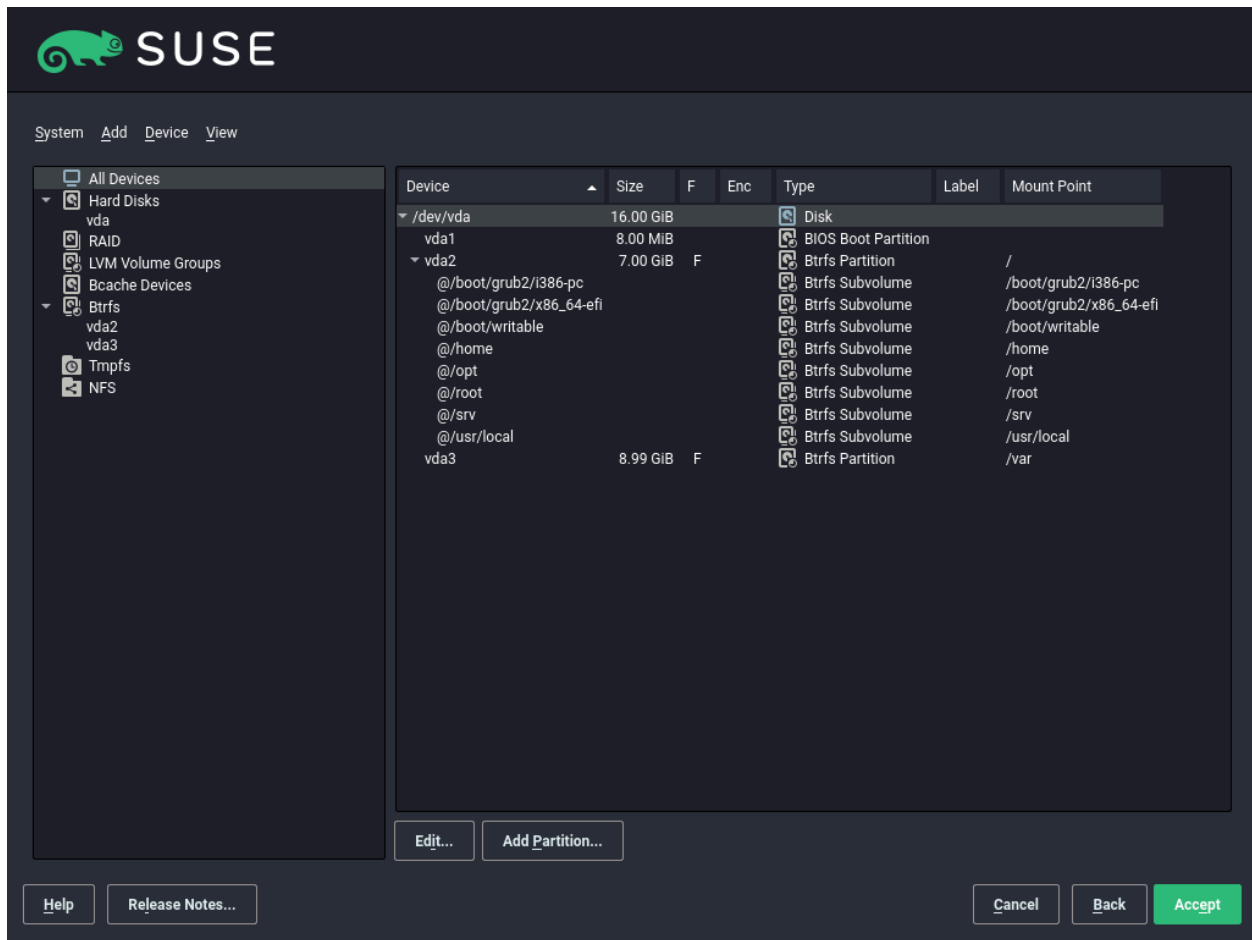


FIGURE 10.12: EXPERT PARTITIONER

Expert partitioner enables you to set up logical volume management (LVM), configure software RAID and device mapping (DM), encrypt partitions, mount NFS shares and manage tmpfs volumes. To fine-tune settings such as the subvolume and snapshot handling for each Btrfs partition, choose *Btrfs*.

All existing or suggested partitions on all connected hard disks are displayed in the left part of the *Expert Partitioner* dialog. Entire hard disks are listed as devices without numbers, such as `/dev/sda` (or `/dev/dasda`). Partitions are listed as parts of these devices, such as `/dev/sda1` (or `/dev/dasda1`, respectively). The size, type, encryption status, file system, and mount point of the hard disks and their partitions are also displayed. The mount point describes where the partition appears in the Linux file system tree.

10.9.1.1.1 Partition tables

SUSE Linux Enterprise Micro allows to use and create different *partition tables*. In some cases the partition table is called *disk label*. The partition table is important to the boot process of your computer. To boot your machine from a partition in a newly created partition table, make sure that the table format is supported by the firmware.

To change the partition table, click the relevant disk name in the left part and choose *Device > Create New Partition Table*. You can create the following partition tables:

Master boot record

The *master boot record (MBR)* is the legacy partition table used on IBM PCs. It is sometimes also called an *MS-DOS* partition table. The MBR only supports four primary partitions. If the disk already has an MBR, SUSE Linux Enterprise Micro allows you to create additional partitions in it which can be used as the installation target.

The limit of four partitions can be overcome by creating an *extended partition*. The extended partition itself is a primary partition and can contain more *logical partitions*.

GPT partition table

UEFI computers use a *GUID Partition Table (GPT)* by default. SUSE Linux Enterprise Micro will create a GPT on a disk if no other partition table exists.

Old BIOS firmware does not support booting from GPT partitions.

You need a GPT partition table to use more than four primary partitions, UEFI Secure Boot or use disks larger than 2 TB.

10.9.1.1.2 Creating partitions

The expert partitioner enables you to add partitions. Bear in mind that the root file system must be formatted to Btrfs and snapshots must be enabled.

The procedure below creates a Btrfs partition with enabled snapshots.

1. Select the desired hard disk in the left part and click *Add Partition*.
2. Define size of the partition or define the region of disk for the partition. Proceed with *Next*

Add Partition on /dev/vda

New Partition Size

Maximum Size (59.99 GiB)

Custom Size

Size

20 GiB

Custom Region

Start Block

2048

End Block

125829086

Help

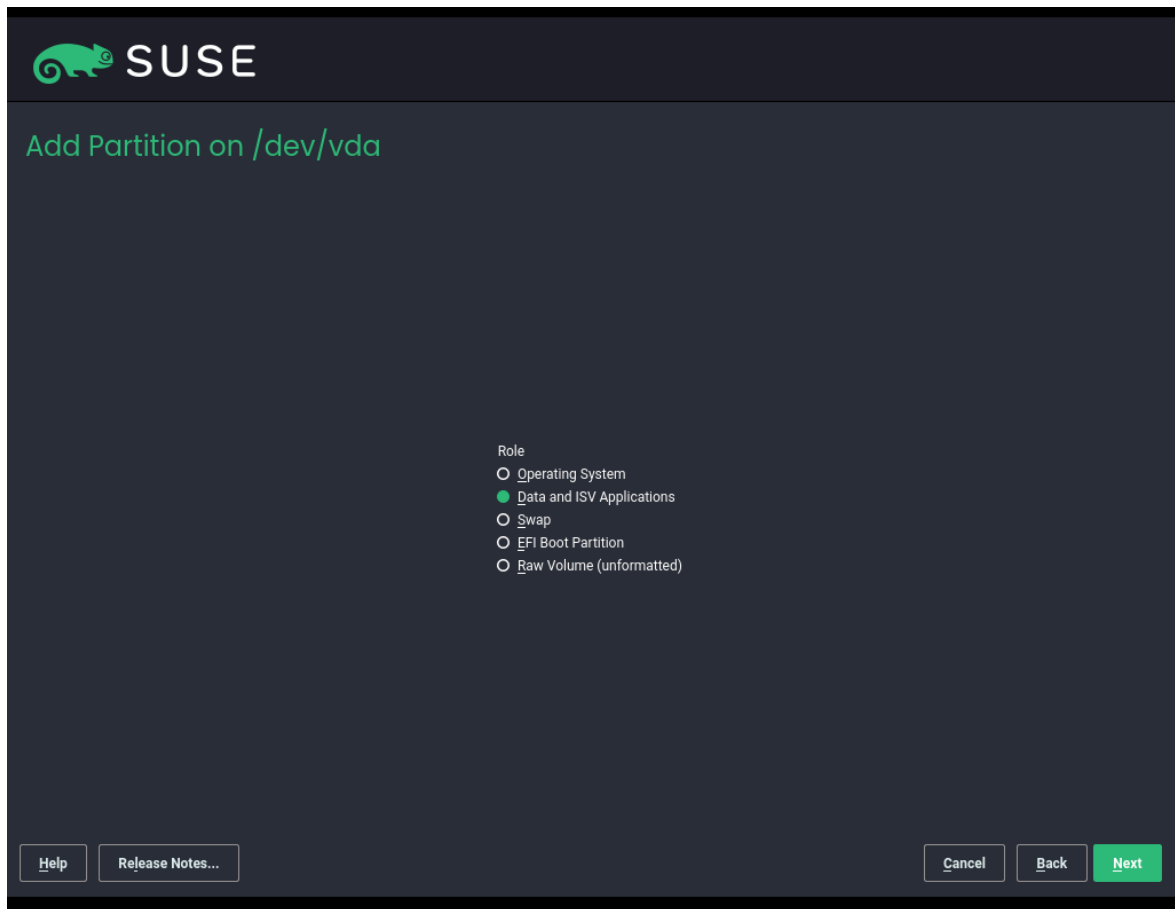
Release Notes...

Cancel

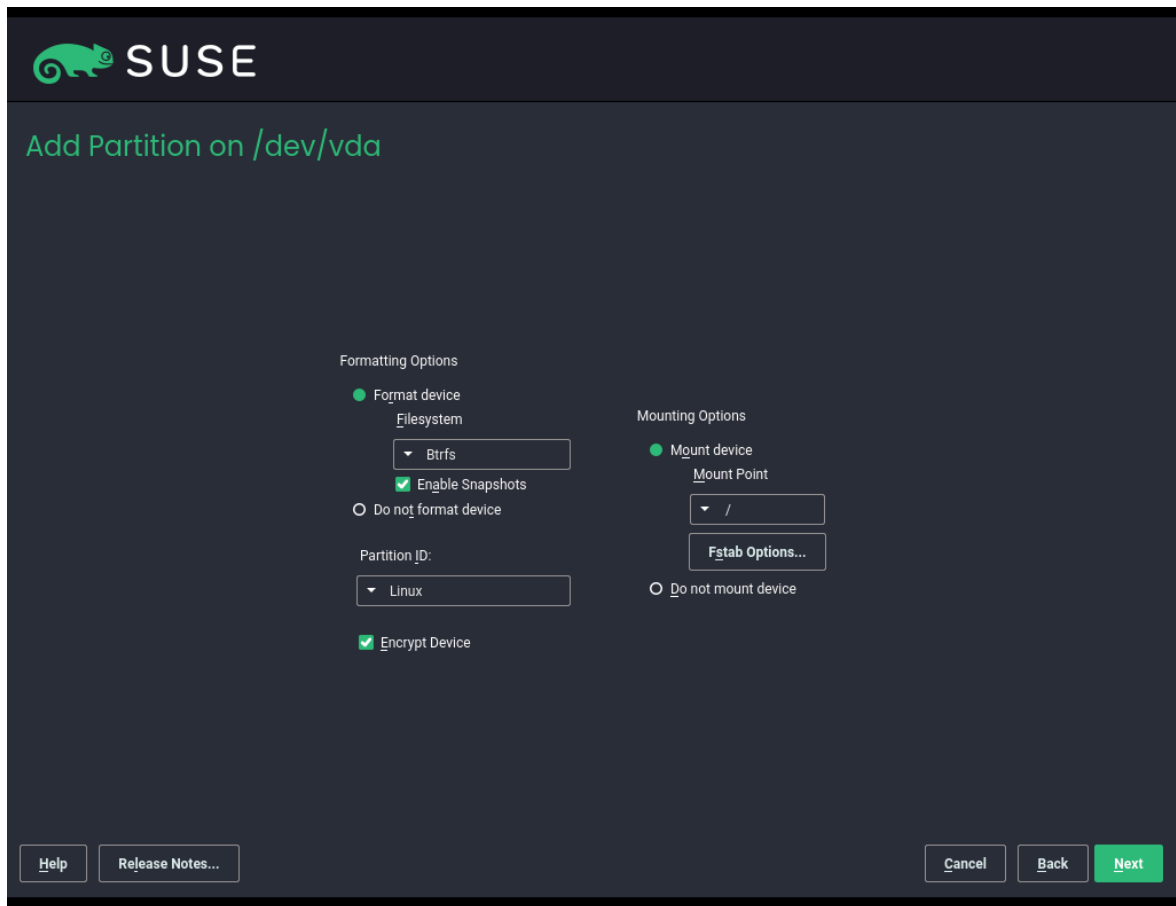
Back

Next

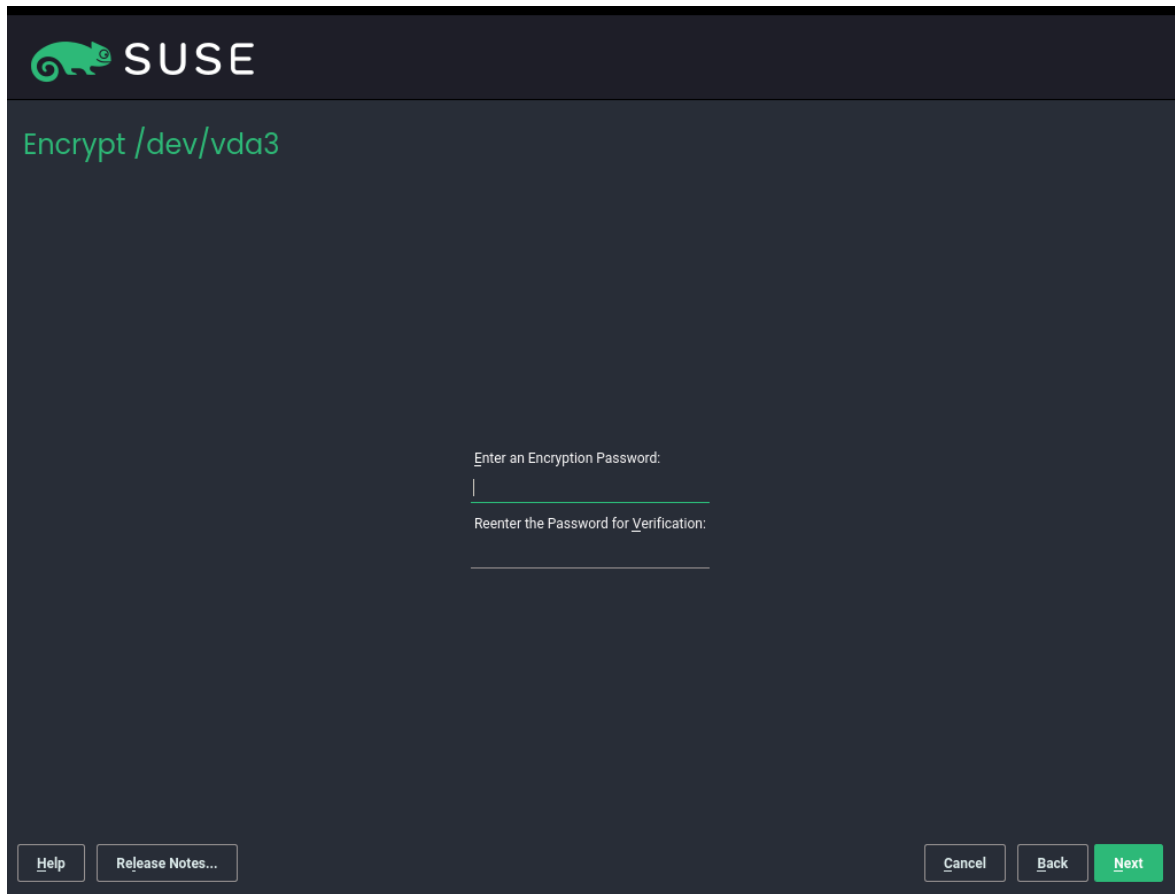
3. Select a role:



4. Format and mount the partition as needed and proceed with *Next*:



5. (Optional) If you chose to encrypt the partition, enter the encryption password and complete the process with *Next*:

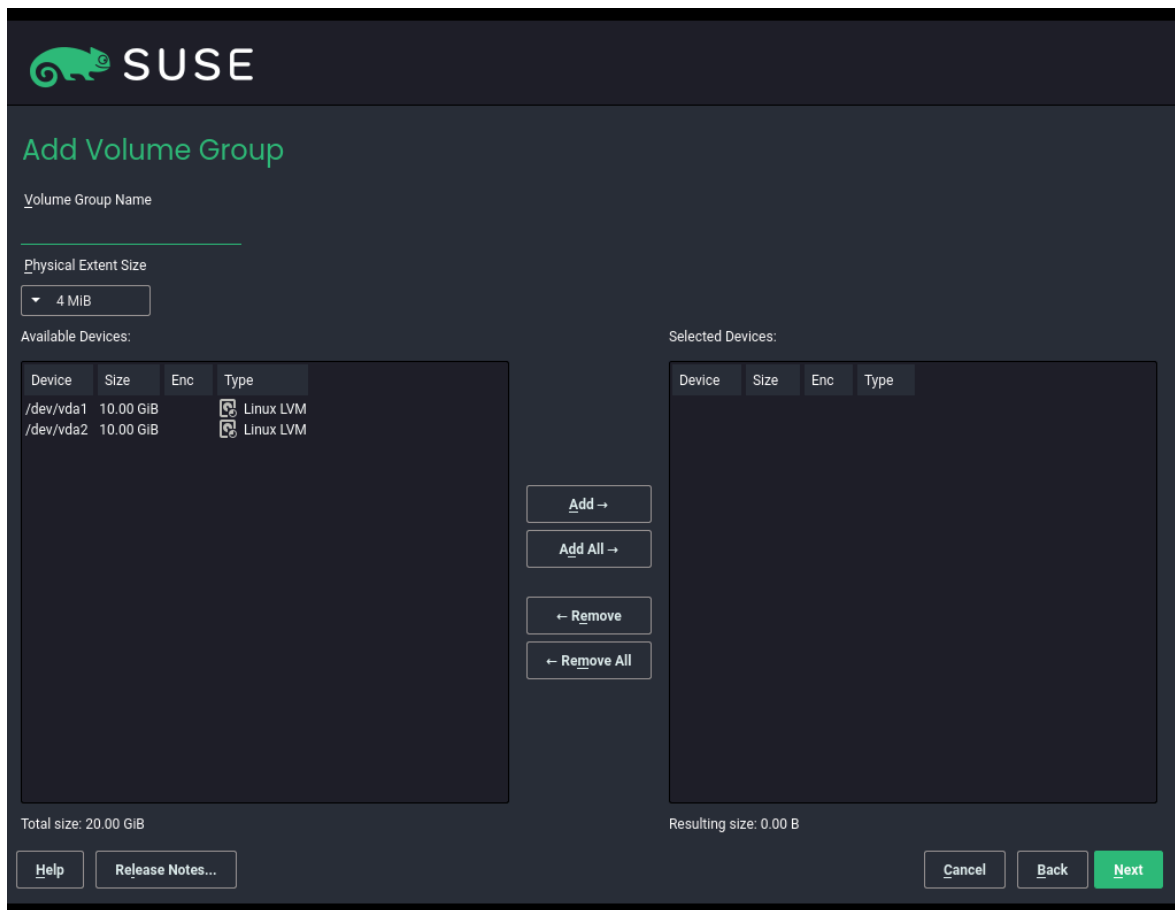


10.9.1.1.3 Creating volume groups

To create a volume group follow these steps:

PROCEDURE 10.1: CREATING VOLUME GROUPS

1. Create partitions with the following attributes:
 - the partitions are not mounted
 - *PartitionID* is Linux LVM
2. Click *LVM Volume Groups* > *Add Volume Group*
3. Select partitions to be added to the volume group and click *Add*. Name the volume group, select the *Physical Extent Size* and click *Next* to proceed further.



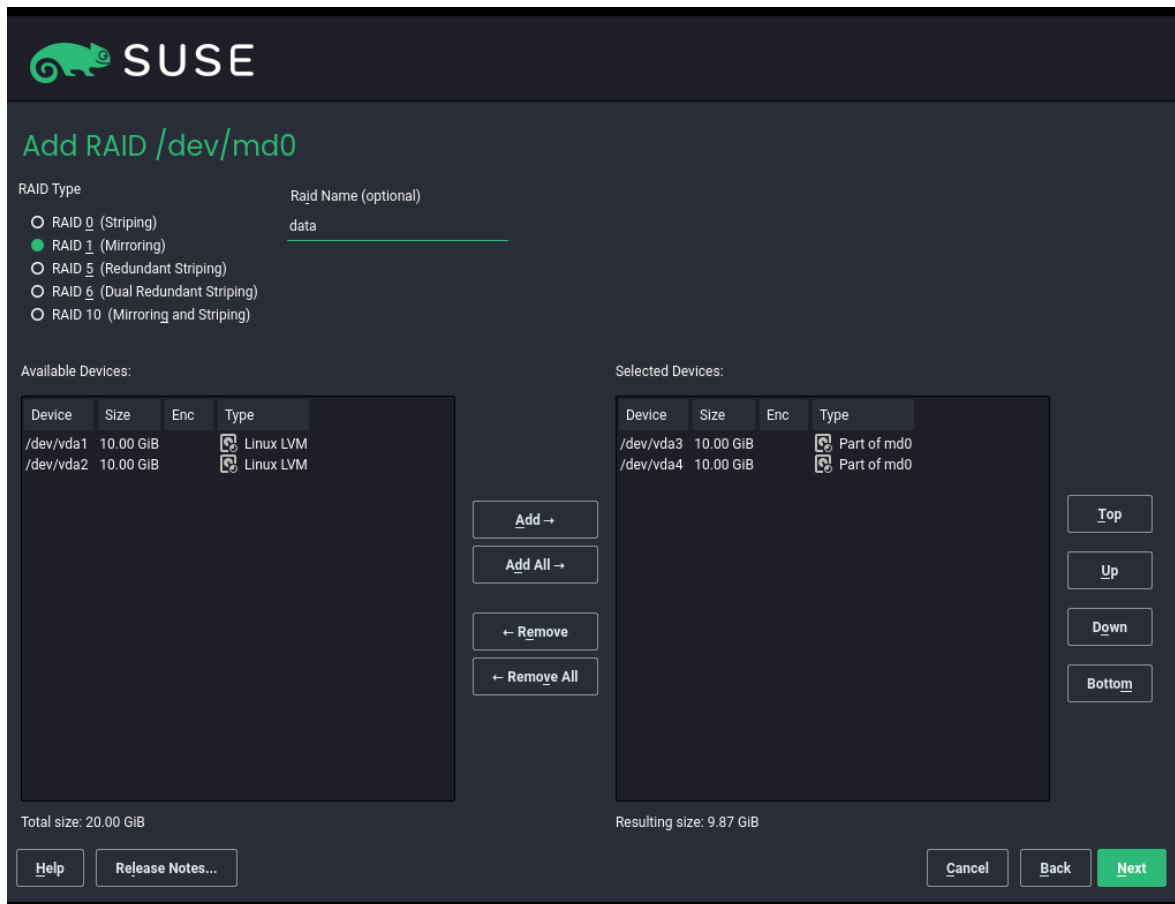
10.9.1.1.4 Creating RAIDs

SLE Micro supports the following RAID levels: 0, 1, 5, 6 and 10. To create a RAID follow proceed as follows:

PROCEDURE 10.2: CREATING RAID

1. Create partitions (the count of partitions depend on the RAID level) with these parameters:
 - The partitions have the *Raw Volume* role assigned.
 - The partitions are not formatted to any file system.
 - The partitions are not mounted.
 - The partitions have the Linux RAID PartitionID.
2. Click *RAID* in the left pane and then click *Add RAID*. The *Add RAID* dialog box opens.

3. Choose the partitions and add them to the RAID. Select RAID level and optionally you can name the RAID. Proceed with *Next*.



4. Select the *Chunk Size*. The default value is usually sufficient. Click *Next*.
5. In the *Device Overview* select the created RAID and click *Edit*.
6. Select a role of the RAID and click *Next*.
7. Format and mount the device and optionally you can select that the RAID will be encrypted.

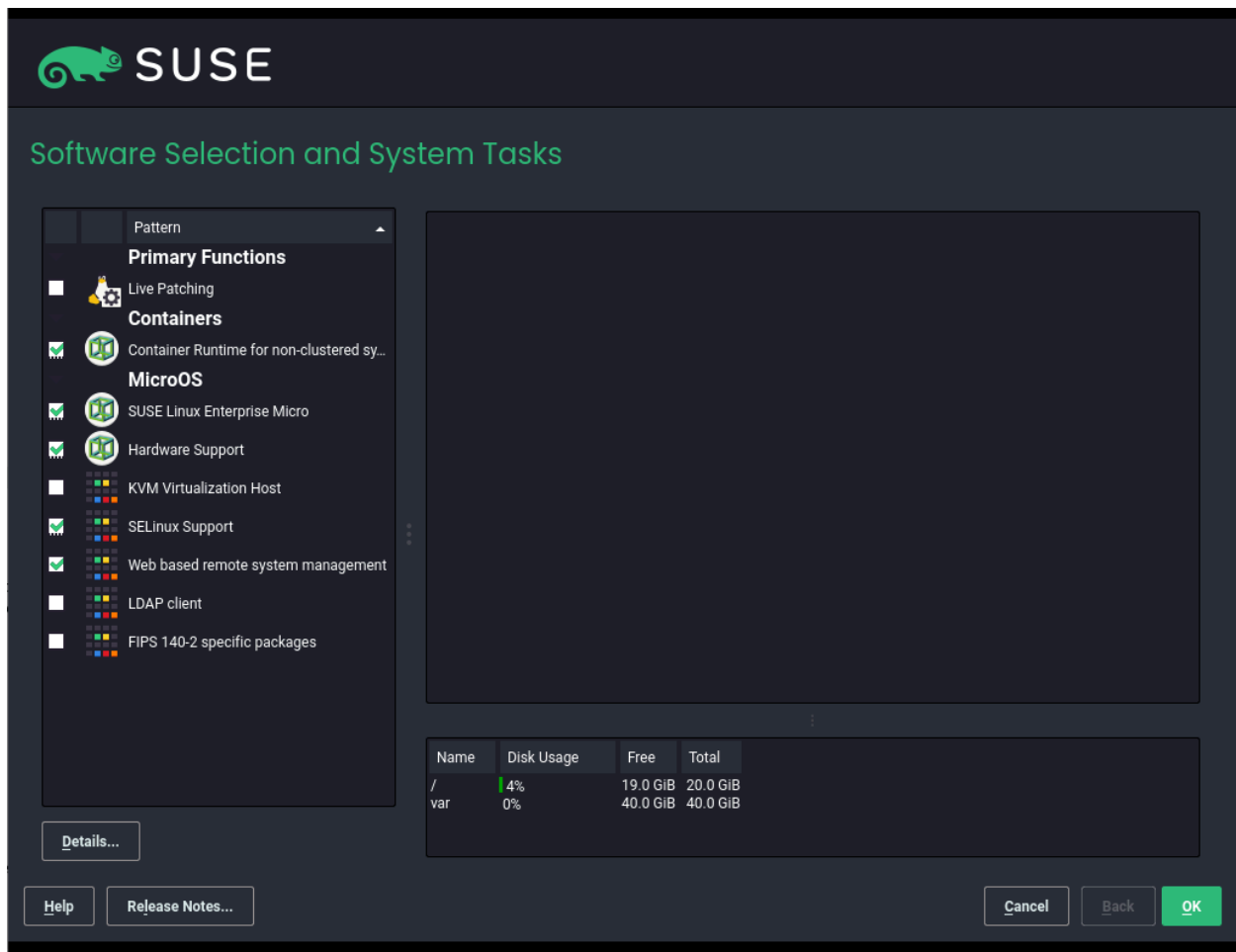


FIGURE 10.13: SOFTWARE CONFIGURATION

SUSE Linux Enterprise Micro contains several software patterns for various application purposes. 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.

In this menu you can select the Web based remote system management pattern that will install Cockpit system. Cockpit is a web monitoring tool that enables you to administer your system. For details, refer to *Article "Administration Guide", Section 4 "SLE Micro administration using Cockpit"*.

Here you can also select the KVM Virtualization Host pattern to install packages required to run SLE Micro as a KVM host server (Xen is not supported). However, you should consider limitations of SLE Micro running as a KVM host server, for details refer to [virtualization limits and support \(https://documentation.suse.com/sles/15-SP3/html/SLES-all/cha-virt-support.html\)](https://documentation.suse.com/sles/15-SP3/html/SLES-all/cha-virt-support.html).

Each pattern contains several software packages needed for specific functions (for example Podman). For a more detailed selection based on software packages to install, select *Details* to switch to the YaST Software Manager.

10.9.3 Timezone

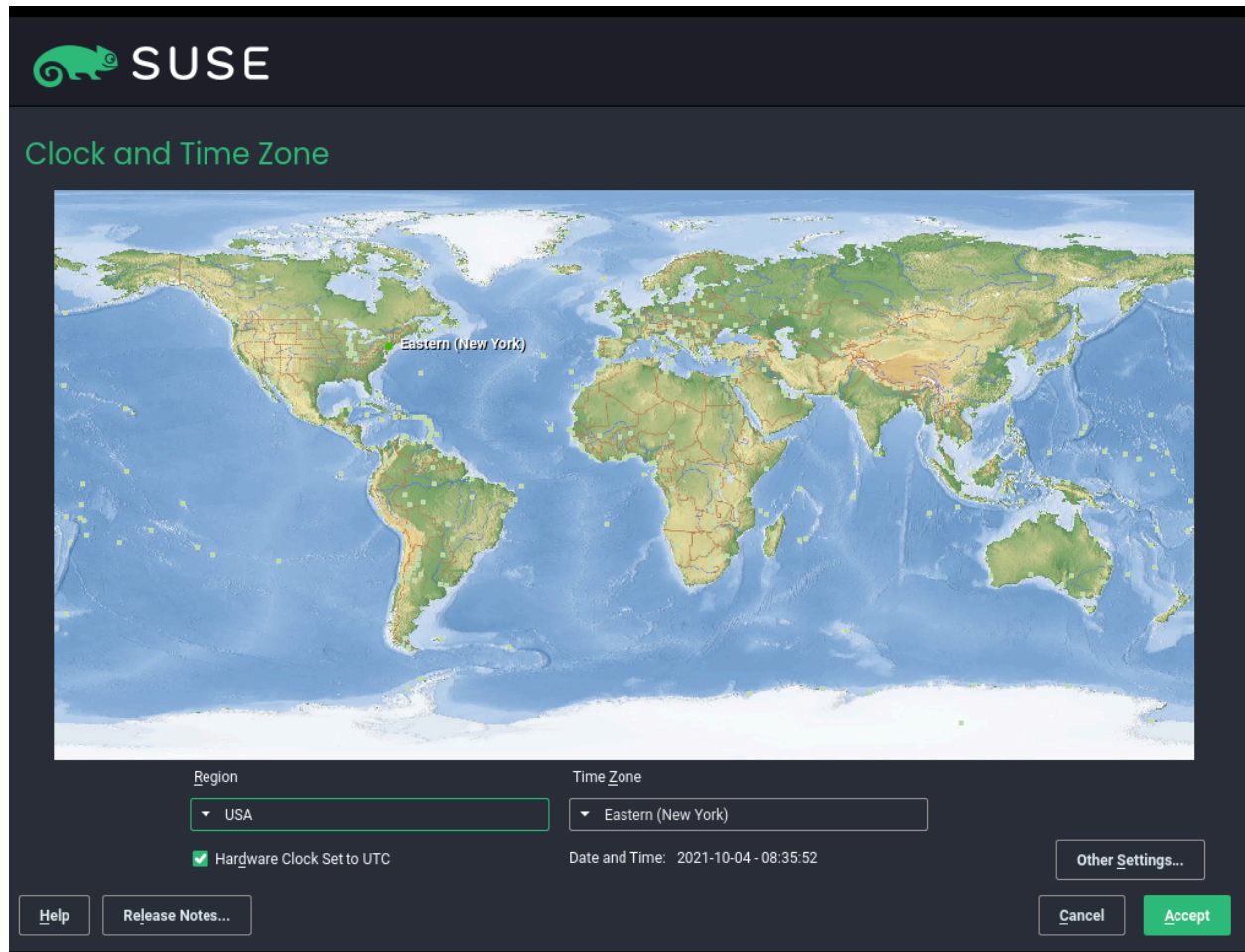


FIGURE 10.14: TIME ZONE CONFIGURATION

By default, the time is synchronized by using the NTP servers you provided in the previous steps of the installation procedure. You can select the region and time zone either by clicking a particular place on the map or by selecting a region and time zone in the drop-down menus.

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, it is strongly recommended to *always* set the hardware clock to UTC.

The button *Other settings* enables you to set the date and time manually or configure NTP servers synchronization.

If you want to set the time and date manually, click the *Other settings* button and select *Manually*.

Note: Time cannot be changed on IBM Z

Since the operating system is not allowed to change time and date directly, the *Other Settings* option is not available on IBM Z.

10.9.4 Network Configuration

Network is automatically configured at the beginning of the installation process, but if it is necessary you can change the configuration by clicking *Network Configuration*. A dialog box opens, for details refer to [Section 10.2, "Network settings"](#).

10.9.5 Booting

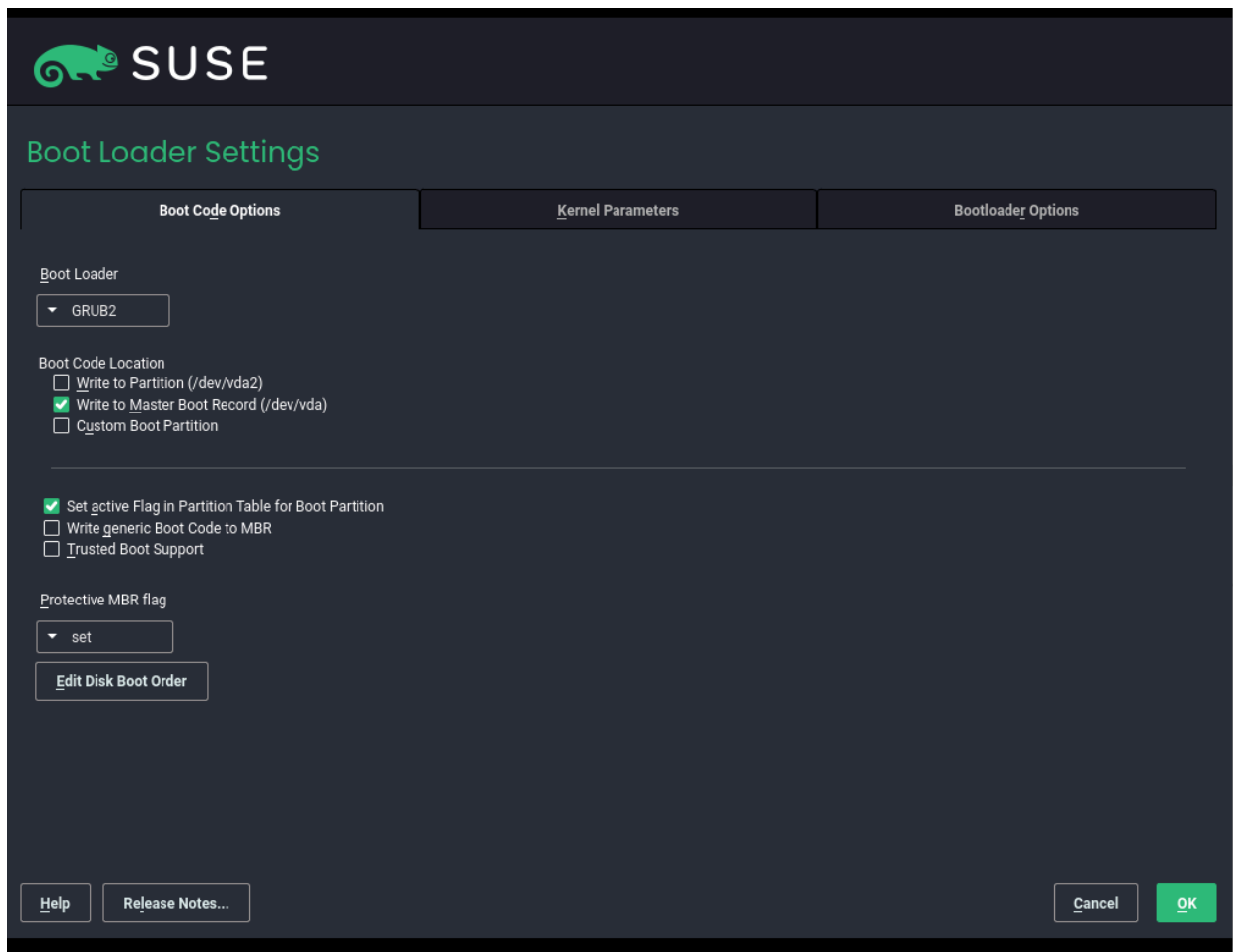


FIGURE 10.15: BOOTING

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, SUSE Linux Enterprise Micro 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.

! Important: Software RAID 1

Booting a configuration where `/boot` resides on a software RAID 1 device is supported, but it requires installing 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.

10.9.6 Kdump

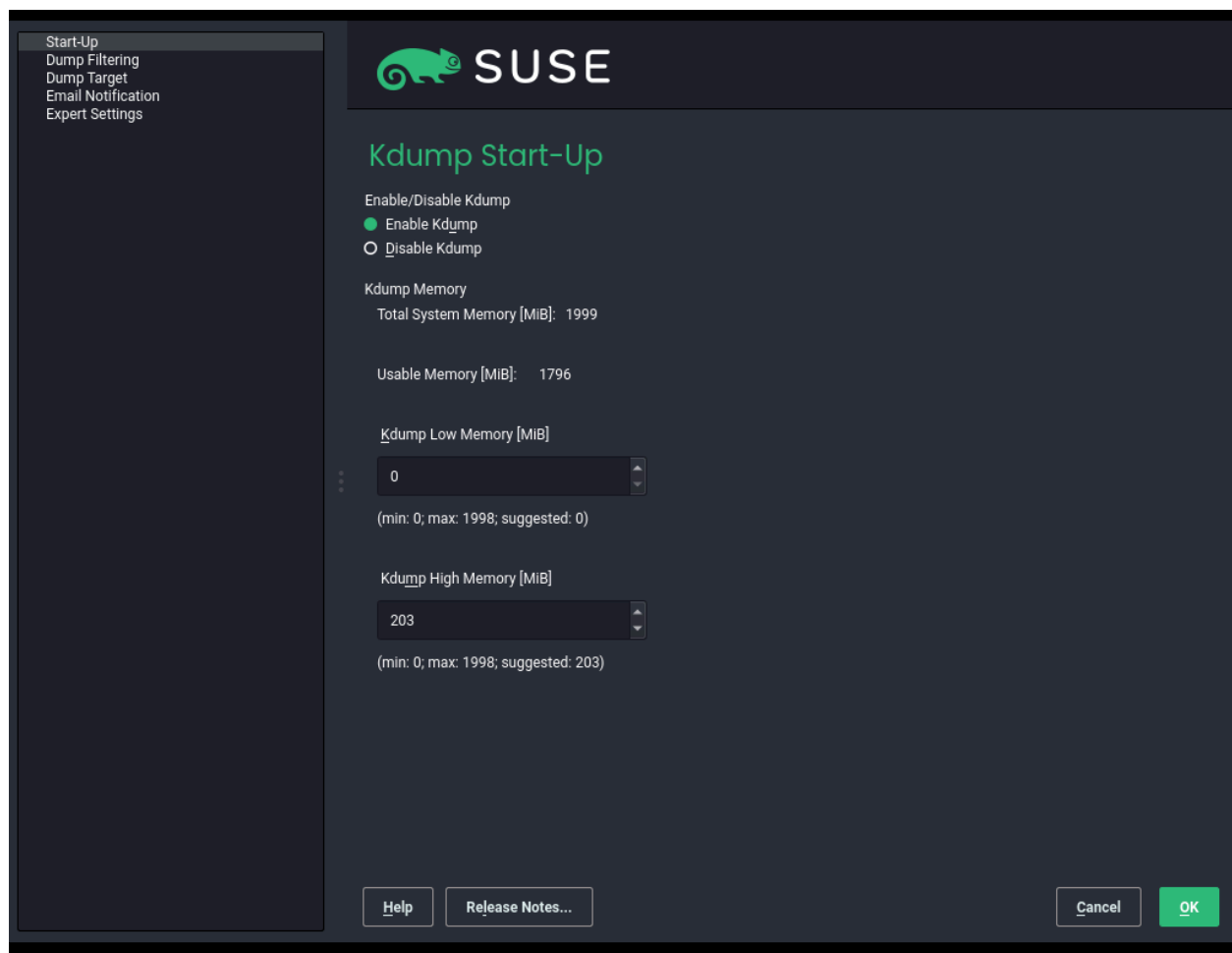


FIGURE 10.16: KDUMP CONFIGURATION

Using Kdump, you can save a dump of the kernel (in case of a crash) to analyze what went wrong. By default, Kdump is enabled. By clicking *Kdump* you open a dialog box for configuring Kdump.

Start-Up

Here you can disable Kdump and configure the amount of memory reserved for Kdump. Usually you do not have to change the prefilled values.

Dump Filtering

Dump filtering enables you to select which pages will be included in the Kdump, and to define the format of of Kdump.

Dump Target

You can select a local directory or you can save KDump to a remote location. If you prefer a remote location, you also need to configure connection details according to the respective protocol.

Email Notifications

To receive email notifications if an event occurs, specify an email address.

Expert Settings

This option enables you to define command-line parameters, custom kernel dump and other advanced settings related to Kdump.

10.9.7 System

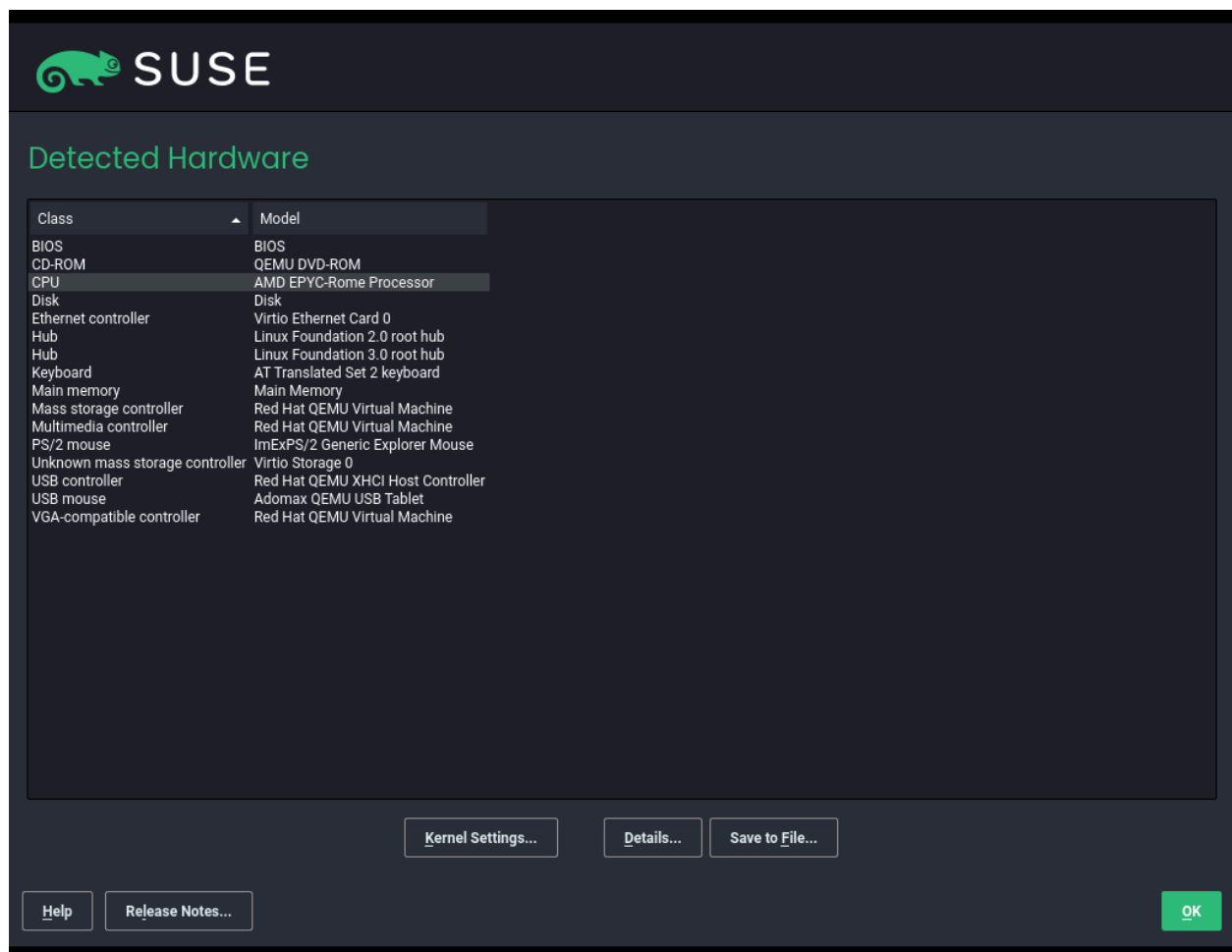


FIGURE 10.17: SYSTEM OVERVIEW

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

Activating the *Enable SysRq Keys* item, 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.

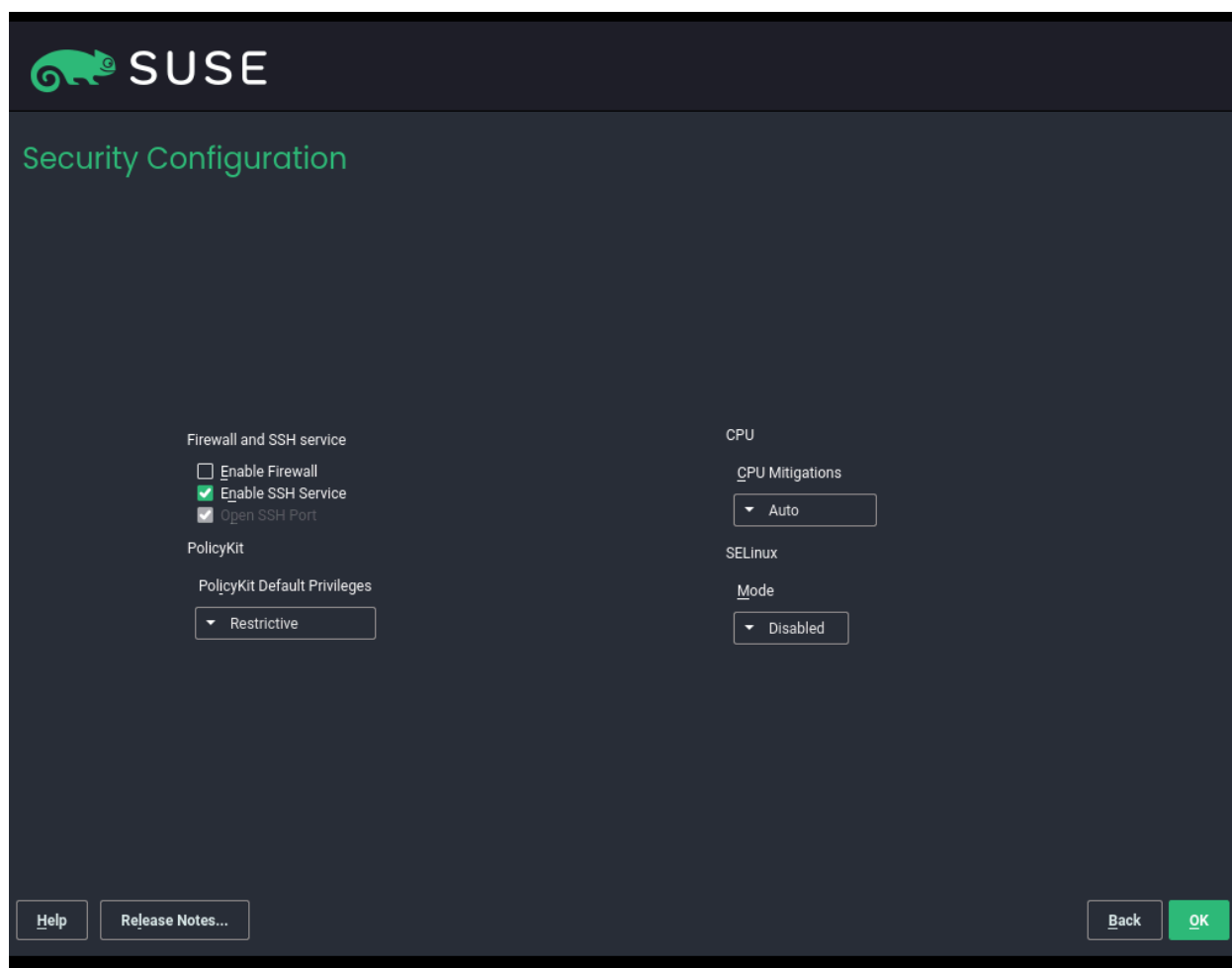


FIGURE 10.18: SECURITY CONFIGURATION

! Important: Do not use firewall along with Podman

Using firewall along with Podman may cause missing Podman related firewall rules after reloading the `firewalld` service. Therefore, it is recommended to keep the firewall in its default setting—disabled if you intend to use Podman.

You can enable the firewall or disable the SSH service directly by clicking the respective button. Clicking on the button next to *CPU mitigations* opens the *Boot Loader Settings* dialog box, where you can change kernel parameters including the `CPU mitigations` configuration.

The *CPU Mitigations* refer to kernel boot command line parameters for software mitigations that have been deployed to prevent CPU side-channel attacks. You can configure the following values:

Auto

All CPU side channel mitigations are enabled as they are detected based on the CPU type. The auto-detection handles both unaffected older CPUs and unaffected newly released CPUs and transparently disables mitigations. This options leave SMT enabled.

Off

All CPU side channel mitigations are disabled. While this option gives the higher performance, it also bears the highest risk. Do not use this setting where there is a risk of untrusted code.

Auto + No Smt

All CPU side channel mitigations are enabled as they are detected based on the CPU type. Additionally the symmetric multi-threading of the CPU is disabled if necessary, for instance to mitigate the L1 Terminal Fault side channel issue.

Manually

CPU mitigations are detected manually.

By default, the firewall is disabled. Click *enable* to change the default.

The SSH service is enabled by default. Click *disable* to change the setting. If you disable the SSH service, you will not be able to login to your system remotely. The SSH port (22) is open by default.

The default SELinux option is *disabled*. You can change the value by clicking *Security* and selecting another option in the *Mode* menu.

In the *Security* dialog box, you can also select PolicyKit privileges in the *PolicyKit Default Privileges* dropdown menu.

11 Remote installation

The installation of SUSE® Linux Enterprise Micro can be fully performed over the network. This chapter describes how to provide the required environment for booting, installing and controlling the installation via the network.

11.1 Overview

For a remote installation you need to consider how to boot, how to control the installation, and the source of the installation data. All available options can be combined with each other, if they are available for your hardware platform.

Boot method

Depending on the hardware, several options for booting a system exist. Common options are DVD, USB drive or PXE boot. For more information about your platform, refer to *Part I, "Installation preparation"*.

Data source

Most commonly, DVDs or USB drives are used as a source for installing SUSE Linux Enterprise Micro. Alternatively, installation servers can be used. In this case, use the `install` boot parameter to specify the source. For details, refer to *Section 9.3.3, "Specifying the installation source"*.

Controlling the installation

Instead of using a keyboard and monitor directly attached to the target machine, the installation can be controlled via SSH, VNC, or by using the serial console of a machine. This is described in the sections *Section 11.3, "Monitoring installation via VNC"*, *Section 11.4, "Monitoring installation via SSH"* and *Section 11.5, "Monitoring installation via serial console"*. Instead of manually controlling the installation, AutoYaST can be used for fully automating the installation process. For details, refer to *Book "AutoYaST Guide"*.

11.2 Scenarios for remote installation

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

11.2.1 Installation from source media via VNC

This type of installation still requires some degree of physical access to the target system to boot for installation. The installation is controlled by a remote workstation using VNC to connect to the installation program. User interaction is required as with the manual installation in [Chapter 10, Installation steps](#).

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

- Target system with working network connection.
- Controlling system with working network connection and VNC viewer software or JavaScript-enabled browser (Firefox, Chromium, Internet Explorer, Opera, etc.).
- Installation DVD or USB flash drive.

To perform this kind of installation, proceed as follows:



Important: No access via VNC after the installation is complete.

After the installation of SLE Micro is finished, you will no longer be able to log in to the system using VNC.

1. Boot the target system using the installation medium (USB flash drive) of the SUSE Linux Enterprise Micro media kit.
2. When the boot screen of the target system appears, use the boot parameters prompt to set the VNC options and, if required, the static network configuration. For information about boot parameters, see [Chapter 9, Boot parameters](#).

- a. Boot parameters for a static network configuration:

```
netdevice=NETDEVICE hostip=IP_ADDRESS netmask=NETMASK gateway=IP_GATEWAY vnc=1
VNCPassword=PASSWORD
```

- b. Boot parameters for a dynamic (DHCP) network configuration:

```
vnc=1 VNCPassword=PASSWORD
```

3. The target system boots to a text-based environment, giving the network address and display number under which the graphical installation environment can be addressed by any VNC viewer application or browser. VNC installations announce themselves over OpenSLP and if the firewall settings permit. They can be found using `slptool` as described in [Section 11.3.1, “Preparing for VNC installation”](#).
4. On the controlling workstation, open a VNC viewing application or Web browser and connect to the target system as described in [Section 11.3, “Monitoring installation via VNC”](#).
5. Perform the installation as described in [Chapter 10, Installation steps](#).

11.2.2 Installation from network via VNC

This type of installation does not require a direct interaction with the target machine. The system is booted via PXE and the installation data is fetched from a server.

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

- At least one machine that can be used for installing a DHCP, NFS, HTTP, FTP, TFTP, or SMB server.
- Target system capable of PXE boot, networking, and Wake on LAN, plugged in and connected to the network.
- Controlling system with working network connection and VNC viewer software or JavaScript-enabled browser (Firefox, Chromium, Microsoft Edge, Opera, etc.).

To perform this type of installation, proceed as follows:

 **Important:** No access via VNC after the installation is complete.

After the installation of SLE Micro is finished, you will no longer be able to log in to the system using VNC.

1. Set up the server that contains the installation data.
2. Set up a DHCP and TFTP server for the network. Add the required boot parameters to enable the VNC server.
3. Enable PXE boot in the target machine firmware.
4. Initiate the boot process of the target system using Wake on LAN.
5. On the controlling workstation, open a VNC viewing application or Web browser and connect to the target system.
6. Perform the installation as described in [Chapter 10, Installation steps](#).

11.2.3 Installation from source media via SSH

This type of installation still requires some degree of physical access to the target system to boot for installation and to determine the IP address of the installation target. The installation itself is entirely controlled from a remote workstation using SSH to connect to the installer. User interaction is required as with the regular installation described in [Chapter 10, Installation steps](#).

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

- Target system with working network connection.
- Controlling system with working network connection and working SSH client software.
- Installation DVD or USB flash drive.

To perform this kind of installation, proceed as follows:

1. Set up the installation target and installation server.

2. Boot the target system using the installation medium (USB flash drive) of the SUSE Linux Enterprise Micro media kit.
3. When the boot screen of the target system appears, use the boot parameters prompt to set the SSH options and, if required, the static network configuration. For information about boot parameters, see [Chapter 9, Boot parameters](#).
 - a. Boot parameters for a static network configuration:

```
netdevice=NETDEVICE hostip=IP_ADDRESS netmask=NETMASK gateway=IP_GATEWAY ssh=1
ssh.password=PASSWORD
```
 - b. Boot parameters for a dynamic (DHCP) network configuration:

```
ssh=1 ssh.password=PASSWORD
```
4. The target system boots to a text-based environment, giving the network address under which the graphical installation environment can be addressed by any SSH client.
5. On the controlling workstation, open a terminal window and connect to the target system as described in [Section 11.4.2, “Connecting to the installation program”](#).
6. Perform the installation as described in [Chapter 10, Installation steps](#).
7. Reconnect to the target system after it reboots for the initial system configuration.

11.2.4 Installation from network via SSH

This type of installation does not require a direct interaction with the target machine. The system is booted via PXE and the installation data is fetched from a server.

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

- At least one machine that can be used for installing a DHCP, NFS, HTTP, FTP, TFTP, or SMB server.
- Target system capable of PXE boot, networking, and Wake on LAN, plugged in and connected to the network.
- Controlling system with working network connection and SSH viewer software.

To perform this type of installation, proceed as follows:

1. Set up the server that contains the installation data.

2. Set up a DHCP and TFTP server for the network. Add the required boot parameters to enable the SSH server.
3. Enable PXE boot in the target machine firmware.
4. Initiate the boot process of the target system using Wake on LAN.
5. On the controlling workstation, open an SSH client software and connect to the target system.
6. Perform the installation as described in *Chapter 10, Installation steps*.
7. Reconnect to the target system after it reboots for the initial system configuration.

11.3 Monitoring installation via VNC

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

11.3.1 Preparing for VNC installation

To enable VNC on the installation target, specify the appropriate boot parameters at the initial boot for installation (see *Chapter 9, Boot parameters*). The target system boots into a text-based environment and waits for a VNC client to connect to the installation program.

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

Because the installation target announces itself via OpenSLP, you can retrieve the address information of the installation target via an SLP browser. There is no need for any physical contact with the installation target itself, provided your network setup and all machines support OpenSLP:

PROCEDURE 11.1: LOCATING VNC INSTALLATIONS VIA OPENSLLP


1. Run `slptool findsrvtypes | grep vnc` to get a list of all services offering VNC. The VNC installation targets should be available under a service named `YaST.installation.suse`.

2. Run `slptool findsrvs YaST.installation.suse` to get a list of installations available. Use the IP address and the port (usually `5901`) provided with your VNC viewer.

11.3.2 Connecting to the installation program

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

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

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

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

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

```
IP_ADDRESS:DISPLAY_NUMBER
```

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

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

Note that the browser VNC connection is not encrypted.

To perform a VNC installation, proceed as follows:

1. Launch your preferred Web browser.
2. Enter the following at the address prompt:

```
http://IP_ADDRESS_OF_TARGET:5801
```

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

11.4 Monitoring installation via SSH

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

11.4.1 Preparing for SSH installation

In addition to installing the required software package (OpenSSH for Linux and PuTTY for Windows), you need to specify the appropriate boot parameters to enable SSH for installation. See [Chapter 9, Boot parameters](#) for details. OpenSSH is installed by default on any SUSE Linux-based operating system.

11.4.2 Connecting to the installation program

After you have started the SSH installation, use this procedure to connect to the SSH session.

1. Retrieve the installation target's IP address. If you have physical access to the target machine, take the IP address the installation routine provides in the console after the initial boot. Otherwise take the IP address that has been assigned to this particular host in the DHCP server configuration.
2. In a command line, enter the following command:

```
ssh -X root@TARGET_IP_ADDRESS
```

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

3. When prompted for a user name, enter `root`.
4. When prompted for the password, enter the password that has been set with the SSH boot parameter. After you have successfully authenticated, a command line prompt for the installation target appears.
5. Enter `yast` to launch the installation program. A window opens showing the normal YaST screens as described in [Chapter 10, Installation steps](#).

11.5 Monitoring installation via serial console

For this installation method, you need a second computer connected by a *null modem* cable to the computer on which to install SUSE Linux Enterprise Micro. Hardware and firmware of both machines need to support the serial console. Some firmware implementations are already configured to send the boot console output to a serial console (by providing a device tree with `/CHOSEN_stdout_PATH` set appropriately). In this case no additional configuration is required.

If the firmware does not use the serial console for the boot console output, set the following boot parameter for the installation: `console=TTY,BAUDRATE`. For details [Chapter 9, Boot parameters](#).

`BAUDRATE` needs to be replaced by the baud rate for the interface. Valid values are 115200, 38400, or 9600. `TTY` needs to be replaced by the name of the interface. On most computers, there is one or more serial interfaces. Depending on the hardware, the names of the interfaces may vary:

- `ttyS0` for APM
- `ttyAMA0` for Server Base System Architecture (SBSA)
- `ttyPS0` for Xilinx

For the installation, you need a terminal program like `minicom` or `screen`. To initiate the serial connection, launch the `screen` program in a local console by entering the following command:

```
> screen /dev/ttyUSB0 115200
```

This means that `screen` listens to the first serial port with a baud rate of 115200. From this point on, the installation proceeds similarly to the text-based installation over this terminal.

12 Troubleshooting

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

12.1 Checking media

If you encounter any problems using the SUSE Linux Enterprise Micro 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 and 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.

12.2 No bootable drive available

If your computer cannot boot from USB or DVD drive, there are several alternatives. This is also an option if your drive is not supported by SUSE Linux Enterprise Micro.

Using an external USB flash drive or DVD drive

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

Network boot via PXE

If a machine lacks both a USB flash drive and 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.

12.3 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 USB flash drive or 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 firmware 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 12.1: 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 USB flash drive or 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 12.2: 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 USB flash drive or 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 USB flash drive or 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 12.1: US KEYBOARD LAYOUT

12.4 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 installation media 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 10, 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.

12.5 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 12.3: 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 10, Installation steps](#).

PROCEDURE 12.4: 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 10, Installation steps](#).

PROCEDURE 12.5: 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 SUSE Linux Enterprise Micro 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 10, 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.

12.6 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 12.4, "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**.

A GNU licenses

This appendix contains the GNU Free Documentation License version 1.2.

GNU Free Documentation License

Copyright (C) 2000, 2001, 2002 Free Software Foundation, Inc. 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA. Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other 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. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License 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 <https://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.