



JeOS Quick Start Guide

The following document provides an overview of SUSE Linux Enterprise Server JeOS and describes the setup procedure.

Publication Date: May 05, 2021

Contents

- 1 Introduction 2
- 2 Available image flavors 5
- 3 Installing and running JeOS on KVM 8

1 Introduction

JeOS (Just enough Operating System) refers to a minimal customizable operating system that is tailored for a specific usage scenario: a container host, a virtual machine guest, an appliance base, or a small server image. JeOS acts as a foundation for building systems for use in a data center, creating desired virtual application images or appliances. You can choose what packages and components to install based on a specific use case. To that end, JeOS provides access to all regular software repositories.

If you already have SUSE Linux Enterprise Server certification, you are automatically certified for JeOS, as it is derived from the same code base. SUSE Linux Enterprise Server JeOS allows you to reduce the operating system to the packages essential for the specific usage scenario. This minimizes the number of packages and patches that need to be managed and applied. By removing unnecessary packages, you also reduce the number of potential vulnerabilities, thus making the resulting system more secure. Reducing the system's footprint also results in lower resource consumption and improved performance.

JeOS is delivered as pre-configured and ready-to-run virtual machine images. The system comes with the `jeos-firstboot` wizard for configuring system locales and the root password during the first boot. JeOS ships with the SSH server enabled and excluded from the firewall configuration. The supplied KIWI templates make it possible to create customized JeOS images.

JeOS provides ready-to-deploy server images on the AMD64/Intel 64 architecture for the following usage scenarios:

- KVM/Xen with HVM
- Xen Paravirtualized
- Microsoft Hyper-V
- VMware
- OpenStack Cloud

1.1 System requirements

The minimum JeOS system requirements are similar to SUSE Linux Enterprise Server. For more information, refer to *Book “Deployment Guide”, Chapter 2 “Installation on AMD64 and Intel 64”, Section 2.1 “Hardware requirements”*.

1.2 Differences between JeOS and SUSE Linux Enterprise Server

There are several significant differences between JeOS and SUSE Linux Enterprise Server.

DIFFERENCES BETWEEN JEOS AND SUSE LINUX ENTERPRISE SERVER

jeos-firstboot

JeOS comes with the jeos-firstboot tool, which can be used to configure basic settings during the first boot. This includes keyboard layout and language, time zone, and root password.

The default network configuration is set to use DHCP, and the system automatically creates a Btrfs snapshot of the initial configuration.

kernel-default-base

Because of size constraints, JeOS images use the kernel-default-base as the default kernel. Since the kernel does not contain drivers for bare metal, this allows for a smaller footprint.

If a specific usage scenario requires any of the omitted modules, install the kernel-default package to get a full tree of the kernel modules. This is the same kernel used by a regular SUSE Linux Enterprise Server installation.

No repositories by default

When installing a SUSE Linux Enterprise Server from the installation media, the system automatically sets the installation media as the default offline channel.


JeOS has no channel by default, and you need to register your system to access online channels.

Zypper configuration

JeOS is a slimmed-down version of SUSE Linux Enterprise Server. The reduction in size is achieved in part by enabling the following two options in /etc/zypp/zypp.conf: rpm.install.excludedocs = yes (excludes all files marked as documentation from being installed by default) and solver.onlyRequires = true (disables installation of suggested and recommended packages by default).

Base packages

JeOS images are built using the KIWI tool. KIWI's configuration files, called templates, explicitly list packages that must be included in the images. Keep in mind that these base packages pull their dependencies during the image building process. Therefore, the list of preinstalled packages in an image usually contains more packages than the list of base packages.

The KIWI configuration files used to create the official JeOS images are available from the openSUSE Build Service at https://build.opensuse.org/package/view_file/SUSE:SLE-15-SP1:GA/kiwi-templates-SLES15-JeOS/JeOS.kiwi?expand=1 .



Note: Specific host tools packages

Each SUSE Linux Enterprise Server JeOS flavor contains certain host tools packages. Refer to the specific flavors section in the documentation for further information.

Disk layout

Unlike SUSE Linux Enterprise Server, which calculates the disk layout during installation, JeOS features the following fixed virtual image partitioning scheme:


- 2 MB BIOS boot
- 33 MB EFI system
- 24 GB Linux file system

To get additional disk space, you can either extend the existing disk from the virtual host or add a secondary disk to the virtual machine. If you choose the latter option, you must partition and format it manually.

systemd-coredump disabled

Due to the limited amount of memory and disk space available by default on JeOS images, systemd-coredump is disabled by default on JeOS images. To collect application core dumps necessary for troubleshooting, follow these steps:

- Install the `systemd-coredump` package, which contains `/usr/lib/sysctl.d/50-coredump.conf`
- To apply the configuration changes to your system, reboot or use the `sysctl --system` command.

For further information, refer to <https://documentation.suse.com/sles/single-html/SLES-tuning/#cha-tuning-systemd-coredump> .

2 Available image flavors

Ready-to-use JeOS images are available for the x86_64 architecture only. You can download JeOS for the following deployment targets:

- Fully Virtual JeOS for KVM and Xen (FV/HVM)
- JeOS for VMware
- JeOS for Microsoft Hyper-V
- JeOS for OpenStack
- Paravirtual JeOS for Xen (PV)




Important: Supported virtualization hosts (hypervisors)

JeOS can run as a virtualization guest on the same virtualization hosts (hypervisors) as SUSE Linux Enterprise Server. See *Book “Virtualization Guide”, Chapter 7 “Virtualization limits and support”, Section 7.3 “Supported host environments (hypervisors)”* for more information on what hypervisors are supported.

2.1 JeOS Image for OpenStack Cloud

JeOS for OpenStack Cloud is an image designed specifically for use in an OpenStack environment.

Instead of the `jeos-firstboot` tool, JeOS for OpenStack Cloud uses `cloud-init` (<https://cloudinit.readthedocs.io/en/latest/#>)  for configuring the image. The image also uses the XFS file system and the following kernel configuration:

```
"plymouth.enable=0 console=ttyS0,115200 console=tty0 net.ifnames=0"
```

This means that the image ships without packages required by `jeos-firstboot` and Btrfs.

JeOS for OpenStack Cloud includes several OpenStack-related packages, such as `cloud-init`, `cloud-init-config-suse`, and `xfsprogs`.

2.1.1 **cloud-init** basics

cloud-init automatically applies custom configuration to running instances. The tool features a wide range of configuration modules, and it can execute specific commands in the image during the boot phase.

The default configuration for cloud-init is part of the `cloud-init-config-suse` package, and it is located in the `/etc/cloud/cloud.cfg` directory.

2.1.2 **cloud-init** configuration examples

By default, cloud-init creates a user account called `sles`. This acts as the default user and has **sudo** rights. For security reasons, the `root` and `sles` users does not have passwords. Use the following example to create passwords for both users.

EXAMPLE 1: **ADD PASSWORDS**

```
#cloud-config
#cloud-config
chpasswd:
  list: |
    root:$6$SaLTsaLt$.cuqsgjNGxHJZSLmVqkI/
j9.kWIiUQHqjLRswbHjTw9hIXdPomgey2DHZmp6Ho6wgqIKz3ufb0IKdQCnQHPSA1
    sles:$6$SaLTsaLt$.cuqsgjNGxHJZSLmVqkI/
j9.kWIiUQHqjLRswbHjTw9hIXdPomgey2DHZmp6Ho6wgqIKz3ufb0IKdQCnQHPSA1
  expire: False
```

To create the SHA-512 hashes used in the example, run the command: `openssl passwd -6 -salt SaLTsaLt PASSWORD`.



Warning: User passwords

To avoid potential security risks, do not use passwords in production environments. Even when using hashed passwords, you are exposed to dictionary attacks. For better security, use SSH authentication instead of passwords.

When creating users, keep in mind that logging in with a password is disabled by default, and that the newly-created users do not have sudo rights.

The following examples show configurations for creating various users.

EXAMPLE 2: **NEW USER WITH PASSWORD AND SUDO**

```
#cloud-config
```

```

users:
  - default
  - name: USER
    shell: /bin/bash
    groups: users
    # lock_passwd: Disable password login. Defaults to true
    lock_passwd: false
    passwd: $6$Sa1TsaLt$.cuqsgjNGxHJZSLmVqkI/
j9.kWIiUQHqjLRswbHjTw9hIXdPomgey2DHZmp6Ho6wgqIKz3ufb0IKdQCnQHPSA1
    sudo: ALL=(ALL) NOPASSWD:ALL

```

EXAMPLE 3: NEW USER WITH SSH KEYS AND SUDO

```

#cloud-config
users:
  - name: USER
    shell: /bin/bash
    groups: users
    ssh_import_id: None
    lock_passwd: true
    sudo: ALL=(ALL) NOPASSWD:ALL
    ssh_authorized_keys:
      - SSH_PUBLIC_KEY_1
      - SSH_PUBLIC_KEY_2

```

To connect and register with the SUMA, RMT, or SUSEConnect, use the `runcmd` module to run the desired commands, for example:

```

#cloud-config
runcmd:
  - SUSEConnect -r REGISTRATION_CODE -e EMAIL_ADDRESS

```

For further information, see <https://documentation.suse.com/sles/html/SLES-all/cha-register-sle.html>.

For RMT, the configuration is as follows:

```

#cloud-config
runcmd:
  - curl http://RMT_SERVER/tools/rmt-client-setup \ --output rmt-client-setup
  - sh rmt-client-setup https://RMT_SERVER/

```

For further information, see <https://documentation.suse.com/sles/html/SLES-all/cha-rmt-client.html>.

For SUMA, the configuration is as follows:

```

#cloud-config

```

```
runcmd:
- curl -O http://suma01/pub/bootstrap/bootstrap.sh
- /bin/bash bootstrap.sh
```

For further information, see <https://documentation.suse.com/external-tree/en-us/suma/4.0/suse-manager/client-configuration/registration-bootstrap.html>.

Finally, the following example shows a configuration that installs a package and starts a service during the first boot.

```
#cloud-config
packages:
- qemu-guest-agent
runcmd:
- systemctl enable qemu-guest-agent.service
- systemctl start --no-block qemu-guest-agent.service
```

2.2 KVM/Xen and Xen flavor

The JeOS image for KVM/Xen and Xen ships with the following packages:

- xen-tools-domU
- grub2-x86_64-xen
- xen-libs

2.3 Microsoft Hyper-V flavor

The JeOS image for Microsoft Hyper-V ships with hyper-v.

2.4 VMware flavor

The JeOS image for VMware ships with open-vm-tools.

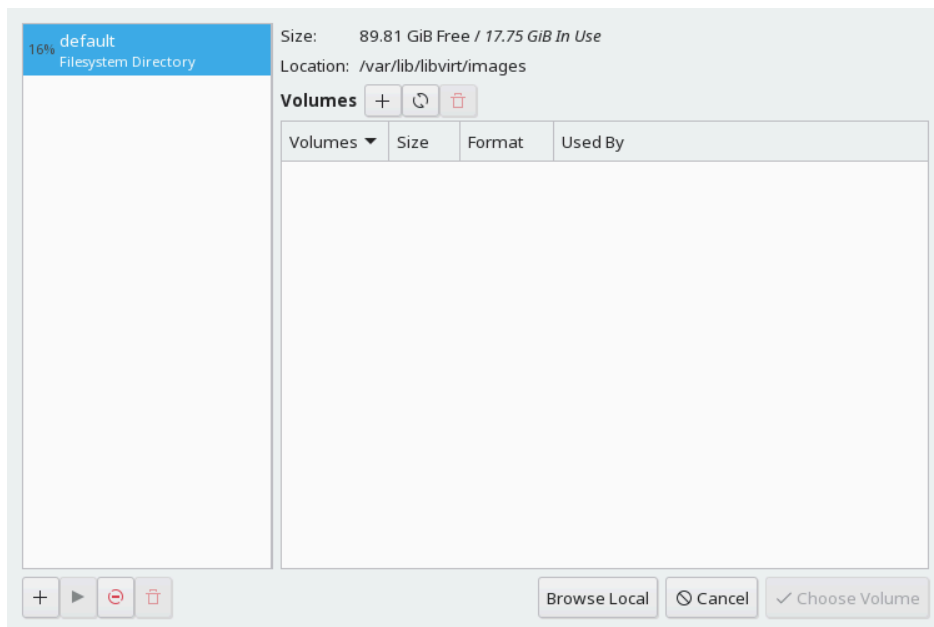
3 Installing and running JeOS on KVM

Provided you have KVM installed on your local machine, you can use a ready-made JeOS image to create a virtual machine. This allows you to experiment with JeOS safely.

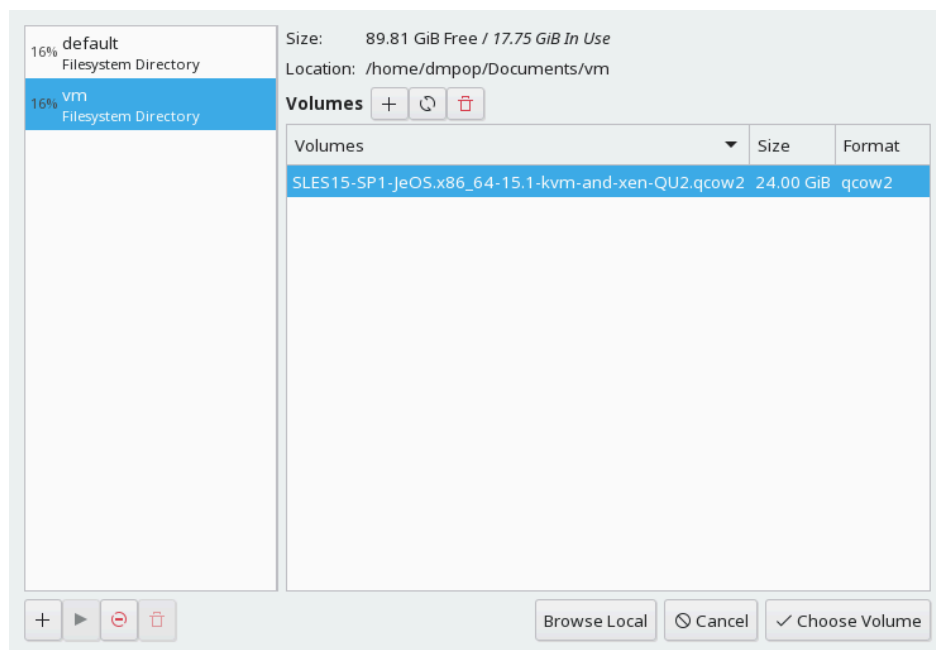
Download a 60-day trial JeOS image for KVM at <https://www.suse.com/products/server/jeos/>. Refer to the following procedure to create a JeOS virtual machine in KVM.

PROCEDURE 10: CREATING A JEOS VIRTUAL MACHINE

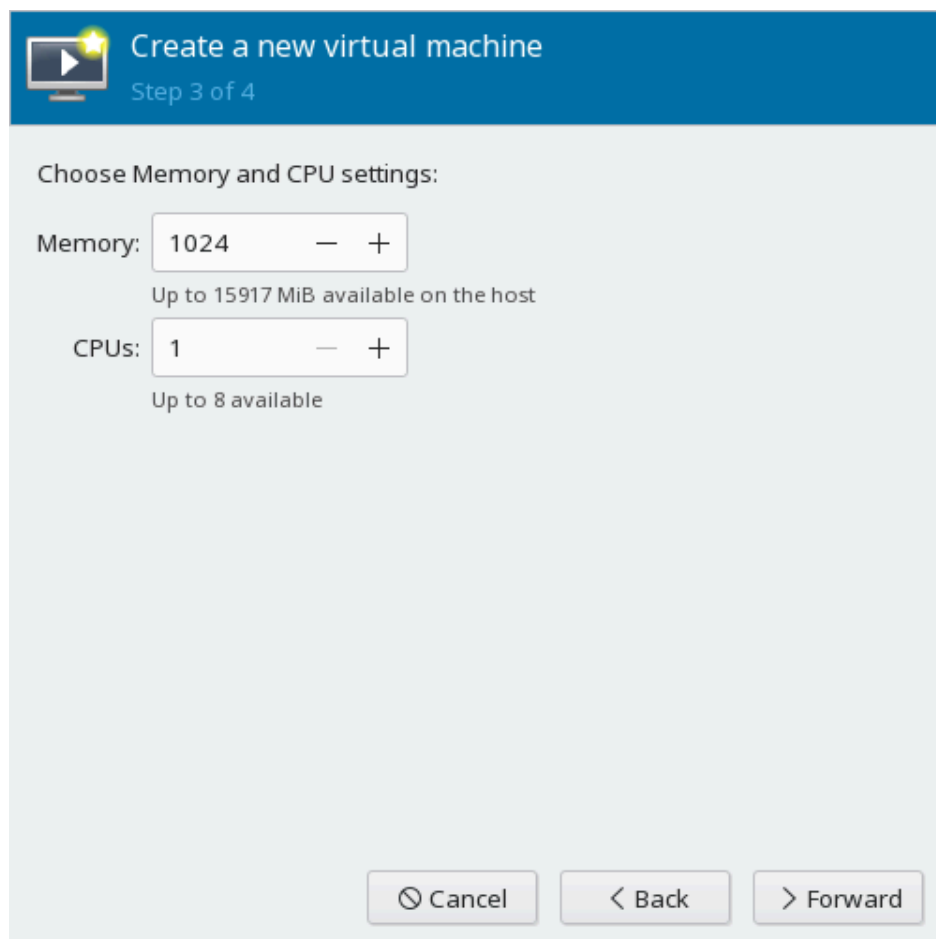
1. Launch Virtual Machine Manager and choose *File > New Virtual Machine*.
2. Select *Import existing disk image* and click *Forward*.
3. Click the *Browse* button next to the *Provide the existing storage path* field.
4. In the *Choose Storage Volume* dialog, click *Add Pool*, give the new pool a name, and select *dir: Filesystem Directory* from the *Type* drop-down box. Click *Forward*.



5. Click the *Browse* button next to the *Target path* field and select the directory containing the JeOS qcow2 image. Click *Finish*.
6. Back in the *Choose Storage Volume* dialog, select the created pool and choose the qcow2 image in the *Volumes* section. Then click *Choose Volume*.



7. Back in the *Create Virtual Machine* dialog, enter Generic in the *Choose the operating system you are installing* field. Click *Forward*.
8. Set the RAM value to 1024, and the number of CPUs to 1. Click *Forward*.



Create a new virtual machine

Step 3 of 4

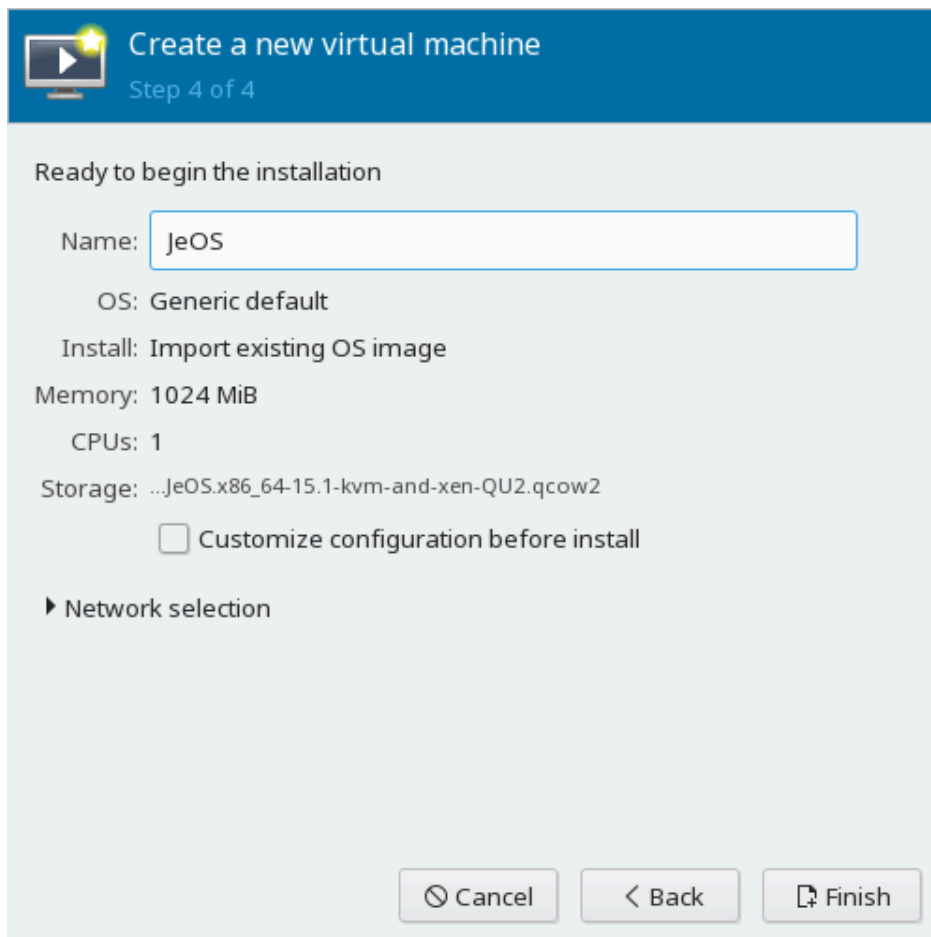
Choose Memory and CPU settings:

Memory: 1024 — +
Up to 15917 MiB available on the host

CPUs: 1 — +
Up to 8 available

Cancel < Back > Forward

9. Give the new virtual machine a name (for example, Je0S) and click *Finish*.



10. If prompted to start the virtual network, click *Yes*.

After the system is booted, `jeos-firstboot` guides you through the initial system setup.



Note: Change locale

The JeOS image comes with the `en_US` locale only. You can install and select the desired system locale after the setup is completed by installing the `glibc-locale` package and running the `localectl set-locale LANG=LOCALE` command (replace `LOCALE` with the desired language locale, for example `de_DE.UTF-8`).

1. As the first step, you are prompted to select the appropriate keyboard layout using the keyboard selection dialog.
2. Next, read and accept the License Agreement. You cannot proceed without accepting the license agreement.

3. Specify the desired timezone.
4. When prompted, specify and confirm the desired root password.
5. As the last step, you are instructed to register your system. If you only plan to test JeOS, you can skip registration.
6. After the initial configuration is completed, you can log in to the system as root, using the password you specified during the setup procedure.



Tip: Installing product patches after first boot

After JeOS has been successfully installed and registered, we highly recommend to install the latest available online updates.