Deploying SUSE Linux Micro Using Raw Disk Images on IBM Z zFCP Disks

WHAT?

SUSE Linux Micro provides raw images (also referred to as pre-built images) that can be deployed directly to your device storage.

WHY?

This article provides you with step-by-step instructions on deploying SUSE Linux Micro on an IBM Z machine.

EFFORT

It takes approximately 20 minutes to read the article.

GOAL

SUSE Linux Micro is successfully deployed on your system.

REQUIREMENTS

- A disk with running Linux.
- A disk where you deploy the raw image and where SUSE Linux Micro will run.
- A disk that serves as a configuration medium.

Publication Date: 03 Nov 2025

Contents

1 About pre-built images

Pre-built images are ready-to-use representations of a running operating system. They are not installed in a traditional way using an installer, but copied to the hard disk of the target host. The topic covers basic information about these pre-built images.

The pre-built images are intended to be configured on the first boot by using tools delivered in the images. The boot loader detects the first boot as described in *Section 1.1, "First boot detection"*.

1.1 First boot detection

The deployment configuration runs on the first boot only. To distinguish between the first and subsequent boots, the file `/etc/machine-id` is created after the first boot finishes. If the file is not present in the file system, the system assumes that this is a first boot and triggers the configuration process. After completing the first boot, the `/etc/machine-id` file is created.

Note: The `/etc/machine-id` file is always created

Even though the configuration may not be successful because of improper or missing configuration files, the `/etc/machine-id` file is created.

1.1.1 Force system reconfiguration on a subsequent boot

If you need to reconfigure your system after the first boot happened, you can force the reconfiguration on the subsequent boot. Here you have two options.

- You can pass the `ignition.firstboot` or `combustion.firstboot` attribute to the kernel command line.
- You can delete the file `/etc/machine-id` and reboot the system.

2 Deployment of SUSE Linux Micro

The deployment procedure of SUSE Linux Micro comprises three phases. Do not skip any of the phases and follow the order as stated below.

1. Ensure that all disks are available and active. For details, refer to *Section 3.1, "Preparing the zFCP/SCSI disks"*.

2. Preparing a configuration device—to configure the network, add users and register the system, prepare the configuration according to the instructions provided in *Section 4, "Preparing the configuration disk"*.

3. Downloading the SUSE Linux Micro image and deploying it on the disk. For details, refer to *Section 5, "Deploying the raw disk image on the disk"*.

3 Preparing the machine

3.1 Preparing the zFCP/SCSI disks

To prepare the machine to deploy on zFCP/SCSI disks, proceed as described below:

1. Find all disks by running the command:

```
# lszdev zfcp

TYPE       ID                                                  ON  PERS  NAMES
zfcp-host 0.0.fa00                                             yes yes
zfcp-host 0.0.fc00                                             yes yes
zfcp-lun  0.0.fa00:0x500507630b181216:0x4021400a00000000 yes no   sda sg0
zfcp-lun  0.0.fc00:0x500507630b101216:0x4021400b00000000 yes no   sdb sg1
```

2. In the output, there should be a LUN device—a device of the type `zfcp-lun`. If the command does not output any device, enable the disk first:

```
# chzdev -e fa00
```

   If you do not know the disk name, contact your system administrator.

3. Verify that the LUN device exists:

```
# lszdev zfcp

TYPE      ID                                                  ON  PERS NAMES
zfcp-host 0.0.fa00                                             yes yes
zfcp-host 0.0.fc00                                             yes yes
```

```
zfcp-lun  0.0.fa00:0x500507630b181216:0x4021400a00000000 yes no   sda sg0
zfcp-lun  0.0.fc00:0x500507630b101216:0x4021400b00000000 yes no   sdb sg1
```

4. Note the information about LUN devices. The `/dev/sda` device is used further for the deployment of SUSE Linux Micro.

5. Even though formatting the disk where SUSE Linux Micro will be deployed is unnecessary, check if the disk is empty and does not contain any partitions. If there are any partitions, remove them with the following command. **Beware, the command will erase data on the provided device!**

```
# parted DEVICE_NAME rm 1
```

4 Preparing the configuration disk

Tip: Virtual versus physical disk

A virtual disk may be more suitable for production deployments, while a physical disk is better suited for development.

During the installation process, you can pass a complex configuration to define users, directories, or to provide SSH keys. To do so, create a configuration device that stores a complete configuration. The following example procedure describes how to create such a device:

1. Identify the disk:

```
# lsdasd

Bus-ID    Status    Name      Device  Type         BlkSz  Size      Blocks
==============================================================================
0.0.0100  active    dasda     94:0    ECKD         4096   30720MB   7864380
0.0.0101  active    dasdc     94:8    ECKD         4096   20480MB   5243040
0.0.0102  active    dasdb     94:4    ECKD         4096   5120MB    1310760
```

   In the output above, the `Bus-ID` value of the disk to be used as a configuration device is *0.0.0102*. Use the value of `Bus-ID` or the disk size to identify the disk, as the device name (`/dev/dasdX`) can change after each reboot.

2. Format the disk. Keep in mind that formatting removes all data on that disk.

```
# dasdfmt -b 4096 -y -p /dev/dasdb
```

3. Create a partition:

```
# parted /dev/dasdb mkpart ext4 0% 100%
```

4. Format the partition:

```
# mkfs.ext4 /dev/dasdb
```

5. Label the partition as `ignition`:

```
# e2label /dev/dasdb1 ignition
```

6. Verify the file system:

```
# blkid
```

7. Mount the partition:

```
# mount /dev/dasdb1 /mnt
```

8. Prepare the directory structure for Ignition and/or Combustion. For Ignition:

```
# mkdir -p  /mnt/ignition
```

   For Combustion:

```
# mkdir -p /mnt/combustion
```

9. Copy your public SSH key to `/mnt/combustion` as `ssh_key.pub`.
10. Prepare the configuration as described further.

4.1 Configuring SUSE Linux Micro deployment with Combustion

Combustion is a dracut module that enables you to configure your system on the first boot. You can use Combustion, for example, to change the default partitions, set user passwords, create files, or install packages.

4.1.1 How does Combustion work?

Combustion is invoked after the `ignition.firstboot` argument is passed to the kernel command line. Combustion reads a provided file named `script`, executes included commands, and thus performs changes to the file system. If `script` includes the network flag, Combustion tries to configure the network. After `/sysroot` is mounted, Combustion tries to activate all mount points in `/etc/fstab` and then calls **`transactional-update`** to apply other changes, for example, setting `root` password or installing packages.

The configuration file `script` must reside in the `combustion` subdirectory on the configuration media labeled `combustion`. The directory structure must look as follows:

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

Tip: Using Combustion together with Ignition

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.

4.1.2 Combustion configuration examples

4.1.2.1 The `script` configuration file

The `script` configuration file is a set of commands that are parsed and executed by Combustion in a **`transactional-update`** shell. This article provides examples of configuration tasks performed by Combustion.

Tip: Use Fuel Ignition to generate the Combustion script

To create the Combustion script, you can use the Fuel Ignition Web application. There you can select appropriate parameters and the application generates a Combustion script that you can download.

Important: Include interpreter declaration

As the `script` file is interpreted by the shell, always start the file with the interpreter declaration on its first line. For example, in case of Bash:

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

To log in to your system, include at least the `root` password. However, it is recommended to establish the authentication using SSH keys. If you need to use a `root` password, make sure to configure a secure password. For 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.

4.1.2.1.1 Default partitioning

Each image has the following subvolumes:

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

The `/etc` directory is mounted as overlayFS, where the upper directory is mounted to `/var/lib/overlay/1/etc/`.

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.

If you want to add a new user or modify any of the files on a subvolume that is not mounted by default, you need to declare such subvolume first so that it is mounted as well.

4.1.2.1.2 Network configuration

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

```
# combustion: network
```

Using this statement passes the `rd.neednet=1` argument to dracut. The network configuration defaults to using DHCP. If a different network configuration is needed, proceed as described in *Section 4.1.2.1.3, "Performing modifications in the initramfs"*.

If you do not use the statement, the system remains configured without any network connection.

4.1.2.1.3 Performing modifications in the initramfs

You may need to perform changes to the initramfs environment, for example, to write a custom network configuration for NetworkManager into `/etc/NetworkManager/system-connections/`. To do so, use the `prepare` statement.

For example, to create a connection with a static IP address and configure DNS:

```
#!/bin/bash
# combustion: network prepare
set -euxo pipefail

nm_config() {
  umask 077 # Required for NM config
  mkdir -p /etc/NetworkManager/system-connections/
  cat >/etc/NetworkManager/system-connections/static.nmconnection <<-EOF
  [connection]
  id=static
  type=ethernet
  autoconnect=true

  [ipv4]
  method=manual
  dns=192.168.100.1
  address1=192.168.100.42/24,192.168.100.1
EOF
}

if [ "${1-}" = "--prepare" ]; then
  nm_config # Configure NM in the initrd
  exit 0
fi
```

```
# Redirect output to the console
exec > >(exec tee -a /dev/tty0) 2>&1

  nm_config # Configure NM in the system
  curl example.com

# Close outputs and wait for tee to finish
exec 1>&- 2>&-; wait;

# Leave a marker
echo "Configured with combustion" > /etc/issue.d/combustion
```

4.1.2.1.4 Waiting for the task to complete

Some processes may be run in background, for example, the **tee** process that redirects output to the terminal. To ensure that all running processes are completed before the `script` execution finishes, add the following line:

```
exec 1>&- 2>&-; wait;
```

4.1.2.1.5 Partitioning

SUSE Linux Micro raw images are delivered with a default partitioning scheme. You might want to use a different partitioning. The following set of example snippets moves the `/home` to a different partition.

Important: Certain directories must reside on the same partition as /

When changing partitioning, do not place the following directories on a different partition than the root file system: `/boot`, `/usr`, `/etc`, `/dev`.

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 is discarded, certain changes remain visible and cannot be reverted, for example, the changes to the `/dev/vdb` device.

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

```
sfdisk /dev/vdb <<EOF
sleep 1
label: gpt
type=linux
EOF

partition=/dev/vdb1
```

As the **`sfdisk`** command may take longer time to complete, postpone **`label`** by using the **`sleep`** command after **`sfdisk`**.

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
```

4.1.2.1.6 Creating new users

As some services, such as Cockpit, require login using a non-root user, define at least one unprivileged user here. Alternatively, you can create such a user from a running system as described in *Section 6.2, "Adding users"*.

To add a new user account, first create a hash string that represents the user's password. Use the **`openssl passwd -6`** command.

After you obtain the password hash, add the following lines to the `script`:

```
mount /home
useradd -m EXAMPLE_USER
```

```
echo 'EXAMPLE_USER:PASSWORD_HASH' | chpasswd -e
```

4.1.2.1.7 Setting a password for `root`

Before you set the `root` password, generate a hash of the password, for example, by using the **`openssl passwd -6`**. To set the password, add the following line to the `script`:

```
echo 'root:PASSWORD_HASH' | chpasswd -e
```

4.1.2.1.8 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 4.1.2.1.9, "Enabling services"*.

4.1.2.1.9 Enabling services

To enable system services, for example, the SSH service, add the following line to `script`:

```
systemctl enable sshd.service
```

4.1.2.1.10 Installing packages

Important: Network connection and registering your system may be necessary

As certain packages may require additional subscription, you may need to register your system beforehand. An available network connection may also 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.

4.1.2.2 A complete example of the `script` file

The following `script` provides complete settings that may serve you as a guide on how to write your own Combustion configuration. The example does not require any further Ignition configuration.

```
    #!/bin/bash
    # combustion: network prepare


    set -euxo pipefail

    ## The OSA subchannels to enable
    ZNET_SUBCHANNELS=0.0.1000,0.0.1001,0.0.1002

    ## Network information to configure
    IPADDRESS="10.144.64.155/24" ## Formet is ipaddress/cidr
    GATEWAY="10.144.64.254"
    NAMESERVERS="10.144.53.53;10.144.53.54" ## A semicolon-separated list of name servers

    ## Hostname information
    NODE_HOSTNAME="micro6"

    ## Add password for root user
    ## Use either 'openssl passwd -6' or 'mkpasswd --method=sha-512' to encrypt the
password.
    ROOT_USER_PASSWORD='PASSWORD_HASH'
    SSH_ROOT_PUBLIC_KEY=ssh_key.pub

    ## Add a regular user, because root login may be disallowed in some services.
    CREATE_NORMAL_USER=user ## Replace the "user" with a desired username here.
    NORMAL_USER_PASSWORD='PASSWORD_HASH'
```

```
   SSH_USER_PUBLIC_KEY=ssh_key.pub

   ## Register to SUSE Customer Center and install additional packages
   REG_EMAIL='tux@suse.com' ## Email address for product registration
   SLMICRO_REGCODE='REGISTRATIONCODE' ## A registration code required to install
additional packages
   ADDITIONAL_PACKAGES='' ## A space separated list of additional packages to install


   nm_config() {
     umask 077 # Required for Network Manager configuration
     mkdir -p /etc/NetworkManager/system-connections/
     cat >'/etc/NetworkManager/system-connections/Wired connection 1.nmconnection' <<EOF
     [connection]
     id=static
     type=ethernet
     autoconnect=true

     [ipv4]
     method=manual
     address1=$IPADDRESS
     gateway=$GATEWAY
     dns=$NAMESERVERS
   EOF
   }

   if [ "${1-}" = "--prepare" ]; then
     # Configure Network Manager in the initrd
     nm_config
     # Enable OSA network devices
     chzdev qeth $ZNET_SUBCHANNELS -ep
     chzdev qeth $ZNET_SUBCHANNELS -e
     exit 0
   fi



   ## Post output on stdout
   exec > >(exec tee -a /dev/ttyS0) 2>&1

   ## Set hostname
   echo $NODE_HOSTNAME > /etc/hostname

   ## Set root password
   echo root:$ROOT_USER_PASSWORD | chpasswd -e
   ## Add ssh public key as authorized key for the root user
   mkdir -pm700 /root/.ssh/
```

```
cat $SSH_ROOT_PUBLIC_KEY >> /root/.ssh/authorized_keys

## Mount /var and /home so user can be created smoothly
if [ "$CREATE_NORMAL_USER" ]
then
  mount /var && mount /home
fi
## User creation
if [ "$CREATE_NORMAL_USER" ]
then
  echo "User creation is requested, creating user."
  useradd -m $CREATE_NORMAL_USER -s /bin/bash -g users
  echo $CREATE_NORMAL_USER:$NORMAL_USER_PASSWORD | chpasswd -e
  echo $CREATE_NORMAL_USER "ALL=(ALL) NOPASSWD: ALL" >> /etc/sudoers.d/adminusers
  mkdir -pm700 /home/$CREATE_NORMAL_USER/.ssh/
  chown -R $CREATE_NORMAL_USER:users /home/$CREATE_NORMAL_USER/.ssh/
  cat $SSH_USER_PUBLIC_KEY >> /home/$CREATE_NORMAL_USER/.ssh/authorized_keys
  echo "Requested user has been created, requested password has been set."
else
  echo "No user will be created"
fi

# Configure NM in the system
nm_config
# Enable OSA network device
chzdev qeth $ZNET_SUBCHANNELS -ep
chzdev qeth $ZNET_SUBCHANNELS -e

## Enable services
echo "Enabling services."
systemctl enable cockpit.socket
systemctl enable sshd

## Unmount var and home
if [ "$CREATE_NORMAL_USER" ]
then
  umount /var && umount /home
fi

# Close outputs and wait for tee to finish
exec 1>&- 2>&-; wait;

echo "Configured with Combustion at $(date)" > /etc/issue.d/combustion
```

4.2 Configuring SUSE Linux Micro deployment with Ignition

Ignition (https://coreos.github.io/ignition/) is a provisioning tool that enables you to configure a system according to your specification on the first boot.

4.2.1 How does Ignition work?

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 the kernel switches from the temporary file system to the real root file system (before the `switch_root` command is issued).

Ignition uses a configuration file in the JSON format named `config.ign`. You can either write the configuration manually or use the Fuel Ignition Web application at https://ignite.opensuse.org to generate it.

Important

Fuel Ignition does not cover the complete Ignition vocabulary yet, and the resulting JSON file may need additional manual tweaking.

4.2.1.1 `config.ign`

The configuration file `config.ign` must reside in the `ignition` subdirectory on the configuration media, for example, a USB stick labeled `ignition`. The directory structure must look as follows:

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

Tip

To create a disk image with the Ignition configuration, you can use the Fuel Ignition Web application at https://ignite.opensuse.org.

The `config.ign` contains multiple 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/).

The `version` attribute is mandatory and in case of SUSE Linux Micro, its value must be set either to `3.4.0` or to any lower version. Otherwise, Ignition will fail.

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. 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.

4.2.2 Ignition configuration examples

This section provides several examples of the Ignition configuration in the built-in JSON format.

Note: The `version` attribute is mandatory

Each `config.ign` must include version 3.4.0 or lower that is then converted to the corresponding Ignition specification.

4.2.2.1 Default partitioning

Each image has the following subvolumes:

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

The `/etc` directory is mounted as overlayFS, where the upper directory is mounted to `/var/lib/overlay/1/etc/`.

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.

If you want to add a new user or modify any of the files on a subvolume that is not mounted by default, you need to declare such subvolume first so that it is mounted as well.

4.2.2.2 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 `filesystems` attribute is used to format partitions. The `files` attribute can be used to create files in the file system.

The example below configures four partitions, including a dedicated swap partition, and creates a file system on each partition.

```
{
  "ignition": {
    "version": "3.0.0"
  },
  "storage": {
    "disks": [
      {
        "device": "/dev/vda",
        "partitions": [
          {
            "label": "root",
            "number": 1,
            "sizeMiB": 30720
          },
          {
            "label": "boot",
            "number": 2,
            "sizeMiB": 8720
          },
          {
            "label": "swap",
            "number": 3,
            "sizeMiB": 4096
          },
          {
            "label": "home",
            "number": 4,
            "sizeMiB": 30720
          }
        ],
        "wipeTable": true
      }
    ]
    "filesystems": [
      {
        "device": "/dev/disk/by-partlabel/root",
        "format": "btrfs",
        "label": "root"
```

```
      },
      {
        "device": "/dev/disk/by-partlabel/swap",
        "format": "swap",
        "label": "swap"
      }
      {
        "device": "/dev/disk/by-partlabel/boot",
        "format": "btrfs",
        "label": "boot"
      }
      {
        "device": "/dev/disk/by-partlabel/home",
        "format": "ext4",
        "label": "home"
      }
    ]
  }
}
```

Each of the mentioned attributes is described in the following sections.

4.2.2.2.1 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. Keep in mind that at least the `root` and `boot` partitions (`swap` if configured) need to be formatted to bear a file system.

The following example uses a single virtual device and divides the disk into four partitions:

```
...
  "storage": {
    "disks": [
      {
        "device": "/dev/vda",
        "partitions": [
          {
            "label": "root", ❶
            "number": 1, ❷
            "sizeMiB": 30720 ❸
          },
          {
            "label": "boot",
            "number": 2,
            "startMiB": 30720, ❹
```

```
            "sizeMiB": 8720
          },
          {
            "label": "swap",
            "number": 3,
            "sizeMiB": 4096
          },
          {
            "label": "home",
            "number": 4,
            "sizeMiB": 30720
          }
        ],
        "wipeTable": true
      }
    ]
   ...
```

1. The partition identification. Depending on the partition file system, it can have up to 16 characters for EXT-type file systems and 256 characters in the case of Btrfs.
2. The position of the partition in the partition table. If set to 0, the next free position is used.
3. The size of the partition in MiB.
4. Identifies the starting point of the particular partition.

4.2.2.2.2 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

For example:

```
{
  "ignition": {
    "version": "3.0.0"
```

```
  },
  "storage": {
    "raid": [
      {
        "devices": [
          "/dev/sda",
          "/dev/sdb"
        ],
        "level": "raid1",
        "name": "system"
      }
    ]
  }
}
```

4.2.2.2.3 The `filesystems` attribute

Note: Ignition does not perform modifications to mount units

The `filesystems` attribute does not modify mount units. If you add a new partition or remove an existing partition, you must manually adjust the mount units.

Important: Certain directories must reside on the same partition as /

When changing partitioning, do not place the following directories on a different partition than the root file system: `/boot`, `/usr`, `/etc`, `/dev`.

`filesystems` 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, ext3, xfs, vfat or swap)

Note

In case of SUSE Linux Micro, the `root` file system must be formatted to Btrfs.

The following example demonstrates using the `filesystems` attribute. The `/opt` directory will be mounted to the `/dev/sda1` partition, which is formatted to Btrfs. The device will not be erased.

For example:

```
{
  "ignition": {
    "version": "3.0.0"
  },
  "storage": {
    "filesystems": [
      {
        "device": "/dev/sda1",
        "format": "btrfs",
        "path": "/opt",
        "wipeFilesystem": false
      }
    ]
  }
}
```

Normally, a regular user's home directory is located in the `/home/USER_NAME` directory. Since `/home` is not mounted by default in the initrd, the mount has to be explicitly defined for the user creation to succeed:

```
{
  "ignition": {
    "version": "3.1.0"
  },
  "passwd": {
    "users": [
      {
        "name": "root",
        "passwordHash": "PASSWORD_HASH",
        "sshAuthorizedKeys": [
          "ssh-rsa SSH_KEY_HASH"
        ]
      }
    ]
  },
  "storage": {
    "filesystems": [
      {
        "device": "/dev/sda3",
        "format": "btrfs",
```

```
        "mountOptions": [
          "subvol=/@/home"
        ],
        "path": "/home",
        "wipeFilesystem": false
      }
    ]
  }
}
```

4.2.2.2.4 The `files` attribute

You can use the `files` attribute to create any files on your machine. Bear in mind that to create files outside the default partitioning schema, you need to define the directories by using the `filesystems` attribute.

In the following example, a host name is created by using the `files` attribute. The file `/etc/hostname` will be created with the *sl-micro1* host name:

Important

Keep in mind that JSON accepts file modes in decimal numbers, for example, `420`.

JSON:

```
{
  "ignition": {
    "version": "3.0.0"
  },
  "storage": {
    "files": [
      {
        "overwrite": true,
        "path": "/etc/hostname",
        "contents": {
          "source": "data:,sl-micro1"
        },
        "mode": 420
      }
    ]
  }
}
```

4.2.2.2.5 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.

For example:

```
{
  "ignition": {
    "version": "3.0.0"
  },
  "storage": {
    "directories": [
      {
        "path": "/home/tux",
        "user": {
          "name": "tux"
        }
      }
    ]
  }
}
```

4.2.2.3 Users administration

The `passwd` attribute is used to add users. As some services, such as Cockpit, require login using a non-root user, define at least one unprivileged user here. Alternatively, you can create such a user from a running system as described in *Section 6.2, "Adding users"*.

To log in to your system, create `root` and a regular user and set their passwords. You need to hash the passwords, 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.

For example:

```
{
  "ignition": {
    "version": "3.0.0"
  },
  "passwd": {
    "users": [
```

```
      {
        "name": "root",
        "passwordHash": "PASSWORD_HASH",
        "sshAuthorizedKeys": [
          "ssh-rsa SSH_KEY_HASH USER@HOST"
        ]
      }
    ]
  }
}
```

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

4.2.2.4 Enabling `systemd` services

You can enable `systemd` services by specifying them in the `systemd` attribute.

For example:

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

4.2.2.5 Converting YAML formatted files into JSON

JSON is a universal file format for storing structured data. Applications, for example, Ignition, use it to store and retrieve their configuration. Because JSON's syntax is complex and hard to read for human beings, you can write the configuration in a more friendly format called YAML and then convert it into JSON.

4.2.2.5.1 Converting YAML files into JSON format

The tool that converts Ignition-specific vocabularies in YAML files into JSON format is `butane`. It also verifies the syntax of the YAML file to catch potential errors in the structure. For the latest version of `butane`, add the following repository:

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

Replace `openSUSE_Tumbleweed` with one of the following (depending on your distribution):

- `'openSUSE_Leap_$releasever'`
- `15.5`

Now you can install the `butane` tool:

```
> sudo  zypper ref && zypper in butane
```

After the installation is complete, you can invoke `butane` by running:

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

- `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.

5 Deploying the raw disk image on the disk

To deploy SUSE Linux Micro on your machine, proceed as follows:

1. Download the raw disk image using **`wget`** or **`curl`**. For example:

   ```
   > curl -L0kO
   ```

2. Extract the image:

   ```
   > unpack xz -d BUILD_IDENTIFICATION.raw.xz
   ```

3. Copy the raw disk image to the disk:

   ```
   dd if=IMAGE_NAME.raw status=progress  of=/dev/sda bs=4096
   ```

4. To check the proper setup with the following command:

```
# lsblk
```

   The command should return two partitions on `/dev/sda`

5. Shut down the running Linux:

```
# init 0
```

6. Open the x3270 terminal and define the device to load the system from. Use the LUN number of the `/dev/sda` device, but split it to 8-digit chunks. For example, the command for LUN *0.0.fa00:0x500507630b181216:0x4021400a00000000* looks as follows:

```
# SET LOADDEV PORTNAME 50050763 0b181216 LUN 4021400a 00000000
```

7. Start SUSE Linux Micro:

```
# ipl FA00
```

6 Post-deployment steps

6.1 Registering SUSE Linux Micro from CLI

If your system was not registered during the deployment process by using the Combustion `script`, you can register from the running system.

To register SUSE Linux Micro with SUSE Customer Center, proceed as follows:

1. Run **`transactional-update register`** as follows:

```
# transactional-update register -r REGISTRATION_CODE -e EMAIL_ADDRESS
```

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

```
# transactional-update register -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 Micro. Replace *`EMAIL_ADDRESS`* with the e-mail address associated with the SUSE account you or your organization uses to manage subscriptions.

2. Reboot your system to switch to the latest snapshot.
3. SUSE Linux Micro is now registered.

Note: Other registration options

For information that goes beyond the scope of this section, refer to the inline documentation with **`SUSEConnect --help`**.

6.2 Adding users

Since SUSE Linux Micro requires having an unprivileged user to log in via SSH or to access Cockpit by default, we recommend to create such an account.

This step is optional if you have defined an unprivileged user in the Combustion.

1. Run the **`useradd`** command as follows:

```
# useradd -m USER_NAME
```

2. Set a password for that account:

```
# passwd USER_NAME
```

3. If needed, add the user to the `wheel` group:

```
# usermod -aG wheel USER_NAME
```

7 Legal Notice

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.

A GNU Free Documentation License

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or non-commercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

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

We have designed this License to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or non-commercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material

on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See 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.