Red Hat

Red Hat Enterprise Linux 10

Interactively installing RHEL over the network

Installing RHEL on several systems using network resources or on a headless system with the graphical installer

Last Updated: 2026-01-13

Red Hat Enterprise Linux 10 Interactively installing RHEL over the network

Installing RHEL on several systems using network resources or on a headless system with the graphical installer

Legal Notice

Abstract

You can install RHEL by using the graphical installer over your local network. Use this method to install RHEL on one or a few systems if you prefer the graphical interface during the installation and your systems have no peripherals, such as a display. The installation source is a server in your local network or the Red Hat content delivery network (CDN).

Table of Contents

PROVIDING FEEDBACK ON RED HAT DOCUMENTATION

We appreciate your feedback on our documentation. Let us know how we can improve it.

Submitting feedback through Jira (account required)

1. Log in to the Jira website.
2. Click **Create** in the top navigation bar
3. Enter a descriptive title in the **Summary** field.
4. Enter your suggestion for improvement in the **Description** field. Include links to the relevant parts of the documentation.
5. Click **Create** at the bottom of the dialogue.

CHAPTER 1. SYSTEM REQUIREMENTS AND SUPPORTED ARCHITECTURES

Red Hat Enterprise Linux 10 delivers a stable, secure, consistent foundation across hybrid cloud deployments with the tools needed to deliver workloads faster with less effort. You can deploy RHEL as a guest on supported hypervisors and Cloud provider environments as well as on physical infrastructure. It helps the applications take advantage of innovations in the leading hardware architecture platforms.

Review the guidelines provided for system, hardware, security, memory, and storage configuration before installing.

If you want to use your system as a virtualization host, review the necessary hardware requirements for virtualization.

RHEL supports the following architectures:

- AMD and Intel 64-bit architectures
- The 64-bit ARM architecture
- IBM Power Systems, Little Endian
- 64-bit IBM Z architectures

1.1. SUPPORTED INSTALLATION TARGETS

An installation target is a storage device that stores Red Hat Enterprise Linux and boots the system. Red Hat Enterprise Linux supports the following installation targets for IBM Z, IBM Power, AMD64, Intel 64, and 64-bit ARM systems:

- Storage connected by a standard internal interface, such as DASD, SCSI, SATA, or SAS
- BIOS/firmware RAID devices on the Intel64, AMD64 and arm64 architectures
- Fibre Channel Host Bus Adapters and multipath devices. Some can require vendor-provided drivers.
- Xen block devices on Intel processors in Xen virtual machines.
- VirtIO block devices on Intel processors in KVM virtual machines.

Red Hat does not support installation to USB drives or SD memory cards. For information about support for third-party virtualization technologies, see the Red Hat Hardware Compatibility List .

1.2. DISK AND MEMORY REQUIREMENTS

If several operating systems are installed, it is important that you verify that the allocated disk space is separate from the disk space required by Red Hat Enterprise Linux. In some cases, it is important to dedicate specific partitions to Red Hat Enterprise Linux, for example, for AMD64, Intel 64, and 64-bit ARM, at least two partitions (/ and **swap**) must be dedicated to RHEL and for IBM Power Systems servers, up to three partitions (/, **swap**, and potentially a **PReP** boot partition) must be dedicated to RHEL.

Additionally, you must have a minimum of 10 GiB of available disk space. To install Red Hat Enterprise Linux, you must have a minimum of 10 GiB of space in either unpartitioned disk space or in partitions that can be deleted.

Table 1.1. Minimum RAM requirements

Installation type	Minimum RAM
Local media installation (USB, DVD)	● 1.5 GiB for aarch64, IBM Z and x86_64 architectures ● 3 GiB for ppc64le architecture
NFS network installation	● 1.5 GiB for aarch64, IBM Z and x86_64 architectures ● 3 GiB for ppc64le architecture
HTTP, HTTPS or FTP network installation	● 3 GiB for IBM Z and x86_64 architectures ● 4 GiB for aarch64 and ppc64le architectures

It is possible to complete the installation with less memory than the minimum requirements. The exact requirements depend on your environment and installation path. Test various configurations to determine the minimum required RAM for your environment. Installing Red Hat Enterprise Linux using a Kickstart file has the same minimum RAM requirements as a standard installation. However, additional RAM may be required if your Kickstart file includes commands that require additional memory, or write data to the RAM disk. For more information, see the Automatically installing RHEL document.

1.3. GRAPHICS DISPLAY RESOLUTION REQUIREMENTS

Your system must have the following minimum resolution to ensure a smooth and error-free installation of Red Hat Enterprise Linux.

Table 1.2. Display resolution

Product version	Resolution
Red Hat Enterprise Linux 10	**Minimum**: 800 x 600 **Recommended**: 1024 x 768

1.4. UEFI SECURE BOOT AND BETA RELEASE REQUIREMENTS

If you plan to install a Beta release of Red Hat Enterprise Linux, on systems having UEFI Secure Boot enabled, first disable the UEFI Secure Boot option and then begin the installation.

UEFI Secure Boot requires that the operating system kernel is signed with a recognized private key, which the system's firmware verifies using the corresponding public key.

For Red Hat Enterprise Linux Beta releases, the kernel is signed with a Red Hat Beta-specific public key, which the system fails to recognize by default. As a result, the system fails to even boot the installation media.

Additional resources

- IBM installation documentation
- Security hardening
- Composing a customized RHEL system image
- Red Hat ecosystem catalog
- RHEL technology capabilities and limits

CHAPTER 2. THE VALUE OF REGISTERING YOUR RHEL SYSTEM TO RED HAT

Registration establishes an authorized connection between your system and Red Hat. Red Hat issues the registered system, whether a physical or virtual machine, a certificate that identifies and authenticates the system. The certificate helps to receive protected content, software updates, security patches, support, and managed services from Red Hat.

With a valid subscription, you can register a Red Hat Enterprise Linux (RHEL) system in the following ways:

- During the installation process, using an installer graphical user interface (GUI)
- After installation, using the command line interface (CLI)
- Automatically, during or after installation, using a kickstart script or an activation key

The specific steps to register your system depend on the version of RHEL that you are using and the registration method that you choose.

Registering your system to Red Hat enables features and capabilities that you can use to manage your system and report data. For example, a registered system is authorized to access protected content repositories for subscribed products through the Red Hat Content Delivery Network (CDN) or a Red Hat Satellite Server. These content repositories contain Red Hat software packages and updates, available only to customers with an active subscription. The packages and updates include security patches, bug fixes, and new features for RHEL and other Red Hat products.

CHAPTER 3. CUSTOMIZING THE INSTALLATION MEDIA

Customize RHEL installation media to create tailored system images with specific configurations, packages, and settings. It enables deployment of pre-configured systems that meet specific organizational requirements and reduces post-installation configuration time.

For details, see Composing a customized RHEL system image .

CHAPTER 4. INSTALLING RHEL USING SATELLITE SERVER

Red Hat Satellite is a centralized tool for provisioning, remote management, and monitoring of multiple Red Hat Enterprise Linux deployments. With Satellite, you can deploy physical and virtual hosts. This includes setting up network topology and configuring services. It also provides configuration details needed to provision hosts on your network.

IMPORTANT

For information about provisioning hosts, refer to the *Satellite documentation*. From the documentation portal, select the version of Satellite that matches your deployment to access the appropriate *Provisioning Hosts* guide. Documentation for unsupported versions is marked as deprecated and is intended for reference only. Verify that you are viewing a supported version to ensure access to the most accurate and up-to-date information.

CHAPTER 5. PREPARING NETWORK-BASED REPOSITORIES

You must prepare repositories to install RHEL from your network system.

5.1. CREATING AN INSTALLATION SOURCE ON AN NFS SERVER

You can configure an NFS server to host RHEL installation files for network-based installations. It enables multiple systems to install from a shared network location, providing efficient deployment across your infrastructure without requiring physical media for each system.

Prerequisites

- You have administrator-level access to a server with Red Hat Enterprise Linux 10, and this server is on the same network as the system to be installed.
- You have downloaded the full installation DVD ISO from the Product Downloads page.

IMPORTANT

Ensure that you use different paths in **inst.ks** and **inst.repo**. When using NFS to host the installation source, you cannot use the same nfs share to host the Kickstart.

Procedure

1. Install the **nfs-utils** package:

   ```
   # dnf install nfs-utils
   ```

2. Copy the DVD ISO image to a directory on the NFS server. This example assumes the DVD ISO is copied to the /**nfs/rhel10-install**/ directory on the NFS server.

3. Open the /**etc/exports** file using a text editor and add a line with the following syntax:

   ```
   /exported_directory/ clients
   ```

   - Replace */exported_directory/* with the full path to the directory with the ISO image.
   - Replace *clients* with one of the following:
     - The host name or IP address of the target system
     - The subnetwork that all target systems can use to access the ISO image
     - To allow any system with network access to the NFS server to use the ISO image, the asterisk sign (*)

   See the **exports(5)** man page for detailed information about the format of this field.

   For example, a basic configuration that makes the /**nfs/rhel10-install**/ directory available as read-only to all clients is:

   ```
   /nfs/rhel10-install *
   ```

4. Save the **/etc/exports** file and exit the text editor.

5. Enable the NFS service in firewalld:

   ```
   # firewall-cmd --permanent --add-service nfs
   # firewall-cmd --permanent --add-service=mountd
   # firewall-cmd --permanent --add-service=rpc-bind
   # firewall-cmd --reload
   ```

6. Start the nfs service:

   ```
   # systemctl enable --now nfs-server.service
   ```

   If the service was running before you changed the **/etc/exports** file, reload the NFS server configuration:

   ```
   # systemctl reload nfs-server.service
   ```

   The ISO image is now accessible over NFS and ready to be used as an installation source.

   When configuring the installation source, use **nfs:** as the protocol, the server host name or IP address, the colon sign **(:)**, and the directory holding the ISO image. For example, if the server host name is **myserver.example.com** and you have saved the ISO image in **/nfs/rhel10-install/**:

   - On kernel command line, define the NFS installation source as **inst.repo=nfs:myserver.example.com:/nfs/rhel10-install/** and

   - In the Kickstart file, define the NFS installation source as **nfs --server=myserver.example.com --dir=/nfs/rhel10-install/**.

Additional resources

- Deploying an NFS server

5.2. CREATING AN INSTALLATION SOURCE BY USING HTTP OR HTTPS

You can create an installation source for a network-based installation by using an installation tree. It is a directory containing extracted contents of the DVD ISO image and a valid **.treeinfo** file. The installation source is accessed over HTTP or HTTPS.

Prerequisites

- You have administrator-level access to a server with Red Hat Enterprise Linux 10, and this server is on the same network as the system to be installed.

- You have downloaded the full installation DVD ISO from the Product Downloads page.

IMPORTANT

If you use an HTTPS server with a self-signed certificate, you must boot the installation program with the **noverifyssl** option.

Procedure

1. Install the httpd package.

   ```
   # dnf install httpd
   ```

2. Optional: Install the **mod_ssl** package if you want to use the https installation source:

   ```
   # dnf install mod_ssl
   ```

3. Copy the DVD ISO image to the HTTP(S) server.

4. Create a suitable directory for mounting the DVD ISO image, for example:

   ```
   # mkdir /mnt/rhel10-install/
   ```

5. Mount the DVD ISO image to the directory:

   ```
   # mount -o loop,ro -t iso9660 /image_directory/image.iso /mnt/rhel10-install/
   ```

   Replace */image_directory/image.iso* with the path to the DVD ISO image.

6. Copy the files from the mounted image to the HTTP(S) server root.

   ```
   # cp -r /mnt/rhel10-install/ /var/www/html/
   ```

   This command creates the **/var/www/html/rhel10-install/** directory with the content of the image. Note that some other copying methods might skip the **.treeinfo** file which is required for a valid installation source. Entering the **cp** command for entire directories copies **.treeinfo** correctly.

7. Unmount the DVD ISO:

   ```
   # umount /mnt/rhel10-install/
   ```

8. Enable the http service in firewalld:

   ```
   # firewall-cmd --permanent --add-service=http
   # firewall-cmd --reload
   ```

9. Optional: Enable the https service in firewalld:

   ```
   # firewall-cmd --permanent --add-service=https
   # firewall-cmd --reload
   ```

10. Start the **httpd** service:

    ```
    # systemctl enable --now httpd.service
    ```

    The installation tree is now accessible and ready to be used as the installation source.

NOTE

When configuring the installation source, use **http://** or **https://** as the protocol, the server host name or IP address, and the directory that contains the files from the ISO image, relative to the HTTP server root. For example, if you use HTTP, the server host name is **myserver.example.com**, and you have copied the files from the image to **/var/www/html/rhel10-install/**, specify **http://myserver.example.com/rhel10-install/** as the installation source.

Additional resources

- Deploying different types of servers
- Setting up the Apache HTTP web server

5.3. CREATING AN INSTALLATION SOURCE BY USING FTP

You can set up an FTP server to host RHEL installation files for network-based installations. It enables multiple systems to install from a centralized source, providing efficient deployment across your network infrastructure using the FTP protocol.

Prerequisites

- You have administrator-level access to a server with Red Hat Enterprise Linux 10, and this server is on the same network as the system to be installed.
- You have downloaded the full installation DVD ISO from the Product Downloads page.
- The **vsftpd** package is installed.

Procedure

1. Install the vsftpd package.

   ```
   # dnf install vsftpd
   ```

2. Open and edit the **/etc/vsftpd/vsftpd.conf** configuration file in a text editor.

   a. Change the line **anonymous_enable=NO** to **anonymous_enable=YES**

   b. Change the line **write_enable=YES** to **write_enable=NO**.

   c. Add lines **pasv_min_port=<*min_port*>** and **pasv_max_port=<*max_port*>**. Replace <*min_port*> and <*max_port*> with the port number range used by the FTP server in passive mode, for example, **10000** and **11000**.
   This step might be necessary in network environments featuring various firewall/NAT setups.

   d. Optional: Add custom changes to your configuration. For available options, see the **vsftpd.conf(5)** man page. This procedure assumes that default options are used.

   e. Configure the firewall to allow the FTP port and port range from the previous step:

   ```
   # firewall-cmd --add-port min_port-max_port/tcp --permanent
   ```

Replace <*min_port*> and <*max_port*> with the port numbers you entered into the **/etc/vsftpd/vsftpd.conf** configuration file.

f. Configure the firewall to allow FTP service

```
# firewall-cmd --add-service ftp --permanent
```

g. Reload the firewall to apply the new rules:

```
# firewall-cmd --reload
```

3. Copy the DVD ISO image to the FTP server.

4. Create a suitable directory for mounting the DVD ISO image, for example:

```
# mkdir /mnt/rhel10-install
```

5. Mount the DVD ISO image to the directory:

```
# mount -o loop,ro -t iso9660 /image-directory/image.iso /mnt/rhel10-install
```

Replace ***/image-directory/image.iso*** with the path to the DVD ISO image.

6. Copy the files from the mounted image to the FTP server root:

```
# cp -r /mnt/rhel10-install/ /var/ftp/
```

This command creates the **/var/ftp/rhel10-install/** directory with the content of the image. Some copying methods can skip the **.treeinfo** file which is required for a valid installation source. Entering the **cp** command for whole directories as shown in this procedure will copy **.treeinfo** correctly.

7. Start the **vsftpd** service:

```
# systemctl enable --now vsftpd.service
```

If the service was running before you changed the **/etc/vsftpd/vsftpd.conf** file, restart the service to load the edited file:

```
# systemctl restart vsftpd.service
```

The installation tree is now accessible and ready to be used as the installation source.

When configuring the installation source, use **ftp://** as the protocol, the server host name or IP address, and the directory in which you have stored the files from the ISO image, relative to the FTP server root. For example, if the server host name is **myserver.example.com** and you have copied the files from the image to **/var/ftp/rhel10-install/**, specify **ftp://myserver.example.com/rhel10-install/** as the installation source.

CHAPTER 6. PREPARING A UEFI HTTP INSTALLATION SOURCE

As an administrator of a server on a local network, you can configure an HTTP server to enable HTTP boot and network installation for other systems on your network.

6.1. NETWORK INSTALLATION OVERVIEW

A network installation allows you to install Red Hat Enterprise Linux to a system that has access to an installation server. At a minimum, two systems are required for a network installation:

Server

A system running a DHCP server, an HTTP, HTTPS, FTP, or NFS server, and in the PXE boot case, a TFTP server. Although each server can run on a different physical system, the procedures in this section assume a single system is running all servers.

Client

The system to which you are installing Red Hat Enterprise Linux. Once installation starts, the client queries the DHCP server, receives the boot files from the HTTP or TFTP server, and downloads the installation environment image from the HTTP, HTTPS, FTP or NFS server. Unlike other installation methods, the client does not require any physical boot media for the installation to start.

To boot a client from the network, enable network boot in the firmware or in a quick boot menu on the client. On some hardware, the option to boot from a network might be disabled, or not available.

The workflow to prepare for an installation of Red Hat Enterprise Linux from a network by using HTTP or PXE is as follows:

1. Export the installation ISO image or the installation tree to an NFS, HTTPS, HTTP, or FTP server.

2. Configure the HTTP or TFTP server and DHCP server, and start the HTTP or TFTP service on the server.

3. Boot the client and start the installation.

You can choose between the following network boot protocols:

HTTP

Use the HTTP boot if your client UEFI supports it. HTTP boot is usually more reliable.

PXE (TFTP)

PXE boot is more widely supported by client systems, but sending the boot files over TFTP protocol might be slow and result in timeout failures.

Additional resources

- Preparing network based repositories

- Red Hat Satellite product documentation

6.2. CONFIGURING THE DHCPV4 SERVER FOR NETWORK BOOT

Enable the DHCP version 4 (DHCPv4) service on your server, so that it can provide network boot functionality.

Prerequisites

- You are preparing network installation over the IPv4 protocol.
- Find the network addresses of the server.
  In the following examples, the server has a network interface **enp1s0** with this configuration:

  IPv4 address

  192.168.124.2/24

  IPv4 gateway + DNS server

  192.168.124.1

Procedure

1. Install the DHCP server:

```
# dnf install kea
```

2. Set up a DHCPv4 server. Enter the following configuration in the **/etc/kea/kea-dhcp4.conf** file. Replace the addresses to match your network card.

```
{
 "Dhcp4": {
  "interfaces-config": {
   "interfaces": [ "enp1s0" ]
  },
  "subnet4": [
   {
    "id": 1,
    "subnet": "192.168.124.0/24",
    "pools": [
     {
      "pool": "192.168.124.100 - 192.168.124.200"
     }
    ],
    "option-data": [
     {
      "space": "dhcp4",
      "name": "routers",
      "code": 3,
      "data": "192.168.124.10"
     },
     {
      "space": "dhcp4",
      "name": "domain-name-servers",
      "code": 6,
      "data": "192.168.124.10"
     }
    ]
   }
  ],
    "client-classes": [
      {
        "name": "uefi PXE Clients",
        "test": "substring(option[60].hex,0,9) == 'PXEClient' and option[93].hex == 0x0007",
        "next-server": "192.168.124.2",
        "boot-file-name": "/uefi/BOOT/BOOTX64.EFI"
      },
      {
        "name": "bios PXE Clients",
        "test": "substring(option[60].hex,0,9) == 'PXEClient' and option[93].hex == 0x0000",
        "next-server": "192.168.124.2",
        "boot-file-name": "syslinux/pxelinux.0"
      },
      {
        "name": "uefi HTTP Clients",
        "test": "substring(option[60].hex,0,10) == 'HTTPClient' and option[93].hex == 0x0007",
        "option-data": [
          {
            "space": "dhcp4",
            "name": "vendor-class-identifier",
            "code": 60,
            "data": "HTTPClient"
          }
        ],
        "next-server": "192.168.124.2",
        "boot-file-name": "http://192.168.124.2/uefi/BOOT/BOOTX64.EFI"
      }
    ]
  }
}
```

3. Start the DHCPv4 service:

```
# systemctl enable --now kea-dhcp4
```

6.3. CONFIGURING THE DHCPV6 SERVER FOR NETWORK BOOT

Enable the DHCP version 6 (DHCPv4) service on your server, so that it can provide network boot functionality.

Prerequisites

- You are preparing network installation over the IPv6 protocol.
- Find the network addresses of the server.
  In the following examples, the server has a network interface **enp1s0** with this configuration:

  IPv6 address

  fd33::2/64

  IPv6 gateway

  fd33::1

Procedure

1. Install the DHCP server:

```
# dnf install kea
```

2. Set up a DHCPv6 server. Enter the following configuration in the **/etc/kea/kea-dhcp6.conf** file. Replace the addresses to match your network card.

```
{
  "Dhcp6": {
    "interfaces-config": {
      "interfaces": [ "enp1s0" ]
    },
    "subnet6": [
      {
        "id": 1,
        "subnet": "fd33::/64",
        "interface": "enp1s0",
        "pools": [
          {
            "pool": "fd33::100-fd33::200"
          }
        ]
      }
    ],
    "client-classes": [
      {
        "name": "uefi PXE Clients",
        "test": "substring(option[16].hex,6,9) == 'PXEClient' and substring(option[16].hex,21,5)
== '00007'",
        "option-data": [
          {
            "space": "dhcp6",
            "name": "bootfile-url",
            "code": 59,
            "data": "tftp://[fd33::2]/uefi/BOOT/BOOTX64.EFI"
          }
        ]
      },
      {
        "name": "bios PXE Clients",
        "test": "substring(option[16].hex,6,9) == 'PXEClient' and substring(option[16].hex,21,5)
== '00000'",
        "option-data": [
          {
            "space": "dhcp6",
            "name": "bootfile-url",
            "code": 59,
            "data": "tftp://[fd33::2]/syslinux/pxelinux.0"
          }
        ]
      },
      {
        "name": "uefi HTTP Clients",
        "test": "substring(option[16].hex,6,10) == 'HTTPClient' and
substring(option[16].hex,22,5) == '00007'",
        "option-data": [
     {
      "space": "dhcp6",
      "name": "bootfile-url",
      "code": 59,
      "data": "http://[fd33::2]/uefi/BOOT/BOOTX64.EFI"
     }
    ]
   }
  ]
 }
}
```

3. Start the DHCPv6 service:

```
# systemctl enable --now kea-dhcp6
```

4. If DHCPv6 packets are dropped by the RP filter in the firewall, check its log. If the log contains the **rpfilter_DROP** entry, disable the filter using the following configuration in the **/etc/firewalld/firewalld.conf** file:

```
IPv6_rpfilter=no
```

6.4. CONFIGURING THE HTTP SERVER FOR HTTP BOOT

You must install and enable the **httpd** service on your server so that the server can provide HTTP boot resources on your network.

Prerequisites

- Find the network addresses of the server.
  In the following examples, the server has a network card with the **192.168.124.2** IPv4 address.

Procedure

1. Install the HTTP server:

```
# dnf install httpd
```

2. Create the **/var/www/html/redhat** directory:

```
# mkdir -p /var/www/html/redhat
```

3. Download the RHEL DVD ISO file. See All Red Hat Enterprise Linux Downloads .

4. Create a mount point for the ISO file:

```
# mkdir -p /var/www/html/redhat/iso
```

5. Mount the ISO file:

```
# mount -o loop,ro -t iso9660 path-to-RHEL-DVD.iso /var/www/html/redhat/iso
```

6. Copy the boot loader, kernel, and **initramfs** from the mounted ISO file into your HTML directory:

```
# cp -r /var/www/html/redhat/iso/images /var/www/html/redhat
# cp -r /var/www/html/redhat/iso/EFI /var/www/html/redhat
```

7. Make the boot loader configuration editable and ensure the boot files are owned by the user running httpd server (apache):

```
# chmod 644 /var/www/html/redhat/EFI/BOOT/grub.cfg
# chown -R apache:apache /var/www/html/redhat/EFI
```

8. Edit the **/var/www/html/redhat/EFI/BOOT/grub.cfg** file and replace its content with the following:

```
set default="1"

function load_video {
  insmod efi_gop
  insmod efi_uga
  insmod video_bochs
  insmod video_cirrus
  insmod all_video
}

load_video
set gfxpayload=keep

set timeout=60

menuentry 'Install Red Hat Enterprise Linux 10.0' --class fedora --class gnu-linux --class gnu
--class os {
    linuxefi /redhat/images/pxeboot/vmlinuz inst.repo=http://192.168.124.2/redhat/iso quiet
    initrdefi /redhat/images/pxeboot/initrd.img
}
submenu 'Troubleshooting -->' {
    menuentry 'Install Red Hat Enterprise Linux 10.0 in text mode' --class fedora --class gnu-
linux --class gnu --class os {
        linuxefi /redhat/images/pxeboot/vmlinuz inst.repo=http://192.168.124.2/redhat/iso
inst.text quiet
        initrdefi /redhat/images/pxeboot/initrd.img
    }
    menuentry 'Rescue a Red Hat Enterprise Linux system' --class fedora --class gnu-linux --
class gnu --class os {
        linuxefi /redhat/images/pxeboot/vmlinuz inst.stage2=http://192.168.124.2/redhat/iso
inst.rescue quiet
        initrdefi /redhat/images/pxeboot/initrd.img
    }
}
```

In this file, update the following strings:

Install Red Hat Enterprise Linux 10.0

Edit the version number to match the version of RHEL that you downloaded.

192.168.124.2

Replace with the IP address of your server.

9. Open ports in the firewall to allow HTTP (80), DHCP (67, 68) and DHCPv6 (546, 547) traffic:

   ```
   # firewall-cmd --zone public \
                   --add-port={80/tcp,67/udp,68/udp,546/udp,547/udp}
   ```

   This command enables temporary access until the next server reboot.

10. Optional: To enable permanent access, add the **--permanent** option to the command.

11. Reload firewall rules:

    ```
    # firewall-cmd --reload
    ```

12. Start the HTTP server:

    ```
    # systemctl enable --now httpd
    ```

13. Make the **html** directory and its content readable and executable:

    ```
    # chmod -cR u=rwX,g=rX,o=rX /var/www/html
    ```

14. Restore the SELinux context of the **html** directory:

    ```
    # restorecon -FvvR /var/www/html
    ```

CHAPTER 7. PREPARING A PXE INSTALLATION SOURCE

You must configure TFTP and DHCP on a PXE server to enable PXE boot and network installation.

7.1. NETWORK INSTALLATION OVERVIEW

A network installation allows you to install Red Hat Enterprise Linux to a system that has access to an installation server. At a minimum, two systems are required for a network installation:

Server

A system running a DHCP server, an HTTP, HTTPS, FTP, or NFS server, and in the PXE boot case, a TFTP server. Although each server can run on a different physical system, the procedures in this section assume a single system is running all servers.

Client

The system to which you are installing Red Hat Enterprise Linux. Once installation starts, the client queries the DHCP server, receives the boot files from the HTTP or TFTP server, and downloads the installation environment image from the HTTP, HTTPS, FTP or NFS server. Unlike other installation methods, the client does not require any physical boot media for the installation to start.

To boot a client from the network, enable network boot in the firmware or in a quick boot menu on the client. On some hardware, the option to boot from a network might be disabled, or not available.

The workflow to prepare for an installation of Red Hat Enterprise Linux from a network by using HTTP or PXE is as follows:

1. Export the installation ISO image or the installation tree to an NFS, HTTPS, HTTP, or FTP server.

2. Configure the HTTP or TFTP server and DHCP server, and start the HTTP or TFTP service on the server.

3. Boot the client and start the installation.

You can choose between the following network boot protocols:

HTTP

Use the HTTP boot if your client UEFI supports it. HTTP boot is usually more reliable.

PXE (TFTP)

PXE boot is more widely supported by client systems, but sending the boot files over TFTP protocol might be slow and result in timeout failures.

Additional resources

- Preparing network based repositories

- Red Hat Satellite product documentation

7.2. CONFIGURING THE DHCPV4 SERVER FOR NETWORK BOOT

Enable the DHCP version 4 (DHCPv4) service on your server, so that it can provide network boot functionality.

Prerequisites

- You are preparing network installation over the IPv4 protocol.
- Find the network addresses of the server.
  In the following examples, the server has a network interface **enp1s0** with this configuration:

  IPv4 address

  192.168.124.2/24

  IPv4 gateway + DNS server

  192.168.124.1

Procedure

1. Install the DHCP server:

```
# dnf install kea
```

2. Set up a DHCPv4 server. Enter the following configuration in the **/etc/kea/kea-dhcp4.conf** file. Replace the addresses to match your network card.

```
{
  "Dhcp4": {
    "interfaces-config": {
      "interfaces": [ "enp1s0" ]
    },
    "subnet4": [
      {
        "id": 1,
        "subnet": "192.168.124.0/24",
        "pools": [
          {
            "pool": "192.168.124.100 - 192.168.124.200"
          }
        ],
        "option-data": [
          {
            "space": "dhcp4",
            "name": "routers",
            "code": 3,
            "data": "192.168.124.10"
          },
          {
            "space": "dhcp4",
            "name": "domain-name-servers",
            "code": 6,
            "data": "192.168.124.10"
          }
        ]
      }
    ],
    "client-classes": [
      {
        "name": "uefi PXE Clients",
        "test": "substring(option[60].hex,0,9) == 'PXEClient' and option[93].hex == 0x0007",
        "next-server": "192.168.124.2",
        "boot-file-name": "/uefi/BOOT/BOOTX64.EFI"
      },
      {
        "name": "bios PXE Clients",
        "test": "substring(option[60].hex,0,9) == 'PXEClient' and option[93].hex == 0x0000",
        "next-server": "192.168.124.2",
        "boot-file-name": "syslinux/pxelinux.0"
      },
      {
        "name": "uefi HTTP Clients",
        "test": "substring(option[60].hex,0,10) == 'HTTPClient' and option[93].hex == 0x0007",
        "option-data": [
          {
            "space": "dhcp4",
            "name": "vendor-class-identifier",
            "code": 60,
            "data": "HTTPClient"
          }
        ],
        "next-server": "192.168.124.2",
        "boot-file-name": "http://192.168.124.2/uefi/BOOT/BOOTX64.EFI"
      }
    ]
  }
}
```

3. Start the DHCPv4 service:

```
# systemctl enable --now kea-dhcp4
```

7.3. CONFIGURING THE DHCPV6 SERVER FOR NETWORK BOOT

Enable the DHCP version 6 (DHCPv4) service on your server, so that it can provide network boot functionality.

Prerequisites

- You are preparing network installation over the IPv6 protocol.
- Find the network addresses of the server.
  In the following examples, the server has a network interface **enp1s0** with this configuration:

  IPv6 address

  fd33::2/64

  IPv6 gateway

  fd33::1

Procedure

1. Install the DHCP server:

```
# dnf install kea
```

2. Set up a DHCPv6 server. Enter the following configuration in the **/etc/kea/kea-dhcp6.conf** file. Replace the addresses to match your network card.

```
{
  "Dhcp6": {
    "interfaces-config": {
      "interfaces": [ "enp1s0" ]
    },
    "subnet6": [
      {
        "id": 1,
        "subnet": "fd33::/64",
        "interface": "enp1s0",
        "pools": [
          {
            "pool": "fd33::100-fd33::200"
          }
        ]
      }
    ],
    "client-classes": [
      {
        "name": "uefi PXE Clients",
        "test": "substring(option[16].hex,6,9) == 'PXEClient' and substring(option[16].hex,21,5)
== '00007'",
        "option-data": [
          {
            "space": "dhcp6",
            "name": "bootfile-url",
            "code": 59,
            "data": "tftp://[fd33::2]/uefi/BOOT/BOOTX64.EFI"
          }
        ]
      },
      {
        "name": "bios PXE Clients",
        "test": "substring(option[16].hex,6,9) == 'PXEClient' and substring(option[16].hex,21,5)
== '00000'",
        "option-data": [
          {
            "space": "dhcp6",
            "name": "bootfile-url",
            "code": 59,
            "data": "tftp://[fd33::2]/syslinux/pxelinux.0"
          }
        ]
      },
      {
        "name": "uefi HTTP Clients",
        "test": "substring(option[16].hex,6,10) == 'HTTPClient' and
substring(option[16].hex,22,5) == '00007'",
        "option-data": [
          {
            "space": "dhcp6",
            "name": "bootfile-url",
            "code": 59,
              "data": "http://[fd33::2]/uefi/BOOT/BOOTX64.EFI"
            }
          ]
        }
      ]
    }
  }
```

3. Start the DHCPv6 service:

```
# systemctl enable --now kea-dhcp6
```

4. If DHCPv6 packets are dropped by the RP filter in the firewall, check its log. If the log contains the **rpfilter_DROP** entry, disable the filter using the following configuration in the **/etc/firewalld/firewalld.conf** file:

```
IPv6_rpfilter=no
```

7.4. CONFIGURING A TFTP SERVER FOR BIOS-BASED CLIENTS

You must configure a TFTP server and DHCP server and start the TFTP service on the PXE server for BIOS-based AMD and Intel 64-bit systems to enable automated network booting.

Procedure

1. As root, install the following package.

```
# dnf install tftp-server
```

2. Allow incoming connections to the **tftp service** in the firewall:

```
# firewall-cmd --add-service=tftp
```

This command enables temporary access until the next server reboot.

3. Optional: To enable permanent access, repeat the previous command together with the **--permanent** option.
Depending on the location of the installation ISO file, you might have to allow incoming connections for HTTP or other services.

4. Install **syslinux-tftpboot** package:

```
# dnf -y install syslinux-tftpboot
```

5. Create a **pxelinux** directory in the **/var/lib/tftpboot** directory and copy all the files provided by the **syslinux-tftpboot** package (located in **/tftpboot** directory) into the **pxelinux** directory:

```
# mkdir /var/lib/tftpboot/pxelinux
# cp /tftpboot/* /var/lib/tftpboot/pxelinux
```

6. Create the directory **pxelinux.cfg** in the **pxelinux** directory:

```
# mkdir /var/lib/tftpboot/pxelinux/pxelinux.cfg
```

7. Create a configuration file named **default** and add it to the **pxelinux.cfg** directory as shown in the following example:

```
DEFAULT menu.c32

MENU TITLE Red Hat Enterprise Linux 10.0 Installation
TIMEOUT 600

LABEL default
 MENU LABEL ^Install Red Hat Enterprise Linux 10.0
 KERNEL /pxelinux/images/RHEL-10/vmlinuz
 APPEND initrd=/pxelinux/images/RHEL-10/initrd.img
inst.repo=http://192.168.124.2/redhat/iso quiet

LABEL text
 MENU LABEL Install Red Hat Enterprise Linux 10.0 in ^text mode
 KERNEL /pxelinux/images/RHEL-10/vmlinuz
 APPEND initrd=/pxelinux/images/RHEL-10/initrd.img
inst.repo=http://192.168.124.2/redhat/iso inst.text quiet

LABEL rescue
 MENU LABEL ^Rescue a Red Hat Enterprise Linux system
 KERNEL /pxelinux/images/RHEL-10/vmlinuz
 APPEND initrd=/pxelinux/images/RHEL-10/initrd.img
inst.stage2=http://192.168.124.2/redhat/iso inst.rescue quiet

LABEL local
 MENU LABEL Boot from ^local drive
 LOCALBOOT 0xffff
```

- The installation program cannot boot without its runtime image. Use the **inst.stage2** boot option to specify location of the image. Alternatively, you can use the **inst.repo=** option to specify the image as well as the installation source.

- The installation source location used with **inst.repo** must contain a valid **.treeinfo** file.

- When you select the content of RHEL10 installation DVD as the installation source, the **.treeinfo** file points to the BaseOS and the AppStream repositories. You can use a single **inst.repo** option to load both repositories.

8. Create a subdirectory to store the boot image files in the **/var/lib/tftpboot** directory, and copy the boot image files to the directory. In this example, the directory is **/var/lib/tftpboot/pxelinux/images/RHEL-10**:

```
# mkdir -p /var/lib/tftpboot/pxelinux/images/RHEL-10
# cp /path_to_x86_64_images/pxeboot/{vmlinuz,initrd.img}
/var/lib/tftpboot/pxelinux/images/RHEL-10/
```

9. Start and enable the **tftp.socket** service:

```
# systemctl enable --now tftp.socket
```

The PXE boot server is now ready to serve PXE clients. You can start the client, which is the system to which you are installing Red Hat Enterprise Linux. Select **PXE Boot** or **Network Boot** when prompted to specify a boot source, and start the network installation.

7.5. CONFIGURING A TFTP SERVER FOR UEFI-BASED CLIENTS

You must configure a TFTP server and DHCP server and start the TFTP service on the PXE server for UEFI-based AMD64, Intel 64, and 64-bit ARM systems.

IMPORTANT

Red Hat Enterprise Linux 10 UEFI PXE boot supports, apart from the default grub menu file (grub.cfg), a lowercase file format for a MAC-based grub menu file. For example, the MAC address file format for grub2 can be **grub.cfg-01-aa-bb-cc-dd-ee-ff**

Procedure

1. As root, install the following package.

```
# dnf install tftp-server
```

2. Allow incoming connections to the **tftp service** in the firewall:

```
# firewall-cmd --add-service=tftp
```

This command enables temporary access until the next server reboot.

3. Optional: To enable permanent access, repeat the previous command together with the **--permanent** option.
Depending on the location of the installation ISO file, you might have to allow incoming connections for HTTP or other services.

4. Access the EFI boot image files from the DVD ISO image:

```
# mount -t iso9660 /path_to_image/name_of_image.iso /mount_point -o loop,ro
```

5. Copy the EFI boot images from the DVD ISO image:

```
# mkdir /var/lib/tftpboot/redhat
# cp -r /mount_point/EFI /var/lib/tftpboot/redhat/
```

6. Create a subdirectory to store the boot image files in the **/var/lib/tftpboot/** directory, and copy the boot image files to the directory. In this example, the directory is **/var/lib/tftpboot/images/RHEL-10/**:

```
# mkdir -p /var/lib/tftpboot/images/RHEL-10/
# cp /mount_point/images/pxeboot/{vmlinuz,initrd.img} /var/lib/tftpboot/images/RHEL-10/
# umount /mount_point
```

7. Fix the permissions of the copied files:

```
# chmod -R 755 /var/lib/tftpboot/redhat/
```

8. Replace the content of **/var/lib/tftpboot/redhat/EFI/BOOT/grub.cfg** with the following example:

   ```
   set default="1"
   set timeout=60

   function load_video {
     insmod efi_gop
     insmod efi_uga
     insmod video_bochs
     insmod video_cirrus
     insmod all_video
   }

   load_video
   set gfxpayload=keep

   menuentry 'Install Red Hat Enterprise Linux 10.0' --class fedora --class gnu-linux --class gnu --class os {
       linuxefi /images/RHEL-10/vmlinuz inst.repo=http://192.168.124.2/redhat/iso quiet
       initrdefi /images/RHEL-10/initrd.img
   }
   submenu 'Troubleshooting -->' {
       menuentry 'Install Red Hat Enterprise Linux 10.0 in text mode' --class fedora --class gnu-linux --class gnu --class os {
           linuxefi /images/RHEL-10/vmlinuz inst.repo=http://192.168.124.2/redhat/iso inst.text quiet
           initrdefi /images/RHEL-10/initrd.img
       }
       menuentry 'Rescue a Red Hat Enterprise Linux system' --class fedora --class gnu-linux --class gnu --class os {
           linuxefi /images/RHEL-10/vmlinuz inst.stage2=http://192.168.124.2/redhat/iso inst.rescue quiet
           initrdefi /images/RHEL-10/initrd.img
       }
   }
   ```

   - The installation program cannot boot without its runtime image. Use the **inst.stage2** boot option to specify location of the image. Alternatively, you can use the **inst.repo=** option to specify the image as well as the installation source.
   - The installation source location used with **inst.repo** must contain a valid **.treeinfo** file.
   - When you select the RHEL10 installation DVD as the installation source, the **.treeinfo** file points to the BaseOS and the AppStream repositories. You can use a single **inst.repo** option to load both repositories.

9. Start and enable the **tftp.socket** service:

   ```
   # systemctl enable --now tftp.socket
   ```

   The PXE boot server is now ready to serve PXE clients. You can start the client, which is the system to which you are installing Red Hat Enterprise Linux. Select **PXE Boot** or **Network Boot** when prompted to specify a boot source, and start the network installation.

Additional resources

- Using the Shim Program

CHAPTER 8. RHEL INSTALLATIONS ON IBM POWER SERVERS

You can install Red Hat Enterprise Linux on various IBM Power System servers.

8.1. SUPPORTED IBM POWER SERVERS

You can install Red Hat Enterprise Linux on IBM Power Systems. You can find the complete list of supported IBM Power servers on the Red Hat Ecosystem Catalog .

8.2. OVERVIEW OF THE INSTALLATION PROCESS ON POWERVM LPAR BY USING THE HMC

You can install RHEL on the PowerVM logical partition (LPAR) by using the Hardware Management Console. A Hardware Management Console (HMC) is a hardware appliance that you can use to administer IBM Power Systems servers.

The installation workflow involves the following general steps:

1. Download the RHEL installation ISO.
2. Prepare a bootable physical installation medium based on your installation method.
3. Verify that the Power System is added to the HMC.
   For more information, see add or remove connections to HMC in the IBM documentation.
4. Configure VIOS and LPAR on the managed system or configure full system LPAR based on the requirements.
5. Log in to the HMC console.
   a. Run ssh **hscroot@*<HMC_IP_address>*** to access the LPAR console by using either the **mkvterm -m *<managed_system_name>* -p *<lpar_name>*** command or the **vtmenu** interface.
   b. Power on the LPAR by using the HMC GUI.
   c. Optional: Switch to a remote desktop session.
6. Install Red Hat Enterprise Linux.
7. Optional: Install IBM Tools Repository to use Service and Productivity tools, IBM Advance Toolchain for Linux on Power, and IBM SDK for PowerLinux.

For detailed instructions, see Installing Linux on PowerVM LPAR by using the HMC and Service and productivity tools in the IBM documentation.

8.3. OVERVIEW OF THE INSTALLATION PROCESS ON IBM POWER SERVERS WITH THE GRAPHICS CARD

You can install RHEL on IBM Power Systems servers with the graphics card.

The installation workflow involves the following general steps:

1. Download the RHEL installation ISO.

2. Prepare a bootable physical installation medium based on your installation method.
3. Prepare the machine for RHEL installation.
4. Boot the installer kernel.
5. Install Red Hat Enterprise Linux.
6. Optional: Install IBM Tools Repository to use Service and Productivity tools, IBM AdvanceD Toolchain for Linux on Power, and IBM SDK for PowerLinux.

For detailed instructions, see Installing Linux on Power Systems servers with a graphics card and Service and productivity tools in the IBM documentation.

Additional resources

- IBM Knowledge Center

CHAPTER 9. PREPARING A RHEL INSTALLATION ON 64-BIT IBM Z

You can install Red Hat Enterprise Linux on the 64-bit IBM Z architecture.

9.1. PLANNING FOR INSTALLATION ON 64-BIT IBM Z

Red Hat Enterprise Linux 10 runs on **IBM z14** or **IBM LinuxONE II** systems, or later.

The installation process assumes that you are familiar with the 64-bit IBM Z and can set up *logical partitions* (LPARs) and z/VM guest virtual machines. For installation of Red Hat Enterprise Linux on 64-bit IBM Z, Red Hat supports Direct Access Storage Device (DASD), SCSI disk devices attached over Fiber Channel Protocol (FCP), and **virtio-blk** and **virtio-scsi** devices. When using FCP devices, use them in multipath configuration for better reliability.

IMPORTANT

DASDs are disks that allow a maximum of three partitions per device. For example, **dasda** can have partitions **dasda1**, **dasda2**, and **dasda3**.

Pre-installation decisions

- Whether the operating system is to be run on an LPAR, KVM, or as a z/VM guest operating system.
- Network configuration. Red Hat Enterprise Linux 10 for 64-bit IBM Z supports the following network devices:
  - Real and virtual *Open Systems Adapter* (OSA)
  - Real and virtual HiperSockets
  - **virtio-net** devices
  - RDMA over Converged Ethernet (RoCE)
- Ensure you select machine type as ***ESA*** for your z/VM VMs, because selecting any other machine types might prevent RHEL from installing. See the *IBM documentation*.

NOTE

When initializing swap space on a Fixed Block Architecture (FBA) DASD by using the **SWAPGEN** utility, the **FBAPART** option must be used.

Additional resources

- RHEL Technology Capabilities and Limits
- IBM documentation
- Secure boot for Linux on IBM Z
- IBM installation documentation

- Red Hat Ecosystem Catalog

9.2. BOOT MEDIA COMPATIBILITY FOR IBM Z SERVERS

The following table provides detailed information about the supported boot media options for installing Red Hat Enterprise Linux (RHEL) on 64-bit IBM Z servers. It outlines the compatibility of each boot medium with different system types and indicates whether the **zipl** boot loader is used. This information helps you determine the most suitable boot medium for your specific environment.

System type / Boot media	Uses zipl boot loader	z/VM	KVM	LPAR
z/VM Reader	No	Yes	N/A	N/A
SE or HMC (remote SFTP, FTPS, FTP server, DVD)	No	N/A	N/A	Yes
DASD	Yes	Yes	Yes	Yes
FCP SCSI LUNs	Yes	Yes	Yes	Yes
FCP SCSI DVD	Yes	Yes	Yes	Yes

N/A indicates that the boot medium is not applicable for the specified system type.

9.3. SUPPORTED ENVIRONMENTS AND COMPONENTS FOR IBM Z SERVERS

The following tables provide information about the supported environments, network devices, machine types, and storage types for different system types when installing Red Hat Enterprise Linux (RHEL) on 64-bit IBM Z servers. Use these tables to identify the compatibility of various components with your specific system configuration.

Table 9.1. Network device compatibility for system types

Network device	z/VM	KVM	LPAR
Open Systems Adapter (OSA)	Yes	N/A	Yes
HiperSockets	Yes	N/A	Yes
virtio-net	N/A	Yes	N/A
RDMA over Converged Ethernet (RoCE)	Yes	Yes	Yes

N/A indicates that the component is not applicable for the specified system type.

Table 9.2. Machine type compatibility for system types

Machine type	z/VM	KVM	LPAR
ESA	Yes	N/A	N/A
s390-virtio-ccw	N/A	Yes	N/A

N/A indicates that the component is not applicable for the specified system type.

Table 9.3. Storage type compatibility for system types

Storage type	z/VM	KVM	LPAR
DASD	Yes	Yes	Yes
FCP SCSI	Yes	Yes[a]	Yes
virtio-blk	N/A	Yes	N/A

[a] Conditional support based on configuration

N/A indicates that the component is not applicable for the specified system type.

9.4. OVERVIEW OF INSTALLATION PROCESS ON 64-BIT IBM Z SERVERS

You can install Red Hat Enterprise Linux on 64-bit IBM Z interactively or in unattended mode. Installation on 64-bit IBM Z differs from other architectures as it is typically performed over a network, and not from local media. The installation consists of three phases:

1. Booting the installation.
   - Connect to the mainframe.
   - Customize the boot parameters.
   - Perform an initial program load (IPL), or boot from the media containing the installation program.
2. Connecting to the installation system.
   - From a local machine, connect to the remote 64-bit IBM Z system by using SSH, and start the installation program by using Remote Desktop Protocol (RDP).
3. Completing the installation by using the RHEL installation program.

9.5. BOOT MEDIA FOR INSTALLING RHEL ON 64-BIT IBM Z SERVERS

After establishing a connection with the mainframe, you need to perform an initial program load (IPL), or boot, from the medium containing the installation program. In general, any method may be used to boot the Linux installation system, which consists of a kernel (**kernel.img**) and initial RAM disk (**initrd.img**) with parameters in the **generic.prm** file supplemented by user defined parameters. Additionally, a **generic.ins** file is loaded which determines file names and memory addresses for the initrd, kernel and **generic.prm**.

The Linux installation system is also called the *installation program* in this book.

You can use the following boot media only if Linux is to run as a guest operating system under z/VM:

- z/VM reader

You can use the following boot media only if Linux is to run in LPAR mode:

- SE or HMC through a remote SFTP, FTPS or FTP server
- SE or HMC DVD

You can use the following boot media for both z/VM and LPAR:

- DASD
- SCSI disk device that is attached through an FCP channel

If you use DASD or an FCP-attached SCSI disk device as boot media, you must have a configured **zipl** boot loader.

9.6. CUSTOMIZATION OF BOOT PARAMETERS

Before the installation begins, you must configure some mandatory boot parameters. When installing through **z/VM**, these parameters must be configured before you boot into the **generic.prm** file. When installing on an **LPAR**, the **rd.cmdline** parameter is set to **ask** by default. It means that you will be given a prompt to enter these boot parameters. In both cases, the required parameters are the same.

All network configuration can either be specified by using a parameter file, or at the prompt.

Installation source

An installation source must always be configured.

You can use the **inst.repo** option to specify the package source for the installation.

Network devices

Network configuration must be provided if network access will be required during the installation. If you plan to perform an unattended (Kickstart-based) installation by using only local media such as a disk, network configuration can be omitted.

ip=

You can use the **ip=** option for basic network configuration, and other options as required.

rd.znet=

Also, you can use the **rd.znet=** kernel option, which takes a network protocol type. It is a comma delimited list of sub-channels, and, optionally, comma delimited **sysfs** parameter and value pairs for qeth devices. This parameter can be specified multiple times to activate multiple network devices.
For example:

```
rd.znet=qeth,0.0.0600,0.0.0601,0.0.0602,layer2=1,portno=<number>
```

- When specifying multiple **rd.znet** boot options, only the last one is passed on to the kernel command line of the installed system. This does not affect the networking of the system since all network devices configured during installation are properly activated and configured at boot.

  The qeth device driver assigns the same interface name for Ethernet and Hipersockets devices: **enc*<device number>***. The bus ID is composed of the channel subsystem ID, subchannel set ID, and device number, separated by dots; the device number is the last part of the bus ID, without leading zeroes and dots. For example, the interface name will be **enca00** for a device with the bus ID **0.0.0a00**.

Storage devices

At least one storage device must always be configured for text mode installations.
The **rd.dasd=** option takes a Direct Access Storage Device (DASD) adapter device bus identifier. For multiple DASDs, specify the parameter multiple times, or use a comma separated list of bus IDs. To specify a range of DASDs, specify the first and the last bus ID.

For example:

```
rd.dasd=0.0.0200 rd.dasd=0.0.0202(ro),0.0.0203(ro:failfast),0.0.0205-0.0.0207
```

All DASDs that were specified using the **rd.dasd=** kernel argument during the installation process are automatically activated on the installed system. This activation occurs regardless of whether the DASDs were used for partitioning during the installation. All disks made visible to the installer via **rd.dasd=** will be online and available on the first boot of the installed system.

The **rd.zfcp=** option takes a SCSI over FCP (zFCP) adapter device bus identifier, a target world wide port name (WWPN), and an FCP LUN, then activates one path to a SCSI disk. This parameter needs to be specified at least twice to activate multiple paths to the same disk. This parameter can be specified multiple times to activate multiple disks, each with multiple paths.

Since 10, a target world wide port name (WWPN) and an FCP LUN have to be provided only if the **zFCP** device is not configured in NPIV mode or when **auto LUN** scanning is disabled by the **zfcp.allow_lun_scan=0** kernel module parameter. It provides access to all SCSI devices found in the storage area network attached to the FCP device with the specified bus ID. This parameter needs to be specified at least twice to activate multiple paths to the same disks.

```
rd.zfcp=0.0.4000,0x5005076300C213e9,0x5022000000000000
rd.zfcp=0.0.4000
```

Kickstart options

If you are using a Kickstart file to perform an automatic installation, you must always specify the location of the Kickstart file by using the **inst.ks=** option. For an unattended, fully automatic Kickstart installation, the **inst.cmdline** option is also useful. An example customized **generic.prm** file containing all mandatory parameters look similar to the following example:

```
ro ramdisk_size=40000 cio_ignore=all,!condev
inst.repo=http://example.com/path/to/repository
rd.znet=qeth,0.0.0600,0.0.0601,0.0.0602,layer2=1,portno=0,portname=foo
ip=192.168.17.115::192.168.17.254:24:foobar.systemz.example.com:enc600:none
nameserver=192.168.17.1
rd.dasd=0.0.0200 rd.dasd=0.0.0202
rd.zfcp=0.0.4000,0x5005076300c213e9,0x5022000000000000
rd.zfcp=0.0.5000,0x5005076300dab3e9,0x5022000000000000
inst.ks=http://example.com/path/to/kickstart
```

Some installation methods also require a file with a mapping of the location of installation data in the file system of the HMC DVD or FTP server and the memory locations where the data is to be copied.

The file is typically named **generic.ins**, and contains file names for the initial RAM disk, kernel image, and parameter file (**generic.prm**) and a memory location for each file. An example **generic.ins** will look similar to the following example:

```
images/kernel.img 0x00000000
images/initrd.img 0x02000000
images/genericdvd.prm 0x00010480
images/initrd.addrsize 0x00010408
```

A valid **generic.ins** file is provided by Red Hat along with all other files required to boot the installer. Modify this file only if you want to, for example, load a different kernel version than default.

9.7. PREPARING AN INSTALLATION IN A Z/VM GUEST VIRTUAL MACHINE

Use the **x3270** or **c3270** terminal emulator, to log in to z/VM from other Linux systems, or use the IBM 3270 terminal emulator on the 64-bit IBM Z Hardware Management Console (HMC). If you are running Microsoft Windows operating system, there are several options available, and can be found through an internet search. A free native Windows port of **c3270** called **wc3270** also exists.

Ensure you select machine type as ***ESA*** for your z/VM VMs, because selecting any other machine types might prevent installing RHEL. See the *IBM documentation*.

Procedure

1. Log on to the z/VM guest virtual machine chosen for the Linux installation.

2. optional: If your 3270 connection is interrupted and you cannot log in again because the previous session is still active, you can replace the old session with a new one by entering the following command on the z/VM logon screen:

   ```
   logon user here
   ```

   Replace *user* with the name of the z/VM guest virtual machine. Depending on whether an external security manager, for example RACF, is used, the logon command might vary.

3. If you are not already running **CMS** (single-user operating system shipped with z/VM) in your guest, boot it now by entering the command:

   ```
   cp ipl cms
   ```

4. Be sure not to use CMS disks such as your A disk (often device number 0191) as installation targets. To find out which disks are in use by CMS, use the following query:

   ```
   query disk
   ```

5. You can use the following CP (z/VM Control Program, which is the z/VM hypervisor) query commands to find out about the device configuration of your z/VM guest virtual machine:

    a. Query the available main memory, which is called *storage* in 64-bit IBM Z terminology. Your guest should have at least 1 GiB of main memory.

    ```
    cp query virtual storage
    ```

    b. Query available network devices by type:

    osa

    OSA - CHPID type OSD, real or virtual (VSWITCH or GuestLAN), both in QDIO mode

    hsi

    HiperSockets - CHPID type IQD, real or virtual (GuestLAN type Hipers)

    lcs

    LCS - CHPID type OSE

    For example, to query all of the network device types mentioned above, run:

    ```
    cp query virtual osa
    ```

    c. Query available DASDs. Only those that are flagged **RW** for read-write mode can be used as installation targets:

    ```
    cp query virtual dasd
    ```

    d. Query available FCP devices (vHBAs):

    ```
    cp query virtual fcp
    ```

9.8. PARAMETERS AND CONFIGURATION FILES ON 64-BIT IBM Z

You can customize and configure RHEL by using the parameters and configuration files on 64-bit IBM Z.

9.8.1. Required configuration file parameters on 64-bit IBM Z

Several parameters are required and must be included in the parameter file. These parameters are also provided in the file **generic.prm** in directory **images/** of the installation DVD.

- **ro**
  Mounts the root file system, which is a RAM disk, read-only.

- **ramdisk_size=*size***
  Modifies the memory size reserved for the RAM disk to ensure that the Red Hat Enterprise Linux installation program fits within it. For example: **ramdisk_size=40000**.

The **generic.prm** file also contains the additional parameter cio_ignore=all,!condev. This setting speeds up boot and device detection on systems with many devices. The installation program transparently handles the activation of ignored devices.

9.8.2. 64-bit IBM z/VM configuration file

Under z/VM, you can use a configuration file on a CMS-formatted disk. The purpose of the CMS configuration file is to save space in the parameter file by moving the parameters that configure the initial network setup, the DASD, and the FCP specification out of the parameter file.

Each line of the CMS configuration file contains a single variable and its associated value, in the following shell-style syntax: ***variable=value***.

You must also add the **CMSDASD** and **CMSCONFFILE** parameters to the parameter file. These parameters point the installation program to the configuration file:

CMSDASD=*cmsdasd_address*

Where *cmsdasd_address* is the device number of a CMS-formatted disk that contains the configuration file. This is usually the CMS user's **A** disk.
For example: **CMSDASD=191**

CMSCONFFILE=*configuration_file*

Where *configuration_file* is the name of the configuration file. **This value must be specified in lower case.** It is specified in a Linux file name format: ***CMS_file_name.CMS_file_type***.
The CMS file **REDHAT CONF** is specified as **redhat.conf**. The CMS file name and the file type can each be from one to eight characters that follow the CMS conventions.

For example: **CMSCONFFILE=redhat.conf**

9.8.3. Installation network, DASD and FCP parameters on 64-bit IBM Z

These parameters can be used to automatically set up the preliminary network, and can be defined in the CMS configuration file. These parameters are the only parameters that can also be used in a CMS configuration file. All other parameters in other sections must be specified in the parameter file.

NETTYPE="*type*"

Where *type* must be one of the following: **qeth**, **lcs**, or **ctc**. The default is **qeth**.
Choose **qeth** for:

- OSA-Express features
- HiperSockets
- Virtual connections on z/VM, including VSWITCH and Guest LAN
  Select **ctc** for:
- Channel-to-channel network connections

SUBCHANNELS="*device_bus_IDs*"

Where *device_bus_IDs* is a comma-separated list of two or three device bus IDs. The IDs must be specified in lowercase.
Provides required device bus IDs for the various network interfaces:

> qeth: SUBCHANNELS="*read_device_bus_id*,*write_device_bus_id*,*data_device_bus_id*"
> lcs or ctc: SUBCHANNELS="*read_device_bus_id*,*write_device_bus_id*"

For example (a sample qeth SUBCHANNEL statement):

```
SUBCHANNELS="0.0.f5f0,0.0.f5f1,0.0.f5f2"
```

PORTNO="*portnumber*"

You can add either **PORTNO="0"** (to use port 0) or **PORTNO="1"** (to use port 1 of OSA features with two ports per CHPID).

LAYER2="*value*"

Where *value* can be **0** or **1**.
Use **LAYER2="0"** to operate an OSA or HiperSockets device in layer 3 mode (**NETTYPE="qeth"**). Use **LAYER2="1"** for layer 2 mode. For virtual network devices under z/VM this setting must match the definition of the GuestLAN or VSWITCH to which the device is coupled.

To use network services that operate on layer 2 (the Data Link Layer or its MAC sublayer) such as DHCP, layer 2 mode is a good choice.

The qeth device driver default for OSA devices is now layer 2 mode. To continue using the previous default of layer 3 mode, set **LAYER2="0"** explicitly.

VSWITCH="*value*"

Where *value* can be **0** or **1**.
Specify **VSWITCH="1"** when connecting to a z/VM VSWITCH or GuestLAN, or **VSWITCH="0"** (or nothing at all) when using directly attached real OSA or directly attached real HiperSockets.

MACADDR="*MAC_address*"

If you specify **LAYER2="1"** and **VSWITCH="0"**, you can optionally use this parameter to specify a MAC address. Linux requires six colon-separated octets as pairs lower case hex digits - for example, **MACADDR=62:a3:18:e7:bc:5f**. This is different from the notation used by z/VM.
If you specify **LAYER2="1"** and **VSWITCH="1"**, you must not specify the **MACADDR**, because z/VM assigns a unique MAC address to virtual network devices in layer 2 mode.

CTCPROT="*value*"

Where *value* can be **0**, **1**, or **3**.
Specifies the CTC protocol for **NETTYPE="ctc"**. The default is **0**.

HOSTNAME="*string*"

Where *string* is the host name of the newly-installed Linux instance.

IPADDR="*IP*"

Where *IP* is the IP address of the new Linux instance.

NETMASK="*netmask*"

Where *netmask* is the netmask.
The netmask supports the syntax of a prefix integer (from 1 to 32) as specified in IPv4 *classless interdomain routing* (CIDR). For example, you can specify **24** instead of **255.255.255.0**, or **20** instead of **255.255.240.0**.

GATEWAY="*gw*"

Where *gw* is the gateway IP address for this network device.

MTU="*mtu*"

Where *mtu* is the *Maximum Transmission Unit* (MTU) for this network device.

DNS="*server1:server2:additional_server_terms:serverN*"

Where "*server1:server2:additional_server_terms:serverN*" is a list of DNS servers, separated by colons. For example:

```
DNS="10.1.2.3:10.3.2.1"
```

SEARCHDNS="*domain1:domain2:additional_dns_terms:domainN*"

Where "*domain1:domain2:additional_dns_terms:domainN*" is a list of the search domains, separated by colons. For example:

```
SEARCHDNS="subdomain.domain:domain"
```

You only need to specify **SEARCHDNS=** if you specify the **DNS=** parameter.

DASD=

Defines the DASD or range of DASDs to configure for the installation.
The installation program supports a comma-separated list of device bus IDs, or ranges of device bus IDs with the optional attributes **ro**, **diag**, **erplog**, and **failfast**. Optionally, you can abbreviate device bus IDs to device numbers with leading zeros stripped. Any optional attributes should be separated by colons and enclosed in parentheses. Optional attributes follow a device bus ID or a range of device bus IDs.

The only supported global option is **autodetect**. This does not support the specification of non-existent DASDs to reserve kernel device names for later addition of DASDs. Use persistent DASD device names such as **/dev/disk/by-path/name** to enable transparent addition of disks later. Other global options such as **probeonly**, **nopav**, or **nofcx** are not supported by the installation program.

Only specify those DASDs that need to be installed on your system. All unformatted DASDs specified here must be formatted after a confirmation later on in the installation program.

Add any data DASDs that are not needed for the root file system or the **/boot** partition after installation.

For example:

```
DASD="eb1c,0.0.a000-0.0.a003,eb10-eb14(diag),0.0.ab1c(ro:diag)"
```

FCP_*n*="*device_bus_ID* [*WWPN FCP_LUN*]"

For FCP-only environments, remove the **DASD=** option from the CMS configuration file to indicate no DASD is present.

```
FCP_n="device_bus_ID [WWPN FCP_LUN]"
```

Where:

- *n* is typically an integer value (for example **FCP_1** or **FCP_2**) but could be any string with alphabetic or numeric characters or underscores.

- *device_bus_ID* specifies the device bus ID of the FCP device representing the *host bus adapter* (HBA) (for example **0.0.fc00** for device fc00).
- *WWPN* is the world wide port name used for routing (often in conjunction with multipathing) and is as a 16-digit hex value (for example **0x50050763050b073d**).
- *FCP_LUN* refers to the storage logical unit identifier and is specified as a 16-digit hexadecimal value padded with zeroes to the right (for example **0x4020400100000000**).

NOTE

A target world wide port name (WWPN) and an FCP_LUN have to be provided if the **zFCP** device is not configured in NPIV mode, when auto LUN scanning is disabled by the **zfcp.allow_lun_scan=0** kernel module parameter or when installing RHEL-9.0 or older releases. Otherwise only the **device_bus_ID** value is mandatory.

- These variables can be used on systems with FCP devices to activate FCP LUNs such as SCSI disks. Additional FCP LUNs can be activated during the installation interactively or by means of a Kickstart file. An example value looks similar to the following:

```
FCP_1="0.0.fc00 0x50050763050b073d 0x4020400100000000"
FCP_2="0.0.4000"
```

Each of the values used in the FCP parameters (for example **FCP_1** or **FCP_2**) are site-specific and are normally supplied by the FCP storage administrator.

9.8.4. Additional Boot Parameters for 64-bit IBM Z

You can define the following parameters in a parameter file but they do not work in a CMS configuration file.

rd.live.check

Turns on testing of an ISO-based installation source; for example, when using **inst.repo=** with an ISO on local disk or mounted with NFS.

inst.nompath

Disables support for multipath devices.

inst.proxy=[*protocol*://][*username*[:*password*]@]*host*[:*port*]

Specify a proxy to use with installation over HTTP, HTTPS or FTP.

inst.rescue

Boot into a rescue system running from a RAM disk that can be used to fix and restore an installed system.

inst.stage2=*URL*

Specifies a path to a tree containing **install.img**, not to the **install.img** directly. Otherwise, follows the same syntax as **inst.repo=**. If **inst.stage2** is specified, it typically takes precedence over other methods of finding **install.img**. However, if **Anaconda** finds **install.img** on local media, the **inst.stage2** URL will be ignored.
If **inst.stage2** is not specified and **install.img** cannot be found locally, **Anaconda** looks to the location given by **inst.repo=** or **method=**.

If only **inst.stage2=** is given without **inst.repo=** or **method=**, **Anaconda** uses whatever repos the installed system would have enabled by default for installation.

Use the option multiple times to specify multiple HTTP, HTTPS or FTP sources. The HTTP, HTTPS or FTP paths are then tried sequentially until one succeeds:

```
inst.stage2=http://hostname/path_to_install_tree/
inst.stage2=http://hostname/path_to_install_tree/
inst.stage2=http://hostname/path_to_install_tree/
```

inst.syslog=*IP/hostname*[:*port*]

Sends log messages to a remote syslog server.

The boot parameters described here are the most useful for installations and trouble shooting on 64-bit IBM Z, but only a subset of those that influence the installation program.

9.8.5. Sample parameter file and CMS configuration file on 64-bit IBM Z

To change the parameter file, begin by extending the shipped **generic.prm** file.

Example of **generic.prm** file:

```
ro ramdisk_size=40000 cio_ignore=all,!condev
CMSDASD="191" CMSCONFFILE="redhat.conf"
inst.rdp
inst.repo=http://example.com/path/to/dvd-contents
```

Example of **redhat.conf** file configuring a QETH network device (pointed to by **CMSCONFFILE** in **generic.prm**):

```
NETTYPE="qeth"
SUBCHANNELS="0.0.0600,0.0.0601,0.0.0602"
PORTNAME="FOOBAR"
PORTNO="0"
LAYER2="1"
MACADDR="02:00:be:3a:01:f3"
HOSTNAME="foobar.systemz.example.com"
IPADDR="192.168.17.115"
NETMASK="255.255.255.0"
GATEWAY="192.168.17.254"
DNS="192.168.17.1"
SEARCHDNS="systemz.example.com:example.com"
DASD="200-203"
```

CHAPTER 10. CREATING A KERNEL-BASED VIRTUAL MACHINE AND BOOTING THE INSTALLATION ISO IN THE VM

You can create a kernel-based virtual machine (KVM) and start the RHEL installation.

Prerequisites

- On the IBM Z platform, the KVM host runs RHEL installed in LPAR mode. See Installing in an LPAR.

Procedure

- Create virtual machine with the instance of Red Hat Enterprise Linux as a KVM guest operating system, use the following **virt-install** command on the KVM host:

```
$ virt-install --name=<guest_name> --disk size=<disksize_in_GB> --memory=
<memory_size_in_MB> --cdrom <filepath_to_iso> --graphics vnc
```

Additional resources

- Creating virtual machines by using the command line

CHAPTER 11. BOOTING THE INSTALLATION MEDIA

After you have created bootable media, you are ready to boot the Red Hat Enterprise Linux installation.

You can register RHEL by using the Red Hat Content Delivery Network (CDN). CDN is a geographically distributed series of web servers. These servers provide, for example, packages and updates to RHEL hosts with a valid subscription.

During the installation, registering and installing RHEL from the CDN offers following benefits:

- Utilizing the latest packages for an up-to-date system immediately after installation and
- Integrated support for connecting to Red Hat Lightspeed and enabling System Purpose.

11.1. BOOTING THE INSTALLATION FROM A NETWORK USING HTTP

When installing Red Hat Enterprise Linux on a large number of systems simultaneously, the best approach is to boot and install from a server on the local network. You can boot the Red Hat Enterprise Linux installation from a network by using HTTP.

IMPORTANT

To boot the installation process from a network, you must use a physical network connection, for example, Ethernet. You cannot boot the installation process with a wireless connection.

Prerequisites

- You have configured an HTTP boot server, and there is a network interface in your system. See **Additional resources** for more information.
- You have configured your system to boot from the network interface. This option is in the UEFI, and can be labeled **Network Boot** or **Boot Services**.
- You have verified that the UEFI is configured to boot from the specified network interface and supports the HTTP boot standard. For more information, see your hardware’s documentation.
- Your platform is x86_64 or you install in KVM.

Procedure

1. Verify that the network cable is attached. The link indicator light on the network socket should be lit, even if the computer is not switched on.
2. Turn on the system.
   Depending on your hardware, some network setup and diagnostic information can be displayed before your system connects to an HTTP boot server. When connected, a menu is displayed according to the HTTP boot server configuration.
3. Use the arrow keys on your keyboard to select the boot option that you require, and press **Enter** to select the boot option. The installation program gets started and you can install Red Hat Enterprise Linux using the graphical or text user interface.
   The installation program starts automatically if no action is performed in the boot menu window, by default within 60 seconds.

4. Optional: Edit the available boot options.
   Press **E** to enter edit mode. Change the predefined command line to add or remove boot options. Press **Ctrl**+**X** to confirm your choice.

Additional resources

- Automatically installing RHEL

11.2. BOOTING THE INSTALLATION FROM A NETWORK USING PXE

When installing Red Hat Enterprise Linux on a large number of systems simultaneously, the best approach is to boot and install from a server on the local network. You can boot the Red Hat Enterprise Linux installation from a network by using PXE.

> **IMPORTANT**
>
> To boot the installation process from a network, you must use a physical network connection, for example, Ethernet. You cannot boot the installation process with a wireless connection.

Prerequisites

- You have configured a TFTP server, and there is a network interface in your system that supports PXE. See **Additional resources** for more information.
- You have configured your system to boot from the network interface. This option is in the BIOS, and can be labeled **Network Boot** or **Boot Services**.
- You have verified that the BIOS is configured to boot from the specified network interface and supports the PXE standard. For more information, see your hardware's documentation.
- Your platform is x86_64 or you install in KVM.

Procedure

1. Verify that the network cable is attached. The link indicator light on the network socket should be lit, even if the computer is not switched on.
2. Switch on the system.
   Depending on your hardware, some network setup and diagnostic information can be displayed before your system connects to a PXE server. When connected, a menu is displayed according to the PXE server configuration.
3. Use the arrow keys on your keyboard to select the boot option that you require, and press **Enter** to select the boot option. The installation program gets started and you can install Red Hat Enterprise Linux by using the graphical or text user interface.
   The installation program starts automatically if no action is performed in the boot menu, by default within 60 seconds.
4. Optional: Edit the available boot options:

   UEFI-based systems

   Press **E** to enter edit mode. Change the predefined command line to add or remove boot options. Press **Ctrl**+**X** to confirm your choice.

BIOS-based systems

Press the **Tab** key on your keyboard to enter edit mode. Change the predefined command line to add or remove boot options. Press **Enter** to confirm your choice.

Additional resources

- Automatically installing RHEL
- Boot options reference

11.3. BOOTING THE INSTALLATION ON IBM Z TO INSTALL RHEL IN AN LPAR

You can boot the installation on IBM Z to install RHEL in an LPAR.

11.3.1. Booting the RHEL installation from an SFTP, FTPS, or FTP server to install in an IBM Z LPAR

You can install RHEL into an LPAR by using an SFTP, FTPS, or FTP server.

Procedure

1. Log in on the IBM Z Hardware Management Console (HMC) or the Support Element (SE) as a user with sufficient privileges to install a new operating system to an LPAR.
2. On the **Systems** tab, select the mainframe you want to work with, then on the **Partitions** tab select the LPAR to which you wish to install.
3. At the bottom of the screen, under **Daily**, find **Operating System Messages**. Double-click **Operating System Messages** to show the text console on which Linux boot messages will appear.
4. Double-click **Load from Removable Media or Server**.
5. In the dialog box that follows, select **SFTP/FTPS/FTP Server**, and enter the following information:
   - **Host Computer** - Host name or IP address of the FTP server you want to install from, for example **ftp.redhat.com**
   - **User ID** - Your user name on the FTP server. Or, specify anonymous.
   - **Password** - Your password. Use your email address if you are logging in as anonymous.
   - **File location (optional)** - Directory on the FTP server holding the Red Hat Enterprise Linux for IBM Z, for example **/rhel/s390x/**.
6. Click **Continue**.
7. In the dialog that follows, keep the default selection of **generic.ins** and click **Continue**.

11.3.2. Booting the RHEL installation from a prepared DASD to install in an IBM Z LPAR

You can install Red Hat Enterprise Linux into an LPAR by using an already prepared DASD.

Procedure

1. Log in on the IBM Z Hardware Management Console (HMC) or the Support Element (SE) as a user with sufficient privileges to install a new operating system to an LPAR.
2. On the **Systems** tab, select the mainframe you want to work with, then on the **Partitions** tab select the LPAR to which you wish to install.
3. At the bottom of the screen, under **Daily**, find **Operating System Messages**. Double-click **Operating System Messages** to show the text console on which Linux boot messages will appear.
4. Double-click **Load**.
5. In the dialog box that follows, select **Normal** as the **Load type**.
6. As Load **address**, fill in the device number of the DASD.
7. Click the **OK** button.

11.3.3. Booting the RHEL installation from an FCP-attached SCSI disk to install in an IBM Z LPAR

You can install Red Hat Enterprise Linux into an LPAR by using an already prepared FCP attached SCSI disk.

Procedure

1. Log in on the IBM Z Hardware Management Console (HMC) or the Support Element (SE) as a user with sufficient privileges to install a new operating system to an LPAR.
2. On the **Systems** tab, select the mainframe you want to work with, then on the **Partitions** tab select the LPAR to which you wish to install.
3. At the bottom of the screen, under **Daily**, find **Operating System Messages**. Double-click **Operating System Messages** to show the text console on which Linux boot messages will appear.
4. Double-click **Load**.
5. In the dialog box that follows, select **SCSI** as the **Load type**.
6. As **Load address**, fill in the device number of the FCP channel connected with the SCSI disk.
7. As **World wide port name**, fill in the WWPN of the storage system containing the disk as a 16-digit hexadecimal number.
8. As **Logical unit number**, fill in the LUN of the disk as a 16-digit hexadecimal number.
9. Leave the **Boot record logical block address** as **0** and the **Operating system specific load parameters** empty.
10. Click the **OK** button.

11.4. BOOTING THE INSTALLATION ON IBM Z TO INSTALL RHEL IN Z/VM

When installing under z/VM, you can boot from:

- The z/VM virtual reader
- A DASD or an FCP-attached SCSI disk prepared with the **zipl** boot loader

11.4.1. Booting the RHEL installation by using the z/VM Reader

Booting the Red Hat Enterprise Linux (RHEL) installation by using the z/VM Reader is a method designed for IBM Z (s390x) mainframe environments. This approach delivers installation boot images directly to a virtual machine through the z/VM Reader. It enables installations even when other boot methods (such as DASD or FCP) are unavailable or impractical.

Procedure

1. If necessary, add the device containing the z/VM TCP/IP tools to your CMS disk list. For example:

   ```
   cp link tcpmaint 592 592
   acc 592 fm
   ```

   Replace *fm* with any **FILEMODE** letter.

2. For a connection to an FTPS server, enter:

   ```
   ftp <host> (secure
   ```

   Where **host** is the host name or IP address of the FTP server that hosts the boot images (**kernel.img** and **initrd.img**).

3. Log in and execute the following commands. Use the **(repl** option if you are overwriting existing **kernel.img**, **initrd.img**, **generic.prm**, or **redhat.exec** files:

   ```
   cd /location/of/install-tree/images/
   ascii
   get generic.prm (repl
   get redhat.exec (repl
   locsite fix 80
   binary
   get kernel.img (repl
   get initrd.img (repl
   quit
   ```

4. Optional: Check whether the files were transferred correctly by using the CMS command **filelist** to show the received files and their format. It is important that **kernel.img** and **initrd.img** have a fixed record length format denoted by F in the Format column and a record length of 80 in the Lrecl column. For example:

   ```
   VMUSER FILELIST A0 V 169 Trunc=169 Size=6 Line=1 Col=1 Alt=0
   Cmd Filename Filetype Fm Format Lrecl Records Blocks Date Time
   REDHAT EXEC  B1 V 22 1  1 4/15/10 9:30:40
   ```

```
GENERIC PRM  B1 V 44 1 1 4/15/10 9:30:32
INITRD IMG  B1 F 80 118545 2316 4/15/10 9:30:25
KERNEL IMG  B1 F 80 74541 912 4/15/10 9:30:17
```

Press **PF3** to quit filelist and return to the CMS prompt.

5. Customize boot parameters in **generic.prm** as necessary. For details, see Customizing boot parameters.
   Another way to configure storage and network devices is by using a CMS configuration file. In such a case, add the ***CMSDASD=*** and ***CMSCONFFILE=*** parameters to **generic.prm**.

6. Finally, execute the REXX script redhat.exec to boot the installation program:

   ```
   redhat
   ```

11.4.2. Booting the RHEL installation by using a prepared DASD

Booting the Red Hat Enterprise Linux (RHEL) installation from a prepared Direct Access Storage Device (DASD) is a method commonly used on IBM Z (s390x) mainframe systems. Using a prepared DASD ensures that the installation process is consistent, repeatable, and can be integrated into larger provisioning workflows.

Procedure

- Boot from the prepared DASD and select the **zipl** boot menu entry referring to the Red Hat Enterprise Linux installation program. Use a command of the following form:

  ```
  cp ipl DASD_device_number loadparm boot_entry_number
  ```

  Replace *DASD_device_number* with the device number of the boot device, and *boot_entry_number* with the **zipl** configuration menu for this device. For example:

  ```
  cp ipl eb1c loadparm 0
  ```

11.4.3. Booting the RHEL installation by using a prepared FCP-attached SCSI disk

Booting Red Hat Enterprise Linux (RHEL) from a prepared FCP-attached SCSI disk is a common method on IBM Z (s390x) mainframe systems, especially when using Fibre Channel Protocol (FCP) storage. This approach is used when you need to install RHEL on systems that rely on external SAN storage rather than local DASD devices.

Procedure

1. Configure the SCSI boot loader of z/VM to access the prepared SCSI disk in the FCP Storage Area Network. Select the prepared **zipl** boot menu entry referring to the Red Hat Enterprise Linux installation program. Use a command of the following form:

   ```
   cp set loaddev portname WWPN lun LUN bootprog boot_entry_number
   ```

   Replace *WWPN* with the World Wide Port Name of the storage system and *LUN* with the Logical Unit Number of the disk. The 16-digit hexadecimal numbers must be split into two pairs of eight digits each. For example:

```
cp set loaddev portname 50050763 050b073d lun 40204011 00000000 bootprog 0
```

2. Optional: Confirm your settings with the command:

```
query loaddev
```

3. Boot the FCP device connected with the storage system containing the disk with the following command:

```
cp ipl FCP_device
```

For example:

```
cp ipl fc00
```

CHAPTER 12. OPTIONAL: CUSTOMIZING BOOT OPTIONS

When you are installing RHEL on **x86_64** or **ARM64** architectures, you can edit the boot options to customize the installation process based on your specific environment.

12.1. BOOT OPTIONS

You can append multiple options separated by space to the boot command line. Boot options specific to the installation program always start with **inst**. The following are the available boot options:

Options with an equals "=" sign

You must specify a value for boot options that use the **=** symbol. For example, the **inst.lang=** option must contain a value, in this example, a language code. The correct syntax for this example is **inst.lang=en_US**.

Options without an equals "=" sign

This boot option does not accept any values or parameters. For example, the **rd.live.check** option forces the installation program to verify the installation media before starting the installation. If this boot option is present, the installation program performs the verification and if the boot option is not present, the verification is skipped.

You can customize boot options for a particular menu entry by pressing the **e** key and adding custom boot options to the command line. When ready, press **Ctrl+X** to boot the modified option.

Additional resources

- Editing the GRUB2 menu
- Boot options reference

12.2. EDITING THE GRUB2 MENU

You can edit the GRUB2 boot menu during RHEL installation to customize boot parameters and kernel options. It enables configuration of specific settings such as FIPS mode, network parameters, and other system requirements before the installation begins.

Prerequisites

- You have created bootable installation media (USB or DVD) or have set up a server providing PXE or UEFI HTTP boot-related services.
- You have booted the installation from the media or from the network, and the installation boot menu is open.

Procedure

1. From the boot menu window, select the required option and press **e**.
2. Move the cursor to the end of the kernel command line and add the parameters as required. For example, to enable the cryptographic module self-checks mandated by the Federal Information Processing Standard (FIPS) 140, add **fips=1**:

```
linuxefi /images/pxeboot/vmlinuz inst.stage2=hd:LABEL=RHEL-10-0-BaseOS-x86_64
rd.live.\
check quiet fips=1
```

3. When you finish editing, press **Ctrl**+**X** to start the installation using the specified options.

CHAPTER 13. UPDATING DRIVERS DURING INSTALLATION

You can update drivers during the Red Hat Enterprise Linux installation process. Updating drivers is completely optional. Do not perform a driver update unless it is necessary. Ensure you have been notified by Red Hat, your hardware vendor, or a trusted third-party vendor that a driver update is required during Red Hat Enterprise Linux installation.

13.1. OVERVIEW

Red Hat Enterprise Linux supports drivers for many hardware devices but some newly-released drivers may not be supported. A driver update should only be performed if an unsupported driver prevents the installation from completing. Updating drivers during installation is typically only required to support a particular configuration. For example, installing drivers for a storage adapter card that provides access to your system's storage devices.

> **WARNING**
>
> Driver update disks may disable conflicting kernel drivers. In rare cases, unloading a kernel module may cause installation errors.

13.2. TYPES OF DRIVER UPDATE

Red Hat, your hardware vendor, or a trusted third party provides the driver update as an ISO image file. Once you receive the ISO image file, choose the type of driver update.

Types of driver update

- **Automatic**: In this driver update method; a storage device (including a CD, DVD, or USB flash drive) labeled **OEMDRV** is physically connected to the system. If the **OEMDRV** storage device is present when the installation starts, it is treated as a driver update disk, and the installation program automatically loads its drivers.

- **Assisted**: The installation program prompts you to locate a driver update. You can use any local storage device with a label other than **OEMDRV**. The **inst.dd** boot option is specified when starting the installation. If you use this option without any parameters, the installation program displays all of the storage devices connected to the system, and prompts you to select a device that contains a driver update.

- **Manual**: Manually specify a path to a driver update image or an RPM package. You can use any local storage device with a label other than **OEMDRV**, or a network location accessible from the installation system. The **inst.dd=location** boot option is specified when starting the installation, where *location* is the path to a driver update disk or ISO image. When you specify this option, the installation program attempts to load any driver updates found at the specified location. With manual driver updates, you can specify local storage devices or a network location (HTTP, HTTPS or FTP server). You can use both **inst.dd=location** and **inst.dd** simultaneously, where *location* is the path to a driver update disk or ISO image. In this scenario, the installation program attempts to load any available driver updates from the location and also prompts you to select a device that contains the driver update.

Limitations

On UEFI systems with the Secure Boot technology enabled, all drivers must be signed with a valid certificate. Red Hat drivers are signed by one of Red Hat's private keys and authenticated by its corresponding public key in the kernel. If you load additional, separate drivers, verify that they are signed.

13.3. PREPARING A DRIVER UPDATE CD OR DVD

You can prepare a driver update on a CD or DVD to perform a driver update during installation.

Prerequisites

- You have received the driver update ISO image from Red Hat, your hardware vendor, or a trusted third-party vendor.
- You have burned the driver update ISO image to a CD or DVD.

> **WARNING**
>
> If only a single ISO image file ending in **.iso** is available on the CD or DVD, the burn process has not been successful. See your system's burning software documentation for instructions on how to burn ISO images to a CD or DVD.

Procedure

1. Insert the driver update CD or DVD into your system's CD/DVD drive, and browse it by using the system's file manager tool.
2. Verify that a single file **rhdd3** is available. **rhdd3** is a signature file that contains the driver description and a directory named **rpms**, which contains the RPM packages with the actual drivers for various architectures.

13.4. PREPARING A DRIVER UPDATE USB DRIVE

You can prepare a driver update on a USB flash drive to perform a driver update during installation.

Prerequisites

- You have received the driver update ISO image from Red Hat, your hardware vendor, or a trusted third-party vendor.
- You have prepared a USB flash drive with a file system compatible with RHEL to place the driver update ISO on.

Procedure

1. Connect the USB drive to your computer and find out what device it was assigned by the system. To find the assigned device, you can inspect the output of **dmesg** or **lsblk -o +MOUNTPOINT** command.

2. Find out if the drive has been mounted and what the mount point is (based on the output of **lsblk** command). If it’s not mounted, mount it manually:

   a. Optional: Create a mount point directory:

      ```
      mkdir /path/to/mountpoint
      ```

   b. Mount the USB device’s partition in the mount point directory. The following example assumes that the device’s partition that will be used for storing the driver update ISO file is **/dev/sdb1** and the mount point is **/mnt/usbdrive**:

      ```
      sudo mount /dev/sdb1 /mnt/usbdrive
      ```

3. Copy the driver update ISO file on the USB drive.
   The following example assumes that the mount point is **/mnt/usbdrive** and the ISO file’s location is **/home/user/driverdisk.iso**:

   ```
   sudo cp /home/user/driverdisk.iso /mnt/usbdrive
   ```

4. Unmount the USB device by using the **umount** command.

   ```
   sudo umount /mnt/usbdrive
   ```

13.5. PERFORMING AN AUTOMATIC DRIVER UPDATE

You can perform an automatic driver update during installation.

Prerequisites

- You have placed the driver update image on a standard disk partition with an **OEMDRV** label or burnt the **OEMDRV** driver update image to a CD or DVD or written it to a USB drive. Advanced storage, such as RAID or LVM volumes, may not be accessible during the driver update process.
- You have connected a block device with an **OEMDRV** volume label containing the driver update disk to your system, or inserted the prepared CD or DVD into your system’s CD/DVD drive or connected the prepared USB drive before starting the installation process.

Procedure

- When you complete the prerequisite steps, the drivers load automatically when the installation program starts and installs during the system’s installation process.

13.6. PERFORMING AN ASSISTED DRIVER UPDATE

You can perform an assisted driver update during installation to install necessary drivers for new or existing hardware.

Prerequisites

- You have connected a block device without an **OEMDRV** volume label to your system and copied the driver disk image to this device, or you have prepared a driver update CD or DVD and inserted it into your system’s CD or DVD drive or you have prepared a USB device with the driver disk and connected it to your computer before starting the installation process.

NOTE

If you burn an ISO image file to a CD or DVD or write an ISO image file to a USB drive but it does not have the **OEMDRV** volume label, you can use the **inst.dd** option with no arguments. The installation program provides an option to scan and select drivers from the CD, DVD or USB drive. In this scenario, the installation program does not prompt you to select a driver update ISO image. Another scenario is to use the CD, DVD or USB drive with the **inst.dd=*location*** boot option; this allows the installation program to automatically scan the CD, DVD or USB drive for driver updates. For more information, see Performing a driver update

Procedure

1. From the boot menu window, press the **E** key on your keyboard to display the boot command line.

2. Append the **inst.dd** boot option to the command line beginning with **linux** or **linuxefi** and press **Ctrl+X** to execute the boot process.

3. From the menu, select a local disk partition or a CD, DVD, or USB device. The installation program scans for ISO files, or driver update RPM packages.

4. Optional: Select the driver update ISO file.
   This step is not required if the selected device or partition contains driver update RPM packages rather than an ISO image file, for example, an optical drive containing a driver update CD, DVD or a USB drive that the ISO file has been written on.

5. Select the required drivers.

   a. Use the number keys on your keyboard followed by **Enter** key to toggle the driver selection.

   b. Press **c** followed by **Enter** to install the selected driver. The selected driver is loaded and after pressing **c** and **Enter** again to exit the driver disk device selection, the installation process starts.

13.7. PERFORMING A MANUAL DRIVER UPDATE

You can perform a manual driver update during installation to install necessary drivers for new or existing hardware.

Prerequisites

- You have placed the driver update ISO image file on a USB flash drive or a web server and connected it to your computer.

Procedure

1. From the boot menu window, press the **E** key on your keyboard to display the boot command line.

2. Append the **inst.dd=*location*** boot option to the command line, where location is a path to the driver update. Typically, the image file is located on a web server, for example, http://server.example.com/dd.iso, or on a USB flash drive, for example, **/dev/sdb1**. It is also

possible to specify an RPM package containing the driver update, for example http://server.example.com/dd.rpm.

3. Press **Ctrl+X** to execute the boot process. The drivers available at the specified location are automatically loaded and the installation process starts.

Additional resources

- The **inst.dd** boot option

13.8. DISABLING A DRIVER

You can disable a malfunctioning driver during the installation process.

Prerequisites

- You have booted into the installation media's bootloader (GRUB) menu.

Procedure

1. From the boot menu, press the **E** key on your keyboard to display the boot command line.
2. Append the **modprobe.blacklist=*driver_name*** boot option to the command line.
   Replace *driver_name* with the name of the driver or drivers you want to disable, for example:

   ```
   modprobe.blacklist=ahci
   ```

   Drivers disabled by using the **modprobe.blacklist=** boot option remain disabled on the installed system and appear in the **/etc/modprobe.d/anaconda-blacklist.conf** file.
3. Press **Enter** to execute the boot process.

CHAPTER 14. PREPARING A REMOTE INSTALLATION BY USING RDP

You must prepare to perform a remote installation by using the Remote Desktop Protocol.

14.1. REMOTE DESKTOP PROTOCOL OVERVIEW

Use the graphical user interface method of installing RHEL when you boot the system from a CD, DVD, or USB flash drive, or from a network using PXE. However, many enterprise systems, for example, IBM Power Systems and 64-bit IBM Z, are located in remote data center environments that are run autonomously and are not connected to a display, keyboard, and mouse. These systems are often referred to as headless systems and they are typically controlled over a network connection. The RHEL installation program includes a Remote Desktop Protocol (RDP) installation that runs the graphical installation on the target machine, but control of the graphical installation is handled by another system on the network.

The RHEL installation program is configured to start on the target system and wait for an RDP viewer that is installed on another system before proceeding. The IP address and port are displayed on the target system. You can use the RDP viewer to connect to the target system remotely by using the IP address and port, and complete the graphical installation.

14.2. CONSIDERATIONS FOR USING REMOTE DESKTOP PROTOCOL

By default, the installation program has an RDP server included. Consider the following items when performing a remote RHEL installation by using RDP:

RDP client application

An RDP client application is required to perform an RDP installation. RDP client applications are available in the repositories of most Linux distributions, and free RDP client applications are also available for other operating systems such as Windows or might even be built-in. The **GNOME Connections** RDP client application is available in RHEL. It is a part of the GNOME desktop environment and is installed as part of the **gnome-connections** package.

Network and firewall

If the target system is not allowed inbound connections by a firewall, then you must enable incoming RDP connections or disable the firewall. Disabling a firewall can have security implications. See the Security hardening document for more information about configuring the firewall.

Custom Boot Options

You must specify custom boot options to start an RDP installation and the installation instructions might differ depending on your system architecture.

CHAPTER 15. STARTING A REMOTE INSTALLATION BY USING THE REMOTE DESKTOP PROTOCOL

You can start remote installation by using the Remote Desktop Protocol.

15.1. PERFORMING A REMOTE RHEL INSTALLATION WITH AN RDP CLIENT

You can perform a remote RHEL installation by using an RDP client directly connecting to the target system that is being installed with RHEL. The system with the RDP viewer is referred to as the remote system. You are prompted by the RHEL installation program to initiate the connection from the RDP viewer on the remote system to the target system.

> **NOTE**
>
> You can use an RDP client of your choice (such as GNOME Connections). Specific instructions for other viewers might differ, but the general principles apply.

Prerequisites

- You have installed an RDP viewer on a remote system.

Procedure

1. From the RHEL boot menu on the target system, press the **e** key on your keyboard to edit the boot options.
2. Append the **inst.rdp** option to the end of the command line.
   You can also use the **inst.rdp.username** and **inst.rdp.password** boot options to set the RDP credentials. For example,

   ```
   inst.rdp inst.rdp.username=my_username inst.rdp.password=my_password
   ```

3. Press **CTRL+X** or **F10** to start the installation. The target system initializes the installation program and starts the necessary services. When the system is ready, a message is displayed providing the IP address of the system.
4. Provide username and password details, if not provided earlier by using the boot options.
5. Open the RDP viewer on the remote system.
6. Enter the IP address into the RDP server field.
7. Click **Connect**.
8. Enter the RDP username and password and click **OK**.
   A new window opens with the RDP connection established, displaying the RHEL installation menu. From this window, you can install RHEL on the target system by using the graphical user interface.

15.2. PERFORMING A REMOTE RHEL INSTALLATION BY USING RDP ON IBM Z

After the Initial Program Load (IPL) of the Anaconda installation program is complete, connect to the 64-bit IBM Z system from a local machine, as an install user by using an ssh connection. You need to connect to the installation system to continue the installation process. Use the RDP mode to run a GUI-based installation or use the established connection to run a text mode installation.

Prerequisites

- You booted the installation media
- The initial program boot is complete on the 64-bit IBM Z system, and the command prompt displays:

  ```
  Starting installer, one moment…
  Please ssh install@my-z-system (system IP address) to begin the install.
  ```

Procedure

1. To set up a remote connection with the 64-bit IBM Z system from a local machine, open the terminal and on the command prompt, run the following command:

   ```
   $ ssh install@my-z-system-domain-name
   ```

   or

   ```
   $ ssh install@my-z-system-IP-address
   ```

2. The ssh session displays the following output:

   ```
   Starting installer, one moment...
   libreport is not available in this environment - bug reporting disabled
   anaconda 40.22.3.26-1.el10_0 for Red Hat Enterprise Linux 10.0 started.
    * installation log files are stored in /tmp during the installation
    * shell is available in second TMUX pane (ctrl+b, then press 2)
    * when reporting a bug add logs from /tmp as separate text/plain attachments

   Wayland startup failed, falling back to text mode.
   ==============================================================================
   =====
   ==============================================================================
   =====
   Wayland was unable to start on your machine. Would you like to start an RDP
   session to connect to this computer from another computer and perform a
   graphical installation or continue with a text mode installation?

   1) Use graphical mode via Remote Desktop Protocol
   2) Use text mode

   Please make a selection from the above ['c' to continue, 'q' to quit, 'r' to
   refresh]:
   ```

   If you have configured the **inst.rdp parameter**, proceed to step 4.

3. Enter **1** to start RDP.

4. Enter RDP username and password, if you have not set these values previously via **inst.rdp.username** and **inst.rdp.password** boot options.

5. Use an RDP client to connect to the provided network address.
   You will be asked to provide the username and password you have set in the previous steps.

NOTE

If you have started the installation with the **inst.rdp** boot option, the installer will not ask you to connect via ssh. Instead, provide the RDP username and password in the console, as described in the step 4 above.

CHAPTER 16. CONSOLES AND LOGGING DURING INSTALLATION

The RHEL installer uses the **tmux** terminal multiplexer to display and control several windows in addition to the main interface. Each of these windows serve a different purpose; they display several different logs, which can be used to troubleshoot issues during the installation process. One of the windows provides an interactive shell with **root** privileges, unless this was specifically disabled by using a boot option or a Kickstart command.

The terminal multiplexer is running in virtual console 1. To switch from the actual installation environment to **tmux**, press **Ctrl**+**Alt**+**F1**. To go back to the main installation interface which runs in virtual console 6, press **Ctrl**+**Alt**+**F6**. During the text mode, installation starts in virtual console 1 (**tmux**), and switching to console 6 will open a shell prompt instead of a graphical interface.

The console running **tmux** has five available windows; their contents are described in the following table, along with keyboard shortcuts. Note that the keyboard shortcuts are two-part: first press **Ctrl**+**b**, then release both keys, and press the number key for the window you want to use.

You can also use **Ctrl**+**b n**, **Alt+ Tab**, and **Ctrl**+**b p** to switch to the next or previous **tmux** window, respectively.

Table 16.1. Available tmux windows

Shortcut	Contents
Ctrl+**b 1**	Main installation program window. Contains text-based prompts (used for text mode and also for interactive entry of RDP credentials), and also some debugging information.
Ctrl+**b 2**	Interactive shell with **root** privileges.
Ctrl+**b 3**	Installation log; displays messages stored in **/tmp/anaconda.log**.
Ctrl+**b 4**	Storage log; displays messages related to storage devices and configuration, stored in **/tmp/storage.log**.
Ctrl+**b 5**	Program log; displays messages from utilities executed during the installation process, stored in **/tmp/program.log**.
Ctrl+**b 6**	Packaging log; displays messages related to packages, stored in **/tmp/packaging.log**.

CHAPTER 17. CUSTOMIZING THE SYSTEM IN THE INSTALLER

During the customization phase of the installation, you must perform certain configuration tasks to enable the installation of Red Hat Enterprise Linux. These tasks include:

- Configuring the storage and assigning mount points.
- Selecting a base environment with software to be installed.
- Setting a password for the root user or creating a local user.

Optionally, you can further customize the system, for example, by configuring system settings and connecting the host to a network.

17.1. SETTING THE INSTALLER LANGUAGE

You can select the language and regional settings for the RHEL installation program interface to ensure proper localization during the installation process. It affects the installer interface language and regional formatting conventions.

Prerequisites

- You have created installation media.
- You have specified an installation source if you are using the Boot ISO image file.
- You have booted from the media to the bootloader menu.

Procedure

1. After you select the **Red hat Enterprise Linux**option from the boot menu, the installation program starts and **Welcome to Red Hat Enterprise Screen**appears.
2. From the left-hand pane of the **Welcome to Red Hat Enterprise Linux**window, select a language. Alternatively, search the preferred language by using the text box.

   NOTE

   A language is pre-selected by default. The pre-selected language is determined by the automatic location detection feature of the **GeoIP** module. If you use the **inst.lang=** option on the boot command line, then the language that you define with the boot option is selected.

3. From the right-hand pane of the **Welcome to Red Hat Enterprise Linux**window, select a location specific to your region.
4. Click **Continue** to proceed to the graphical installation window.
5. If you are installing a pre-release version of Red Hat Enterprise Linux, a warning message is displayed about the pre-release status of the installation media.
   a. To continue with the installation, click **I want to proceed**, or
   b. To quit the installation and reboot the system, click **I want to exit**.

17.2. CONFIGURING THE STORAGE DEVICES

You can install RHEL on a large variety of storage devices. You can configure basic, locally accessible, storage devices in the **Installation Destination** window. Basic storage devices directly connected to the local system, such as disks and solid-state drives, are displayed in the **Local Standard Disks** section of the window. On 64-bit IBM Z systems, this section contains activated Direct Access Storage Devices (DASDs).

17.2.1. Configuring installation destination

You can select and configure storage devices for RHEL installation the **Installation Destination** window, including disk selection, partitioning options, and encryption settings. It determines where the operating system is installed and how storage is managed on the target system.

Prerequisites

- The **Installation Summary** window is open.
- Ensure to back up your data if you plan to use a disk that already contains data. Manipulating partitions always carries a risk. For example, if the process is interrupted or fails for any reason data on the disk can be lost.

Procedure

1. From the **Installation Summary** window, click **Installation Destination**. Perform the following operations in the **Installation Destination** window that opens:
    a. From the **Local Standard Disks** section, select the storage device that you require; a white check mark indicates your selection. Disks without a white check mark are not used during the installation process; they are ignored if you choose automatic partitioning, and they are not available in manual partitioning.
    The **Local Standard Disks** shows all locally available storage devices. For example, SATA, NVMe™ and SCSI disks, USB flash and external disks. Storage devices connected post starting the installation program are not detected, unless you follow the step 2 below. If you use a removable drive to install RHEL, your system is unusable if you remove the device.

    b. Optional: Click the **Refresh** link in the lower right-hand side of the window if you want to configure additional local storage devices connected after the installation program has started. The **Rescan Disks** dialog box opens.

        i. Click **Rescan Disks** and wait until the scanning process completes.
        All storage changes that you make during the installation are lost when you click **Rescan Disks**.

        ii. Click **OK** to return to the **Installation Destination** window. All detected disks including any new ones are displayed under the **Local Standard Disks** section.

2. Optional: Click **Add a disk…** to add a specialized storage device.
    The **Storage Device Selection** window opens and lists all storage devices that the installation program has access to.

3. Optional: Under **Storage Configuration**, select the **Automatic** radio button for automatic partitioning.
    You can also configure custom partitioning. For more details, see Configuring manual partitioning.

4. Optional: Select **Free up space by removing or shrinking existing partitions**to reclaim space from an existing partitioning layout. For example, if a disk you want to use already has a different operating system and you want to make this system’s partitions smaller to allow more room for Red Hat Enterprise Linux.

5. Optional: Select **Encrypt my data** to encrypt all partitions except the ones needed to boot the system (such as /**boot**) by using *Linux Unified Key Setup* (LUKS). Encrypting your disk adds an extra layer of security.

   a. Click **Done**. The **Disk Encryption Passphrase**dialog box opens.

      i. Type your passphrase in the **Passphrase** and **Confirm** fields.

      ii. Click **Save Passphrase** to complete disk encryption.

> **WARNING**
>
> If you lose the LUKS passphrase, any encrypted partitions and their data is completely inaccessible. There is no way to recover a lost passphrase. However, if you perform a Kickstart installation, you can save encryption passphrases and create backup encryption passphrases during the installation.

6. Optional: Click the **Full disk summary and bootloader**link in the lower left-hand side of the window to select which storage device contains the boot loader. For more information, see Configuring boot loader.
   In most cases it is sufficient to leave the boot loader in the default location. Some configurations, for example, systems that require chain loading from another boot loader require the boot drive to be specified manually.

7. Click **Done**.

8. Optional: The **Reclaim Disk Space** dialog box appears if you selected **automatic partitioning** and the **Free up space by removing or shrinking existing partitions**option, or if there is not enough free space on the selected disks to install Red Hat Enterprise Linux. It lists all configured disk devices and all partitions on those devices. The dialog box displays information about the minimal disk space the system needs for an installation with the currently selected package set and how much space you have reclaimed. To start the reclaiming process:

   a. Review the displayed list of available storage devices. The **Reclaimable Space** column shows how much space can be reclaimed from each entry.

   b. Select a disk or partition to reclaim space.

   c. Use the **Shrink** button to use free space on a partition while preserving the existing data.

   d. Use the **Delete** button to delete that partition or all partitions on a selected disk including existing data.

   e. Use the **Delete all** button to delete all existing partitions on all disks including existing data and make this space available to install Red Hat Enterprise Linux.

f. Click **Reclaim space** to apply the changes and return to graphical installations.
   No disk changes are made until you click **Begin Installation** on the **Installation Summary** window. The **Reclaim Space** dialog only marks partitions for resizing or deletion; no action is performed.

Additional resources

- How to use dm-crypt on IBM Z, LinuxONE and with the PAES cipher

17.2.2. Advanced considerations for installation destination configuration

Following are some special cases to consider when you are configuring installation destinations:

- Some BIOS types do not support booting from a RAID card. In these instances, the **/boot** partition must be created on a partition outside of the RAID array, such as on a separate disk. It is necessary to use an internal disk for partition creation with problematic RAID cards. A **/boot** partition is also necessary for software RAID setups. If you choose to partition your system automatically, you should manually edit your **/boot** partition.

- To configure the Red Hat Enterprise Linux boot loader to *chain load* from a different boot loader, you must specify the boot drive manually by clicking the **Full disk summary and bootloader** link from the **Installation Destination** window.

- When you install Red Hat Enterprise Linux on a system with both multipath and non-multipath storage devices, the automatic partitioning layout in the installation program creates volume groups that contain a mix of multipath and non-multipath devices. This defeats the purpose of multipath storage. Select either multipath or non-multipath devices on the **Installation Destination** window. Alternatively, proceed to manual partitioning.

17.2.3. Configuring boot loader

RHEL uses GRand Unified Bootloader version 2 (**GRUB2**) as the boot loader for AMD64 and Intel 64, IBM Power Systems, and ARM. For 64-bit IBM Z, the **zipl** boot loader is used.

The boot loader is the first program that runs when the system starts and is responsible for loading and handing over control to an operating system. **GRUB2** can boot any compatible operating system (including Microsoft Windows) and can also use chain loading to hand over control to other boot loaders for unsupported operating systems.

> **WARNING**
>
> Installing **GRUB2** may overwrite your existing boot loader.

If an operating system is already installed, the Red Hat Enterprise Linux installation program attempts to automatically detect and configure the boot loader to start the other operating system. If the boot loader is not detected, you can manually configure any additional operating systems after you finish the installation.

If you are installing a Red Hat Enterprise Linux system with more than one disk, you might want to manually specify the disk where you want to install the boot loader.

Procedure

1. From the **Installation Destination** window, click the **Full disk summary and bootloader** link. The **Selected Disks** dialog box opens.
   The boot loader is installed on the device of your choice, or, on a UEFI system; the **EFI system partition** is created on the target device and used to store the boot loader files.

2. To change the boot device, select a device from the list and click **Set as Boot Device**. You can set only one device as the boot device.

3. To disable a new boot loader installation, select the device currently marked for boot and click **Do not install boot loader**. This ensures **GRUB2** is not installed on any device.

> **WARNING**
>
> If you choose not to install a boot loader, you cannot boot the system directly and you must use another boot method, such as a standalone commercial boot loader application. Use this option only if you have another way to boot your system.

The boot loader may also require a special partition to be created, depending on if your system uses BIOS or UEFI firmware, or if the boot drive has a *GUID Partition Table* (GPT) or a **Master Boot Record** (MBR, also known as **msdos**) label. If you use automatic partitioning, the installation program creates the partition.

17.2.4. Storage device selection

The storage device selection window lists all storage devices that the installation program can access. Depending on your system and available hardware, some tabs might not be displayed. The devices are grouped under the following tabs:

Multipath Devices

Storage devices accessible through more than one path, such as through multiple SCSI controllers or Fiber Channel ports on the same system. The installation program only detects multipath storage devices with serial numbers that are 16 or 32 characters long.

Other SAN Devices

Devices available on a Storage Area Network (SAN).

Firmware RAID

Storage devices attached to a firmware RAID controller.

IBM Z Devices

Storage devices, or Logical Units (LUNs), DASD, attached through the zSeries Linux FCP (Fiber Channel Protocol) driver.

17.2.5. Filtering storage devices

You can filter and select specific storage devices during RHEL installation by using WWID, port, target, or LUN identifiers. It helps identify and configure specialized storage devices for installation, ensuring proper device selection in complex storage environments.

Prerequisites

- The **Installation Summary** window is open.

Procedure

1. From the **Installation Summary** window, click **Installation Destination**. The **Installation Destination** window opens, listing all available drives.
2. Under the **Specialized & Network Disks**section, click **Add a disk**. The storage devices selection window opens.
3. Click the **Search by** tab to search by port, target, LUN, or WWID.
   Searching by WWID or LUN requires additional values in the corresponding input text fields.
4. Select the option that you require from the **Search** drop-down menu.
5. Click **Find** to start the search. Each device is presented on a separate row with a corresponding check box.
6. Select the check box to enable the device that you require during the installation process.
   Later in the installation process you can choose to install Red Hat Enterprise Linux on any of the selected devices, and you can choose to mount any of the other selected devices as part of the installed system automatically. Selected devices are not automatically erased by the installation process and selecting a device does not put the data stored on the device at risk.

   > NOTE
   >
   > You can add devices to the system after installation by modifying the **/etc/fstab** file.

7. Click **Done** to return to the **Installation Destination** window.
   Any storage devices that you do not select are hidden from the installation program entirely. To chain load the boot loader from a different boot loader, select all the devices present.

17.3. ADVANCED STORAGE OPTIONS

To use an advanced storage device, you can configure an iSCSI (SCSI over TCP/IP) target or FCoE (Fibre Channel over Ethernet) SAN (Storage Area Network).

To use iSCSI storage devices for the installation, the installation program must be able to discover them as iSCSI targets and be able to create an iSCSI session to access them. Each of these steps might require a user name and password for Challenge Handshake Authentication Protocol (CHAP) authentication. Additionally, you can configure an iSCSI target to authenticate the iSCSI initiator on the system to which the target is attached (reverse CHAP), both for discovery and for the session. Used together, CHAP and reverse CHAP are called mutual CHAP or two-way CHAP. Mutual CHAP provides the greatest level of security for iSCSI connections, particularly if the user name and password are different for CHAP authentication and reverse CHAP authentication.

Repeat the iSCSI discovery and iSCSI login steps to add all required iSCSI storage. You cannot change the name of the iSCSI initiator after you attempt discovery for the first time. To change the iSCSI initiator name, you must restart the installation.

17.3.1. Discovering and starting an iSCSI session

You can discover and connect to iSCSI storage targets during RHEL installation to enable network-based storage for system installation. It allows you to use remote storage devices as installation targets, providing flexibility in storage configuration and enabling centralized storage management.

iSCSI Boot Firmware Table (iBFT)

When the installer starts, it checks if the BIOS or add-on boot ROMs of the system support iBFT. It is a BIOS extension for systems that can boot from iSCSI. If the BIOS supports iBFT, the installer reads the iSCSI target information for the configured boot disk from the BIOS and logs in to this target, making it available as an installation target. To automatically connect to an iSCSI target, activate a network device for accessing the target. To do so, use the **ip=ibft** boot option. For more information, see Network boot options.

Discover and add iSCSI targets manually

You can discover and start an iSCSI session to identify available iSCSI targets (network storage devices) in the installer's graphical user interface.

Prerequisites

- The **Installation Summary** window is open.

Procedure

1. From the **Installation Summary** window, click **Installation Destination**. The **Installation Destination** window opens, listing all available drives.

2. Under the **Specialized & Network Disks**section, click **Add a disk**. The storage devices selection window opens.

3. Click **Add iSCSI target**. The **Add iSCSI Storage Target**window opens.

   IMPORTANT

   You cannot place the /**boot** partition on iSCSI targets that you have manually added by using this method. An iSCSI target containing a /**boot** partition must be configured for use with iBFT. However, in instances where the installed system is expected to boot from iSCSI with iBFT configuration provided by a method other than firmware iBFT. For example, by using iPXE, you can remove the /**boot** partition restriction by using the **inst.nonibftiscsiboot** installer boot option.

4. Enter the IP address of the iSCSI target in the **Target IP Address** field.

5. Type a name in the **iSCSI Initiator Name** field for the iSCSI initiator in iSCSI qualified name (IQN) format. A valid IQN entry contains the following information:

   - The string **iqn.** (note the period).

   - A date code that specifies the year and month in which your organization's Internet domain or subdomain name was registered, represented as four digits for the year, a dash, and two digits for the month, followed by a period. For example, represent September 2010 as **2010-09.**

   - Your organization's Internet domain or subdomain name, presented in reverse order with the top-level domain first. For example, represent the subdomain **storage.example.com** as **com.example.storage**.

- A colon followed by a string that uniquely identifies this particular iSCSI initiator within your domain or subdomain. For example **:diskarrays-sn-a8675309**.
  A complete IQN is as follows: **iqn.2010-09.storage.example.com:diskarrays-sn-a8675309**. The installation program pre-populates the **iSCSI Initiator Name** field with a name in this format to help you with the structure. For more information about IQNs, see *3.2.6. iSCSI Names* in *RFC 3720 - Internet Small Computer Systems Interface (iSCSI)* available from tools.ietf.org and *1. iSCSI Names and Addresses* in *RFC 3721 - Internet Small Computer Systems Interface (iSCSI) Naming and Discovery* available from tools.ietf.org.

6. Select the **Discovery Authentication Type** drop-down menu to specify the type of authentication to use for iSCSI discovery. The following options are available:
   - No credentials
   - CHAP pair
   - CHAP pair and a reverse pair
7. Do one of the following:
   a. If you selected the **CHAP pair** as the authentication type, enter the user name and password for the iSCSI target in the **CHAP Username** and **CHAP Password** fields.
   b. If you selected the **CHAP pair and a reverse pair** as the authentication type, enter the user name and password for the iSCSI target in the **CHAP Username** and **CHAP Password** field, and the user name and password for the iSCSI initiator in the **Reverse CHAP Username** and **Reverse CHAP Password** fields.
8. Optional: Select the **Bind targets to network interfaces** check box.
9. Click **Start Discovery**.
   The installation program attempts to discover an iSCSI target based on the information provided. If discovery succeeds, the **Add iSCSI Storage Target** window displays a list of all iSCSI nodes discovered on the target.
10. Select the check boxes for the node that you want to use for installation.
    The **Node login authentication type** menu contains the same options as the **Discovery Authentication Type** menu. However, if you need credentials for discovery authentication, use the same credentials to log in to a discovered node.
11. Click the additional **Use the credentials from discovery** drop-down menu. When you provide the proper credentials, the **Log In** button becomes available.
12. Click **Log In** to initiate an iSCSI session.
    While the installer uses **iscsiadm** to find and log into iSCSI targets, **iscsiadm** automatically stores any information about these targets in the **iscsiadm** iSCSI database. The installer then copies this database to the installed system and marks any iSCSI targets that are not used for root partition, so that the system automatically logs in to them when it starts. If the root partition is placed on an iSCSI target, **initrd** logs into this target and the installer does not include this target in start up scripts to avoid multiple attempts to log into the same target.

17.3.2. Configuring FCoE parameters

You can discover the FCoE (Fibre Channel over Ethernet) devices from the **Installation Destination** window by configuring the FCoE parameters accordingly.

Prerequisites

- The **Installation Summary** window is open.

Procedure

1. From the **Installation Summary** window, click **Installation Destination**. The **Installation Destination** window opens, listing all available drives.
2. Under the **Specialized & Network Disks**section, click **Add a disk**. The storage devices selection window opens.
3. Click **Add FCoE SAN**. A dialog box opens for you to configure network interfaces for discovering FCoE storage devices.
4. Select a network interface that is connected to an FCoE switch in the **NIC** drop-down menu.
5. Click **Add FCoE disk(s)** to scan the network for SAN devices.
6. Select the required check boxes:
   - **Use DCB:***Data Center Bridging* (DCB) is a set of enhancements to the Ethernet protocols designed to increase the efficiency of Ethernet connections in storage networks and clusters. Select the check box to enable or disable the installation program's awareness of DCB. Enable this option only for network interfaces that require a host-based DCBX client. For configurations on interfaces that use a hardware DCBX client, disable the check box.
   - **Use auto vlan:***Auto VLAN* is enabled by default and indicates whether VLAN discovery should be performed. If this check box is enabled, then the FIP (FCoE Initiation Protocol) VLAN discovery protocol runs on the Ethernet interface when the link configuration has been validated. If they are not already configured, network interfaces for any discovered FCoE VLANs are automatically created and FCoE instances are created on the VLAN interfaces.
7. Discovered FCoE devices are displayed under the **Other SAN Devices** tab in the **Installation Destination** window.

17.3.3. Configuring DASD storage devices

You can discover and configure the DASD storage devices from the **Installation Destination** window.

Prerequisites

- The **Installation Summary** window is open.

Procedure

1. From the **Installation Summary** window, click **Installation Destination**. The **Installation Destination** window opens, listing all available drives.
2. Under the **Specialized & Network Disks**section, click **Add a disk**. The storage devices selection window opens.
3. Click **Add DASD ECKD**. The **Add DASD Storage Target**dialog box opens and prompts you to specify a device number, such as **0.0.0204**, and attach additional DASDs that were not detected when the installation started.

4. Type the device number of the DASD that you want to attach in the **Device number** field.

5. Click **Start Discovery**.
   If a DASD with the specified device number is found and if it is not already attached, the dialog box closes and the newly-discovered drives appear in the list of drives. You can then select the check boxes for the required devices and click **Done**. The new DASDs are available for selection, marked as **DASD device 0.0.*xxxx*** in the **Local Standard Disks** section of the **Installation Destination** window.

   If you entered an invalid device number, or if the DASD with the specified device number is already attached to the system, an error message appears in the dialog box, explaining the error and prompting you to try again with a different device number.

17.3.4. Configuring FCP devices

FCP devices enable 64-bit IBM Z to use SCSI devices rather than, or in addition to, Direct Access Storage Device (DASD) devices. FCP devices provide a switched fabric topology that enables 64-bit IBM Z systems to use SCSI LUNs as disk devices in addition to traditional DASD devices.

Prerequisites

- The **Installation Summary** window is open.
- For an FCP-only installation, you have removed the **DASD=** option from the CMS configuration file or the **rd.dasd=** option from the parameter file to indicate that no DASD is present.

Procedure

1. From the **Installation Summary** window, click **Installation Destination**. The **Installation Destination** window opens, listing all available drives.

2. Under the **Specialized & Network Disks**section, click **Add a disk**. The storage devices selection window opens.

3. Click **Add ZFCP LUN**. The **Add zFCP Storage Target** dialog box opens allowing you to add a FCP (Fibre Channel Protocol) storage device.
   64-bit IBM Z requires that you enter any FCP device manually so that the installation program can activate FCP LUNs. You can enter FCP devices either in the graphical installation, or as a unique parameter entry in the parameter or CMS configuration file. The values that you enter must be unique to each site that you configure.

4. Type the 4 digit hexadecimal device number in the **Device number** field.

5. Provide following details when the **zFCP** device is not configured in NPIV mode or **auto LUN** scanning is disabled by the **zfcp.allow_lun_scan=0** kernel module parameter:

   a. Type the 16 digit hexadecimal World Wide Port Number (WWPN) in the **WWPN** field.

   b. Type the 16 digit hexadecimal FCP LUN identifier in the **LUN** field.

6. Click **Start Discovery** to connect to the FCP device.
   The newly-added devices are displayed in the **IBM Z** tab of the **Installation Destination** window.

   Use only lower-case letters in hex values. If you enter an incorrect value and click **Start Discovery**, the installation program displays a warning. You can edit the configuration information and retry the discovery attempt. For more information about these values, consult the hardware documentation and check with your system administrator.

17.3.5. Configuring an NVMe fabrics devices using the graphical installation mode

Configure the Non-volatile Memory Express™ (NVMe™) over fabrics by using the graphical installation to use it as an installation target. Additionally, if the device meets the requirements for booting, you can also set the device as a boot device.

Prerequisites

- A NVMe™ over fabrics device is present on the system.
- The initial installation process has been completed and the **Installation Summary** window is open.

Procedure

1. From the **Installation Summary** window, click **Installation Destination**.
   The **Installation Destination** window opens, listing all available devices. This includes local (PCI Express transport) NVMe™ devices.
2. Under the **Specialized & Network Disks**section, click **Add a disk…**.
   The storage devices selection window opens.
3. Click the **NVMe™ Fabrics Devices** tab.
4. Optional: If the device list is too long, use the **Filter by** option to view specific devices.
5. Select the devices from the list by using check boxes.
6. Click **Done** to return to the **Installation Destination** window.
   The NVMe™ device that you reconfigured is displayed in the **Specialized & Network Disks** section.
7. Click **Done** to return to the Installation Summary window.

Additional resources

- Configuring NVMe™ over fabrics using NVMe™

17.4. CONFIGURING THE ROOT ACCOUNT AND CREATING USERS

You can configure the root account and create users to access the system from the **Installation Summary** screen.

17.4.1. Configuring a root account

You can configure a **root** account during the installation process to log in to the administrator (also known as superuser or root) account for system administration tasks. These tasks include:

- Installing and updating software packages
- Changing system-wide configuration such as network and firewall settings, storage options

- Adding or modifying users, groups and file permissions.

To gain root privileges for the installed systems, you can either use a **root** account or create a user account with administrative privileges (member of the **wheel** group). The **root** account is always created during the installation. Switch to the administrator account only when you need administrator access for tasks.

> WARNING
>
> The **root** account has complete control over the system. If unauthorized personnel gain access to the account, they can access or delete users' personal files.

Procedure

1. From the **Installation Summary** window, select **User Settings > Root Account** The **Root Account** window opens.
   By default, the **Disable root account** option is selected.
2. To enable root account, select the **Enable root account** option.
3. Type your password in the **Root Password** field.
   The root password is case-sensitive and must be at least **eight** characters long containing **numbers**, **letters** (upper and lower case) and **symbols**.
4. Type the same password in the **Confirm** field.
5. Optional: Select the **Allow root SSH login with password** option to enable SSH access (with password) to this system as the root user. By default the password-based SSH root access is disabled.
6. Click **Done** to confirm your root password and return to the **Installation Summary** window.
   If you proceed with a weak password, you must click **Done** twice.

17.4.2. Creating a user account

Create a user account during RHEL installation to establish a non-root user for daily system operations from the **Installation Summary** window. It improves security by avoiding the use of the root account for regular tasks and provides proper user management from the start of system deployment.

+ NOTE: You should only use the root account to perform privileged tasks. Using the root account instead of a non-privileged user account to perform regular tasks can introduce a security risk.

Procedure

1. On the **Installation Summary** window, select **User Settings > User Creation** The **Create User** window opens.
2. Type the user account name into the **Full name** field, for example: John Smith.
3. Type the username into the **User name** field, for example: jsmith.

The **User name** is used to log in from a command line; if you install a graphical environment, then your graphical login manager uses the **Full name**.

4. The **Add administrative privileges to this...**option is selected by default. Deselect this option if you do not want to share administrative privileges to this account. By default, new users have administrative privileges to the system.
   An administrator user can use the **sudo** command to perform tasks that are only available to **root** by using the user password, instead of the **root** password. Though it is more convenient, it can also introduce a security risk.

5. The **Require a password to use this account**option is selected by default. Disable it if you want to use this account without a password.
   If you give administrator privileges to a user, ensure the account is password protected. Never give a user administrator privileges without assigning a password to the account.

6. Type a password into the **Password** field.

7. Type the same password into the **Confirm password** field.

8. Click **Done** to apply the changes and return to the **Installation Summary** window.

17.4.3. Editing advanced user settings

You can configure advanced user account settings during RHEL installation to customize home directories, user and group IDs, and group memberships. This provides fine-grained control over user account configuration and system security policies.

Procedure

1. On the **Create User** window, click **Advanced**.

2. Edit the details in the **Home directory** field, if required. The field is populated by default with **/home/*username***.

3. In the **User and Groups IDs**section you can:

   a. Select the **Specify a user ID manually**check box and use **+** or **-** to enter the required value.
      The default value is 1000. User IDs (UIDs) 0-999 are reserved by the system so they cannot be assigned to a user.

   b. Select the **Specify a group ID manually**check box and use **+** or **-** to enter the required value.
      The default group name is the same as the user name, and the default Group ID (GID) is 1000. GIDs 0-999 are reserved by the system so they cannot be assigned to a user group.

4. Specify additional groups as a comma-separated list in the **Group Membership** field. Groups that do not already exist are created; you can specify custom GIDs for additional groups in parentheses. If you do not specify a custom GID for a new group, the new group receives a GID automatically.
   The user account created always has one default group membership (the user’s default group with an ID set in the **Specify a group ID manually**field).

5. Click **Save Changes** to apply the updates and return to the **Create User** window.

17.5. CONFIGURING MANUAL PARTITIONING

You can use manual partitioning to configure your disk partitions and mount points and define the file system that Red Hat Enterprise Linux is installed on. Before installation, you should consider whether you want to use partitioned or unpartitioned disk devices. For more information about the advantages and disadvantages of partitioning on LUNs, either directly or with LVM, see the Red Hat Knowledgebase solution at advantages and disadvantages to using partitioning on LUNs .

You have different partitioning and storage options available, including **Standard Partitions**, **LVM**, and **LVM thin provisioning**. These options provide various benefits and configurations for managing your system's storage effectively.

Standard partition

A standard partition contains a file system or swap space. Standard partitions are most commonly used for /**boot** and the **BIOS Boot** and **EFI System partitions**. You can use the LVM logical volumes in most other uses.

LVM

Choosing **LVM** (or Logical Volume Management) as the device type creates an LVM logical volume. LVM improves performance when using physical disks, and it allows for advanced setups such as using multiple physical disks for one mount point, and setting up software RAID for increased performance, reliability, or both.

LVM thin provisioning

With thin provisioning, you can manage a storage pool of free space, known as a thin pool, which can be allocated to an arbitrary number of devices when needed by applications. You can dynamically expand the pool when needed for cost-effective allocation of storage space.

An installation of Red Hat Enterprise Linux requires a minimum of one partition but uses at least the following partitions or volumes: /, /**home**, /**boot**, and **swap**. You can also create additional partitions and volumes as you require.

To prevent data loss it is recommended that you backup your data before proceeding. If you are upgrading or creating a dual-boot system, you should back up any data you want to keep on your storage devices.

17.5.1. Recommended partitioning scheme

Create separate file systems at the following mount points. However, if required, you can also create the file systems at /**usr**, /**var**, and /**tmp** mount points.

- /**boot**
- / (root)
- /**home**
- **swap**
- /**boot/efi**
- **PReP**

This partition scheme is recommended for bare metal deployments and it does not apply to virtual and cloud deployments.

/**boot** partition - recommended size at least 1 GiB

The partition mounted on /**boot** contains the operating system kernel, which allows your system to boot Red Hat Enterprise Linux 10, along with files used during the bootstrap process. Due to the limitations of most firmwares, create a small partition to hold these. In most scenarios, a 1 GiB boot partition is adequate. Unlike other mount points, using an LVM volume for /**boot** is not possible - /**boot** must be located on a separate disk partition.

If you have a RAID card, be aware that some BIOS types do not support booting from the RAID card. In such a case, the /**boot** partition must be created on a partition outside of the RAID array, such as on a separate disk.

WARNING

- Normally, the /**boot** partition is created automatically by the installation program. However, if the / (root) partition is larger than 2 TiB and (U)EFI is used for booting, you need to create a separate /**boot** partition that is smaller than 2 TiB to boot the machine successfully.

- Ensure the /**boot** partition is located within the first 2 TB of the disk while manual partitioning. Placing the /**boot** partition beyond the 2 TB boundary might result in a successful installation, but the system fails to boot because BIOS cannot read the /**boot** partition beyond this limit.

root - recommended size of at least 10 GiB

This is where "/", or the root directory, is located. The root directory is the top-level of the directory structure. By default, all files are written to this file system unless a different file system is mounted in the path being written to, for example, /**boot** or /**home**.
While a 5 GiB root file system allows you to install a minimal installation, it is recommended to allocate at least 10 GiB so that you can install as many package groups as you want.

Do not confuse the / directory with the /**root** directory. The /**root** directory is the home directory of the root user. The /**root** directory is sometimes referred to as *slash root* to distinguish it from the root directory.

/home - recommended size at least 1 GiB

To store user data separately from system data, create a dedicated file system for the /**home** directory. Base the file system size on the amount of data that is stored locally, number of users, and so on. You can upgrade or reinstall Red Hat Enterprise Linux 10 without erasing user data files. If you select automatic partitioning, it is recommended to have at least 55 GiB of disk space available for the installation, to ensure that the /**home** file system is created.

swap partition - recommended size at least 1 GiB

Swap file systems support virtual memory; data is written to a swap file system when there is not enough RAM to store the data your system is processing. Swap size is a function of system memory workload, not total system memory and therefore is not equal to the total system memory size. It is important to analyze what applications a system will be running and the load those applications will serve in order to determine the system memory workload. Application providers and developers can provide guidance.
When the system runs out of swap space, the kernel terminates processes as the system RAM

memory is exhausted. Configuring too much swap space results in storage devices being allocated but idle and is a poor use of resources. Too much swap space can also hide memory leaks. The maximum size for a swap partition and other additional information can be found in the **mkswap(8)** manual page.

The following table provides the recommended size of a swap partition depending on the amount of RAM in your system and if you want sufficient memory for your system to hibernate. If you let the installation program partition your system automatically, the swap partition size is established using these guidelines. Automatic partitioning setup assumes hibernation is not in use. The maximum size of the swap partition is limited to 10 percent of the total size of the disk, and the installation program cannot create swap partitions more than 1TiB. To set up enough swap space to allow for hibernation, or if you want to set the swap partition size to more than 10 percent of the system’s storage space, or more than 1TiB, you must edit the partitioning layout manually.

Table 17.1. Recommended system swap space

Amount of RAM in the system	Recommended swap space	Recommended swap space if allowing for hibernation
Less than 2 GiB	2 times the amount of RAM	3 times the amount of RAM
2 GiB - 8 GiB	Equal to the amount of RAM	2 times the amount of RAM
8 GiB - 64 GiB	4 GiB to 0.5 times the amount of RAM	1.5 times the amount of RAM
More than 64 GiB	Workload dependent (at least 4GiB)	Hibernation not recommended

/**boot**/**efi** partition - recommended size of 500 MiB

UEFI-based AMD64, Intel 64, and 64-bit ARM require a 500 MiB EFI system partition. The recommended minimum size is 500 MiB and the maximum size is 600 MiB. BIOS systems do not require an EFI system partition.

At the border between each range, for example, a system with 2 GiB, 8 GiB, or 64 GiB of system RAM, discretion can be exercised with regard to chosen swap space and hibernation support. If your system resources allow for it, increasing the swap space can lead to better performance.

Distributing swap space over multiple storage devices - particularly on systems with fast drives, controllers and interfaces - also improves swap space performance.

Many systems have more partitions and volumes than the minimum required. Choose partitions based on your particular system needs. If you are unsure about configuring partitions, accept the automatic default partition layout provided by the installation program.

PReP boot partition - recommended size of 4 to 8 MiB

When installing Red Hat Enterprise Linux on IBM Power System servers, the first partition of the disk should include a **PReP** boot partition. This contains the GRUB2 boot loader, which allows other IBM Power Systems servers to boot Red Hat Enterprise Linux.

BIOS boot partition - recommended size of 1 MiB

When installing Red Hat Enterprise Linux on a BIOS system with a disk partitioned using GPT (GUID Partition Table), which is the default, a separate partition is required in order to boot from the particular disk. This partition is called **BIOS** boot and is used to store the boot loader.

NOTE

Allocate storage capacity only to the partitions that are required immediately. Additional free space can be assigned at any time to address future requirements.

17.5.2. Supported hardware storage

It is important to understand how storage technologies are configured and how support for them may have changed between major versions of Red Hat Enterprise Linux.

Hardware RAID

Any RAID functions provided by the mainboard of your computer, or attached controller cards, need to be configured before you begin the installation process. Each active RAID array appears as one drive within Red Hat Enterprise Linux.

Software RAID

On systems with more than one disk, you can use the Red Hat Enterprise Linux installation program to operate several of the drives as a Linux software RAID array. With a software RAID array, RAID functions are controlled by the operating system rather than the dedicated hardware.

NOTE

When pre-existing RAID array's member devices are all unpartitioned disks/drives, the installation program treats the array as a disk and there is no method provided by the installation program to remove the array.

USB Disks

You can connect and configure external USB storage after installation. Most devices are recognized by the kernel, but some devices may not be recognized. If it is not a requirement to configure these disks during installation, disconnect them to avoid potential problems.

Considerations for Intel BIOS RAID Sets

Red Hat Enterprise Linux uses **mdraid** for installing on Intel BIOS RAID sets. These sets are automatically detected during the boot process and their device node paths can change across several booting processes. Replace device node paths (such as **/dev/sda**) with file system labels or device UUIDs. You can find the file system labels and device UUIDs by using the **blkid** command.

17.5.3. Starting manual partitioning

You can configure custom disk partitioning during RHEL installation to create a storage layout that meets your specific requirements. Manual partitioning provides full control over disk usage, mount points, and file system types for optimized system performance and data organization.

Prerequisites

- The **Installation Summary** screen is open.
- All disks are available to the installation program.

Procedure

1. Select disks for installation:
    a. Click **Installation Destination** to open the **Installation Destination** window.
    b. Select the disks that you require for installation by clicking the corresponding icon. A selected disk has a check-mark displayed on it.
    c. Under **Storage Configuration**, select the **Custom** radio-button.
    d. Click **Done**.
2. Detected mount points are listed in the left-hand pane. The mount points are organized by detected operating system installations. As a result, some file systems may be displayed multiple times if a partition is shared among several installations.
    a. Select the mount points in the left pane; the options that can be customized are displayed in the right pane.
    b. Optional: If your system contains existing file systems, ensure that enough space is available for the installation. To remove any partitions, select them in the list and click the **-** button. The dialog has a check box that you can use to remove all other partitions used by the system to which the deleted partition belongs.
    c. Optional: If there are no existing partitions and you want to create a set of partitions as a starting point, select your preferred partitioning scheme from the left pane (default for Red Hat Enterprise Linux is LVM) and click the **Click here to create them automatically** link.

    > NOTE
    >
    > A /**boot** partition, a / (root) volume, a **swap** volume proportional to the size of the available storage and optionally some other partitions, depending on system properties, such as architecture, are created and listed in the left pane. These are the file systems for a typical installation, but you can add additional file systems and mount points.

3. Optional: Continue with adding mount points and configuring the individual mount points.
4. Click **Done** to confirm any changes and return to the **Installation Summary** window.

17.5.4. Supported file systems

When configuring manual partitioning, you can optimize performance, ensure compatibility, and effectively manage disk space by utilizing the various file systems and partition types available in Red Hat Enterprise Linux.

xfs

The XFS file system is the default file system on Red Hat Enterprise Linux. It is a highly scalable, high-performance file system that supports file system size up to 16 exabytes (approximately 16 million terabytes), files up to 8 exabytes (approximately 8 million terabytes), and directory structures containing tens of millions of entries. **XFS** also supports metadata journaling, which facilitates quicker crash recovery. The maximum supported size of a single XFS file system is 1 PB. XFS cannot be shrunk to get free space.

ext4

The **ext4** file system is based on the **ext3** file system and features a number of improvements. These include support for larger file systems and larger files, faster and more efficient allocation of disk space, no limit on the number of subdirectories within a directory, faster file system checking, and more robust journaling. The maximum supported size of a single **ext4** file system is 50 TB.

ext3

The **ext3** file system is based on the **ext2** file system and has one main advantage - journaling. Using a journaling file system reduces the time spent recovering a file system after it terminates unexpectedly, as there is no need to check the file system for metadata consistency by running the fsck utility every time.

ext2

An **ext2** file system supports standard Unix file types, including regular files, directories, or symbolic links. It provides the ability to assign long file names, up to 255 characters.

swap

Swap partitions are used to support virtual memory. In other words, data is written to a swap partition when there is not enough RAM to store the data your system is processing.

vfat

The **VFAT** file system is a Linux file system that is compatible with Microsoft Windows long file names on the FAT file system.

> NOTE
>
> Support for the **VFAT** file system is not available for Linux system partitions. For example, /, **/var**, **/usr** and so on.

BIOS Boot

A very small partition required for booting from a device with a GUID partition table (GPT) on BIOS systems and UEFI systems in BIOS compatibility mode.

EFI System Partition

A small partition required for booting a device with a GUID partition table (GPT) on a UEFI system.

PReP

This small boot partition is located on the first partition of the disk. The **PReP** boot partition contains the GRUB2 boot loader, which allows other IBM Power Systems servers to boot Red Hat Enterprise Linux.

Additional resources

- Red Hat Enterprise Linux Technology Capabilities and Limits (Red Hat Knowledgebase)

17.5.5. Adding a file system mount point

You can add multiple file system mount points. You can use any of the file systems and partition types available, such as XFS, ext4, ext3, ext2, swap, and VFAT. You can also use specific partitions such as BIOS Boot, EFI System Partition, and PReP. It helps you to effectively configure your system’s storage.

Prerequisites

- You have planned your partitions.

- Ensure you haven't specified mount points at paths with symbolic links, such as **/var/mail**, **/usr/tmp**, **/lib**, **/sbin**, **/lib64**, and **/bin**. The payload, including RPM packages, depends on creating symbolic links to specific directories.

Procedure

1. Click **+** to create a new file system and related mount point. The **Add a New Mount Point** dialog opens.
2. Select one of the preset paths from the **Mount Point** drop-down menu or type your own; for example, select / for the root partition or **/boot** for the boot partition.
3. Enter the size of the file system into the **Desired Capacity** field; for example, **2GiB**.
   If you do not specify a value in **Desired Capacity**, or if you specify a size bigger than available space, then all remaining free space is used.
4. Click **Add mount point** to create the partition and return to the **Manual Partitioning** window.

17.5.6. Configuring storage for a mount point file system

You can set the partitioning scheme for each mount point that was created manually. The available options are **Standard Partition**, **LVM**, and **LVM Thin Provisioning**. Btfrs support has been removed in Red Hat Enterprise Linux 10.

NOTE

The **/boot** partition is always located on a standard partition, regardless of the value selected.

Procedure

1. To change the devices that a single non-LVM mount point should be located on, select the required mount point from the left-hand pane.
2. Under the **Device(s)** heading, click **Modify**. The **Configure Mount Point** dialog opens.
3. Select one or more devices and click **Select** to confirm your selection and return to the **Manual Partitioning** window.
4. Click **Update Settings** to apply the changes.
5. In the lower left-hand side of the **Manual Partitioning** window, click the **storage device selected** link to open the **Selected Disks** dialog and review disk information.
6. Optional: Click the **Rescan** button (circular arrow button) to refresh all local disks and partitions; this is only required after performing advanced partition configuration outside the installation program. Clicking the **Rescan Disks** button resets all configuration changes made in the installation program.

17.5.7. Customizing a mount point file system

You can customize a partition or volume if you want to set specific settings. If **/usr** or **/var** is partitioned separately from the rest of the root volume, the boot process becomes complex. This is because critical components are located in these directories. In some situations, such as when these directories are placed on an iSCSI drive or an FCoE location, the system is unable to boot. Alternatively, the system may hung up with a **Device is busy** error when powering off or rebooting.

This limitation only applies to /**usr** or /**var**, not to directories below them. For example, a separate partition for /**var**/**www** works successfully.

Procedure

1. From the left pane, select the mount point.

2. From the right-hand pane, you can customize the following options:

   a. Enter the file system mount point into the **Mount Point** field. For example, if a file system is the root file system, enter /; enter /**boot** for the /**boot** file system, and so on. For a swap file system, do not set the mount point as setting the file system type to **swap** is sufficient.

   b. Enter the size of the file system in the **Desired Capacity** field. You can use common size units such as KiB or GiB. The default is MiB if you do not set any other unit.

   c. Select the device type that you require from the drop-down **Device Type** menu: **Standard Partition**, **LVM**, or **LVM Thin Provisioning**.

      NOTE

      RAID is available only if two or more disks are selected for partitioning. If you choose **RAID**, you can also set the **RAID Level**. Similarly, if you select **LVM**, you can specify the **Volume Group**.

   d. Select the **Encrypt** check box to encrypt the partition or volume. You must set a password later in the installation program.

   e. Select the appropriate file system type for this partition or volume from the **File system** drop-down menu.

      NOTE

      Support for the **VFAT** file system is not available for Linux system partitions. For example, /, /**var**, /**usr**, and so on.

   f. Select the **Reformat** check box to format an existing partition, or clear the **Reformat** check box to retain your data. The newly-created partitions and volumes must be reformatted, and the check box cannot be cleared.

   g. Type a label for the partition in the **Label** field. Use labels to easily recognize and address individual partitions.

   h. Type a name in the **Name** field. The standard partitions are named automatically when they are created and you cannot edit the names of standard partitions. For example, you cannot edit the /**boot** name **sda1**.

3. Click **Update Settings** to apply your changes and if required, select another partition to customize. Changes are not applied until you click **Begin Installation** from the **Installation Summary** window.

4. Optional: Click **Reset All** to discard your partition changes.

5. Click **Done** when you have created and customized all file systems and mount points. If you choose to encrypt a file system, you are prompted to create a passphrase.
   A **Summary of Changes** dialog box opens, displaying a summary of all storage actions for the installation program.

6. Click **Accept Changes** to apply the changes and return to the **Installation Summary** window.

17.5.8. Preserving the /home directory

In a Red Hat Enterprise Linux 10 graphical installation, you can preserve the **/home** directory that was used on your RHEL 9 system. Preserving **/home** is only possible if the **/home** directory is located on a separate **/home** partition on your RHEL 9 system.

Preserving the **/home** directory that includes various configuration settings, makes it possible that the GNOME Shell environment on the new Red Hat Enterprise Linux 10 system is set in the same way as it was on your RHEL 9 system. Note that this applies only for users on Red Hat Enterprise Linux 10 with the same user name and ID as on the previous RHEL 9 system.

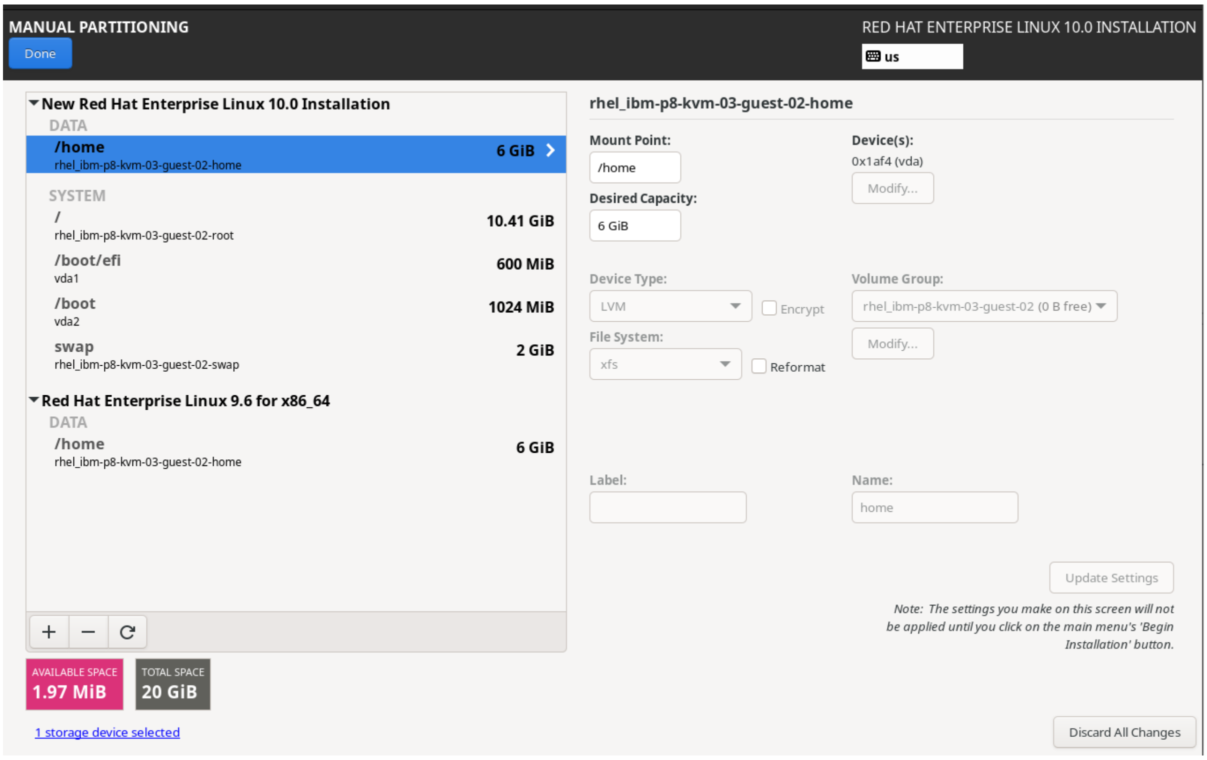
Prerequisites

- You have RHEL 9 installed on your computer.
- The **/home** directory is located on a separate **/home** partition on your RHEL 9 system.
- The Red Hat Enterprise Linux 10 **Installation Summary** window is open.

Procedure

1. Click **Installation Destination** to open the **Installation Destination** window.
2. Under **Storage Configuration**, select the **Custom** radio button. Click **Done**.
3. Click **Done**, the **Manual Partitioning** window opens.
4. Choose the **/home** partition, fill in **/home** under **Mount Point:** and clear the **Reformat** check box.

Figure 17.1. Ensuring that /home is not formatted



MANUAL PARTITIONING

Done

RED HAT ENTERPRISE LINUX 10.0 INSTALLATION

us

▾ New Red Hat Enterprise Linux 10.0 Installation

DATA

/home 6 GiB
rhel_ibm-p8-kvm-03-guest-02-home

SYSTEM

/ 10.41 GiB
rhel_ibm-p8-kvm-03-guest-02-root

/boot/efi 600 MiB
vda1

/boot 1024 MiB
vda2

swap 2 GiB
rhel_ibm-p8-kvm-03-guest-02-swap

▾ Red Hat Enterprise Linux 9.6 for x86_64

DATA

/home 6 GiB
rhel_ibm-p8-kvm-03-guest-02-home

+ − ⟳

AVAILABLE SPACE 1.97 MiB

TOTAL SPACE 20 GiB

1 storage device selected

rhel_ibm-p8-kvm-03-guest-02-home

Mount Point: /home

Desired Capacity: 6 GiB

Device(s): 0x1af4 (vda)

Modify...

Device Type: LVM

□ Encrypt

Volume Group: rhel_ibm-p8-kvm-03-guest-02 (0 B free)

Modify...

File System: xfs

□ Reformat

Label:

Name: home

Update Settings

Note: The settings you make on this screen will not be applied until you click on the main menu's 'Begin Installation' button.

Discard All Changes

5. Optional: You can also customize various aspects of the /**home** partition required for your Red Hat Enterprise Linux 10 system as described in Customizing a mount point file system. However, to preserve /**home** from your RHEL 9 system, it is necessary to clear the **Reformat** check box.

6. After you customized all partitions according to your requirements, click **Done**. The **Summary of changes** dialog box opens.

7. Verify that the **Summary of changes** dialog box does not show any change for /**home**. This means that the /**home** partition is preserved.

8. Click **Accept Changes** to apply the changes, and return to the **Installation Summary** window.

17.5.9. Creating a software RAID during the installation

Redundant Arrays of Independent Disks (RAID) devices are constructed from multiple storage devices. These devices are arranged to provide increased performance and, in some configurations, greater fault tolerance. A RAID device is created in one step and disks are added or removed as necessary. \

You can configure one RAID partition for each physical disk in your system. The number of disks available to the installation program determines the levels of RAID device available. For example, if your system has two disks, you cannot create a **RAID 10** device, as it requires a minimum of three separate disks. To optimize your system's storage performance and reliability, RHEL supports software **RAID 0**, **RAID 1**, **RAID 4**, **RAID 5**, **RAID 6**, and **RAID 10** types with LVM and LVM Thin Provisioning to set up storage on the installed system.

NOTE

On 64-bit IBM Z, the storage subsystem uses RAID transparently. You do not have to configure software RAID manually.

Prerequisites

- You have selected two or more disks for installation before RAID configuration options are visible. Depending on the RAID type you want to create, at least two disks are required.
- You have selected the **Custom** radio button on the **Installation Destination** window and you have entered the Manual Partitioning window by clicking on the **Done** button.
- You have created a mount point. By configuring a mount point, you can configure the RAID device.

Procedure

1. From the left pane of the **Manual Partitioning** window, select the required partition.
2. Click the **Device Type** drop-down menu and select **RAID**.
3. Under the **Device(s)** section, click **Modify**. The **Configure Mount Point** dialog box opens.
4. Select the disks that you want to include in the RAID device and click **Select** to close the dialog box.
5. Click the **File System** drop-down menu and select your preferred file system type.
6. Click the **RAID Level** drop-down menu and select your preferred level of RAID.
7. Click **Update Settings** to save your changes.
8. Click **Done** to apply the settings to return to the **Installation Summary** window.

Additional resources

- Creating a RAID LV with DM integrity
- Managing RAID

17.5.10. Creating an LVM logical volume

Logical Volume Management (LVM) presents a logical view of underlying physical storage space, such as disks or LUNs. Partitions on physical storage are represented as physical volumes that you can group together into volume groups. You can divide each volume group into multiple logical volumes, each of which is similar to a standard disk partition. Therefore, LVM logical volumes function as partitions that can span multiple physical disks.

IMPORTANT

- LVM configuration is available only in the graphical and the Kickstart installation. During text-mode installation, LVM configuration is not available.
- To create an LVM configuration in the text mode, press **Ctrl**+**Alt**+**F2** to use a shell prompt in a different virtual console. You can run **vgcreate** and **lvm** commands in this shell. To return to the text-mode installation, press **Ctrl**+**Alt**+**F1**.

Procedure

1. From the **Manual Partitioning** window, create a new mount point by using any of the following options:
   - Use the **Click here to create them automatically** option or click the **+** button.
   - Select Mount Point from the drop-down list or enter manually.
   - Enter the size of the file system in to the **Desired Capacity** field; for example, 70 GiB for **/**, 1 GiB for **/boot**.
     Note: Skip this step to use the existing mount point.
2. Select the mount point.
3. Select **LVM** in the drop-down menu. The **Volume Group** drop-down menu is displayed with the newly-created volume group name.

   NOTE

   You cannot specify the size of the volume group's physical extents in the configuration dialog. The size is always set to the default value of 4 MiB. If you want to create a volume group with different physical extents, you must create it manually by switching to an interactive shell and by using the **vgcreate** command, or use a Kickstart file with the **volgroup --pesize=*size*** command.
4. Click **Done** to return to the **Installation Summary** window.

Additional resources

- Configuring and managing logical volumes

17.5.11. Configuring an LVM logical volume

You can configure a newly-created LVM logical volume based on your requirements.

> WARNING
>
> Placing the **/boot** partition on an LVM volume is not supported.

Procedure

1. From the **Manual Partitioning** window, create a mount point by using any of the following options:
   - Use the **Click here to create them automatically** option or click the **+** button.
   - Select Mount Point from the drop-down list or enter manually.
   - Enter the size of the file system in to the **Desired Capacity** field; for example, 70 GiB for **/**, 1 GiB for **/boot**.
     Skip this step to use the existing mount point.

2. Select the mount point.
3. Click the **Device Type** drop-down menu and select **LVM**. The **Volume Group** drop-down menu is displayed with the newly-created volume group name.
4. Click **Modify** to configure the newly-created volume group. The **Configure Volume Group** dialog box opens.

> NOTE
>
> You cannot specify the size of the volume group's physical extents in the configuration dialog. The size is always set to the default value of 4 MiB. If you want to create a volume group with different physical extents, you must create it manually by switching to an interactive shell and by using the **vgcreate** command, or use a Kickstart file with the **volgroup --pesize=*size*** command. For more information, see the Automatically installing RHEL document.

5. Optional: From the **RAID Level** drop-down menu, select the RAID level that you require. The available RAID levels are the same as with actual RAID devices.
6. Select the **Encrypt** check box to mark the volume group for encryption.
7. From the **Size policy** drop-down menu, select any of the following size policies for the volume group:
   The available policy options are:

   Automatic

   The size of the volume group is set automatically so that it is large enough to contain the configured logical volumes. This is optimal if you do not need free space within the volume group.

   As large as possible

   The volume group is created with maximum size, regardless of the size of the configured logical volumes it contains. This is optimal if you plan to keep most of your data on LVM and later need to increase the size of some existing logical volumes, or if you need to create additional logical volumes within this group.

   Fixed

   You can set an exact size of the volume group. Any configured logical volumes must then fit within this fixed size. This is useful if you know exactly how large you need the volume group to be.
8. Click **Save** to apply the settings and return to the **Manual Partitioning** window.
9. Click **Update Settings** to save your changes.
10. Click **Done** to return to the **Installation Summary** window.

17.5.12. Understanding disk partitioning strategies

There is no best way to partition every system; the optimal setup depends on how you plan to use the system being installed. However, the following tips may help you find the optimal layout for your needs:

- Create partitions that have specific requirements first, for example, if a particular partition must be on a specific disk.

- Consider encrypting any partitions and volumes which might contain sensitive data. Encryption prevents unauthorized people from accessing the data on the partitions, even if they have access to the physical storage device. In most cases, you should at least encrypt the **/home** partition, which contains user data.

- In some cases, creating separate mount points for directories other than /, **/boot** and **/home** may be useful. For example, on a server running a **MySQL** database, having a separate mount point for **/var/lib/mysql** allows you to preserve the database during a reinstallation without having to restore it from backup afterward. However, having unnecessary separate mount points will make storage administration more difficult.

- Some special restrictions apply to certain directories with regards to which partitioning layouts can be placed. Notably, the **/boot** directory must always be on a physical partition (not on an LVM volume).

- If you are new to Linux, consider reviewing the Linux File system Hierarchy Standard for information about various system directories and their contents.

- Each kernel requires approximately: 60MiB (initrd 28MiB, 16MiB vmlinuz, and 9MiB System.map)

- For rescue mode: 100MiB (initrd 56MiB, 16MiB vmlinuz, and 9MiB System map)

- When **kdump** is enabled in system it will take approximately another 40MiB (another initrd with 33MiB)
  The default partition size of 1 GiB for **/boot** should suffice for most common use cases. However, increase the size of this partition if you are planning on retaining multiple kernel releases or errata kernels.

- The **/var** directory holds content for a number of applications, including the Apache web server, and is used by the DNF package manager to temporarily store downloaded package updates. Make sure that the partition or volume containing **/var** has at least 5 GiB.

- The **/usr** directory holds the majority of software on a typical Red Hat Enterprise Linux installation. The partition or volume containing this directory should therefore be at least 5 GiB for minimal installations, and at least 10 GiB for installations with a graphical environment.

- If **/usr** or **/var** is partitioned separately from the rest of the root volume, the boot process becomes much more complex because these directories contain boot-critical components. In some situations, such as when these directories are placed on an iSCSI drive or an FCoE location, the system may either be unable to boot, or it may hang with a **Device is busy** error when powering off or rebooting.
  This limitation only applies to **/usr** or **/var**, not to directories under them. For example, a separate partition for **/var/www** works without issues.

  IMPORTANT

  Some security policies require the separation of **/usr** and **/var**, even though it makes administration more complex.

- Consider leaving a portion of the space in an LVM volume group unallocated. This unallocated space gives you flexibility if your space requirements change but you do not wish to remove data from other volumes. You can also select the **LVM Thin Provisioning** device type for the partition to have the unused space handled automatically by the volume.

- The size of an XFS file system cannot be reduced - if you need to make a partition or volume with this file system smaller, you must back up your data, destroy the file system, and create a new, smaller one in its place. Therefore, if you plan to alter your partitioning layout later, you should use the **ext4** file system instead.

- Use Logical Volume Management (LVM) if you anticipate expanding your storage by adding more disks or expanding virtual machine disks after the installation. With LVM, you can create physical volumes on the new drives, and then assign them to any volume group and logical volume as you see fit - for example, you can easily expand your system's **/home** (or any other directory residing on a logical volume).

- Creating a BIOS Boot partition or an EFI System Partition may be necessary, depending on your system's firmware, boot drive size, and boot drive disk label. Note that you cannot create a BIOS Boot or EFI System Partition in graphical installation if your system does **not** require one - in that case, they are hidden from the menu.

Additional resources

- How to use dm-crypt on IBM Z, LinuxONE and with the PAES cipher

17.6. SELECTING THE BASE ENVIRONMENT AND ADDITIONAL SOFTWARE

You can choose the appropriate base environment and additional software packages during RHEL installation from the **Software Selection** window to customize your system for specific use cases. It determines the software stack that will be installed and affects system performance, security, and functionality.

- **Base Environment** contains predefined packages. You can select only one base environment, for example, Server with GUI (default), Server, Minimal Install, Workstation, Custom operating system, Virtualization Host. The availability is dependent on the installation ISO image that is used as the installation source.

- **Additional Software for Selected Environment**contains additional software packages for the base environment. You can select multiple software packages.

Use a predefined environment and additional software to customize your system. However, in a standard installation, you cannot select individual packages to install. To view the packages contained in a specific environment, see the ***repository*/repodata/*-comps-*repository*.*architecture*.xml** file on your installation source media (DVD, CD, USB). The XML file contains details of the packages installed as part of a base environment. Available environments are marked by the **<environment>** tag, and additional software packages are marked by the **<group>** tag.

If you are unsure about which packages to install, select the **Minimal Install** base environment. Minimal install installs a basic version of Red Hat Enterprise Linux with only a minimal amount of additional software. After the system finishes installing and you log in for the first time, you can use the DNF package manager to install additional software.

NOTE

- Use the **dnf group list** command from any RHEL 10 system to view the list of packages being installed on the system as a part of software selection.
- If you need to control which packages are installed, you can use a Kickstart file and define the packages in the **%packages** section. For more information, see the Automatically installing RHEL document.
- By default, RHEL 10 does not install the TuneD package. You can manually install the TuneD package by using the **dnf install tuned** command.

Prerequisites

- You have configured the installation source.
- The installation program has downloaded package metadata.
- The **Installation Summary** window is open.

Procedure

1. From the **Installation Summary** window, click **Software Selection**. The **Software Selection** window opens.
2. From the **Base Environment** pane, select a base environment. You can select only one base environment, for example, Server with GUI (default), Server, Minimal Install, Workstation, Custom Operating System, Virtualization Host. By default, the **Server with GUI** base environment is selected.
3. Optional: For installations on ARM based systems, select desired **Page size** from **Kernel Options**.
   By default, Kernel with a **4k** page size is selected.
4. From the **Additional Software for Selected Environment** pane, select one or more options.
5. Click **Done** to apply the settings and return to graphical installations.

Additional resources

- The 4k and 64k page size Kernel Options

17.7. OPTIONAL: CONFIGURING THE NETWORK AND HOST NAME

You can use the **Network and Host name** window to configure network interfaces. Options that you select here are available both during the installation for tasks such as downloading packages from a remote location, and on the installed system.

17.7.1. Configuring network and host name

You can configure network interfaces and set the system hostname during RHEL installation from the **Network and Hostname** section. It enables network connectivity for package downloads, remote installations, and ensures proper system identification on your network infrastructure.

Procedure

1. From the **Installation Summary** window, click **Network and Host Name**.
2. From the list in the left-hand pane, select an interface. The details are displayed in the right-hand pane.
3. Toggle the **ON/OFF** switch to enable or disable the selected interface.
   You cannot add or remove interfaces manually.
4. Click **+** to add a virtual network interface, which can be either: Bond, Bridge, or VLAN.
5. Click **-** to remove a virtual interface.
6. Click **Configure** to change settings such as IP addresses, DNS servers, or routing configuration for an existing interface (both virtual and physical).
7. Type a host name for your system in the **Host Name** field.
   The host name can either be a fully qualified domain name (FQDN) in the format ***hostname.domainname***, or a short hostname without the domain. Many networks have a Dynamic Host Configuration Protocol (DHCP) service that automatically supplies connected systems with a domain name. To allow the DHCP service to assign the domain name to this system, specify only the short host name.

   Host names can only contain alphanumeric characters and **-** or **.**. Host name should be equal to or less than 64 characters. Host names cannot start or end with **-** and **.**. To be compliant with DNS, each part of a FQDN should be equal to or less than 63 characters and the FQDN total length, including dots, should not exceed 255 characters.

   The value **localhost** means that no specific static host name for the target system is configured, and the actual host name of the installed system is configured during the processing of the network configuration, for example, by NetworkManager using DHCP or DNS.

   When using static IP and host name configuration, it depends on the planned system use case whether to use a short name or FQDN. Red Hat Identity Management configures FQDN during provisioning but some 3rd party software products may require a short name. In either case, to ensure availability of both forms in all situations, add an entry for the host in **/etc/hosts** in the format **IP FQDN short-alias**.
8. Click **Apply** to apply the host name to the installer environment.
9. Alternatively, in the **Network and Hostname** window, you can choose the Wireless option. Click **Select network** in the right-hand pane to select your wifi connection, enter the password if required, and click **Connect**.

Additional resources

- Automatically installing RHEL
- Configuring and managing networking

17.7.2. Adding a virtual network interface

You can add virtual network interfaces during RHEL installation to configure advanced networking features such as bonding, bridging, and VLANs. It enables high availability, load balancing, and network segmentation for improved network performance and reliability.

Procedure

1. From the **Network & Host name**window, click the **+** button to add a virtual network interface. The **Add a device** dialog opens.
2. Select one of the four available types of virtual interfaces:
   - **Bond**: NIC (*Network Interface Controller*) Bonding, a method to bind multiple physical network interfaces together into a single bonded channel.
   - **Bridge**: Represents NIC Bridging, a method to connect multiple separate networks into one aggregate network.
   - **Vlan** (*Virtual LAN*): A method to create multiple distinct broadcast domains which are mutually isolated.
3. Select the interface type and click **Add**. An editing interface dialog box opens, allowing you to edit any available settings for your chosen interface type. For more information, see Editing network interface.
4. Click **Save** to confirm the virtual interface settings and return to the **Network & Host name** window.
5. Optional: To change the settings of a virtual interface, select the interface and click **Configure**.

17.7.3. Editing network interface configuration

You can edit network interface configuration during RHEL installation to customize network settings for wired and wireless connections. This enables configuration of IP addresses, DNS settings, routing, and other network parameters to meet specific network requirements.

+

NOTE

On 64-bit IBM Z, you cannot add a new connection as the network subchannels need to be grouped and set online beforehand, and this is currently done only in the booting phase.

Procedure

- To configure a network connection manually, select the interface from the **Network and Host name** window and click **Configure**.
  An editing dialog specific to the selected interface opens. The options present depend on the connection type - the available options are slightly different depending on whether the connection type is a physical interface (wired or wireless network interface controller) or a virtual interface (Bond, Bridge, Team (deprecated), or Vlan) that was previously configured in Adding a virtual interface.

17.7.4. Enabling or Disabling the Interface Connection

You can enable or disable network interface connections during RHEL installation to control network access and connectivity. This configuration determines which network interfaces are active and how they connect automatically, affecting system network behavior and security.

Procedure

1. Click the **General** tab.

2. Select the **Connect automatically with priority** check box to enable connection by default. Keep the default priority setting at **0**.

3. Optional: Enable or disable all users on the system from connecting to this network by using the **All users may connect to this network** option. If you disable this option, only **root** will be able to connect to this network.

   IMPORTANT

   When enabled on a wired connection, the system automatically connects during startup or reboot. On a wireless connection, the interface attempts to connect to any known wireless networks in range. For further information about NetworkManager, including the **nm-connection-editor** tool, see the *Configuring and managing networking* document.

4. Click **Save** to apply the changes and return to the **Network and Host name** window.
   It is not possible to only allow a specific user other than **root** to use this interface, as no other users are created at this point during the installation. If you need a connection for a different user, you must configure it after the installation.

17.7.5. Setting up Static IPv4 or IPv6 Settings

By default, both IPv4 and IPv6 use automatic configuration. This means when the interface is connected, IP addresses, DNS, and other settings are set automatically. In many cases, this is sufficient, but you can also provide static configuration in the **IPv4 Settings** and **IPv6 Settings** tabs.

Procedure

1. To set static network configuration, navigate to one of the IPv Settings tabs and from the **Method** drop-down menu, select a method other than **Automatic**, for example, **Manual**. The **Addresses** pane is enabled.

2. Optional: In the **IPv6 Settings** tab, you can also set the method to **Ignore** to disable IPv6 on this interface.

3. Click **Add** and enter your address settings.

4. Type the IP addresses in the **Additional DNS servers** field; it accepts one or more IP addresses of DNS servers, for example, **10.0.0.1,10.0.0.8**.

5. Select the **Require IPv*X* addressing for this connection to complete** check box.
   Selecting this option in the **IPv4 Settings** or **IPv6 Settings** tabs allow this connection only if IPv4 or IPv6 was successful. If this option remains disabled for both IPv4 and IPv6, the interface is able to connect if configuration succeeds on either IP protocol.

6. Click **Save** to apply the changes and return to the **Network & Host name** window.

17.7.6. Configuring Routes

You can control the access of specific connections by configuring routes.

Procedure

1. In the **IPv4 Settings** and **IPv6 Settings** tabs, click **Routes** to configure routing settings for a specific IP protocol on an interface. An editing routes dialog specific to the interface opens.
2. Click **Add** to add a route.
3. Select the **Ignore automatically obtained routes** check box to configure at least one static route and to disable all routes not specifically configured.
4. Select the **Use this connection only for resources on its network** check box to prevent the connection from becoming the default route.
   This option can be selected even if you did not configure any static routes. This route is used only to access certain resources, such as intranet pages that require a local or VPN connection. Another (default) route is used for publicly available resources. Unlike the additional routes configured, this setting is transferred to the installed system. This option is useful only when you configure more than one interface.
5. Click **OK** to save your settings and return to the editing routes dialog that is specific to the interface.
6. Click **Save** to apply the settings and return to the **Network and Host Name** window.

17.8. OPTIONAL: CONFIGURING THE KEYBOARD LAYOUT

You can configure keyboard layouts during RHEL installation from the **Installation Summary** screen to support different languages and regional input methods. It ensures proper character input, special symbols, and keyboard shortcuts for international users and multilingual environments.

+

IMPORTANT

If you use a keyboard layout without Latin characters (for example, **Russian**), add the **English** (**United States**) layout too. You need Latin characters to enter paraphrases or user credentials during the installation process.

Procedure

1. From the **Installation Summary** window, click **Keyboard**.
2. Click **+** to open the **Add a Keyboard Layout** window to change to a different layout.
3. Select a layout by browsing the list or use the **Search** field.
4. Select the required layout and click **Add**. The new layout appears under the default layout.
5. Optional: Click **Options** to configure a keyboard switch that you can use to cycle between available layouts in the installed system (this is not used during the installation). The **Layout Switching Options** window opens.

- To configure key combinations for switching, select one or more key combinations and click **OK** to confirm your selection.

6. Optional: When you select a layout, click the **Keyboard** button to open a new dialog box displaying a visual representation of the selected layout.

7. Click **Done** to apply the settings and return to graphical installations.

17.9. OPTIONAL: CONFIGURING THE LANGUAGE SUPPORT

You can configure additional language support during RHEL installation from the **Installation Summary** screen to enable localization for specific regions and languages. It ensures proper character encoding, date formats, and cultural conventions for international deployments and multilingual environments.

Procedure

1. From the **Installation Summary** window, click **Language Support**. The **Language Support** window opens. The left pane lists the available language groups. If at least one language from a group is configured, a check mark is displayed and the supported language is highlighted.

2. From the left pane, click a group to select additional languages, and from the right pane, select regional options. Repeat this process for all the languages that you want to configure.

3. Optional: Search the language group by typing in the text box, if required.

4. Click **Done** to apply the settings and return to graphical installations.

17.10. OPTIONAL: CONFIGURING THE DATE AND TIME-RELATED SETTINGS

You can configure date, time, and timezone settings during RHEL installation from the **Installation Summary** screen to ensure proper system time synchronization and regional time zone configuration. It includes setting up NTP for automatic time synchronization and selecting appropriate time zones for your location.

Procedure

1. From the **Installation Summary** window, click **Time & Date**. The **Time & Date** window opens.

2. Under **Time zone**, select the **Region** by using the drop-down menu. Select **Etc** as your region to configure a time zone relative to Greenwich Mean Time (GMT) without setting your location to a specific region.

3. From the **City** drop-down menu, select the city, or the city closest to your location in the same time zone.

4. The **Automatic date and time** option is selected by default. It uses network time synchronization by using the Network Time Protocol (NTP). NTP keeps your system time correct as long as the system can access the internet. By default, one NTP pool is configured.

5. Optional: Use the **Configure NTP** button to add a new NTP, or disable or remove the default options.

6. Select **Manual date and time** option to set date and time manually by using the controls displayed on the screen.

7. Click **Done** to apply the settings and return to graphical installations.

17.11. OPTIONAL: SUBSCRIBING THE SYSTEM AND ACTIVATING RED HAT LIGHTSPEED

Red Hat Lightspeed is a Software-as-a-Service (SaaS) offering that provides continuous, in-depth analysis of registered Red Hat-based systems to proactively identify threats to security, performance and stability across physical, virtual and cloud environments, and container deployments. By registering your RHEL system in Red Hat Lightspeed, you gain access to predictive analytics, security alerts, and performance optimization tools, enabling you to maintain a secure, efficient, and stable IT environment.

You can register to Red Hat by using either your Red Hat account or your activation key details. You can connect your system to Red Hat Lightspeed by using the **Connect to Red Hat**option.

Procedure

1. From the **Installation Summary** screen, under **Software**, click **Connect to Red Hat**
2. Select **Account** or **Activation Key**.
    a. If you select **Account**, enter your Red Hat Customer Portal username and password details.
    b. If you select **Activation Key**, enter your organization ID and activation key. You can enter more than one activation key, separated by a comma, as long as the activation keys are registered to your subscription.
3. Select the **Set System Purpose** check box. Setting system purpose values are important for accurate reporting of consumption in the subscription services.
4. Select the required **Role**, **SLA**, and **Usage** from the corresponding drop-down lists.
5. The **Connect to Red Hat Lightspeed**check box is enabled by default. Clear the check box if you do not want to connect to Red Hat Lightspeed.
6. Optional: Expand **Options**.
    a. Select the **Use HTTP proxy** check box if your network environment only allows external Internet access or access to content servers through an HTTP proxy. Clear the **Use HTTP proxy** check box if an HTTP proxy is not used.
7. Click **Register** to register the system. When the system is successfully registered and subscriptions are attached, the **Connect to Red Hat**window displays the attached subscription details.
   Depending on the amount of subscriptions, the registration and attachment process might take up to a minute to complete.
8. Click **Done** to return to the **Installation Summary** window.
   A *Registered* message is displayed under **Connect to Red Hat**

Additional resources

- About Red Hat Lightspeed

17.12. OPTIONAL: USING NETWORK-BASED REPOSITORIES FOR THE INSTALLATION

You can configure an installation source from auto-detected media, Red Hat CDN, or the network. When the **Installation Summary** window opens, the installation source is set based on the boot media. The full Red Hat Enterprise Linux Server DVD configures the source as local media.

Prerequisites

- You have downloaded the full installation DVD ISO or minimal installation Boot ISO image from the Product Downloads page.
- You have created bootable installation media.
- The **Installation Summary** window is open.

Procedure

1. From the **Installation Summary** window, click **Installation Source**. The **Installation Source** window opens.
   The **Auto-detected installation media** option is selected by default if you started the installation program from media containing an installation source, for example, a DVD.
2. Optional: Select the **Red Hat CDN** option to register your system, attach RHEL subscriptions, and install RHEL from the Red Hat Content Delivery Network (CDN).
3. Optional: Select the **On the network** option to download and install packages from a network location instead of local media. This option is available only when a network connection is active. See Configuring network and host name options for information about how to configure network connections in the GUI.
   a. Select the **On the network** drop-down menu to specify the protocol for downloading packages. This setting depends on the server that you want to use.
   b. Type the server address (without the protocol) into the address field. If you choose NFS, a second input field opens where you can specify custom **NFS mount options**. This field accepts options listed in the **nfs(5)** man page on your system.
   c. When selecting an NFS installation source, specify the address with a colon (**:**) character separating the host name from the path. For example, **server.example.com:/path/to/directory**.
      The following steps are optional and are only required if you use a proxy for network access.
   d. Click **Proxy setup** to configure a proxy for an HTTP or HTTPS source.
      i. Select the **Enable HTTP proxy** check box and type the URL into the **Proxy Host** field.
      ii. Select the **Use Authentication** check box if the proxy server requires authentication.
      iii. Type in your user name and password.
      iv. Click **OK** to finish the configuration and exit the **Proxy Setup...** dialog box.
4. Optional: If your HTTP or HTTPS URL refers to a repository mirror, select the required option from the **URL type** drop-down list. All environments and additional software packages are available for selection when you finish configuring the sources.

5. Click **Done** to apply the settings and return to the **Installation Summary** window.

17.13. OPTIONAL: CONFIGURING KDUMP KERNEL CRASH-DUMPING MECHANISM

Kdump is a kernel crash-dumping mechanism. In the event of a system crash, **Kdump** captures the contents of the system memory at the moment of failure. The captured memory can be analyzed to find the cause of the crash. If **Kdump** is enabled, it must have a small portion of the system's memory (RAM) reserved to itself. This reserved memory is not accessible to the main kernel.

Procedure

1. From the **Installation Summary** window, click **Kdump**. The **Kdump** window opens.
2. Select the **Enable kdump** check box.
3. Select either the **Automatic** or **Manual** memory reservation setting.
4. If you select **Manual**, enter the amount of memory (in megabytes) that you want to reserve in the **Memory to be reserved**field using the **+** and **-** buttons. The **Usable System Memory** readout below the reservation input field shows how much memory is accessible to your main system after reserving the amount of RAM that you select.
5. Click **Done** to apply the settings and return to graphical installations.
   The amount of memory that you reserve is determined by your system architecture (AMD64 and Intel 64 have different requirements than IBM Power) as well as the total amount of system memory. In most cases, automatic reservation is satisfactory.

   Additional settings, such as the location where kernel crash dumps will be saved, can only be configured after the installation using either the **system-config-kdump** graphical interface, or manually in the **/etc/kdump.conf** configuration file.

CHAPTER 18. REGISTERING YOUR RHEL SYSTEM

After installing RHEL, register your system to receive updates and to access Red Hat services. You can register your system by using various methods:

- RHC client
- Graphical user interface
- Subscription manager
- Registration assistant

The RHC client provides the following flexible features to register your RHEL system and manage the extent of data collection:

- Access to Red Hat content: Provides access to Red Hat CDN repositories.
- Analytics for Red Hat Lightspeed: Enables data collection for Red Hat Lightspeed.
- Remote management: Establishes an additional MQTT network connection to Red Hat services for remote execution of certain actions from console.redhat.com.

By default, the RHC client enables all available features. However, you can override this behavior by enabling or disabling specific features based on your connectivity requirements. Review the available options and select the appropriate features based on your internal policies on data and network connectivity.

> **NOTE**
>
> Use the RHC client only for systems directly connecting to Hybrid Cloud Console, the RHC client currently does not support Satellite. For more information on registering a system to a Red Hat Satellite or Capsule, see Red Hat Satellite Documentation .

18.1. REGISTERING A SYSTEM BY USING THE RHC CLIENT

Register your RHEL system by using the RHC client to connect to Red Hat services with the default feature levels. This enables all available features: access to Red Hat content, analytics for Red Hat Lightspeed, and remote management.

Prerequisites

- You have an activation key and an organization ID to register your system.

Procedure

1. Open the terminal window.
2. Do one of the following:
    a. To register the system with the default feature level and ensure system executes the **Remediation** and **Tasks** from Red Hat Lightspeed:

```
# rhc connect --activation-key=<activation_key> --organization=<organization_ID>
# dnf install -y rhc-worker-playbook
```

b. To disable any of the features while registering your system:

```
# rhc connect --activation-key=<activation_key> --organization=<organization_ID>
--disable-feature <feature>
```

Where *feature* can be replaced with:

- content - Provides access to Red Hat CDN repositories.
- analytics - Enables data collection for Red Hat Lightspeed.
- remote-management - Establishes an additional MQTT network connection to Red Hat services for remote execution of certain actions from console.redhat.com.
  For example, if you want to register your system with a remote management feature disabled so that your system cannot be managed remotely but the system has access to RHEL content and collects data for Red Hat Lightspeed analytics, enter:

  ```
  # rhc connect --activation-key=<activation_key> --
  organization=<organization_ID> --disable-feature remote-management
  ```

Verification

- Confirm the active features when registering with default options:

  ```
  # rhc status
  Connection status:
  ✓ Connected to Red Hat Subscription Management
  ✓ Connected to Red Hat Lightspeed
  ✓ The yggdrasil service is active
  Manage your connected systems: https://red.ht/connector
  ```

- Confirm the active features with remote-management disabled:

  ```
  # rhc status
  Connection status:
  ✓ Connected to Red Hat Subscription Management
  ✓ Connected to Red Hat Lightspeed
  ● The yggdrasil service is inactive
  ```

Additional resources

- Remote Host Configuration and Management

18.2. REGISTERING YOUR SYSTEM USING THE SUBSCRIPTION MANAGER

You can register your Red Hat Enterprise Linux subscription by using the command line to get software updates and access Red Hat services.

TIP

For an improved and simplified experience registering your hosts to Red Hat, use the RHC client. The RHC client registers your system to Red Hat making your system ready for Red Hat Lightspeed data collection and enabling direct issue remediation from Red Hat Lightspeed for Red Hat Enterprise Linux. For more information, see information, see RHC registration.

Prerequisites

- You have an active, non-evaluation Red Hat Enterprise Linux subscription.
- Your Red Hat subscription status is verified.
- You have successfully installed Red Hat Enterprise Linux and logged into the system as root.

Procedure

1. Open a terminal window as a root user.
2. Register your Red Hat Enterprise Linux system by using the activation key:

   ```
   # subscription-manager register --activationkey=<activation_key_name> --
   org=<organization_ID>
   ```

   When the system is successfully registered, an output similar to the following is displayed:

   ```
   The system has been registered with ID:
   62edc0f8-855b-4184-b1b8-72a9dc793b96
   ```

Additional resources

- Using an activation key to register a system with Red Hat Subscription Manager
- Getting Started with RHEL System Registration

18.3. REGISTERING RHEL 10 USING THE INSTALLER GUI

You can register a Red Hat Enterprise Linux by using the RHEL installer GUI to get software updates and access Red Hat services.

Prerequisites

- You have a valid user account on the Red Hat Customer Portal. See the Create a Red Hat Login page.
- You have a valid Activation Key and Organization ID.

Procedure

1. From the **Installation Summary** screen, under **Software**, click **Connect to Red Hat**
2. Authenticate your Red Hat account using the **Account** or **Activation Key** option.

3. Optional: In the **Set System Purpose** field select the **Role**, **SLA**, and **Usage** attribute that you want to set from the drop-down menu.
   At this point, your Red Hat Enterprise Linux system has been successfully registered.

18.4. REGISTRATION ASSISTANT

Registration Assistant is designed to help you choose the most suitable registration option for your Red Hat Enterprise Linux environment:

- For assistance with using a username and password to register RHEL with the Subscription Manager client, see the RHEL registration assistant on the Customer Portal.
- For assistance with registering your RHEL system to Red Hat Lightspeed, see the Red Hat Lightspeed registration assistant on the Hybrid Cloud Console.

18.5. ADDITIONAL RESOURCES

- Using an activation key to register a system with Red Hat Subscription Manager
- Getting Started with RHEL System Registration

CHAPTER 19. CONFIGURING SYSTEM PURPOSE USING THE SUBSCRIPTION-MANAGER COMMAND-LINE TOOL

System purpose is a feature of the Red Hat Enterprise Linux installation to help RHEL customers get the benefit of our subscription experience and services offered in the Red Hat Hybrid Cloud Console, a dashboard-based, Software-as-a-Service (SaaS) application that enables you to view subscription usage in your Red Hat account.

You can configure system purpose attributes either on the activation keys or by using the subscription manager tool.

Prerequisites

- You have installed and registered your Red Hat Enterprise Linux 10 system, but system purpose is not configured.

- You are logged in as a **root** user.

Procedure

1. From a terminal window, run the following command to set the intended role of the system:

   ```
   # subscription-manager syspurpose role --set VALUE
   ```

   Replace **VALUE** with the role that you want to assign:

   - **Red Hat Enterprise Linux Server**

   - **Red Hat Enterprise Linux Workstation**

   - **Red Hat Enterprise Linux Compute Node**

   For example:

   ```
   # subscription-manager syspurpose role --set Red Hat Enterprise Linux Server
   ```

   a. Optional: Before setting a value, see the available roles supported by the subscriptions for your organization:

      ```
      # subscription-manager syspurpose role --list
      ```

   b. Optional: Run the following command to unset the role:

      ```
      # subscription-manager syspurpose role --unset
      ```

2. Run the following command to set the intended Service Level Agreement (SLA) of the system:

   ```
   # subscription-manager syspurpose service-level --set VALUE
   ```

   Replace **VALUE** with the SLA that you want to assign:

   - **Premium**

   - **Standard**
   - **Self-Support**

   For example:

   ```
   # subscription-manager syspurpose service-level --set "Standard"
   ```

   a. Optional: Before setting a value, see the available service-levels supported by the subscriptions for your organization:

      ```
      # subscription-manager syspurpose service-level --list
      ```

   b. Optional: Run the following command to unset the SLA:

      ```
      # subscription-manager syspurpose service-level --unset
      ```

3. Run the following command to set the intended usage of the system:

   ```
   # subscription-manager syspurpose usage --set "VALUE"
   ```

   Replace **VALUE** with the usage that you want to assign:

   - **Production**
   - **Disaster Recovery**
   - **Development/Test**

   For example:

   ```
   # subscription-manager syspurpose usage --set "Production"
   ```

   a. Optional: Before setting a value, see the available usages supported by the subscriptions for your organization:

      ```
      # subscription-manager syspurpose usage --list
      ```

   b. Optional: Run the following command to unset the usage:

      ```
      # subscription-manager syspurpose usage --unset
      ```

4. Run the following command to show the current system purpose properties:

   ```
   # subscription-manager syspurpose --show
   ```

   a. Optional: For more detailed syntax information run the following command to access the **subscription-manager** man page and browse to the SYSPURPOSE OPTIONS:

      ```
      # man subscription-manager
      ```

Verification

- To verify the system’s subscription status:

  ```
  # subscription-manager status
  +-------------------------------------------+
     System Status Details
  +-------------------------------------------+
  Overall Status: Registered
  ```

  For more information on the subscriptions services, see the Getting Started with the Subscriptions Service guide.

CHAPTER 20. POST-INSTALLATION SECURITY HARDENING

RHEL is designed with robust security features enabled by default. However, you can enhance its security further through additional hardening measures.

For more information about:

- Processes and practices for securing RHEL servers and workstations against local and remote intrusion, exploitation, and malicious activity, see Security hardening.
- Control how users and processes interact with the files on the system or control which users can perform which actions by mapping them to specific SELinux confined users, see Using SELinux.
- Tools and techniques to improve the security of your networks and lower the risks of data breaches and intrusions, see Securing networks.
- Packet filters, such as firewalls, that use rules to control incoming, outgoing, and forwarded network traffic, see Configuring firewalls and packet filters.

CHAPTER 21. CHANGING A SUBSCRIPTION SERVICE

To manage the subscriptions, you can register a RHEL system with either Red Hat Subscription Management Server or Red Hat Satellite Server. If required, you can change the subscription service at a later point. To change the subscription service under which you are registered, unregister the system from the current service and then register it with a new service.

To receive the system updates, register your system with either of the management servers.

This section contains information about how to unregister your RHEL system from the Red Hat Subscription Management Server and Red Hat Satellite Server.

21.1. PREREQUISITES

You have registered your system with any one of the following:

- Red Hat Subscription Management Server
- Red Hat Satellite Server version 6.17

To receive the system updates, register your system with either of the management servers.

21.2. UNREGISTERING FROM SUBSCRIPTION MANAGEMENT SERVER

You can unregister a RHEL system from Red Hat Subscription Management Server by using a command line and the Subscription Manager user interface.

21.2.1. Unregistering using command line

Use the **unregister** command to unregister a RHEL system from Red Hat Subscription Management Server.

Procedure

- Run the unregister command as a root user or **sudo**, without any additional parameters.

  ```
  # subscription-manager unregister
  ```

  The system is unregistered from the Subscription Management Server, and the status 'System Not Registered' is displayed with the **Register System..** button enabled.

  To continue uninterrupted services, re-register the system with either of the management services. If you do not register the system with a management service, you may fail to receive the system updates.

Additional resources

- Using and Configuring Red Hat Subscription Manager
- Registering your system using the command line

21.2.2. Unregistering by using Subscription Manager user interface

You can unregister a RHEL system from Red Hat Subscription Management Server by using the Subscription Manager user interface.

Procedure

1. Log in to your system.
2. From the top left-hand side of the window, click **Activities**.
3. From the menu options, click the **Show Applications** icon.
4. Click the **Settings** icon, or enter **Settings Manager** in the search.
5. In the left side column, select the **System** options, then on the new screen select **Registration**. The Registration screen displays the current status of your Subscription.
6. Click the **Remove Registration…** button and confirm your choice.
7. Enter your administrator password in the **Authentication Required** dialog box. The system is unregistered from the **Subscription Management Server**, and the status **System Not Registered** is displayed with the **Register System…** button enabled.
   To continue uninterrupted services, re-register the system with either of the management services. If you do not register the system with a management service, you may fail to receive the system updates.

Additional resources

- Registering your system using the Subscription Manager User Interface
- Using and Configuring Red Hat Subscription Manager

21.3. UNREGISTERING FROM SATELLITE SERVER

Removing a Red Hat Enterprise Linux (RHEL) system from Satellite Server updates the system’s status on the server side. However, the system’s local registration remains unchanged. To ensure a complete transition to Red Hat Subscription Management (RHSM), manually unregister the system and restore the default configuration in the **/etc/rhsm/rhsm.conf** file.

Procedure

1. Remove the host from the Satellite server .
2. Unregister the system:

   ```
   # subscription-manager unregister
   ```

3. Revert the **/etc/rhsm/rhsm.conf** configuration file to its default values:

   ```
   # cd /etc/rhsm/
   # mv rhsm.conf.bak rhsm.conf
   ```

NOTE

You can restore this file only if the original content of the file was backed up earlier. Otherwise, you must manually restore the file from a non-modified RHEL machine.

4. Verify that the **/etc/rhsm/rhsm.conf** file has been reverted to default values:

```
# subscription-manager config --list
[server]
   hostname = [subscription.rhsm.redhat.com]
   insecure = [0]
   no_proxy = []
   port = [443]
   prefix = [/subscription]
   proxy_hostname = []
   proxy_password = []
   proxy_port = []
   proxy_scheme = [http]
   proxy_user = []
   server_timeout = [180]
   ssl_verify_depth = [3]

[rhsm]
   auto_enable_yum_plugins = [1]
   baseurl = [https://cdn.redhat.com]
   ca_cert_dir = [/etc/rhsm/ca/]
   consumercertdir = [/etc/pki/consumer]
   entitlementcertdir = [/etc/pki/entitlement]
   full_refresh_on_yum = [0]
   inotify = [1]
   manage_repos = [1]
   package_profile_on_trans = [0]
   pluginconfdir = [/etc/rhsm/pluginconf.d]
   plugindir = [/usr/share/rhsm-plugins]
   productcertdir = [/etc/pki/product]
   repo_ca_cert = /etc/rhsm/ca/redhat-uep.pem
   repomd_gpg_url = []
   report_package_profile = [1]

[rhsmcertd]
   auto_registration = [0]
   auto_registration_interval = [60]
   autoattachinterval = [1440]
   certcheckinterval = [240]
   disable = [0]
   splay = [1]

[logging]
   default_log_level = [INFO]

[] - Default value in use
```

5. Register the system:

```
# subscription-manager register
Username: <_redhat_portal_admin_username_>
Password:
```

6. Enable the required repositories:

```
# subscription-manager repos --disable "*"
# subscription-manager repos
# subscription-manager repos --enable=<repo-id>
```

CHAPTER 22. CHANGING BASIC ENVIRONMENT SETTINGS

If your requirements change after installation, you can update the basic system settings accordingly. This includes the following settings:

- Keyboard layout
- Date and time
- System locale

22.1. CONFIGURING THE KEYBOARD LAYOUT

The keyboard layout settings control the layout used on the text console. If RHEL was installed with a different keyboard layout than the one you use after the installation, you can change the layout.

Procedure

1. To list available keymaps:

   ```
   $ localectl list-keymaps
   ANSI-dvorak
   al
   al-plisi
   amiga-de
   de
   de-nodeadkeys
   ...
   ```

2. Optional: Display the current keymap settings:

   ```
   $ localectl status
   ...
   VC Keymap: us
   ...
   ```

3. Change the system keymap, for example:

   ```
   # localectl set-keymap de-nodeadkeys
   ```

   For more information, see the **localectl(1)**, **locale(7)**, and **locale.conf(5)** man pages on your system.

22.2. MANUALLY CONFIGURING THE DATE, TIME AND TIMEZONE SETTINGS

You can manually configure date, time, and timezone settings on RHEL systems to ensure accurate timekeeping across your environment. Proper time configuration is essential for authentication, logging, troubleshooting, and maintaining consistency with other systems in your infrastructure.

Note, prefer time synchronization by using chrony.

Procedure

1. Optional: List the timezones:

```
# timedatectl list-timezones

  Europe/Berlin
```

2. Set the time zone:

```
# timedatectl set-timezone <time_zone>
```

3. Set the date and time:

```
# timedatectl set-time <YYYY-mm-dd HH:MM-SS>
```

Verification

1. Display the date, time, and timezone:

```
# date
Mon May 5 09:11:55 CEST 2025
```

2. To see more details, use the timedatectl command:

```
# timedatectl

 Local time: Mon 2025-05-05 09:11:55 CEST
           Universal time: Mon 2025-05-05 07:11:55 UTC
                 RTC time: Mon 2025-05-05 07:11:55
                Time zone: Europe/Berlin (CEST, +0200)
System clock synchronized: no
              NTP service: inactive
          RTC in local TZ: no
```

For more information, see the **date(1)** and **timedatectl(1)** man pages on your system.

22.3. CONFIGURING THE SYSTEM LOCALE

The system locale specifies the language settings of system services and user interfaces. System-wide locale settings determine the language, region, and formatting conventions used by an operating system and its applications. System-wide locale settings are stored in the **/etc/locale.conf** file that is read at early boot by the **systemd** daemon.

Every service or user inherits the locale settings configured in **/etc/locale.conf**, unless individual programs or individual users override them.

Procedure

1. Optional: Display the current system locales settings:

```
# localectl status
System Locale: LANG=C.UTF-8
 VC Keymap: de-nodeadkeys
   X11 Layout: de
```

2. List available system locale settings:

```
$ localectl list-locales
C.UTF-8
...
en_US.UTF-8
en_ZA.UTF-8
en_ZW.UTF-8
...
```

3. Update the syste locale setting, for example::

```
# localectl set-locale LANG=en_US.UTF-8
```

For more information, see the **localectl(1)**, **locale(7)**, and **locale.conf(5)** man pages on your system.

CHAPTER 23. CONFIGURING A LINUX INSTANCE ON 64-BIT IBM Z

Perform following common tasks for configuring Red Hat Enterprise Linux on 64-bit IBM Z.

23.1. ADDING DASDS TO A Z/VM SYSTEM

Direct Access Storage Devices (DASDs) are a type of storage commonly used with 64-bit IBM Z. For more information, see Working with DASDs in the IBM Knowledge Center. The following example is how to set a DASD online, format it, and make the change persistent.

Verify that the device is attached or linked to the Linux system if running under z/VM.

```
CP ATTACH EB1C TO *
```

To link a mini disk to which you have access, run the following commands:

```
CP LINK RHEL7X 4B2E 4B2E MR
DASD 4B2E LINKED R/W
```

23.2. DYNAMICALLY SETTING DASDS ONLINE

You can set DASDs online dynamically to add or activate storage devices without rebooting.

Procedure

1. Use the **cio_ignore** utility to remove the DASD from the list of ignored devices and make it visible to Linux:

   ```
   # cio_ignore -r device_number
   ```

   Replace *device_number* with the device number of the DASD. For example:

   ```
   # cio_ignore -r 4b2e
   ```

2. Set the device online. Use a command of the following form:

   ```
   # chccwdev -e device_number
   ```

   Replace *device_number* with the device number of the DASD. For example:

   ```
   # chccwdev -e 4b2e
   ```

   For instructions on how to set a DASD online persistently, see Persistently setting DASDs online.

23.3. PREPARING A NEW DASD WITH LOW-LEVEL FORMATTING

Once the disk is online, change back to the **/root** directory and low-level format the device. This is required only once for a DASD during its entire lifetime.

Procedure

1. Change back to the **/root** directory.

```
# cd /root
# dasdfmt -b 4096 -d cdl -p /dev/disk/by-path/ccw-0.0.4b2e
Drive Geometry: 10017 Cylinders * 15 Heads =  150255 Tracks

I am going to format the device /dev/disk/by-path/ccw-0.0.4b2e in the following way:
Device number of device : 0x4b2e
Labelling device        : yes
Disk label              : VOL1
Disk identifier         : 0X4B2E
Extent start (trk no)   : 0
Extent end (trk no)     : 150254
Compatible Disk Layout  : yes
Blocksize               : 4096

--->> ATTENTION! <<---
All data of that device will be lost.
Type "yes" to continue, no will leave the disk untouched: yes
cyl    97 of  3338 |#----------------------------------------------|   2%
```

2. When the progress bar reaches the end and the format is complete, **dasdfmt** prints the following output:

```
Rereading the partition table...
Exiting...
```

3. Use **fdasd** to partition the DASD. You can create up to three partitions on a DASD. In our example here, we create one partition spanning the whole disk:

```
# fdasd -a /dev/disk/by-path/ccw-0.0.4b2e
reading volume label ..: VOL1
reading vtoc ..........: ok

auto-creating one partition for the whole disk...
writing volume label...
writing VTOC...
rereading partition table...
```

After a (low-level formatted) DASD is online, it can be used like any other disk under Linux. For example, you can create file systems, LVM physical volumes, or swap space on its partitions, for example **/dev/disk/by-path/ccw-0.0.4b2e-part1**. Never use the full DASD device (**dev/dasdb**) for anything but the commands **dasdfmt** and **fdasd**. If you want to use the entire DASD, create one partition spanning the entire drive as in the **fdasd** example above.

To add additional disks later without breaking existing disk entries in, for example, **/etc/fstab**, use the persistent device symbolic links under **/dev/disk/by-path/**.

23.4. PERSISTENTLY SETTING DASDS ONLINE

The above instructions described how to activate DASDs dynamically in a running system. However, such changes are not persistent and do not survive a reboot. Making changes to the DASD configuration persistent in your Linux system depends on whether the DASDs belong to the root file system. Those DASDs required for the root file system need to be activated very early during the boot process by the **initramfs** to be able to mount the root file system.

The **cio_ignore** commands are handled seamlessly for persistent device configurations and you do not need to free devices from the ignore list manually.

23.5. DASDS THAT ARE PART OF THE ROOT FILE SYSTEM

The file you have to modify to add DASDs that are part of the root file system has changed in Red Hat Enterprise Linux 8. Instead of editing the **/etc/zipl.conf** file, the new file to be edited, and its location, may be found by running the following commands:

```
# machine_id=$(cat /etc/machine-id)
# kernel_version=$(uname -r)
# ls /boot/loader/entries/$machine_id-$kernel_version.conf
```

There is one boot option to activate DASDs early in the boot process: **rd.dasd=**. This option takes a Direct Access Storage Device (DASD) adapter device bus identifier. For multiple DASDs, specify the parameter multiple times, or use a comma separated list of bus IDs. To specify a range of DASDs, specify the first and the last bus ID. Below is an example of the **/boot/loader/entries/4ab74e52867b4f998e73e06cf23fd761-4.18.0-80.el8.s390x.conf** file for a system that uses physical volumes on partitions of two DASDs for an LVM volume group **vg_devel1** that contains a logical volume **lv_root** for the root file system.

```
title Red Hat Enterprise Linux (4.18.0-80.el8.s390x) 8.0 (Ootpa)
version 4.18.0-80.el8.s390x
linux /boot/vmlinuz-4.18.0-80.el8.s390x
initrd /boot/initramfs-4.18.0-80.el8.s390x.img
options root=/dev/mapper/vg_devel1-lv_root crashkernel=auto rd.dasd=0.0.0200 rd.dasd=0.0.0207
rd.lvm.lv=vg_devel1/lv_root rd.lvm.lv=vg_devel1/lv_swap cio_ignore=all,!condev
rd.znet=qeth,0.0.0a00,0.0.0a01,0.0.0a02,layer2=1,portno=0
id rhel-20181027190514-4.18.0-80.el8.s390x
grub_users $grub_users
grub_arg --unrestricted
grub_class kernel
```

To add another physical volume on a partition of a third DASD with device bus ID **0.0.202b**. To do this, add **rd.dasd=0.0.202b** to the parameters line of your boot kernel in **/boot/loader/entries/4ab74e52867b4f998e73e06cf23fd761-4.18.0-32.el8.s390x.conf**:

```
title Red Hat Enterprise Linux (4.18.0-80.el8.s390x) 8.0 (Ootpa)
version 4.18.0-80.el8.s390x
linux /boot/vmlinuz-4.18.0-80.el8.s390x
initrd /boot/initramfs-4.18.0-80.el8.s390x.img
options root=/dev/mapper/vg_devel1-lv_root crashkernel=auto rd.dasd=0.0.0200 rd.dasd=0.0.0207
rd.dasd=0.0.202b rd.lvm.lv=vg_devel1/lv_root rd.lvm.lv=vg_devel1/lv_swap cio_ignore=all,!condev
rd.znet=qeth,0.0.0a00,0.0.0a01,0.0.0a02,layer2=1,portno=0
id rhel-20181027190514-4.18.0-80.el8.s390x
grub_users $grub_users
grub_arg --unrestricted
grub_class kernel
```

Run **zipl** to apply the changes of the configuration file for the next IPL:

```
# zipl -V
Using config file '/etc/zipl.conf'
Using BLS config file '/boot/loader/entries/4ab74e52867b4f998e73e06cf23fd761-4.18.0-
80.el8.s390x.conf'
Target device information
  Device..........................: 5e:00
  Partition.......................: 5e:01
  Device name.....................: dasda
  Device driver name..............: dasd
  DASD device number..............: 0201
  Type............................: disk partition
  Disk layout.....................: ECKD/compatible disk layout
  Geometry - heads................: 15
  Geometry - sectors..............: 12
  Geometry - cylinders............: 13356
  Geometry - start................: 24
  File system block size..........: 4096
  Physical block size.............: 4096
  Device size in physical blocks..: 262152
Building bootmap in '/boot'
Building menu 'zipl-automatic-menu'
Adding #1: IPL section '4.18.0-80.el8.s390x' (default)
  initial ramdisk...: /boot/initramfs-4.18.0-80.el8.s390x.img
  kernel image......: /boot/vmlinuz-4.18.0-80.el8.s390x
  kernel parmline...: 'root=/dev/mapper/vg_devel1-lv_root crashkernel=auto rd.dasd=0.0.0200
rd.dasd=0.0.0207 rd.dasd=0.0.202b rd.lvm.lv=vg_devel1/lv_root rd.lvm.lv=vg_devel1/lv_swap
cio_ignore=all,!condev rd.znet=qeth,0.0.0a00,0.0.0a01,0.0.0a02,layer2=1,portno=0'
  component address:
    kernel image....: 0x00010000-0x0049afff
    parmline........: 0x0049b000-0x0049bfff
    initial ramdisk.: 0x004a0000-0x01a26fff
    internal loader.: 0x0000a000-0x0000cfff
Preparing boot menu
  Interactive prompt......: enabled
  Menu timeout............: 5 seconds
  Default configuration...: '4.18.0-80.el8.s390x'
Preparing boot device: dasda (0201).
Syncing disks...
Done.
```

23.6. DASDS THAT ARE NOT PART OF THE ROOT FILE SYSTEM

Direct Access Storage Devices (DASDs) that are not part of the root file system, that is, *data disks*, are persistently configured in the **/etc/dasd.conf** file. This file contains one DASD per line, where each line begins with the DASD's bus ID.

When adding a DASD to the **/etc/dasd.conf** file, use key-value pairs to specify the options for each entry. Separate the key and its value with an equal (=) sign. When adding multiple options, use a space or a tab to separate each option. For example, the **/etc/dasd.conf** file contains:

```
0.0.0207
0.0.0200 use_diag=1 readonly=1
```

Changes to the **/etc/dasd.conf** file take effect after a system reboot or after a new DASD is dynamically added by changing the system's I/O configuration (that is, the DASD is attached under z/VM).

Alternatively, to activate a DASD that you have added to the **/etc/dasd.conf** file, complete the following steps:

1. Remove the DASD from the list of ignored devices and make it visible using the **cio_ignore** utility:

   ```
   # cio_ignore -r device_number
   ```

   where ***device_number*** is the DASD device number.

   For example, if the device number is **021a**, run:

   ```
   # cio_ignore -r 021a
   ```

2. Activate the DASD by writing to the device's **uevent** attribute:

   ```
   # echo add > /sys/bus/ccw/devices/dasd-bus-ID/uevent
   ```

   where ***dasd-bus-ID*** is the DASD's bus ID.

   For example, if the bus ID is **0.0.021a**, run:

   ```
   # echo add > /sys/bus/ccw/devices/0.0.021a/uevent
   ```

23.7. FCP LUNS THAT ARE PART OF THE ROOT FILE SYSTEM

The only file you have to modify for adding FCP LUNs that are part of the root file system has changed in Red Hat Enterprise Linux 10. Instead of editing the **/etc/zipl.conf** file, the new file to be edited, and its location, may be found by running the following commands:

```
# machine_id=$(cat /etc/machine-id)
# kernel_version=$(uname -r)
# ls /boot/loader/entries/$machine_id-$kernel_version.conf
```

Red Hat Enterprise Linux provides a parameter to activate FCP LUNs early in the boot process: **rd.zfcp=**. The value is a comma-separated list containing the FCP device bus ID, the target WWPN as 16 digit hexadecimal number prefixed with **0x**, and the FCP LUN prefixed with 0x and padded with zeroes to the right to have 16 hexadecimal digits.

The WWPN and FCP LUN values are only necessary if the **zFCP** device is not configured in NPIV mode, when auto LUN scanning is disabled by the **zfcp.allow_lun_scan=0** kernel module parameter or when installing RHEL-9.0 or older releases. Otherwise they can be omitted, for example, **rd.zfcp=0.0.4000**. Below is an example of the **/boot/loader/entries/4ab74e52867b4f998e73e06cf23fd761-5.14.0-55.el9.s390x.conf** file for a system that uses a physical volume on a partition of an FCP-attached SCSI disk, with two paths, for an LVM volume group **vg_devel1** that contains a logical volume **lv_root** for the root file system.

```
title Red Hat Enterprise Linux (5.14.0-55.el9.s390x) 9.0 (Plow)
version 5.14.0-55.el9.s390x
```

```
linux /boot/vmlinuz-5.14.0-55.el9.s390x
initrd /boot/initramfs-5.14.0-55.el9.s390x.img
options root=/dev/mapper/vg_devel1-lv_root crashkernel=auto
rd.zfcp=0.0.fc00,0x5105074308c212e9,0x401040a000000000
rd.zfcp=0.0.fcd0,0x5105074308c2aee9,0x401040a000000000 rd.lvm.lv=vg_devel1/lv_root
rd.lvm.lv=vg_devel1/lv_swap cio_ignore=all,!condev
rd.znet=qeth,0.0.0a00,0.0.0a01,0.0.0a02,layer2=1,portno=0
id rhel-20181027190514-5.14.0-55.el9.s390x
grub_users $grub_users
grub_arg --unrestricted
grub_class kernel
```

To add another physical volume on a partition of a second FCP-attached SCSI disk with FCP LUN **0x401040a300000000** using the same two paths as the already existing physical volume, add **rd.zfcp=0.0.fc00,0x5105074308c212e9,0x401040a300000000** and **rd.zfcp=0.0.fcd0,0x5105074308c2aee9,0x401040a300000000** to the parameters line of your boot kernel in **/boot/loader/entries/4ab74e52867b4f998e73e06cf23fd761-5.14.0-55.el9.s390x.conf**. For example:

```
title Red Hat Enterprise Linux (5.14.0-55.el9.s390x) 9.0 (Plow)
version 5.14.0-55.el9.s390x
linux /boot/vmlinuz-5.14.0-55.el9.s390x
initrd /boot/initramfs-5.14.0-55.el9.s390x.img
options root=/dev/mapper/vg_devel1-lv_root crashkernel=auto
rd.zfcp=0.0.fc00,0x5105074308c212e9,0x401040a000000000
rd.zfcp=0.0.fcd0,0x5105074308c2aee9,0x401040a000000000
rd.zfcp=0.0.fc00,0x5105074308c212e9,0x401040a300000000
rd.zfcp=0.0.fcd0,0x5105074308c2aee9,0x401040a300000000 rd.lvm.lv=vg_devel1/lv_root
rd.lvm.lv=vg_devel1/lv_swap cio_ignore=all,!condev
rd.znet=qeth,0.0.0a00,0.0.0a01,0.0.0a02,layer2=1,portno=0
id rhel-20181027190514-5.14.0-55.el9.s390x
grub_users $grub_users
grub_arg --unrestricted
grub_class kernel
```

WARNING

Make sure the length of the kernel command line in the configuration file does not exceed 896 bytes. Otherwise, the boot loader cannot be saved, and the installation fails.

- Run **dracut -f** to update the initial RAM disk of your target kernel.
- Run **zipl** to apply the changes of the configuration file for the next IPL:

```
# zipl -V
Using config file '/etc/zipl.conf'
Using BLS config file '/boot/loader/entries/4ab74e52867b4f998e73e06cf23fd761-5.14.0-
55.el9.s390x.conf'
Run /lib/s390-tools/zipl_helper.device-mapper /boot
```

```
Target device information
Device..........................: fd:00
Partition.......................: fd:01
Device name.....................: dm-0
Device driver name..............: device-mapper
Type............................: disk partition
Disk layout.....................: SCSI disk layout
Geometry - start................: 2048
File system block size..........: 4096
Physical block size.............: 512
Device size in physical blocks..: 10074112
Building bootmap in '/boot/'
Building menu 'zipl-automatic-menu'
Adding #1: IPL section '5.14.0-55.el9.s390x' (default)
kernel image......: /boot/vmlinuz-5.14.0-55.el9.s390x
kernel parmline...: 'root=/dev/mapper/vg_devel1-lv_root crashkernel=auto
rd.zfcp=0.0.fc00,0x5105074308c212e9,0x401040a000000000
rd.zfcp=0.0.fcd0,0x5105074308c2aee9,0x401040a000000000
rd.zfcp=0.0.fc00,0x5105074308c212e9,0x401040a300000000
rd.zfcp=0.0.fcd0,0x5105074308c2aee9,0x401040a300000000 rd.lvm.lv=vg_devel1/lv_root
rd.lvm.lv=vg_devel1/lv_swap cio_ignore=all,!condev
rd.znet=qeth,0.0.0a00,0.0.0a01,0.0.0a02,layer2=1,portno=0'
initial ramdisk...: /boot/initramfs-5.14.0-55.el9.s390x.img component address:
kernel image....: 0x00010000-0x007a21ff
parmline........: 0x00001000-0x000011ff
initial ramdisk.: 0x02000000-0x028f63ff
internal loader.: 0x0000a000-0x0000a3ff
Preparing boot device: dm-0.
Detected SCSI PCBIOS disk layout.
Writing SCSI master boot record.
Syncing disks...
Done.
```

WARNING

IBM Z systems use **zipl** to create a bootmap containing raw block addresses of the kernel and **initramfs**. If the **initramfs** is regenerated, its physical layout on disk might change. Because **dracut** does not automatically update the bootmap, manually run **zipl** after regenerating the **initramfs**. Failing to do so may result in a stale bootloader reference and cause the system to fail to boot.

Additional resources

- Kernel panic on s390x after manual **initramfs** regeneration without zipl execution (Red Hat Knowledgebase)

23.8. FCP LUNS THAT ARE NOT PART OF THE ROOT FILE SYSTEM

FCP LUNs that are not part of the root file system, such as data disks, are persistently configured in the file **/etc/zfcp.conf**. It contains one FCP LUN per line. Each line contains the device bus ID of the FCP adapter, the target WWPN as 16 digit hexadecimal number prefixed with **0x**, and the FCP LUN prefixed with **0x** and padded with zeroes to the right to have 16 hexadecimal digits, separated by a space or tab.

The WWPN and FCP LUN values are only necessary if the **zFCP** device is not configured in NPIV mode, when **auto LUN** scanning is disabled by the **zfcp.allow_lun_scan=0** kernel module parameter or when installing RHEL-9.0 or older releases. Otherwise they can be omitted and only the device bus ID is mandatory.

Entries in /**etc/zfcp.conf** are activated and configured by udev when an FCP adapter is added to the system. At boot time, all FCP adapters visible to the system are added and trigger **udev**.

Example content of /**etc/zfcp.conf**:

```
0.0.fc00 0x5105074308c212e9 0x401040a000000000
0.0.fc00 0x5105074308c212e9 0x401040a100000000
0.0.fc00 0x5105074308c212e9 0x401040a300000000
0.0.fcd0 0x5105074308c2aee9 0x401040a000000000
0.0.fcd0 0x5105074308c2aee9 0x401040a100000000
0.0.fcd0 0x5105074308c2aee9 0x401040a300000000
0.0.4000
0.0.5000
```

Modifications of /**etc/zfcp.conf** only become effective after a reboot of the system or after the dynamic addition of a new FCP channel by changing the system’s I/O configuration (for example, a channel is attached under z/VM). Alternatively, you can trigger the activation of a new entry in /**etc/zfcp.conf** for an FCP adapter which was previously not active, by executing the following commands:

1. Use the **zfcp_cio_free** utility to remove the FCP adapters from the list of ignored devices and make them visible to Linux:

   ```
   # zfcp_cio_free
   ```

2. To apply the additions from /**etc/zfcp.conf** to the running system, issue:

   ```
   # zfcpconf.sh
   ```

23.9. ADDING A QETH DEVICE

Add qeth network devices on 64-bit IBM Z systems to enable network connectivity by using OSA-Express features, HiperSockets, z/VM guest LAN, and z/VM VSWITCH. This configuration provides network access for IBM Z systems in various virtualization and networking environments.

For more information about the qeth device driver naming scheme, see Customizing boot parameters.

23.10. DYNAMICALLY ADDING A QETH DEVICE

You can dynamically add a **qeth** device to the system to make a new network interface available without rebooting. Adding devices dynamically supports maintaining connectivity and expanding capacity on running systems.

Procedure

1. Determine whether the **qeth** device driver modules are loaded. The following example shows loaded **qeth** modules:

   ```
   # lsmod | grep qeth
   qeth_l3               69632  0
   qeth_l2               49152  1
   qeth                 131072  2 qeth_l3,qeth_l2
   qdio                  65536  3 qeth,qeth_l3,qeth_l2
   ccwgroup              20480  1 qeth
   ```

   If the output of the **lsmod** command shows that the **qeth** modules are not loaded, run the **modprobe** command to load them:

   ```
   # modprobe qeth
   ```

2. Use the **cio_ignore** utility to remove the network channels from the list of ignored devices and make them visible to Linux:

   ```
   # cio_ignore -r read_device_bus_id,write_device_bus_id,data_device_bus_id
   ```

   Replace *read_device_bus_id*,*write_device_bus_id*,*data_device_bus_id* with the three device bus IDs representing a network device. For example, if the *read_device_bus_id* is **0.0.f500**, the *write_device_bus_id* is **0.0.f501**, and the *data_device_bus_id* is **0.0.f502**:

   ```
   # cio_ignore -r 0.0.f500,0.0.f501,0.0.f502
   ```

3. Use the **znetconf** utility to sense and list candidate configurations for network devices:

   ```
   # znetconf -u
   Scanning for network devices...
   Device IDs                 Type    Card Type      CHPID Drv.
   ------------------------------------------------------------
   0.0.f500,0.0.f501,0.0.f502 1731/01 OSA (QDIO)        00 qeth
   0.0.f503,0.0.f504,0.0.f505 1731/01 OSA (QDIO)        01 qeth
   0.0.0400,0.0.0401,0.0.0402 1731/05 HiperSockets      02 qeth
   ```

4. Select the configuration you want to work with and use **znetconf** to apply the configuration and to bring the configured group device online as network device.

   ```
   # znetconf -a f500
   Scanning for network devices...
   Successfully configured device 0.0.f500 (encf500)
   ```

5. Optional: You can also pass arguments that are configured on the group device before it is set online:

   ```
   # znetconf -a f500 -o portname=myname
   Scanning for network devices...
   Successfully configured device 0.0.f500 (encf500)
   ```

   Now you can continue to configure the **encf500** network interface. Alternatively, you can use **sysfs** attributes to set the device online as follows:

6. Create a **qeth** group device:

   ```
   # echo read_device_bus_id,write_device_bus_id,data_device_bus_id >
   /sys/bus/ccwgroup/drivers/qeth/group
   ```

   For example:

   ```
   # echo 0.0.f500,0.0.f501,0.0.f502 > /sys/bus/ccwgroup/drivers/qeth/group
   ```

7. Next, verify that the **qeth** group device was created properly by looking for the read channel:

   ```
   # ls /sys/bus/ccwgroup/drivers/qeth/0.0.f500
   ```

   You can optionally set additional parameters and features, depending on the way you are setting up your system and the features you require, such as:

   - **portno**
   - **layer2**
   - **portname**

8. Bring the device online by writing **1** to the online **sysfs** attribute:

   ```
   # echo 1 > /sys/bus/ccwgroup/drivers/qeth/0.0.f500/online
   ```

9. Then verify the state of the device:

   ```
   # cat /sys/bus/ccwgroup/drivers/qeth/0.0.f500/online
             1
   ```

   A return value of **1** indicates that the device is online, while a return value **0** indicates that the device is offline.

10. Find the interface name that was assigned to the device:

    ```
    # cat /sys/bus/ccwgroup/drivers/qeth/0.0.f500/if_name
    encf500
    ```

    Now you can continue to configure the **encf500** network interface.

    The following command from the **s390utils** package shows the most important settings of your **qeth** device:

    ```
    # lsqeth encf500
    Device name                     : encf500
    -------------------------------------------------
    card_type               : OSD_1000
    cdev0                  : 0.0.f500
    cdev1                  : 0.0.f501
    cdev2                  : 0.0.f502
    chpid                  : 76
    online                 : 1
    portname                : OSAPORT
    ```

```
portno                  : 0
state                   : UP (LAN ONLINE)
priority_queueing       : always queue 0
buffer_count            : 16
layer2                  : 1
isolation               : none
```

23.11. PERSISTENTLY ADDING A QETH DEVICE

To make a new **qeth** device persistent, create a configuration file for the new interface. The network interface configuration files are placed in the **/etc/NetworkManager/system-connections/** directory.

The network configuration files use the naming convention *device*.nmconnection, where *device* is the value found in the interface-name file in the *qeth* group device that was created earlier, for example *enc9a0*. The *cio_ignore* commands are handled seamlessly for persistent device configurations and you do not need to free devices from the ignore list manually.

Procedure

1. Optional: If a configuration file for another device of the same type already exists, copy it to the new name and edit it:

```
# cd /etc/NetworkManager/system-connections/
# cp enc9a0.nmconnection enc600.nmconnection
```

2. Optional: To learn IDs of your network devices, use the *lsqeth* utility:

```
# lsqeth -p
devices                    CHPID interface        cardtype       port chksum prio-q'ing rtr4 rtr6 lay'2
cnt
-------------------------- ----- ---------------- -------------- ---- ------ ---------- ---- ---- ----- -----
0.0.09a0/0.0.09a1/0.0.09a2 x00   enc9a0    Virt.NIC QDIO  0    sw     always_q_2 n/a  n/a  1
64
0.0.0600/0.0.0601/0.0.0602 x00   enc600    Virt.NIC QDIO  0    sw     always_q_2 n/a  n/a  1
64
```

3. Optional: If you do not have a similar device defined, create a new file. Use this example:

```
[connection]
type=ethernet
interface-name=enc600

[ipv4]
address1=10.12.20.136/24,10.12.20.1
dns=10.12.20.53;
method=manual

[ethernet]
mac-address=00:53:00:8f:fa:66
```

Edit the new *enc600.nmconnection* file as follows:

4. Ensure the new connection file is owned by **root:root**:

```
# chown root:root /etc/NetworkManager/system-connections/enc600.nmconnection
```

5. Add more details in this file or modify these parameters based on your connection requirements.

6. Save the file.

7. Reload the connection profile:

   ```
   # nmcli connection reload
   ```

8. To view complete details of the connection newly added, enter:

   ```
   # nmcli connection show enc600
   ```

   Changes to the *enc600.nmconnection* file become effective after either rebooting the system, dynamic addition of new network device channels by changing the system’s I/O configuration (for example, attaching under z/VM), or reloading network connections. Alternatively, you can trigger the activation of *enc600.nmconnection* for network channels, which were previously not active yet, by executing the following commands:

9. Use the **cio_ignore** utility to remove the network channels from the list of ignored devices and make them visible to Linux:

   ```
   # cio_ignore -r read_device_bus_id,write_device_bus_id,data_device_bus_id
   ```

   Replace *read_device_bus_id*, *write_device_bus_id*, *data_device_bus_id* with the three device bus IDs representing a network device. For example, if the *read_device_bus_id* is **0.0.0600**, the *write_device_bus_id* is **0.0.0601**, and the *data_device_bus_id* is **0.0.0602**:

   ```
   #  cio_ignore -r 0.0.0600,0.0.0601,0.0.0602
   ```

10. To trigger the uevent that activates the change, issue:

    ```
    # echo add > /sys/bus/ccw/devices/read-channel/uevent
    ```

    For example:

    ```
    # echo add > /sys/bus/ccw/devices/0.0.0600/uevent
    ```

11. Check the status of the network device:

    ```
    # lsqeth
    ```

12. If the default route information has changed, you must also update the *ipaddress1* parameters in both the **[ipv4]** and **[ipv6]** sections of the **/etc/NetworkManager/system-connections/<profile_name>.nmconnection** file accordingly:

    ```
    [ipv4]
    address1=10.12.20.136/24,10.12.20.1
    [ipv6]
    address1=2001:db8:1::1,2001:db8:1::fffe
    ```

13. Now start the new interface:

```
# nmcli connection up enc600
```

14. Check the status of the interface:

```
# ip addr show enc600
3: enc600:  <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state
UP group default qlen 1000
link/ether 3c:97:0e:51:38:17 brd ff:ff:ff:ff:ff:ff
10.12.20.136/24 brd 10.12.20.1 scope global dynamic enc600
valid_lft 81487sec preferred_lft 81487sec
inet6 1574:12:5:1185:3e97:eff:fe51:3817/64 scope global noprefixroute dynamic
valid_lft 2591994sec preferred_lft 604794sec
inet6 fe45::a455:eff:d078:3847/64 scope link
valid_lft forever preferred_lft forever
```

15. Check the routing for the new interface:

```
# ip route
default via 10.12.20.136 dev enc600 proto dhcp src
```

16. Verify your changes by using the **ping** utility to ping the gateway or another host on the subnet of the new device:

```
# ping -c 1 10.12.20.136
PING 10.12.20.136 (10.12.20.136) 56(84) bytes of data.
64 bytes from 10.12.20.136: icmp_seq=0 ttl=63 time=8.07 ms
```

17. If the default route information has changed, you must also update **/etc/sysconfig/network** accordingly.
For more information, see the **nm-settings-keyfile** man page on your system.

23.12. CONFIGURING AN 64-BIT IBM Z NETWORK DEVICE FOR NETWORK ROOT FILE SYSTEM

To add a network device that is required to access the root file system, you only have to change the boot options. The boot options can be in a parameter file, however, the **/etc/zipl.conf** file no longer contains specifications of the boot records.

Procedure

- Locate the file by using the following commands that needs to be modified:

```
# machine_id=$(cat /etc/machine-id)
# kernel_version=$(uname -r)
# ls /boot/loader/entries/$machine_id-$kernel_version.conf
```

Dracut, the **mkinitrd** successor that provides the functionality in the initramfs that in turn replaces **initrd**, provides a boot parameter to activate network devices on 64-bit IBM Z early in the boot process: **rd.znet=**.

As input, this parameter takes a comma-separated list of the **NETTYPE** (qeth, lcs, ctc), two (lcs, ctc) or three (qeth) device bus IDs, and optional additional parameters consisting of key-value pairs corresponding to network device sysfs attributes. This parameter configures and activates the 64-bit IBM Z network hardware. The configuration of IP addresses and other network specifics works the same as for other platforms. See the **dracut** documentation for more details.

The **cio_ignore** commands for the network channels are handled seamlessly on boot.

Example boot options for a root file system accessed over the network through NFS:

```
root=10.16.105.196:/nfs/nfs_root cio_ignore=all,!condev
rd.znet=qeth,0.0.0a00,0.0.0a01,0.0.0a02,layer2=1,portno=0,portname=OSAPORT
ip=10.16.105.197:10.16.105.196:10.16.111.254:255.255.248.0:nfs-server.subdomain.domain:e
nc9a0:none rd_NO_LUKS rd_NO_LVM rd_NO_MD rd_NO_DM LANG=en_US.UTF-8
SYSFONT=latarcyrheb-sun16 KEYTABLE=us
```

23.13. ADDITIONAL RESOURCES

- Device Drivers, Features, and Commands on RHEL

CHAPTER 24. TOOLS AND TIPS FOR TROUBLESHOOTING AND BUG REPORTING

The troubleshooting information in the following sections might be helpful when diagnosing issues at the start of the installation process. The following sections are for all supported architectures. However, if an issue is for a particular architecture, it is specified at the start of the section.

24.1. DRACUT

Dracut is a tool that manages the **initramfs** image during the Linux operating system boot process. The **dracut** emergency shell is an interactive mode that can be initiated while the **initramfs** image is loaded. You can run basic troubleshooting commands from the **dracut** emergency shell. For more information, see the **Troubleshooting** section of the **dracut** man page on your system.

24.2. USING INSTALLATION LOG FILES

For debugging purposes, the installation program logs installation actions in files that are located in the **/tmp** directory. These log files are listed in the following table.

Table 24.1. Log files generated during the installation

Log file	Contents
/tmp/anaconda.log	General messages.
/tmp/program.log	All external programs run during the installation.
/tmp/storage.log	Extensive storage module information.
/tmp/packaging.log	dnf and rpm package installation messages.
/tmp/dbus.log	Information about the **dbus** session that is used for installation program modules.
/tmp/sensitive-info.log	Configuration information that is not part of other logs and not copied to the installed system.
/tmp/syslog	Hardware-related system messages. This file contains messages from other Anaconda files.

If the installation fails, the messages are consolidated into **/tmp/anaconda-tb-identifier**, where identifier is a random string. After a successful installation, these files are copied to the installed system under the directory **/var/log/anaconda/**. However, if the installation is unsuccessful, or if the **inst.nosave=all** or **inst.nosave=logs** options are used when booting the installation system, these logs only exist in the installation program's RAM disk. This means that the logs are not saved permanently and are lost when the system is powered down. To store them permanently, copy the files to another system on the network or copy them to a mounted storage device such as a USB flash drive.

24.2.1. Creating pre-installation log files

You can set the **inst.debug** option to create log files before the installation process starts. These log files contain, for example, the current storage configuration.

Prerequisites

- The Red Hat Enterprise Linux boot menu is open.

Procedure

1. Select the **Install Red Hat Enterprise Linux**option from the boot menu.
2. Press the **Tab** key on BIOS-based systems or the **e** key on UEFI-based systems to edit the selected boot options.
3. Append **inst.debug** to the options. For example:

```
vmlinuz ... inst.debug
```

4. Press the **Enter** key on your keyboard. The system stores the pre-installation log files in the **/tmp/pre-anaconda-logs/** directory before the installation program starts.
5. Press **Ctrl** + **Alt** + **F2** to switch to the console and access the log files.
6. Change to the **/tmp/pre-anaconda-logs/** directory:

```
# cd /tmp/pre-anaconda-logs/
```

Additional resources

- Boot options reference
- Console logging during installation

24.2.2. Transferring installation log files to a USB drive

You can transfer installation log files to a USB drive to share it further.

Prerequisites

- You have backed up data from the USB drive.
- You are logged into a root account and you have access to the installation program’s temporary file system.

Procedure

1. Press **Ctrl** + **Alt** + **F2** to access a shell prompt on the system you are installing.
2. Connect a USB flash drive to the system and run the **dmesg** command:

```
# dmesg
```

A log detailing all recent events is displayed. At the end of this log, a set of messages is displayed. For example:

```
[ 170.171135] sd 5:0:0:0: [sdb] Attached SCSI removable disk
```

3. Note the name of the connected device. In the above example, it is **sdb**.

4. Navigate to the **/mnt** directory and create a new directory that serves as the mount target for the USB drive. This example uses the name **usb**:

```
# mkdir usb
```

5. Mount the USB flash drive onto the newly created directory. In most cases, you do not want to mount the whole drive, but a partition on it. Do not use the name **sdb**, use the name of the partition you want to write the log files to. In this example, the name **sdb1** is used:

```
# mount /dev/sdb1 /mnt/usb
```

6. Verify that you mounted the correct device and partition by accessing it and listing its contents:

```
# cd /mnt/usb
```

```
# ls
```

7. Copy the log files to the mounted device.

```
# cp /tmp/*log /mnt/usb
```

8. Unmount the USB flash drive. If you receive an error message that the target is busy, change your working directory to outside the mount (for example, /).

```
# umount /mnt/usb
```

24.2.3. Transferring installation log files over the network

Transfer RHEL installation log files over the network to analyze installation issues and troubleshoot problems. By using this method, you can collect diagnostic information from systems where local access is limited or when you need to share logs with support teams.

Prerequisites

- You are logged into a root account and you have access to the installation program’s temporary file system.

Procedure

1. Press **Ctrl** + **Alt** + **F2** to access a shell prompt on the system you are installing.

2. Switch to the **/tmp** directory where the log files are located:

```
# cd /tmp
```

3. Copy the log files onto another system on the network using the **scp** command:

   ```
   # scp *log user@address:path
   ```

   a. Replace **user** with a valid user name on the target system, **address** with the target system's address or host name, and **path** with the path to the directory where you want to save the log files. For example, if you want to log in as **john** on a system with an IP address of 192.168.0.122 and place the log files into the **/home/john/logs/** directory on that system, the command is as follows:

      ```
      # scp *log john@192.168.0.122:/home/john/logs/
      ```

      When connecting to the target system for the first time, the SSH client asks you to confirm that the fingerprint of the remote system is correct and that you want to continue:

      ```
      The authenticity of host '192.168.0.122 (192.168.0.122)' can't be established.
      ECDSA key fingerprint is a4:60:76:eb:b2:d0:aa:23:af:3d:59:5c:de:bb:c4:42.
      Are you sure you want to continue connecting (yes/no)?
      ```

   b. Type **yes** and press **Enter** to continue. Provide a valid password when prompted. The files are transferred to the specified directory on the target system.

24.3. VERIFYING BOOT MEDIA

Verifying ISO images helps to avoid problems that are sometimes encountered during installation. These sources include DVD and ISO images stored on a disk or NFS server. You can test the integrity of an ISO-based installation source before using it to install Red Hat Enterprise Linux.

Prerequisites

- You have accessed the Red Hat Enterprise Linux boot menu.

Procedure

1. From the boot menu, select **Test this media & install Red Hat Enterprise Linux 10**to test the boot media.

2. The boot process tests the media and highlights any issues.

3. Optional: You can start the verification process by appending **rd.live.check** to the boot command line. For more information, see Customizing boot options.

24.4. DISPLAY SETTINGS AND DEVICE DRIVERS

Some video cards have trouble booting into the Red Hat Enterprise Linux graphical installation program. If the installation program does not run using its default settings, it attempts to run in a lower resolution mode. If that fails, the installation program attempts to run in text mode.

There are several possible solutions to resolve display issues, most of which involve specifying custom boot options:

For more information, see Console boot options.

Table 24.2. Solutions

Solution	Description
Use the basic graphics mode	You can attempt to perform the installation using the basic graphics driver. To do this, either select **Troubleshooting > Install Red Hat Enterprise Linux in basic graphics mode** from the boot menu.
Use the text mode	You can attempt to perform the installation using the text mode. For details, refer to Installing RHEL in text mode.
Specify the display resolution manually	If the installation program fails to detect your screen resolution, you can override the automatic detection and specify it manually. To do this, append the **inst.resolution=x** option at the boot menu, where**x** is your display's resolution, for example, 1024x768. For more information, see Customizing boot options.
Use an alternate video driver	You can attempt to specify a custom video driver, overriding the installation program's automatic detection.
Perform the installation using RDP	If the above options fail, you can use a separate system to access the graphical installation over the network, using the Remote Desktop Protocol (RDP).

- If specifying a custom video driver solves your problem, you should report it as a bug in Jira. The installation program should be able to detect your hardware automatically and use the appropriate driver without intervention.

CHAPTER 25. TROUBLESHOOTING AFTER INSTALLATION

The troubleshooting information in the following sections might be helpful when diagnosing issues after the installation process. The following sections are for all supported architectures. However, if an issue is for a particular architecture, it is specified at the start of the section.

25.1. RESUMING AN INTERRUPTED DOWNLOAD ATTEMPT

You can resume interrupted RHEL ISO image downloads using the **curl** command to avoid re-downloading large files from the beginning. It saves time and bandwidth when network interruptions occur during the download process, ensuring efficient acquisition of installation media.

Prerequisites

- You have navigated to the **Product Downloads** section of the Red Hat Customer Portal at https://access.redhat.com/downloads, and selected the required variant, version, and architecture.

- You have right-clicked on the required ISO file, and selected **Copy Link Location** to copy the URL of the ISO image file to your clipboard.

Procedure

1. Download the ISO image from the new link. Add the **--continue-at -** option to automatically resume the download:

```
$ curl --output directory-path/filename.iso 'new_copied_link_location' --continue-at -
```

2. Use a checksum utility such as **sha256sum** to verify the integrity of the image file after the download finishes:

```
$ sha256sum rhel-x.x-x86_64-dvd.iso
  `85a...46c rhel-x.x-x86_64-dvd.iso`
```

Compare the output with reference checksums provided on the Red Hat Enterprise Linux **Product Download** web page.

Resuming an interrupted download attempt

The following is an example of a **curl** command for a partially downloaded ISO image:

```
$ curl --output _rhel-x.x-x86_64-dvd.iso
'https://access.cdn.redhat.com//content/origin/files/sha256/85/85a...46c/rhel-x.x-x86_64-
dvd.iso?_auth=141...963' --continue-at -
```

25.2. DISKS ARE NOT DETECTED

If the installation program cannot find a writable storage device to install to, it returns the following error message in the **Installation Destination** window: **No disks detected. Please shut down the computer, connect at least one disk, and restart to complete installation.**

Check the following items:

- Your system has at least one storage device attached.
- If your system uses a hardware RAID controller; verify that the controller is properly configured and working as expected. See your controller's documentation for instructions.
- If you are installing into one or more iSCSI devices and there is no local storage present on the system, verify that all required LUNs are presented to the appropriate Host Bus Adapter (HBA).

If the error message is still displayed after rebooting the system and starting the installation process, the installation program failed to detect the storage. In many cases the error message is a result of attempting to install on an iSCSI device that is not recognized by the installation program.

In this scenario, you must perform a driver update before starting the installation. Check your hardware vendor's website to determine if a driver update is available.

For more general information about driver updates, see the *Updating drivers during installation* .

You can also consult the Red Hat Hardware Compatibility List, available at https://access.redhat.com/ecosystem/search/#/category/Server.

25.3. CANNOT BOOT WITH A RAID CARD

If you cannot boot your system after the installation, you might need to reinstall and repartition your system's storage. Some BIOS types do not support booting from RAID cards. After you finish the installation and reboot the system for the first time, a text-based screen displays the boot loader prompt (for example, **grub>**) and a flashing cursor might be displayed. If this is the case, you must repartition your system and move your **/boot** partition and the boot loader outside of the RAID array. The **/boot** partition and the boot loader must be on the same drive. Once these changes have been made, you should be able to finish your installation and boot the system properly.

25.4. GRAPHICAL BOOT SEQUENCE IS NOT RESPONDING

When rebooting your system for the first time after installation, the system might be unresponsive during the graphical boot sequence. If this occurs, a reset is required. In this scenario, the boot loader menu is displayed successfully, but selecting any entry and attempting to boot the system results in a halt. This usually indicates that there is a problem with the graphical boot sequence. To resolve the issue, you must disable the graphical boot by temporarily altering the setting at boot time before changing it permanently.

25.4.1. Disabling the graphical boot permanently

You can disable the graphical boot permanantly.

Procedure

1. Log in to the root account on your system.
2. Use the grubby tool to find the default GRUB2 kernel:

   ```
   # grubby --default-kernel
   /boot/vmlinuz-6.12.0-0.el10_0.x86_64
   ```

3. Use the grubby tool to remove the **rhgb** boot option from the default kernel in your GRUB2 configuration. For example:

```
# grubby --remove-args="rhgb" --update-kernel /boot/vmlinuz-6.12.0-0.el10_0.x86_64
```

4. Reboot the system. The graphical boot sequence is no longer used. If you want to enable the graphical boot sequence, follow the same procedure, replacing the **--remove-args="rhgb"** parameter with the **--args="rhgb"** parameter. This restores the **rhgb** boot option to the default kernel in your GRUB2 configuration.

25.4.2. Disabling the graphical boot temporarily

You can disable the graphical boot temporarily.

Procedure

1. Start your system and wait until the boot loader menu is displayed. If you set your boot timeout period to **0**, press the **Esc** key to access it.

2. From the boot loader menu, use your cursor keys to highlight the entry you want to boot. Press the **Tab** key on BIOS-based systems or the **e** key on UEFI-based systems to edit the selected entry options.

3. In the list of options, find the kernel line - that is, the line beginning with the keyword **linux**. On this line, locate and delete **rhgb**.

4. Press **F10** or **Ctrl**+**X** to boot your system with the edited options.
   If the system started successfully, you can log in normally. However, if you do not disable graphical boot permanently, you must perform this procedure every time the system boots.

25.5. SETTING THE AMOUNT OF MEMORY MANUALLY IF RAM IS NOT RECOGNIZED

In some scenarios, the kernel does not recognize all memory (RAM), which causes the system to use less memory than is installed. If the total amount of memory that your system reports does not match your expectations, it is likely that at least one of your memory modules is faulty. On BIOS-based systems, you can use the **Memtest86+** utility to test your system's memory.

Some hardware configurations have part of the system's RAM reserved, and as a result, it is unavailable to the system. Some laptop computers with integrated graphics cards reserve a portion of memory for the GPU. For example, a laptop with 4 GiB of RAM and an integrated Intel graphics card shows roughly 3.7 GiB of available memory. Additionally, the **kdump** crash kernel dumping mechanism, which is enabled by default on most Red Hat Enterprise Linux systems, reserves some memory for the secondary kernel used in case of a primary kernel failure. This reserved memory is not displayed as available.

You can manually set the amount of memory.

Procedure

1. Check the amount of memory that your system currently reports in MiB:

   ```
   $ free -m
   ```

2. Reboot your system and wait until the boot loader menu is displayed.
   If your boot timeout period is set to **0**, press the **Esc** key to access the menu.

3. From the boot loader menu, use your cursor keys to highlight the entry you want to boot, and press the **Tab** key on BIOS-based systems or the **e** key on UEFI-based systems to edit the selected entry options.

4. In the list of options, find the kernel line: that is, the line beginning with the keyword **linux**. Append the following option to the end of this line:

```
mem=xxM
```

5. Replace ***xx*** with the amount of RAM you have in MiB.

6. Press **F10** or **Ctrl**+**X** to boot your system with the edited options.

7. Wait for the system to boot, log in, and open a command line.

8. Check the amount of memory that your system reports in MiB:

```
$ free -m
```

9. If the total amount of RAM displayed by the command now matches your expectations, make the change permanent:

```
# grubby --update-kernel=ALL --args="mem=xxM"
```

25.6. SYSTEM IS DISPLAYING SIGNAL 11 ERRORS

A signal 11 error, commonly known as a segmentation fault, means that a program accessed a memory location that it was not assigned. A signal 11 error can occur due to a bug in one of the software programs that are installed, or faulty hardware. If you receive a signal 11 error during the installation process, verify that you are using the most recent installation images and prompt the installation program to verify them to ensure they are not corrupt.

For more information, see Verifying Boot media.

Faulty installation media (such as an improperly burned or scratched optical disk) are a common cause of signal 11 errors. Verify the integrity of the installation media before every installation. For information about obtaining the most recent installation media, refer to the Product Downloads page.

To perform a media check before the installation starts, append the **rd.live.check** boot option at the boot menu. If you performed a media check without any errors and you still have issues with segmentation faults, it usually indicates that your system encountered a hardware error. In this scenario, the problem is most likely in the system’s memory (RAM). This can be a problem even if you previously used a different operating system on the same computer without any errors.

NOTE

For AMD and Intel 64-bit and 64-bit ARM architectures: On BIOS-based systems, you can use the **Memtest86+** memory testing module included on the installation media to perform a thorough test of your system’s memory.

Other possible causes are beyond this document’s scope. Consult your hardware manufacturer’s documentation and also see the Red Hat Hardware Compatibility List, available online at https://access.redhat.com/ecosystem/search/#/category/Server.

25.7. UNABLE TO IPL FROM NETWORK STORAGE SPACE ON IBM POWER SYSTEMS

If you experience difficulties when trying to IPL from Network Storage Space (NWSSTG), it is most likely due to a missing PReP partition. In this scenario, you must reinstall the system and create this partition during the partitioning phase or in the Kickstart file.

25.8. USING RESCUE MODE

The installation program's rescue mode is a minimal Linux environment that can be booted from the Red Hat Enterprise Linux DVD or other boot media. It contains command-line utilities for repairing a wide variety of issues. Rescue mode can be accessed from the **Troubleshooting** menu of the boot menu. In this mode, you can mount file systems as read-only, blacklist or add a driver provided on a driver disc, install or upgrade system packages, or manage partitions.

> **NOTE**
>
> The installation program's rescue mode is different from rescue mode (an equivalent to single-user mode) and emergency mode, which are provided as parts of the **systemd** system and service manager.

To boot into rescue mode, you must be able to boot the system using one of the Red Hat Enterprise Linux boot media, such as a minimal boot disc or USB drive, or a full installation DVD.

> **IMPORTANT**
>
> Advanced storage, such as iSCSI or zFCP devices, must be configured either using **dracut** boot options such as **rd.zfcp=** or **root=iscsi:** *options*, or in the CMS configuration file on 64-bit IBM Z. It is not possible to configure these storage devices interactively after booting into rescue mode. For information about **dracut** boot options, see the **dracut.cmdline(7)** man page on your system.

25.8.1. Booting into rescue mode

Boot your RHEL system into rescue mode to troubleshoot and repair system issues when the normal boot process fails. Rescue mode provides a minimal environment to diagnose problems, recover data, and fix configuration issues that prevent normal system operation.

Procedure

1. Boot the system from either minimal boot media, or a full installation DVD or USB drive, and wait for the boot menu to be displayed.

2. From the boot menu, either select **Troubleshooting** > **Rescue a Red Hat Enterprise Linux system** option, or append the **inst.rescue** option to the boot command line. To enter the boot command line, press the **Tab** key on BIOS-based systems or the **e** key on UEFI-based systems.

3. Optional: If your system requires a third-party driver provided on a driver disc to boot, append the **inst.dd=driver_name** to the boot command line:

   ```
   inst.rescue inst.dd=driver_name
   ```

4. Optional: If a driver that is part of the Red Hat Enterprise Linux distribution prevents the system from booting, append the **modprobe.blacklist=** option to the boot command line:

```
inst.rescue modprobe.blacklist=driver_name
```

5. Press **Enter** (BIOS-based systems) or **Ctrl**+**X** (UEFI-based systems) to boot the modified option. Wait until the following message is displayed:

```
The rescue environment will now attempt to find your Linux installation and mount it under the
directory: /mnt/sysroot/. You can then make any changes required to your system. Choose 1
to proceed with this step. You can choose to mount your file systems read-only instead of
read-write by choosing 2. If for some reason this process does not work choose 3 to skip
directly to a shell.

1) Continue
2) Read-only mount
3) Skip to shell
4) Quit (Reboot)
```

If you select **1**, the installation program attempts to mount your file system under the directory **/mnt/sysroot/**. You are notified if it fails to mount a partition. If you select **2**, it attempts to mount your file system under the directory **/mnt/sysroot/**, but in read-only mode. If you select **3**, your file system is not mounted.

For the system root, the installer supports two mount points **/mnt/sysimage** and **/mnt/sysroot**. The **/mnt/sysroot** path is used to mount / of the target system. Usually, the physical root and the system root are the same, so **/mnt/sysroot** is attached to the same file system as **/mnt/sysimage**. The only exceptions are rpm-ostree systems, where the system root changes based on the deployment. Then, **/mnt/sysroot** is attached to a subdirectory of **/mnt/sysimage**. Use **/mnt/sysroot** for chroot.

6. Select **1** to continue. Once your system is in rescue mode, a prompt appears on VC (virtual console) 1 and VC 2. Use the **Ctrl+Alt+F1** key combination to access VC 1 and **Ctrl+Alt+F2** to access VC 2:

```
sh-4.2#
```

7. Even if your file system is mounted, the default root partition while in rescue mode is a temporary root partition, not the root partition of the file system used during normal user mode (**multi-user.target** or **graphical.target**). If you selected to mount your file system and it mounted successfully, you can change the root partition of the rescue mode environment to the root partition of your file system by executing the following command:

```
sh-4.2# chroot /mnt/sysroot
```

This is useful if you need to run commands, such as **rpm**, that require your root partition to be mounted as /. To exit the chroot environment, type **exit** to return to the prompt.

8. If you selected **3**, you can still try to mount a partition or LVM2 logical volume manually inside rescue mode by creating a directory, such as **/directory/**, and typing the following command:

```
sh-4.2# mount -t xfs /dev/mapper/VolGroup00-LogVol02 /directory
```

In the above command, /**directory**/ is the directory that you created and /**dev**/**mapper**/**VolGroup00-LogVol02** is the LVM2 logical volume you want to mount. If the partition is a different type than XFS, replace the xfs string with the correct type (such as ext4).

9. If you do not know the names of all physical partitions, use the following command to list them:

   ```
   sh-4.2# fdisk -l
   ```

   If you do not know the names of all LVM2 physical volumes, volume groups, or logical volumes, use the **pvdisplay**, **vgdisplay** or **lvdisplay** commands.

25.8.2. Using an SOS report in rescue mode

The **sos report** command-line utility collects configuration and diagnostic information, such as the kernel version, loaded modules, and system and service configuration files from the system. The utility output is stored in a tar archive in the /**var**/**tmp**/ directory. The **sos report** utility is useful for analyzing system errors and troubleshooting. You can capture an **sos report** output in rescue mode.

Prerequisites

- You have booted into rescue mode.
- You have mounted the installed system / **(root)** partition in read-write mode.
- You have contacted Red Hat Support about your case and received a case number.

Procedure

1. Change the root directory to the /**mnt**/**sysroot**/ directory:

   ```
   sh-4.2# chroot /mnt/sysroot/
   ```

2. Execute **sos report** to generate an archive with system configuration and diagnostic information:

   ```
   sh-4.2# sos report
   ```

   sos report prompts you to enter your name and the case number you received from Red Hat Support. Use only letters and numbers because adding any of the following characters or spaces could render the report unusable: **# % & { } \ < > > * ? / $ ~ ' " : @ + ` | =**

3. Optional: If you want to transfer the generated archive to a new location by using the network, it is necessary to have a network interface configured. In this scenario, use the dynamic IP addressing as no other steps are required. However, when using static addressing, enter the following command to assign an IP address (for example 10.13.153.64/23) to a network interface, for example dev eth0:

   ```
   bash-4.2# ip addr add 10.13.153.64/23 dev eth0
   ```

4. Exit the chroot environment:

   ```
   sh-4.2# exit
   ```

5. Store the generated archive in a new location, from where it can be easily accessible:

```
sh-4.2# cp /mnt/sysroot/var/tmp/sos report new_location
```

6. For transferring the archive through the network, use the **scp** utility:

```
sh-4.2# scp /mnt/sysroot/var/tmp/sos report username@hostname:sos report
```

Additional resources

- What is an sos report and how to create one in Red Hat Enterprise Linux?
- How to generate sos report from the rescue environment
- How do I make sos report write to an alternative location?
- sos report fails. What data should I provide in its place?

25.8.3. Reinstalling the GRUB2 boot loader

In some scenarios, the GRUB2 boot loader is mistakenly deleted, corrupted, or replaced by other operating systems. You can reinstall GRUB2 on the master boot record (MBR) on AMD64 and Intel 64 systems with BIOS.

Prerequisites

- You have booted into rescue mode.
- You have mounted the installed system / **(root)** partition in read-write mode.
- You have mounted the /**boot** mount point in read-write mode.

Procedure

1. Change the root partition:

```
sh-4.2# chroot /mnt/sysroot/
```

2. Reinstall the GRUB2 boot loader, where the **install_device** block device was installed:

```
sh-4.2# /sbin/grub2-install install_device
```

IMPORTANT

Running the **grub2-install** command could lead to the machine being unbootable if all the following conditions apply:

- The system is an AMD64 or Intel 64 with Extensible Firmware Interface (EFI).
- Secure Boot is enabled.

After you run the **grub2-install** command, you cannot boot the AMD64 or Intel 64 systems that have Extensible Firmware Interface (EFI) and Secure Boot enabled. This issue occurs because the **grub2-install** command installs an unsigned GRUB2 image that boots directly instead of using the shim application. When the system boots, the shim application validates the image signature, which when not found fails to boot the system.

3. Reboot the system.

25.8.4. Using dnf to add or remove a driver

Missing or malfunctioning drivers cause problems when booting the system. Rescue mode provides an environment in which you can add or remove a driver even when the system fails to boot. Wherever possible, use the **dnf** package manager to remove malfunctioning drivers or to add updated or missing drivers.

IMPORTANT

When you install a driver from a driver disc, the driver disc updates all **initramfs** images on the system to use this driver. If a problem with a driver prevents a system from booting, you cannot rely on booting the system from another **initramfs** image.

Prerequisites

- You have booted into rescue mode.
- You have mounted the installed system in read-write mode.

Procedure

1. To add a new driver using **dnf**:

   a. Make the RPM package that contains the driver available. For example, mount a CD or USB flash drive and copy the RPM package to a location of your choice under **/mnt/sysroot/**, for example: **/mnt/sysroot/root/drivers/**.

   b. Change the root directory to **/mnt/sysroot/**:

   ```
   sh-4.2# chroot /mnt/sysroot/
   ```

   c. Use the **dnf install** command to install the driver package. For example, run the following command to install the **xorg-x11-drv-wacom** driver package from **/root/drivers/**:

   ```
   sh-4.2# dnf install /root/drivers/xorg-x11-drv-wacom-0.23.0-6.el7.x86_64.rpm
   ```

NOTE

The **/root/drivers/** directory in this chroot environment is the **/mnt/sysroot/root/drivers/** directory in the original rescue environment.

d. Exit the chroot environment:

```
sh-4.2# exit
```

2. To remove a driver by using **dnf**:

   a. Change the root directory to the **/mnt/sysroot/** directory:

   ```
   sh-4.2# chroot /mnt/sysroot/
   ```

   b. Use the **dnf remove** command to remove the driver package. For example, to remove the **xorg-x11-drv-wacom** driver package, run:

   ```
   sh-4.2# dnf remove xorg-x11-drv-wacom
   ```

   c. Exit the chroot environment:

   ```
   sh-4.2# exit
   ```

   If you cannot remove a malfunctioning driver for some reason, you can instead blocklist the driver so that it does not load at boot time.

   d. When you have finished removing drivers, reboot the system.

25.9. IP= BOOT OPTION RETURNS AN ERROR

Using the **ip=** boot option format **ip=[ip address]** for example, **ip=192.168.1.1** returns the error message **Fatal for argument 'ip=[insert ip here]'\n sorry, unknown value [ip address] refusing to continue**.

In previous releases of Red Hat Enterprise Linux, the boot option format was:

```
ip=192.168.1.15 netmask=255.255.255.0 gateway=192.168.1.254 nameserver=192.168.1.250
hostname=myhost1
```

However, in Red Hat Enterprise Linux 10, the boot option format is:

```
ip=192.168.1.15::192.168.1.254:255.255.255.0:myhost1::none: nameserver=192.168.1.250
```

To resolve the issue, use the format: **ip=ip::gateway:netmask:hostname:interface:none** where:

- **ip** specifies the client ip address. You can specify IPv6 addresses in square brackets, for example, **[2001:DB8::1]**.
- **gateway** is the default gateway. IPv6 addresses are also accepted.

- **netmask** is the netmask to be used. This can be either a full netmask, for example, 255.255.255.0, or a prefix, for example, **64**.
- **hostname** is the host name of the client system. This parameter is optional.

Additional resources

- Network boot options

25.10. CANNOT BOOT INTO THE GRAPHICAL INSTALLATION ON ILO OR IDRAC DEVICES

The graphical installer for a remote ISO installation on Integrated Lights-Out (iLO) or Integrated Dell Remote Access Controller (iDRAC) devices may not be available due to a slow internet connection. To proceed with the installation, you can choose one of the following ways:

Procedure

1. Avoid the timeout. To do so:
   a. Press the **Tab** key in case of BIOS usage, or the **e** key in case of UEFI usage when booting from an installation media. That will allow you to modify the kernel command line arguments.
   b. To proceed with the installation, append the **rd.live.ram=1** and press **Enter** in case of BIOS usage, or **Ctrl+x** in case of UEFI usage.
      This might take longer to load the installation program.
2. Another option to extend the loading time for the graphical installer is to set the **inst.xtimeout** kernel argument in seconds.

   ```
   inst.xtimeout=N
   ```

3. You can install the system in text mode. For more details, see Installing RHEL in text mode .
4. In the remote management console, such as iLO or iDRAC, instead of a local media source, use the direct URL to the installation ISO file from the Download center on the Red Hat Customer Portal. You must be logged in to access this section.

25.11. ROOTFS IMAGE IS NOT INITRAMFS

If you get the following message on the console during booting the installer, the transfer of the installer **initrd.img** might have had errors:

```
[ ...] rootfs image is not initramfs
```

To resolve this issue, download **initrd** again or run the **sha256sum** with **initrd.img** and compare it with the checksum stored in the **.treeinfo** file on the installation medium, for example,

```
$ sha256sum dvd/images/pxeboot/initrd.img
fdb1a70321c06e25a1ed6bf3d8779371b768d5972078eb72b2c78c925067b5d8
dvd/images/pxeboot/initrd.img
```

To view the checksum in **.treeinfo**:

```
$ grep sha256 dvd/.treeinfo
images/efiboot.img =
sha256:d357d5063b96226d643c41c9025529554a422acb43a4394e4ebcaa779cc7a917
images/install.img =
sha256:8c0323572f7fc04e34dd81c97d008a2ddfc2cfc525aef8c31459e21bf3397514
images/pxeboot/initrd.img =
sha256:fdb1a70321c06e25a1ed6bf3d8779371b768d5972078eb72b2c78c925067b5d8
images/pxeboot/vmlinuz =
sha256:b9510ea4212220e85351cbb7f2ebc2b1b0804a6d40ccb93307c165e16d1095db
```

Despite having correct **initrd.img**, if you get the following kernel messages during booting the installer, often a boot parameter is missing or mis-spelled, and the installer could not load **stage2**, typically referred to by the **inst.repo=** parameter, providing the full installer initial ramdisk for its in-memory root file system:

```
[ ...] No file system could mount root, tried:
[ ...] Kernel panic - not syncing: VFS: Unable to mount root fs on unknown-block(1,0)
[ ...] CPU: 0 PID: 1 Comm: swapper/0 Not tainted 6.12.0.el10_0.s390x #1
[ ...] ...
[ ...] Call Trace:
[ ...] ([<...>] show_trace+0x.../0x...)
[ ...]  [<...>] show_stack+0x.../0x...
[ ...]  [<...>] panic+0x.../0x...
[ ...]  [<...>] mount_block_root+0x.../0x...
[ ...]  [<...>] prepare_namespace+0x.../0x...
[ ...]  [<...>] kernel_init_freeable+0x.../0x...
[ ...]  [<...>] kernel_init+0x.../0x...
[ ...]  [<...>] kernel_thread_starter+0x.../0x...
[ ...]  [<...>] kernel_thread_starter+0x.../0x…
```

To resolve this issue, check

- if the installation source specified is correct on the kernel command line (**inst.repo=**) or in the kickstart file
- the network configuration is specified on the kernel command line (if the installation source is specified as network)
- the network installation source is accessible from another system

CHAPTER 26. BOOT OPTIONS REFERENCE

You can use the boot options to modify the default behavior of the installation program.

26.1. INSTALLATION SOURCE BOOT OPTIONS

You can configure installation source boot options to specify where RHEL installation files are located. These options enable installations from various sources including local media, network servers, and additional repositories, providing flexibility in deployment scenarios.

inst.repo=

The **inst.repo=** boot option specifies the installation source, that is, the location providing the package repositories and a valid **.treeinfo** file that describes them. For example: **inst.repo=cdrom**. The target of the **inst.repo=** option must be one of the following installation media:

- an installable tree, which is a directory structure containing the installation program images, packages, and repository data as well as a valid **.treeinfo** file
- a DVD (a physical disk present in the system DVD drive)
- an ISO image of the full Red Hat Enterprise Linux installation DVD, placed on a disk or a network location accessible to the system.
  Use the **inst.repo=** boot option to configure different installation methods by using different formats. The following table contains details of the **inst.repo=** boot option syntax:

Table 26.1. Types and format for the inst.repo= boot option and installation source

Source type	Boot option format	Source format
CD/DVD drive	**inst.repo=cdrom:*<device>***	Installation DVD as a physical disk. [a]
Mountable device (HDD and USB stick)	**inst.repo=hd:*<device>*:/*<path>***	Image file of the installation DVD.
NFS Server	**inst.repo=nfs:[*options*:]*<server>*:/*<path>***	Image file of the installation DVD, or an installation tree, which is a complete copy of the directories and files on the installation DVD. [b]
HTTP Server	**inst.repo=http://*<host>*/*<path>***	Installation tree that is a complete copy of the directories and files on the installation DVD.
HTTPS Server	**inst.repo=https://*<host>*/*<path>***	
FTP Server	**inst.repo=ftp://*<username>*:*<password>*@*<host>*/*<path>***	

Source type	Boot option format	Source format
HMC	**inst.repo=hmc**	

[a] If *device* is left out, installation program automatically searches for a drive containing the installation DVD.

[b] The NFS Server option uses NFS protocol version 3 by default. To use a different version, add **nfsvers=*X*** to *options*, replacing *X* with the version number that you want to use.

Set disk device names with the following formats:

- Kernel device name, for example **/dev/sda1** or **sdb2**
- File system label, for example **LABEL=Flash** or **LABEL=RHEL8**
- File system UUID, for example **UUID=8176c7bf-04ff-403a-a832-9557f94e61db**
  Non-alphanumeric characters must be represented as **\xNN**, where *NN* is the hexadecimal representation of the character. For example, **\x20** is a white space **(" ")**.

  inst.addrepo=

  Use the **inst.addrepo=** boot option to add an additional repository that you can use as another installation source along with the main repository (**inst.repo=**). You can use the **inst.addrepo=** boot option multiple times during one boot. The following table contains details of the **inst.addrepo=** boot option syntax.

  NOTE

  The **REPO_NAME** is the name of the repository and is required in the installation process. These repositories are only used during the installation process; they are not installed on the installed system.

Table 26.2. Installation sources and boot option format

Installation source	Boot option format	Additional information
Installable tree at a URL	**inst.addrepo=REPO_NAME,[http,https,ftp]://*<host>*/*<path>***	Looks for the installable tree at a given URL.
Installable tree at an NFS path	**inst.addrepo=REPO_NAME,nfs://*<server>*:/*<path>***	Looks for the installable tree at a given NFS path. A colon is required after the host. The installation program passes everything after **nfs://** directly to the mount command instead of parsing URLs according to RFC 2224.

Installation source	Boot option format	Additional information
Installable tree in the installation environment	**inst.addrepo=REPO_NAME,file://*<path>***	Looks for the installable tree at the given location in the installation environment. To use this option, the repository must be mounted before the installation program attempts to load the available software groups. The benefit of this option is that you can have multiple repositories on one bootable ISO, and you can install both the main repository and additional repositories from the ISO. The path to the additional repositories is **/run/install/source/REPO_ISO_PATH**. Additionally, you can mount the repository directory in the **%pre** section in the Kickstart file. The path must be absolute and start with /, for example **inst.addrepo=REPO_NAME,file:///*<path>***
Disk	**inst.addrepo=REPO_NAME,hd:*<device>*:*<path>***	Mounts the given *<device>* partition and installs from the ISO that is specified by the *<path>*. If the*<path>* is not specified, the installation program looks for a valid installation ISO on the *<device>*. This installation method requires an ISO with a valid installable tree.

inst.stage2=

The **inst.stage2=** boot option specifies the location of the installation program’s runtime image. This option expects the path to a directory that contains a valid **.treeinfo** file and reads the runtime image location from the **.treeinfo** file. If the **.treeinfo** file is not available, the installation program attempts to load the image from **images/install.img**.
When you do not specify the **inst.stage2** option, the installation program attempts to use the location specified with the **inst.repo** option.

You can use this option when you want to manually specify the installation source in the installation program at a later time. For example, when you want to select the Content Delivery Network (CDN) as an installation source. The installation DVD and Boot ISO already contain a suitable **inst.stage2** option to boot the installation program from the respective ISO.

If you want to specify an installation source, use the **inst.repo=** option instead.

NOTE

By default, the **inst.stage2=** boot option is used on the installation media and is set to a specific label; for example, **inst.stage2=hd:LABEL=RHEL-x-0-0-BaseOS-x86_64**. If you modify the default label of the file system that contains the runtime image, or if you use a customized procedure to boot the installation system, verify that the **inst.stage2=** boot option is set to the correct value.

inst.noverifyssl

Use the **inst.noverifyssl** boot option to prevent the installer from verifying SSL certificates for all HTTPS connections with the exception of additional Kickstart repositories, where **--noverifyssl** can be set per repository.
For example, if your remote installation source is using self-signed SSL certificates, the **inst.noverifyssl** boot option enables the installer to complete the installation without verifying the SSL certificates. For example, when specifying the source by using **inst.stage2=**:

```
inst.stage2=https://hostname/path_to_install_image/ inst.noverifyssl
```

And for example when specifying the source by using **inst.repo=**:

```
inst.repo=https://hostname/path_to_install_repository/ inst.noverifyssl
```

inst.stage2.all

Use the **inst.stage2.all** boot option to specify several HTTP, HTTPS, or FTP sources. You can use the **inst.stage2=** boot option multiple times with the **inst.stage2.all** option to fetch the image from the sources sequentially until one succeeds. For example:

```
inst.stage2.all
inst.stage2=http://hostname1/path_to_install_tree/
inst.stage2=http://hostname2/path_to_install_tree/
inst.stage2=http://hostname3/path_to_install_tree/
```

inst.dd=

The **inst.dd=** boot option is used to perform a driver update during the installation. For more information about how to update drivers during installation, see the *Updating drivers during installation*.

inst.repo=hmc

This option eliminates the requirement of an external network setup and expands the installation options. When booting from a Binary DVD, the installation program prompts you to enter additional kernel parameters. To set the DVD as an installation source, append the **inst.repo=hmc** option to the kernel parameters. The installation program then enables support element (SE) and hardware management console (HMC) file access, fetches the images for stage2 from the DVD, and provides access to the packages on the DVD for software selection.

IMPORTANT

To use the **inst.repo** boot option, ensure the user is configured with a **minimum of Privilege Class B**. For more information about the user configuration, see IBM documentation.

inst.proxy=

This boot option is used when performing an installation from a HTTP, HTTPS, and FTP protocol in the following form:

```
[PROTOCOL://][USERNAME[:PASSWORD]@]HOST[:PORT]
For example:
http://proxyuser:proxypassword@10.1.2.3:3128
```

inst.nosave=

Use the **inst.nosave=** boot option to control the installation logs and related files that are not saved to the installed system, for example **input_ks**, **output_ks**, **all_ks**, **logs** and **all**. You can combine multiple values separated by a comma. For example,

```
inst.nosave=input_ks,logs
```

NOTE

The **inst.nosave** boot option is used for excluding files from the installed system that cannot be removed by a Kickstart %post script, such as logs and input/output Kickstart results.

input_ks

Disables the ability to save the input Kickstart results.

output_ks

Disables the ability to save the output Kickstart results generated by the installation program.

all_ks

Disables the ability to save the input and output Kickstart results.

logs

Disables the ability to save all installation logs.

all

Disables the ability to save all Kickstart results, and all logs.

inst.multilib

Use the **inst.multilib** boot option to set DNF's **multilib_policy** to **all**, instead of **best**.

inst.memcheck

The **inst.memcheck** boot option performs a check to verify that the system has enough RAM to complete the installation. If there is not enough RAM, the installation process is stopped. The system check is approximate and memory usage during installation depends on the package selection, user interface, for example graphical or text, and other parameters.

inst.nomemcheck

The **inst.nomemcheck** boot option does not perform a check to verify if the system has enough RAM to complete the installation. Any attempt to perform the installation with less than the minimum amount of memory is unsupported, and might result in the installation process failing.

26.2. NETWORK BOOT OPTIONS

You can configure network boot options to enable RHEL installation over network connections. These options allow you to set up network interfaces, specify IP addresses, configure bonding, bridging, and VLANs for network-based installations and remote system deployment.

+

NOTE

Initialize the network with the **dracut** tool. For a complete list of **dracut** options, see the **dracut.cmdline(7)** man page on your system.

ip=

Use the **ip=** boot option to configure one or more network interfaces. To configure multiple interfaces, use one of the following methods

- use the **ip** option multiple times, once for each interface; to do so, use the **rd.neednet=1** option, and specify a primary boot interface using the **bootdev** option.
- use the **ip** option once, and then use Kickstart to set up further interfaces. While the configuration with ip= option is applied early in the installation process, the configuration defined in kickstart is applied at a later stage of installation, after the start of the installation program.
  This option accepts several different formats. The following tables contain information about the most common options. In the following tables:
- The **ip** parameter specifies the client IP address and **IPv6** requires square brackets, for example 192.0.2.1 or [2001:db8::99].
- The **gateway** parameter is the default gateway. **IPv6** requires square brackets.
- The **netmask** parameter is the netmask to be used. This can be either a full netmask (for example, 255.255.255.0) or a prefix (for example, 64).
- The **hostname** parameter is the host name of the client system. This parameter is optional.

Table 26.3. Boot option formats to configure the network interface

Boot option format	Configuration method
ip=*method*	Automatic configuration of all interfaces.
ip=*interface:method*	Automatic configuration of a specific interface

Boot option format	Configuration method
ip=*ip::gateway:netmask:hostname:interface*:none	Static configuration, for example, IPv4: **ip=192.0.2.1::192.0.2.254:255.255.255.0:server.example.com:enp1s0:none** IPv6: **ip=[2001:db8::1]::[2001:db8::fffe]:64:server.example.com:enp1s0:none**
ip=*ip::gateway:netmask:hostname:interface:method:mtu*	Automatic configuration of a specific interface with an override

Configuration methods for the automatic interface

The method **automatic configuration of a specific interface with an override** opens the interface by using the specified method of automatic configuration, such as **dhcp**, but overrides the automatically obtained IP address, gateway, netmask, host name or other specified parameters. All parameters are optional, so specify only the parameters that you want to override. For the values of the **method** parameter, see the **dracut.cmdline(7)** man page on your system.

NOTE

- If you use a boot option that requires network access, such as **inst.ks=http://host/path**, without specifying the **ip** option, the default value of the **ip** option is **ip=dhcp**..
- To connect to an iSCSI target automatically, activate a network device for accessing the target by using the **ip=ibft** boot option.

- **nameserver=** - The **nameserver=** option specifies the address of the name server. You can use this option multiple times.

  NOTE

  The **ip=** parameter requires square brackets. However, an IPv6 address does not work with square brackets. An example of the correct syntax to use for an IPv6 address is **nameserver=2001:db8::1**.

- **bootdev=** - The **bootdev=** option specifies the boot interface. This option is mandatory if you use more than one **ip** option.
- **ifname=** - The **ifname=** options assigns an interface name to a network device with a given MAC address. You can use this option multiple times. The syntax is **ifname=interface:MAC**. For example:

```
ifname=eth0:01:23:45:67:89:ab
```

> NOTE
>
> The **ifname=** option is the only supported way to set custom network interface names during installation.

- **inst.dhcpclass=** - The **inst.dhcpclass=** option specifies the DHCP vendor class identifier. The **dhcpd** service recognizes this value as **vendor-class-identifier**. The default value is **anaconda-$(uname -srm)**. To ensure the **inst.dhcpclass=** option is applied correctly, request network activation during the early stage of installation by also adding the **ip** option.

- **inst.waitfornet=** - Using the **inst.waitfornet=SECONDS** boot option causes the installation system to wait for network connectivity before installation. The value given in the **SECONDS** argument specifies the maximum amount of time to wait for network connectivity before timing out and continuing the installation process even if network connectivity is not present.

- **vlan=**- Use the **vlan=** option to configure a Virtual LAN (VLAN) device on a specified interface with a given name. The syntax is **vlan=name:interface**. For example:

  ```
  vlan=vlan5:enp0s1
  ```

  This configures a VLAN device named **vlan5** on the **enp0s1** interface. The name can take the following forms:

  - VLAN_PLUS_VID: **vlan0005**
  - VLAN_PLUS_VID_NO_PAD: **vlan5**
  - DEV_PLUS_VID: **enp0s1.0005**
  - DEV_PLUS_VID_NO_PAD: **enp0s1.5**

- **bond=** - Use the **bond=** option to configure a bonding device with the following syntax: **bond=name[:interfaces][:options]**. Replace *name* with the bonding device name, *interfaces* with a comma-separated list of physical (Ethernet) interfaces, and *options* with a comma-separated list of bonding options. For example:

  ```
  bond=bond0:enp0s1,enp0s2:mode=active-backup,tx_queues=32,downdelay=5000
  ```

  For a list of available options, execute the **modinfo bonding** command.

- **bridge=** - Use the **bridge=** option to configure a bridge device with the following syntax: **bridge=name:interfaces**. Replace *name* with the desired name of the bridge device and *interfaces* with a comma-separated list of physical (Ethernet) devices to be used as underlying interfaces in the bridge device. For example:

  ```
  bridge=bridge0:enp0s1,enp0s2
  ```

Additional resources

- Configuring and managing networking

26.3. CONSOLE BOOT OPTIONS

You can configure boot options for your console, monitor display, and keyboard to customize the installation process.

console=

Use the **console=** option to specify a device that you want to use as the primary console. For example, to use a console on the first serial port, use **console=ttyS0**. When using the **console=** argument, the installation starts with a text UI. If you must use the **console=** option multiple times, the boot message is displayed on all specified consoles. However, the installation program uses only the last specified console. For example, if you specify **console=ttyS0 console=ttyS1**, the installation program uses **ttyS1**.

inst.lang=

Use the **inst.lang=** option to set the language that you want to use during the installation. To view the list of locales, enter the command **locale -a | grep _** or the **localectl list-locales | grep _** command.

inst.geoloc=

Use the **inst.geoloc=** option to configure geolocation usage in the installation program. Geolocation is used to preset the language and time zone, and uses the following syntax: **inst.geoloc=value**. The **value** can be any of the following parameters:

- Disable geolocation: **inst.geoloc=0**
- Use the Fedora GeoIP API: **inst.geoloc=provider_fedora_geoip**. This option is deprecated.
- Use the Hostip.info GeoIP API: **inst.geoloc=provider_hostip**. This option is deprecated.

inst.keymap=

Use the **inst.keymap=** option to specify the keyboard layout to use for the installation.

inst.cmdline

Use the **inst.cmdline** option to force the installation program to run in command-line mode. This mode does not allow any interaction, and you must specify all options in a Kickstart file or on the command line.

inst.graphical

Use the **inst.graphical** option to force the installation program to run in graphical mode. The graphical mode is the default.

inst.text

Use the **inst.text** option to force the installation program to run in text mode instead of graphical mode.

inst.noninteractive

Use the **inst.noninteractive** boot option to run the installation program in a non-interactive mode. User interaction is not permitted in the non-interactive mode, and **inst.noninteractive** you can use the **inst.nointeractive** option with a graphical or text installation. When you use the **inst.noninteractive** option in text mode, it behaves the same as the **inst.cmdline** option.

NOTE

It only makes sense to use the **inst.noninteractive** option when performing a Kickstart installation.

inst.resolution=

Use the **inst.resolution=** option to specify the screen resolution in graphical mode. The format is **NxM**, where *N* is the screen width and *M* is the screen height (in pixels). The recommended resolution is 1024x768.

inst.rdp

Use the **inst.rdp** option to run the graphical installation using remote desktop protocol. If no RDP username (by using **inst.rdpuser=**) or password (by using **inst.rdp.password=**) are specified, the installation program asks users to provide interactively. This option only applies if used together with the **inst.rdp** option.

inst.rdp.password=

Use the **inst.rdp.password=** option to set a password on the RDP server that is used by the installation program.

modprobe.blacklist=

Use the **modprobe.blacklist=** option to blocklist or completely disable one or more drivers. Drivers (modules) that you disable using this option cannot load when the installation starts. After the installation finishes, the installed system retains these settings. You can find a list of the blocklisted drivers in the **/etc/modprobe.d/** directory. Use a comma-separated list to disable multiple drivers. For example:

```
modprobe.blacklist=ahci,firewire_ohci
```

NOTE

You can use **modprobe.blacklist** in combination with the different command line options. For example, use it with the **inst.dd** option to ensure that an updated version of an existing driver is loaded from a driver update disc:

```
modprobe.blacklist=virtio_blk
```

inst.sshd

Use the **inst.sshd** option to start the **sshd** service during installation, so that you can connect to the system during the installation using SSH, and monitor the installation progress. For more information about SSH, see the **ssh(1)** man page on your system. By default, the **sshd** service is automatically started only on the 64-bit IBM Z architecture. On other architectures, **sshd** is not started unless you use the **inst.sshd** option.

NOTE

During installation, the root account has no password by default. You can set a root password during installation with the **sshpw** Kickstart command.

inst.kdump_addon=

Use the **inst.kdump_addon=** option to enable or disable the Kdump configuration screen (add-on) in the installation program. This screen is enabled by default; use **inst.kdump_addon=off** to disable it. Disabling the add-on disables the Kdump screens in both the graphical and text-based interface as well as the **%addon com_redhat_kdump** Kickstart command.

26.4. DEBUG BOOT OPTIONS

You can use the following options when debugging issues to troubleshoot and fix issues.

inst.rescue

Use the **inst.rescue** option to run the rescue environment for diagnosing and fixing systems. For more information, see the Red Hat Knowledgebase solution repair a filesystem in rescue mode .

inst.updates=

Use the **inst.updates=** option to specify the location of the **updates.img** file that you want to apply during installation. The **updates.img** file can be derived from one of several sources.

Table 26.4. **updates.img** file sources

Source	Description	Example
Updates from a network	Specify the network location of **updates.img**. This does not require any modification to the installation tree. To use this method, edit the kernel command line to include **inst.updates**.	**inst.updates=http://website.com/path/to/updates.img**.
Updates from a disk image	Save an **updates.img** on a USB key.	**inst.updates=sda1:/images/updates.img** for an updates image located in the **images** directory on the **sda1** device or **inst.updates=UUID=b4234403-dafb-44c1-b878-4d57b40c9843:/updates.img** for an update image located in the root directory of a partition identified by its UUID.
Updates from an installation tree	If you are using a CD, disk, HTTP, HTTPS, or FTP installation, save the **updates.img** in the installation tree so that all installations can detect the **.img** file. The file name must be **updates.img**.	For NFS installs, save the file in the **images/** directory, or in the **RHupdates/** directory.

inst.syslog=

Sends log messages to the **syslog** process on the specified host when the installation starts. You can use **inst.syslog=<host>[:port]** only if the remote **syslog** process is configured to accept incoming connections.

inst.virtiolog=

Use the **inst.virtiolog=<name>** option to specify which virtio port (a character device at **/dev/virtio-ports/<name>**) to use for forwarding logs. The default value is **org.fedoraproject.anaconda.log.0**.

rd.live.ram

Copies the stage 2 image in **images/install.img** into RAM. Note that this increases the memory required for installation by the size of the image, which may result in an additional requirement of approximately 1 GiB of RAM or more.

inst.nokill

Prevent the installation program from rebooting when a fatal error occurs, or at the end of the installation process. Use it capture installation logs which would be otherwise lost upon reboot.

inst.noshell

Prevent a shell on terminal session 2 (tty2) including tmux window 2, during installation.

inst.notmux

Prevent the use of tmux during installation. The output is generated without terminal control characters and is meant for non-interactive uses.

26.5. STORAGE BOOT OPTIONS

You can specify the following options to customize booting from a storage device.

inst.nompath

Disables support for multipath devices. Use this option only if your system has a false-positive that incorrectly identifies a normal block device as a multipath device.

WARNING

Use this option with caution. Do not use this option with multipath hardware. Using this option to install to a single path of a multipath device is not supported.

inst.gpt

Prefer creation of GPT disk labels. This option is deprecated and will be removed in future releases. Use **inst.disklabel=gpt** instead.

inst.disklabel=

Prefer creation of the specified disk label type. Specify **gpt** to prefer creation of GPT disk labels (default). Specify **mbr** to prefer creation of MBR disk labels if supported.

inst.wait_for_disks=

Use the **inst.wait_for_disks=** option to specify the number of seconds installation program to wait for disk devices to appear at the beginning of the installation. Use this option when you use the **OEMDRV-labeled** device to automatically load the Kickstart file or the kernel drivers but the device takes a longer time to appear during the boot process. By default, the installation program waits for **5** seconds. Use **0** seconds to minimize the delay.

inst.nonibftiscsiboot

The **inst.nonibftiscsiboot** option enables placement of the boot loader on **iSCSI** devices that are not configured through the **iSCSI** boot firmware table (**iBFT**).