Red Hat Enterprise Linux 6

Installation Guide

Installing Red Hat Enterprise Linux 6.9 for all architectures

Last Updated: 2020-07-14

Red Hat Enterprise Linux 6 Installation Guide

Installing Red Hat Enterprise Linux 6.9 for all architectures

Clayton Spicer
Red Hat Customer Content Services
cspicer@redhat.com

Petr Bokoč
Red Hat Customer Content Services

Tomáš Čapek
Red Hat Customer Content Services

Jack Reed
Red Hat Customer Content Services

Rüdiger Landmann
Red Hat Customer Content Services

David Cantrell
VNC installation

Hans De Goede
iSCSI

Jon Masters
Driver updates

Red Hat Customer Content Services

Legal Notice

Abstract

This manual explains how to boot the Red Hat Enterprise Linux 6.9 installation program (anaconda) and to install Red Hat Enterprise Linux 6.9 on 32-bit and 64-bit x86 systems, 64-bit Power Systems servers, and IBM System z. It also covers advanced installation methods such as kickstart installations, PXE installations, and installations over VNC. Finally, it describes common post-installation tasks and explains how to troubleshoot installation problems.

Table of Contents

CHAPTER 1. OBTAINING RED HAT ENTERPRISE LINUX

If you have a Red Hat subscription, you can download *ISO image files* of the Red Hat Enterprise Linux 6.9 installation DVD from the Software & Download Center that is part of the Red Hat Customer Portal. If you do not already have a subscription, either purchase one or obtain a free evaluation subscription from the Software & Download Center at https://access.redhat.com/downloads.

The following table indicates the types of boot and installation media available for different architectures and notes the image file that you need to produce the media.

Table 1.1. Boot and installation media

Architecture	Installation DVD	Boot CD or boot DVD	Boot USB flash drive
BIOS-based 32-bit x86	x86 DVD ISO image file	**rhel-*variant*-*version*-i386-boot.iso**	**rhel-*variant*-*version*-i386-boot.iso**
UEFI-based 32-bit x86	Not available		
BIOS-based AMD64 and Intel 64	x86_64 DVD ISO image file (to install 64-bit operating system) or x86 DVD ISO image file (to install 32-bit operating system)	**rhel-*variant*-*version*-x86_64boot.iso** or**rhel-*variant*-*version*-i386-boot.iso**	**rhel-*variant*-*version*-x86_64boot.iso** or **rhel-*variant*-*version*-i386-boot.iso**
UEFI-based AMD64 and Intel 64	x86_64 DVD ISO image file	**rhel-*variant*-*version*-x86_64-boot.iso**	**efidisk.img** (from x86_64 DVD ISO image file)
POWER (64-bit only)	ppc DVD ISO image file	**rhel-server-*version*-ppc64-boot.iso**	Not available
System z	s390 DVD ISO image file	Not available	Not available

Where *variant* is the variant of Red Hat Enterprise Linux (for example,server or workstation) and *version* is the latest version number (for example, 6.5).

If you have a subscription or evaluation subscription, follow these steps to obtain the Red Hat Enterprise Linux 6.9 ISO image files:

Procedure 1.1. Downloading Red Hat Enterprise Linux ISO Images

1. Visit the Customer Portal at https://access.redhat.com/home. If you are not logged in, click **LOG IN** on the right side of the page. Enter your account credentials when prompted.
2. Click **DOWNLOADS** at the top of the page.
3. Click **Red Hat Enterprise Linux**.
4. Ensure that you select the appropriate **Product Variant**, **Version** and **Architecture** for your

installation target. By default, **Red Hat Enterprise Linux Server** and **x86_64** are selected. If you are not sure which variant best suits your needs, see http://www.redhat.com/en/technologies/linux-platforms/enterprise-linux.

5. A list of available downloads is displayed; most notably, a minimal **Boot ISO** image and a full installation **Binary DVD** ISO image. The Boot ISO is a minimal boot image which only contains the installer and requires a source to install packages from (such as an HTTP or FTP server). The Binary DVD download contains both the installer and necessary packages, and therefore requires less setup.

   Additional images may be available, such as preconfigured virtual machine images, which are beyond the scope of this document.

6. Choose the image file that you want to use. There are several ways to download an ISO image from Red Hat Customer Portal:

   - Click its name to begin downloading it to your computer using your web browser.
   - Right-click the name and then click **Copy Link Location** or a similar menu item, the exact wording of which depends on the browser that you are using. This action copies the URL of the file to your clipboard, which allows you to use an alternative application to download the file to your computer. This approach is especially useful if your Internet connection is unstable: in that case, you browser may fail to download the whole file, and an attempt to resume the interrupted download process fails because the download link contains an authentication key which is only valid for a short time. Specialized applications such as **curl** can, however, be used to resume interrupted download attempts from the Customer Portal, which means that you need not download the whole file again and thus you save your time and bandwidth consumption.

     Procedure 1.2. Using curl to Download Installation Media

     1. Make sure the curl package is installed by running the following command as root:

        ```
        # yum install curl
        ```

        If your Linux distribution does not use **yum**, or if you do not use Linux at all, download the most appropriate software package from the curl website.

     2. Open a terminal window, enter a suitable directory, and type the following command:

        ```
        $ curl -o filename.iso 'copied_link_location'
        ```

        Replace *filename.iso* with the ISO image name as displayed in the Customer Portal, such as **rhel-server-6.9-x86_64-dvd.iso**. This is important because the download link in the Customer Portal contains extra characters which **curl** would otherwise use in the downloaded file name, too. Then, keep the single quotation mark in front of the next parameter, and replace *copied_link_location* with the link that you have copied from the Customer Portal.

        Note that in Linux, you can paste the content of the clipboard into the terminal window by middle-clicking anywhere in the window, or by pressing **Shift**+**Insert**. Finally, use another single quotation mark after the last parameter, and press **Enter** to run the command and start transferring the ISO image. The single quotation marks prevent the command line interpreter from misinterpreting any special characters that might be included in the download link.

Example 1.1. Downloading an ISO image with curl

The following is an example of a **curl** command line:

```
$ curl -o rhel-server-6.9-x86_64-dvd.iso
'https://access.cdn.redhat.com//content/origin/files/sha256/85/85a...46c/rhel-server-6.9-x86_64-dvd.iso?_auth_=141...7bf'
```

Note that the actual download link is much longer because it contains complicated identifiers.

3. If your Internet connection does drop before the transfer is complete, refresh the download page in the Customer Portal; log in again if necessary. Copy the new download link, use the same basic **curl** command line parameters as earlier but be sure to use the new download link, and add **-C -** to instruct **curl** to automatically determine where it should continue based on the size of the already downloaded file.

Example 1.2. Resuming an interrupted download attempt

The following is an example of a **curl** command line that you use if you have only partially downloaded the ISO image of your choice:

```
$ curl -o rhel-server-6.9-x86_64-dvd.iso
'https://access.cdn.redhat.com//content/origin/files/sha256/85/85a...46c/rhel-server-6.9-x86_64-dvd.iso?_auth_=141...963' -C -
```

7. Optionally, you can use a checksum utility such as **sha256sum** to verify the integrity of the image file after the download finishes. All downloads on the Download Red Hat Enterprise Linux page are provided with their checksums for reference:

```
$ sha256sum rhel-server-6.9-x86_64-dvd.iso
85a...46c rhel-server-6.9-x86_64-dvd.iso
```

Similar tools are available for Microsoft Windows and Mac OS X. You can also use the installation program to verify the media when starting the installation; see Section 28.6.1, "Verifying Boot Media" for details.

After you download an ISO image file of the installation DVD from the Red Hat Customer Portal, you can:

- burn it to a physical DVD (refer to Section 2.1, "Making an Installation DVD").
- use it to prepare minimal boot media (refer to Section 2.2, "Making Minimal Boot Media").
- place it on a server to prepare for installations over a network (refer to Section 4.1, "Preparing for a Network Installation" for x86 architectures, Section 12.1, "Preparing for a Network Installation" for Power Systems servers or Section 19.1, "Preparing for a Network Installation" for IBM System z).
- place it on a hard drive to prepare for installation to use the hard drive as an installation source (refer to Section 4.2, "Preparing for a Hard Drive Installation" for x86 architectures, Section 12.2, "Preparing for a Hard Drive Installation" for Power Systems servers or Section 19.2, "Preparing for a Hard Drive Installation" for IBM System z).

- place it on a *pre-boot execution environment* (PXE) server to prepare for installations using PXE boot (refer to Chapter 30, *Setting Up an Installation Server*).

CHAPTER 2. MAKING MEDIA

Use the methods described in this section to create the following types of installation and boot media:

- an installation DVD
- a minimal boot CD or DVD that can boot the installer
- a USB flash drive to boot the installer

2.1. MAKING AN INSTALLATION DVD

You can make an installation DVD using the CD or DVD burning software on your computer.

Make sure that your disc burning software is capable of burning discs from image files. Although this is true of most disc burning software, exceptions exist. In particular, note that the disc burning feature built into Windows XP and Windows Vista cannot burn DVDs; and that earlier Windows operating systems did not have any disc burning capability installed by default at all. Therefore, if your computer has a Windows operating system prior to Windows 7 installed on it, you need separate software for this task. Examples of popular disc burning software for Windows that you might already have on your computer include **Nero Burning ROM** and **Roxio Creator**.

Most widely used disc burning software for Linux, such as **Brasero** and **K3b** has the built-in ability to burn discs from ISO image files.

The exact series of steps that produces a DVD from an ISO image file varies greatly from computer to computer, depending on the operating system and disc burning software installed. Consult your disc burning software's documentation for detailed information on burning DVDs.

2.2. MAKING MINIMAL BOOT MEDIA

A piece of *minimal boot media* is a CD, DVD, or USB flash drive that contains the software to boot the system and launch the installation program, but which does not contain the software that must be transferred to the system to create a Red Hat Enterprise Linux installation.

Use minimal boot media:

- to boot the system to install Red Hat Enterprise Linux over a network
- to boot the system to install Red Hat Enterprise Linux from a hard drive
- to use a kickstart file during installation (refer to Section 32.9.1, “Creating Kickstart Boot Media”
- to commence a network or hard-drive installation or to use an **anaconda** update or a kickstart file with a DVD installation.

You can use minimal boot media to start the installation process on 32-bit x86 systems, AMD64 or Intel 64 systems, and Power Systems servers. The process by which you create minimal boot media for systems of these various types is identical except in the case of AMD64 and Intel 64 systems with UEFI firmware interfaces – refer to Section 2.2.2, “Minimal USB Boot Media for UEFI-based Systems” .

To make minimal boot media for 32-bit x86 systems, BIOS-based AMD64 or Intel 64 systems, and Power Systems servers:

1. Download the ISO image file named **rhel-*variant*-*version*-*architecture*-boot.iso** that is available at the same location as the images of the Red Hat Enterprise Linux 6.9 installation DVD – refer to Chapter 1, *Obtaining Red Hat Enterprise Linux* .

2. Burn the **.iso** file to a blank CD or DVD using the same procedure detailed in Section 2.1, "Making an Installation DVD" for the installation disc.

Alternatively, transfer the **.iso** file to a USB device with the **dd** command. As the **.iso** file is only around 200 MB in size, you do not need an especially large USB flash drive.

2.2.1. Minimal USB Boot Media for BIOS-based Systems

WARNING

When you perform this procedure any data on the USB flash drive is destroyed with no warning. Make sure that you specify the correct USB flash drive, and make sure that this flash drive does not contain any data that you want to keep.

1. Plug in your USB flash drive.

2. Find the flash drive's device name. If the media has a volume name, use it to look up the device name in **/dev/disk/by-label**, or use the **findfs** command:

```
findfs LABEL=MyLabel
```

   If the media does not have a volume name or you do not know it, you can also use the **dmesg** command shortly after connecting the media to your computer. After running the command, the device name (such as **sdb** or **sdc**) should appear in several lines towards the end of the output.

3. Become root:

```
su -
```

4. Use the **dd** command to transfer the boot ISO image to the USB device:

```
# dd if=path/image_name.iso of=/dev/device
```

   where *path/image_name*.iso is the boot ISO image file that you downloaded and ***device*** is the device name for the USB flash drive. Ensure you specify the device name (such as **sdc**), not the partition name (such as **sdc1**). For example:

```
# dd if=~/Downloads/RHEL6.9-Server-x86_64-boot.iso of=/dev/sdc
```

2.2.2. Minimal USB Boot Media for UEFI-based Systems

WARNING

When you perform this procedure any data on the USB flash drive is destroyed with no warning. Make sure that you specify the correct USB flash drive, and make sure that this flash drive does not contain any data that you want to keep.

To creater minimal USB boot media for Red Hat Enterprise Linux, use the **efidisk.img** file in the **images**/ directory on the Red Hat Enterprise Linux 6.9 installation DVD:

1. Download an ISO image file of the Red Hat Enterprise Linux 6.9 installation DVD as described in Chapter 1, *Obtaining Red Hat Enterprise Linux* .
2. Become root:

```
su -
```

3. Create a mount point for the ISO image file:

```
# mkdir /mnt/dvdiso
```

4. Mount the image file:

```
# mount DVD.iso /mnt/dvdiso -o loop
```

Where *DVD.iso* is the name of the ISO image file, for example **RHEL6.9-Server-x86_64-DVD.iso**.

5. Transfer **efidisk.img** from the ISO image file to your USB flash drive:

```
# dd if=/mnt/dvdiso/images/efidisk.img of=/dev/device_name
```

For example:

```
# dd if=/mnt/dvdiso/images/efidisk.img of=/dev/sdc
```

NOTE

Use the **dd** command to write the image file directly to the device. Using **cp** to copy the file or transferring the file using a file manager will make the device unbootable.

6. Unmount the ISO image file:

```
# umount /mnt/dvdiso
```

2.3. CREATING A USGCB-COMPLIANT INSTALLATION IMAGE

The scap-security-guide package in Red Hat Enterprise Linux 6 contains a specialized Kickstart file, which can be used to install a hardened system conforming to the *United States Government Configuration Baseline* (USGCB) standard. This is useful in cases where compliance with this standard is required by government regulations.

This Kickstart configuration can be used with the Server variant of Red Hat Enterprise Linux 6. If used, the system will be automatically configured by **OpenSCAP** to be USGCB profile compliant as part of the post-installation script. After the installation finishes, you can review a report placed in the **/root/** directory on the installed system.

NOTE

The Kickstart file provided by scap-security-guide contains all required commands, making the installation completely automatic.

Also note that the Kickstart file requires access to the internet during the installation in order to download the latest benchmark.

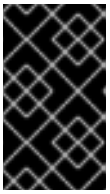
For more information about compliance and vulnerability scanning using **OpenSCAP**, see the appropriate chapter of the Red Hat Enterprise Linux 6 Security Guide .

To obtain the Kickstart file, install the scap-security-guide package on an existing Red Hat Enterprise Linux 6 system. Once the package is installed, you can find the Kickstart file at **/usr/share/scap-security-guide/kickstart/ssg-rhel6-usgcb-server-with-gui-ks.cfg**.

After obtaining the file, copy it into your home directory and edit it using a plain text editor. Use Section 32.4, “Kickstart Options” and comments in the file for reference. Some of the comments mention *Common Configuration Enumeration* (CCE) identifier numbers; you can find information about these at the CCE Archive.

Notable parts of the Kickstart file which can be changed are:

- Package repository location - the **url** command. To use a package repository on an HTTP or FTP server, replace the default IP address with an address of a server containing a package repository. Replace this command with one of **nfs**, **cdrom**, or **harddrive** to install from a NFS server, optical drive, or local hard drive, respectively.
- System language, keyboard layout, and time zone - the **lang**, **keyboard** and **timezone** commands.
- Root password - the **rootpw** command. By default, the root password configured in this Kickstart is "server". Make sure to generate a new checksum and change it.
- Boot loader password - the **bootloader --password=** command. The default password is "password". Make sure to generate a new checksum and change it.
- Network configuration - the **network** command. Automatic configuration using DHCP is enabled by default - adjust the settings if necessary.
- Package selection - modify the **%packages** section of the file to install packages and groups you need.

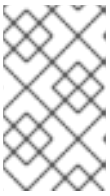


IMPORTANT

Packages git, aide and openscap-utils must always be installed. They are required for the Kickstart file and post installation **OpenSCAP** system evaluation to work.

- Disk partitioning layout - the **part**, **volgroup** and **logvol** commands.

  The USGCB standard defines concrete requirements for a compliant system's disk layout, which means that the logical volumes defined in the default Kickstart file - **/home**, **/tmp**, **/var**, **/var/log**, and **/var/log/audit** - must always be created as separate partitions or logical volumes. Additionally, Red Hat Enterprise Linux requires you to create a **/boot** physical partition and volumes for / and **swap**. These are all defined in the default Kickstart; you can add additional separate logical volumes or partitions, and you can change the sizes of the default ones.



NOTE

By default, the **/var/log/audit** volume only takes up 512 MB of space. Due to the high number of calls being audited, it is highly recommended to increase its size to at least 1024 MB.

The rest of the Kickstart file can be used as-is. Once you finish modifying the file, proceed with Section 32.9.1, “Creating Kickstart Boot Media” to place it on an ISO image and use it to install a new system.

PART I. X86, AMD64, AND INTEL 64 – INSTALLATION AND BOOTING

This part of the *Red Hat Enterprise Linux Installation Guide* for Intel and AMD 32-bit and 64-bit systems discusses the installation of Red Hat Enterprise Linux and some basic post-installation troubleshooting.

For advanced installation options, refer to Part IV, “Advanced Installation Options”.

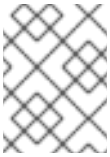
CHAPTER 3. PLANNING FOR INSTALLATION ON THE X86 ARCHITECTURE

3.1. UPGRADE OR INSTALL?

There are two procedures available for upgrading your current system to the next major version of Red Hat Enterprise Linux. To decide which procedure is the right one for your system, read the following descriptions:

Clean Install

A clean install is performed by backing up all data from the system, formatting disk partitions, performing an installation of Red Hat Enterprise Linux 7 from installation media, and then restoring any user data.



NOTE

This is the recommended method for upgrading between major versions of Red Hat Enterprise Linux.

In-Place Upgrade

An in-place upgrade is a way of upgrading your system without removing the older version first. The procedure requires installing the migration utilities available for your system and running them as any other software. In Red Hat Enterprise Linux, the **Preupgrade Assistant** assesses your current system and identifies potential problems you might encounter during and/or after the upgrade. It also performs minor fixes and modifications to the system. The **Red Hat Upgrade Tool** utility downloads the packages and performs the actual upgrade. An in-place upgrade requires a lot of troubleshooting and planning and should only be done if there is no other choice. For more information on the **Preupgrade Assistant**, see Chapter 37, *Upgrading Your Current System*.

WARNING

Never perform an in-place upgrade on a production system without first testing it on a cloned backup copy of the system.

3.2. IS YOUR HARDWARE COMPATIBLE?

Hardware compatibility is particularly important if you have an older system or a system that you built yourself. Red Hat Enterprise Linux 6.9 should be compatible with most hardware in systems that were factory built within the last two years.

However, hardware specifications change almost daily, so it is difficult to guarantee that your hardware is 100% compatible.

One consistent requirement is your processor. Red Hat Enterprise Linux 6.9 supports, at minimum, all 32-bit and 64-bit implementations of Intel microarchitecture from P6 and onwards and AMD microarchitecture from Athlon and onwards.

The most recent list of supported hardware can be found at:

> https://hardware.redhat.com/

3.3. HARDWARE REQUIREMENTS

For a list of minimum hardware requirements of Red Hat Enterprise Linux 6, see the Red Hat Enterprise Linux technology capabilities and limits page. Also note that the minimum memory requirements listed on that page assume that you create a swap space based on the recommendations in Section 9.15.5, "Recommended Partitioning Scheme". Systems with low memory (1 GB and less) and less than the recommended amount of swap space may have issues ranging from low responsivity up to and including complete inability to boot after the installation.

For installation of Red Hat Enterprise Linux on x86, AMD64, and Intel 64 systems, Red Hat supports the following installation targets:

- Hard drives connected by a standard internal interface, such as SCSI, SATA, or SAS
- BIOS/firmware RAID devices

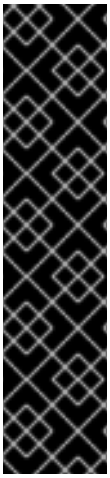
Fibre Channel Host Bus Adapters and multipath devices are also supported. Vendor-provided drivers may be required for certain hardware.

Red Hat does not support installation to USB drives or SD memory cards.

Red Hat also supports installations that use the following virtualization technologies:

- Xen block devices on Intel processors in Xen virtual machines.
- VirtIO block devices on Intel processors in KVM virtual machines.

3.4. RAID AND OTHER DISK DEVICES



IMPORTANT

Red Hat Enterprise Linux 6 uses **mdraid** instead of **dmraid** for installation onto Intel BIOS RAID sets. These sets are detected automatically, and devices with Intel ISW metadata are recognized as mdraid instead of dmraid. Note that the device node names of any such devices under **mdraid** are different from their device node names under **dmraid**. Therefore, special precautions are necessary when you migrate systems with Intel BIOS RAID sets.

Local modifications to **/etc/fstab**, **/etc/crypttab** or other configuration files which refer to devices by their device node names will not work in Red Hat Enterprise Linux 6. Before migrating these files, you must therefore edit them to replace device node paths with device UUIDs instead. You can find the UUIDs of devices with the **blkid** command.

3.4.1. Hardware RAID

RAID, or Redundant Array of Independent Disks, allows a group, or array, of drives to act as a single

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

On systems with more than one hard drive you may configure Red Hat Enterprise Linux to operate several of the drives as a Linux RAID array without requiring any additional hardware.

3.4.2. Software RAID

You can use the Red Hat Enterprise Linux installation program to create Linux software RAID arrays, where RAID functions are controlled by the operating system rather than dedicated hardware. These functions are explained in detail in Section 9.15, " Creating a Custom Layout or Modifying the Default Layout ".

3.4.3. FireWire and USB Disks

Some FireWire and USB hard disks may not be recognized by the Red Hat Enterprise Linux installation system. If configuration of these disks at installation time is not vital, disconnect them to avoid any confusion.

NOTE

You can connect and configure external FireWire and USB hard disks after installation. Most such devices are automatically recognized and available for use once connected.

3.5. NOTES ON UEFI SUPPORT

3.5.1. Feature Support

Red Hat Enterprise Linux 6.9 supports both BIOS and UEFI firmware on AMD64 and Intel 64 systems (x86_64). UEFI-based systems are supported with the following limitations:

- The system must support UEFI Specification 2.0 or later. Earlier revisions are not supported.
- The Secure Boot technology is not supported, and will prevent Red Hat Enterprise Linux from being installed. Systems using UEFI Specification 2.2 or later must have Secure Boot disabled in order to install and run Red Hat Enterprise Linux 6.9.

Systems using UEFI 2.0 later with Secure Boot disabled (if present) can install and boot Red Hat Enterprise Linux without issues, although not all features in the relevant UEFI specification are supported.

For more information about UEFI specifications, see http://www.uefi.org/specifications.

3.5.2. Disk Drives with MBR on UEFI Systems

Systems with UEFI firmware require a disk with a GUID Partition Table (GPT). When installing Red Hat Enterprise Linux on a disk with a Master Boot Record (MBR; sometimes also called **msdos**) label, the disk must be relabeled. This means you can not reuse existing partitions on a MBR-partitioned disk, and all data on the disk will be lost. Make sure to back up all data on the drive before installing Red Hat Enterprise Linux.

A GUID Partition Table is only required on the system's boot drive - the disk where the boot loader is installed. Other drives can be labeled with a Master Boot Record and their partition layout can be reused.

There are several ways to install Red Hat Enterprise Linux on an UEFI system and use a drive which has a Master Boot Record. You can:

- Attach the drive to an existing Linux system and use an utility such as **parted** or **fdisk** to create a GPT label on the drive. For example, to create a GPT label on disk **/dev/sdc** using **parted**, use the following command:

  ```
  # parted /dev/sdc mklabel gpt
  ```

  WARNING

  Make sure you specify the correct drive. Relabeling a disk will destroy all data on it, and **parted** will not ask you for a confirmation.

- Perform an automated Kickstart installation, and use the **clearpart** and **zerombr** commands. If your system uses UEFI firmware, using these commands on the boot drive will relabel it with a GPT.

- During a manual installation in the graphical user interface, when you get to the partitioning screen. Select an option *other than* custom partitioning (for example **Use All Space**). Make sure to check the **Review and modify partitioning layout** check box, and click **Next**.

  On the following screen, modify the automatically created layout so it suits your needs. After you finish and click **Next**, **Anaconda** will use your layout and relabel the drive automatically.

3.6. DO YOU HAVE ENOUGH DISK SPACE?

Nearly every modern-day operating system (OS) uses *disk partitions*, and Red Hat Enterprise Linux is no exception. When you install Red Hat Enterprise Linux, you may have to work with disk partitions. If you have not worked with disk partitions before (or need a quick review of the basic concepts), refer to Appendix A, *An Introduction to Disk Partitions* before proceeding.

The disk space used by Red Hat Enterprise Linux must be separate from the disk space used by other OSes you may have installed on your system, such as Windows, OS/2, or even a different version of Linux. For x86, AMD64, and Intel 64 systems, at least two partitions (/ and **swap**) must be dedicated to Red Hat Enterprise Linux.

Before you start the installation process, you must

- have enough *unpartitioned*[1] disk space for the installation of Red Hat Enterprise Linux, or

- have one or more partitions that may be deleted, thereby freeing up enough disk space to install Red Hat Enterprise Linux.

To gain a better sense of how much space you really need, refer to the recommended partitioning sizes discussed in Section 9.15.5, "Recommended Partitioning Scheme".

If you are not sure that you meet these conditions, or if you want to know how to create free disk space for your Red Hat Enterprise Linux installation, refer to Appendix A, *An Introduction to Disk Partitions*.

3.7. SELECTING AN INSTALLATION METHOD

What type of installation method do you wish to use? The following installation methods are available:

DVD

If you have a DVD drive and the Red Hat Enterprise Linux DVD you can use this method. Refer to Section 8.3.1, "Installing from a DVD" , for DVD installation instructions.

If you booted the installation from a piece of media other than the installation DVD, you can specify the DVD as the installation source with the **linux askmethod** or **linux repo=cdrom:*device*:/*device*** boot option, or by selecting **Local CD/DVD** on the **Installation Method** menu (refer to Section 8.3, "Installation Method").

Hard Drive

If you have copied the Red Hat Enterprise Linux ISO images to a local hard drive, you can use this method. You need a boot CD-ROM (use the **linux askmethod** or **linux repo=hd:*device*:/*path*** boot option), or by selecting **Hard drive** on the **Installation Method** menu (refer to Section 8.3, "Installation Method"). Refer to Section 8.3.2, "Installing from a Hard Drive" , for hard drive installation instructions.

NFS

If you are installing from an NFS server using ISO images or a mirror image of Red Hat Enterprise Linux, you can use this method. You need a boot CD-ROM (use the **linux askmethod** or **linux repo=nfs:*server :options*:/*path*** boot option, or the **NFS directory** option on the **Installation Method** menu described in Section 8.3, "Installation Method"). Refer to Section 8.3.4, "Installing via NFS" for network installation instructions. Note that NFS installations may also be performed in GUI mode.

URL

If you are installing directly from an HTTP or HTTPS (Web) server or an FTP server, use this method. You need a boot CD-ROM (use the **linux askmethod**, **linux repo=ftp://*user*:*password*@*host*/*path***, or **linux repo=http://*host*/*path*** boot option, or **linux repo=https://*host*/*path*** boot option,or the **URL** option on the **Installation Method** menu described in Section 8.3, "Installation Method"). Refer to Section 8.3.5, "Installing via FTP, HTTP, or HTTPS" , for FTP, HTTP, and HTTPS installation instructions.

If you booted the distribution DVD and did not use the alternate installation source option **askmethod**, the next stage loads automatically from the DVD. Proceed to Section 8.2, "Language Selection".

NOTE

If you boot from a Red Hat Enterprise Linux installation DVD, the installation program loads its next stage from that disc. This happens regardless of which installation method you choose, unless you eject the disc before you proceed. The installation program still downloads *package data* from the source you choose.

3.8. CHOOSE A BOOT METHOD

You can use several methods to boot Red Hat Enterprise Linux.

Installing from a DVD requires that you have purchased a Red Hat Enterprise Linux product, you have a Red Hat Enterprise Linux 6.9 DVD, and you have a DVD drive on a system that supports booting from it. Refer to Chapter 2, *Making Media* for instructions to make an installation DVD.

Your BIOS may need to be changed to allow booting from your DVD/CD-ROM drive. For more information about changing your BIOS, refer to Section 7.1.1, "Booting the Installation Program on x86, AMD64, and Intel 64 Systems".

Other than booting from an installation DVD, you can also boot the Red Hat Enterprise Linux installation program from *minimal boot media* in the form of a bootable CD or USB flash drive. After you boot the system with a piece of minimal boot media, you complete the installation from a different installation source, such as a local hard drive or a location on a network. Refer to Section 2.2, "Making Minimal Boot Media" for instructions on making boot CDs and USB flash drives.

Finally, you can boot the installer over the network from a *preboot execution environment* (PXE) server. Refer to Chapter 30, *Setting Up an Installation Server*. Again, after you boot the system, you complete the installation from a different installation source, such as a local hard drive or a location on a network.

---

[1] Unpartitioned disk space means that available disk space on the hard drives you are installing to has not been divided into sections for data. When you partition a disk, each partition behaves like a separate disk drive.

CHAPTER 4. PREPARING FOR INSTALLATION

4.1. PREPARING FOR A NETWORK INSTALLATION

NOTE

Make sure no installation DVD (or any other type of DVD or CD) is in your system's CD or DVD drive if you are performing a network-based installation. Having a DVD or CD in the drive might cause unexpected errors.

Ensure that you have boot media available on CD, DVD, or a USB storage device such as a flash drive.

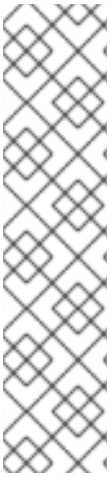
The Red Hat Enterprise Linux installation medium must be available for either a network installation (via NFS, FTP, HTTP, or HTTPS) or installation via local storage. Use the following steps if you are performing an NFS, FTP, HTTP, or HTTPS installation.

The NFS, FTP, HTTP, or HTTPS server to be used for installation over the network must be a separate, network-accessible server. It must provide the complete contents of the installation DVD-ROM.

NOTE

anaconda has the ability to test the integrity of the installation media. It works with the DVD, hard drive ISO, and NFS ISO installation methods. We recommend that you test all installation media before starting the installation process, and before reporting any installation-related bugs (many of the bugs reported are actually due to improperly-burned DVDs). To use this test, type the following command at the **boot:** prompt:

```
linux mediacheck
```



NOTE

The public directory used to access the installation files over FTP, NFS, HTTP, or HTTPS is mapped to local storage on the network server. For example, the local directory **/var/www/inst/rhel6.9** on the network server can be accessed as **http://network.server.com/inst/rhel6.9**.

In the following examples, the directory on the installation staging server that will contain the installation files will be specified as ***/location/of/disk/space***. The directory that will be made publicly available via FTP, NFS, HTTP, or HTTPS will be specified as */publicly_available_directory*. For example, ***/location/of/disk/space*** may be a directory you create called **/var/isos**. ***/publicly_available_directory*** might be **/var/www/html/rhel6.9**, for an HTTP install.

In the following, you will require an *ISO image*. An ISO image is a file containing an exact copy of the content of a DVD. To create an ISO image from a DVD use the following command:

```
dd if=/dev/dvd of=/path_to_image/name_of_image.iso
```

where *dvd* is your DVD drive device, *name_of_image* is the name you give to the resulting ISO image file, and *path_to_image* is the path to the location on your system where the resulting ISO image will be stored.

To copy the files from the installation DVD to a Linux instance, which acts as an installation staging server, continue with either Section 4.1.1, "Preparing for FTP, HTTP, and HTTPS Installation" or Section 4.1.2, "Preparing for an NFS Installation" .

4.1.1. Preparing for FTP, HTTP, and HTTPS Installation

WARNING

If your **Apache** web server or **tftp** FTP server configuration enables SSL security, make sure to only enable the **TLSv1** protocol, and disable **SSLv2** and **SSLv3**. This is due to the POODLE SSL vulnerability (CVE-2014-3566). See https://access.redhat.com/solutions/1232413 for details about securing **Apache**, and https://access.redhat.com/solutions/1234773 for information about securing **tftp**.

Extract the files from the ISO image of the installation DVD and place them in a directory that is shared over FTP, HTTP, or HTTPS.

Next, make sure that the directory is shared via FTP, HTTP, or HTTPS, and verify client access. Test to see whether the directory is accessible from the server itself, and then from another machine on the same subnet to which you will be installing.

4.1.2. Preparing for an NFS Installation

For NFS installation it is not necessary to extract all the files from the ISO image. It is sufficient to make the ISO image itself, the **install.img** file, and optionally the **product.img** file available on the network server via NFS.

1. Transfer the ISO image to the NFS exported directory. On a Linux system, run:

   ```
   mv /path_to_image/name_of_image.iso /publicly_available_directory/
   ```

   where *path_to_image* is the path to the ISO image file, *name_of_image* is the name of the ISO image file, and *publicly_available_directory* is a directory that is available over NFS or that you intend to make available over NFS.

2. Use a SHA256 checksum program to verify that the ISO image that you copied is intact. Many SHA256 checksum programs are available for various operating systems. On a Linux system, run:

   ```
   $ sha256sum name_of_image.iso
   ```

   where *name_of_image* is the name of the ISO image file. The SHA256 checksum program displays a string of 64 characters called a *hash*. Compare this hash to the hash displayed for this particular image on the **Downloads** page in the Red Hat Customer Portal (refer to Chapter 1, *Obtaining Red Hat Enterprise Linux*). The two hashes should be identical.

3. Copy the **images/** directory from inside the ISO image to the same directory in which you stored the ISO image file itself. Enter the following commands:

```
mount -t iso9660 /path_to_image/name_of_image.iso /mount_point -o loop,ro
cp -pr /mount_point/images /publicly_available_directory/
umount /mount_point
```

where ***path_to_image*** is the path to the ISO image file, ***name_of_image*** is the name of the ISO image file, and ***mount_point*** is a mount point on which to mount the image while you copy files from the image. For example:

```
mount -t iso9660 /var/isos/RHEL6.iso /mnt/tmp -o loop,ro
cp -pr /mnt/tmp/images /var/isos/
umount /mnt/tmp
```

The ISO image file and an **images/** directory are now present, side-by-side, in the same directory.

4. Verify that the **images/** directory contains at least the **install.img** file, without which installation cannot proceed. Optionally, the **images/** directory should contain the **product.img** file, without which only the packages for a **Minimal** installation will be available during the package group selection stage (refer to Section 9.17, "Package Group Selection").

IMPORTANT

install.img and **product.img** must be the only files in the **images/** directory.

5. Ensure that an entry for the publicly available directory exists in the **/etc/exports** file on the network server so that the directory is available via NFS.

   To export a directory read-only to a specific system, use:

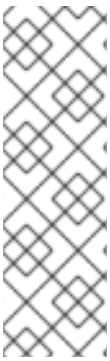
   ```
   /publicly_available_directory client.ip.address (ro)
   ```

   To export a directory read-only to all systems, use:

   ```
   /publicly_available_directory * (ro)
   ```

6. On the network server, start the NFS daemon (on a Red Hat Enterprise Linux system, use **/sbin/service nfs start**). If NFS is already running, reload the configuration file (on a Red Hat Enterprise Linux system use **/sbin/service nfs reload**).

7. Be sure to test the NFS share following the directions in the Red Hat Enterprise Linux Deployment Guide. Refer to your NFS documentation for details on starting and stopping the NFS server.

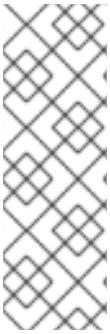


NOTE

anaconda has the ability to test the integrity of the installation media. It works with the DVD, hard drive ISO, and NFS ISO installation methods. We recommend that you test all installation media before starting the installation process, and before reporting any installation-related bugs (many of the bugs reported are actually due to improperly-burned DVDs). To use this test, type the following command at the **boot:** prompt:

```
linux mediacheck
```

4.2. PREPARING FOR A HARD DRIVE INSTALLATION



NOTE

Hard drive installations only work from ext2, ext3, ext4, or FAT file systems. You cannot use a hard drives formatted for any other file system as an installation source for Red Hat Enterprise Linux.

To check the file system of a hard drive partition on a Windows operating system, use the **Disk Management** tool. To check the file system of a hard drive partition on a Linux operating system, use the **fdisk** tool.

IMPORTANT

You cannot use ISO files on partitions controlled by LVM (Logical Volume Management).

Use this option to install Red Hat Enterprise Linux on systems without a DVD drive or network connection.

Hard drive installations use the following files:

- an *ISO image* of the installation DVD. An ISO image is a file that contains an exact copy of the content of a DVD.
- an **install.img** file extracted from the ISO image.
- optionally, a **product.img** file extracted from the ISO image.

With these files present on a hard drive, you can choose **Hard drive** as the installation source when you boot the installation program (refer to Section 8.3, "Installation Method").

Ensure that you have boot media available on CD, DVD, or a USB storage device such as a flash drive.

To prepare a hard drive as an installation source, follow these steps:

1. Obtain an ISO image of the Red Hat Enterprise Linux installation DVD (refer to Chapter 1, *Obtaining Red Hat Enterprise Linux*). Alternatively, if you have the DVD on physical media, you can create an image of it with the following command on a Linux system:

   ```
   dd if=/dev/dvd of=/path_to_image/name_of_image.iso
   ```

   where *dvd* is your DVD drive device, *name_of_image* is the name you give to the resulting ISO image file, and *path_to_image* is the path to the location on your system where the resulting ISO image will be stored.

2. Transfer the ISO image to the hard drive.

   The ISO image must be located on a hard drive that is either internal to the computer on which you will install Red Hat Enterprise Linux, or on a hard drive that is attached to that computer by USB.

3. Use a SHA256 checksum program to verify that the ISO image that you copied is intact. Many SHA256 checksum programs are available for various operating systems. On a Linux system, run:

   ```
   
   ```

```
$ sha256sum name_of_image.iso
```

where *name_of_image* is the name of the ISO image file. The SHA256 checksum program displays a string of 64 characters called a *hash*. Compare this hash to the hash displayed for this particular image on the **Downloads** page in the Red Hat Customer Portal (refer to Chapter 1, *Obtaining Red Hat Enterprise Linux*). The two hashes should be identical.

4. Copy the **images/** directory from inside the ISO image to the same directory in which you stored the ISO image file itself. Enter the following commands:

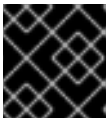
```
mount -t iso9660 /path_to_image/name_of_image.iso /mount_point -o loop,ro
cp -pr /mount_point/images /publicly_available_directory/
umount /mount_point
```

where ***path_to_image*** is the path to the ISO image file, ***name_of_image*** is the name of the ISO image file, and ***mount_point*** is a mount point on which to mount the image while you copy files from the image. For example:

```
mount -t iso9660 /var/isos/RHEL6.iso /mnt/tmp -o loop,ro
cp -pr /mnt/tmp/images /var/isos/
umount /mnt/tmp
```

The ISO image file and an **images/** directory are now present, side-by-side, in the same directory.

5. Verify that the **images/** directory contains at least the **install.img** file, without which installation cannot proceed. Optionally, the **images/** directory should contain the **product.img** file, without which only the packages for a **Minimal** installation will be available during the package group selection stage (refer to Section 9.17, "Package Group Selection").



IMPORTANT

install.img and **product.img** must be the only files in the **images/** directory.

NOTE

anaconda has the ability to test the integrity of the installation media. It works with the DVD, hard drive ISO, and NFS ISO installation methods. We recommend that you test all installation media before starting the installation process, and before reporting any installation-related bugs (many of the bugs reported are actually due to improperly-burned DVDs). To use this test, type the following command at the **boot:** prompt:

```
linux mediacheck
```

CHAPTER 5. SYSTEM SPECIFICATIONS LIST

The most recent list of supported hardware can be found at https://hardware.redhat.com/.

The installation program automatically detects and installs your computer's hardware. Although you should make sure that your hardware meets the minimum requirements to install Red Hat Enterprise Linux (refer to Section 3.2, "Is Your Hardware Compatible?") you do not usually need to supply the installation program with any specific details about your system.

However, when performing certain types of installation, some specific details might be useful or even essential.

- If you plan to use a customized partition layout, record:
  - The model numbers, sizes, types, and interfaces of the hard drives attached to the system. For example, Seagate ST3320613AS 320 GB on SATA0, Western Digital WD7500AAKS 750 GB on SATA1. This will allow you to identify specific hard drives during the partitioning process.
- If you are installing Red Hat Enterprise Linux as an additional operating system on an existing system, record:
  - The mount points of the existing partitions on the system. For example, /**boot** on **sda1**, / on **sda2**, and /**home** on **sdb1**. This will allow you to identify specific partitions during the partitioning process.
- If you plan to install from an image on a local hard drive:
  - The hard drive and directory that holds the image.
- If you plan to install from a network location, or install on an iSCSI target:
  - The make and model numbers of the network adapters on your system. For example, Netgear GA311. This will allow you to identify adapters when manually configuring the network.
  - IP, DHCP, and BOOTP addresses
  - Netmask
  - Gateway IP address
  - One or more name server IP addresses (DNS)

  If any of these networking requirements or terms are unfamiliar to you, contact your network administrator for assistance.
- If you plan to install from a network location:
  - The location of the image on an FTP server, HTTP (web) server, HTTPS (web) server, or NFS server – see Section 8.3.5, "Installing via FTP, HTTP, or HTTPS" and Section 8.3.4, "Installing via NFS" for examples.
- If you plan to install on an iSCSI target:

- The location of the iSCSI target. Depending on your network, you might also need a CHAP username and password, and perhaps a reverse CHAP username and password – see Section 9.6.1.1, " Advanced Storage Options " .

- If you are installing using Intel iSCSI Remote Boot:
  - All attached iSCSI storage devices must be disabled, otherwise the installation will succeed but the installed system will not boot.

- If your computer is part of a domain:
  - You should verify that the domain name will be supplied by the DHCP server. If not, you will need to input the domain name manually during installation.

CHAPTER 6. UPDATING DRIVERS DURING INSTALLATION ON INTEL AND AMD SYSTEMS

In most cases, Red Hat Enterprise Linux already includes drivers for the devices that make up your system. However, if your system contains hardware that has been released very recently, drivers for this hardware might not yet be included. Sometimes, a driver update that provides support for a new device might be available from Red Hat or your hardware vendor on a *driver disc* that contains *rpm packages*. Typically, the driver disc is available for download as an *ISO image file*.

Often, you do not need the new hardware during the installation process. For example, if you use a DVD to install to a local hard drive, the installation will succeed even if drivers for your network card are not available. In situations like this, complete the installation and add support for the piece of hardware afterward – refer to Section 35.1.1, "Driver Update rpm Packages" for details of adding this support.

In other situations, you might want to add drivers for a device during the installation process to support a particular configuration. For example, you might want to install drivers for a network device or a storage adapter card to give the installer access to the storage devices that your system uses. You can use a driver disc to add this support during installation in one of two ways:

1. place the ISO image file of the driver disc in a location accessible to the installer:
    1. on a local hard drive
    2. a USB flash drive
2. create a driver disc by extracting the image file onto:
    1. a CD
    2. a DVD

    Refer to the instructions for making installation discs in Section 2.1, "Making an Installation DVD" for more information on burning ISO image files to CD or DVD.

If Red Hat, your hardware vendor, or a trusted third party told you that you will require a driver update during the installation process, choose a method to supply the update from the methods described in this chapter and test it before beginning the installation. Conversely, do not perform a driver update during installation unless you are certain that your system requires it. Although installing an unnecessary driver update will not cause harm, the presence of a driver on a system for which it was not intended can complicate support.

6.1. LIMITATIONS OF DRIVER UPDATES DURING INSTALLATION

Unfortunately, some situations persist in which you cannot use a driver update to provide drivers during installation:

Devices already in use

You cannot use a driver update to replace drivers that the installation program has already loaded. Instead, you must complete the installation with the drivers that the installation program loaded and update to the new drivers after installation, or, if you need the new drivers for the installation process, consider performing an initial RAM disk driver update – refer to Section 6.2.3, "Preparing an Initial RAM Disk Update".

Devices with an equivalent device available

Because all devices of the same type are initialized together, you cannot update drivers for a device

if the installation program has loaded drivers for a similar device. For example, consider a system that has two different network adapters, one of which has a driver update available. The installation program will initialize both adapters at the same time, and therefore, you will not be able to use this driver update. Again, complete the installation with the drivers loaded by the installation program and update to the new drivers after installation, or use an initial RAM disk driver update.

6.2. PREPARING FOR A DRIVER UPDATE DURING INSTALLATION

If a driver update is necessary and available for your hardware, Red Hat or a trusted third party such as the hardware vendor will typically provide it in the form of an image file in ISO format. Some methods of performing a driver update require you to make the image file available to the installation program, while others require you to use the image file to make a driver update disk:

Methods that use the image file itself

- local hard drive
- USB flash drive

Methods that use a driver update disk produced from an image file

- CD
- DVD

Choose a method to provide the driver update, and refer to Section 6.2.1, "Preparing to Use a Driver Update Image File", Section 6.2.2, "Preparing a Driver Disc" or Section 6.2.3, "Preparing an Initial RAM Disk Update". Note that you can use a USB storage device either to provide an image file, or as a driver update disk.

6.2.1. Preparing to Use a Driver Update Image File

6.2.1.1. Preparing to use an image file on local storage

To make the ISO image file available on local storage, such as a hard drive or USB flash drive, you must first determine whether you want to install the updates automatically or select them manually.

For manual installations, copy the file onto the storage device. You can rename the file if you find it helpful to do so, but you must not change the filename extension, which must remain **.iso**. In the following example, the file is named **dd.iso**:

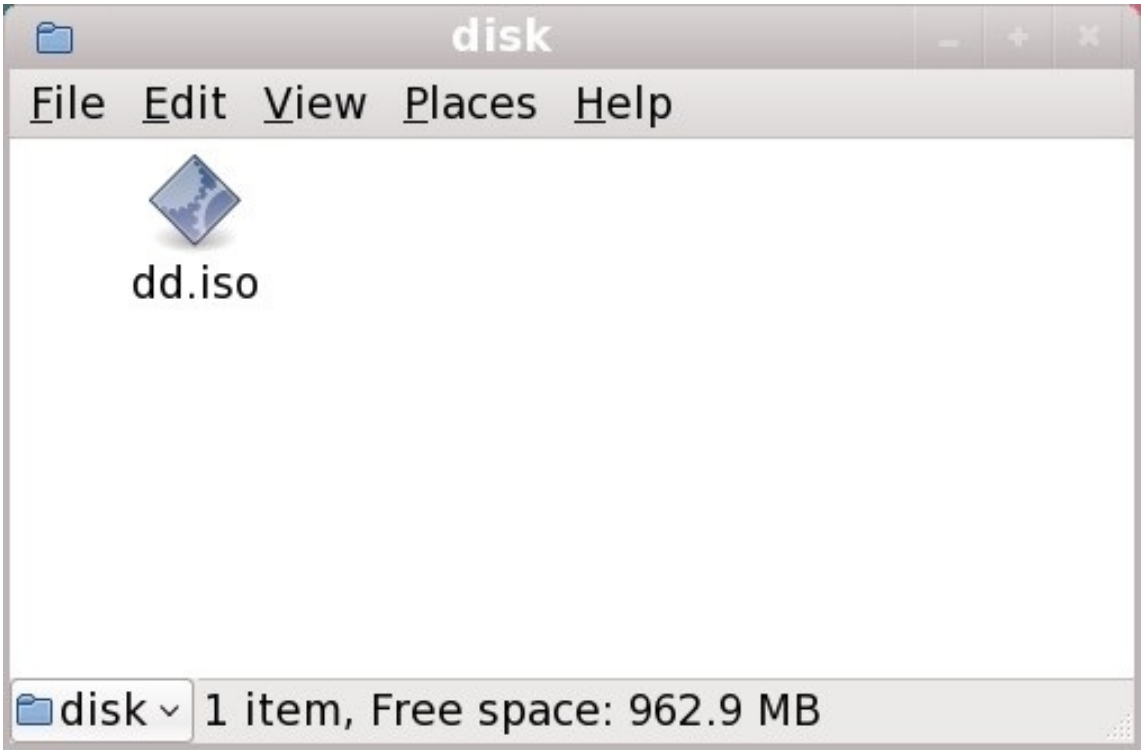


Figure 6.1. Content of a USB flash drive holding a driver update image file

Note that if you use this method, the storage device will contain only a single file. This differs from driver discs on formats such as CD and DVD, which contain many files. The ISO image file contains all of the files that would normally be on a driver disc.

Refer to Section 6.3.2, "Let the Installer Prompt You for a Driver Update" and Section 6.3.3, "Use a Boot Option to Specify a Driver Update Disk" to learn how to select the driver update manually during installation.

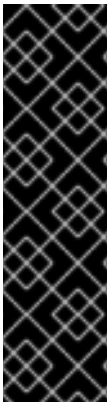
For automatic installations, you will need to extract the ISO to the root directory of the storage device rather than copy it. Copying the ISO is only effective for manual installations. You must also change the file system label of the device to **OEMDRV**.

The installation program will then automatically examine the extracted ISO for driver updates and load any that it detects. This behavior is controlled by the **dlabel=on** boot option, which is enabled by default. Refer to Section 6.3.1, "Let the Installer Find a Driver Update Disk Automatically" .

6.2.2. Preparing a Driver Disc

You can create a driver update disc on CD or DVD.

6.2.2.1. Creating a driver update disc on CD or DVD



IMPORTANT

CD/DVD Creator is part of the GNOME desktop. If you use a different Linux desktop, or a different operating system altogether, you will need to use another piece of software to create the CD or DVD. The steps will be generally similar.

Make sure that the software that you choose can create CDs or DVDs from image files. While this is true of most CD and DVD burning software, exceptions exist. Look for a button or menu entry labeled **burn from image** or similar. If your software lacks this feature, or you do not select it, the resulting disc will hold only the image file itself, instead of the contents of the image file.

1. Use the desktop file manager to locate the ISO image file of the driver disc, supplied to you by Red Hat or your hardware vendor.

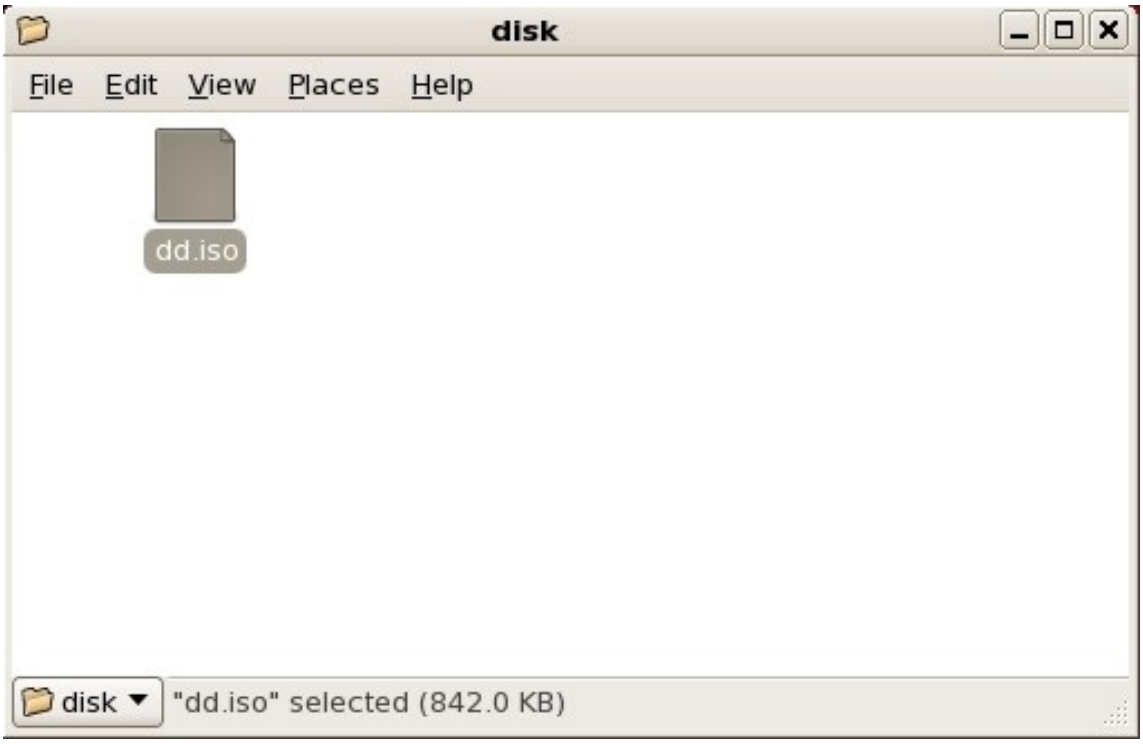


Figure 6.2. A typical .iso file displayed in a file manager window

2. Right-click on this file and choose **Write to disc**. You will see a window similar to the following:

Figure 6.3. CD/DVD Creator's Write to Disc dialog

3. Click the **Write** button. If a blank disc is not already in the drive, **CD/DVD Creator** will prompt you to insert one.

After you burn a driver update disc CD or DVD, verify that the disc was created successfully by inserting it into your system and browsing to it using the file manager. You should see a single file named **rhdd3** and a directory named **rpms**:

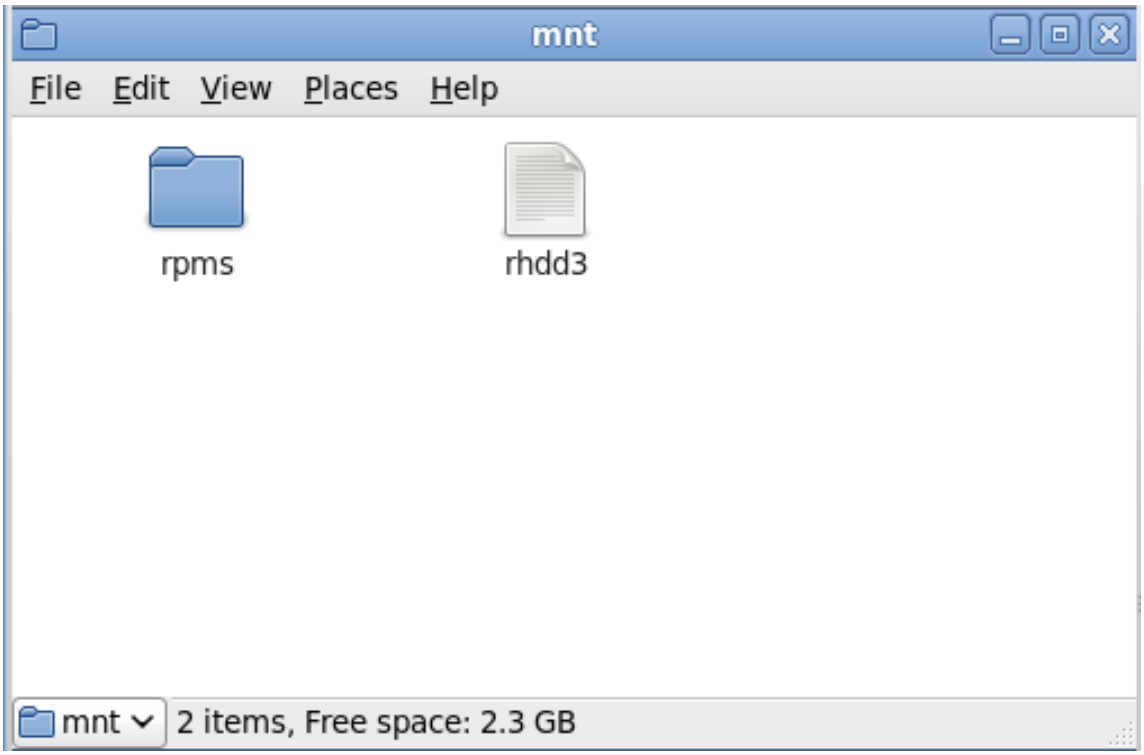


Figure 6.4. Contents of a typical driver update disc on CD or DVD

If you see only a single file ending in **.iso**, then you have not created the disc correctly and should try again. Ensure that you choose an option similar to **burn from image** if you use a Linux desktop other than GNOME or if you use a different operating system.

Refer to Section 6.3.2, "Let the Installer Prompt You for a Driver Update" and Section 6.3.3, "Use a Boot Option to Specify a Driver Update Disk" to learn how to use the driver update disc during installation.

6.2.3. Preparing an Initial RAM Disk Update

IMPORTANT

This is an advanced procedure that you should consider only if you cannot perform a driver update with any other method.

The Red Hat Enterprise Linux installation program can load updates for itself early in the installation process from a *RAM disk* — an area of your computer's memory that temporarily behaves as if it were a disk. You can use this same capability to load driver updates. To perform a driver update during installation, your computer must be able to boot from a *preboot execution environment* (PXE) server, and you must have a PXE server available on your network. Refer to Chapter 30, *Setting Up an Installation Server* for instructions on using PXE during installation.

To make the driver update available on your PXE server:

1. Place the driver update image file on your installation server. Usually, you would do this by downloading it to the server from a location on the Internet specified by Red Hat or your hardware vendor. Names of driver update image files end in **.iso**.

2. Copy the driver update image file into the **/tmp/initrd_update** directory.

3. Rename the driver update image file to **dd.img**.

4. At the command line, change into the **/tmp/initrd_update** directory, type the following command, and press **Enter**:

```
find . | cpio --quiet -o -H newc | gzip -9 >/tmp/initrd_update.img
```

5. Copy the file **/tmp/initrd_update.img** into the directory the holds the target that you want to use for installation. This directory is placed under the **/var/lib/tftpboot/pxelinux/** directory. For example, **/var/lib/tftpboot/pxelinux/rhel6/** might hold the PXE target for Red Hat Enterprise Linux 6.

6. Edit the **/var/lib/tftpboot/pxelinux/pxelinux.cfg/default** file to include an entry that includes the initial RAM disk update that you just created, in the following format:

```
label target-dd
kernel target/vmlinuz
append initrd=target/initrd.img,target/dd.img
```

   Where *target* is the target that you want to use for installation.

Refer to Section 6.3.4, "Select a PXE Target that Includes a Driver Update" to learn how to use an initial RAM disk update during installation.

Example 6.1. Preparing an initial RAM disk update from a driver update image file

In this example, **driver_update.iso** is a driver update image file that you downloaded from the Internet to a directory on your PXE server. The target that you want to PXE boot from is located in **/var/lib/tftpboot/pxelinux/rhel6/**

At the command line, change to the directory that holds the file and enter the following commands:

```
$ cp driver_update.iso /tmp/initrd_update/dd.img
$ cd /tmp/initrd_update
$ find . | cpio --quiet -c -o -H newc | gzip -9 >/tmp/initrd_update.img
$ cp /tmp/initrd_update.img /var/lib/tftpboot/pxelinux/rhel6/dd.img
```

Edit the **/var/lib/tftpboot/pxelinux/pxelinux.cfg/default** file and include the following entry:

```
label rhel6-dd
kernel rhel6/vmlinuz
append initrd=rhe6/initrd.img,rhel6/dd.img
```

6.3. PERFORMING A DRIVER UPDATE DURING INSTALLATION

You can perform a driver update during installation in the following ways:

- let the installer automatically find a driver update disk.
- let the installer prompt you for a driver update.
- use a boot option to specify a driver update disk.

6.3.1. Let the Installer Find a Driver Update Disk Automatically

Attach a block device with the filesystem label **OEMDRV** before starting the installation process. The installer will automatically examine the device and load any driver updates that it detects and will not prompt you during the process. Refer to Section 6.2.1.1, “Preparing to use an image file on local storage” to prepare a storage device for the installer to find.

6.3.2. Let the Installer Prompt You for a Driver Update

1. Begin the installation normally for whatever method you have chosen. If the installer cannot load drivers for a piece of hardware that is essential for the installation process (for example, if it cannot detect any network or storage controllers), it prompts you to insert a driver update disk:

Figure 6.5. The no driver found dialog

2. Select **Use a driver disk** and refer to Section 6.4, "Specifying the Location of a Driver Update Image File or a Driver Update Disk".

6.3.3. Use a Boot Option to Specify a Driver Update Disk

IMPORTANT

This method only works to introduce completely new drivers, not to update existing drivers.

1. Type **linux dd** at the boot prompt at the start of the installation process and press **Enter**. The installer prompts you to confirm that you have a driver disk:

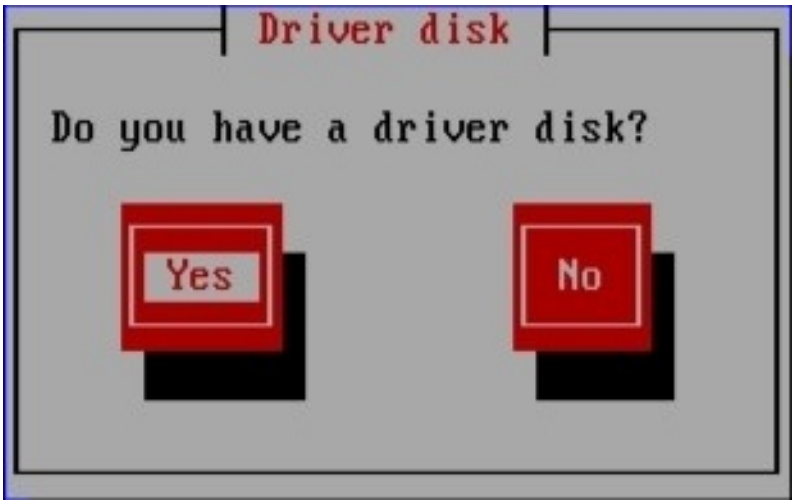


Figure 6.6. The driver disk prompt

2. Insert the driver update disk that you created on CD, DVD, or USB flash drive and select **Yes**. The installer examines the storage devices that it can detect. If there is only one possible location that could hold a driver disk (for example, the installer detects the presence of a DVD drive, but no other storage devices) it will automatically load any driver updates that it finds at this location.

   If the installer finds more than one location that could hold a driver update, it prompts you to specify the location of the update. See Section 6.4, "Specifying the Location of a Driver Update Image File or a Driver Update Disk".

6.3.4. Select a PXE Target that Includes a Driver Update

1. Select **network boot** in your computer's BIOS or boot menu. The procedure to specify this option varies widely among different computers. Consult your hardware documentation or the hardware vendor for specifics relevant to your computer.

2. In the *preboot execution environment* (PXE), choose the boot target that you prepared on your PXE server. For example, if you labeled this environment **rhel6-dd** in the **/var/lib/tftpboot/pxelinux/pxelinux.cfg/default** file on your PXE server, type **rhel6-dd** at the prompt and press **Enter**.

Refer to Section 6.2.3, "Preparing an Initial RAM Disk Update" and Chapter 30, *Setting Up an Installation Server* for instructions on using PXE to perform an update during installation. Note that this is an advanced procedure – do not attempt it unless other methods of performing a driver update fail.

6.4. SPECIFYING THE LOCATION OF A DRIVER UPDATE IMAGE FILE OR A DRIVER UPDATE DISK

If the installer detects more than one possible device that could hold a driver update, it prompts you to select the correct device. If you are not sure which option represents the device on which the driver update is stored, try the various options in order until you find the correct one.

Figure 6.7. Selecting a driver disk source

If the device that you choose contains no suitable update media, the installer will prompt you to make another choice.

If you made a driver update disk on CD, DVD, or USB flash drive, the installer now loads the driver update. However, if the device that you selected is a type of device that could contain more than one partition (whether the device currently has more than one partition or not), the installer might prompt you to select the partition that holds the driver update.

Figure 6.8. Selecting a driver disk partition

The installer prompts you to specify which file contains the driver update:

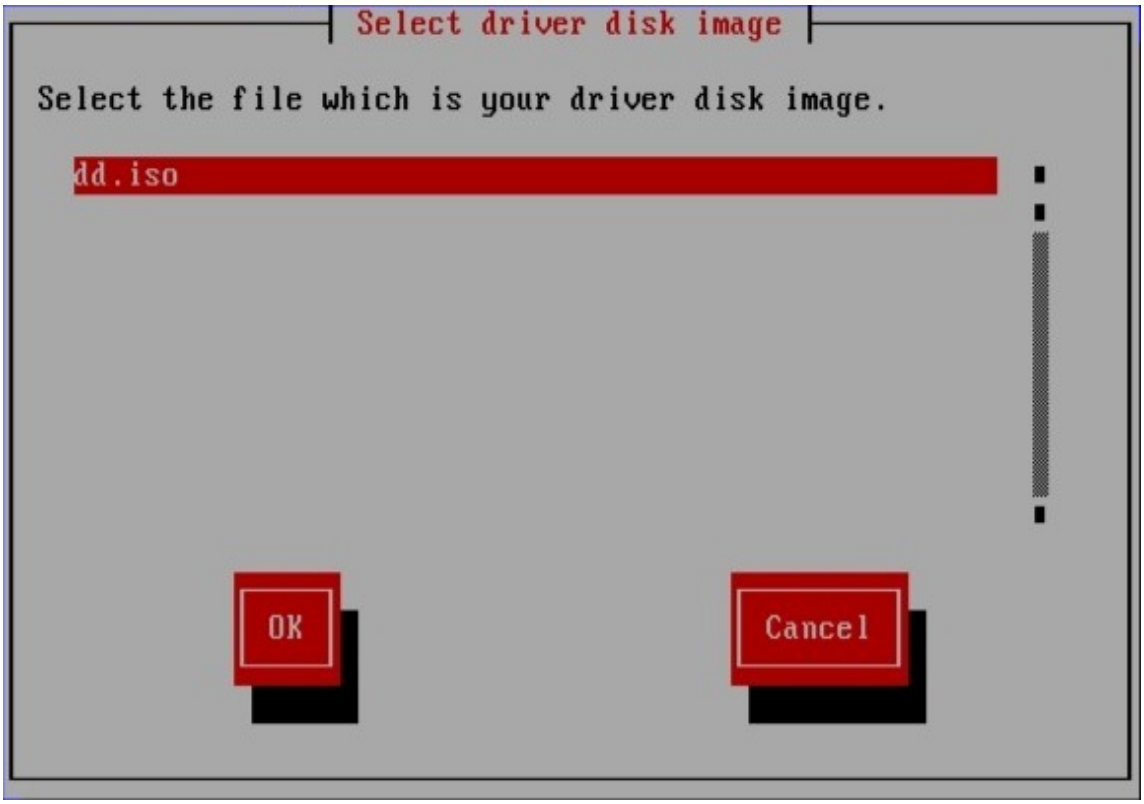


Figure 6.9. Selecting an ISO image

Expect to see these screens if you stored the driver update on an internal hard drive or on a USB storage device. You should not see them if the driver update is on a CD or DVD.

Regardless of whether you are providing a driver update in the form of an image file or with a driver update disk, the installer now copies the appropriate update files into a temporary storage area (located in system RAM and not on disk). The installer might ask whether you would like to use additional driver updates. If you select **Yes**, you can load additional updates in turn. When you have no further driver

updates to load, select **No**. If you stored the driver update on removable media, you can now safely eject or disconnect the disk or device. The installer no longer requires the driver update, and you can re-use the media for other purposes.

CHAPTER 7. BOOTING THE INSTALLER

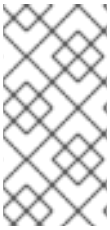
7.1. STARTING THE INSTALLATION PROGRAM

IMPORTANT

Red Hat Enterprise Linux 6.9 does not support UEFI for 32-bit x86 systems.

On 64-bit systems, boot configurations of UEFI and BIOS differ significantly from each other. Therefore, the installed system must boot using the same firmware that was used during installation. You cannot install the operating system on a system that uses BIOS and then boot this installation on a system that uses UEFI.

To start, first make sure that you have all necessary resources for the installation. If you have already read through Chapter 3, *Planning for Installation on the x86 Architecture* , and followed the instructions, you should be ready to start the installation process. When you have verified that you are ready to begin, boot the installation program using the Red Hat Enterprise Linux DVD or any boot media that you have created.



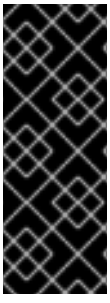
NOTE

Occasionally, some hardware components require a *driver update* during the installation. A driver update adds support for hardware that is not otherwise supported by the installation program. Refer to Chapter 6, *Updating Drivers During Installation on Intel and AMD Systems* for more information.

7.1.1. Booting the Installation Program on x86, AMD64, and Intel 64 Systems

You can boot the installation program using any one of the following media (depending upon what your system can support):

- *Red Hat Enterprise Linux DVD* — Your machine supports a bootable DVD drive and you have the Red Hat Enterprise Linux installation DVD.
- *Boot CD-ROM* — Your machine supports a bootable CD-ROM drive and you want to perform network or hard drive installation.
- *USB flash drive* — Your machine supports booting from a USB device.
- *PXE boot via network* — Your machine supports booting from the network. This is an advanced installation path. Refer to Chapter 30, *Setting Up an Installation Server* for additional information on this method.



IMPORTANT

Red Hat Enterprise Linux 6.9 does not support UEFI for 32-bit x86 systems.

On 64-bit systems, boot configurations of UEFI and BIOS differ significantly from each other. Therefore, the installed system must boot using the same firmware that was used during installation. You cannot install the operating system on a system that uses BIOS and then boot this installation on a system that uses UEFI.

To start the installation program from a Red Hat Enterprise Linux DVD or from minimal boot media, follow this procedure:

1. Disconnect any external FireWire or USB disks that you do not need for installation. Refer to Section 3.4.3, "FireWire and USB Disks" for more information.
2. Power on your computer system.
3. Insert the media in your computer.
4. Power off your computer with the boot media still inside.
5. Power on your computer system.

To create a boot CD-ROM or to prepare your USB flash drive for booting or installation, refer to Section 2.2, "Making Minimal Boot Media".

Insert the boot media and reboot the system.

You might need to press a specific key or combination of keys to boot from the media. On most computers, a message appears briefly on the screen very soon after you turn on the computer. Typically, it is worded something like **Press F10 to select boot device**, although the specific wording and the key that you must press varies widely from computer to computer. Consult the documentation for your computer or motherboard, or seek support from the hardware manufacturer or vendor.

If your computer does not allow you to select a boot device as it starts up, you might need to configure your system's *Basic Input/Output System* (BIOS) to boot from the media.

To change your BIOS settings on an x86, AMD64, or Intel 64 system, watch the instructions provided on your display when your computer first boots. A line of text appears, telling you which key to press to enter the BIOS settings.

Once you have entered your BIOS setup program, find the section where you can alter your boot sequence. The default is often C, A or A, C (depending on whether you boot from your hard drive [C] or a diskette drive [A]). Change this sequence so that the DVD is first in your boot order and that C or A (whichever is your typical boot default) is second. This instructs the computer to first look at the DVD drive for bootable media; if it does not find bootable media on the DVD drive, it then checks your hard drive or diskette drive.

Save your changes before exiting the BIOS. For more information, refer to the documentation that came with your system.

After a short delay, the graphical boot screen appears, which contains information on a variety of boot options. Installation program automatically begins if you take no action within the first minute. For a description of the options available on this screen, refer to Section 7.1.2, "The Boot Menu".

Alternatively, press the **Esc** key to access the **boot:** prompt, at which you can enter additional boot options as described in Section 7.1.3, "Additional Boot Options".

IMPORTANT

Excessive input (e.g. clicking the mouse repeatedly) during the boot sequence may cause the installer to ignore keyboard input later in the installation process.

7.1.2. The Boot Menu

The boot media displays a graphical boot menu with several options. If no key is hit within 60 seconds, the default boot option runs. To choose the default, either wait for the timer to run out or hit **Enter** on the keyboard. To select a different option than the default, use the arrow keys on your keyboard, and hit **Enter** when the correct option is highlighted. If you want to customize the boot options for a particular option, press the **Tab** key. To access the **boot:** prompt at which you can specify custom boot options, press the **Esc** key and refer to Section 7.1.3, "Additional Boot Options".

Figure 7.1. The boot screen

For a listing and explanation of common boot options, refer to Chapter 28, *Boot Options*.

The boot menu options are:

Install or upgrade an existing system

This option is the default. Choose this option to install Red Hat Enterprise Linux onto your computer system using the graphical installation program.

Install system with basic video driver

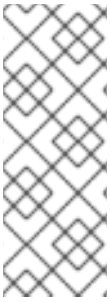
This option allows you to install Red Hat Enterprise Linux in graphical mode even if the installation program is unable to load the correct driver for your video card. If your screen appears distorted or goes blank when using the **Install or upgrade an existing system** option, restart your computer and try this option instead.

Rescue installed system

Choose this option to repair a problem with your installed Red Hat Enterprise Linux system that prevents you from booting normally. Although Red Hat Enterprise Linux is an exceptionally stable computing platform, it is still possible for occasional problems to occur that prevent booting. The rescue environment contains utility programs that allow you fix a wide variety of these problems.

Boot from local drive

This option boots the system from the first installed disk. If you booted this disc accidentally, use this option to boot from the hard disk immediately without starting the installer.



NOTE

To abort the installation, either press **Ctrl**+**Alt**+**Del** or power off your computer with the power switch. You may abort the installation process without consequence at any time prior to selecting **Write changes to disk** on the **Write partitioning to disk** screen. Red Hat Enterprise Linux makes no permanent changes to your computer until that point. Please be aware that stopping the installation after partitioning has begun can leave your computer unusable.

7.1.3. Additional Boot Options

While it is easiest to boot using a DVD and perform a graphical installation, sometimes there are installation scenarios where booting in a different manner may be needed. This section discusses additional boot options available for Red Hat Enterprise Linux.

To pass options to the boot loader on an x86, AMD64, or Intel 64 system, press the **Esc** key at boot time. The **boot:** prompt appears, at which you can use the boot loader options described below.

NOTE

Refer to Chapter 28, *Boot Options* for additional boot options not covered in this section.

- To perform a text mode installation, at the installation boot prompt, type:

```
linux text
```

- To specify an installation source, use the **linux repo=** option. For example:

```
linux repo=cdrom:device
```

```
linux repo=ftp://username:password@URL
```

```
linux repo=http://URL
```

```
linux repo=hd:device
```

```
linux repo=nfs:options:server:/path
```

```
linux repo=nfsiso:options:server:/path
```

  In these examples, **cdrom** refers to a CD or DVD drive, **ftp** refers to a location accessible by FTP, **http** refers to a location accessible by HTTP, **hd** refers to an ISO image file accessible on a hard drive partition, **nfs** refers to an expanded tree of installation files accessible by NFS, and **nfsiso** refers to an ISO image file accessible by NFS.

- ISO images have an SHA256 checksum embedded in them. To test the checksum integrity of an ISO image, at the installation boot prompt, type:

```
linux mediacheck
```

The installation program prompts you to insert a DVD or select an ISO image to test, and select **OK** to perform the checksum operation. This checksum operation can be performed on any Red Hat Enterprise Linux DVD. It is strongly recommended to perform this operation on any Red Hat Enterprise Linux DVD that was created from downloaded ISO images. This command works with the DVD, hard drive ISO, and NFS ISO installation methods.

- If you need to perform the installation in *serial mode*, type the following command:

```
linux console=<device>
```

For text mode installations, use:

```
linux text console=<device>
```

In the above command, *<device>* should be the device you are using (such as ttyS0 or ttyS1). For example, **linux text console=ttyS0**.

Text mode installations using a serial terminal work best when the terminal supports UTF-8. Under UNIX and Linux, Kermit supports UTF-8. For Windows, Kermit '95 works well. Non-UTF-8 capable terminals works as long as only English is used during the installation process. An enhanced serial display can be used by passing the **utf8** command as a boot-time option to the installation program. For example:

```
linux console=ttyS0 utf8
```

7.1.3.1. Kernel Options

Options can also be passed to the kernel. For example, to apply updates for the anaconda installation program from a USB storage device enter:

```
linux updates
```

For text mode installations, use:

```
linux text updates
```

This command results in a prompt for the path to the device that contains updates for **anaconda**. It is not needed if you are performing a network installation and have already placed the updates image contents in **rhupdates/** on the server.

After entering any options, press **Enter** to boot using those options.

If you need to specify boot options to identify your hardware, please write them down. The boot options are needed during the boot loader configuration portion of the installation (refer to Section 9.18, “x86, AMD64, and Intel 64 Boot Loader Configuration” for more information).

For more information on kernel options refer to Chapter 28, *Boot Options*.

7.2. INSTALLING FROM A DIFFERENT SOURCE

You can install Red Hat Enterprise Linux from the ISO images stored on hard disk, or from a network using NFS, FTP, HTTP, or HTTPS methods. Experienced users frequently use one of these methods because it is often faster to read data from a hard disk or network server than from a DVD.

The following table summarizes the different boot methods and recommended installation methods to use with each:

Table 7.1. Boot methods and installation sources

Boot method	Installation source
Installation DVD	DVD, network, or hard disk
Installation USB flash drive	Installation DVD, network, or hard disk
Minimal boot CD or USB, rescue CD	Network or hard disk

Refer to Section 3.7, "Selecting an Installation Method" for information about installing from locations other than the media with which you booted the system.

7.3. BOOTING FROM THE NETWORK USING PXE

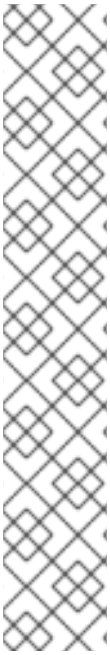
To boot with PXE, you need a properly configured server, and a network interface in your computer that supports PXE. For information on how to configure a PXE server, refer to Chapter 30, *Setting Up an Installation Server*.

Configure the computer to boot from the network interface. This option is in the BIOS, and may be labeled **Network Boot** or **Boot Services**. Once you properly configure PXE booting, the computer can boot the Red Hat Enterprise Linux installation system without any other media.

To boot a computer from a PXE server:

1. Ensure 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 computer.
3. A menu screen appears. Press the number key that corresponds to the desired option.

If your PC does not boot from the netboot server, ensure that the BIOS is configured to boot first from the correct network interface. Some BIOS systems specify the network interface as a possible boot device, but do not support the PXE standard. Refer to your hardware documentation for more information.



NOTE

Some servers with multiple network interfaces might not assign eth0 to the first network interface as the firmware interface knows it, which can cause the installer to try to use a different network interface from the one that was used by PXE. To change this behavior, use the following in **pxelinux.cfg/*** config files:

```
IPAPPEND 2
APPEND ksdevice=bootif
```

These configuration options above cause the installer to use the same network interface the firmware interface and PXE use. You can also use the following option:

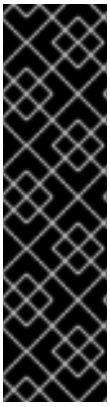
```
ksdevice=link
```

This option causes the installer to use the first network device it finds that is linked to a network switch.

CHAPTER 8. CONFIGURING LANGUAGE AND INSTALLATION SOURCE

Before the graphical installation program starts, you need to configure the language and installation source.

8.1. THE TEXT MODE INSTALLATION PROGRAM USER INTERFACE



IMPORTANT

We recommend that you install Red Hat Enterprise Linux using the graphical interface. If you are installing Red Hat Enterprise Linux on a system that lacks a graphical display, consider performing the installation over a VNC connection – see Chapter 31, *Installing Through VNC*. If **anaconda** detects that you are installing in text mode on a system where installation over a VNC connection might be possible, **anaconda** asks you to verify your decision to install in text mode even though your options during installation are limited.

If your system has a graphical display, but graphical installation fails, try booting with the **xdriver=vesa** option – refer to Chapter 28, *Boot Options*

Both the loader and later **anaconda** use a screen-based interface that includes most of the on-screen *widgets* commonly found on graphical user interfaces. Figure 8.1, "Installation Program Widgets as seen in **URL Setup**", and Figure 8.2, "Installation Program Widgets as seen in **Choose a Language**", illustrate widgets that appear on screens during the installation process.

NOTE

Not every language supported in graphical installation mode is also supported in text mode. Specifically, languages written with a character set other than the Latin or Cyrillic alphabets are not available in text mode. If you choose a language written with a character set that is not supported in text mode, the installation program will present you with the English versions of the screens.

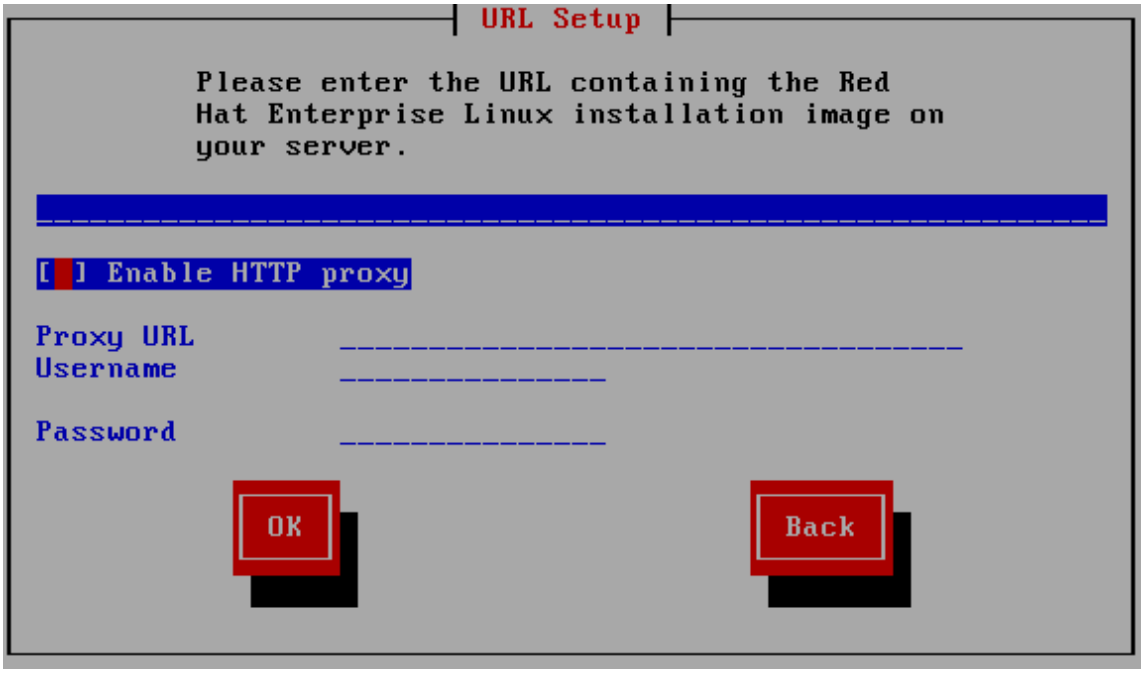


Figure 8.1. Installation Program Widgets as seen in **URL Setup**

Figure 8.2. Installation Program Widgets as seen in **Choose a Language**

The widgets include:

- Window — Windows (usually referred to as *dialogs* in this manual) appear on your screen throughout the installation process. At times, one window may overlay another; in these cases, you can only interact with the window on top. When you are finished in that window, it disappears, allowing you to continue working in the window underneath.

- Checkbox — Checkboxes allow you to select or deselect a feature. The box displays either an asterisk (selected) or a space (unselected). When the cursor is within a checkbox, press **Space** to select or deselect a feature.

- Text Input — Text input lines are regions where you can enter information required by the installation program. When the cursor rests on a text input line, you may enter and/or edit information on that line.

- Text Widget — Text widgets are regions of the screen for the display of text. At times, text widgets may also contain other widgets, such as checkboxes. If a text widget contains more information than can be displayed in the space reserved for it, a scroll bar appears; if you position the cursor within the text widget, you can then use the **Up** and **Down** arrow keys to scroll through all the information available. Your current position is shown on the scroll bar by a **#** character, which moves up and down the scroll bar as you scroll.

- Scroll Bar — Scroll bars appear on the side or bottom of a window to control which part of a list or document is currently in the window's frame. The scroll bar makes it easy to move to any part of a file.

- Button Widget — Button widgets are the primary method of interacting with the installation program. You progress through the windows of the installation program by navigating these buttons, using the **Tab** and **Enter** keys. Buttons can be selected when they are highlighted.

- Cursor — Although not a widget, the cursor is used to select (and interact with) a particular widget. As the cursor is moved from widget to widget, it may cause the widget to change color, or the cursor itself may only appear positioned in or next to the widget. In Figure 8.1, "Installation Program Widgets as seen in **URL Setup**", the cursor is positioned on the **Enable HTTP proxy** checkbox. Figure 8.2, "Installation Program Widgets as seen in **Choose a Language**", shows the cursor on the **OK** button.

8.1.1. Using the Keyboard to Navigate

Navigation through the installation dialogs is performed through a simple set of keystrokes. To move the cursor, use the **Left**, **Right**, **Up**, and **Down** arrow keys. Use **Tab**, and **Shift**-**Tab** to cycle forward or backward through each widget on the screen. Along the bottom, most screens display a summary of available cursor positioning keys.

To "press" a button, position the cursor over the button (using **Tab**, for example) and press **Space** or **Enter**. To select an item from a list of items, move the cursor to the item you wish to select and press **Enter**. To select an item with a checkbox, move the cursor to the checkbox and press **Space** to select an item. To deselect, press **Space** a second time.

Pressing **F12** accepts the current values and proceeds to the next dialog; it is equivalent to pressing the **OK** button.

WARNING

Unless a dialog box is waiting for your input, do not press any keys during the installation process (doing so may result in unpredictable behavior).

8.2. LANGUAGE SELECTION

Use the arrow keys on your keyboard to select a language to use during the installation process (refer to Figure 8.3, "Language Selection"). With your selected language highlighted, press the **Tab** key to move to the **OK** button and press the **Enter** key to confirm your choice.

The language you select here will become the default language for the operating system once it is installed. Selecting the appropriate language also helps target your time zone configuration later in the installation. The installation program tries to define the appropriate time zone based on what you specify on this screen.

To add support for additional languages, customize the installation at the package selection stage. For more information, refer to Section 9.17.2, " Customizing the Software Selection " .

Figure 8.3. Language Selection

Once you select the appropriate language, click **Next** to continue.

8.3. INSTALLATION METHOD

If you booted the installation from minimal boot media or with the **askmethod** boot option, use the arrow keys on your keyboard to select an installation method (refer to Figure 8.4, "Installation Method"). With your selected method highlighted, press the **Tab** key to move to the **OK** button and press the **Enter** key to confirm your choice.

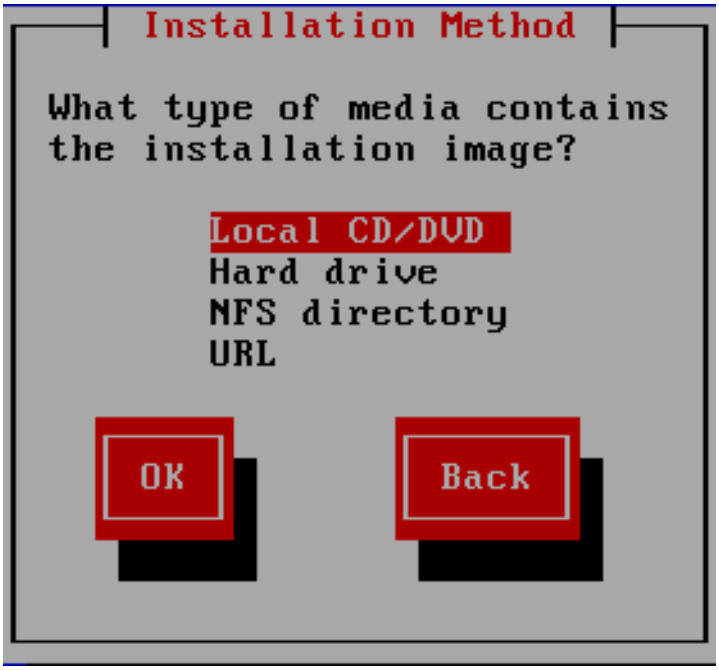


Figure 8.4. Installation Method

8.3.1. Installing from a DVD

To install Red Hat Enterprise Linux from a DVD, place the DVD your DVD drive and boot your system from the DVD. Even if you booted from alternative media, you can still install Red Hat Enterprise Linux from DVD media.

The installation program then probes your system and attempts to identify your DVD drive. It starts by looking for an IDE (also known as an ATAPI) DVD drive.

NOTE

To abort the installation process at this time, reboot your machine and then eject the boot media. You can safely cancel the installation at any point before the **Write changes to disk** screen. Refer to Section 9.16, "Write Changes to Disk" for more information.

If your DVD drive is not detected, and it is a SCSI DVD, the installation program prompts you to choose a SCSI driver. Choose the driver that most closely resembles your adapter. You may specify options for the driver if necessary; however, most drivers detect your SCSI adapter automatically.

If the DVD drive is found and the driver loaded, the installer will present you with the option to perform a media check on the DVD. This will take some time, and you may opt to skip over this step. However, if you later encounter problems with the installer, you should reboot and perform the media check before calling for support. From the media check dialog, continue to the next stage of the installation process (refer to Section 9.3, "Welcome to Red Hat Enterprise Linux").

8.3.2. Installing from a Hard Drive

The **Select Partition** screen applies only if you are installing from a disk partition (that is, you selected **Hard Drive** in the **Installation Method** dialog). This dialog allows you to name the disk partition and directory from which you are installing Red Hat Enterprise Linux. If you used the **repo=hd** boot option, you already specified a partition.

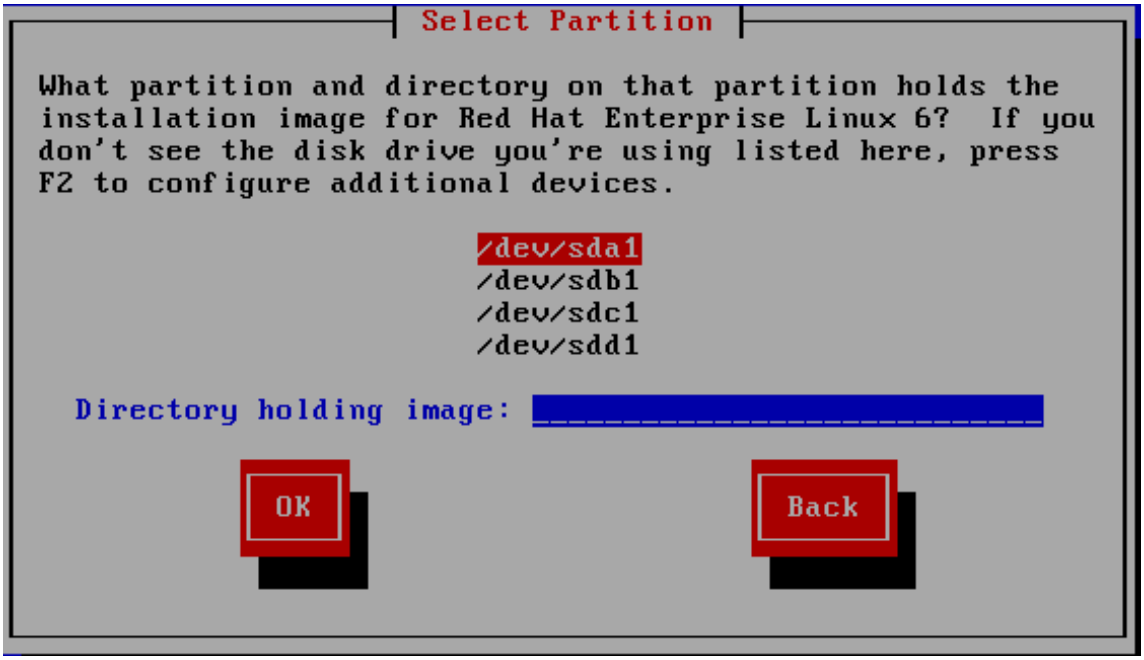


Figure 8.5. Selecting Partition Dialog for Hard Drive Installation

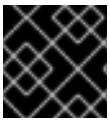
Select the partition containing the ISO files from the list of available partitions. Internal IDE, SATA, SCSI, and USB drive device names begin with **/dev/sd**. Each individual drive has its own letter, for example **/dev/sda**. Each partition on a drive is numbered, for example **/dev/sda1**.

Also specify the **Directory holding images**. Enter the full directory path from the drive that contains the ISO image files. The following table shows some examples of how to enter this information:

Table 8.1. Location of ISO images for different partition types

Partition type	Volume	Original path to files	Directory to use
VFAT	D:\	D:\Downloads\RHEL6.9	/Downloads/RHEL6.9
ext2, ext3, ext4	/home	/home/user1/RHEL6.9	/user1/RHEL6.9

If the ISO images are in the root (top-level) directory of a partition, enter a /. If the ISO images are located in a subdirectory of a mounted partition, enter the name of the directory holding the ISO images within that partition. For example, if the partition on which the ISO images is normally mounted as **/home/**, and the images are in **/home/new/**, you would enter **/new/**.



IMPORTANT

An entry without a leading slash may cause the installation to fail.

Select **OK** to continue. Proceed with Chapter 9, *Installing Using Anaconda*.

8.3.3. Performing a Network Installation

When you start an installation with the **askmethod** or **repo=** options, you can install Red Hat Enterprise Linux from a network server using FTP, HTTP, HTTPS, or NFS protocols. **Anaconda** uses the same network connection to consult additional software repositories later in the installation process.

If your system has more than one network device, **anaconda** presents you with a list of all available devices and prompts you to select one to use during installation. If your system only has a single network device, **anaconda** automatically selects it and does not present this dialog.

Figure 8.6. Networking Device

If you are not sure which device in the list corresponds to which physical socket on the system, select a device in the list then press the **Identify** button. The **Identify NIC** dialog appears.

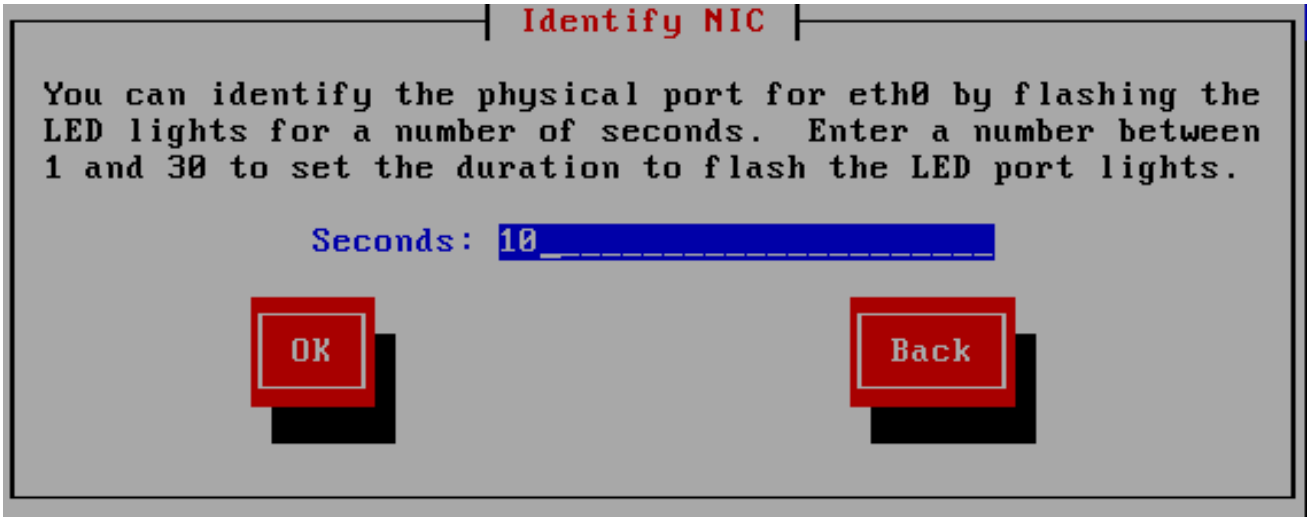


Figure 8.7. Identify NIC

The sockets of most network devices feature an *activity light* (also called a *link light*) – an LED that flashes to indicate that data is flowing through the socket. **Anaconda** can flash the activity light of the network device that you selected in the **Networking Device** dialog for up to 30 seconds. Enter the number of seconds that you require, then press **OK**. When **anaconda** finishes flashing the light, it returns you to the **Networking Device** dialog.

When you select a network device, **anaconda** prompts you to choose how to configure TCP/IP:

IPv4 options

Dynamic IP configuration (DHCP)

Anaconda uses DHCP running on the network to supply the network configuration automatically.

Manual configuration

Anaconda prompts you to enter the network configuration manually, including the IP address for this system, the netmask, the gateway address, and the DNS address.

IPv6 options

Automatic

Anaconda uses *router advertisement* (RA) and DHCP for automatic configuration, based on the network environment. (Equivalent to the **Automatic** option in **NetworkManager**)

Automatic, DHCP only

Anaconda does not use RA, but requests information from DHCPv6 directly to create a stateful configuration. (Equivalent to the **Automatic, DHCP only** option in **NetworkManager**)

Manual configuration

Anaconda prompts you to enter the network configuration manually, including the IP address for this system, the netmask, the gateway address, and the DNS address.

Anaconda supports the IPv4 and IPv6 protocols. However, if you configure an interface to use both IPv4 and IPv6, the IPv4 connection must succeed or the interface will not work, even if the IPv6 connection succeeds.

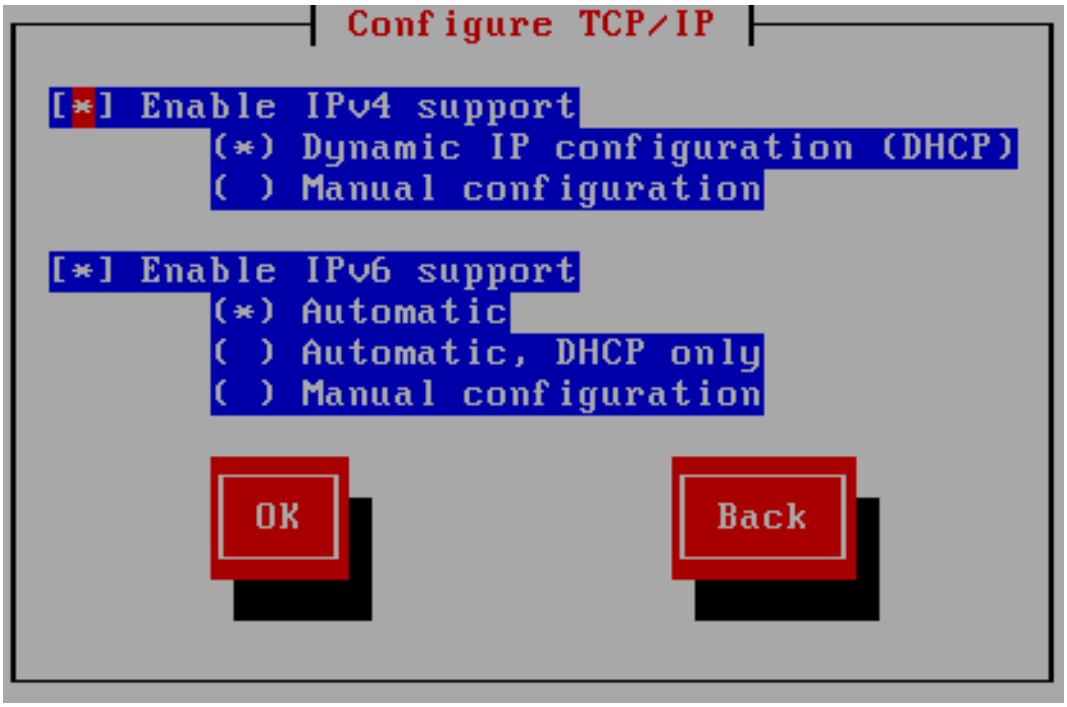


Figure 8.8. Configure TCP/IP

By default, **anaconda** uses DHCP to provide network settings automatically for IPv4 and automatic configuration to provide network settings for IPv6. If you choose to configure TCP/IP manually, **anaconda** prompts you to provide the details in the **Manual TCP/IP Configuration** dialog:

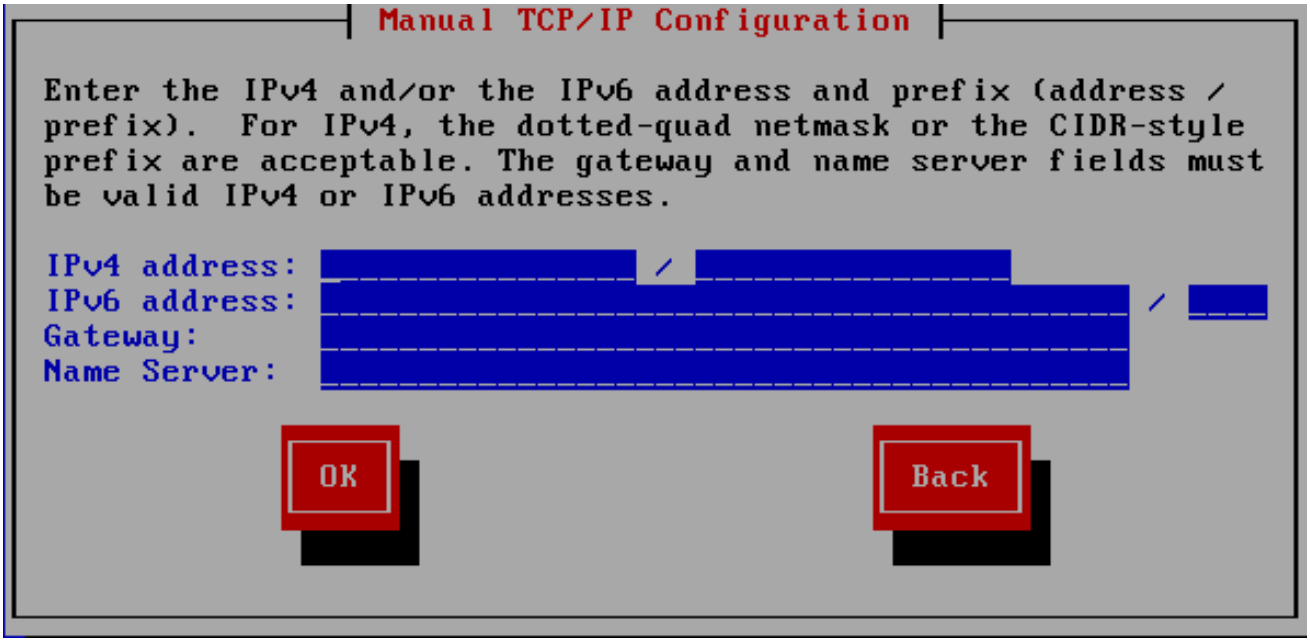


Figure 8.9. Manual TCP/IP Configuration

The dialog provides fields for IPv4 and IPv6 addresses and prefixes, depending on the protocols that you chose to configure manually, together with fields for the network gateway and name server. Enter the details for your network, then press **OK**.

When the installation process completes, it will transfer these settings to your system.

- If you are installing via NFS, proceed to Section 8.3.4, "Installing via NFS" .
- If you are installing via Web or FTP, proceed to Section 8.3.5, "Installing via FTP, HTTP, or HTTPS".

8.3.4. Installing via NFS

The NFS dialog applies only if you selected **NFS Image** in the **Installation Method** dialog. If you used the **repo=nfs** boot option, you already specified a server and path.

Figure 8.10. NFS Setup Dialog

1. Enter the domain name or IP address of your NFS server in the **NFS server name** field. For example, if you are installing from a host named **eastcoast** in the domain **example.com**, enter **eastcoast.example.com**.

2. Enter the name of the exported directory in the **Red Hat Enterprise Linux 6.9 directory** field:

   - If the NFS server is exporting a mirror of the Red Hat Enterprise Linux installation tree, enter the directory which contains the root of the installation tree. If everything was specified properly, a message appears indicating that the installation program for Red Hat Enterprise Linux is running.

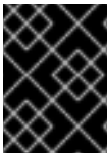
   - If the NFS server is exporting the ISO image of the Red Hat Enterprise Linux DVD, enter the directory which contains the ISO image.

   If you followed the setup described in Section 4.1.2, "Preparing for an NFS Installation" , the exported directory is the one that you specified as ***publicly_available_directory***.

3. Specify any NFS mount options that you require in the **NFS mount options** field. Refer to the man pages for **mount** and **nfs** for a comprehensive list of options. If you do not require any mount options, leave the field empty.

4. Proceed with Chapter 9, *Installing Using Anaconda*.

8.3.5. Installing via FTP, HTTP, or HTTPS



IMPORTANT

When you provide a URL to an installation source, you must explicitly specify **http://** or **https://** or **ftp://** as the protocol.

The URL dialog applies only if you are installing from a FTP, HTTP, or HTTPS server (if you selected **URL** in the **Installation Method** dialog). This dialog prompts you for information about the FTP, HTTP, or HTTPS server from which you are installing Red Hat Enterprise Linux. If you used the **repo=ftp** or **repo=http** boot options, you already specified a server and path.

Enter the name or IP address of the FTP, HTTP, or HTTPS site from which you are installing, and the name of the directory that contains the **/images** directory for your architecture. For example:

/mirrors/redhat/rhel-6.9/Server/i386/

To install via a secure HTTPS connection, specify **https://** as the protocol.

Specify the address of a proxy server, and if necessary, provide a port number, username, and password. If everything was specified properly, a message box appears indicating that files are being retrieved from the server.

If your FTP, HTTP, or HTTPS server requires user authentication, specify user and password as part of the URL as follows:

{ftp|http|https}://<user>:<password>@<hostname>[:<port>]/<directory>/

For example:

http://install:rhel6.9pw@name.example.com/mirrors/redhat/rhel-6.9/Server/i386/

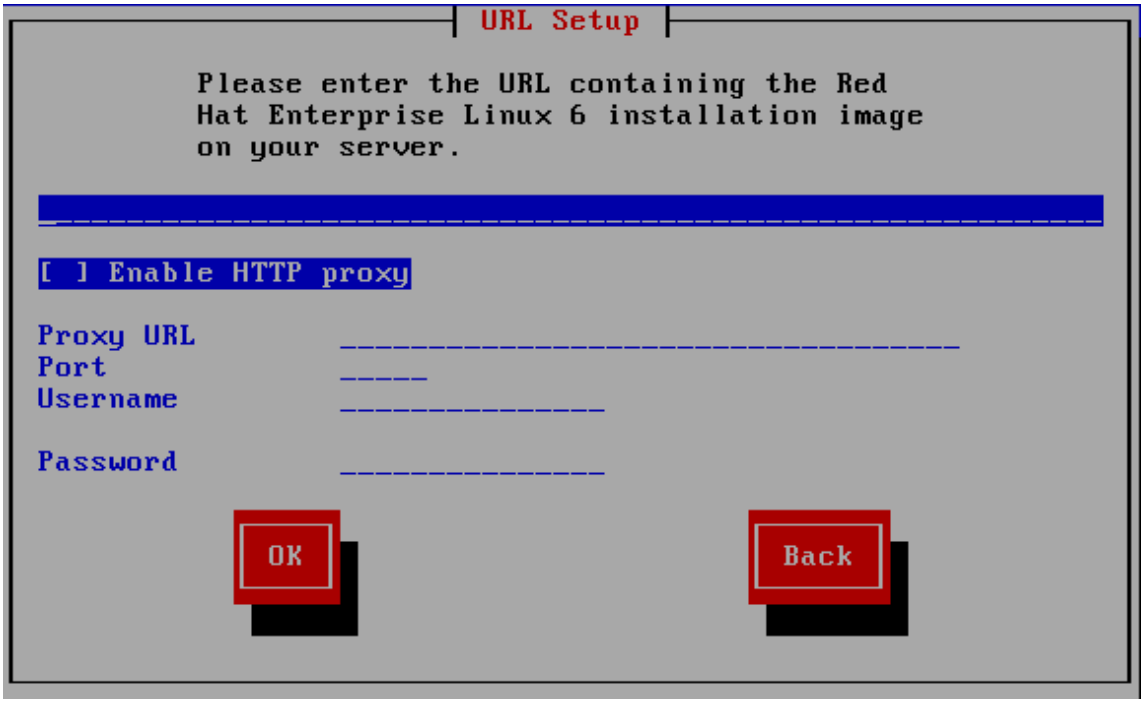


Figure 8.11. URL Setup Dialog

Proceed with Chapter 9, *Installing Using Anaconda*.

8.4. VERIFYING MEDIA

The DVD offers an option to verify the integrity of the media. Recording errors sometimes occur while producing DVD media. An error in the data for package chosen in the installation program can cause the installation to abort. To minimize the chances of data errors affecting the installation, verify the media before installing.

If the verification succeeds, the installation process proceeds normally. If the process fails, create a new DVD using the ISO image you downloaded earlier.

CHAPTER 9. INSTALLING USING ANACONDA

This chapter describes an installation using the graphical user interface of **anaconda**.

9.1. THE TEXT MODE INSTALLATION PROGRAM USER INTERFACE

IMPORTANT

Installing in text mode does not prevent you from using a graphical interface on your system once it is installed.

Apart from the graphical installer, **anaconda** also includes a text-based installer.

If one of the following situations occurs, the installation program uses text mode:

- The installation system fails to identify the display hardware on your computer
- You choose the text mode installation from the boot menu

While text mode installations are not explicitly documented, those using the text mode installation program can easily follow the GUI installation instructions. However, because text mode presents you with a simpler, more streamlined installation process, certain options that are available in graphical mode are not also available in text mode. These differences are noted in the description of the installation process in this guide, and include:

- configuring advanced storage methods such as LVM, RAID, FCoE, zFCP, and iSCSI.
- customizing the partition layout
- customizing the bootloader layout
- selecting packages during installation
- configuring the installed system with **firstboot**

If you choose to install Red Hat Enterprise Linux in text mode, you can still configure your system to use a graphical interface after installation. Refer to Section 35.3, “Switching to a Graphical Login” for instructions.

To configure options not available in text mode, consider using a boot option. For example, the **linux ip** option can be used to configure network settings. Refer to Section 28.1, “Configuring the Installation System at the Boot Menu” for instructions.

9.2. THE GRAPHICAL INSTALLATION PROGRAM USER INTERFACE

If you have used a *graphical user interface (GUI)* before, you are already familiar with this process; use your mouse to navigate the screens, click buttons, or enter text fields.

You can also navigate through the installation using the keyboard. The **Tab** key allows you to move around the screen, the Up and Down arrow keys to scroll through lists, **+** and **-** keys expand and collapse lists, while **Space** and **Enter** selects or removes from selection a highlighted item. You can also use the **Alt**+***X*** key command combination as a way of clicking on buttons or making other screen selections, where ***X*** is replaced with any underlined letter appearing within that screen.

NOTE

If you are using an x86, AMD64, or Intel 64 system, and you do not wish to use the GUI installation program, the text mode installation program is also available. To start the text mode installation program, use the following command at the **boot:** prompt:

```
linux text
```

Refer to Section 7.1.2, "The Boot Menu" for a description of the Red Hat Enterprise Linux boot menu and to Section 8.1, "The Text Mode Installation Program User Interface" for a brief overview of text mode installation instructions.

It is highly recommended that installs be performed using the GUI installation program. The GUI installation program offers the full functionality of the Red Hat Enterprise Linux installation program, including LVM configuration which is not available during a text mode installation.

Users who must use the text mode installation program can follow the GUI installation instructions and obtain all needed information.

9.2.1. Screenshots During Installation

Anaconda allows you to take screenshots during the installation process. At any time during installation, press **Shift**+**Print Screen** and **anaconda** will save a screenshot to **/root/anaconda-screenshots**.

If you are performing a Kickstart installation, use the **autostep --autoscreenshot** option to generate a screenshot of each step of the installation automatically. Refer to Section 32.3, "Creating the Kickstart File" for details of configuring a Kickstart file.

9.2.2. A Note About Virtual Consoles

The Red Hat Enterprise Linux installation program offers more than the dialog boxes of the installation process. Several kinds of diagnostic messages are available to you, as well as a way to enter commands from a shell prompt. The installation program displays these messages on five *virtual consoles*, among which you can switch using a single keystroke combination.

A virtual console is a shell prompt in a non-graphical environment, accessed from the physical machine, not remotely. Multiple virtual consoles can be accessed simultaneously.

These virtual consoles can be helpful if you encounter a problem while installing Red Hat Enterprise Linux. Messages displayed on the installation or system consoles can help pinpoint a problem. Refer to Table 9.1, "Console, Keystrokes, and Contents" for a listing of the virtual consoles, keystrokes used to switch to them, and their contents.

Generally, there is no reason to leave the default console (virtual console #6) for graphical installations unless you are attempting to diagnose installation problems.

Table 9.1. Console, Keystrokes, and Contents

console	keystrokes	contents
1	**ctrl**+**alt**+**f1**	graphical display
2	**ctrl**+**alt**+**f2**	shell prompt

console	keystrokes	contents
3	**ctrl+alt+f3**	install log (messages from installation program)
4	**ctrl+alt+f4**	system-related messages
5	**ctrl+alt+f5**	other messages

9.3. WELCOME TO RED HAT ENTERPRISE LINUX

The **Welcome** screen does not prompt you for any input.

Figure 9.1. The Welcome screen

Click on the **Next** button to continue.

9.4. LANGUAGE SELECTION

Using your mouse, select the language (for example, U.S. English) you would prefer to use for the installation and as the system default (refer to the figure below).

Once you have made your selection, click **Next** to continue.

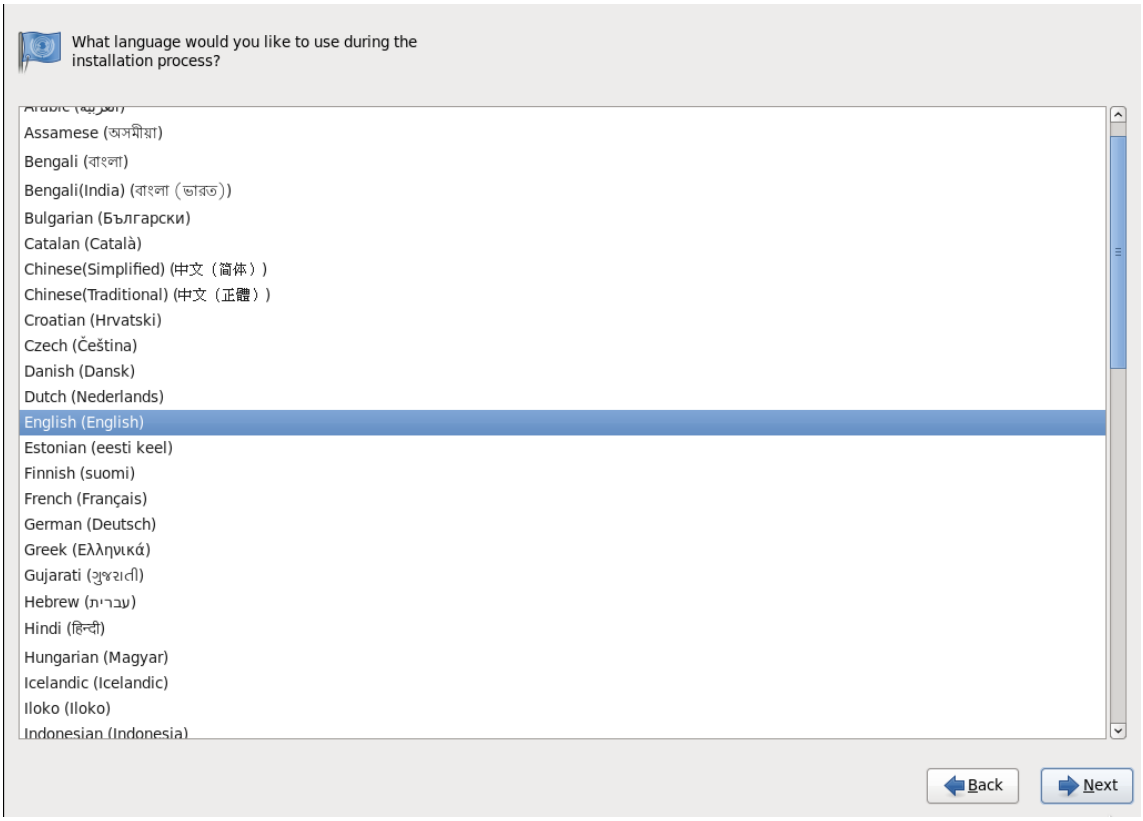


Figure 9.2. Language Configuration

9.5. KEYBOARD CONFIGURATION

Using your mouse, select the correct layout type (for example, U.S. English) for the keyboard you would prefer to use for the installation and as the system default (refer to the figure below).

Once you have made your selection, click **Next** to continue.

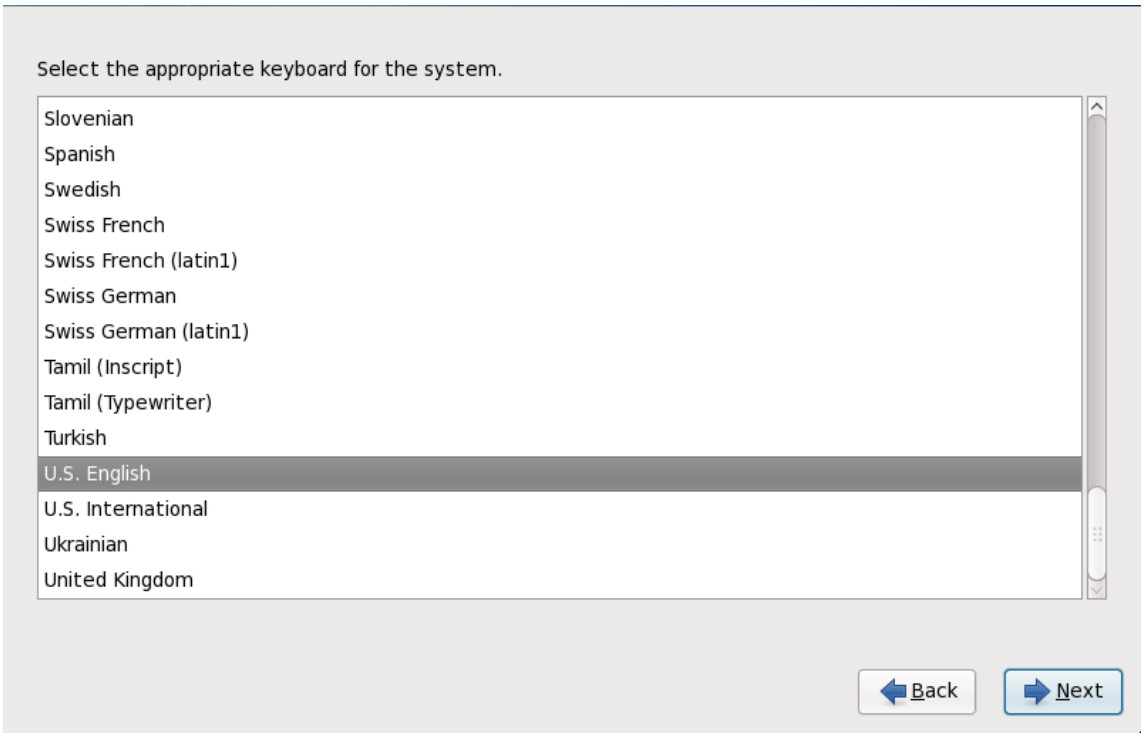
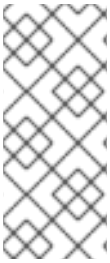


Figure 9.3. Keyboard Configuration

Red Hat Enterprise Linux includes support for more than one keyboard layout for many languages. In particular, most European languages include a **latin1** option, which uses *dead keys* to access certain characters, such as those with diacritical marks. When you press a dead key, nothing will appear on your screen until you press another key to "complete" the character. For example, to type **é** on a latin1 keyboard layout, you would press (and release) the **'** key, and then press the **E** key. By contrast, you access this character on some other keyboards by pressing and holding down a key (such as **Alt-Gr**) while you press the **E** key. Other keyboards might have a dedicated key for this character.



NOTE

To change your keyboard layout type after you have completed the installation, use the **Keyboard Configuration Tool**.

Type the **system-config-keyboard** command in a shell prompt to launch the **Keyboard Configuration Tool**. If you are not root, it prompts you for the root password to continue.

9.6. STORAGE DEVICES

You can install Red Hat Enterprise Linux on a large variety of storage devices. This screen allows you to select either basic or specialized storage devices.

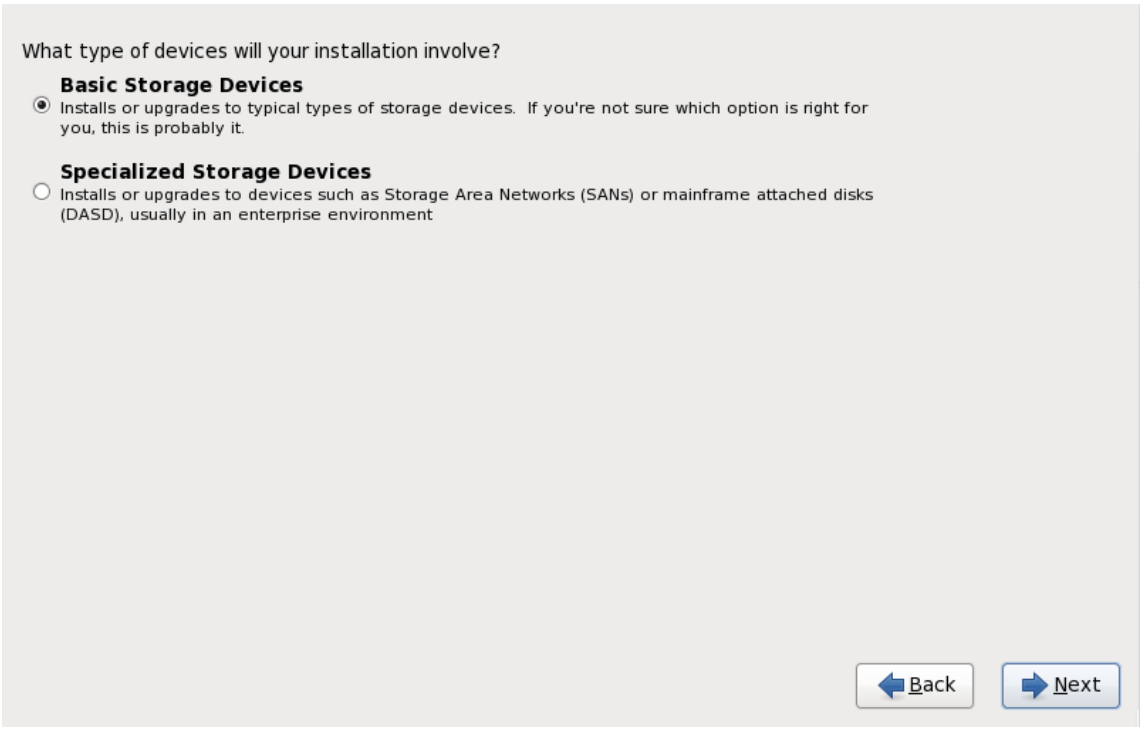


Figure 9.4. Storage devices

Basic Storage Devices

Select **Basic Storage Devices** to install Red Hat Enterprise Linux on the following storage devices:

- hard drives or solid-state drives connected directly to the local system.

Specialized Storage Devices

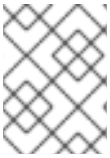
Select **Specialized Storage Devices** to install Red Hat Enterprise Linux on the following storage devices:

- *Storage area networks* (SANs)

- *Direct access storage devices* (DASDs)
- Firmware RAID devices
- Multipath devices

Use the **Specialized Storage Devices**option to configure *Internet Small Computer System Interface* (iSCSI) and *FCoE* (Fiber Channel over Ethernet) connections.

If you select **Basic Storage Devices**, **anaconda** automatically detects the local storage attached to the system and does not require further input from you. Proceed to Section 9.7, "Setting the Hostname".



NOTE

Monitoring of LVM and software RAID devices by the **mdeventd** daemon is not performed during installation.

9.6.1. The Storage Devices Selection Screen

The storage devices selection screen displays all storage devices to which **anaconda** has access.

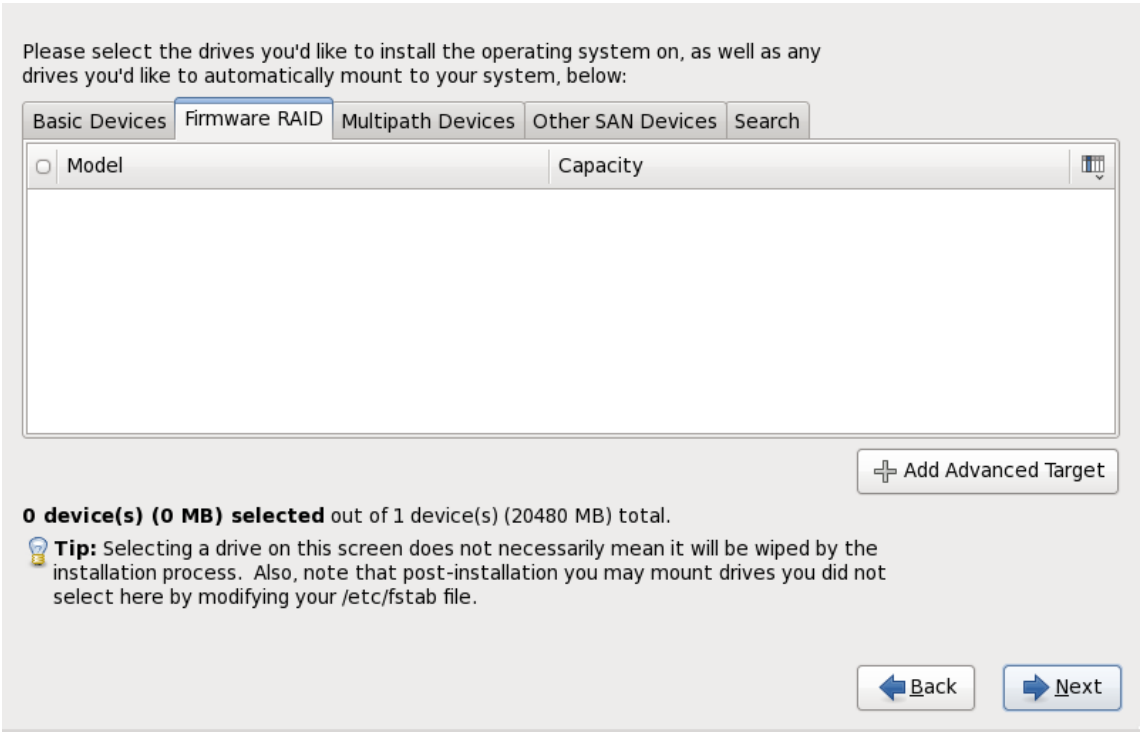


Figure 9.5. Select storage devices — Basic devices

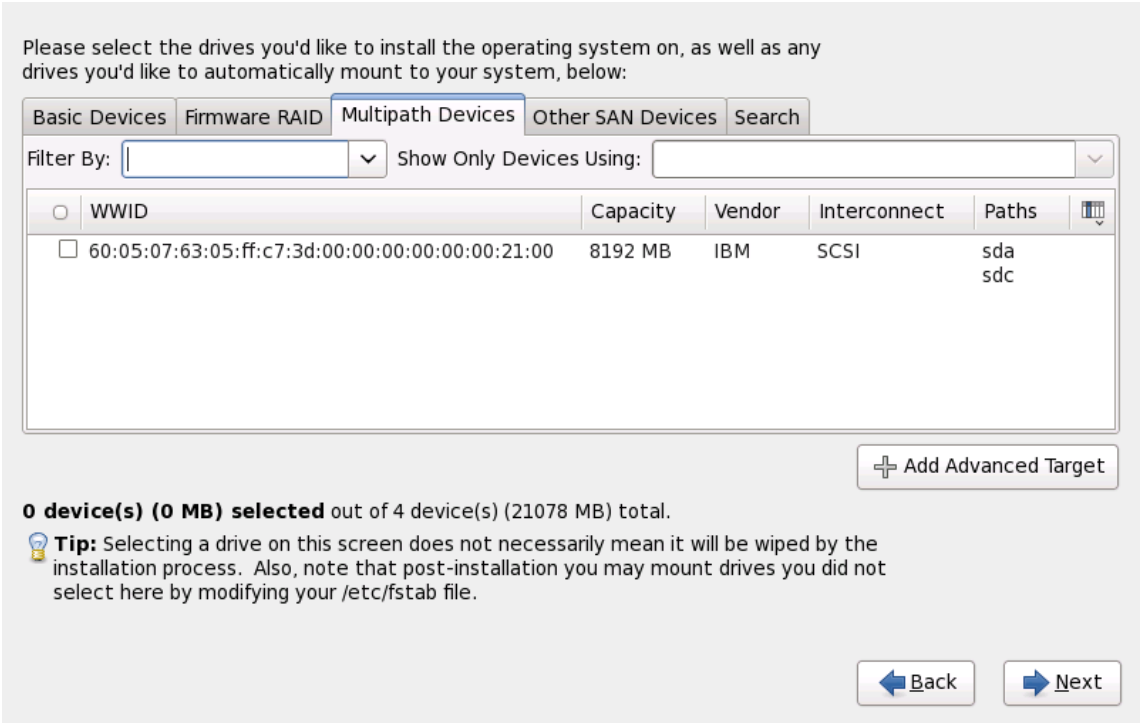


Figure 9.6. Select storage devices — Multipath Devices

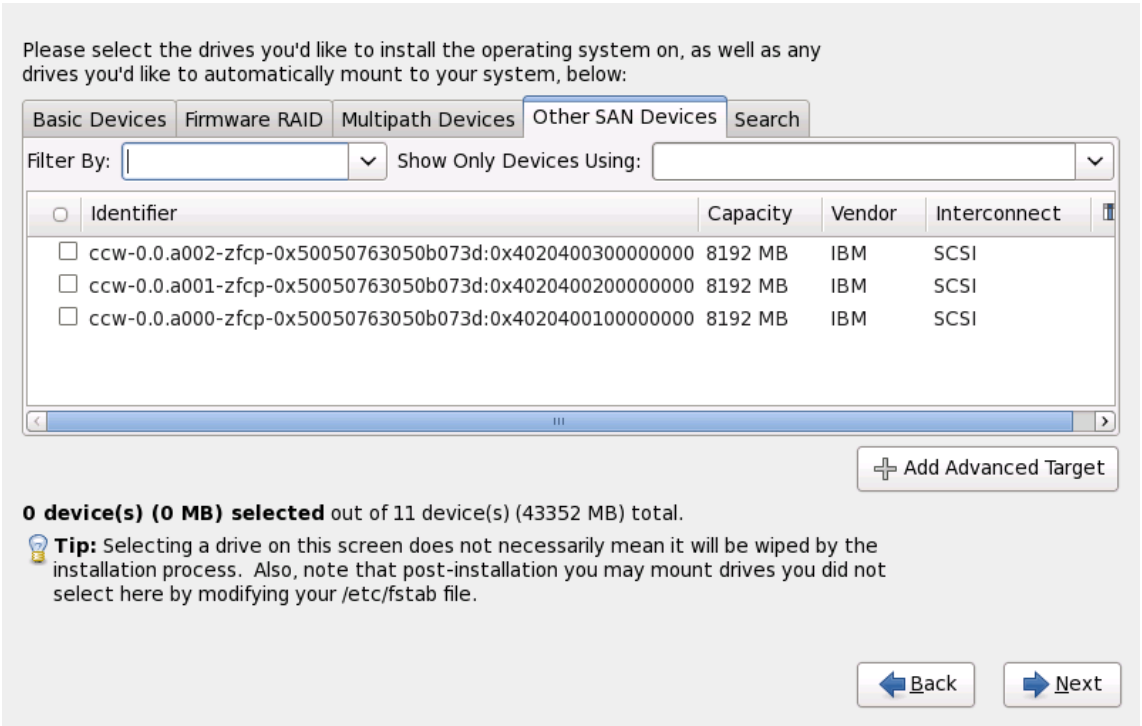


Figure 9.7. Select storage devices — Other SAN Devices

Devices are grouped under the following tabs:

Basic Devices

Basic storage devices directly connected to the local system, such as hard disk drives and solid-state drives.

Firmware RAID

Storage devices attached to a firmware RAID controller.

Multipath Devices

Storage devices accessible through more than one path, such as through multiple SCSI controllers or Fiber Channel ports on the same system.

IMPORTANT

The installer only detects multipath storage devices with serial numbers that are 16 or 32 characters in length.

Other SAN Devices

Any other devices available on a storage area network (SAN).

If you do need to configure iSCSI or FCoE storage, click **Add Advanced Target** and refer to Section 9.6.1.1, " Advanced Storage Options ".

The storage devices selection screen also contains a **Search** tab that allows you to filter storage devices either by their *World Wide Identifier* (WWID) or by the port, target, or *logical unit number* (LUN) at which they are accessed.

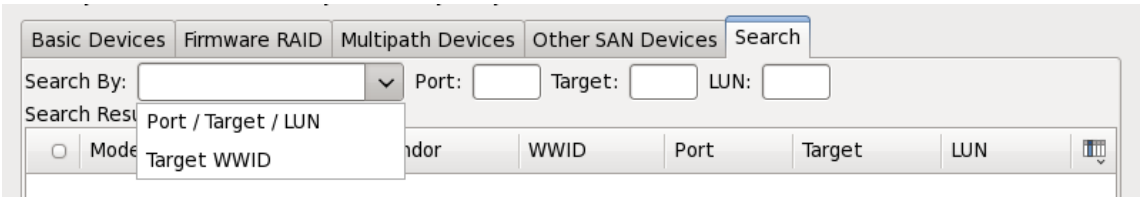


Figure 9.8. The Storage Devices Search Tab

The tab contains a drop-down menu to select searching by port, target, WWID, or LUN (with corresponding text boxes for these values). Searching by WWID or LUN requires additional values in the corresponding text box.

Each tab presents a list of devices detected by **anaconda**, with information about the device to help you to identify it. A small drop-down menu marked with an icon is located to the right of the column headings. This menu allows you to select the types of data presented on each device. For example, the menu on the **Multipath Devices** tab allows you to specify any of **WWID**, **Capacity**, **Vendor**, **Interconnect**, and **Paths** to include among the details presented for each device. Reducing or expanding the amount of information presented might help you to identify particular devices.

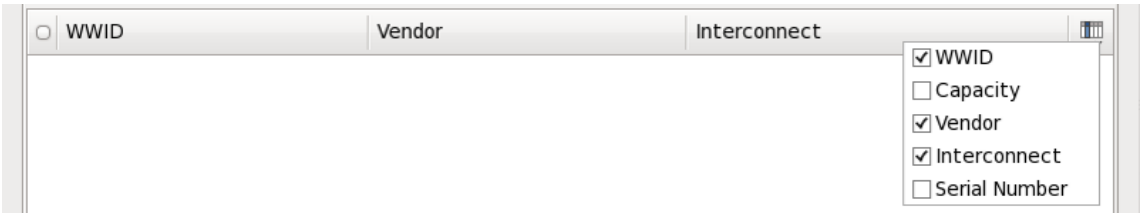


Figure 9.9. Selecting Columns

Each device is presented on a separate row, with a checkbox to its left. Click the checkbox to make a device available during the installation process, or click the *radio button* at the left of the column headings to select or deselect all the devices listed in a particular screen. Later in the installation process, you can choose to install Red Hat Enterprise Linux onto any of the devices selected here, and can choose to automatically mount any of the other devices selected here as part of the installed system.

Note that the devices that you select here are not automatically erased by the installation process. Selecting a device on this screen does not, in itself, place data stored on the device at risk. Note also that any devices that you do not select here to form part of the installed system can be added to the system after installation by modifying the **/etc/fstab** file.

IMPORTANT

Any storage devices that you do not select on this screen are hidden from **anaconda** entirely. To *chain load* the Red Hat Enterprise Linux boot loader from a different boot loader, select all the devices presented in this screen.

when you have selected the storage devices to make available during installation, click **Next** and proceed to Section 9.11, "Initializing the Hard Disk"

9.6.1.1. Advanced Storage Options

From this screen you can configure an *iSCSI* (SCSI over TCP/IP) target or *FCoE* (Fibre channel over ethernet) *SAN* (storage area network). Refer to Appendix B, *iSCSI Disks* for an introduction to iSCSI.

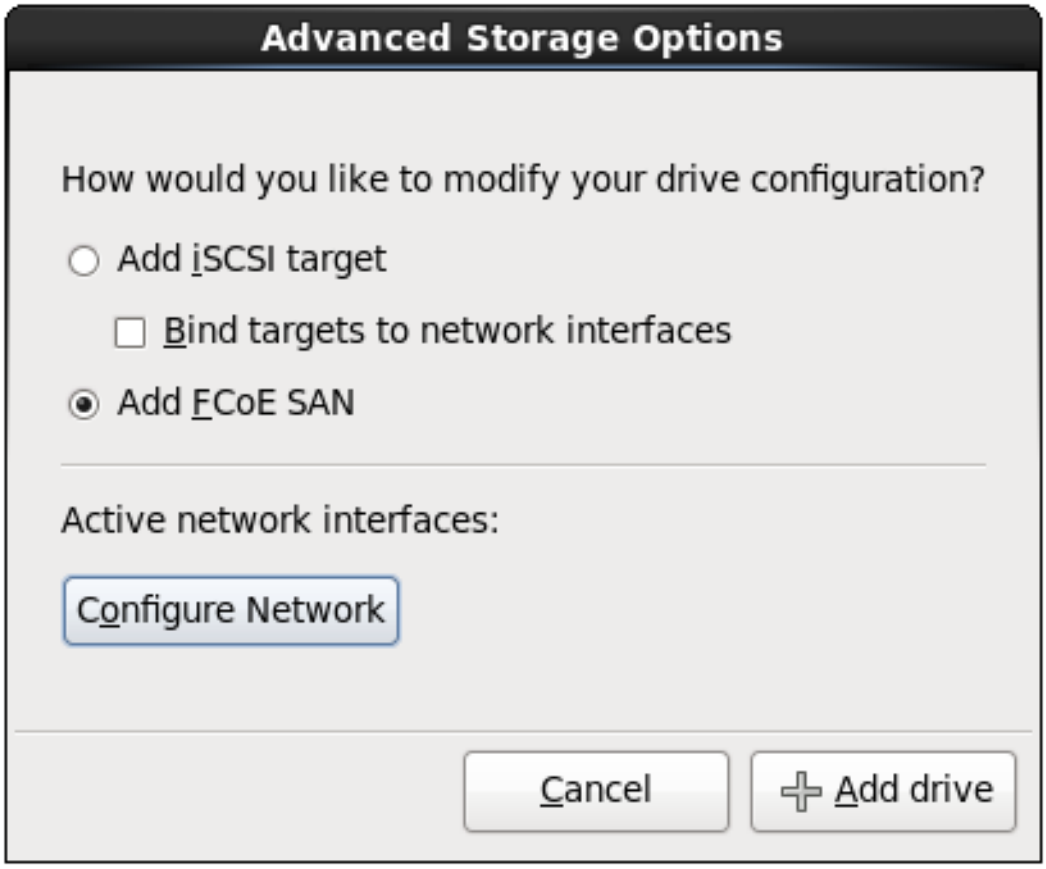


Figure 9.10. Advanced Storage Options

Select **Add iSCSI target** or **Add FCoE SAN** and click **Add drive**. If adding an iSCSI target, optionally check the box labeled **Bind targets to network interfaces**.

9.6.1.1.1. Select and configure a network interface

The **Advanced Storage Options** screen lists the active network interfaces **anaconda** has found on your system. If none are found, **anaconda** must activate an interface through which to connect to the storage devices.

Click **Configure Network** on the **Advanced Storage Options** screen to configure and activate one using **NetworkManager** to use during installation. Alternatively, **anaconda** will prompt you with the **Select network interface** dialog after you click **Add drive**.

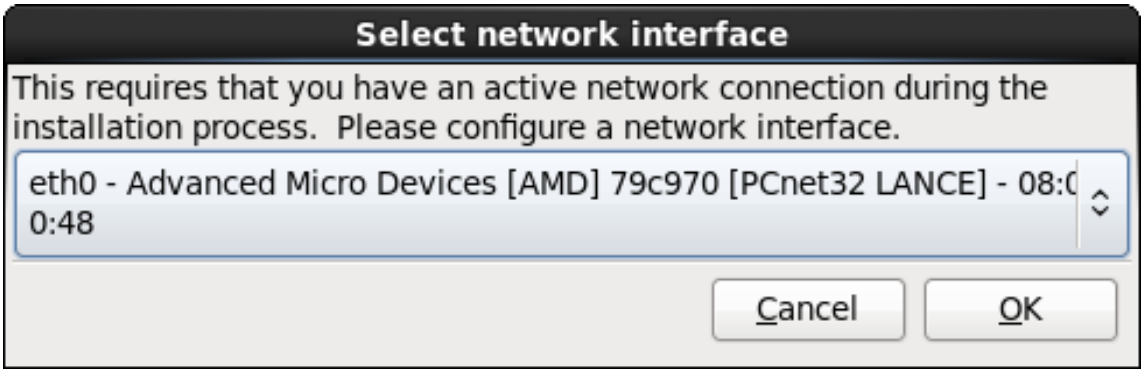


Figure 9.11. Select network interface

1. Select an interface from the drop-down menu.
2. Click **OK**.

Anaconda then starts **NetworkManager** to allow you to configure the interface.

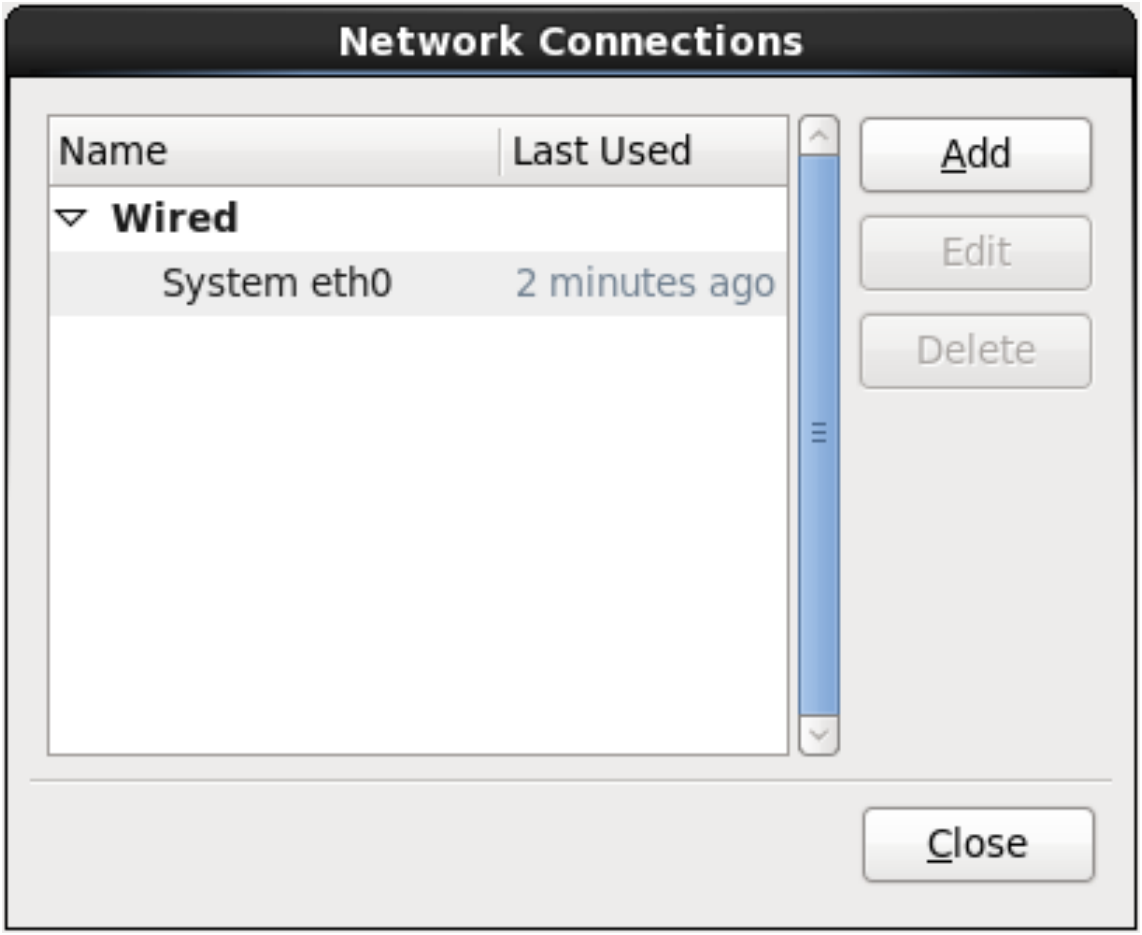


Figure 9.12. Network Connections

For details of how to use **NetworkManager**, refer to Section 9.7, "Setting the Hostname"

9.6.1.1.2. Configure iSCSI parameters

To add an iSCSI target, select **Add iSCSI target** and click **Add drive**.

To use iSCSI storage devices for the installation, **anaconda** 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 username and password for *CHAP* (Challenge Handshake Authentication Protocol) 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 username and password are different for CHAP authentication and reverse CHAP authentication.

Repeat the iSCSI discovery and iSCSI login steps as many times as necessary to add all required iSCSI storage. However, 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.

Procedure 9.1. iSCSI discovery

Use the **iSCSI Discovery Details** dialog to provide **anaconda** with the information that it needs to discover the iSCSI target.

Figure 9.13. The iSCSI Discovery Details dialog

1. Enter the IP address of the iSCSI target in the **Target IP Address** field.
2. Provide a name in the **iSCSI Initiator Name** field for the iSCSI initiator in *iSCSI qualified name* (IQN) format.

   A valid IQN contains:

   - 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 therefore resembles: **iqn.2010-09.storage.example.com:diskarrays-sn-a8675309**, and **anaconda** pre-populates the **iSCSI Initiator Name** field with a name in this format to help you with the structure.

For more information on IQNs, refer to *3.2.6. iSCSI Names* in *RFC 3720 - Internet Small Computer Systems Interface (iSCSI)* available from http://tools.ietf.org/html/rfc3720#section-3.2.6 and *1. iSCSI Names and Addresses* in *RFC 3721 - Internet Small Computer Systems Interface (iSCSI) Naming and Discovery* available from http://tools.ietf.org/html/rfc3721#section-1.

3. Use the drop-down menu to specify the type of authentication to use for iSCSI discovery:

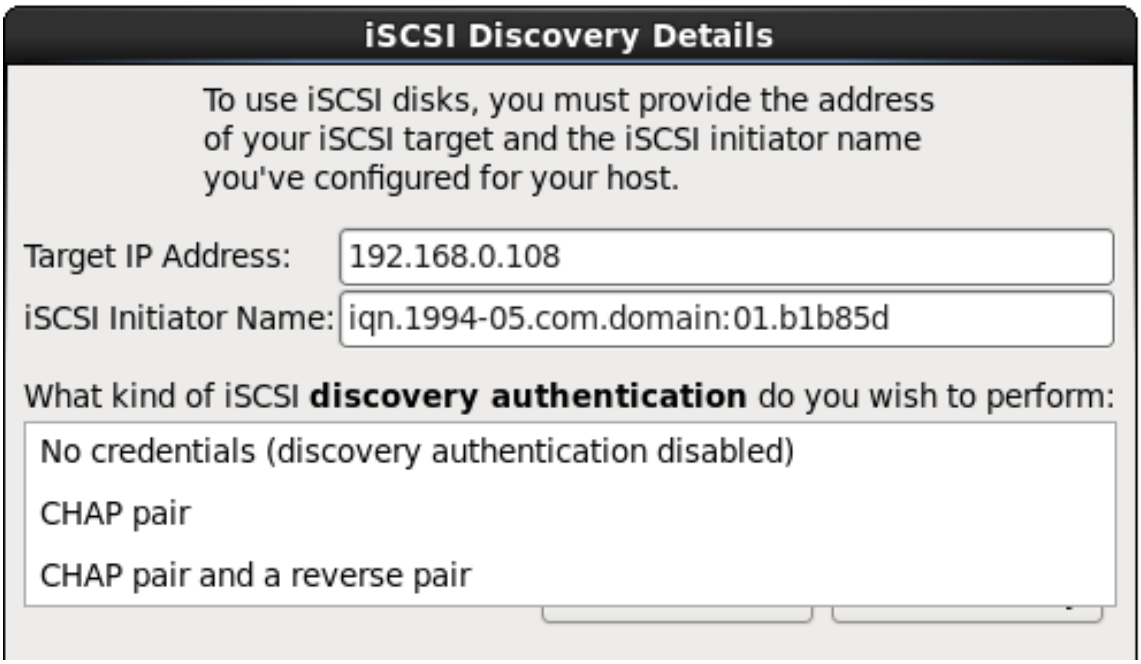


Figure 9.14. iSCSI discovery authentication

- **no credentials**

- **CHAP pair**

- **CHAP pair and a reverse pair**

4. - If you selected **CHAP pair** as the authentication type, provide the username and password for the iSCSI target in the **CHAP Username** and **CHAP Password** fields.

Figure 9.15. CHAP pair

- If you selected **CHAP pair and a reverse pair** as the authentication type, provide the username and password for the iSCSI target in the **CHAP Username** and **CHAP Password** field and the username and password for the iSCSI initiator in the **Reverse CHAP Username** and **Reverse CHAP Password** fields.

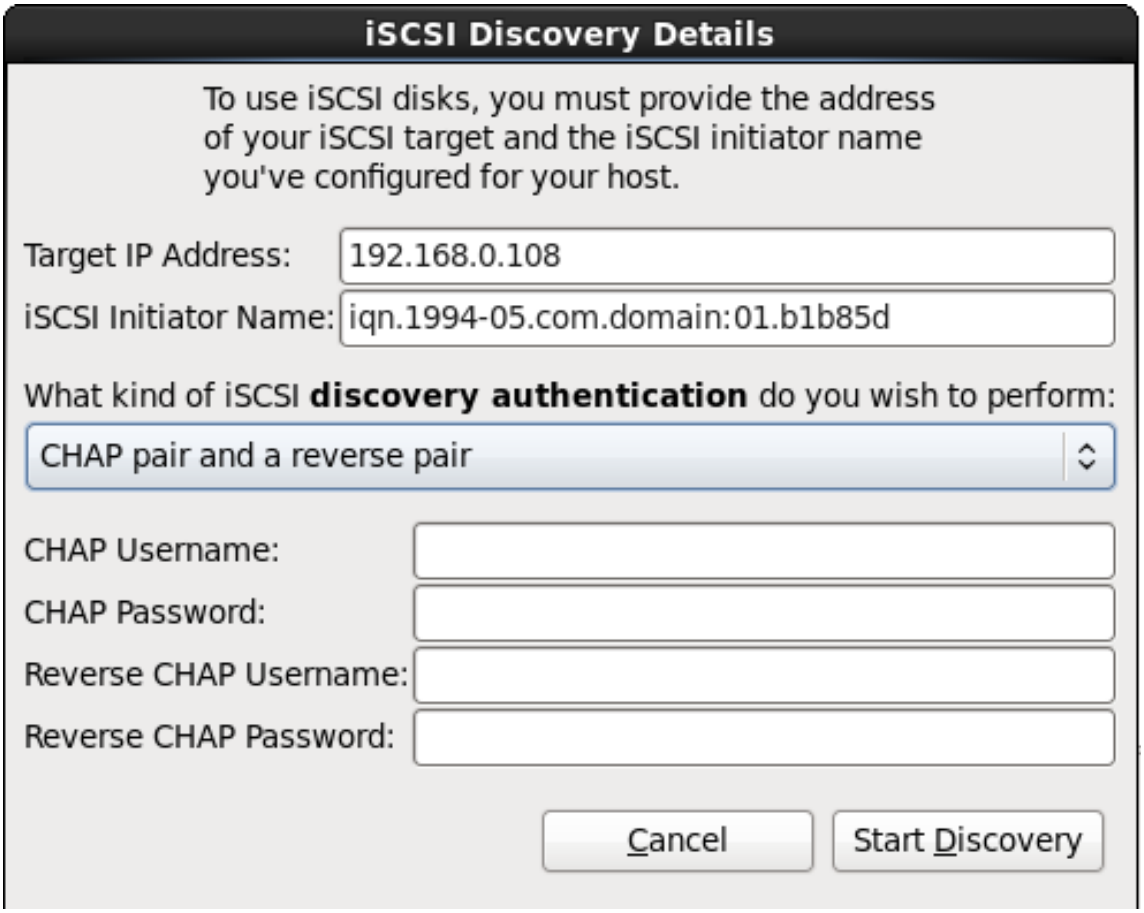


Figure 9.16. CHAP pair and a reverse pair

5. Click **Start Discovery**. **Anaconda** attempts to discover an iSCSI target based on the information that you provided. If discovery succeeds, the **iSCSI Discovered Nodes** dialog presents you with a list of all the iSCSI nodes discovered on the target.

6. Each node is presented with a checkbox beside it. Click the checkboxes to select the nodes to use for installation.

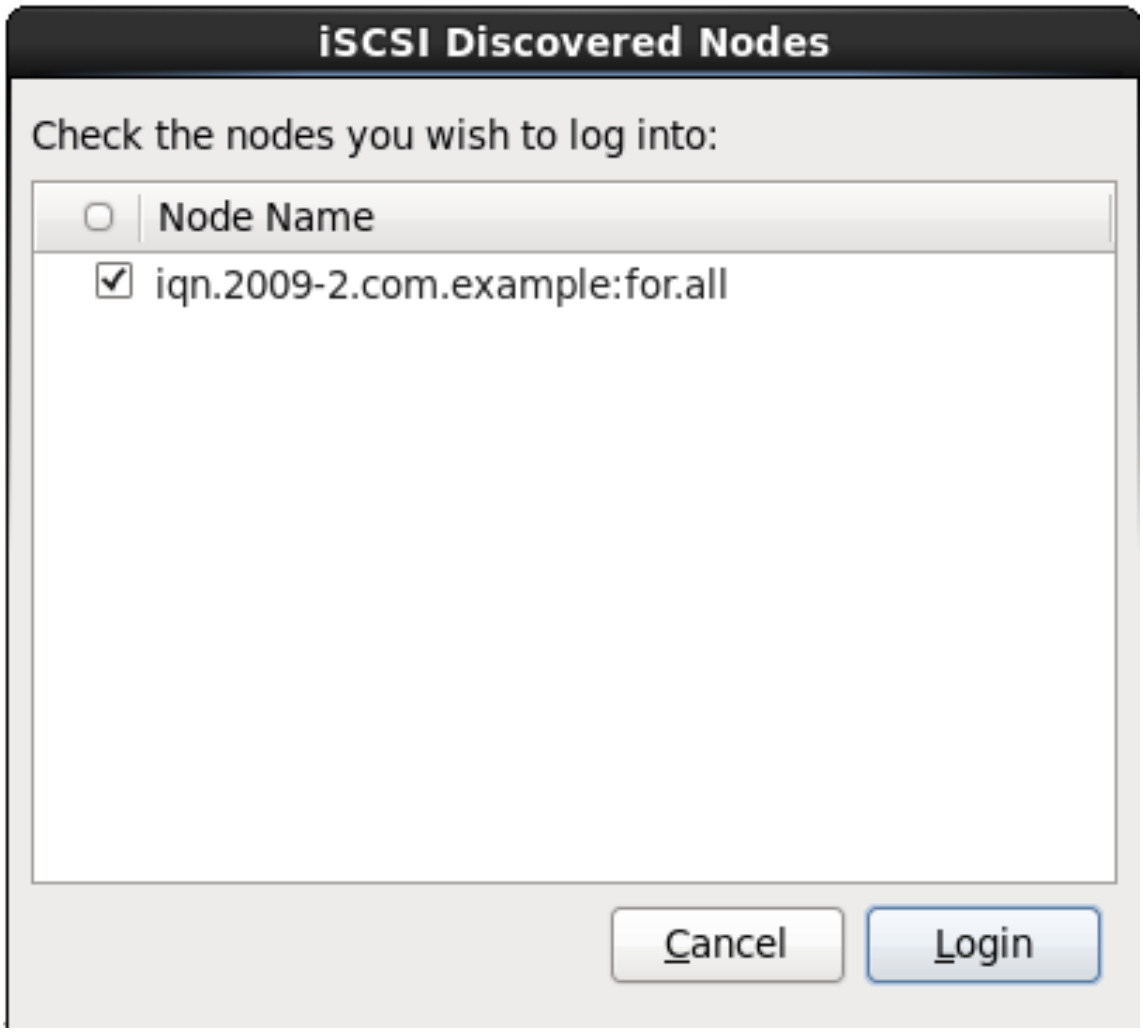


Figure 9.17. The iSCSI Discovered Nodes dialog

7. Click **Login** to initiate an iSCSI session.

Procedure 9.2. Starting an iSCSI session

Use the **iSCSI Nodes Login** dialog to provide **anaconda** with the information that it needs to log into the nodes on the iSCSI target and start an iSCSI session.

Figure 9.18. The iSCSI Nodes Login dialog

1. Use the drop-down menu to specify the type of authentication to use for the iSCSI session:

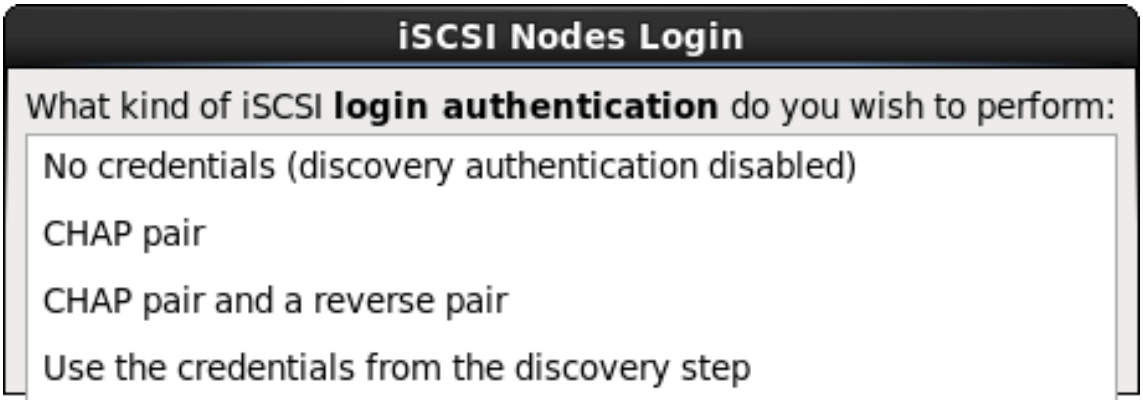


Figure 9.19. iSCSI session authentication

- **no credentials**
- **CHAP pair**
- **CHAP pair and a reverse pair**
- **Use the credentials from the discovery step**

If your environment uses the same type of authentication and same username and password for iSCSI discovery and for the iSCSI session, select **Use the credentials from the discovery step** to reuse these credentials.

2. - If you selected **CHAP pair** as the authentication type, provide the username and password for the iSCSI target in the **CHAP Username** and **CHAP Password** fields.

Figure 9.20. CHAP pair

- If you selected **CHAP pair and a reverse pair** as the authentication type, provide the username and password for the iSCSI target in the **CHAP Username** and **CHAP Password** fields and the username and password for the iSCSI initiator in the **Reverse CHAP Username** and **Reverse CHAP Password** fields.

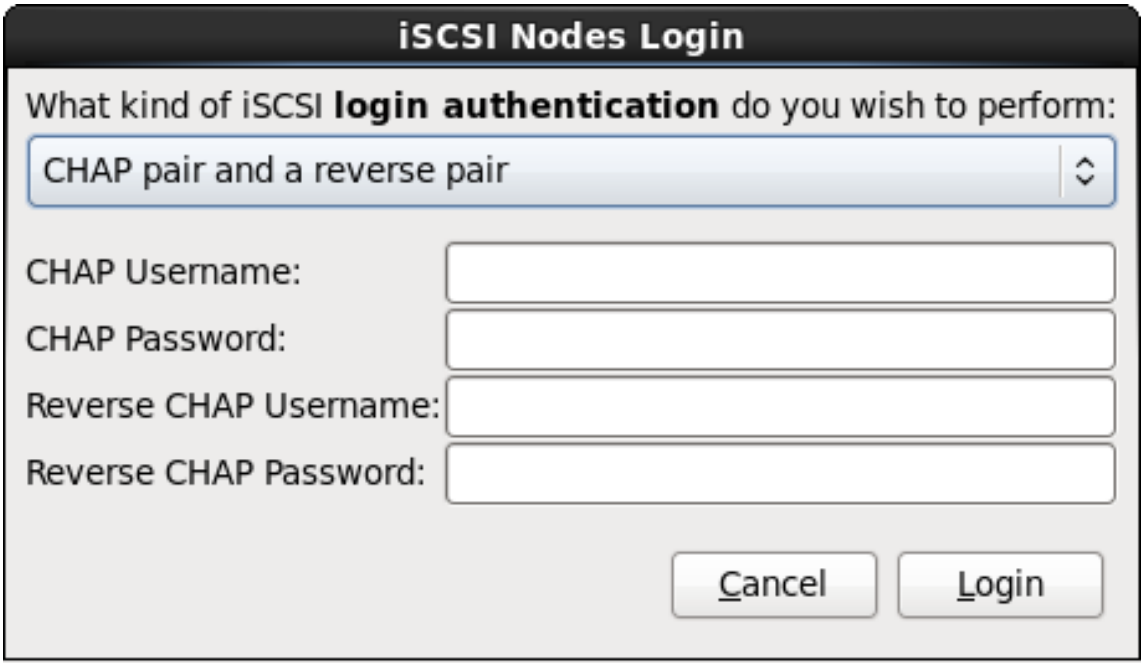


Figure 9.21. CHAP pair and a reverse pair

3. Click **Login**. **Anaconda** attempts to log into the nodes on the iSCSI target based on the information that you provided. The **iSCSI Login Results** dialog presents you with the results.

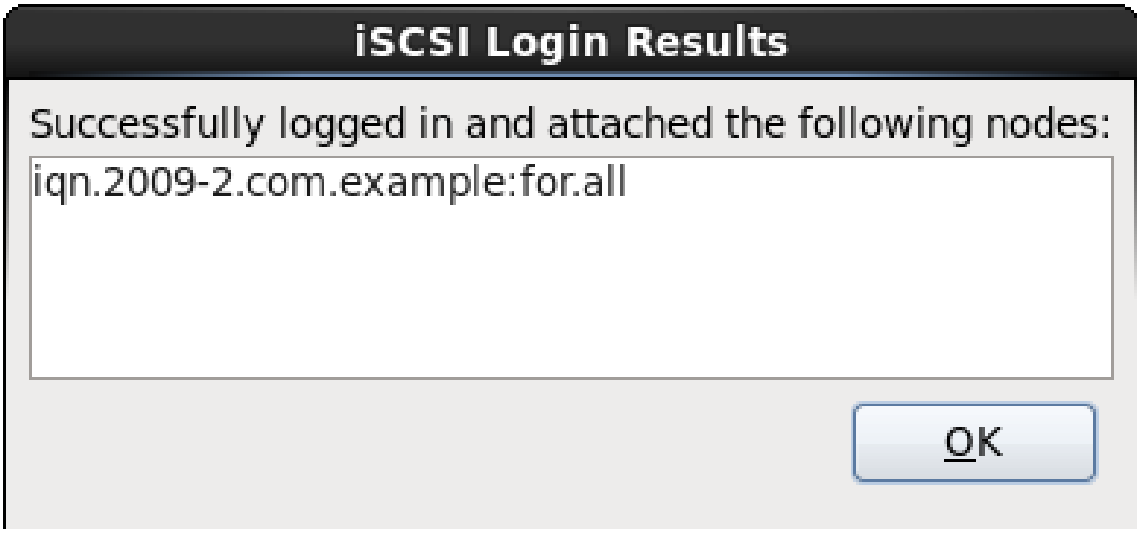


Figure 9.22. The iSCSI Login Results dialog

4. Click **OK** to continue.

9.6.1.1.3. Configure FCoE Parameters

To configure an FCoE SAN, select **Add FCoE SAN** and click **Add Drive**.

In the next dialog box that appears after you click **Add drive**, select the network interface that is connected to your FCoE switch and click **Add FCoE Disk(s)**.

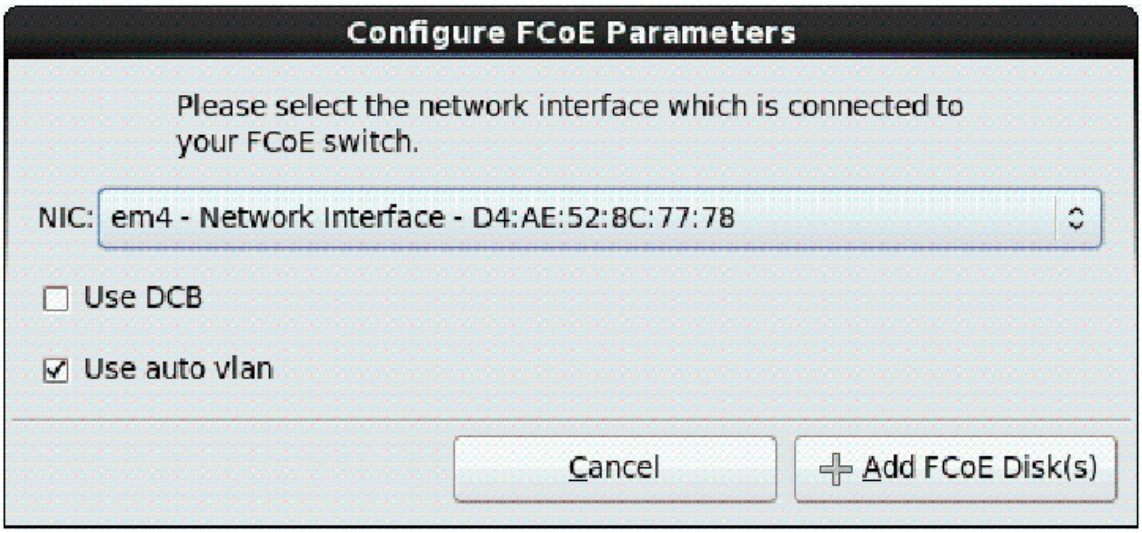


Figure 9.23. Configure FCoE Parameters

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. Enable or disable the installer's awareness of DCB with the checkbox in this dialog. This should only be set for networking interfaces that require a host-based DCBX client. Configurations on interfaces that implement a hardware DCBX client should leave this checkbox empty.

Auto VLAN indicates whether VLAN discovery should be performed. If this box is checked, then the FIP VLAN discovery protocol will run on the Ethernet interface once the link configuration has been validated. If they are not already configured, network interfaces for any discovered FCoE VLANs will be automatically created and FCoE instances will be created on the VLAN interfaces.

9.7. SETTING THE HOSTNAME

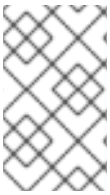
Setup prompts you to supply a host name for this computer, either as a *fully-qualified domain name* (FQDN) in the format *hostname.domainname* or as a *short host name* in the format *hostname*. Many networks have a *Dynamic Host Configuration Protocol* (DHCP) service that automatically supplies connected systems with a domain name. To allow the DHCP service to assign the domain name to this machine, specify the short host name only.

NOTE

You may give your system any name provided that the full hostname is unique. The hostname may include letters, numbers and hyphens.

Figure 9.24. Setting the hostname

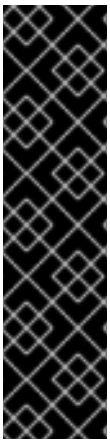
If your Red Hat Enterprise Linux system is connected *directly* to the Internet, you must pay attention to additional considerations to avoid service interruptions or risk action by your upstream service provider. A full discussion of these issues is beyond the scope of this document.



NOTE

The installation program does not configure modems. Configure these devices after installation with the **Network** utility. The settings for your modem are specific to your particular Internet Service Provider (ISP).

9.7.1. Editing Network Connections



IMPORTANT

When a Red Hat Enterprise Linux 6.9 installation boots for the first time, it activates any network interfaces that you configured during the installation process. However, the installer does not prompt you to configure network interfaces on some common installation paths, for example, when you install Red Hat Enterprise Linux from a DVD to a local hard drive.

When you install Red Hat Enterprise Linux from a local installation source to a local storage device, be sure to configure at least one network interface manually if you require network access when the system boots for the first time. You will need to select the **Connect automatically** option manually when editing the connection.

NOTE

To change your network configuration after you have completed the installation, use the **Network Administration Tool**.

Type the **system-config-network** command in a shell prompt to launch the **Network Administration Tool**. If you are not root, it prompts you for the root password to continue.

The **Network Administration Tool** is now deprecated and will be replaced by **NetworkManager** during the lifetime of Red Hat Enterprise Linux 6.

To configure a network connection manually, click the button **Configure Network**. The **Network Connections** dialog appears that allows you to configure wired, wireless, mobile broadband, InfiniBand, VPN, DSL, VLAN, and bonded connections for the system using the **NetworkManager** tool. A full description of all configurations possible with **NetworkManager** is beyond the scope of this guide. This section only details the most typical scenario of how to configure wired connections during installation. Configuration of other types of network is broadly similar, although the specific parameters that you must configure are necessarily different.

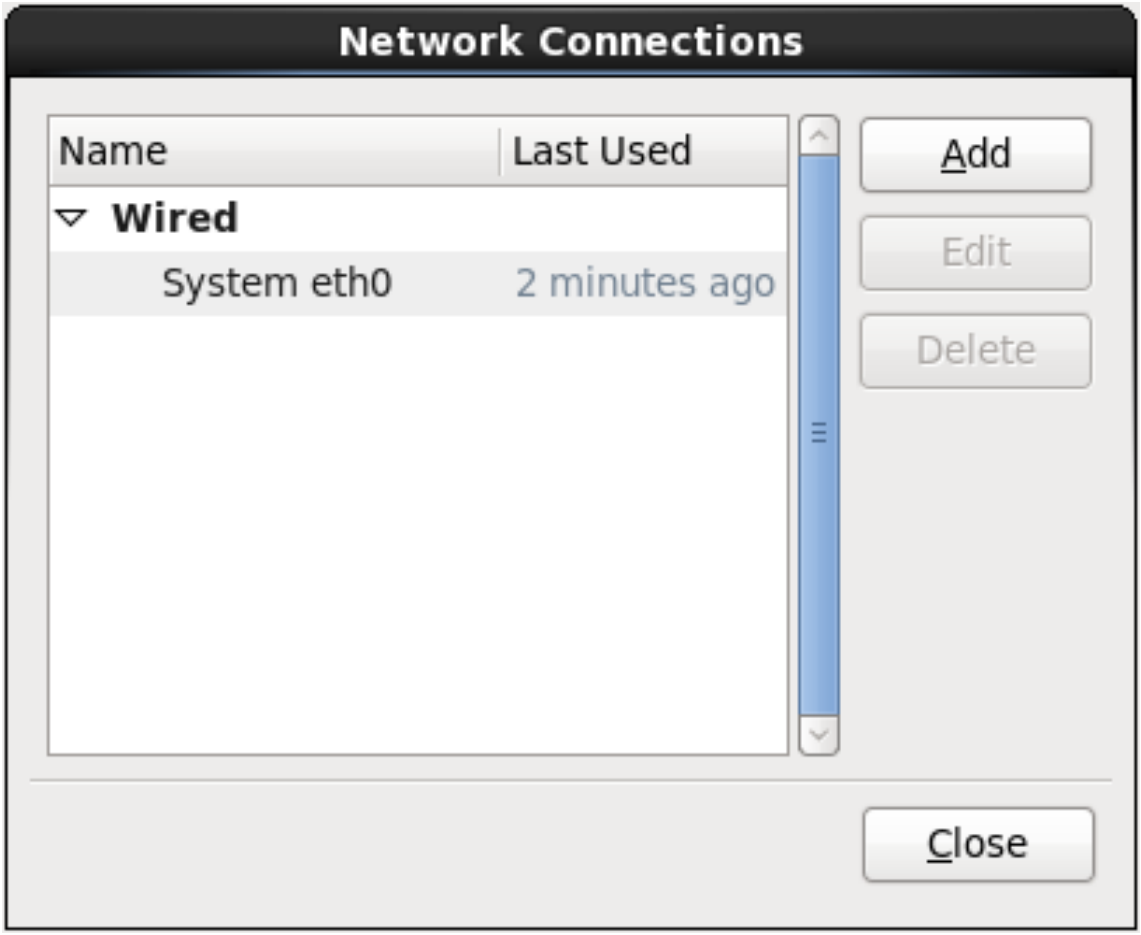


Figure 9.25. Network Connections

To add a new connection, click **Add** and select a connection type from the menu. To modify an existing connection, select it in the list and click **Edit**. In either case, a dialog box appears with a set of tabs that is appropriate to the particular connection type, as described below. To remove a connection, select it in the list and click **Delete**.

When you have finished editing network settings, click **Apply** to save the new configuration. If you reconfigured a device that was already active during installation, you must restart the device to use the new configuration – refer to Section 9.7.1.6, “Restart a network device” .

9.7.1.1. Options common to all types of connection

Certain configuration options are common to all connection types.

Specify a name for the connection in the **Connection name** name field.

Select **Connect automatically** to start the connection automatically when the system boots.

When **NetworkManager** runs on an installed system, the **Available to all users** option controls whether a network configuration is available system-wide or not. During installation, ensure that **Available to all users** remains selected for any network interface that you configure.

9.7.1.2. The Wired tab

Use the **Wired** tab to specify or change the *media access control* (MAC) address for the network adapter, and either set the *maximum transmission unit* (MTU, in bytes) that can pass through the interface.

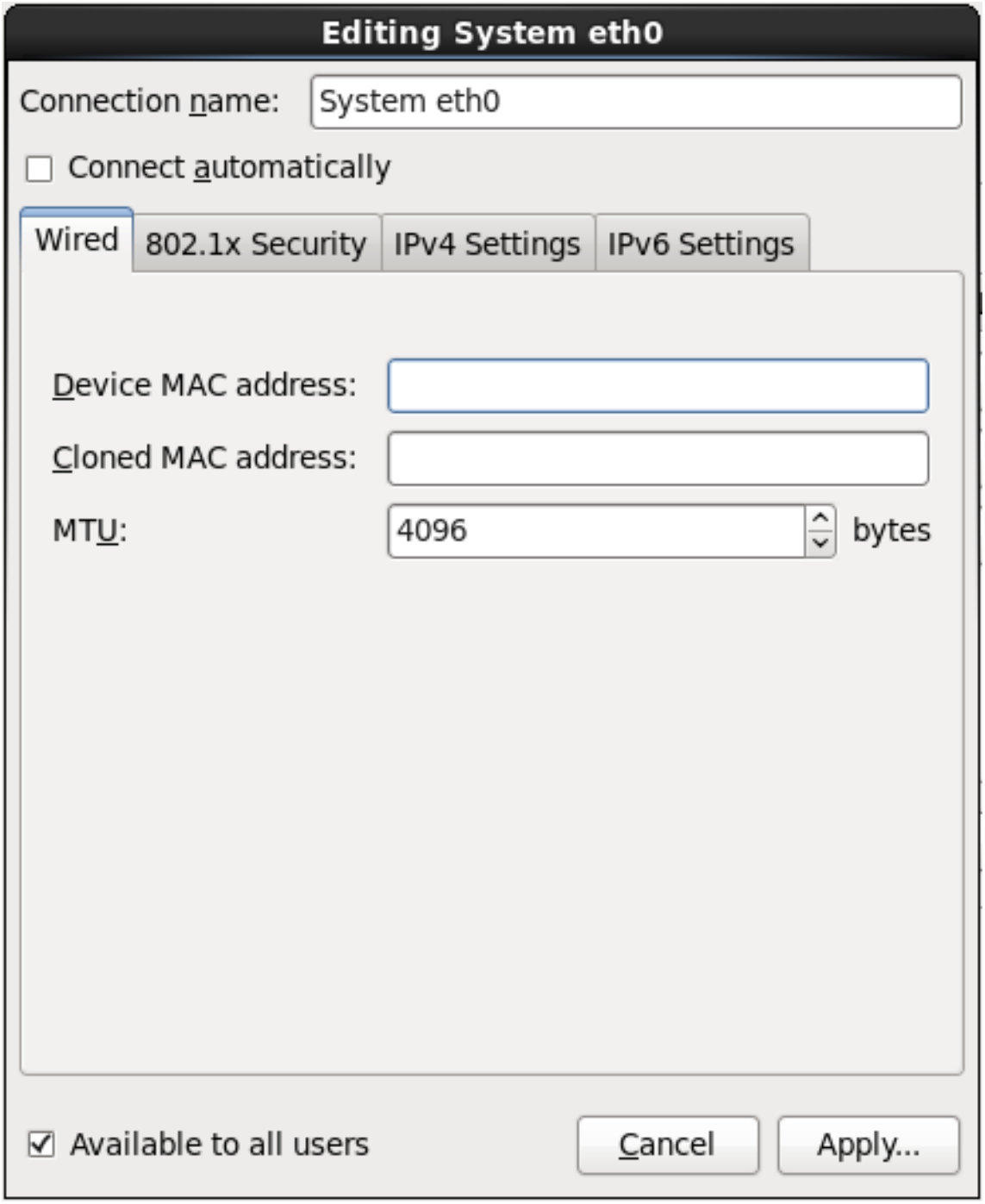


Figure 9.26. The Wired tab

9.7.1.3. The 802.1x Security tab

Use the **802.1x Security** tab to configure 802.1X *port-based network access control* (PNAC). Select **Use 802.1X security for this connection** to enable access control, then specify details of your network. The configuration options include:

Authentication

Choose one of the following methods of authentication:

- **TLS** for *Transport Layer Security*
- **Tunneled TLS** for *Tunneled Transport Layer Security*, otherwise known as TTLS, or EAP-TTLS

- **Protected EAP (PEAP)** for *Protected Extensible Authentication Protocol*

Identity

Provide the identity of this server.

User certificate

Browse to a personal X.509 certificate file encoded with *Distinguished Encoding Rules* (DER) or *Privacy Enhanced Mail* (PEM).

CA certificate

Browse to a X.509 *certificate authority* certificate file encoded with *Distinguished Encoding Rules* (DER) or *Privacy Enhanced Mail* (PEM).

Private key

Browse to a *private key* file encoded with *Distinguished Encoding Rules* (DER), *Privacy Enhanced Mail* (PEM), or the *Personal Information Exchange Syntax Standard* (PKCS#12).

Private key password

The password for the private key specified in the **Private key** field. Select **Show password** to make the password visible as you type it.

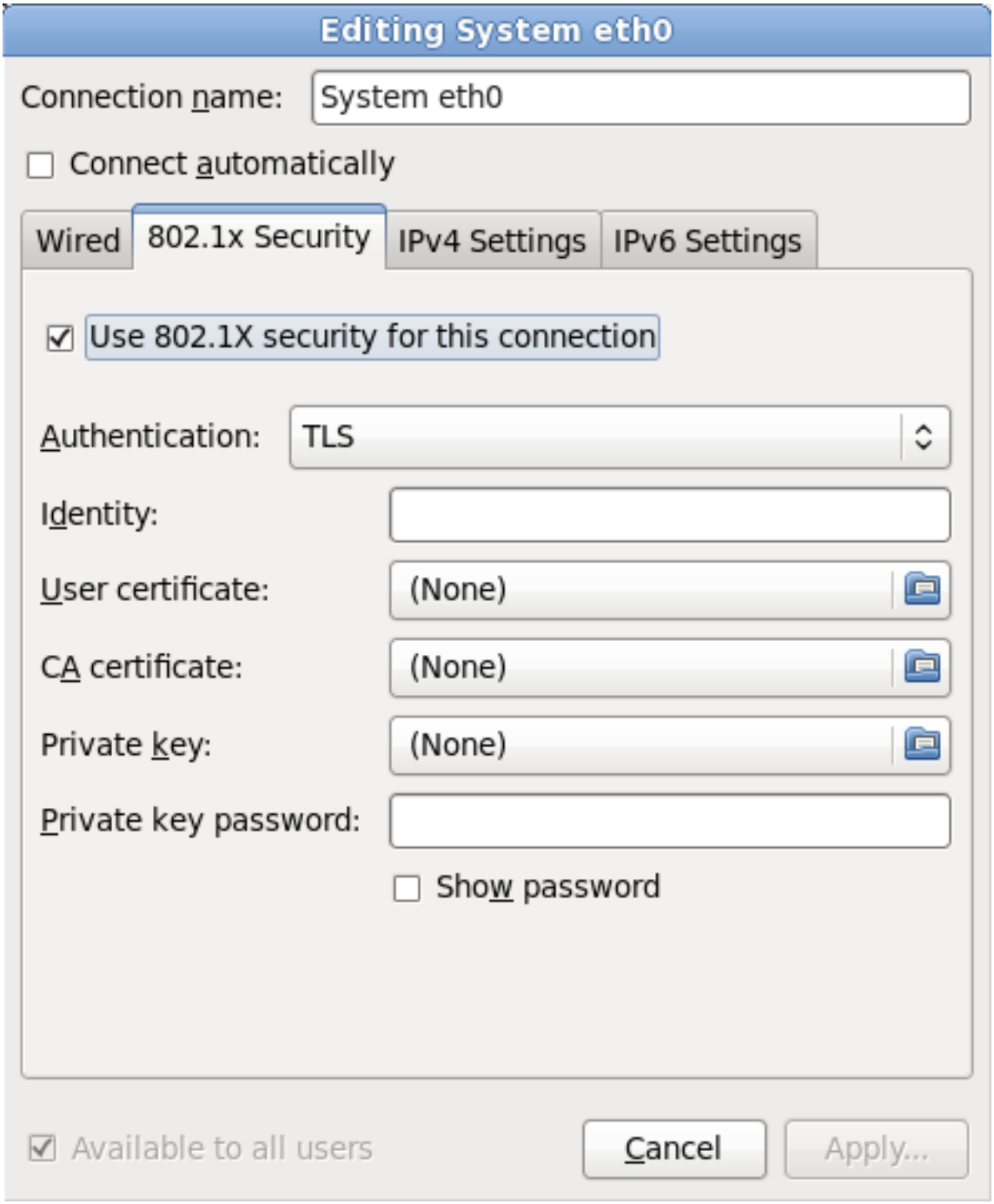


Figure 9.27. The 802.1x Security tab

9.7.1.4. The IPv4 Settings tab

Use the **IPv4 Settings tab** tab to configure the IPv4 parameters for the previously selected network connection.

Use the **Method** drop-down menu to specify which settings the system should attempt to obtain from a *Dynamic Host Configuration Protocol* (DHCP) service running on the network. Choose from the following options:

Automatic (DHCP)

IPv4 parameters are configured by the DHCP service on the network.

Automatic (DHCP) addresses only

The IPv4 address, netmask, and gateway address are configured by the DHCP service on the network, but DNS servers and search domains must be configured manually.

Manual

IPv4 parameters are configured manually for a static configuration.

Link-Local Only

A *link-local* address in the 169.254/16 range is assigned to the interface.

Shared to other computers

The system is configured to provide network access to other computers. The interface is assigned an address in the 10.42.x.1/24 range, a DHCP server and DNS server are started, and the interface is connected to the default network connection on the system with *network address translation* (NAT).

Disabled

IPv4 is disabled for this connection.

If you selected a method that requires you to supply manual parameters, enter details of the IP address for this interface, the netmask, and the gateway in the **Addresses** field. Use the **Add** and **Delete** buttons to add or remove addresses. Enter a comma-separated list of DNS servers in the **DNS servers** field, and a comma-separated list of domains in the **Search domains** field for any domains that you want to include in name server lookups.

Optionally, enter a name for this network connection in the **DHCP client ID** field. This name must be unique on the subnet. When you assign a meaningful DHCP client ID to a connection, it is easy to identify this connection when troubleshooting network problems.

Deselect the **Require IPv4 addressing for this connection to complete** check box to allow the system to make this connection on an IPv6-enabled network if IPv4 configuration fails but IPv6 configuration succeeds.

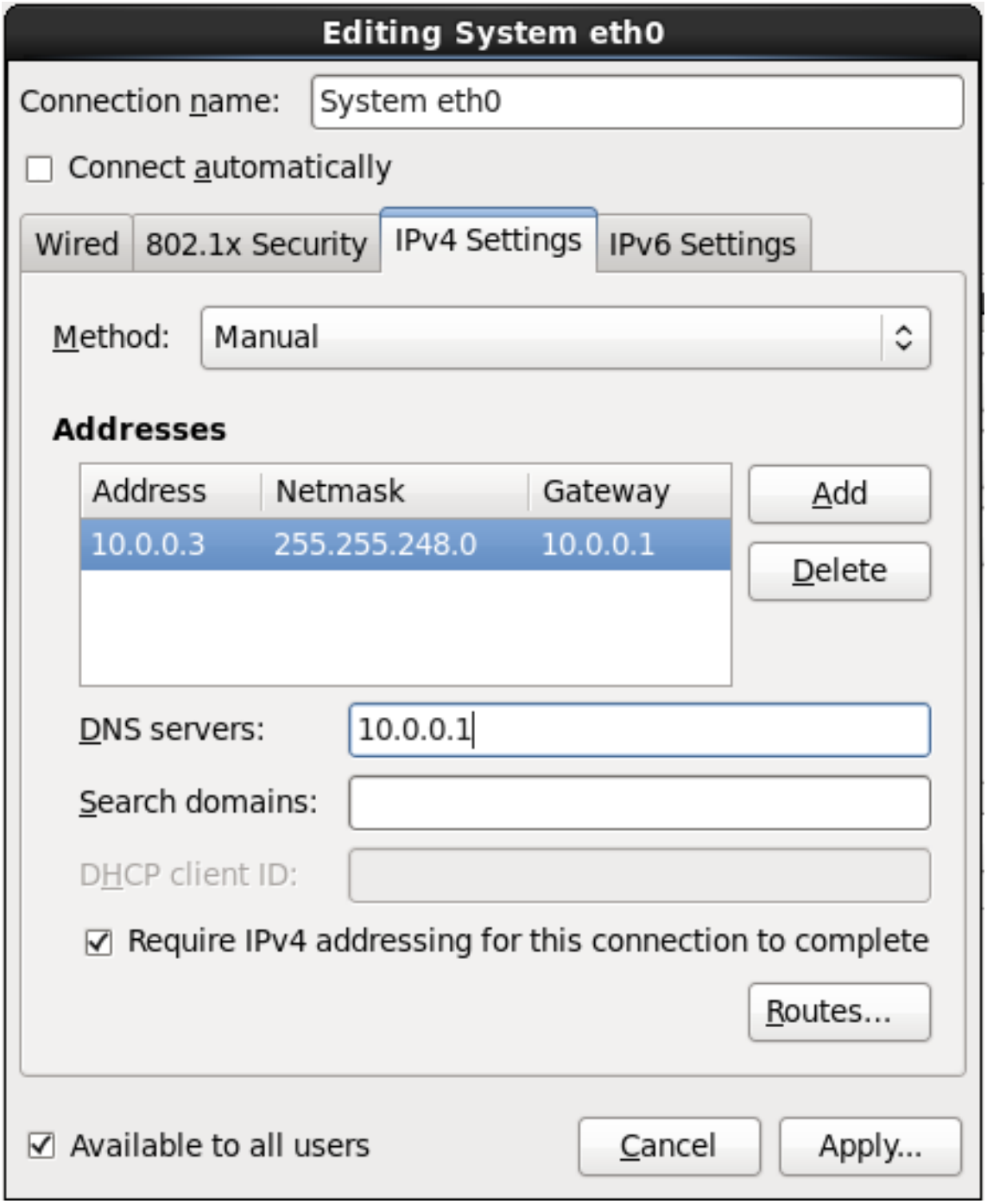


Figure 9.28. The IPv4 Settings tab

9.7.1.4.1. Editing IPv4 routes

Red Hat Enterprise Linux configures a number of routes automatically based on the IP addresses of a device. To edit additional routes, click the **Routes** button. The **Editing IPv4 routes** dialog appears.

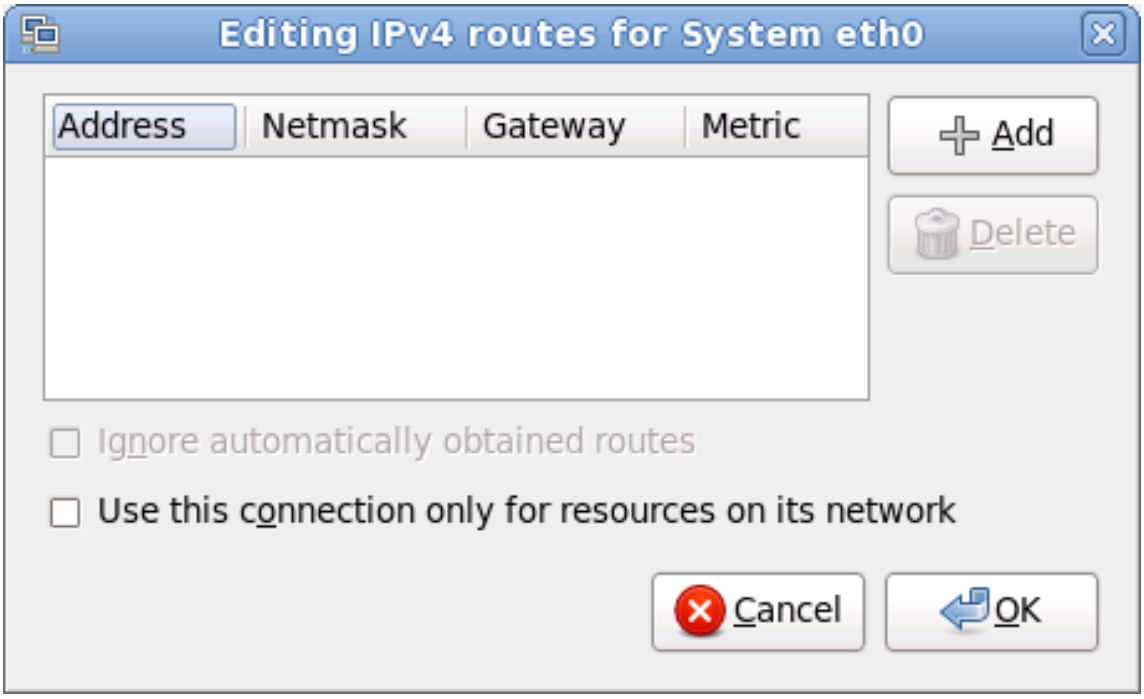


Figure 9.29. The Editing IPv4 Routes dialog

Click **Add** to add the IP address, netmask, gateway address, and metric for a new static route.

Select **Ignore automatically obtained routes** to make the interface use only the routes specified for it here.

Select **Use this connection only for resources on its network** to restrict connections only to the local network.

9.7.1.5. The IPv6 Settings tab

Use the **IPv6 Settings tab** tab to configure the IPv6 parameters for the previously selected network connection.

Use the **Method** drop-down menu to specify which settings the system should attempt to obtain from a *Dynamic Host Configuration Protocol* (DHCP) service running on the network. Choose from the following options:

Ignore

IPv6 is ignored for this connection.

Automatic

NetworkManager uses *router advertisement* (RA) to create an automatic, stateless configuration.

Automatic, addresses only

NetworkManager uses RA to create an automatic, stateless configuration, but DNS servers and search domains are ignored and must be configured manually.

Automatic, DHCP only

NetworkManager does not use RA, but requests information from DHCPv6 directly to create a stateful configuration.

Manual

IPv6 parameters are configured manually for a static configuration.

Link-Local Only

A *link-local* address with the fe80::/10 prefix is assigned to the interface.

If you selected a method that requires you to supply manual parameters, enter details of the IP address for this interface, the netmask, and the gateway in the **Addresses** field. Use the **Add** and **Delete** buttons to add or remove addresses. Enter a comma-separated list of DNS servers in the **DNS servers** field, and a comma-separated list of domains in the **Search domains** field for any domains that you want to include in name server lookups.

Optionally, enter a name for this network connection in the **DHCP client ID** field. This name must be unique on the subnet. When you assign a meaningful DHCP client ID to a connection, it is easy to identify this connection when troubleshooting network problems.

Deselect the **Require IPv6 addressing for this connection to complete** check box to allow the system to make this connection on an IPv4-enabled network if IPv6 configuration fails but IPv4 configuration succeeds.

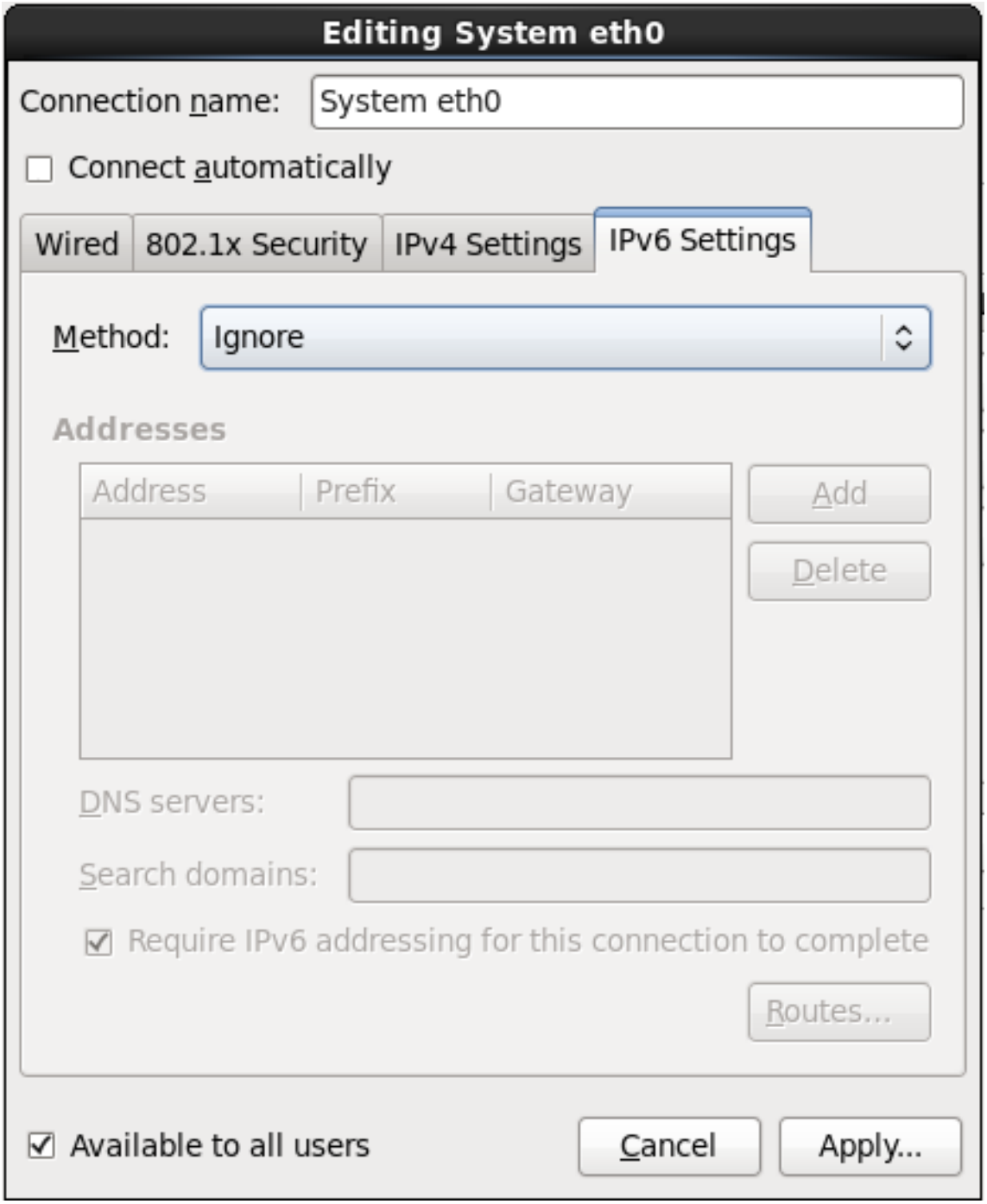


Figure 9.30. The IPv6 Settings tab

9.7.1.5.1. Editing IPv6 routes

Red Hat Enterprise Linux configures a number of routes automatically based on the IP addresses of a device. To edit additional routes, click the **Routes** button. The **Editing IPv6 routes** dialog appears.

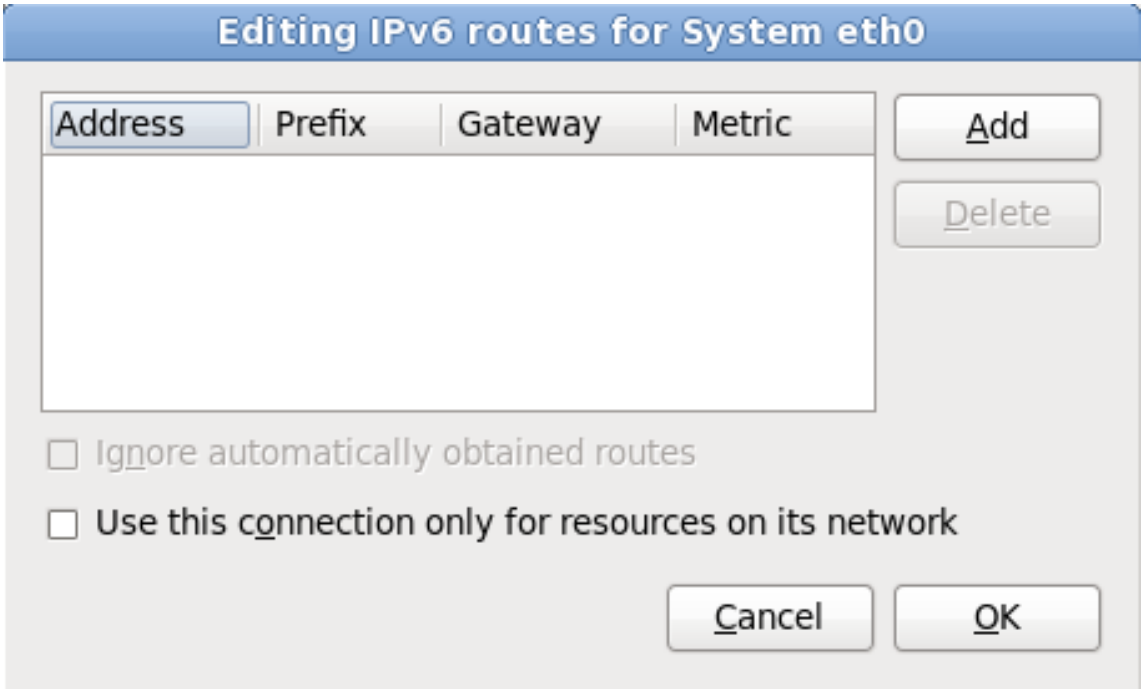


Figure 9.31. The Editing IPv6 Routes dialog

Click **Add** to add the IP address, netmask, gateway address, and metric for a new static route.

Select **Use this connection only for resources on its network** to restrict connections only to the local network.

9.7.1.6. Restart a network device

If you reconfigured a network that was already in use during installation, you must disconnect and reconnect the device in **anaconda** for the changes to take effect. **Anaconda** uses *interface configuration* (ifcfg) files to communicate with **NetworkManager**. A device becomes disconnected when its ifcfg file is removed, and becomes reconnected when its ifcfg file is restored, as long as **ONBOOT=yes** is set. Refer to the *Red Hat Enterprise Linux 6.9 Deployment Guide* available from https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/index.html for more information about interface configuration files.

1. Press **Ctrl**+**Alt**+**F2** to switch to virtual terminal **tty2**.

2. Move the interface configuration file to a temporary location:

   ```
   mv /etc/sysconfig/network-scripts/ifcfg-device_name /tmp
   ```

   where *device_name* is the device that you just reconfigured. For example, **ifcfg-eth0** is the ifcfg file for **eth0**.

   The device is now disconnected in **anaconda**.

3. Open the interface configuration file in the **vi** editor:

   ```
   vi /tmp/ifcfg-device_name
   ```

4. Verify that the interface configuration file contains the line **ONBOOT=yes**. If the file does not already contain the line, add it now and save the file.

5. Exit the **vi** editor.
6. Move the interface configuration file back to the **/etc/sysconfig/network-scripts/** directory:

   ```
   mv /tmp/ifcfg-device_name /etc/sysconfig/network-scripts/
   ```

   The device is now reconnected in **anaconda**.
7. Press **Ctrl**+**Alt**+**F6** to return to **anaconda**.

9.8. TIME ZONE CONFIGURATION

Set your time zone by selecting the city closest to your computer's physical location. Click on the map to zoom in to a particular geographical region of the world.

Specify a time zone even if you plan to use NTP (Network Time Protocol) to maintain the accuracy of the system clock.

From here there are two ways for you to select your time zone:

- Using your mouse, click on the interactive map to select a specific city (represented by a yellow dot). A red **X** appears indicating your selection.
- You can also scroll through the list at the bottom of the screen to select your time zone. Using your mouse, click on a location to highlight your selection.

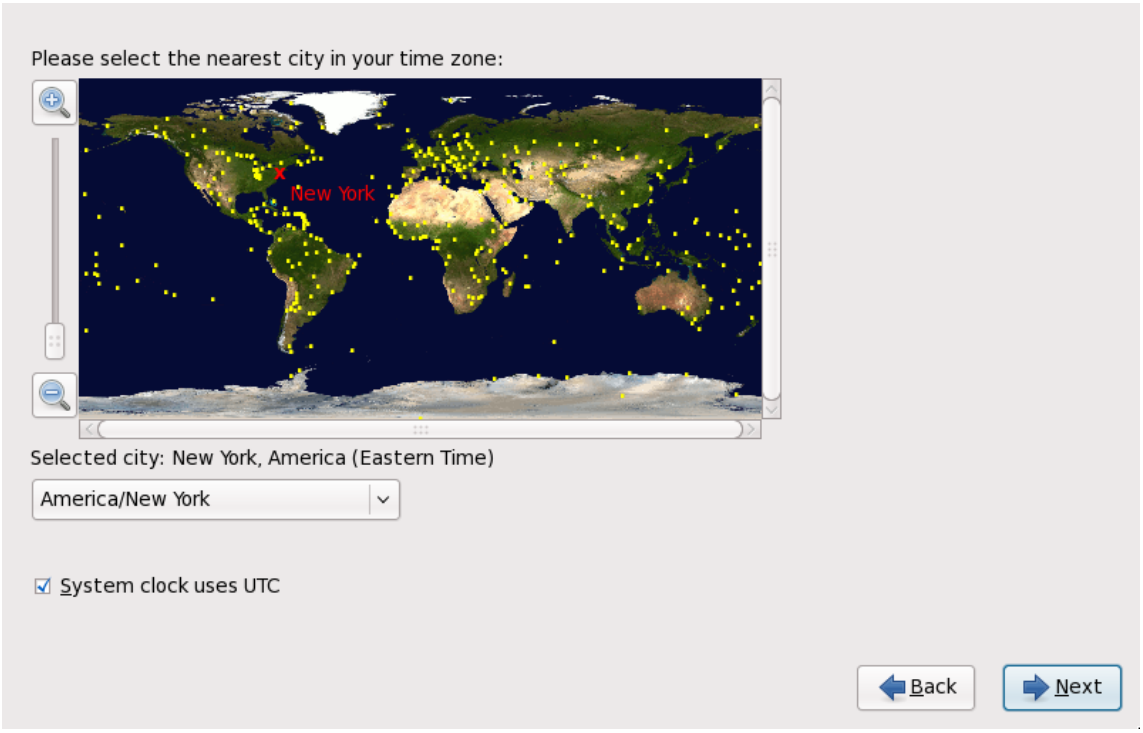


Figure 9.32. Configuring the Time Zone

If Red Hat Enterprise Linux is the only operating system on your computer, select **System clock uses UTC**. The system clock is a piece of hardware on your computer system. Red Hat Enterprise Linux uses the timezone setting to determine the offset between the local time and UTC on the system clock. This behavior is standard for systems that use UNIX, Linux, and similar operating systems.

Click **Next** to proceed.

WARNING

Do not enable the **System clock uses UTC** option if your machine also runs Microsoft Windows. Microsoft operating systems change the BIOS clock to match local time rather than UTC. This may cause unexpected behavior under Red Hat Enterprise Linux.

NOTE

To change your time zone configuration after you have completed the installation, use the **Time and Date Properties Tool**.

Type the **system-config-date** command in a shell prompt to launch the **Time and Date Properties Tool**. If you are not root, it prompts you for the root password to continue.

9.9. SET THE ROOT PASSWORD

Setting up a root account and password is one of the most important steps during your installation. The root account is used to install packages, upgrade RPMs, and perform most system maintenance. Logging in as root gives you complete control over your system.

NOTE

The root user (also known as the superuser) has complete access to the entire system; for this reason, logging in as the root user is best done *only* to perform system maintenance or administration.

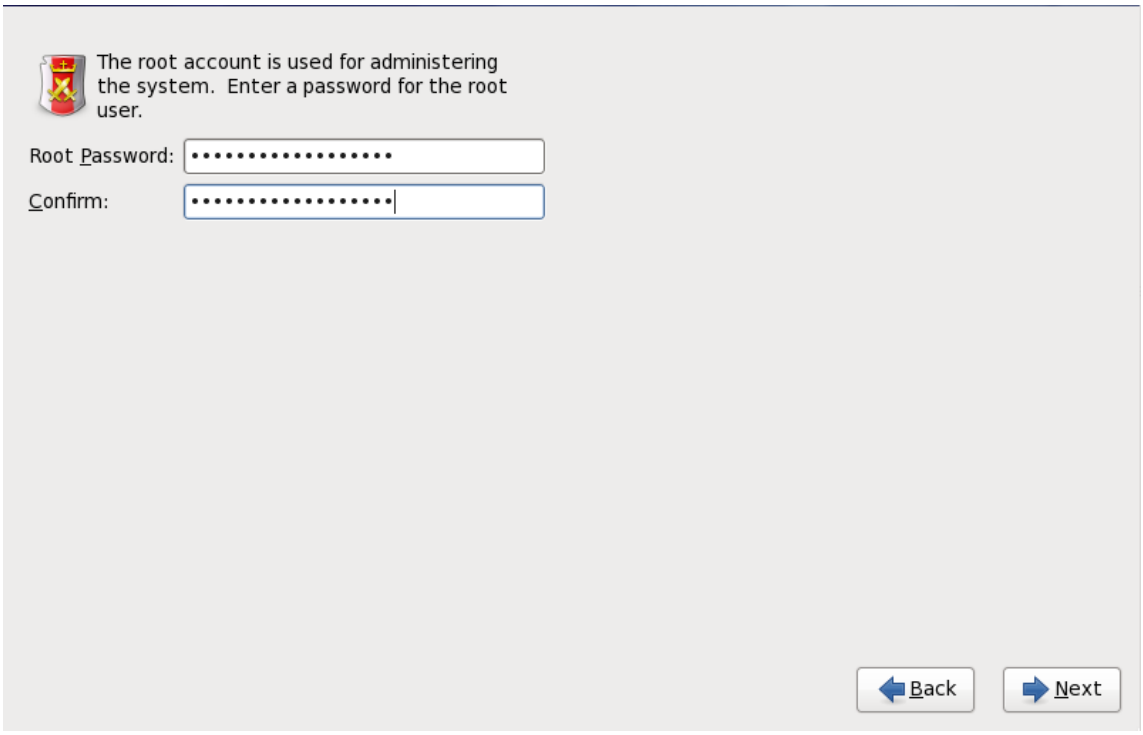
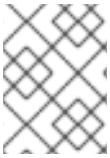


Figure 9.33. Root Password

Use the root account only for system administration. Create a non-root account for your general use and use the **su** command to change to root only when you need to perform tasks that require superuser authorization. These basic rules minimize the chances of a typo or an incorrect command doing damage to your system.



NOTE

To become root, type **su -** at the shell prompt in a terminal window and then press **Enter**. Then, enter the root password and press **Enter**.

The installation program prompts you to set a root password[2] for your system. . *You cannot proceed to the next stage of the installation process without entering a root password.*

The root password must be at least six characters long; the password you type is not echoed to the screen. You must enter the password twice; if the two passwords do not match, the installation program asks you to enter them again.

You should make the root password something you can remember, but not something that is easy for someone else to guess. Your name, your phone number, *qwerty*, *password, root*, *123456*, and *anteater* are all examples of bad passwords. Good passwords mix numerals with upper and lower case letters and do not contain dictionary words: *Aard387vark* or *420BMttNT*, for example. Remember that the password is case-sensitive. If you write down your password, keep it in a secure place. However, it is recommended that you do not write down this or any password you create.

WARNING

Do not use one of the example passwords offered in this manual. Using one of these passwords could be considered a security risk.

To change your root password after you have completed the installation, run the **passwd** command as **root**. If you forget the root password, see Resolving Problems in System Recovery Modes in the Red Hat Enterprise Linux 6 Deployment Guide for instructions on how to set a new one.

9.10. ASSIGN STORAGE DEVICES

If you selected more than one storage device on the storage devices selection screen (refer to Section 9.6, “Storage Devices”), **anaconda** asks you to select which of these devices should be available for installation of the operating system, and which should only be attached to the file system for data storage. If you selected only one storage device, **anaconda** does not present you with this screen.

During installation, the devices that you identify here as being for data storage only are mounted as part of the file system, but are not partitioned or formatted.

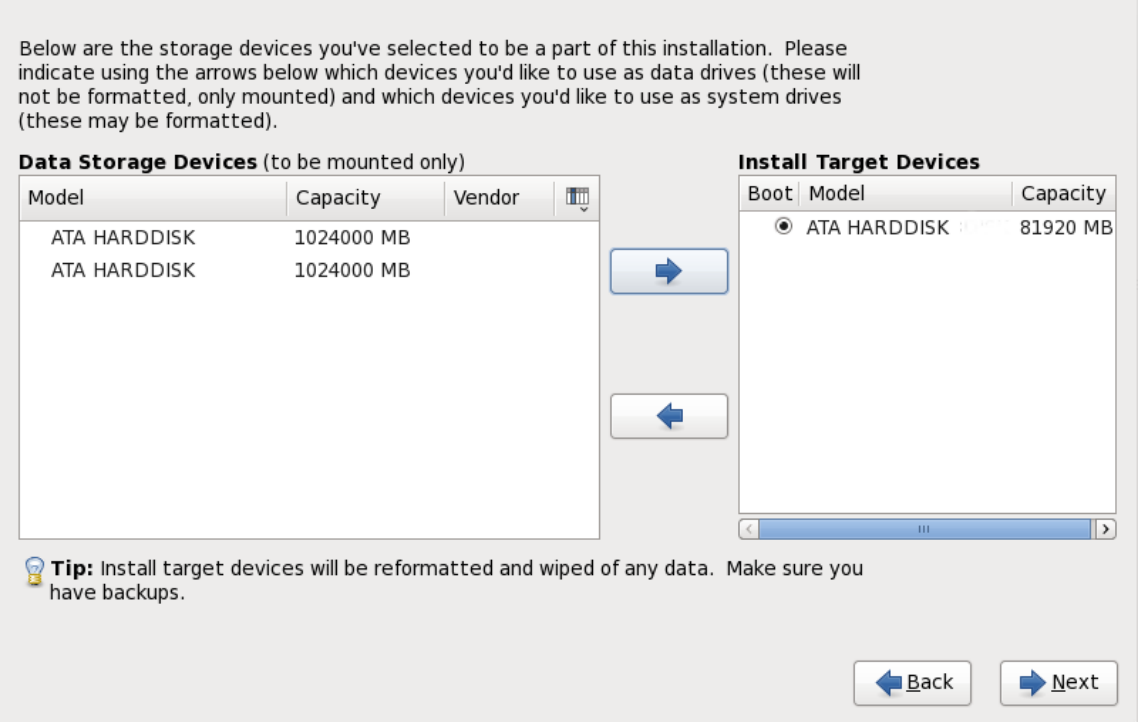


Figure 9.34. Assign storage devices

The screen is split into two panes. The left pane contains a list of devices to be used for data storage only. The right pane contains a list of devices that are to be available for installation of the operating system.

Each list contains information about the devices to help you to identify them. A small drop-down menu marked with an icon is located to the right of the column headings. This menu allows you to select the types of data presented on each device. Reducing or expanding the amount of information presented might help you to identify particular devices.

Move a device from one list to the other by clicking on the device, then clicking either the button labeled with a left-pointing arrow to move it to the list of data storage devices or the button labeled with a right-pointing arrow to move it to the list of devices available for installation of the operating system.

The list of devices available as installation targets also includes a radio button beside each device. Use this radio button to specify the device that you want to use as the boot device for the system.

IMPORTANT

If any storage device contains a boot loader that will chain load the Red Hat Enterprise Linux boot loader, include that storage device among the **Install Target Devices**. Storage devices that you identify as **Install Target Devices** remain visible to **anaconda** during boot loader configuration.

Storage devices that you identify as **Install Target Devices** on this screen are not automatically erased by the installation process unless you selected the **Use All Space** option on the partitioning screen (refer to Section 9.13, "Disk Partitioning Setup").

When you have finished identifying devices to be used for installation, click **Next** to continue.

9.11. INITIALIZING THE HARD DISK

If no readable partition tables are found on existing hard disks, the installation program asks to initialize

the hard disk. This operation makes any existing data on the hard disk unreadable. If your system has a brand new hard disk with no operating system installed, or you have removed all partitions on the hard disk, click **Re-initialize drive**.

The installation program presents you with a separate dialog for each disk on which it cannot read a valid partition table. Click the **Ignore all** button or **Re-initialize all** button to apply the same answer to all devices.

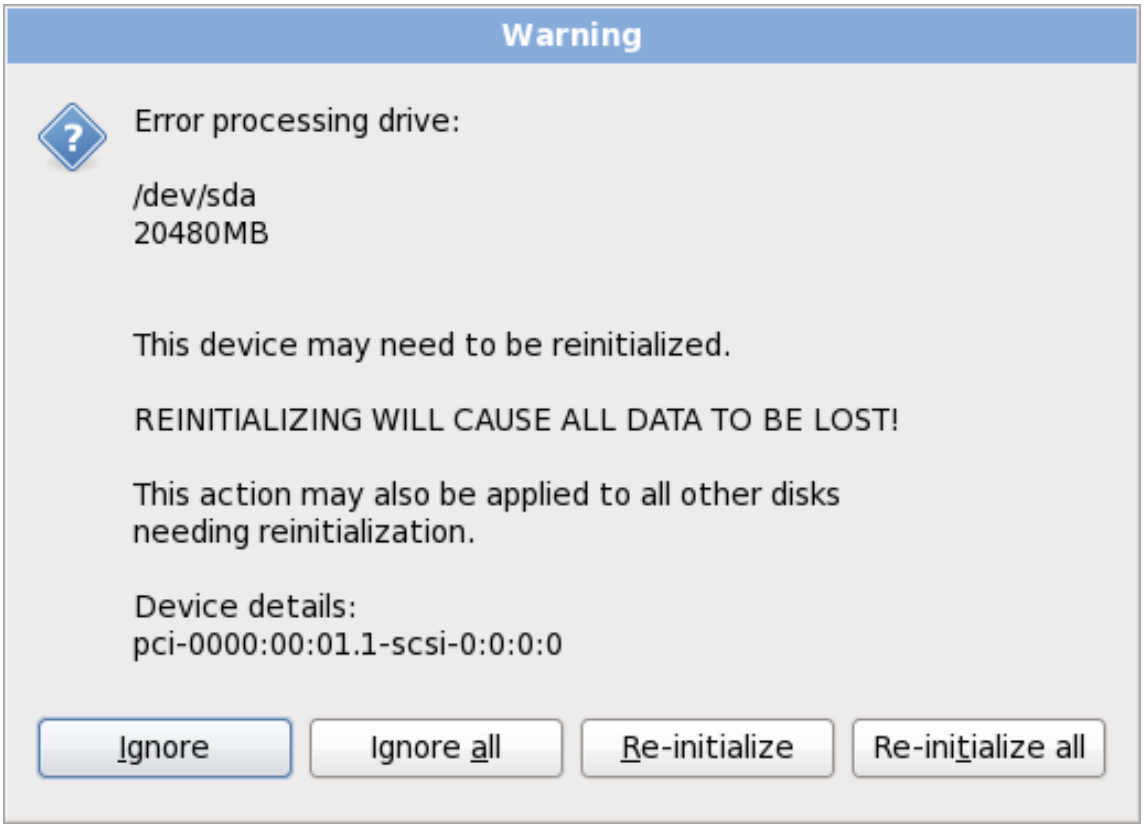


Figure 9.35. Warning screen – initializing hard drive

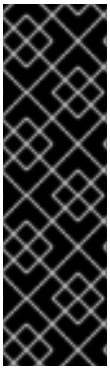
Certain RAID systems or other nonstandard configurations may be unreadable to the installation program and the prompt to initialize the hard disk may appear. The installation program responds to the physical disk structures it is able to detect.

To enable automatic initializing of hard disks for which it turns out to be necessary, use the kickstart command **zerombr** (refer to Chapter 32, *Kickstart Installations*). This command is required when performing an unattended installation on a system with previously initialized disks.

WARNING

If you have a nonstandard disk configuration that can be detached during installation and detected and configured afterward, power off the system, detach it, and restart the installation.

9.12. UPGRADING AN EXISTING SYSTEM



IMPORTANT

The following sections only apply to upgrading Red Hat Enterprise Linux between minor versions, for example, upgrading Red Hat Enterprise Linux 6.4 to Red Hat Enterprise Linux 6.5 or higher. This approach is not supported for upgrades between major versions, for example, upgrading Red Hat Enterprise Linux 6 to Red Hat Enterprise Linux 7.

In-place upgrades between major versions of Red Hat Enterprise Linux can be done, with certain limitations, using the **Red Hat Upgrade Tool** and **Preupgrade Assistant** tools. See Chapter 37, *Upgrading Your Current System* for more information.

The installation system automatically detects any existing installation of Red Hat Enterprise Linux. The upgrade process updates the existing system software with new versions, but does not remove any data from users' home directories. The existing partition structure on your hard drives does not change. Your system configuration changes only if a package upgrade demands it. Most package upgrades do not change system configuration, but rather install an additional configuration file for you to examine later.

Note that the installation medium that you are using might not contain all the software packages that you need to upgrade your computer.

9.12.1. The Upgrade Dialog

If your system contains a Red Hat Enterprise Linux installation, a dialog appears asking whether you want to upgrade that installation. To perform an upgrade of an existing system, choose the appropriate installation from the drop-down list and select **Next**.

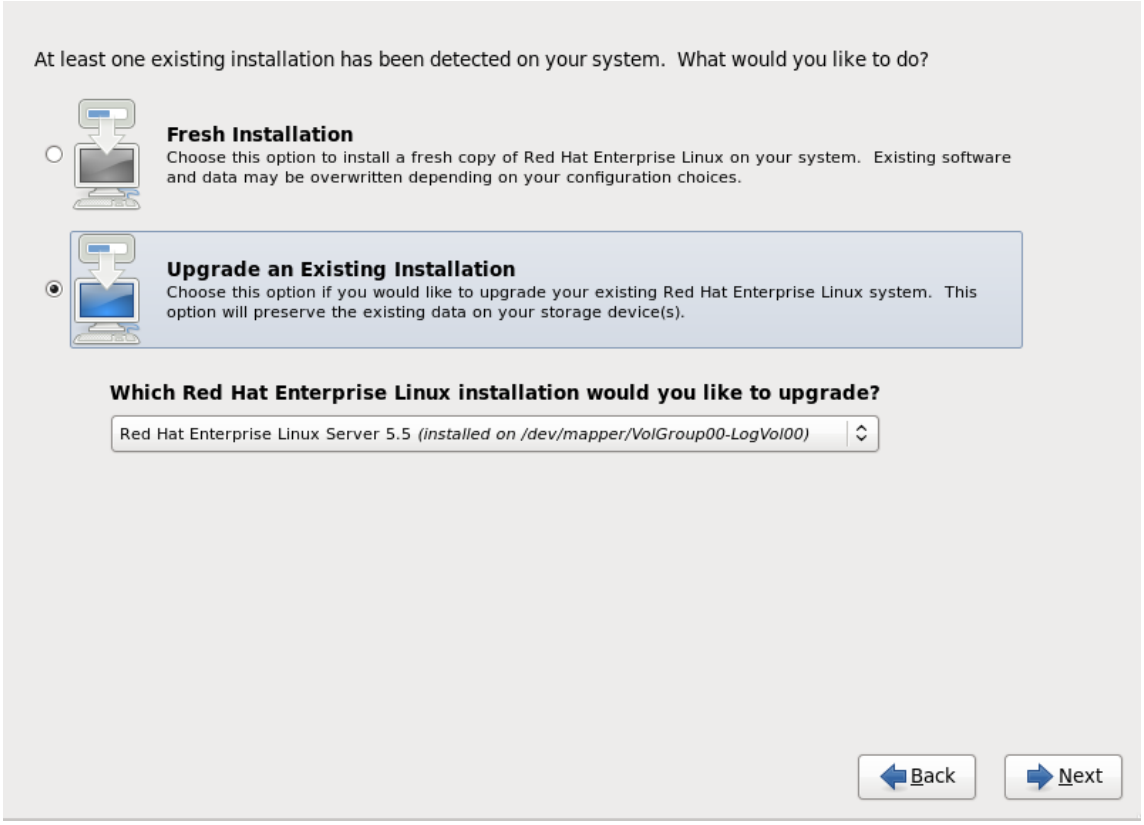
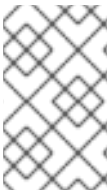


Figure 9.36. The Upgrade Dialog

NOTE

Software you have installed manually on your existing Red Hat Enterprise Linux system may behave differently after an upgrade. You may need to manually reinstall or recompile this software after an upgrade to ensure it performs correctly on the updated system.

9.12.2. Upgrading Using the Installer



NOTE

In general, Red Hat recommends that you keep user data on a separate /**home** partition and perform a fresh installation. For more information on partitions and how to set them up, refer to Section 9.13, "Disk Partitioning Setup" .

If you choose to upgrade your system using the installation program, any software not provided by Red Hat Enterprise Linux that conflicts with Red Hat Enterprise Linux software is overwritten. Before you begin an upgrade this way, make a list of your system's current packages for later reference:

```
rpm -qa --qf '%{NAME} %{VERSION}-%{RELEASE} %{ARCH}\n' > ~/old-pkglist.txt
```

After installation, consult this list to discover which packages you may need to rebuild or retrieve from sources other than Red Hat.

Next, make a backup of any system configuration data:

```
su -c 'tar czf /tmp/etc-`date +%F`.tar.gz /etc'
su -c 'mv /tmp/etc-*.tar.gz /home'
```

Make a complete backup of any important data before performing an upgrade. Important data may include the contents of your entire /**home** directory as well as content from services such as an Apache, FTP, or SQL server, or a source code management system. Although upgrades are not destructive, if you perform one improperly there is a small possibility of data loss.

WARNING

Note that the above examples store backup materials in a /**home** directory. If your /**home** directory is not a separate partition, *you should not follow these examples verbatim!* Store your backups on another device such as CD or DVD discs or an external hard disk.

For more information on completing the upgrade process later, refer to Section 35.2, "Finishing an Upgrade".

9.12.3. Updating the Boot Loader Configuration

Your completed Red Hat Enterprise Linux installation must be registered in the *boot loader* to boot properly. A boot loader is software on your machine that locates and starts the operating system. Refer to Appendix E, *The GRUB Boot Loader* for more information about boot loaders.

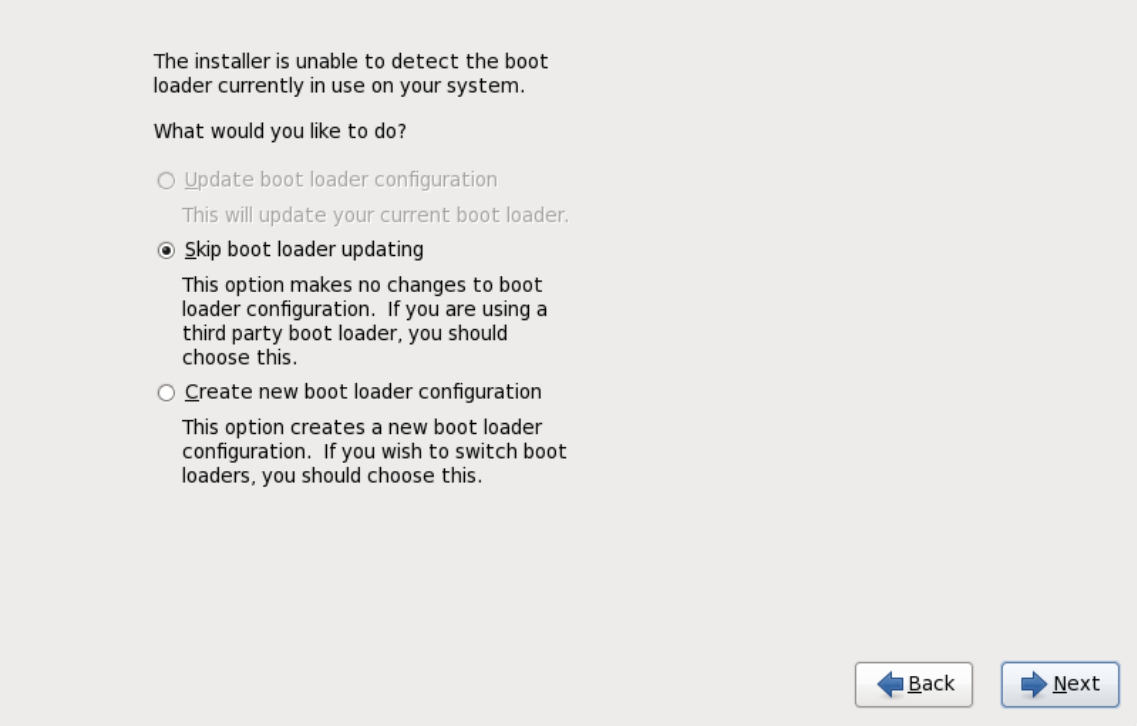


Figure 9.37. The Upgrade Boot Loader Dialog

If the existing boot loader was installed by a Linux distribution, the installation system can modify it to load the new Red Hat Enterprise Linux system. To update the existing Linux boot loader, select **Update boot loader configuration**. This is the default behavior when you upgrade an existing Red Hat Enterprise Linux installation.

GRUB is the standard boot loader for Red Hat Enterprise Linux on 32-bit and 64-bit x86 architectures. If your machine uses another boot loader, such as BootMagic, System Commander, or the loader installed by Microsoft Windows, then the Red Hat Enterprise Linux installation system cannot update it. In this case, select **Skip boot loader updating**. When the installation process completes, refer to the documentation for your product for assistance.

Install a new boot loader as part of an upgrade process only if you are certain you want to replace the existing boot loader. If you install a new boot loader, you may not be able to boot other operating systems on the same machine until you have configured the new boot loader. Select **Create new boot loader configuration** to remove the existing boot loader and install GRUB.

After you make your selection, click **Next** to continue. If you selected the **Create new boot loader configuration** option, refer to Section 9.18, “x86, AMD64, and Intel 64 Boot Loader Configuration” . If you chose to update or skip boot loader configuration, installation continues without further input from you.

9.13. DISK PARTITIONING SETUP

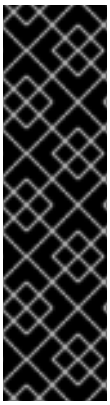
WARNING

It is always a good idea to back up any data that you have on your systems. For example, if you are upgrading or creating a dual-boot system, you should back up any data you wish to keep on your storage devices. Mistakes do happen and can result in the loss of all your data.

IMPORTANT

If you install Red Hat Enterprise Linux in text mode, you can only use the default partitioning schemes described in this section. You cannot add or remove partitions or file systems beyond those that the installer automatically adds or removes. If you require a customized layout at installation time, you should perform a graphical installation over a VNC connection or a kickstart installation.

Furthermore, advanced options such as LVM, encrypted filesystems, and resizable filesystems are available only in graphical mode and kickstart.



IMPORTANT

If you have a RAID card, be aware that some BIOS types do not support booting from the RAID card. In cases such as these, the /**boot**/ partition must be created on a partition outside of the RAID array, such as on a separate hard drive. An internal hard drive is necessary to use for partition creation with problematic RAID cards.

A /**boot**/ partition is also necessary for software RAID setups.

If you have chosen to automatically partition your system, you should select **Review** and manually edit your /**boot**/ partition.

Partitioning allows you to divide your hard drive into isolated sections, where each section behaves as its own hard drive. Partitioning is particularly useful if you run multiple operating systems. If you are not sure how you want your system to be partitioned, read Appendix A, *An Introduction to Disk Partitions* for more information.

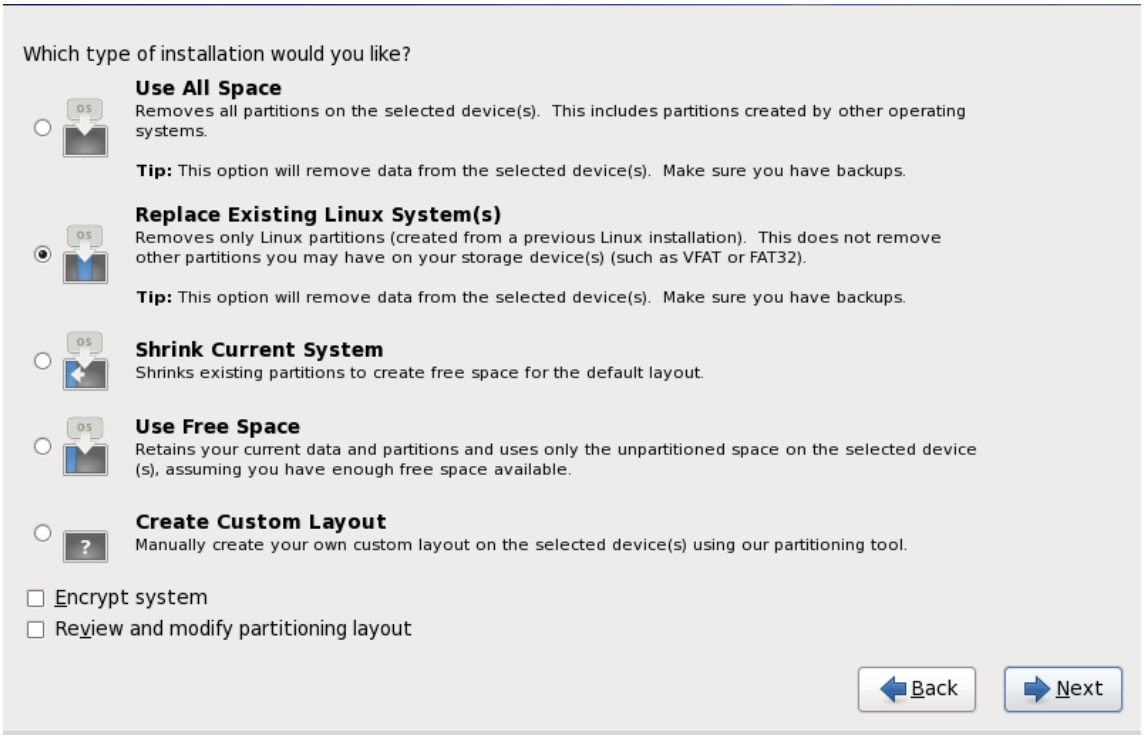


Figure 9.38. Disk Partitioning Setup

On this screen you can choose to create the default partition layout in one of four different ways, or choose to partition storage devices manually to create a custom layout.

The first four options allow you to perform an automated installation without having to partition your storage devices yourself. If you do not feel comfortable with partitioning your system, choose one of these options and let the installation program partition the storage devices for you. Depending on the option that you choose, you can still control what data (if any) is removed from the system.

Your options are:

Use All Space

Select this option to remove all partitions on your hard drives (this includes partitions created by other operating systems such as Windows VFAT or NTFS partitions).

WARNING

If you select this option, all data on the selected hard drives is removed by the installation program. Do not select this option if you have information that you want to keep on the hard drives where you are installing Red Hat Enterprise Linux.

In particular, do not select this option when you configure a system to chain load the Red Hat Enterprise Linux boot loader from another boot loader.

Replace Existing Linux System(s)

Select this option to remove only partitions created by a previous Linux installation. This does not remove other partitions you may have on your hard drives (such as VFAT or FAT32 partitions).

Shrink Current System

Select this option to resize your current data and partitions manually and install a default Red Hat Enterprise Linux layout in the space that is freed.

WARNING

If you shrink partitions on which other operating systems are installed, you might not be able to use those operating systems. Although this partitioning option does not destroy data, operating systems typically require some free space in their partitions. Before you resize a partition that holds an operating system that you might want to use again, find out how much space you need to leave free.

Use Free Space

Select this option to retain your current data and partitions and install Red Hat Enterprise Linux in the unused space available on the storage drives. Ensure that there is sufficient space available on the storage drives before you select this option – refer to Section 3.6, "Do You Have Enough Disk Space?".

WARNING

If your 64-bit x86 system uses UEFI instead of BIOS, you will need to manually create a /boot partition. This partition must have an ext3 file system. If you choose to partition automatically, your system will not boot.

Create Custom Layout

Select this option to partition storage devices manually and create customized layouts. Refer to Section 9.15, " Creating a Custom Layout or Modifying the Default Layout "

Choose your preferred partitioning method by clicking the radio button to the left of its description in the dialog box.

Select **Encrypt system** to encrypt all partitions except the **/boot** partition. Refer to Appendix C, *Disk Encryption* for information on encryption.

To review and make any necessary changes to the partitions created by automatic partitioning, select the **Review** option. After selecting **Review** and clicking **Next** to move forward, the partitions created for you by **anaconda** appear. You can make modifications to these partitions if they do not meet your needs.

IMPORTANT

To configure the Red Hat Enterprise Linux boot loader to *chain load* from a different boot loader, you must specify the boot drive manually. If you chose any of the automatic partitioning options, you must now select the **Review and modify partitioning layout** option before you click **Next** or you cannot specify the correct boot drive.

IMPORTANT

When you install Red Hat Enterprise Linux 6 on a system with multipath and non-multipath storage devices, the automatic partitioning layout in the installer might create volume groups that contain a mix of multipath and non-multipath devices. This defeats the purpose of multipath storage.

We advise that you select only multipath or only non-multipath devices on the disk selection screen that appears after selecting automatic partitioning. Alternatively, select custom partitioning.

Click **Next** once you have made your selections to proceed.

9.14. CHOOSING A DISK ENCRYPTION PASSPHRASE

If you selected the **Encrypt System** option, the installer prompts you for a passphrase with which to encrypt the partitions on the system.

Partitions are encrypted using the *Linux Unified Key Setup* – refer to Appendix C, *Disk Encryption* for more information.

Figure 9.39. Enter passphrase for encrypted partition

Choose a passphrase and type it into each of the two fields in the dialog box. You must provide this passphrase every time that the system boots.

WARNING

If you lose this passphrase, any encrypted partitions and the data on them will become completely inaccessible. There is no way to recover a lost passphrase.

Note that if you perform a kickstart installation of Red Hat Enterprise Linux, you can save encryption passphrases and create backup encryption passphrases during installation. Refer to Section C.3.2, "Saving Passphrases" and Section C.3.3, "Creating and Saving Backup Passphrases".

9.15. CREATING A CUSTOM LAYOUT OR MODIFYING THE DEFAULT LAYOUT

If you chose one of the four automatic partitioning options and did not select **Review**, skip ahead to Section 9.17, "Package Group Selection".

If you chose one of the automatic partitioning options and selected **Review**, you can either accept the current partition settings (click **Next**), or modify the setup manually in the partitioning screen.

If you chose to create a custom layout, you must tell the installation program where to install Red Hat Enterprise Linux. This is done by defining mount points for one or more disk partitions in which Red Hat Enterprise Linux is installed. You may also need to create and/or delete partitions at this time.

WARNING

If your 64-bit x86 system uses UEFI instead of BIOS, you will need to manually create a /boot partition. This partition must have an ext3 file system. If you choose to partition automatically, your system will not boot.

IMPORTANT

On systems using UEFI firmware, the boot drive (the disk where the boot loader will be installed) must contain a special partition (EFI System Partition) at least 50 MB in size with a mount point of **/boot/efi**.

The boot drive must also have a GUID Partition Table (GPT) label. If you want to reuse a disk with existing partitions and a Master Boot Record (MBR) label, the disk must be relabeled. *All existing data on the disk will be lost.*

To relabel a disk to GPT in the graphical installer, first go back to Section 9.13, "Disk Partitioning Setup", and choose an automatic partitioning option such as **Use All Space**. Check the **Review and modify partitioning layout** check box, and click **Next**. On the next screen, modify the automatically created layout as needed.

This workaround is always necessary when reusing a MBR-labeled drive. If you choose **Create a Custom Layout** at the start of the partitioning process, the disk will not be relabeled and you will not be able to proceed.

If you have not yet planned how to set up your partitions, refer to Appendix A, *An Introduction to Disk Partitions* and Section 9.15.5, "Recommended Partitioning Scheme". At a bare minimum, you need an appropriately-sized root partition, and usually a swap partition appropriate to the amount of RAM you have on the system.

Anaconda can handle the partitioning requirements for a typical installation.

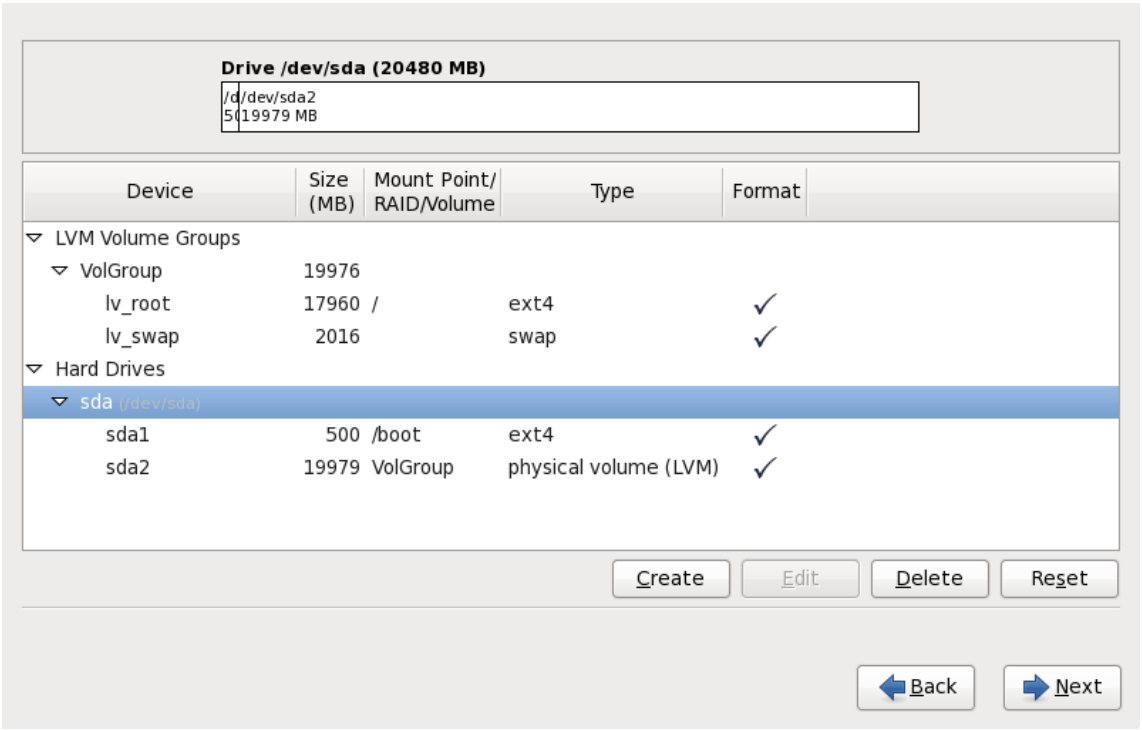


Figure 9.40. Partitioning on x86, AMD64, and Intel 64 Systems

The partitioning screen contains two panes. The top pane contains a graphical representation of the hard drive, logical volume, or RAID device selected in the lower pane.

Above the graphical representation of the device, you can review the name of the drive (such as **/dev/sda** or **LogVol00**), its size (in MB), and its model as detected by the installation program.

Using your mouse, click once to highlight a particular field in the graphical display. Double-click to edit an existing partition or to create a partition out of existing free space.

The lower pane contains a list of all drives, logical volumes, and RAID devices to be used during installation, as specified earlier in the installation process – refer to Section 9.10, " Assign Storage Devices "

Devices are grouped by type. Click on the small triangles to the left of each device type to view or hide devices of that type.

Anaconda displays several details for each device listed:

Device

the name of the device, logical volume, or partition

Size (MB)

the size of the device, logical volume, or partition (in MB)

Mount Point/RAID/Volume

the *mount point* (location within a file system) on which a partition is to be mounted, or the name of the RAID or logical volume group of which it is a part

Type

the type of partition. If the partition is a standard partition, this field displays the type of file system on the partition (for example, ext4). Otherwise, it indicates that the partition is a **physical volume (LVM)**, or part of a **software RAID**

Format

A check mark in this column indicates that the partition will be formatted during installation.

Beneath the lower pane are four buttons: **Create**, **Edit**, **Delete**, and **Reset**.

Select a device or partition by clicking on it in either the graphical representation in the upper pane of in the list in the lower pane, then click one of the four buttons to carry out the following actions:

Create

create a new partition, logical volume, or software RAID

Edit

change an existing partition, logical volume, or software RAID. Note that you can only shrink partitions with the **Resize** button, not enlarge partitions.

Delete

remove a partition, logical volume, or software RAID

Reset

undo all changes made in this screen

9.15.1. Create Storage

The **Create Storage** dialog allows you to create new storage partitions, logical volumes, and software RAIDs. **Anaconda** presents options as available or unavailable depending on the storage already present on the system or configured to transfer to the system.

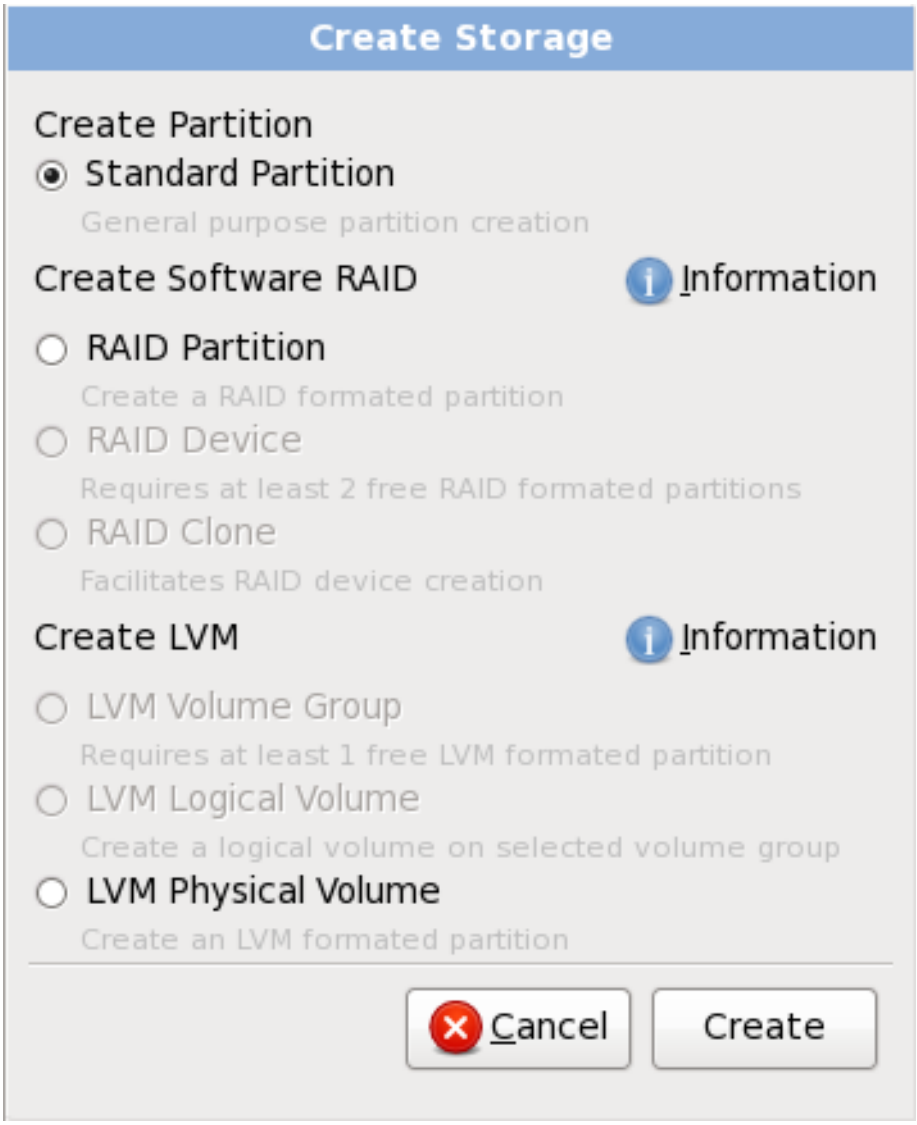


Figure 9.41. Creating Storage

Options are grouped under **Create Partition**, **Create Software RAID** and **Create LVM** as follows:

Create Partition

Refer to Section 9.15.2, "Adding Partitions" for details of the **Add Partition** dialog.

- **Standard Partition** – create a standard disk partition (as described in Appendix A, *An Introduction to Disk Partitions*) in unallocated space.

Create Software RAID

Refer to Section 9.15.3, " Create Software RAID " for more detail.

- **RAID Partition** – create a partition in unallocated space to form part of a software RAID device. To form a software RAID device, two or more RAID partitions must be available on the system.

- **RAID Device** – combine two or more RAID partitions into a software RAID device. When you choose this option, you can specify the type of RAID device to create (the *RAID level*). This option is only available when two or more RAID partitions are available on the system.

Create LVM Logical Volume

Refer to Section 9.15.4, " Create LVM Logical Volume " for more detail.

- **LVM Physical Volume** – create a *physical volume* in unallocated space.
- **LVM Volume Group** – create a *volume group* from one or more physical volumes. This option is only available when at least one physical volume is available on the system.
- **LVM Logical Volume** – create a *logical volume* on a volume group. This option is only available when at least one volume group is available on the system.

9.15.2. Adding Partitions

To add a new partition, select the **Create** button. A dialog box appears (refer to Figure 9.42, "Creating a New Partition").

NOTE

You must dedicate at least one partition for this installation, and optionally more. For more information, refer to Appendix A, *An Introduction to Disk Partitions*.

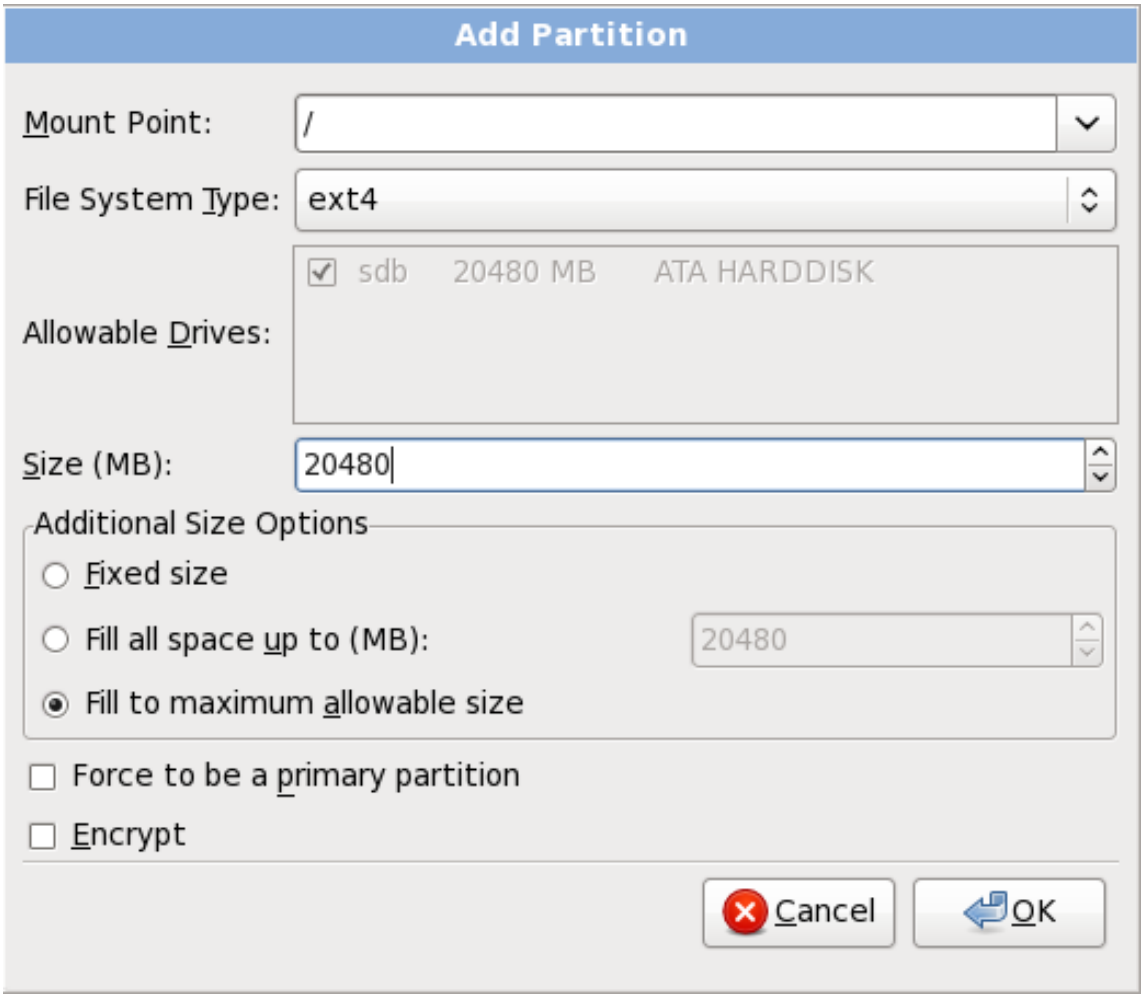


Figure 9.42. Creating a New Partition

- **Mount Point**: Enter the partition's mount point. For example, if this partition should be the root partition, enter /; enter **/boot** for the **/boot** partition, and so on. You can also use the pull-down menu to choose the correct mount point for your partition. For a swap partition the mount point should not be set – setting the filesystem type to **swap** is sufficient.

- **File System Type**: Using the pull-down menu, select the appropriate file system type for this partition. For more information on file system types, refer to Section 9.15.2.1, "File System Types".

- **Allowable Drives**: This field contains a list of the hard disks installed on your system. If a hard disk's box is highlighted, then a desired partition can be created on that hard disk. If the box is *not* checked, then the partition will *never* be created on that hard disk. By using different checkbox settings, you can have **anaconda** place partitions where you need them, or let **anaconda** decide where partitions should go.

- **Size (MB)**: Enter the size (in megabytes) of the partition. Note, this field starts with 200 MB; unless changed, only a 200 MB partition will be created.

- **Additional Size Options**: Choose whether to keep this partition at a fixed size, to allow it to "grow" (fill up the available hard drive space) to a certain point, or to allow it to grow to fill any remaining hard drive space available.

  If you choose **Fill all space up to (MB)**, you must give size constraints in the field to the right of this option. This allows you to keep a certain amount of space free on your hard drive for future use.

- **Force to be a primary partition**: Select whether the partition you are creating should be one of the first four partitions on the hard drive. If unselected, the partition is created as a logical partition. Refer to Section A.1.3, "Partitions Within Partitions – An Overview of Extended Partitions", for more information.

- **Encrypt**: Choose whether to encrypt the partition so that the data stored on it cannot be accessed without a passphrase, even if the storage device is connected to another system. Refer to Appendix C, *Disk Encryption* for information on encryption of storage devices. If you select this option, the installer prompts you to provide a passphrase before it writes the partition to the disk.

- **OK**: Select **OK** once you are satisfied with the settings and wish to create the partition.

- **Cancel**: Select **Cancel** if you do not want to create the partition.

9.15.2.1. File System Types

Red Hat Enterprise Linux allows you to create different partition types and file systems. The following is a brief description of the different partition types and file systems available, and how they can be used.

Partition types

- **standard partition** – A standard partition can contain a file system or swap space, or it can provide a container for software RAID or an LVM physical volume.

- **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. Refer to the Red Hat Enterprise Linux Deployment Guide for additional information.

- **software RAID** – Creating two or more software RAID partitions allows you to create a RAID device. For more information regarding RAID, refer to the chapter *RAID (Redundant Array of Independent Disks)* in the Red Hat Enterprise Linux Deployment Guide .

- **physical volume (LVM)** – Creating one or more physical volume (LVM) partitions allows you to create an LVM logical volume. LVM can improve performance when using physical disks. For more information regarding LVM, refer to the Red Hat Enterprise Linux Deployment Guide .

File systems

- **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. A maximum file system size of 16TB is supported for ext4. The ext4 file system is selected by default and is highly recommended.

NOTE

The **user_xattr** and **acl** mount options are automatically set on ext4 systems by the installation system. These options enable extended attributes and access control lists, respectively. More information about mount options can be found in the Red Hat Enterprise Linux Storage Administration Guide .

- **ext3** – The ext3 file system is based on the ext2 file system and has one main advantage – journaling. Using a journaling file system reduces time spent recovering a file system after a crash as there is no need to **fsck** [3] the file system. A maximum file system size of 16TB is

supported for ext3.

- **ext2** – An ext2 file system supports standard Unix file types (regular files, directories, symbolic links, etc). It provides the ability to assign long file names, up to 255 characters.
- **xfs** – XFS is a highly scalable, high-performance file system that supports filesystems 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 supports metadata journaling, which facilitates quicker crash recovery. The XFS file system can also be defragmented and resized while mounted and active.

NOTE

The maximum size of an XFS partition the installer can create is *100 TB*.

- **vfat** – The VFAT file system is a Linux file system that is compatible with Microsoft Windows long filenames on the FAT file system.
- **Btrfs** – Btrfs is under development as a file system capable of addressing and managing more files, larger files, and larger volumes than the ext2, ext3, and ext4 file systems. Btrfs is designed to make the file system tolerant of errors, and to facilitate the detection and repair of errors when they occur. It uses checksums to ensure the validity of data and metadata, and maintains snapshots of the file system that can be used for backup or repair.

  Because Btrfs is still experimental and under development, the installation program does not offer it by default. If you want to create a Btrfs partition on a drive, you must commence the installation process with the boot option **btrfs**. Refer to Chapter 28, *Boot Options* for instructions.

WARNING

Red Hat Enterprise Linux 6.9 includes Btrfs as a technology preview to allow you to experiment with this file system. You should not choose Btrfs for partitions that will contain valuable data or that are essential for the operation of important systems.

9.15.3. Create Software RAID

Redundant arrays of independent disks (RAIDs) are constructed from multiple storage devices that are arranged to provide increased performance and – in some configurations – greater fault tolerance. Refer to the Red Hat Enterprise Linux Storage Administration Guide for a description of different kinds of RAIDs.

To make a RAID device, you must first create software RAID partitions. Once you have created two or more software RAID partitions, select **RAID** to join the software RAID partitions into a RAID device.

RAID Partition

Choose this option to configure a partition for software RAID. This option is the only choice available if your disk contains no software RAID partitions. This is the same dialog that appears when you add a standard partition – refer to Section 9.15.2, "Adding Partitions" for a description of the available

options. Note, however, that **File System Type** must be set to **software RAID**

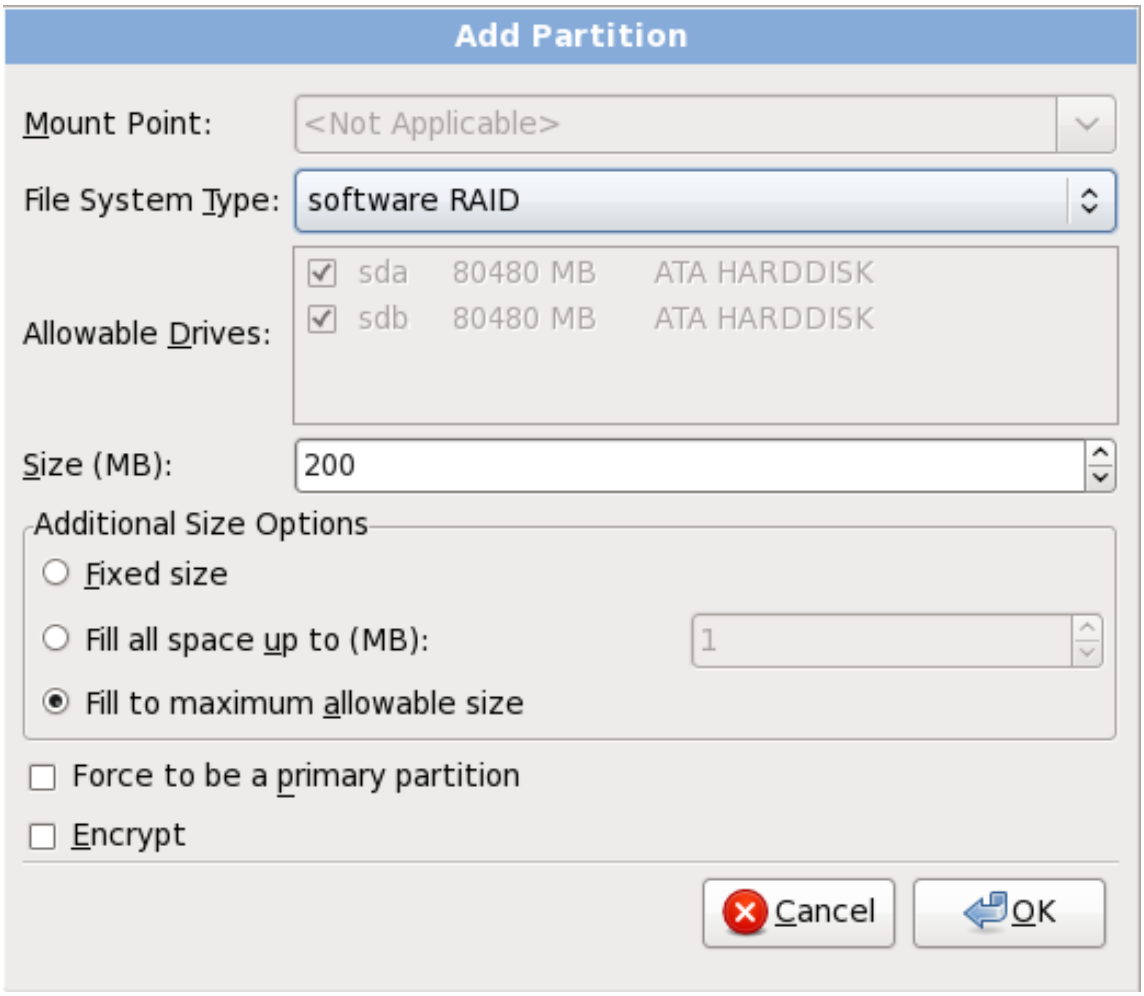


Figure 9.43. Create a software RAID partition

RAID Device

Choose this option to construct a RAID device from two or more existing software RAID partitions. This option is available if two or more software RAID partitions have been configured.

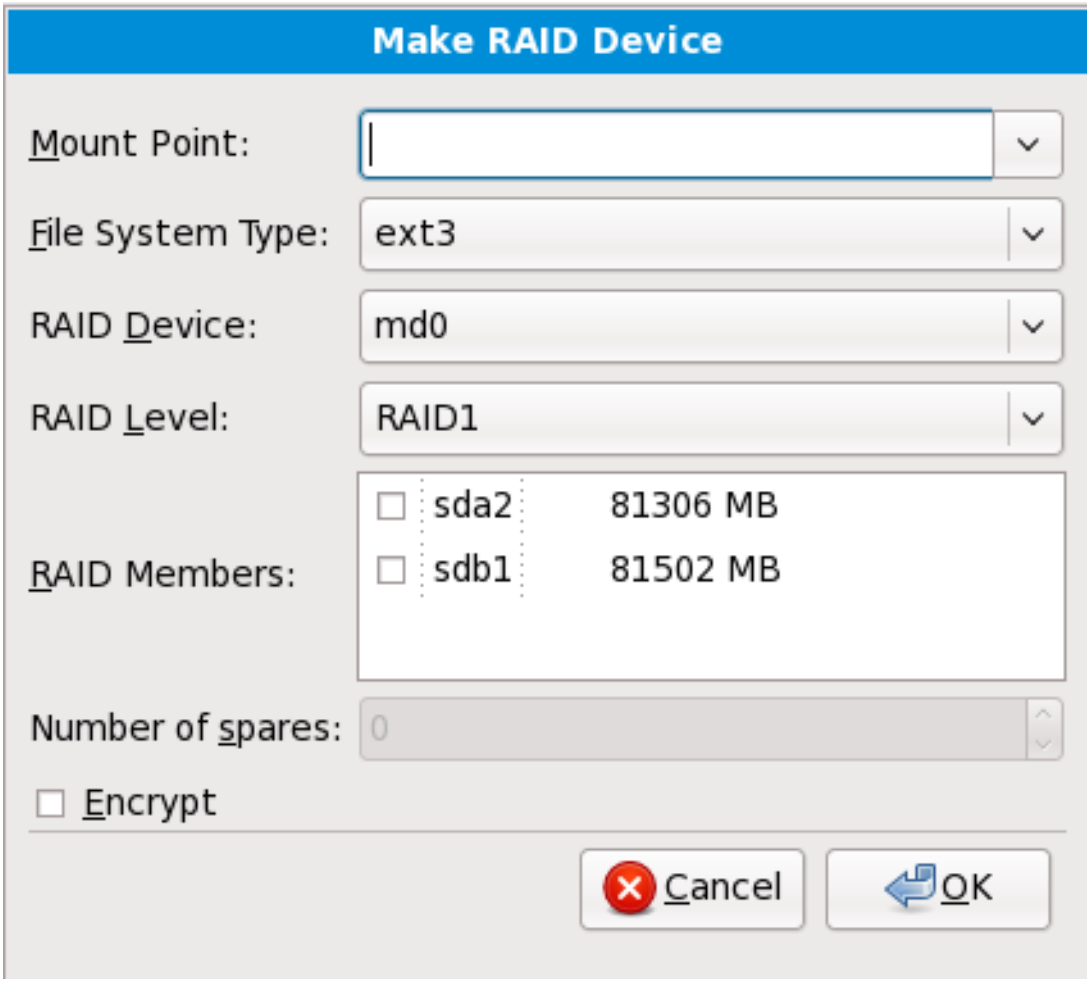


Figure 9.44. Create a RAID device

Select the file system type as for a standard partition.

Anaconda automatically suggests a name for the RAID device, but you can manually select names from **md0** to **md15**.

Click the checkboxes beside individual storage devices to include or remove them from this RAID.

The **RAID Level** corresponds to a particular type of RAID. Choose from the following options:

- **RAID 0** – distributes data across multiple storage devices. Level 0 RAIDs offer increased performance over standard partitions, and can be used to pool the storage of multiple devices into one large virtual device. Note that Level 0 RAIDS offer no redundancy and that the failure of one device in the array destroys the entire array. RAID 0 requires at least two RAID partitions.

- **RAID 1** – mirrors the data on one storage device onto one or more other storage devices. Additional devices in the array provide increasing levels of redundancy. RAID 1 requires at least two RAID partitions.

- **RAID 4** – distributes data across multiple storage devices, but uses one device in the array to store parity information that safeguards the array in case any device within the array fails. Because all parity information is stored on the one device, access to this device creates a bottleneck in the performance of the array. RAID 4 requires at least three RAID partitions.

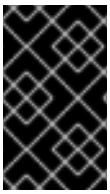
- **RAID 5** – distributes data and parity information across multiple storage devices. Level 5 RAIDs therefore offer the performance advantages of distributing data across multiple

devices, but do not share the performance bottleneck of level 4 RAIDs because the parity information is also distributed through the array. RAID 5 requires at least three RAID partitions.

- **RAID 6** – level 6 RAIDs are similar to level 5 RAIDs, but instead of storing only one set of parity data, they store two sets. RAID 6 requires at least four RAID partitions.

- **RAID 10** – level 10 RAIDs are *nested RAIDs* or *hybrid RAIDs*. Level 10 RAIDs are constructed by distributing data over mirrored sets of storage devices. For example, a level 10 RAID constructed from four RAID partitions consists of two pairs of partitions in which one partition mirrors the other. Data is then distributed across both pairs of storage devices, as in a level 0 RAID. RAID 10 requires at least four RAID partitions.

9.15.4. Create LVM Logical Volume



IMPORTANT

LVM initial set up is not available during text-mode installation. If you need to create an LVM configuration from scratch, press **Alt**+**F2** to use a different virtual console, and run the **lvm** command. To return to the text-mode installation, press **Alt**+**F1**.

Logical Volume Management (LVM) presents a simple logical view of underlying physical storage space, such as a hard drives or LUNs. Partitions on physical storage are represented as *physical volumes* that can be grouped together into *volume groups*. Each volume group can be divided into multiple *logical volumes*, each of which is analogous to a standard disk partition. Therefore, LVM logical volumes function as partitions that can span multiple physical disks.

To read more about LVM, refer to the Red Hat Enterprise Linux Deployment Guide . Note, LVM is only available in the graphical installation program.

LVM Physical Volume

Choose this option to configure a partition or device as an LVM physical volume. This option is the only choice available if your storage does not already contain LVM Volume Groups. This is the same dialog that appears when you add a standard partition – refer to Section 9.15.2, “Adding Partitions” for a description of the available options. Note, however, that **File System Type** must be set to **physical volume (LVM)**

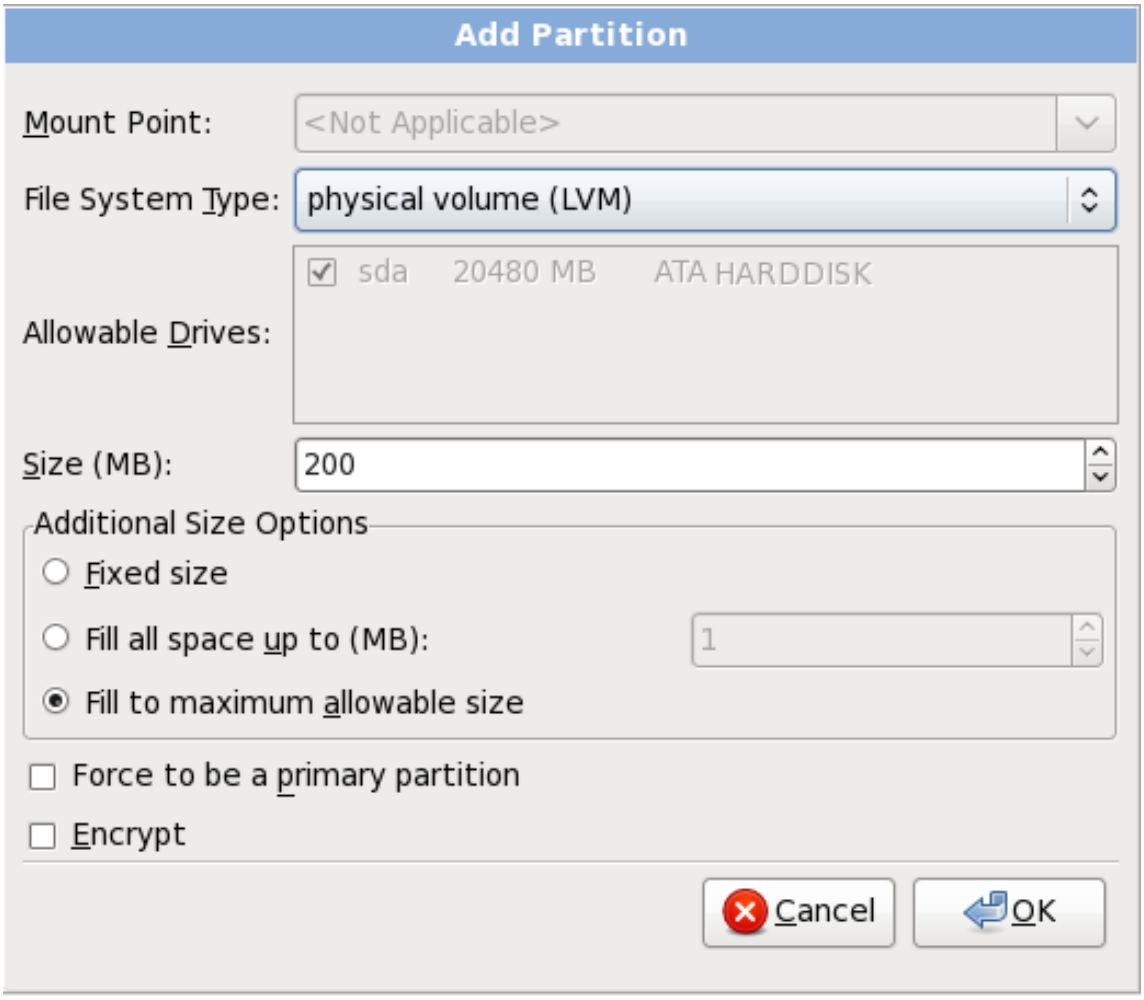


Figure 9.45. Create an LVM Physical Volume

Make LVM Volume Group

Choose this option to create LVM volume groups from the available LVM physical volumes, or to add existing logical volumes to a volume group.

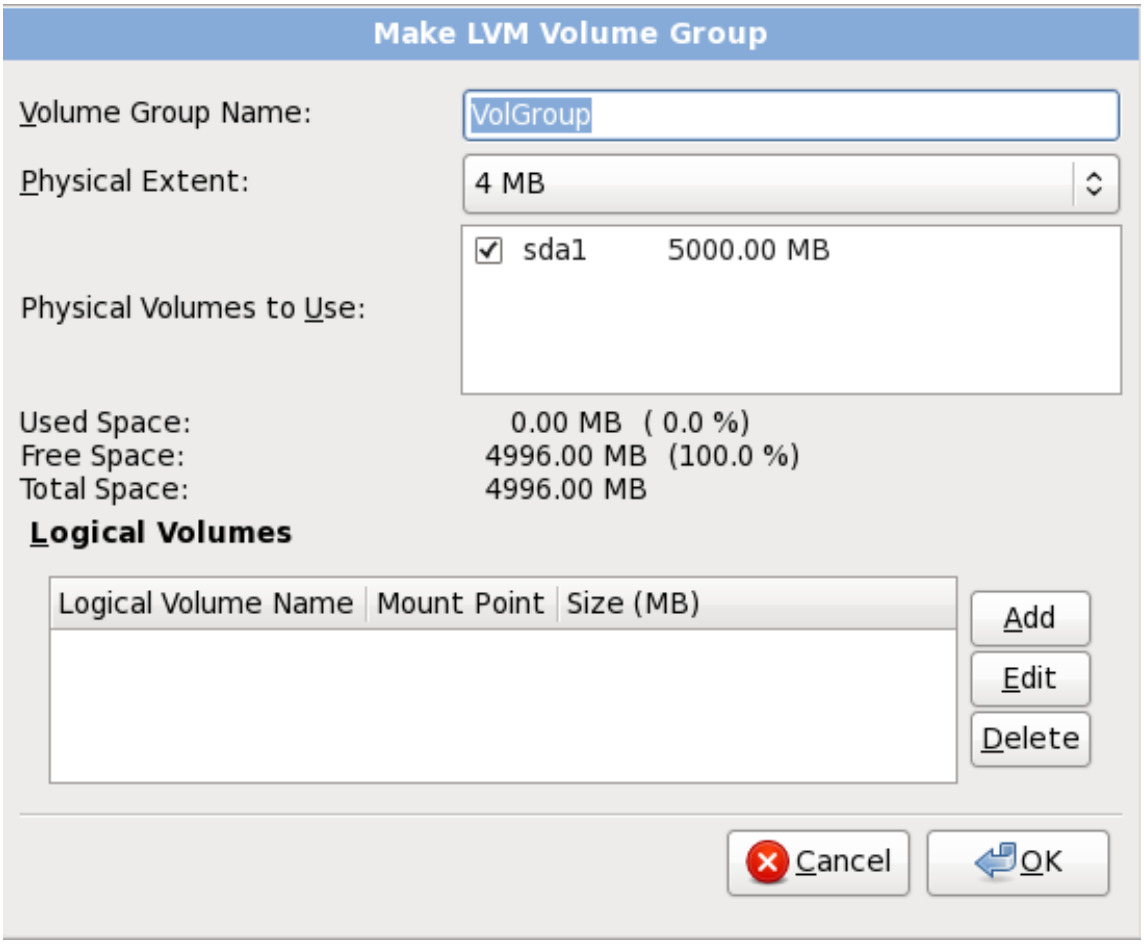


Figure 9.46. Make LVM Volume Group

To assign one or more physical volumes to a volume group, first name the volume group. Then select the physical volumes to be used in the volume group. Finally, configure logical volumes on any volume groups using the **Add**, **Edit** and **Delete** options.

You may not remove a physical volume from a volume group if doing so would leave insufficient space for that group's logical volumes. Take for example a volume group made up of two 5 GB LVM physical volume partitions, which contains an 8 GB logical volume. The installer would not allow you to remove either of the component physical volumes, since that would leave only 5 GB in the group for an 8 GB logical volume. If you reduce the total size of any logical volumes appropriately, you may then remove a physical volume from the volume group. In the example, reducing the size of the logical volume to 4 GB would allow you to remove one of the 5 GB physical volumes.

Make Logical Volume

Choose this option to create an LVM logical volume. Select a mount point, file system type, and size (in MB) just as if it were a standard disk partition. You can also choose a name for the logical volume and specify the volume group to which it will belong.

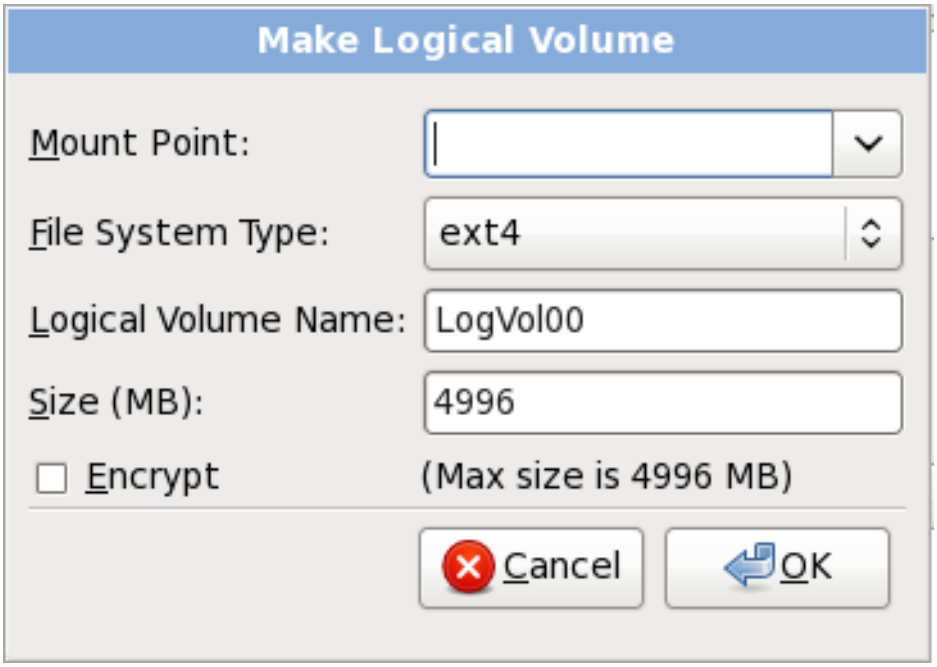


Figure 9.47. Make Logical Volume

9.15.5. Recommended Partitioning Scheme

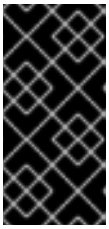
9.15.5.1. x86, AMD64, and Intel 64 systems

We recommend that you create the following partitions for x86, AMD64, and Intel 64 systems :

- A **swap** partition
- A /**boot** partition
- A / partition
- A **home** partition
- A /**boot/efi** partition (EFI System Partition) - only on systems with UEFI firmware
- A **swap** partition (at least 256 MB) – Swap partitions support virtual memory: data is written to a swap partition when there is not enough RAM to store the data your system is processing.

  In years past, the recommended amount of swap space increased linearly with the amount of RAM in the system. Modern systems often include hundreds of gigabytes of RAM, however. As a consequence, recommended swap space is considered a function of system memory workload, not system memory.

  The following table provides the recommended size of a swap partition depending on the amount of RAM in your system and whether you want sufficient memory for your system to hibernate. The recommended swap partition size is established automatically during installation. To allow for hibernation, however, you will need to edit the swap space in the custom partitioning stage.



IMPORTANT

Recommendations in the table below are especially important on systems with low memory (1 GB and less). Failure to allocate sufficient swap space on these systems may cause issues such as instability or even render the installed system unbootable.

Table 9.2. Recommended System Swap Space

Amount of RAM in the system	Recommended swap space	Recommended swap space if allowing for hibernation
⩽ 2GB	2 times the amount of RAM	3 times the amount of RAM
> 2GB – 8GB	Equal to the amount of RAM	2 times the amount of RAM
> 8GB – 64GB	At least 4 GB	1.5 times the amount of RAM
> 64GB	At least 4 GB	Hibernation not recommended

At the border between each range listed above (for example, a system with 2GB, 8GB, or 64GB 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 may lead to better performance.

Note that distributing swap space over multiple storage devices — particularly on systems with fast drives, controllers and interfaces — also improves swap space performance.

NOTE

Swap space size recommendations issued for Red Hat Enterprise Linux 6.0, 6.1, and 6.2 differed from the current recommendations, which were first issued with the release of Red Hat Enterprise Linux 6.3 in June 2012 and did not account for hibernation space. Automatic installations of these earlier versions of Red Hat Enterprise Linux 6 still generate a swap space in line with these superseded recommendations. However, manually selecting a swap space size in line with the newer recommendations issued for Red Hat Enterprise Linux 6.3 is advisable for optimal performance.

- A **/boot/** partition (250 MB)

  The partition mounted on **/boot/** contains the operating system kernel (which allows your system to boot Red Hat Enterprise Linux), along with files used during the bootstrap process. For most users, a 250 MB boot partition is sufficient.

IMPORTANT

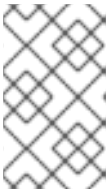
The **/boot** and / (root) partition in Red Hat Enterprise Linux 6.9 can only use the ext2, ext3, and ext4 (recommended) file systems. You cannot use any other file system for this partition, such as Btrfs, XFS, or VFAT. Other partitions, such as **/home**, can use any supported file system, including Btrfs and XFS (if available). See the following article on the Red Hat Customer Portal for additional information: https://access.redhat.com/solutions/667273.

WARNING

Note that normally the **/boot** partition is created automatically by the installer. However, if the / (root) partition is larger than 2 TB and (U)EFI is used for booting, you need to create a separate **/boot** partition that is smaller than 2 TB to boot the machine successfully.

NOTE

If your hard drive is more than 1024 cylinders (and your system was manufactured more than two years ago), you may need to create a **/boot/** partition if you want the / (root) partition to use all of the remaining space on your hard drive.

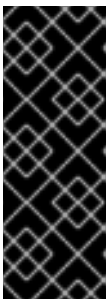


NOTE

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

- A **root** partition (3.0 GB - 5.0 GB) — this is where " /" (the root directory) is located. In this setup, all files (except those stored in **/boot**) are on the root partition.

  A 3.0 GB partition allows you to install a minimal installation, while a 5.0 GB root partition lets you perform a full installation, choosing all package groups.



IMPORTANT

The **/boot** and / (root) partition in Red Hat Enterprise Linux 6.9 can only use the ext2, ext3, and ext4 (recommended) file systems. You cannot use any other file system for this partition, such as Btrfs, XFS, or VFAT. Other partitions, such as **/home**, can use any supported file system, including Btrfs and XFS (if available). See the following article on the Red Hat Customer Portal for additional information: https://access.redhat.com/solutions/667273.

IMPORTANT

The / (or root) partition is the top of the directory structure. The **/root** directory (sometimes pronounced "slash-root") is the home directory of the user account for system administration.

- **A home partition (at least 100 MB)**

  To store user data separately from system data, create a dedicated partition within a volume group for the **/home** directory. This will enable you to upgrade or reinstall Red Hat Enterprise Linux without erasing user data files.

Many systems have more partitions than the minimum listed above. Choose partitions based on your particular system needs. Refer to Section 9.15.5.1.1, “Advice on Partitions” for more information.

If you create many partitions instead of one large / partition, upgrades become easier. Refer to the description of the Edit option in Section 9.15, “ Creating a Custom Layout or Modifying the Default Layout ” for more information.

The following table summarizes minimum partition sizes for the partitions containing the listed directories. You *do not* have to make a separate partition for each of these directories. For instance, if the partition containing **/foo** must be at least 500 MB, and you do not make a separate **/foo** partition, then the / (root) partition must be at least 500 MB.

Table 9.3. Minimum partition sizes

Directory	Minimum size
/	250 MB
/usr	250 MB
/tmp	50 MB
/var	384 MB
/home	100 MB
/boot	250 MB

NOTE

Leave Excess Capacity Unallocated, and only assign storage capacity to those partitions you require immediately. You may allocate free space at any time, to meet needs as they occur. To learn about a more flexible method for storage management, refer to Appendix D, *Understanding LVM*.

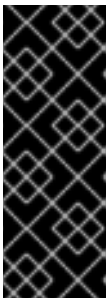
If you are not sure how best to configure the partitions for your computer, accept the default partition layout.

9.15.5.1.1. Advice on Partitions

Optimal partition setup depends on the usage for the Linux system in question. The following tips may help you decide how to allocate your disk space.

- Consider encrypting any partitions that 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.

- Each kernel installed on your system requires approximately 30 MB on the **/boot** partition. Unless you plan to install a great many kernels, the default partition size of 250 MB for **/boot** should suffice.



IMPORTANT

The **/boot** and / (root) partition in Red Hat Enterprise Linux 6.9 can only use the ext2, ext3, and ext4 (recommended) file systems. You cannot use any other file system for this partition, such as Btrfs, XFS, or VFAT. Other partitions, such as **/home**, can use any supported file system, including Btrfs and XFS (if available). See the following article on the Red Hat Customer Portal for additional information: https://access.redhat.com/solutions/667273.

- The **/var** directory holds content for a number of applications, including the **Apache** web server. It also is used to store downloaded update packages on a temporary basis. Ensure that the partition containing the **/var** directory has enough space to download pending updates and hold your other content.

WARNING

The **PackageKit** update software downloads updated packages to **/var/cache/yum/** by default. If you partition the system manually, and create a separate **/var/** partition, be sure to create the partition large enough (3.0 GB or more) to download package updates.

- The **/usr** directory holds the majority of software content on a Red Hat Enterprise Linux system. For an installation of the default set of software, allocate at least 4 GB of space. If you are a software developer or plan to use your Red Hat Enterprise Linux system to learn software development skills, you may want to at least double this allocation.

- 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 partitions to reallocate storage.

- a If you separate subdirectories into partitions, you can retain content in those subdirectories if you decide to install a new version of Red Hat Enterprise Linux over your current system. For instance, if you intend to run a **MySQL** databasge in **/var/lib/mysql**, make a separate partition for that directory in case you need to reinstall later.

- UEFI systems should contain a 50-150MB **/boot/efi** partition with an EFI System Partition filesystem.

The following table is a possible partition setup for a system with a single, new 80 GB hard disk and 1 GB of RAM. Note that approximately 10 GB of the volume group is unallocated to allow for future growth.

NOTE

This setup is an example, and is not optimal for all use cases.

Example 9.1. Example partition setup

Table 9.4. Example partition setup

Partition	Size and type
/boot	250 MB **ext3** partition
swap	2 GB swap
LVM physical volume	Remaining space, as one LVM volume group

The physical volume is assigned to the default volume group and divided into the following logical volumes:

Table 9.5. Example partition setup: LVM physical volume

Partition	Size and type
/	13 GB ext4
/var	4 GB ext4
/home	50 GB ext4

9.16. WRITE CHANGES TO DISK

The installer prompts you to confirm the partitioning options that you selected. Click **Write changes to disk** to allow the installer to partition your hard drive and install Red Hat Enterprise Linux.

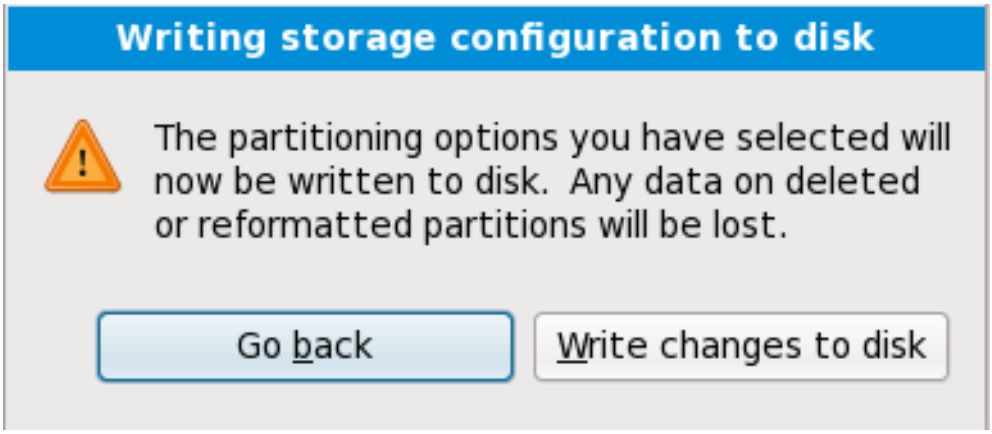


Figure 9.48. Writing storage configuration to disk

If you are certain that you want to proceed, click **Write changes to disk**.

WARNING

Up to this point in the installation process, the installer has made no lasting changes to your computer. When you click **Write changes to disk**, the installer will allocate space on your hard drive and start to transfer Red Hat Enterprise Linux into this space. Depending on the partitioning option that you chose, this process might include erasing data that already exists on your computer.

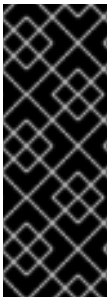
To revise any of the choices that you made up to this point, click **Go back**. To cancel installation completely, switch off your computer. To switch off most computers at this stage, press the power button and hold it down for a few seconds.

After you click **Write changes to disk**, allow the installation process to complete. If the process is interrupted (for example, by you switching off or resetting the computer, or by a power outage) you will probably not be able to use your computer until you restart and complete the Red Hat Enterprise Linux installation process, or install a different operating system.

9.17. PACKAGE GROUP SELECTION

Now that you have made most of the choices for your installation, you are ready to confirm the default package selection or customize packages for your system.

The **Package Installation Defaults** screen appears and details the default package set for your Red Hat Enterprise Linux installation. This screen varies depending on the version of Red Hat Enterprise Linux you are installing.



IMPORTANT

If you install Red Hat Enterprise Linux in text mode, you cannot make package selections. The installer automatically selects packages only from the base and core groups. These packages are sufficient to ensure that the system is operational at the end of the installation process, ready to install updates and new packages. To change the package selection, complete the installation, then use the **Add/Remove Software** application to make desired changes.

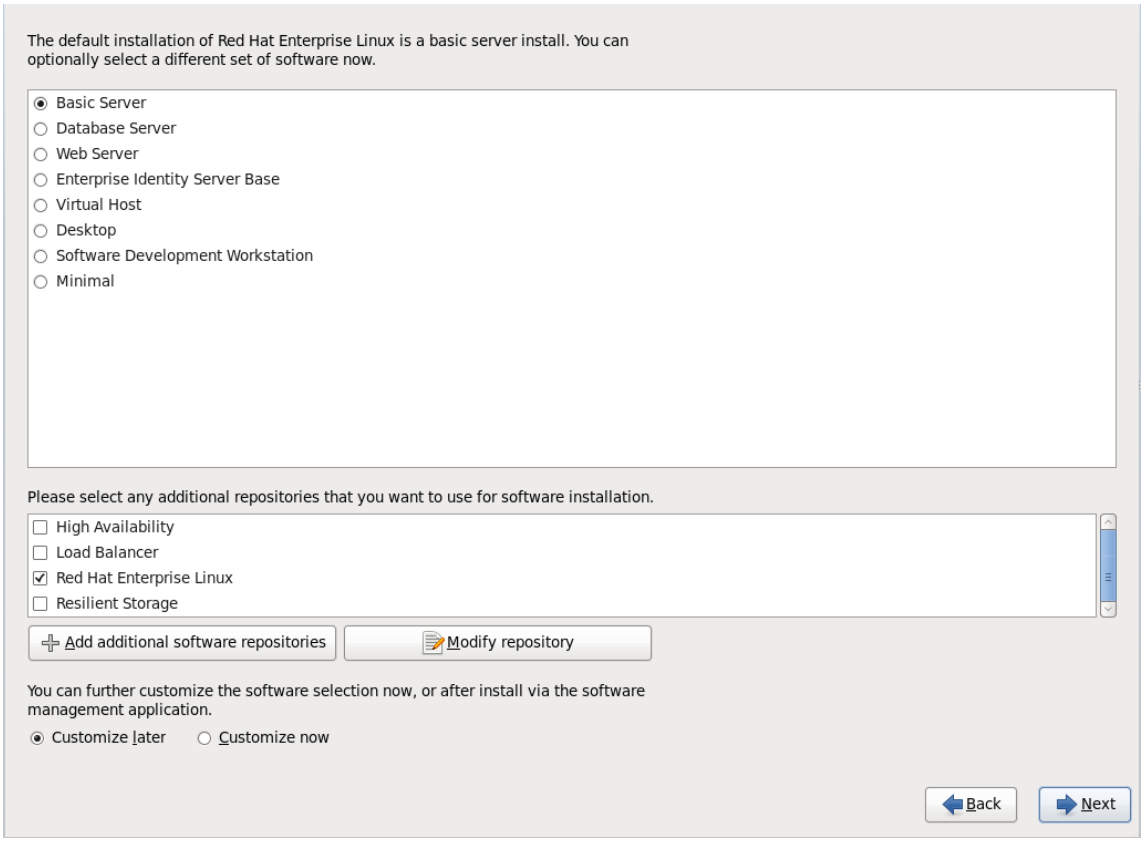


Figure 9.49. Package Group Selection

By default, the Red Hat Enterprise Linux installation process loads a selection of software that is suitable for a system deployed as a basic server. Note that this installation does not include a graphical environment. To include a selection of software suitable for other roles, click the radio button that corresponds to one of the following options:

Basic Server

This option provides a basic installation of Red Hat Enterprise Linux for use on a server.

Database Server

This option provides the **MySQL** and **PostgreSQL** databases.

Web server

This option provides the **Apache** web server.

Enterprise Identity Server Base

This option provides **OpenLDAP** and **Enterprise Identity Management** (IPA) to create an identity and authentication server.

Virtual Host

This option provides the **KVM** and **Virtual Machine Manager** tools to create a host for virtual machines.

Desktop

This option provides the **OpenOffice.org** productivity suite, graphical tools such as the **GIMP**, and multimedia applications.

Software Development Workstation

This option provides the necessary tools to compile software on your Red Hat Enterprise Linux system.

Minimal

This option provides only the packages essential to run Red Hat Enterprise Linux. A minimal installation provides the basis for a single-purpose server or desktop appliance and maximizes performance and security on such an installation.

WARNING

Minimal installation currently does not configure the firewall (**iptables**/**ip6tables**) by default because the authconfig and system-config-firewall-base packages are missing from the selection. To work around this issue, you can use a Kickstart file to add these packages to your selection. See the Red Hat Customer Portal for details about the workaround, and Chapter 32, *Kickstart Installations* for information about Kickstart files.

If you do not use the workaround, the installation will complete successfully, but no firewall will be configured, presenting a security risk.

If you choose to accept the current package list, skip ahead to Section 9.19, "Installing Packages".

To select a component, click on the checkbox beside it (refer to Figure 9.49, "Package Group Selection").

To customize your package set further, select the **Customize now** option on the screen. Clicking **Next** takes you to the **Package Group Selection** screen.

9.17.1. Installing from Additional Repositories

You can define additional *repositories* to increase the software available to your system during installation. A repository is a network location that stores software packages along with *metadata* that describes them. Many of the software packages used in Red Hat Enterprise Linux require other software to be installed. The installer uses the metadata to ensure that these requirements are met for every piece of software you select for installation.

The basic options are:

- The **High Availability** repository includes packages for high-availability clustering (also known as *failover clustering*) using the Red Hat High-availability Service Management component.
- The **Load Balancer** repository includes packages for load-balancing clustering using *Linux Virtual Server* (LVS).
- The **Red Hat Enterprise Linux** repository is automatically selected for you. It contains the complete collection of software that was released as Red Hat Enterprise Linux 6.9, with the various pieces of software in their versions that were current at the time of release.
- The **Resilient Storage** repository includes packages for storage clustering using the Red Hat *global file system* (GFS).

For more information about clustering with Red Hat Enterprise Linux 6.9, refer to the *Red Hat Enterprise Linux 6.9 High Availability Add-On Overview*, available from https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/High_Availability_Add-On_Overview/index.html.

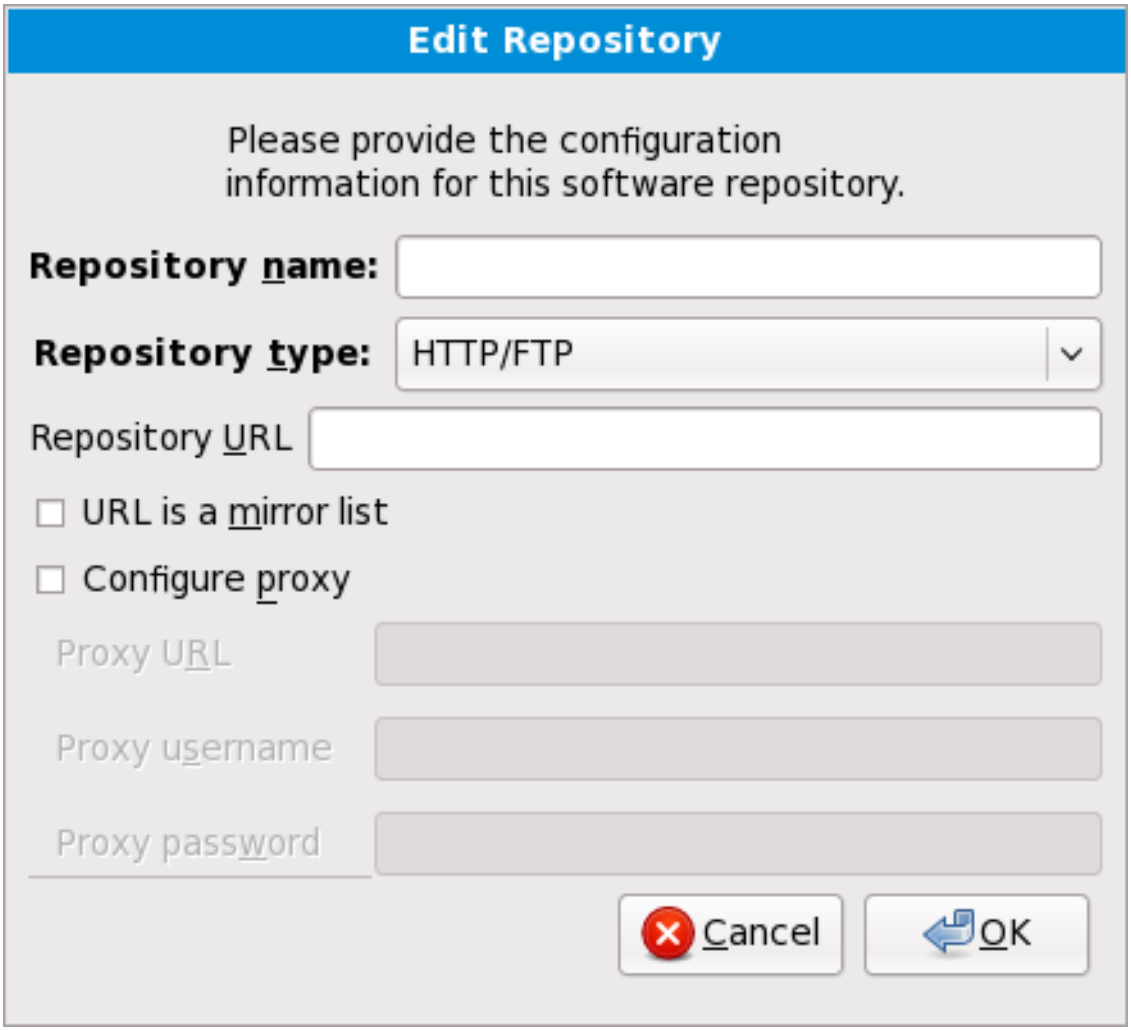


Figure 9.50. Adding a software repository

To include software from extra *repositories*, select **Add additional software repositories** and provide the location of the repository.

To edit an existing software repository location, select the repository in the list and then select **Modify repository**.

If you change the repository information during a non-network installation, such as from a Red Hat Enterprise Linux DVD, the installer prompts you for network configuration information.

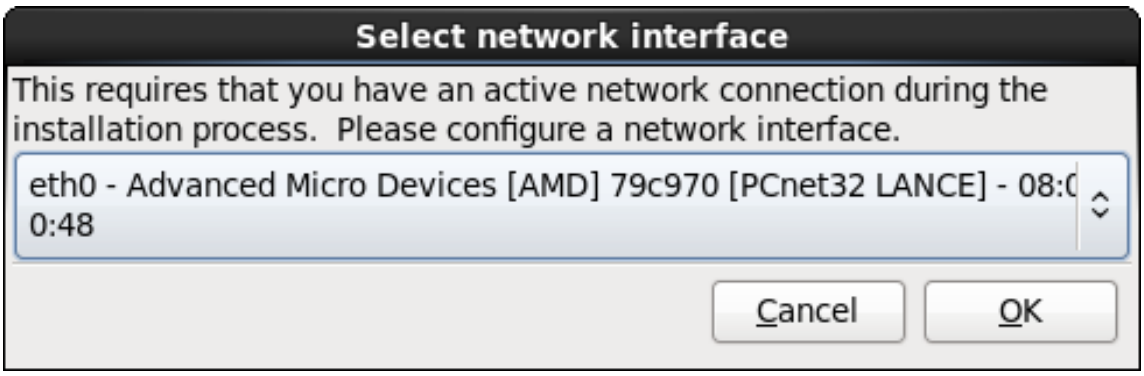


Figure 9.51. Select network interface

1. Select an interface from the drop-down menu.
2. Click **OK**.

Anaconda then starts **NetworkManager** to allow you to configure the interface.

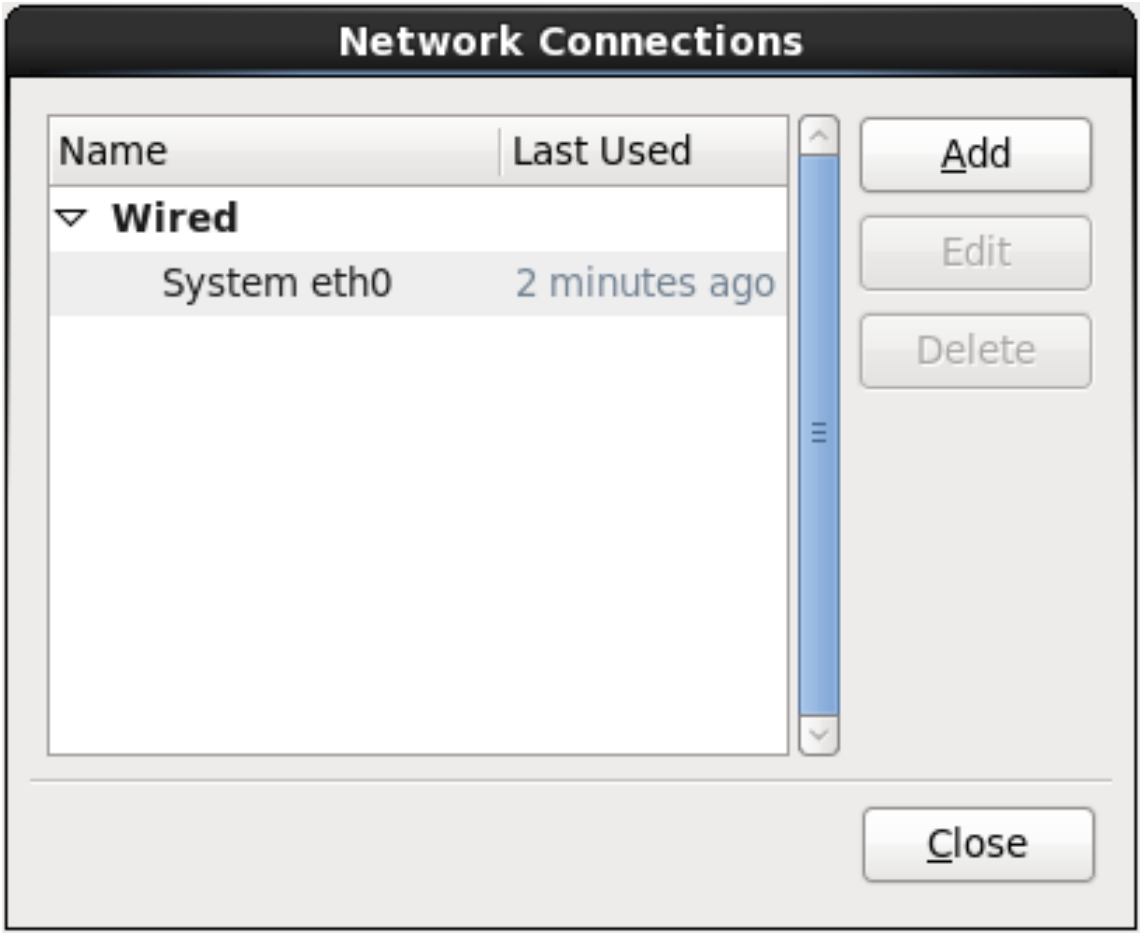


Figure 9.52. Network Connections

For details of how to use **NetworkManager**, refer to Section 9.7, "Setting the Hostname"

If you select **Add additional software repositories**, the **Edit repository** dialog appears. Provide a **Repository name** and the **Repository URL** for its location.

Once you have located a mirror, to determine the URL to use, find the directory on the mirror that *contains* a directory named **repodata**.

Once you provide information for an additional repository, the installer reads the package metadata over the network. Software that is specially marked is then included in the package group selection system.

WARNING

If you choose **Back** from the package selection screen, any extra repository data you may have entered is lost. This allows you to effectively cancel extra repositories. Currently there is no way to cancel only a single repository once entered.

9.17.2. Customizing the Software Selection

NOTE

Your Red Hat Enterprise Linux system automatically supports the language that you selected at the start of the installation process. To include support for additional languages, select the package group for those languages from the **Languages** category.

Select **Customize now** to specify the software packages for your final system in more detail. This option causes the installation process to display an additional customization screen when you select **Next**.

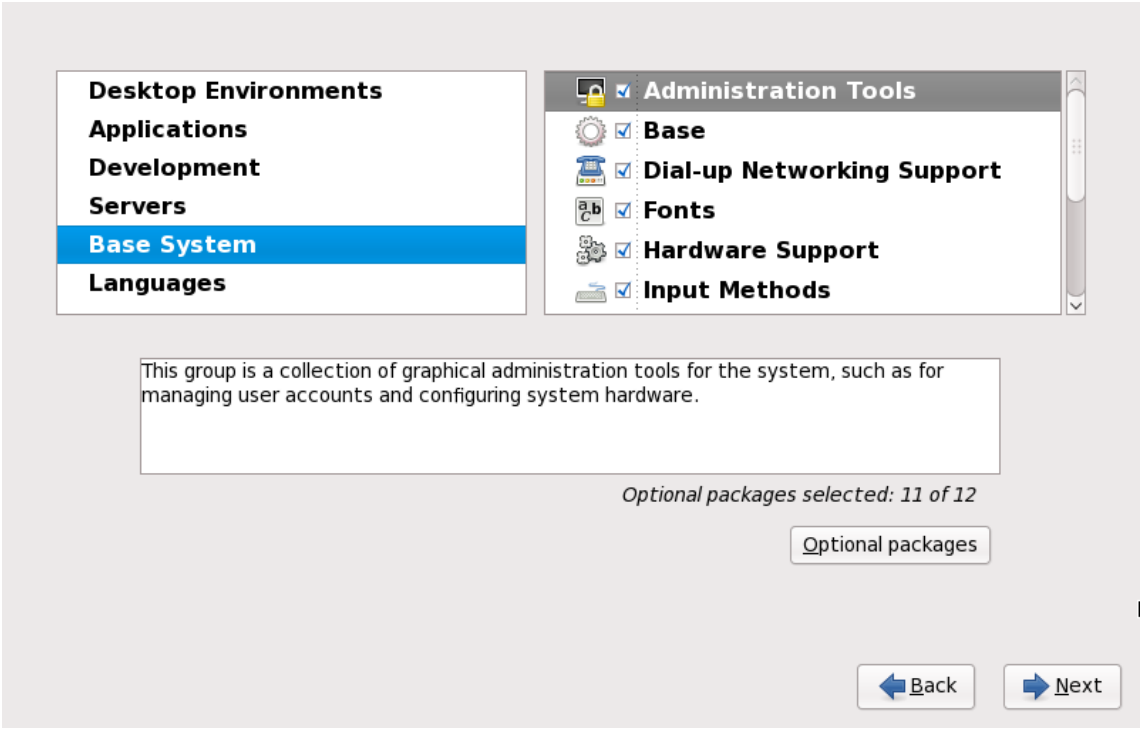


Figure 9.53. Package Group Details

Red Hat Enterprise Linux divides the included software into *package groups*. For ease of use, the package selection screen displays these groups as categories.

You can select package groups, which group components together according to function (for example, **X Window System** and **Editors**), individual packages, or a combination of the two.

To view the package groups for a category, select the category from the list on the left. The list on the right displays the package groups for the currently selected category.

To specify a package group for installation, select the check box next to the group. The box at the bottom of the screen displays the details of the package group that is currently highlighted. *None* of the packages from a group will be installed unless the check box for that group is selected.

If you select a package group, Red Hat Enterprise Linux automatically installs the base and mandatory packages for that group. To change which optional packages within a selected group will be installed, select the **Optional Packages** button under the description of the group. Then use the check box next to an individual package name to change its selection.

In the package selection list on the right, you can use the context menu as a shortcut to select or de-select base and mandatory packages or all optional packages.

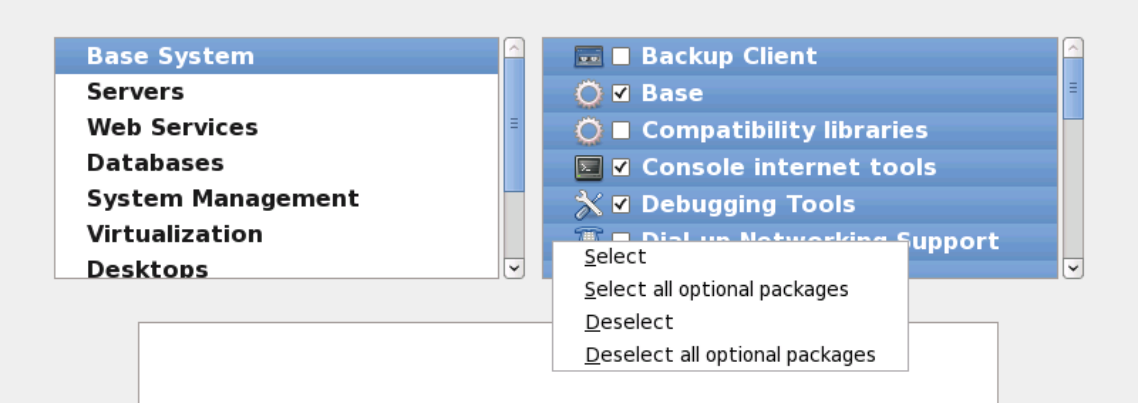


Figure 9.54. Package Selection List Context Menu

After you choose the desired packages, select **Next** to proceed. The installer checks your selection, and automatically adds any extra packages required to use the software you selected. When you have finished selecting packages, click **Close** to save your optional package selections and return to the main package selection screen.

The packages that you select are not permanent. After you boot your system, use the **Add/Remove Software** tool to either install new software or remove installed packages. To run this tool, from the main menu, select **System → Administration → Add/Remove Software**. The Red Hat Enterprise Linux software management system downloads the latest packages from network servers, rather than using those on the installation discs.

9.17.2.1. Core Network Services

All Red Hat Enterprise Linux installations include the following network services:

- centralized logging through syslog
- email through SMTP (Simple Mail Transfer Protocol)
- network file sharing through NFS (Network File System)

- remote access through SSH (Secure SHell)
- resource advertising through mDNS (multicast DNS)

The default installation also provides:

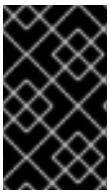
- network file transfer through HTTP (HyperText Transfer Protocol)
- printing through CUPS (Common UNIX Printing System)
- remote desktop access through VNC (Virtual Network Computing)

Some automated processes on your Red Hat Enterprise Linux system use the email service to send reports and messages to the system administrator. By default, the email, logging, and printing services do not accept connections from other systems. Red Hat Enterprise Linux installs the NFS sharing, HTTP, and VNC components without enabling those services.

You may configure your Red Hat Enterprise Linux system after installation to offer email, file sharing, logging, printing and remote desktop access services. The SSH service is enabled by default. You may use NFS to access files on other systems without enabling the NFS sharing service.

9.18. X86, AMD64, AND INTEL 64 BOOT LOADER CONFIGURATION

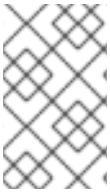
To boot the system without boot media, you usually need to install a boot loader. A boot loader is the first software program that runs when a computer starts. It is responsible for loading and transferring control to the operating system kernel software. The kernel, in turn, initializes the rest of the operating system.



IMPORTANT

If you install Red Hat Enterprise Linux in text mode, the installer configures the bootloader automatically and you cannot customize bootloader settings during the installation process.

GRUB (GRand Unified Bootloader), which is installed by default, is a very powerful boot loader. GRUB can load a variety of free operating systems, as well as proprietary operating systems with chain-loading (the mechanism for loading unsupported operating systems, such as Windows, by loading another boot loader). Note that the version of GRUB in Red Hat Enterprise Linux 6 is an old and stable version now known as "GRUB Legacy" since upstream development moved to GRUB 2.[4] Red Hat remains committed to maintaining the version of GRUB that we ship with Red Hat Enterprise Linux 6, just as we do with all packages that we ship.



NOTE

The GRUB menu defaults to being hidden, except on dual-boot systems. To show the GRUB menu during system boot, press and hold the **Shift** key before the kernel is loaded. (Any other key works as well but the **Shift** key is the safest to use.)

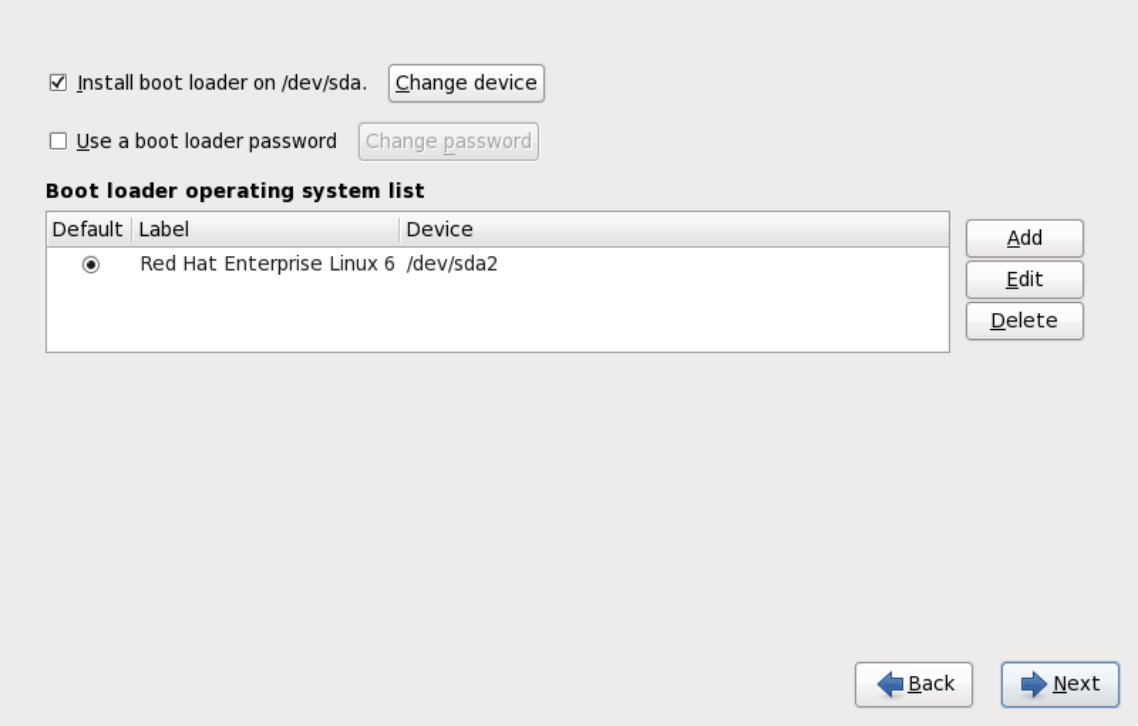


Figure 9.55. Boot Loader Configuration

If there are no other operating systems on your computer, or you are completely removing any other operating systems the installation program will install **GRUB** as your boot loader without any intervention. In that case you may continue on to Section 9.17, “Package Group Selection”.

You may have a boot loader installed on your system already. An operating system may install its own preferred boot loader, or you may have installed a third-party boot loader.If your boot loader does not recognize Linux partitions, you may not be able to boot Red Hat Enterprise Linux. Use **GRUB** as your boot loader to boot Linux and most other operating systems. Follow the directions in this chapter to install **GRUB**.

WARNING

If you install GRUB, it may overwrite your existing boot loader.

By default, the installation program installs GRUB in the master boot record or MBR, of the device for the root file system. To decline installation of a new boot loader, unselect **Install boot loader on /dev/sda**.

WARNING

If you choose not to install GRUB for any reason, you will not be able to boot the system directly, and you must use another boot method (such as a commercial boot loader application). Use this option only if you are sure you have another way of booting the system!

If you have other operating systems already installed, Red Hat Enterprise Linux attempts to automatically detect and configure **GRUB** to boot them. You may manually configure any additional operating systems if **GRUB** does not detect them.

To add, remove, or change the detected operating system settings, use the options provided.

Add

Select **Add** to include an additional operating system in GRUB.

Select the disk partition which contains the bootable operating system from the drop-down list and give the entry a label. **GRUB** displays this label in its boot menu.

Edit

To change an entry in the GRUB boot menu, select the entry and then select **Edit**.

Delete

To remove an entry from the GRUB boot menu, select the entry and then select **Delete**.

Select **Default** beside the preferred boot partition to choose your default bootable OS. You cannot move forward in the installation unless you choose a default boot image.

NOTE

The **Label** column lists what you must enter at the boot prompt, in non-graphical boot loaders, in order to boot the desired operating system.

Once you have loaded the GRUB boot screen, use the arrow keys to choose a boot label or type **e** for edit. You are presented with a list of items in the configuration file for the boot label you have selected.

Boot loader passwords provide a security mechanism in an environment where physical access to your server is available.

If you are installing a boot loader, you should create a password to protect your system. Without a boot loader password, users with access to your system can pass options to the kernel which can compromise your system security. With a boot loader password in place, the password must first be entered before selecting any non-standard boot options. However, it is still possible for someone with physical access to the machine to boot from a diskette, CD-ROM, DVD, or USB media if the BIOS supports it. Security plans which include boot loader passwords should also address alternate boot methods.

NOTE

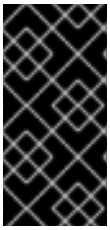
You may not require a **GRUB** password if your system only has trusted operators, or is physically secured with controlled console access. However, if an untrusted person can get physical access to your computer's keyboard and monitor, that person can reboot the system and access **GRUB**. A password is helpful in this case.

If you choose to use a boot loader password to enhance your system security, be sure to select the checkbox labeled **Use a boot loader password**.

Once selected, enter a password and confirm it.

GRUB stores the password in encrypted form, so it *cannot* be read or recovered. If you forget the boot password, boot the system normally and then change the password entry in the **/boot/grub/grub.conf** file. If you cannot boot, you may be able to use the "rescue" mode on the first Red Hat Enterprise Linux installation disc to reset the GRUB password.

If you do need to change the **GRUB** password, use the **grub-md5-crypt** utility. For information on using this utility, use the command **man grub-md5-crypt** in a terminal window to read the manual pages.



IMPORTANT

When selecting a GRUB password, be aware that GRUB recognizes only the QWERTY keyboard layout, regardless of the keyboard actually attached to the system. If you use a keyboard with a significantly different layout, it might be more effective to memorize a pattern of keystrokes rather than the word that the pattern produces.

To configure more advanced boot loader options, such as changing the drive order or passing options to the kernel, be sure **Configure advanced boot loader options** is selected before clicking **Next**.

9.18.1. Advanced Boot Loader Configuration

Now that you have chosen which boot loader to install, you can also determine where you want the boot loader to be installed. You may install the boot loader in one of two places:

- The Master Boot Record (MBR) – This is the recommended place to install a boot loader on systems with BIOS firmware, unless the MBR already starts another operating system loader, such as System Commander. The MBR is a special area on your hard drive that is automatically loaded by your computer's BIOS, and is the earliest point at which the boot loader can take control of the boot process. If you install it in the MBR, when your machine boots, GRUB presents a boot prompt. You can then boot Red Hat Enterprise Linux or any other operating system that you have configured the boot loader to boot.

- The EFI System Partition – Systems with UEFI firmware require a special partition for installing the boot loader. This should be a physical (non-LVM) partition of the **efi** type at least 50 MB in size; the recommended size is 200 MB. The drive containing this partition must be labeled with a GUID Partition Table (GPT) instead of a Master Boot Record. If you are installing Red Hat Enterprise Linux on a drive with a MBR, the drive must be relabeled; all data on the drive will be lost in the process.

- The first sector of your boot partition – This is recommended if you are already using another boot loader on your system. In this case, your other boot loader takes control first. You can then configure that boot loader to start GRUB, which then boots Red Hat Enterprise Linux.

NOTE

If you install GRUB as a secondary boot loader, you must reconfigure your primary boot loader whenever you install and boot from a new kernel. The kernel of an operating system such as Microsoft Windows does not boot in the same fashion. Most users therefore use GRUB as the primary boot loader on dual-boot systems.

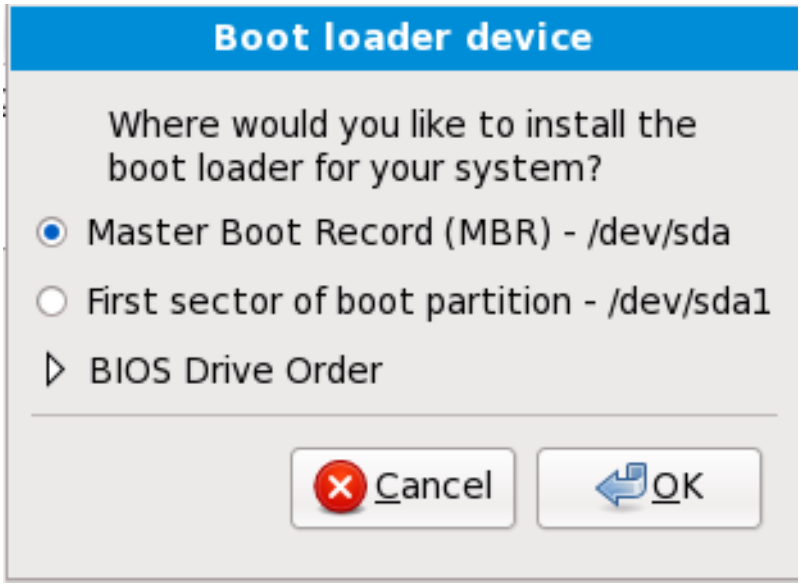
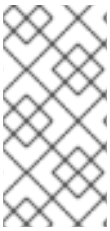


Figure 9.56. Boot Loader Installation

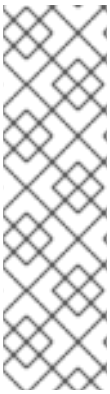


NOTE

If you have a RAID card, be aware that some BIOS types do not support booting from the RAID card. In cases such as these, the boot loader *should not* be installed on the MBR of the RAID array. Rather, the boot loader should be installed on the MBR of the same drive as the **/boot/** partition was created.

If your system only uses Red Hat Enterprise Linux, you should choose the MBR.

Click the **Change Drive Order** button if you would like to rearrange the drive order or if your BIOS does not return the correct drive order. Changing the drive order may be useful if you have multiple SCSI adapters, or both SCSI and IDE adapters, and you want to boot from the SCSI device.



NOTE

While partitioning your hard drive, keep in mind that the BIOS in some older systems cannot access more than the first 1024 cylinders on a hard drive. If this is the case, leave enough room for the **/boot** Linux partition on the first 1024 cylinders of your hard drive to boot Linux. The other Linux partitions can be after cylinder 1024.

In **parted**, 1024 cylinders equals 528MB. For more information, refer to:

> http://www.pcguide.com/ref/hdd/bios/sizeMB504-c.html

9.18.2. Rescue Mode

Rescue mode provides the ability to boot a small Red Hat Enterprise Linux environment entirely from boot media or some other boot method instead of the system's hard drive. There may be times when you are unable to get Red Hat Enterprise Linux running completely enough to access files on your system's hard drive. Using rescue mode, you can access the files stored on your system's hard drive, even if you cannot actually run Red Hat Enterprise Linux from that hard drive. If you need to use rescue mode, try the following method:

- Boot an x86, AMD64, or Intel 64 system from any installation medium, such as CD, DVD, USB, or PXE, and type **linux rescue** at the installation boot prompt. Refer to Chapter 36, *Basic System Recovery* for a more complete description of rescue mode.

For additional information, refer to the Red Hat Enterprise Linux Deployment Guide .

9.18.3. Alternative Boot Loaders

GRUB is the default bootloader for Red Hat Enterprise Linux, but is not the only choice. A variety of open-source and proprietary alternatives to **GRUB** are available to load Red Hat Enterprise Linux, including **LILO**, **SYSLINUX**, and **Acronis Disk Director Suite**.

IMPORTANT

Red Hat does not provide customer support for third-party boot loaders.

9.19. INSTALLING PACKAGES

At this point there is nothing left for you to do until all the packages have been installed. How quickly this happens depends on the number of packages you have selected and your computer's speed.

Depending on the available resources, you might see the following progress bar while the installer resolves dependencies of the packages you selected for installation:

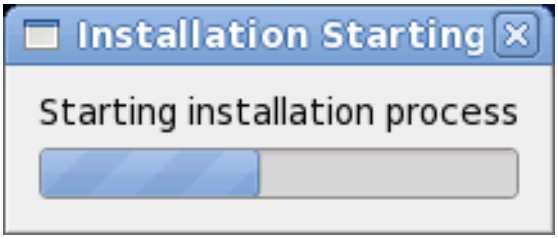


Figure 9.57. Starting installation

Red Hat Enterprise Linux reports the installation progress on the screen as it writes the selected packages to your system.

Figure 9.58. Packages completed

For your reference, a complete log of your installation can be found in **/root/install.log** once you reboot your system.

After installation completes, select **Reboot** to restart your computer. Red Hat Enterprise Linux ejects any loaded discs before the computer reboots.

9.20. INSTALLATION COMPLETE

Congratulations! Your Red Hat Enterprise Linux installation is now complete!

The installation program prompts you to prepare your system for reboot. Remember to remove any installation media if it is not ejected automatically upon reboot.

After your computer's normal power-up sequence has completed, Red Hat Enterprise Linux loads and starts. By default, the start process is hidden behind a graphical screen that displays a progress bar. Eventually, a **login:** prompt or a GUI login screen (if you installed the X Window System and chose to start X automatically) appears.

The first time you start your Red Hat Enterprise Linux system in run level 5 (the graphical run level), the **FirstBoot** tool appears, which guides you through the Red Hat Enterprise Linux configuration. Using this tool, you can set your system time and date, install software, register your machine with Red Hat Network, and more. **FirstBoot** lets you configure your environment at the beginning, so that you can get started using your Red Hat Enterprise Linux system quickly.

Chapter 34, *Firstboot* will guide you through the configuration process.

---

[2] A root password is the administrative password for your Red Hat Enterprise Linux system. You should only log in as root when needed for system maintenance. The root account does not operate within the restrictions placed on normal user accounts, so changes made as root can have implications for your entire system.

[3] The **fsck** application is used to check the file system for metadata consistency and optionally repair one or more Linux file systems.

[4] http://www.gnu.org/software/grub/grub-legacy.html

CHAPTER 10. TROUBLESHOOTING INSTALLATION ON AN INTEL OR AMD SYSTEM

This section discusses some common installation problems and their solutions.

For debugging purposes, **anaconda** logs installation actions into files in the **/tmp** directory. These files include:

/tmp/anaconda.log

general **anaconda** messages

/tmp/program.log

all external programs run by **anaconda**

/tmp/storage.log

extensive storage module information

/tmp/yum.log

yum package installation messages

/tmp/syslog

hardware-related system messages

If the installation fails, the messages from these files are consolidated into **/tmp/anaconda-tb-*identifier***, where *identifier* is a random string.

All of the files above reside in the installer's ramdisk and are thus volatile. To make a permanent copy, copy those files to another system on the network using **scp** on the installation image (not the other way round).

10.1. YOU ARE UNABLE TO BOOT RED HAT ENTERPRISE LINUX

10.1.1. Are You Unable to Boot With Your RAID Card?

If you have performed an installation and cannot boot your system properly, you may need to reinstall and create your partitions differently.

Some BIOS types do not support booting from RAID cards. At the end of an installation, a text-based screen showing the boot loader prompt (for example, **GRUB:**) and a flashing cursor may be all that appears. If this is the case, you must repartition your system.

Whether you choose automatic or manual partitioning, you must install your **/boot** partition outside of the RAID array, such as on a separate hard drive. An internal hard drive is necessary to use for partition creation with problematic RAID cards.

You must also install your preferred boot loader (GRUB or LILO) on the MBR of a drive that is outside of the RAID array. This should be the same drive that hosts the **/boot/** partition.

Once these changes have been made, you should be able to finish your installation and boot the system properly.

10.1.2. Is Your System Displaying Signal 11 Errors?

A signal 11 error, commonly known as a *segmentation fault*, means that the program accessed a memory location that was not assigned to it. A signal 11 error may be due to a bug in one of the software programs that is installed, or faulty hardware.

If you receive a fatal signal 11 error during your installation, it is probably due to a hardware error in memory on your system's bus. Like other operating systems, Red Hat Enterprise Linux places its own demands on your system's hardware. Some of this hardware may not be able to meet those demands, even if they work properly under another OS.

Ensure that you have the latest installation updates and images. Review the online errata to see if newer versions are available. If the latest images still fail, it may be due to a problem with your hardware. Commonly, these errors are in your memory or CPU-cache. A possible solution for this error is turning off the CPU-cache in the BIOS, if your system supports this. You could also try to swap your memory around in the motherboard slots to check if the problem is either slot or memory related.

Another option is to perform a media check on your installation DVD. **Anaconda**, the installation program, has the ability to test the integrity of the installation media. It works with the DVD, hard drive ISO, and NFS ISO installation methods. Red Hat recommends that you test all installation media before starting the installation process, and before reporting any installation-related bugs (many of the bugs reported are actually due to improperly-burned DVDs). To use this test, type the following command at the **boot:** or **yaboot:** prompt:

```
linux mediacheck
```

For more information concerning signal 11 errors, refer to:

```
http://www.bitwizard.nl/sig11/
```

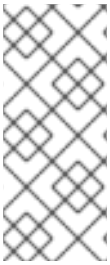
10.1.3. Diagnosing Early Boot Problems

The *boot console* may be useful in cases where your system fails to boot, but does successfully display the **GRUB** boot menu. Messages in the boot console can inform you of the current kernel version, command line parameters which have been passed to the kernel from the boot menu, enabled hardware support for the current kernel, physical memory map and other information which may help you find the cause of your problems.

To enable the boot console, select an entry in the **GRUB** boot menu, and press **e** to edit boot options. On the line starting with the keyword **kernel** (or **linux** in some cases), append the following:

- On a system with BIOS firmware, append **earlyprintk=vga,keep**. Boot console messages should then be displayed on the system display.
- On a system with UEFI firmware, append **earlyprintk=efi,keep**. Boot console messages should then be displayed in the EFI frame buffer.

You can also append the **quiet** option (if not present already) to suppress all other messages and only display messages from the boot console.



NOTE

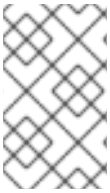
The earlyprintk options for BIOS and UEFI should also be enabled in the kernel's **/boot/config-*version*** file - the **CONFIG_EARLY_PRINTK=** and **CONFIG_EARLY_PRINTK_EFI=** options must be set to the **y** value. They are enabled by default, but if you disabled them, you may need to mount the **/boot** partition in rescue mode and edit the configuration file to re-enable them.

10.2. TROUBLE BEGINNING THE INSTALLATION

10.2.1. Problems with Booting into the Graphical Installation

There are some video cards that have trouble booting into the graphical installation program. If the installation program does not run using its default settings, it tries to run in a lower resolution mode. If that still fails, the installation program attempts to run in text mode.

One possible solution is to use only a basic video driver during installation. You can do this either by selecting **Install system with basic video driver** on the boot menu, or using the **xdriver=vesa** boot option at the boot prompt. Alternatively, you can force the installer to use a specific screen resolution with the **resolution=** boot option. This option may be most helpful for laptop users. Another solution to try is the **driver=** option to specify the driver that should be loaded for your video card. If this works, you should report it as a bug, because the installer failed to detect your video card automatically. Refer to Chapter 28, *Boot Options* for more information on boot options.



NOTE

To disable frame buffer support and allow the installation program to run in text mode, try using the **nofb** boot option. This command may be necessary for accessibility with some screen reading hardware.

10.3. TROUBLE DURING THE INSTALLATION

10.3.1. The "No devices found to install Red Hat Enterprise Linux" Error Message

If you receive an error message stating **No devices found to install Red Hat Enterprise Linux**, there is probably a SCSI controller that is not being recognized by the installation program.

Check your hardware vendor's website to determine if a driver update is available that fixes your problem. For more general information on driver updates, refer to Chapter 6, *Updating Drivers During Installation on Intel and AMD Systems*.

You can also refer to the *Red Hat Hardware Compatibility List* , available online at:

https://hardware.redhat.com/

10.3.2. Saving Traceback Messages

If **anaconda** encounters an error during the graphical installation process, it presents you with a crash reporting dialog box:

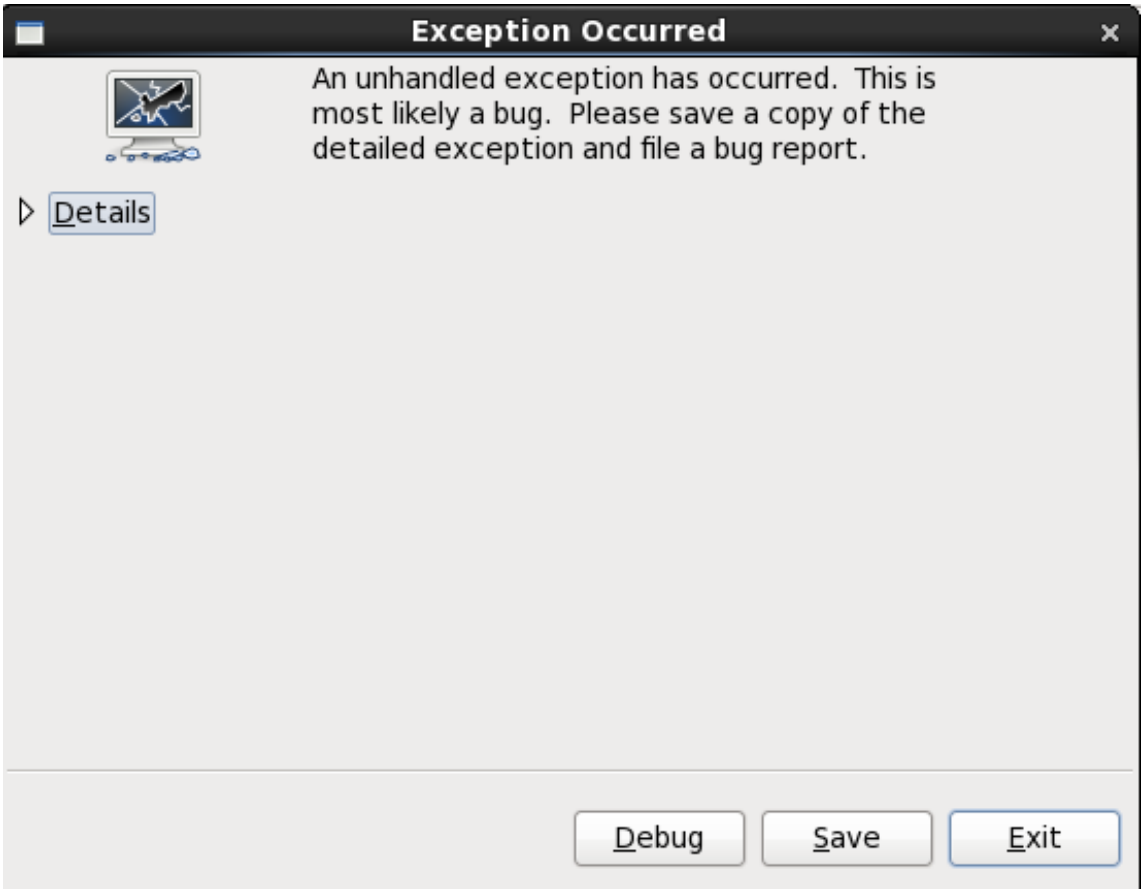


Figure 10.1. The Crash Reporting Dialog Box

Details

shows you the details of the error:

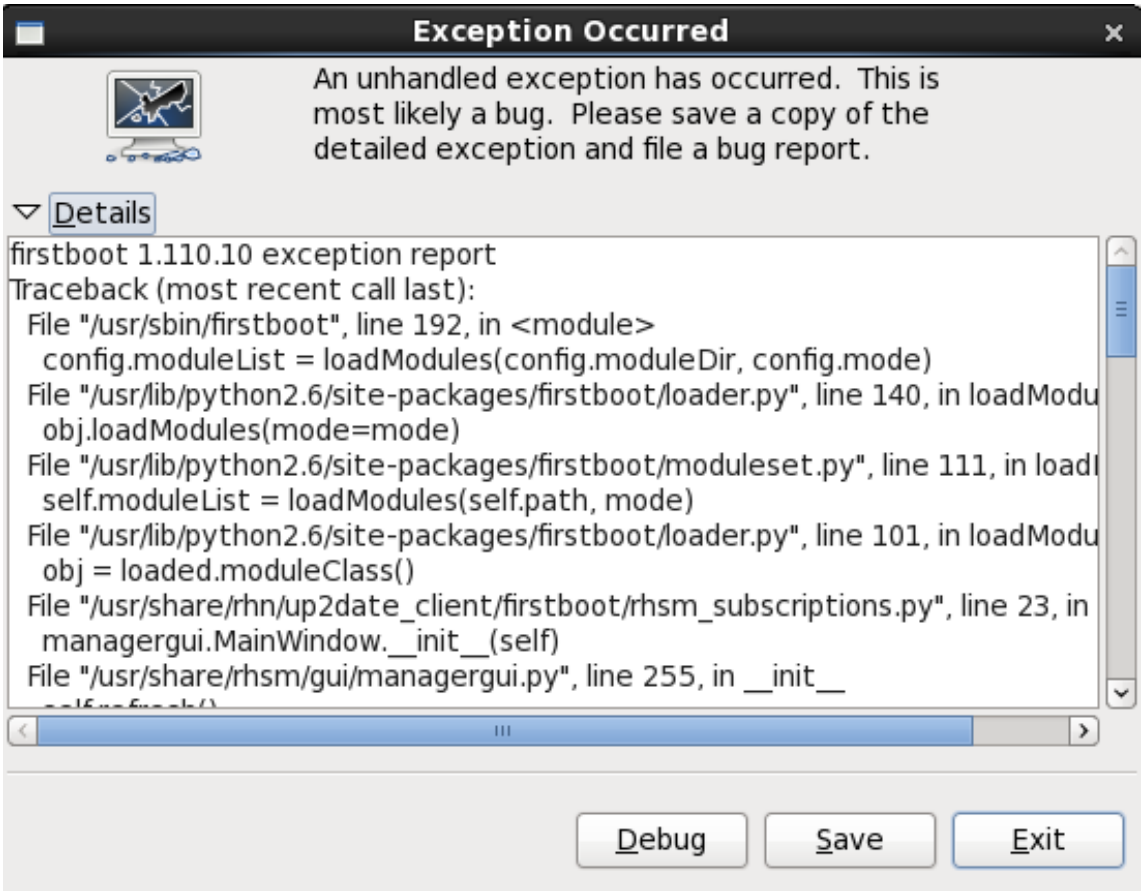


Figure 10.2. Details of the Crash

Save

saves details of the error locally or remotely:

Exit

exits the installation process.

If you select **Save** from the main dialog, you can choose from the following options:

Figure 10.3. Select reporter

Logger

saves details of the error as a log file to the local hard drive at a specified location.

Red Hat Customer Support

submits the crash report to Customer Support for assistance.

Report uploader

uploads a compressed version of the crash report to Bugzilla or a URL of your choice.

Before submitting the report, click **Preferences** to specify a destination or provide authentication details. Select the reporting method you need to configure and click **Configure Event**.

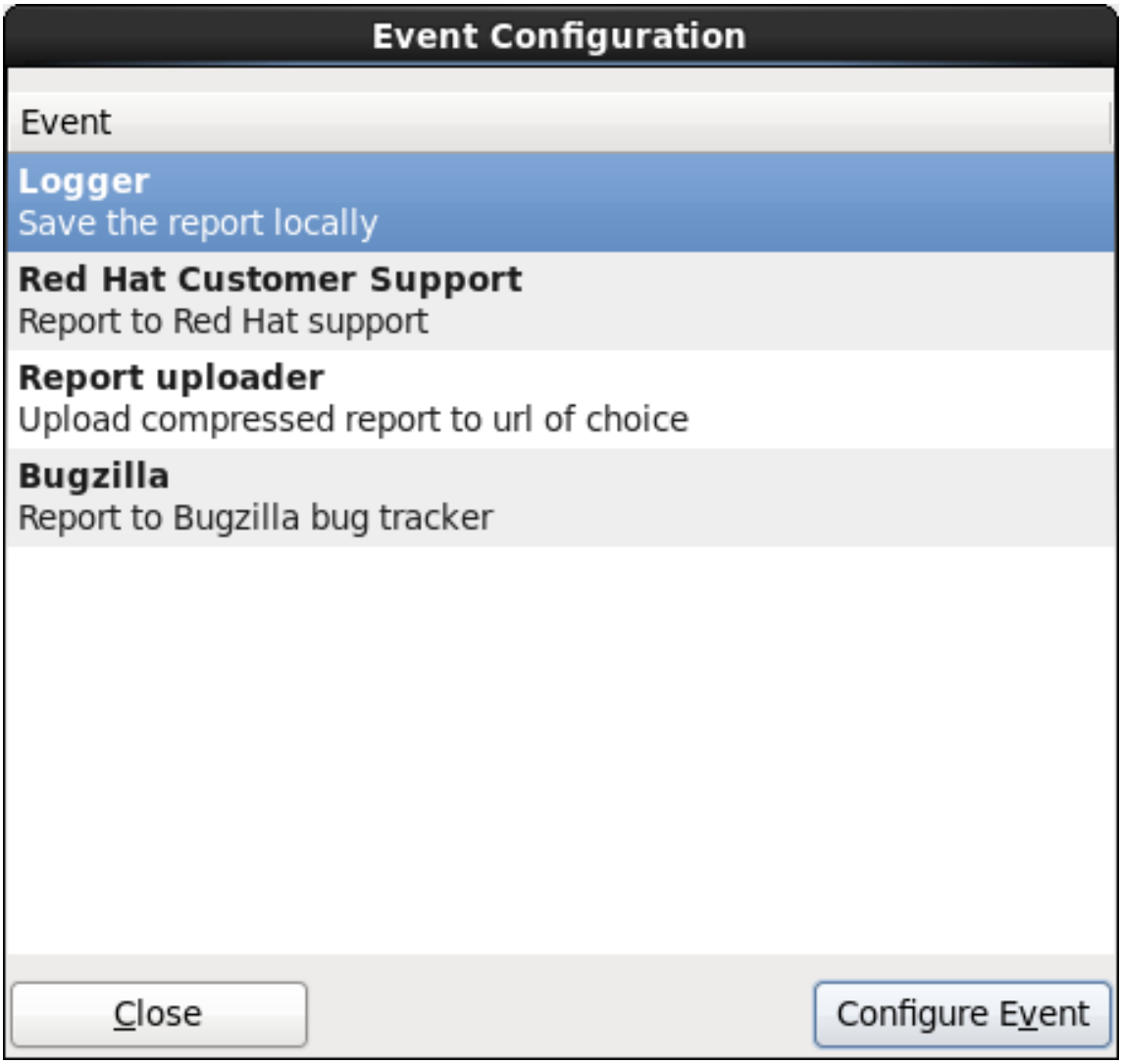


Figure 10.4. Configure reporter preferences

Logger

Specify a path and a filename for the log file. Check **Append** if you are adding to an existing log file.

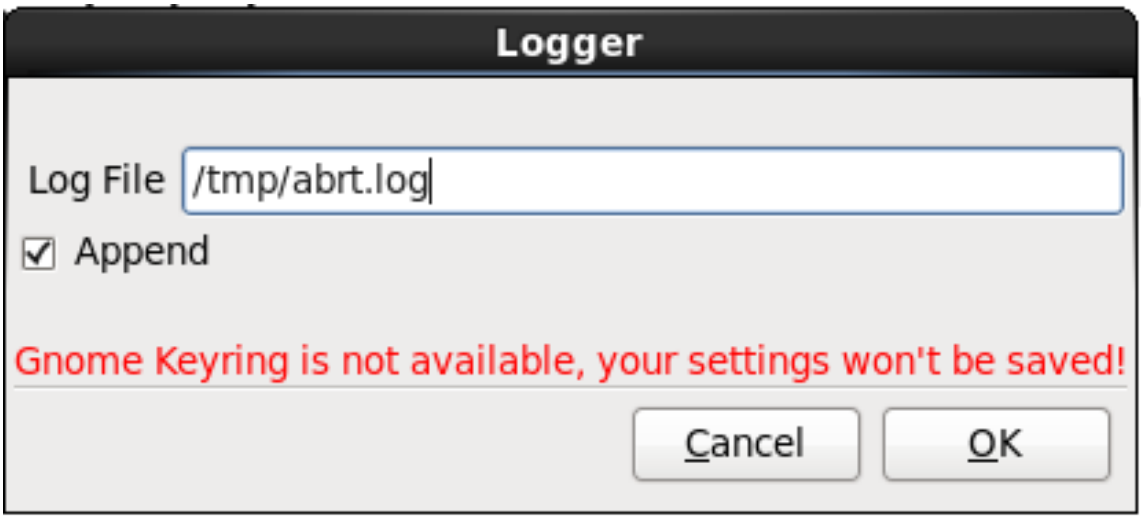


Figure 10.5. Specify local path for log file

Red Hat Customer Support

Enter your Red Hat Network username and password so your report reaches Customer Support and is linked with your account. The URL is prefilled and **Verify SSL** is checked by default.

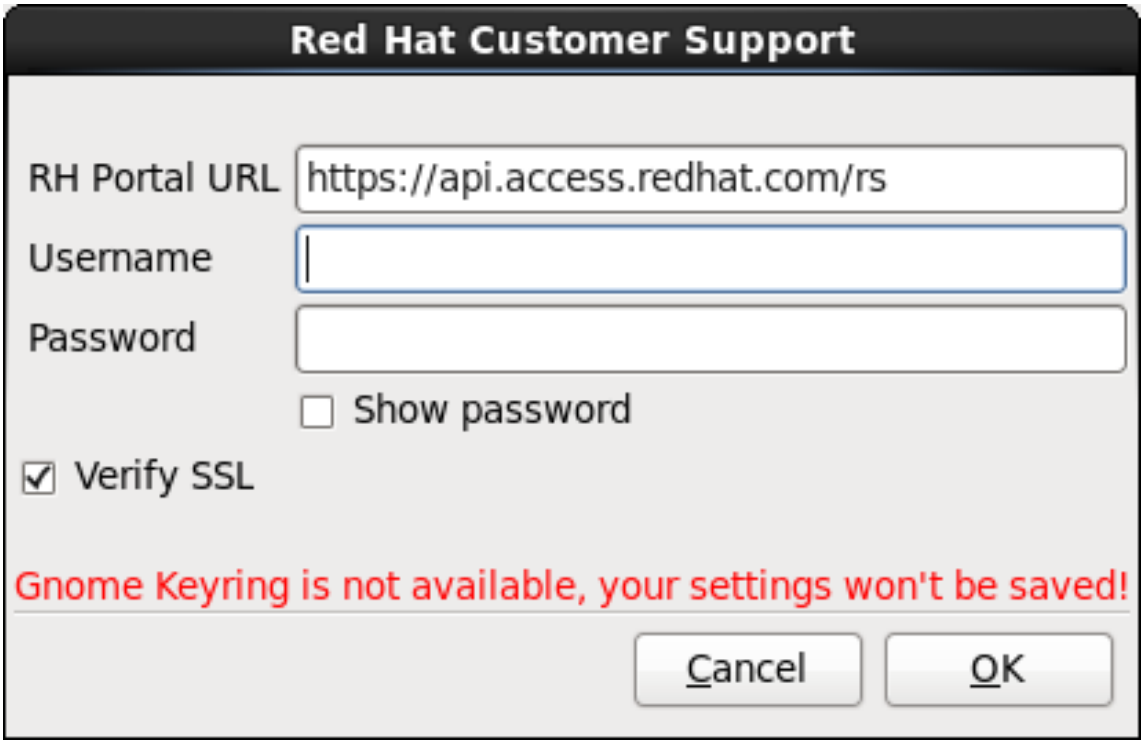


Figure 10.6. Enter Red Hat Network authentication details

Report uploader

Specify a URL for uploading a compressed version of the crash report.

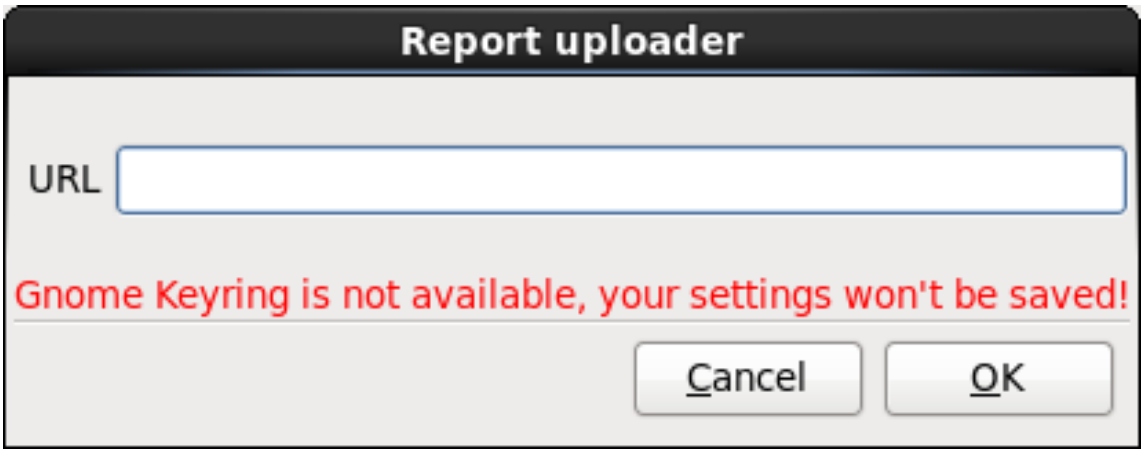


Figure 10.7. Enter URL for uploading crash report

Bugzilla

Enter your Bugzilla username and password to lodge a bug with Red Hat's bug-tracking system using the crash report. The URL is prefilled and **Verify SSL** is checked by default.

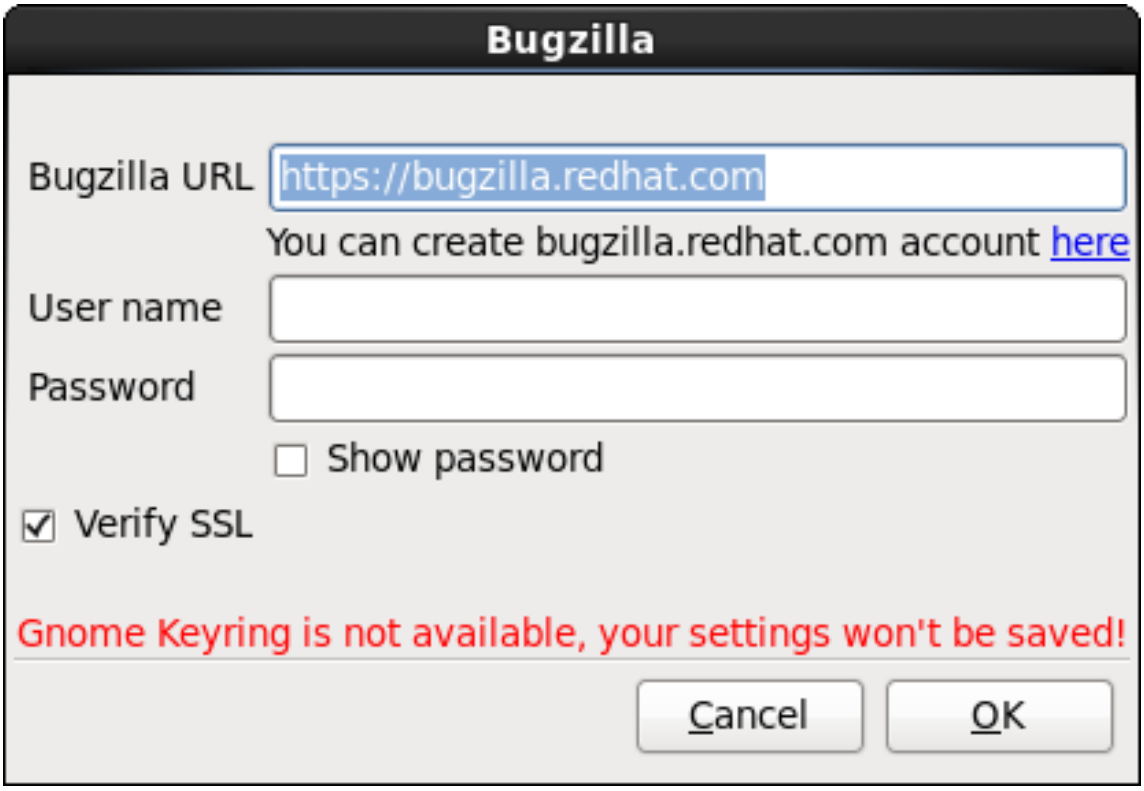


Figure 10.8. Enter Bugzilla authentication details

Once you have entered your preferences, click **OK** to return to the report selection dialog. Select how you would like to report the problem and then click **Forward**.

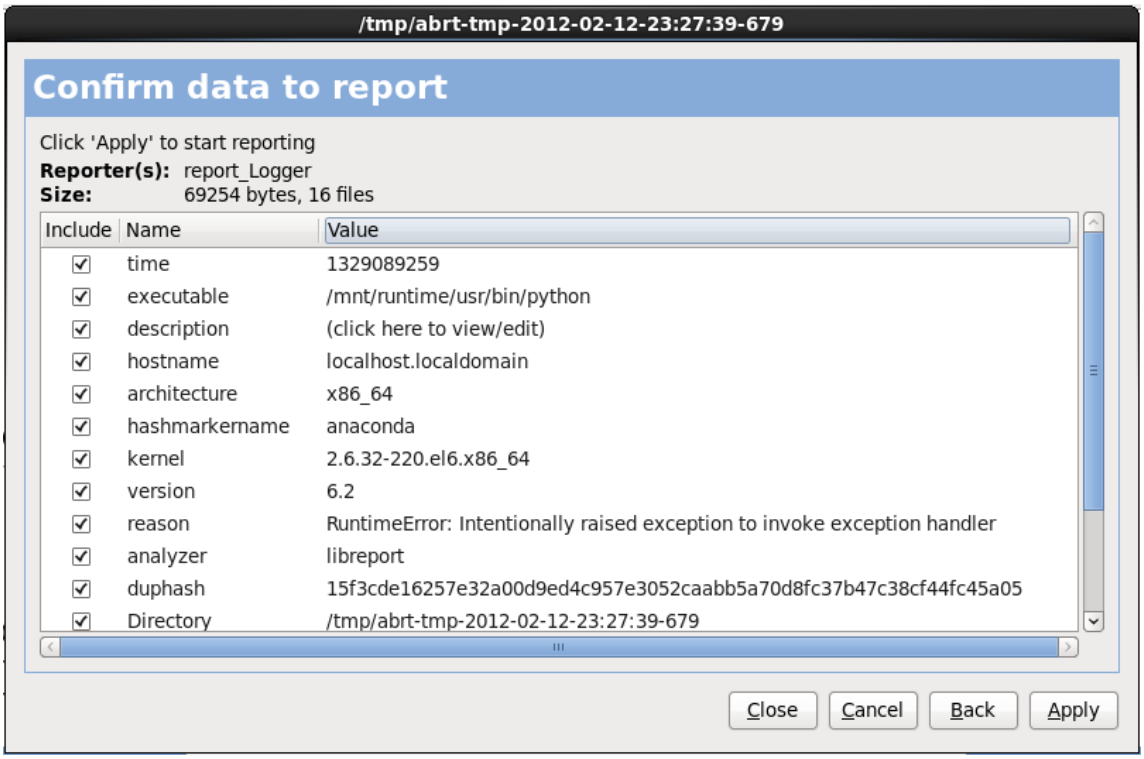


Figure 10.9. Confirm report data

You can now customize the report by checking and unchecking the issues that will be included. When finished, click **Apply**.

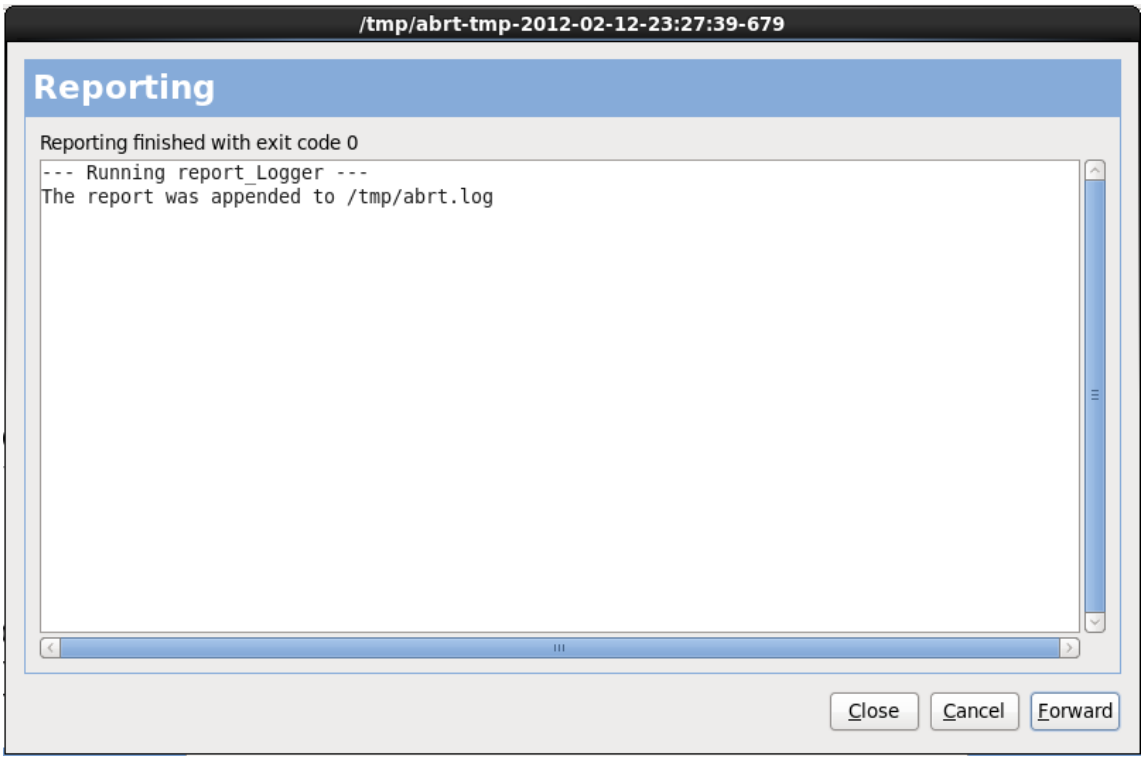


Figure 10.10. Report in progress

This screen displays the outcome of the report, including any errors in sending or saving the log. Click **Forward** to proceed.

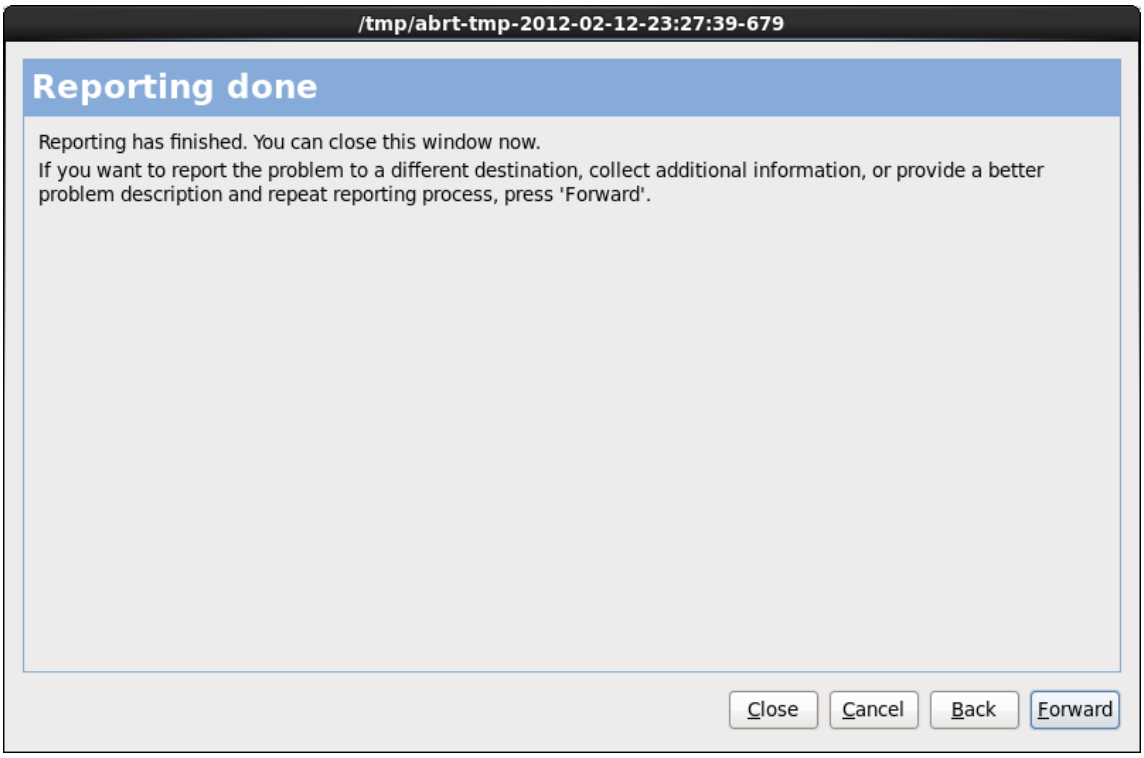


Figure 10.11. Reporting done

Reporting is now complete. Click **Forward** to return to the report selection dialog. You can now make another report, or click **Close** to exit the reporting utility and then **Exit** to close the installation process.

10.3.3. Trouble with Partition Tables

If you receive an error after the **Disk Partitioning Setup** (Section 9.13, "Disk Partitioning Setup") phase of the installation saying something similar to

> The partition table on device hda was unreadable. To create new partitions it must be initialized, causing the loss of ALL DATA on this drive.

you may not have a partition table on that drive or the partition table on the drive may not be recognizable by the partitioning software used in the installation program.

Users who have used programs such as **EZ-BIOS** have experienced similar problems, causing data to be lost (assuming the data was not backed up before the installation began) that could not be recovered.

No matter what type of installation you are performing, backups of the existing data on your systems should always be made.

10.3.4. Using Remaining Space

You have a **swap** and a / (root) partition created, and you have selected the root partition to use the remaining space, but it does not fill the hard drive.

If your hard drive is more than 1024 cylinders, you must create a /**boot** partition if you want the / (root) partition to use all of the remaining space on your hard drive.

10.3.5. The "drive must have a GPT disk label" Error Message

When installing Red Hat Enterprise Linux on a system with UEFI system and using a disk with an existing partitioning layout as the boot drive (the drive where the boot loader is installed), you may encounter the following error message during custom partitioning:

```
sda must have a GPT disk label
```

This happens because the boot drive (in this case, **sda**) has a Master Boot Record (MBR) label, but UEFI systems require a GUID Partition Table (GPT) label. Therefore you can not reuse an existing partitioning layout on a MBR-labeled drive; the disk must be relabeled, which means you will have to create a new partition layout and lose all existing data.

To work around this problem, go back to the screen where you select your partitioning strategy. Select an option *other than* custom partitioning (for example **Use All Space**). Make sure to check the **Review and modify partitioning layout** check box, and click **Next**.

On the following screen, modify the automatically created layout so it suits your needs. After you finish and click **Next**, **Anaconda** will use your layout and relabel the drive automatically.

You can also solve this issue by using a Kickstart file or by relabeling the disk using a different system before you begin the installation. See Section 3.5.2, "Disk Drives with MBR on UEFI Systems" for details. Also see Section A.1.2, "Partitions: Turning One Drive Into Many" for additional information about MBR and GPT.

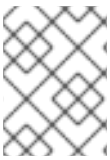
10.3.6. Other Partitioning Problems

If you create partitions manually, but cannot move to the next screen, you probably have not created all the partitions necessary for installation to proceed.

You must have the following partitions as a bare minimum:

- A / (root) partition
- A <swap> partition of type swap

Refer to Section 9.15.5, "Recommended Partitioning Scheme" for more information.



NOTE

When defining a partition's type as swap, do not assign it a mount point. **Anaconda** automatically assigns the mount point for you.

10.4. PROBLEMS AFTER INSTALLATION

10.4.1. Trouble With the Graphical GRUB Screen on an x86-based System?

If you are experiencing problems with GRUB, you may need to disable the graphical boot screen. To do this, become the root user and edit the **/boot/grub/grub.conf** file.

Within the **grub.conf** file, comment out the line which begins with **splashimage** by inserting the **#** character at the beginning of the line.

Press **Enter** to exit the editing mode.

Once the boot loader screen has returned, type **b** to boot the system.

Once you reboot, the **grub.conf** file is reread and any changes you have made take effect.

You may re-enable the graphical boot screen by uncommenting (or adding) the above line back into the **grub.conf** file.

10.4.2. Booting into a Graphical Environment

If you have installed the X Window System but are not seeing a graphical desktop environment once you log into your system, you can start the X Window System graphical interface using the command **startx**.

Once you enter this command and press **Enter**, the graphical desktop environment is displayed.

Note, however, that this is just a one-time fix and does not change the log in process for future log ins.

To set up your system so that you can log in at a graphical login screen, you must edit one file, **/etc/inittab**, by changing just one number in the runlevel section. When you are finished, reboot the computer. The next time you log in, you are presented with a graphical login prompt.

Open a shell prompt. If you are in your user account, become root by typing the **su** command.

Now, type the following to edit the file with **gedit**.

```
gedit /etc/inittab
```

The file **/etc/inittab** opens. Within the first screen, a section of the file which looks like the following appears:

```
# Default runlevel. The runlevels used are:
#   0 - halt (Do NOT set initdefault to this)
```

```
#   1 - Single user mode
#   2 - Multiuser, without NFS (The same as 3, if you do not have networking)
#   3 - Full multiuser mode
#   4 - unused
#   5 - X11
#   6 - reboot (Do NOT set initdefault to this)
#
id:3:initdefault:
```

To change from a console to a graphical login, you should change the number in the line **id:3:initdefault:** from a **3** to a **5**.

WARNING

Change *only* the number of the default runlevel from **3** to **5**.

Your changed line should look like the following:

```
id:5:initdefault:
```

When you are satisfied with your change, save and exit the file using the **Ctrl**+**Q** keys. A window appears and asks if you would like to save the changes. Click **Save**.

The next time you log in after rebooting your system, you are presented with a graphical login prompt.

10.4.3. Problems with the X Window System (GUI)

If you are having trouble getting X (the X Window System) to start, you may not have installed it during your installation.

If you want X, you can either install the packages from the Red Hat Enterprise Linux installation media or perform an upgrade.

If you elect to upgrade, select the X Window System packages, and choose GNOME, KDE, or both, during the upgrade package selection process.

Refer to Section 35.3, “Switching to a Graphical Login” for more detail on installing a desktop environment.

10.4.4. Problems with the X Server Crashing and Non-Root Users

If you are having trouble with the X server crashing when anyone logs in, you may have a full file system (or, a lack of available hard drive space).

To verify that this is the problem you are experiencing, run the following command:

```
df -h
```

The **df** command should help you diagnose which partition is full. For additional information about **df** and an explanation of the options available (such as the **-h** option used in this example), refer to the **df** man page by typing **man df** at a shell prompt.

A key indicator is 100% full or a percentage above 90% or 95% on a partition. The **/home/** and **/tmp/** partitions can sometimes fill up quickly with user files. You can make some room on that partition by removing old files. After you free up some disk space, try running X as the user that was unsuccessful before.

10.4.5. Problems When You Try to Log In

If you did not create a user account in the **firstboot** screens, switch to a console by pressing **Ctrl**+**Alt**+**F2**, log in as root and use the password you assigned to root.

If you cannot remember your root password, boot your system as **linux single**.

If you are using an x86-based system and GRUB is your installed boot loader, type **e** for edit when the GRUB boot screen has loaded. You are presented with a list of items in the configuration file for the boot label you have selected.

Choose the line that starts with **kernel** and type **e** to edit this boot entry.

At the end of the **kernel** line, add:

```
single
```

Press **Enter** to exit edit mode.

Once the boot loader screen has returned, type **b** to boot the system.

Once you have booted into single user mode and have access to the **#** prompt, you must type **passwd root**, which allows you to enter a new password for root. At this point you can type **shutdown -r now** to reboot the system with the new root password.

If you cannot remember your user account password, you must become root. To become root, type **su -** and enter your root password when prompted. Then, type **passwd <username>**. This allows you to enter a new password for the specified user account.

If the graphical login screen does not appear, check your hardware for compatibility issues. The *Hardware Compatibility List* can be found at:

```
https://hardware.redhat.com/
```

10.4.6. Is Your RAM Not Being Recognized?

Sometimes, the kernel does not recognize all of your memory (RAM). You can check this with the **cat /proc/meminfo** command.

Verify that the displayed quantity is the same as the known amount of RAM in your system. If they are not equal, add the following line to the **/boot/grub/grub.conf**:

```
mem=xxM
```

Replace *xx* with the amount of RAM you have in megabytes.

In **/boot/grub/grub.conf**, the above example would look similar to the following:

```
# NOTICE: You have a /boot partition. This means that
#  all kernel paths are relative to /boot/
default=0
timeout=30
splashimage=(hd0,0)/grub/splash.xpm.gz
 title Red Hat Enterprise Linux Client (2.6.32.130.el6.i686)
root (hd0,1)
kernel /vmlinuz-(2.6.32.130.el6.i686 ro root=UUID=04a07c13-e6bf-6d5a-b207-002689545705
mem=1024M
initrd /initrd-(2.6.32.130.el6.i686.img
```

Once you reboot, the changes made to **grub.conf** are reflected on your system.

Once you have loaded the GRUB boot screen, type **e** for edit. You are presented with a list of items in the configuration file for the boot label you have selected.

Choose the line that starts with **kernel** and type **e** to edit this boot entry.

At the end of the **kernel** line, add

```
mem=xxM
```

where *xx* equals the amount of RAM in your system.

Press **Enter** to exit edit mode.

Once the boot loader screen has returned, type **b** to boot the system.

Remember to replace *xx* with the amount of RAM in your system. Press **Enter** to boot.

10.4.7. Your Printer Does Not Work

If you are not sure how to set up your printer or are having trouble getting it to work properly, try using the **Printer Configuration Tool**.

Type the **system-config-printer** command at a shell prompt to launch the **Printer Configuration Tool**. If you are not root, it prompts you for the root password to continue.

10.4.8. Apache HTTP Server or Sendmail Stops Responding During Startup

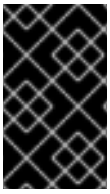
If **Apache HTTP Server** (**httpd**) or **Sendmail** stops responding during startup, make sure the following line is in the **/etc/hosts** file:

```
127.0.0.1  localhost.localdomain  localhost
```

PART II. IBM POWER SYSTEMS – INSTALLATION AND BOOTING

This part of the *Red Hat Enterprise Linux Installation Guide* includes information about installation and basic post-installation troubleshooting for IBM Power Systems servers. IBM Power Systems servers include IBM PowerLinux servers and POWER7 and POWER6 Power Systems servers running Linux.

For advanced installation options, refer to Part IV, “Advanced Installation Options”.



IMPORTANT

Previous releases of Red Hat Enterprise Linux supported 32-bit and 64-bit Power Systems servers (**ppc** and **ppc64** respectively). Red Hat Enterprise Linux 6 supports only 64-bit Power Systems servers (**ppc64**).

CHAPTER 11. PLANNING FOR INSTALLATION ON POWER SYSTEMS SERVERS

11.1. UPGRADE OR INSTALL?

While automated in-place upgrades are now supported, the support is currently limited to AMD64 and Intel 64 systems. If you have an existing installation of Red Hat Enterprise Linux on an IBM Power Systems server, you must perform a clean install to migrate to Red Hat Enterprise Linux 7. A clean install is performed by backing up all data from the system, formatting disk partitions, performing an installation of Red Hat Enterprise Linux 7 from installation media, and then restoring any user data.

11.2. HARDWARE REQUIREMENTS

For installation of Red Hat Enterprise Linux on IBM Power Systems servers, Red Hat supports hard drives connected by a standard internal interface, such as SCSI, SATA, or SAS.

Fibre Channel Host Bus Adapters and multipath devices are supported. Vendor-provided drivers may be required for certain hardware.

Virtualized installation on Power Systems servers is also supported when using Virtual SCSI (vSCSI) adapters in virtual client LPARs.

Note that Red Hat does not support installation to USB drives or SD memory cards.

11.3. INSTALLATION TOOLS

IBM Installation Toolkit is an optional tool that speeds up the installation of Linux and is especially helpful for those unfamiliar with Linux. Use the **IBM Installation Toolkit** for the following actions: [5]

- Install and configure Linux on a non-virtualized Power Systems server.
- Install and configure Linux on servers with previously-configured logical partitions (LPARs, also known as virtualized servers).
- Install IBM service and productivity tools on a new or previously installed Linux system. The IBM service and productivity tools include dynamic logical partition (DLPAR) utilities.
- Upgrade system firmware level on Power Systems servers.
- Perform diagnostics or maintenance operations on previously installed systems.
- Migrate a **LAMP** server (software stack) and application data from a System x to a System p system. A **LAMP** server is a bundle of open source software. **LAMP** is an acronym for Linux, **Apache HTTP Server**, **MySQL** relational database, and PHP (Perl or Python) scripting language.

Documentation for the **IBM Installation Toolkit** for PowerLinux is available in the Linux Information Center at http://pic.dhe.ibm.com/infocenter/lnxinfo/v3r0m0/index.jsp?topic=%2Fliaan%2Fpowerpack.htm

PowerLinux service and productivity tools is an optional set of tools that include hardware service diagnostic aids, productivity tools, and installation aids for Linux operating systems on IBM servers based on POWER7, POWER6, POWER5, and POWER4 technology.

Documentation for the service and productivity tools is available in the Linux Information Center at http://pic.dhe.ibm.com/infocenter/lnxinfo/v3r0m0/index.jsp?topic=%2Fliaau%2Fliaauraskickoff.htm

11.4. PREPARATION FOR IBM POWER SYSTEMS SERVERS

IMPORTANT

Ensure that the real-base boot parameter is set to **c00000**, otherwise you might see errors such as:

```
DEFAULT CATCH!, exception-handler=fff00300
```

IBM Power Systems servers offer many options for partitioning, virtual or native devices, and consoles.

If you are using a non-partitioned system, you do not need any pre-installation setup. For systems using the HVSI serial console, hook up your console to the T2 serial port.

If using a partitioned system the steps to create the partition and start the installation are largely the same. You should create the partition at the HMC and assign some CPU and memory resources, as well as SCSI and Ethernet resources, which can be either virtual or native. The HMC create partition wizard steps you through the creation.

For more information on creating the partition, refer to the *Partitioning for Linux with an HMC* PDF in the IBM Systems Hardware Information Center at:
http://pic.dhe.ibm.com/infocenter/powersys/v3r1m5/topic/iphbi_p5/iphbibook.pdf

If you are using virtual SCSI resources, rather than native SCSI, you must configure a 'link' to the virtual SCSI serving partition, and then configure the virtual SCSI serving partition itself. You create a 'link' between the virtual SCSI client and server slots using the HMC. You can configure a virtual SCSI server on either Virtual I/O Server (VIOS) or IBM i, depending on which model and options you have.

If you are installing using Intel iSCSI Remote Boot, all attached iSCSI storage devices must be disabled. Otherwise, the installation will succeed but the installed system will not boot.

For more information on using virtual devices, see the IBM Redbooks publication *Virtualizing an Infrastructure with System p and Linux* at: http://publib-b.boulder.ibm.com/abstracts/sg247499.html

Once you have your system configured, you need to Activate from the HMC or power it on. Depending on what type of install you are doing, you may need to configure SMS to correctly boot the system into the installation program.

11.5. RAID AND OTHER DISK DEVICES

IMPORTANT

Red Hat Enterprise Linux 6 uses **mdraid** instead of **dmraid** for installation onto Intel BIOS RAID sets. These sets are detected automatically, and devices with Intel ISW metadata are recognized as mdraid instead of dmraid. Note that the device node names of any such devices under **mdraid** are different from their device node names under **dmraid**. Therefore, special precautions are necessary when you migrate systems with Intel BIOS RAID sets.

Local modifications to **/etc/fstab**, **/etc/crypttab** or other configuration files which refer to devices by their device node names will not work in Red Hat Enterprise Linux 6. Before migrating these files, you must therefore edit them to replace device node paths with device UUIDs instead. You can find the UUIDs of devices with the **blkid** command.

11.5.1. Hardware RAID

RAID, or Redundant Array of Independent Disks, allows a group, or array, of drives to act as a single device. Configure any RAID functions provided by the mainboard of your computer, or attached controller cards, before you begin the installation process. Each active RAID array appears as one drive within Red Hat Enterprise Linux.

On systems with more than one hard drive you may configure Red Hat Enterprise Linux to operate several of the drives as a Linux RAID array without requiring any additional hardware.

11.5.2. Software RAID

You can use the Red Hat Enterprise Linux installation program to create Linux software RAID arrays, where RAID functions are controlled by the operating system rather than dedicated hardware. These functions are explained in detail in Section 16.17, " Creating a Custom Layout or Modifying the Default Layout ".

11.5.3. FireWire and USB Disks

Some FireWire and USB hard disks may not be recognized by the Red Hat Enterprise Linux installation system. If configuration of these disks at installation time is not vital, disconnect them to avoid any confusion.

NOTE

You can connect and configure external FireWire and USB hard disks after installation. Most such devices are automatically recognized and available for use once connected.

11.6. DO YOU HAVE ENOUGH DISK SPACE?

Nearly every modern-day operating system (OS) uses *disk partitions*, and Red Hat Enterprise Linux is no exception. When you install Red Hat Enterprise Linux, you may have to work with disk partitions. If you have not worked with disk partitions before (or need a quick review of the basic concepts), refer to Appendix A, *An Introduction to Disk Partitions* before proceeding.

The disk space used by Red Hat Enterprise Linux must be separate from the disk space used by other OSes you may have installed on your system.

Before you start the installation process, you must

- have enough *unpartitioned*[6] disk space for the installation of Red Hat Enterprise Linux, or
- have one or more partitions that may be deleted, thereby freeing up enough disk space to install Red Hat Enterprise Linux.

To gain a better sense of how much space you really need, refer to the recommended partitioning sizes discussed in Section 16.17.5, "Recommended Partitioning Scheme".

11.7. CHOOSE A BOOT METHOD

Installing from a DVD requires that you have purchased a Red Hat Enterprise Linux product, you have a Red Hat Enterprise Linux 6.9 DVD, and you have a DVD drive on a system that supports booting from it. Refer to Chapter 2, *Making Media* for instructions to make an installation DVD.

Other than booting from an installation DVD, you can also boot the Red Hat Enterprise Linux installation program from *minimal boot media* in the form of a bootable CD. After you boot the system with boot CD, you complete the installation from a different installation source, such as a local hard drive or a location on a network. Refer to Section 2.2, "Making Minimal Boot Media" for instructions on making boot CDs.

---

[5] Parts of this section were previously published at IBM's *Linux information for IBM systems* resource at http://pic.dhe.ibm.com/infocenter/lnxinfo/v3r0m0/index.jsp?topic=%2Fliaay%2Ftools_overview.htm

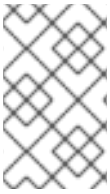
[6] Unpartitioned disk space means that available disk space on the hard drives you are installing to has not been divided into sections for data. When you partition a disk, each partition behaves like a separate disk drive.

CHAPTER 12. PREPARING FOR INSTALLATION

12.1. PREPARING FOR A NETWORK INSTALLATION

IMPORTANT

The eHEA module fails to initialize if 16 GB *huge pages* are assigned to a system or partition and the kernel command line does not contain the huge page parameters. Therefore, when you perform a network installation through an IBM eHEA ethernet adapter, you cannot assign huge pages to the system or partition during the installation. Large pages should work.



NOTE

Make sure no installation DVD (or any other type of DVD or CD) is in your system's CD or DVD drive if you are performing a network-based installation. Having a DVD or CD in the drive might cause unexpected errors.

Ensure that you have boot media available on CD, DVD, or a USB storage device such as a flash drive.

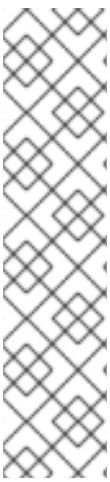
The Red Hat Enterprise Linux installation medium must be available for either a network installation (via NFS, FTP, HTTP, or HTTPS) or installation via local storage. Use the following steps if you are performing an NFS, FTP, HTTP, or HTTPS installation.

The NFS, FTP, HTTP, or HTTPS server to be used for installation over the network must be a separate, network-accessible server. It must provide the complete contents of the installation DVD-ROM.

NOTE

anaconda has the ability to test the integrity of the installation media. It works with the DVD, hard drive ISO, and NFS ISO installation methods. Red Hat recommends that you test all installation media before starting the installation process, and before reporting any installation-related bugs (many of the bugs reported are actually due to improperly-burned DVDs). To use this test, type the following command at the **yaboot:** prompt:

```
linux mediacheck
```



NOTE

The public directory used to access the installation files over FTP, NFS, HTTP, or HTTPS is mapped to local storage on the network server. For example, the local directory **/var/www/inst/rhel6.9** on the network server can be accessed as **http://network.server.com/inst/rhel6.9**.

In the following examples, the directory on the installation staging server that will contain the installation files will be specified as ***/location/of/disk/space***. The directory that will be made publicly available via FTP, NFS, HTTP, or HTTPS will be specified as */publicly_available_directory*. For example, ***/location/of/disk/space*** may be a directory you create called **/var/isos**. ***/publicly_available_directory*** might be **/var/www/html/rhel6.9**, for an HTTP install.

In the following, you will require an *ISO image*. An ISO image is a file containing an exact copy of the content of a DVD. To create an ISO image from a DVD use the following command:

```
dd if=/dev/dvd of=/path_to_image/name_of_image.iso
```

where *dvd* is your DVD drive device, *name_of_image* is the name you give to the resulting ISO image file, and *path_to_image* is the path to the location on your system where the resulting ISO image will be stored.

To copy the files from the installation DVD to a Linux instance, which acts as an installation staging server, continue with either Section 12.1.1, “Preparing for FTP, HTTP, and HTTPS Installation” or Section 12.1.2, “Preparing for an NFS Installation” .

12.1.1. Preparing for FTP, HTTP, and HTTPS Installation

WARNING

If your **Apache** web server or **tftp** FTP server configuration enables SSL security, make sure to only enable the **TLSv1** protocol, and disable **SSLv2** and **SSLv3**. This is due to the POODLE SSL vulnerability (CVE-2014-3566). See https://access.redhat.com/solutions/1232413 for details about securing **Apache**, and https://access.redhat.com/solutions/1234773 for information about securing **tftp**.

Extract the files from the ISO image of the installation DVD and place them in a directory that is shared over FTP, HTTP, or HTTPS.

Next, make sure that the directory is shared via FTP, HTTP, or HTTPS, and verify client access. Test to see whether the directory is accessible from the server itself, and then from another machine on the same subnet to which you will be installing.

12.1.2. Preparing for an NFS Installation

For NFS installation it is not necessary to extract all the files from the ISO image. It is sufficient to make the ISO image itself, the **install.img** file, and optionally the **product.img** file available on the network server via NFS.

1. Transfer the ISO image to the NFS exported directory. On a Linux system, run:

   ```
   mv /path_to_image/name_of_image.iso /publicly_available_directory/
   ```

   where *path_to_image* is the path to the ISO image file, *name_of_image* is the name of the ISO image file, and *publicly_available_directory* is a directory that is available over NFS or that you intend to make available over NFS.

2. Use a SHA256 checksum program to verify that the ISO image that you copied is intact. Many SHA256 checksum programs are available for various operating systems. On a Linux system, run:

```
$ sha256sum name_of_image.iso
```

where *name_of_image* is the name of the ISO image file. The SHA256 checksum program displays a string of 64 characters called a *hash*. Compare this hash to the hash displayed for this particular image on the **Downloads** page in the Red Hat Customer Portal (refer to Chapter 1, *Obtaining Red Hat Enterprise Linux*). The two hashes should be identical.

3. Copy the **images/** directory from inside the ISO image to the same directory in which you stored the ISO image file itself. Enter the following commands:

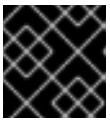
```
mount -t iso9660 /path_to_image/name_of_image.iso /mount_point -o loop,ro
cp -pr /mount_point/images /publicly_available_directory/
umount /mount_point
```

where ***path_to_image*** is the path to the ISO image file, ***name_of_image*** is the name of the ISO image file, and ***mount_point*** is a mount point on which to mount the image while you copy files from the image. For example:

```
mount -t iso9660 /var/isos/RHEL6.iso /mnt/tmp -o loop,ro
cp -pr /mnt/tmp/images /var/isos/
umount /mnt/tmp
```

The ISO image file and an **images/** directory are now present, side-by-side, in the same directory.

4. Verify that the **images/** directory contains at least the **install.img** file, without which installation cannot proceed. Optionally, the **images/** directory should contain the **product.img** file, without which only the packages for a **Minimal** installation will be available during the package group selection stage (refer to Section 16.19, "Package Group Selection").



IMPORTANT

install.img and **product.img** must be the only files in the **images/** directory.

5. Ensure that an entry for the publicly available directory exists in the **/etc/exports** file on the network server so that the directory is available via NFS.

   To export a directory read-only to a specific system, use:

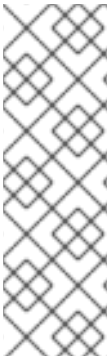
```
/publicly_available_directory client.ip.address (ro)
```

   To export a directory read-only to all systems, use:

```
/publicly_available_directory * (ro)
```

6. On the network server, start the NFS daemon (on a Red Hat Enterprise Linux system, use **/sbin/service nfs start**). If NFS is already running, reload the configuration file (on a Red Hat Enterprise Linux system use **/sbin/service nfs reload**).

7. Be sure to test the NFS share following the directions in the Red Hat Enterprise Linux Deployment Guide. Refer to your NFS documentation for details on starting and stopping the NFS server.



NOTE

anaconda has the ability to test the integrity of the installation media. It works with the DVD, hard drive ISO, and NFS ISO installation methods. We recommend that you test all installation media before starting the installation process, and before reporting any installation-related bugs (many of the bugs reported are actually due to improperly-burned DVDs). To use this test, type the following command at the **boot:** prompt:

```
linux mediacheck
```

12.2. PREPARING FOR A HARD DRIVE INSTALLATION

NOTE

Hard drive installations only work from ext2, ext3, ext4, or FAT file systems. You cannot use a hard drives formatted for any other file system as an installation source for Red Hat Enterprise Linux.

To check the file system of a hard drive partition on a Windows operating system, use the **Disk Management** tool. To check the file system of a hard drive partition on a Linux operating system, use the **fdisk** tool.

IMPORTANT

You cannot use ISO files on partitions controlled by LVM (Logical Volume Management).

Use this option to install Red Hat Enterprise Linux on systems without a DVD drive or network connection.

Hard drive installations use the following files:

- an *ISO image* of the installation DVD. An ISO image is a file that contains an exact copy of the content of a DVD.
- an **install.img** file extracted from the ISO image.
- optionally, a **product.img** file extracted from the ISO image.

With these files present on a hard drive, you can choose **Hard drive** as the installation source when you boot the installation program (refer to Section 15.3, “Installation Method”).

Ensure that you have boot media available on CD, DVD, or a USB storage device such as a flash drive.

To prepare a hard drive as an installation source, follow these steps:

1. Obtain an ISO image of the Red Hat Enterprise Linux installation DVD (refer to Chapter 1, *Obtaining Red Hat Enterprise Linux*). Alternatively, if you have the DVD on physical media, you can create an image of it with the following command on a Linux system:

   ```
   dd if=/dev/dvd of=/path_to_image/name_of_image.iso
   ```

where *dvd* is your DVD drive device, *name_of_image* is the name you give to the resulting ISO image file, and *path_to_image* is the path to the location on your system where the resulting ISO image will be stored.

2. Transfer the ISO image to the hard drive.

   The ISO image must be located on a hard drive that is either internal to the computer on which you will install Red Hat Enterprise Linux, or on a hard drive that is attached to that computer by USB.

3. Use a SHA256 checksum program to verify that the ISO image that you copied is intact. Many SHA256 checksum programs are available for various operating systems. On a Linux system, run:

   ```
   $ sha256sum name_of_image.iso
   ```

   where *name_of_image* is the name of the ISO image file. The SHA256 checksum program displays a string of 64 characters called a *hash*. Compare this hash to the hash displayed for this particular image on the **Downloads** page in the Red Hat Customer Portal (refer to Chapter 1, *Obtaining Red Hat Enterprise Linux*). The two hashes should be identical.

4. Copy the **images/** directory from inside the ISO image to the same directory in which you stored the ISO image file itself. Enter the following commands:

   ```
   mount -t iso9660 /path_to_image/name_of_image.iso /mount_point -o loop,ro
   cp -pr /mount_point/images /publicly_available_directory/
   umount /mount_point
   ```

   where ***path_to_image*** is the path to the ISO image file, ***name_of_image*** is the name of the ISO image file, and ***mount_point*** is a mount point on which to mount the image while you copy files from the image. For example:

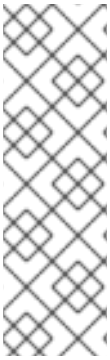
   ```
   mount -t iso9660 /var/isos/RHEL6.iso /mnt/tmp -o loop,ro
   cp -pr /mnt/tmp/images /var/isos/
   umount /mnt/tmp
   ```

   The ISO image file and an **images/** directory are now present, side-by-side, in the same directory.

5. Verify that the **images/** directory contains at least the **install.img** file, without which installation cannot proceed. Optionally, the **images/** directory should contain the **product.img** file, without which only the packages for a **Minimal** installation will be available during the package group selection stage (refer to Section 9.17, “Package Group Selection”).

IMPORTANT

install.img and **product.img** must be the only files in the **images/** directory.



NOTE

anaconda has the ability to test the integrity of the installation media. It works with the DVD, hard drive ISO, and NFS ISO installation methods. We recommend that you test all installation media before starting the installation process, and before reporting any installation-related bugs (many of the bugs reported are actually due to improperly-burned DVDs). To use this test, type the following command at the **boot:** prompt:

```
linux mediacheck
```

CHAPTER 13. UPDATING DRIVERS DURING INSTALLATION ON IBM POWER SYSTEMS SERVERS

In most cases, Red Hat Enterprise Linux already includes drivers for the devices that make up your system. However, if your system contains hardware that has been released very recently, drivers for this hardware might not yet be included. Sometimes, a driver update that provides support for a new device might be available from Red Hat or your hardware vendor on a *driver disc* that contains *rpm packages*. Typically, the driver disc is available for download as an *ISO image file*.

Often, you do not need the new hardware during the installation process. For example, if you use a DVD to install to a local hard drive, the installation will succeed even if drivers for your network card are not available. In situations like this, complete the installation and add support for the piece of hardware afterward – refer to Section 35.1.1, "Driver Update rpm Packages" for details of adding this support.

In other situations, you might want to add drivers for a device during the installation process to support a particular configuration. For example, you might want to install drivers for a network device or a storage adapter card to give the installer access to the storage devices that your system uses. You can use a driver disc to add this support during installation in one of two ways:

1. place the ISO image file of the driver disc in a location accessible to the installer:
    1. on a local hard drive
    2. a USB flash drive
2. create a driver disc by extracting the image file onto:
    1. a CD
    2. a DVD

    Refer to the instructions for making installation discs in Section 2.1, "Making an Installation DVD" for more information on burning ISO image files to CD or DVD.

If Red Hat, your hardware vendor, or a trusted third party told you that you will require a driver update during the installation process, choose a method to supply the update from the methods described in this chapter and test it before beginning the installation. Conversely, do not perform a driver update during installation unless you are certain that your system requires it. Although installing an unnecessary driver update will not cause harm, the presence of a driver on a system for which it was not intended can complicate support.

13.1. LIMITATIONS OF DRIVER UPDATES DURING INSTALLATION

Unfortunately, some situations persist in which you cannot use a driver update to provide drivers during installation:

Devices already in use

You cannot use a driver update to replace drivers that the installation program has already loaded. Instead, you must complete the installation with the drivers that the installation program loaded and update to the new drivers after installation, or, if you need the new drivers for the installation process, consider performing an initial RAM disk driver update – refer to Section 13.2.3, "Preparing an Initial RAM Disk Update".

Devices with an equivalent device available

Because all devices of the same type are initialized together, you cannot update drivers for a device

if the installation program has loaded drivers for a similar device. For example, consider a system that has two different network adapters, one of which has a driver update available. The installation program will initialize both adapters at the same time, and therefore, you will not be able to use this driver update. Again, complete the installation with the drivers loaded by the installation program and update to the new drivers after installation, or use an initial RAM disk driver update.

13.2. PREPARING FOR A DRIVER UPDATE DURING INSTALLATION

If a driver update is necessary and available for your hardware, Red Hat or a trusted third party such as the hardware vendor will typically provide it in the form of an image file in ISO format. Some methods of performing a driver update require you to make the image file available to the installation program, while others require you to use the image file to make a driver update disk:

Methods that use the image file itself

- local hard drive
- USB flash drive

Methods that use a driver update disk produced from an image file

- CD
- DVD

Choose a method to provide the driver update, and refer to Section 13.2.1, “Preparing to Use a Driver Update Image File”, Section 13.2.2, “Preparing a Driver Disc” , or Section 13.2.3, “Preparing an Initial RAM Disk Update”. Note that you can use a USB storage device either to provide an image file, or as a driver update disk.

13.2.1. Preparing to Use a Driver Update Image File

13.2.1.1. Preparing to use an image file on local storage

To make the ISO image file available on local storage, such as a hard drive or USB flash drive, you must first determine whether you want to install the updates automatically or select them manually.

For manual installations, copy the file onto the storage device. You can rename the file if you find it helpful to do so, but you must not change the filename extension, which must remain **.iso**. In the following example, the file is named **dd.iso**:

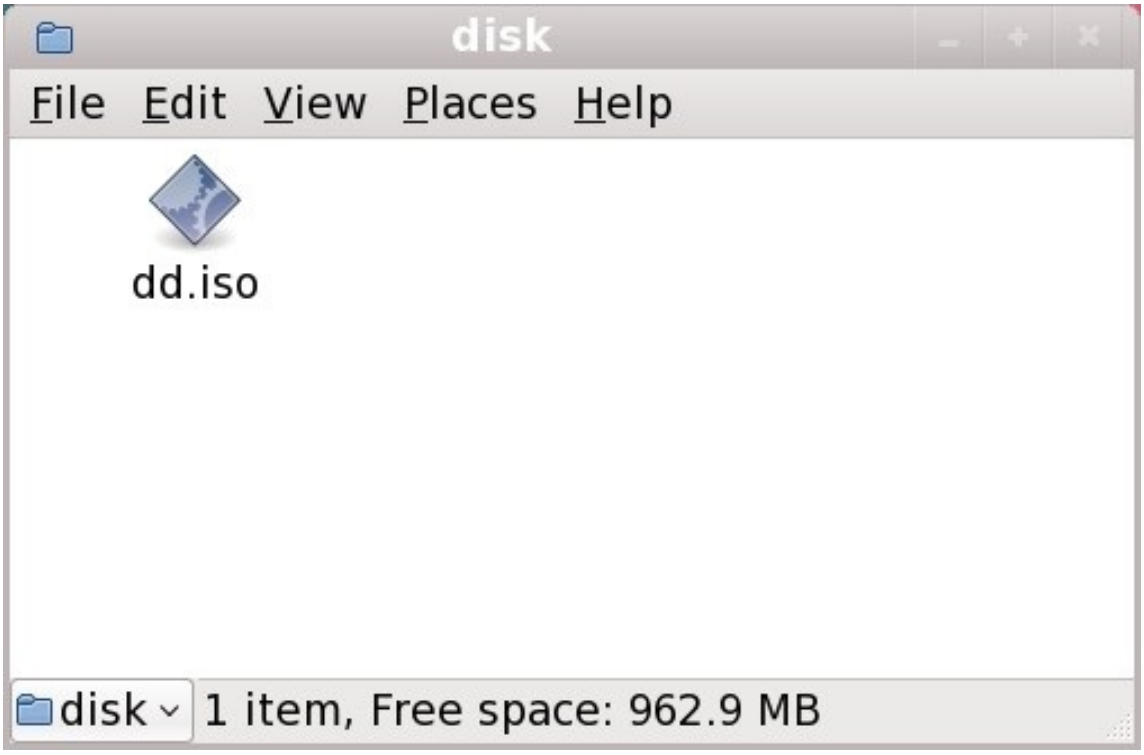


Figure 13.1. Content of a USB flash drive holding a driver update image file

Note that if you use this method, the storage device will contain only a single file. This differs from driver discs on formats such as CD and DVD, which contain many files. The ISO image file contains all of the files that would normally be on a driver disc.

Refer to Section 13.3.2, "Let the Installer Prompt You for a Driver Update" and Section 13.3.3, "Use a Boot Option to Specify a Driver Update Disk" to learn how to select the driver update manually during installation.

For automatic installations, you will need to extract the ISO to the root directory of the storage device rather than simply copy it. Copying the ISO is only effective for manual installations. You must also change the file system label of the device to **OEMDRV**.

The installation program will then automatically examine it for driver updates and load any that it detects. This behavior is controlled by the **dlabel=on** boot option, which is enabled by default. Refer to Section 6.3.1, "Let the Installer Find a Driver Update Disk Automatically" .

13.2.2. Preparing a Driver Disc

You can create a driver update disc on CD or DVD.

13.2.2.1. Creating a driver update disc on CD or DVD

IMPORTANT

CD/DVD Creator is part of the GNOME desktop. If you use a different Linux desktop, or a different operating system altogether, you will need to use another piece of software to create the CD or DVD. The steps will be generally similar.

Make sure that the software that you choose can create CDs or DVDs from image files. While this is true of most CD and DVD burning software, exceptions exist. Look for a button or menu entry labeled **burn from image** or similar. If your software lacks this feature, or you do not select it, the resulting disc will hold only the image file itself, instead of the contents of the image file.

1. Use the desktop file manager to locate the ISO image file of the driver disc, supplied to you by Red Hat or your hardware vendor.

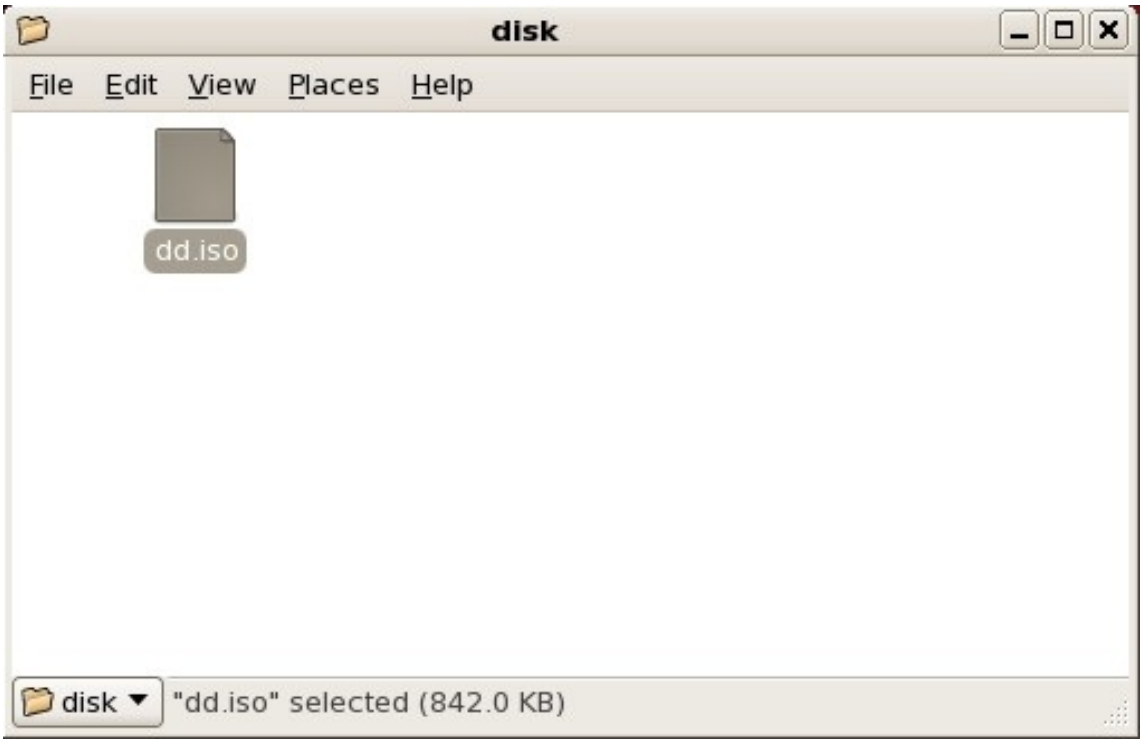


Figure 13.2. A typical .iso file displayed in a file manager window

2. Right-click on this file and choose **Write to disc**. You will see a window similar to the following:

Figure 13.3. CD/DVD Creator's Write to Disc dialog

3. Click the **Write** button. If a blank disc is not already in the drive, **CD/DVD Creator** will prompt you to insert one.

After you burn a driver update disc CD or DVD, verify that the disc was created successfully by inserting it into your system and browsing to it using the file manager. You should see a single file named **rhdd3** and a directory named **rpms**:

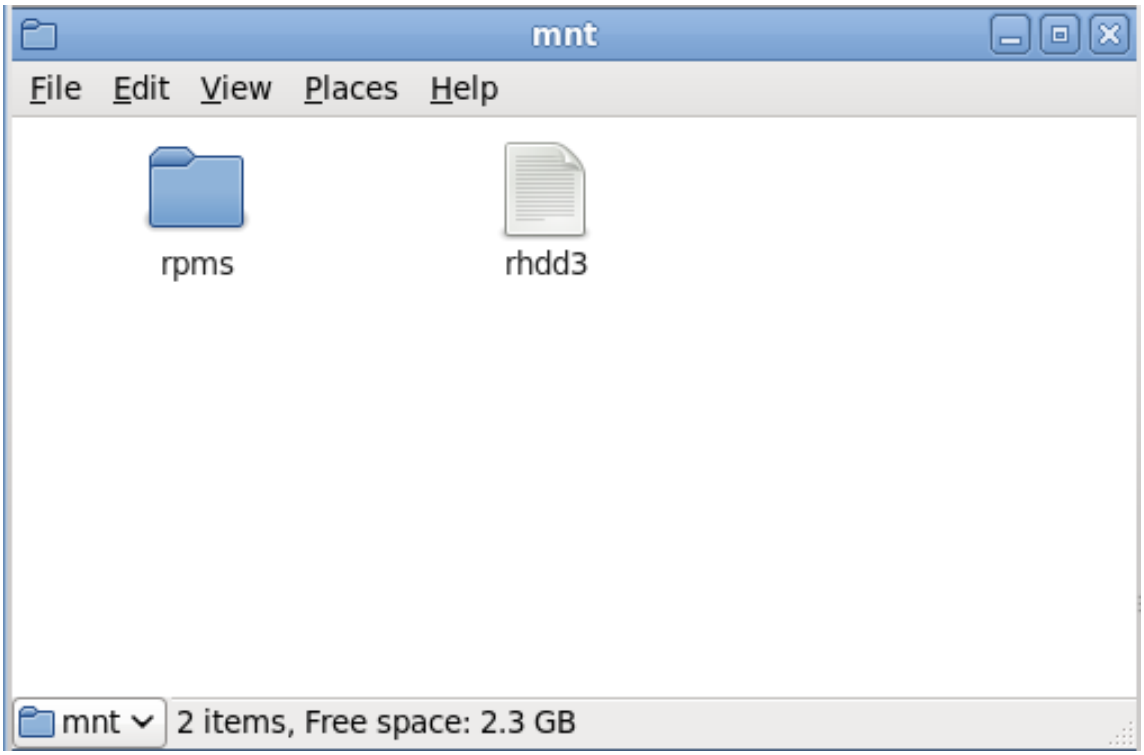


Figure 13.4. Contents of a typical driver update disc on CD or DVD

If you see only a single file ending in **.iso**, then you have not created the disc correctly and should try again. Ensure that you choose an option similar to **burn from image** if you use a Linux desktop other than GNOME or if you use a different operating system.

Refer to Section 13.3.2, "Let the Installer Prompt You for a Driver Update" and Section 13.3.3, "Use a Boot Option to Specify a Driver Update Disk" to learn how to use the driver update disc during installation.

13.2.3. Preparing an Initial RAM Disk Update

IMPORTANT

This is an advanced procedure that you should consider only if you cannot perform a driver update with any other method.

The Red Hat Enterprise Linux installation program can load updates for itself early in the installation process from a *RAM disk* — an area of your computer's memory that temporarily behaves as if it were a disk. You can use this same capability to load driver updates. To perform a driver update during installation, your computer must be able to boot from a **yaboot** installation server, and you must have one available on your network. Refer to Chapter 30, *Setting Up an Installation Server* for instructions on using a **yaboot** installation server.

To make the driver update available on your installation server:

1. Place the driver update image file on your installation server. Usually, you would do this by downloading it to the server from a location on the Internet specified by Red Hat or your hardware vendor. Names of driver update image files end in **.iso**.
2. Copy the driver update image file into the **/tmp/initrd_update** directory.
3. Rename the driver update image file to **dd.img**.
4. At the command line, change into the **/tmp/initrd_update** directory, type the following command, and press **Enter**:

```
find . | cpio --quiet -o -H newc | gzip -9 >/tmp/initrd_update.img
```

5. Copy the file **/tmp/initrd_update.img** into the directory the holds the target that you want to use for installation. This directory is placed under the **/var/lib/tftpboot/yaboot/** directory. For example, **/var/lib/tftpboot/yaboot/rhel6/** might hold the **yaboot** installation target for Red Hat Enterprise Linux 6.
6. Edit the **/var/lib/tftpboot/yaboot/yaboot.conf** file to include an entry that includes the initial RAM disk update that you just created, in the following format:

```
image=target/vmlinuz
label=target-dd
initrd=target/initrd.img,target/dd.img
```

   Where *target* is the target that you want to use for installation.

Refer to Section 13.3.4, "Select an Installation Server Target That Includes a Driver Update" to learn how to use an initial RAM disk update during installation.

Example 13.1. Preparing an initial RAM disk update from a driver update image file

In this example, **driver_update.iso** is a driver update image file that you downloaded from the Internet to a directory on your installation server. The target on your installation server that you want to boot from is located in **/var/lib/tftpboot/yaboot/rhel6/**

At the command line, change to the directory that holds the file and enter the following commands:

```
$ cp driver_update.iso /tmp/initrd_update/dd.img
$ cd /tmp/initrd_update
$ find . | cpio --quiet -c -o -H newc | gzip -9 >/tmp/initrd_update.img
$ cp /tmp/initrd_update.img /tftpboot/yaboot/rhel6/dd.img
```

Edit the **/var/lib/tftpboot/yaboot/yaboot.conf** file and include the following entry:

```
image=rhel6/vmlinuz
label=rhel6-dd
initrd=rhel6/initrd.img,rhel6/dd.img
```

13.3. PERFORMING A DRIVER UPDATE DURING INSTALLATION

You can perform a driver update during installation in the following ways:

- let the installer automatically find a driver update disk.
- let the installer prompt you for a driver update.
- use a boot option to specify a driver update disk.

13.3.1. Let the Installer Find a Driver Update Disk Automatically

Attach a block device with the filesystem label **OEMDRV** before starting the installation process. The installer will automatically examine the device and load any driver updates that it detects and will not prompt you during the process. Refer to Section 13.2.1.1, “Preparing to use an image file on local storage” to prepare a storage device for the installer to find.

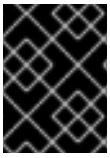
13.3.2. Let the Installer Prompt You for a Driver Update

1. Begin the installation normally for whatever method you have chosen. If the installer cannot load drivers for a piece of hardware that is essential for the installation process (for example, if it cannot detect any network or storage controllers), it prompts you to insert a driver update disk:

Figure 13.5. The no driver found dialog

2. Select **Use a driver disk** and refer to Section 13.4, "Specifying the Location of a Driver Update Image File or a Driver Update Disk".

13.3.3. Use a Boot Option to Specify a Driver Update Disk



IMPORTANT

This method only works to introduce completely new drivers, not to update existing drivers.

1. Type **linux dd** at the boot prompt at the start of the installation process and press **Enter**. The installer prompts you to confirm that you have a driver disk:

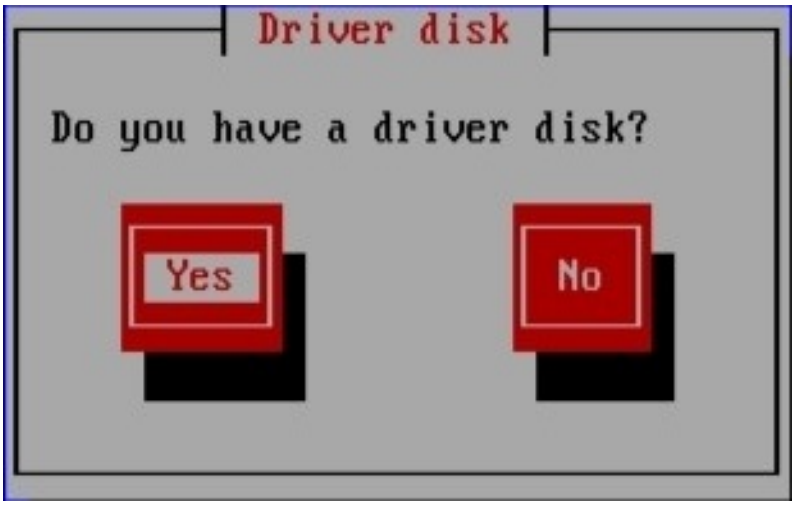


Figure 13.6. The driver disk prompt

2. Insert the driver update disk that you created on CD, DVD, or USB flash drive and select **Yes**. The installer examines the storage devices that it can detect. If there is only one possible location that could hold a driver disk (for example, the installer detects the presence of a DVD drive, but no other storage devices) it will automatically load any driver updates that it finds at this location.

   If the installer finds more than one location that could hold a driver update, it prompts you to specify the location of the update. See Section 13.4, "Specifying the Location of a Driver Update Image File or a Driver Update Disk".

13.3.4. Select an Installation Server Target That Includes a Driver Update

1. Configure the computer to boot from the network interface by selecting **Select Boot Options** in the SMS menu, then **Select Boot/Install Device**. Finally, select your network device from the list of available devices.

2. In the **yaboot** installation server environment, choose the boot target that you prepared on your installation server. For example, if you labeled this environment **rhel6-dd** in the **/var/lib/tftpboot/yaboot/yaboot.conf** file on your installation server, type **rhel6-dd** at the prompt and press **Enter**.

Refer to Section 13.2.3, "Preparing an Initial RAM Disk Update" and Chapter 30, *Setting Up an Installation Server* for instructions on using a **yaboot** installation server to perform an update during installation. Note that this is an advanced procedure — do not attempt it unless other methods of performing a driver update fail.

13.4. SPECIFYING THE LOCATION OF A DRIVER UPDATE IMAGE FILE OR A DRIVER UPDATE DISK

If the installer detects more than one possible device that could hold a driver update, it prompts you to select the correct device. If you are not sure which option represents the device on which the driver update is stored, try the various options in order until you find the correct one.

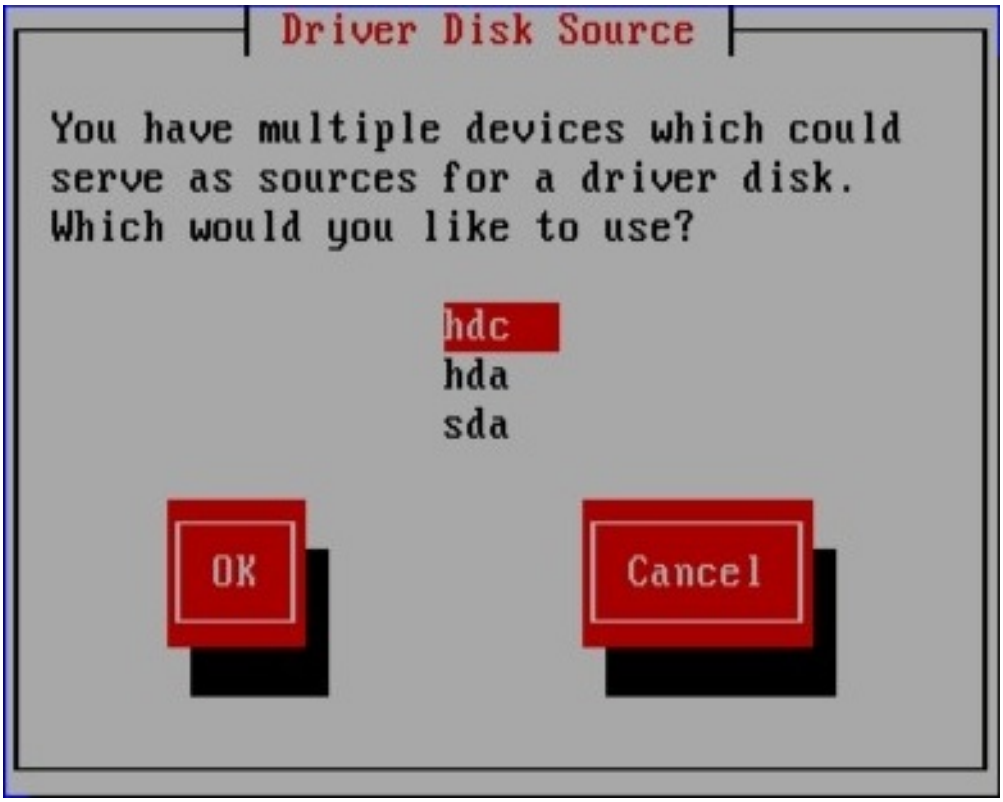


Figure 13.7. Selecting a driver disk source

If the device that you choose contains no suitable update media, the installer will prompt you to make another choice.

If you made a driver update disk on CD, DVD, or USB flash drive, the installer now loads the driver update. However, if the device that you selected is a type of device that could contain more than one partition (whether the device currently has more than one partition or not), the installer might prompt you to select the partition that holds the driver update.

Figure 13.8. Selecting a driver disk partition

The installer prompts you to specify which file contains the driver update:

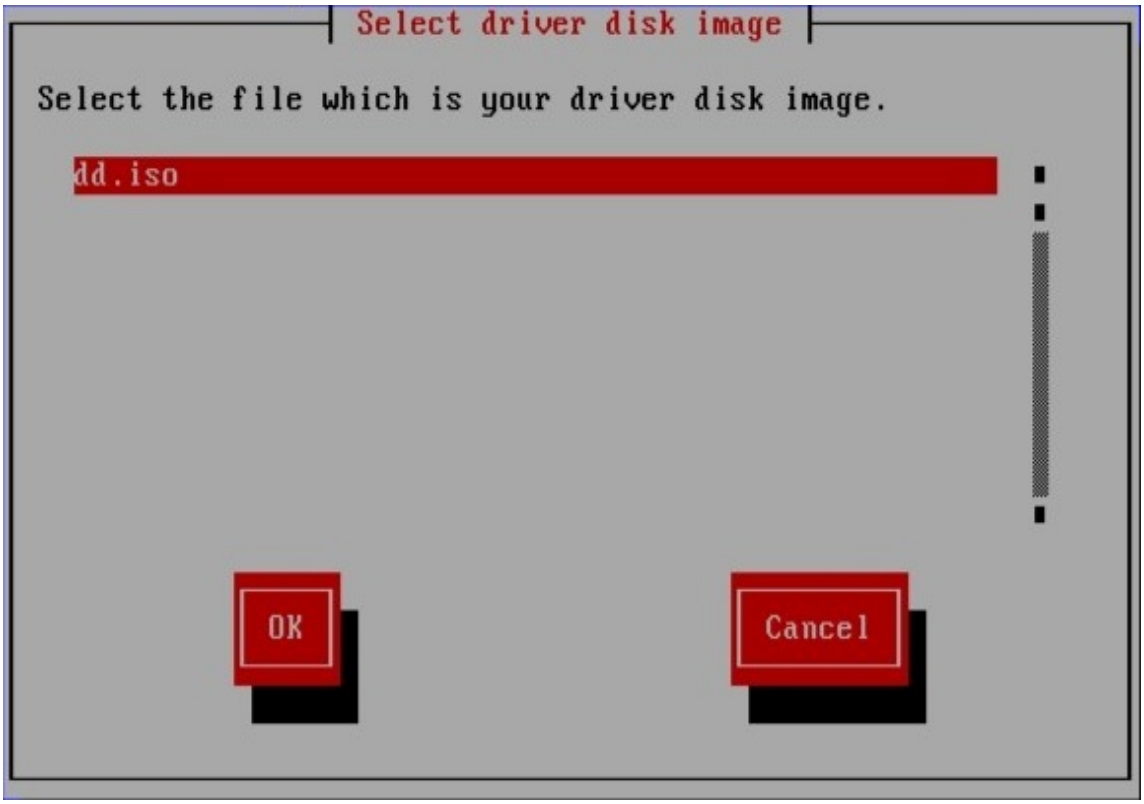


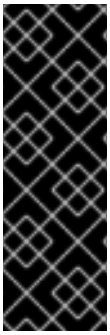
Figure 13.9. Selecting an ISO image

Expect to see these screens if you stored the driver update on an internal hard drive or on a USB storage device. You should not see them if the driver update is on a CD or DVD.

Regardless of whether you are providing a driver update in the form of an image file or with a driver update disk, the installer now copies the appropriate update files into a temporary storage area (located in system RAM and not on disk). The installer might ask whether you would like to use additional driver updates. If you select **Yes**, you can load additional updates in turn. When you have no further driver

updates to load, select **No**. If you stored the driver update on removable media, you can now safely eject or disconnect the disk or device. The installer no longer requires the driver update, and you can re-use the media for other purposes.

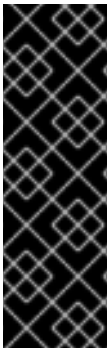
CHAPTER 14. BOOTING THE INSTALLER



IMPORTANT

Graphical installation is recommended. Because Power Systems servers primarily use text consoles, **anaconda** will not automatically start a graphical installation. However, the graphical installer offers more features and customization and is recommended if your system has a graphical display.

To start a graphical installation, pass the **vnc** boot option (refer to Section 28.2.1, "Enabling Remote Access with VNC").



IMPORTANT

On some machines **yaboot** may not boot, returning the error message:

```
Cannot load initrd.img: Claim failed for initrd memory at 02000000 rc=ffffffff
```

To work around this issue, change **real-base** to **c00000**. You can obtain the value of **real-base** from the OpenFirmware prompt with the **printenv** command and set the value with the **setenv** command.

To boot an IBM Power Systems server from a DVD, you must specify the install boot device in the **System Management Services** (SMS) menu.

To enter the **System Management Services** GUI, press the **1** key during the boot process when you hear the chime sound. This brings up a graphical interface similar to the one described in this section.

On a text console, press **1** when the self test is displaying the banner along with the tested components:

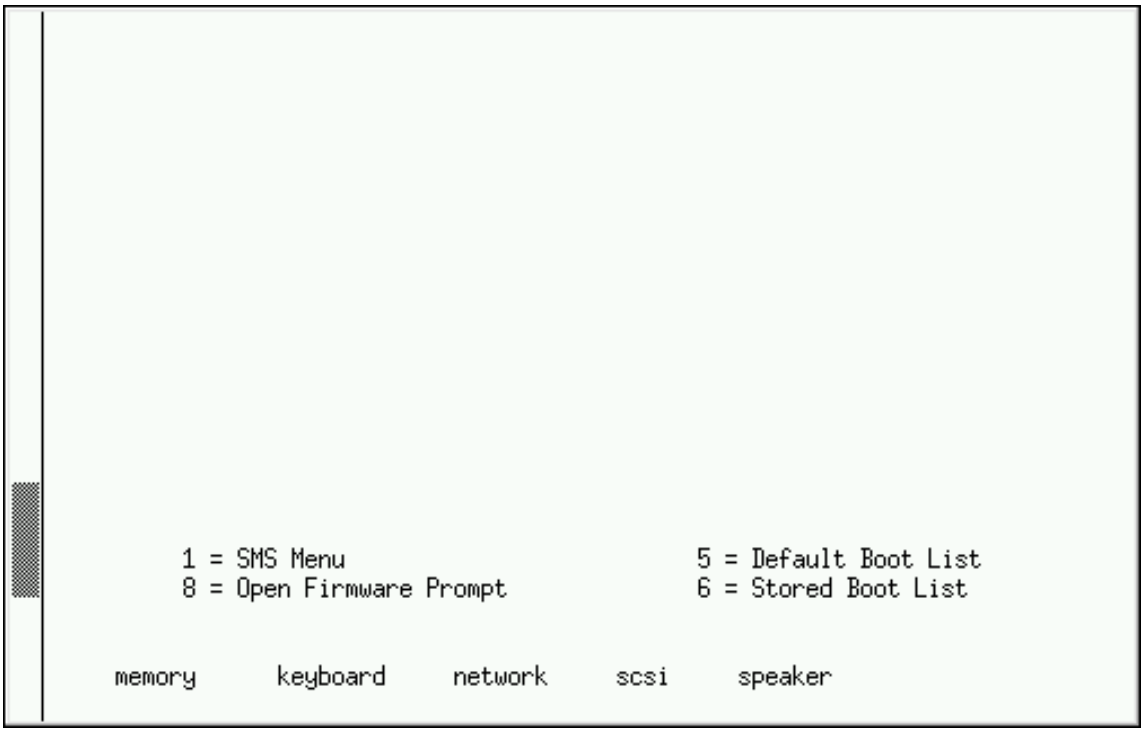


Figure 14.1. SMS console

Once in the SMS menu, select the option for **Select Boot Options**. In that menu, specify **Select Install**

or Boot a Device. There, select **CD/DVD**, and then the bus type (in most cases SCSI). If you are uncertain, you can select to view all devices. This scans all available buses for boot devices, including network adapters and hard drives.

Finally, select the device containing the installation DVD. **Yaboot** is loaded from this device and you are presented with a **boot:** prompt. To begin a graphical installation, pass the **vnc** boot option now. Otherwise. press **Enter** or wait for the timeout to expire for the installation to begin.

Use **yaboot** with **vmlinuz** and **ramdisk** to boot your system over a network. You cannot use the **ppc64.img** to boot over a network; the file is too large for TFTP.

14.1. THE BOOT MENU

The installer displays the **boot:** prompt. For example:

```
IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM
IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM
IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM
IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM
IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM
IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM
IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM
IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM
IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM
IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM IBM
/
Elapsed time since release of system processors: 276 mins 49 secs

System has 128 Mbytes in RMA
Config file read, 227 bytes


Welcome to the 64-bit Red Hat Enterprise Linux 6.0 installer!
Hit <TAB> for boot options.


Welcome to yaboot version 1.3.14 (Red Hat 1.3.14-35.el6)
Enter "help" to get some basic usage information
boot:
```

To proceed with installation, type **linux** and press **Enter**.

You can also specify boot options at this prompt; refer to Chapter 28, *Boot Options* for more information. For example, to use the installer to rescue a previously installed system, type **linux rescue** and press **Enter**.

The following example shows the **vnc** boot option being passed to begin a graphical installation:

```
boot:
* linux
boot: linux vnc
Please wait, loading kernel...
```

14.2. INSTALLING FROM A DIFFERENT SOURCE

You can install Red Hat Enterprise Linux from the ISO images stored on hard disk, or from a network using NFS, FTP, HTTP, or HTTPS methods. Experienced users frequently use one of these methods because it is often faster to read data from a hard disk or network server than from a DVD.

The following table summarizes the different boot methods and recommended installation methods to use with each:

Table 14.1. Boot methods and installation sources

Boot method	Installation source
Installation DVD	DVD, network, or hard disk
Installation USB flash drive	Installation DVD, network, or hard disk
Minimal boot CD or USB, rescue CD	Network or hard disk

Refer to Section 3.7, "Selecting an Installation Method" for information about installing from locations other than the media with which you booted the system.

14.3. BOOTING FROM THE NETWORK USING A YABOOT INSTALLATION SERVER

To boot with a **yaboot** installation server, you need a properly configured server, and a network interface in your computer that can support an installation server. For information on how to configure an installation server, refer to Chapter 30, *Setting Up an Installation Server*.

Configure the computer to boot from the network interface by selecting **Select Boot Options** in the SMS menu, then **Select Boot/Install Device**. Finally, select your network device from the list of available devices.

Once you properly configure booting from an installation server, the computer can boot the Red Hat Enterprise Linux installation system without any other media.

To boot a computer from a **yaboot** installation server:

1. Ensure 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 computer.

3. A menu screen appears. Press the number key that corresponds to the desired option.

If your PC does not boot from the network installation server, ensure that the SMS is configured to boot first from the correct network interface. Refer to your hardware documentation for more information.

CHAPTER 15. CONFIGURING LANGUAGE AND INSTALLATION SOURCE

Before the graphical installation program starts, you need to configure the language and installation source.

15.1. THE TEXT MODE INSTALLATION PROGRAM USER INTERFACE

IMPORTANT

We recommend that you install Red Hat Enterprise Linux using the graphical interface. If you are installing Red Hat Enterprise Linux on a system that lacks a graphical display, consider performing the installation over a VNC connection – see Chapter 31, *Installing Through VNC*. If **anaconda** detects that you are installing in text mode on a system where installation over a VNC connection might be possible, **anaconda** asks you to verify your decision to install in text mode even though your options during installation are limited.

If your system has a graphical display, but graphical installation fails, try booting with the **xdriver=vesa** option – refer to Chapter 28, *Boot Options*

Both the loader and later **anaconda** use a screen-based interface that includes most of the on-screen *widgets* commonly found on graphical user interfaces. Figure 15.1, "Installation Program Widgets as seen in **URL Setup**", and Figure 15.2, "Installation Program Widgets as seen in **Choose a Language**", illustrate widgets that appear on screens during the installation process.

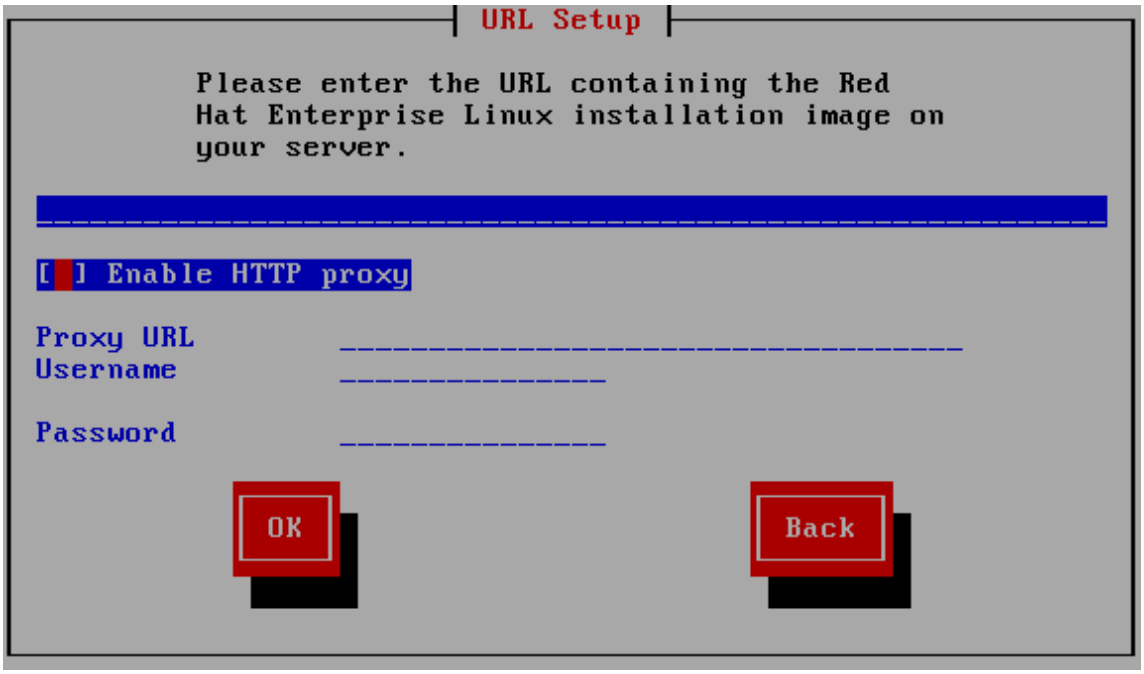


Figure 15.1. Installation Program Widgets as seen in URL Setup

Figure 15.2. Installation Program Widgets as seen in Choose a Language

The widgets include:

- Window — Windows (usually referred to as *dialogs* in this manual) appear on your screen throughout the installation process. At times, one window may overlay another; in these cases, you can only interact with the window on top. When you are finished in that window, it disappears, allowing you to continue working in the window underneath.

- Checkbox — Checkboxes allow you to select or deselect a feature. The box displays either an asterisk (selected) or a space (unselected). When the cursor is within a checkbox, press **Space** to select or deselect a feature.

- Text Input — Text input lines are regions where you can enter information required by the installation program. When the cursor rests on a text input line, you may enter and/or edit information on that line.

- Text Widget — Text widgets are regions of the screen for the display of text. At times, text widgets may also contain other widgets, such as checkboxes. If a text widget contains more information than can be displayed in the space reserved for it, a scroll bar appears; if you position the cursor within the text widget, you can then use the **Up** and **Down** arrow keys to scroll through all the information available. Your current position is shown on the scroll bar by a **#** character, which moves up and down the scroll bar as you scroll.

- Scroll Bar — Scroll bars appear on the side or bottom of a window to control which part of a list or document is currently in the window's frame. The scroll bar makes it easy to move to any part of a file.

- Button Widget — Button widgets are the primary method of interacting with the installation program. You progress through the windows of the installation program by navigating these buttons, using the **Tab** and **Enter** keys. Buttons can be selected when they are highlighted.

- Cursor — Although not a widget, the cursor is used to select (and interact with) a particular

widget. As the cursor is moved from widget to widget, it may cause the widget to change color, or the cursor itself may only appear positioned in or next to the widget. In Figure 15.1, "Installation Program Widgets as seen in **URL Setup**", the cursor is positioned on the **Enable HTTP proxy** checkbox. Figure 8.2, "Installation Program Widgets as seen in **Choose a Language**", shows the cursor on the **OK** button.

15.1.1. Using the Keyboard to Navigate

Navigation through the installation dialogs is performed through a simple set of keystrokes. To move the cursor, use the **Left**, **Right**, **Up**, and **Down** arrow keys. Use **Tab**, and **Shift**-**Tab** to cycle forward or backward through each widget on the screen. Along the bottom, most screens display a summary of available cursor positioning keys.

To "press" a button, position the cursor over the button (using **Tab**, for example) and press **Space** or **Enter**. To select an item from a list of items, move the cursor to the item you wish to select and press **Enter**. To select an item with a checkbox, move the cursor to the checkbox and press **Space** to select an item. To deselect, press **Space** a second time.

Pressing **F12** accepts the current values and proceeds to the next dialog; it is equivalent to pressing the **OK** button.

WARNING

Unless a dialog box is waiting for your input, do not press any keys during the installation process (doing so may result in unpredictable behavior).

15.2. LANGUAGE SELECTION

Use the arrow keys on your keyboard to select a language to use during the installation process (refer to Figure 15.3, "Language Selection"). With your selected language highlighted, press the **Tab** key to move to the **OK** button and press the **Enter** key to confirm your choice.

The language you select here will become the default language for the operating system once it is installed. Selecting the appropriate language also helps target your time zone configuration later in the installation. The installation program tries to define the appropriate time zone based on what you specify on this screen.

To add support for additional languages, customize the installation at the package selection stage. For more information, refer to Section 16.19.2, " Customizing the Software Selection " .

Figure 15.3. Language Selection

Once you select the appropriate language, click **Next** to continue.

15.3. INSTALLATION METHOD

Use the arrow keys on your keyboard to select an installation method (refer to Figure 15.4, "Installation Method"). With your selected method highlighted, press the **Tab** key to move to the **OK** button and press the **Enter** key to confirm your choice.

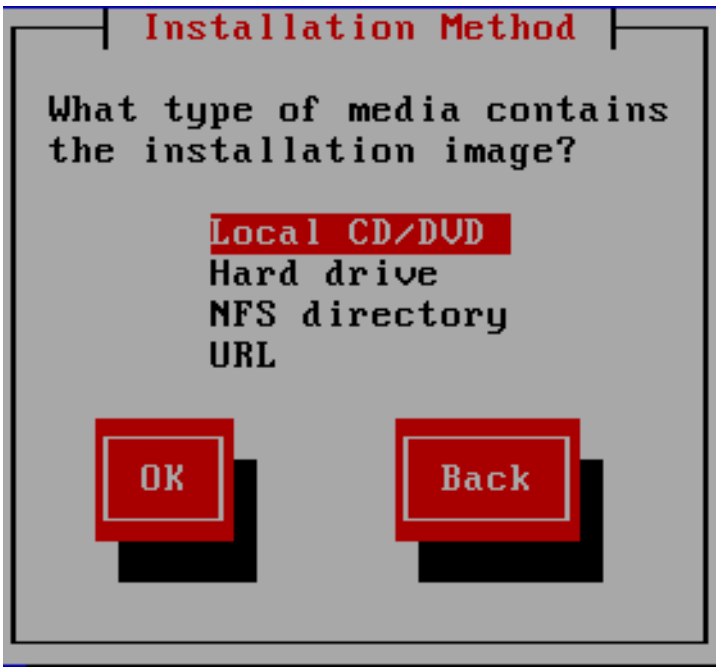


Figure 15.4. Installation Method

15.3.1. Beginning Installation

15.3.1.1. Installing from a DVD

To install Red Hat Enterprise Linux from a DVD, place the DVD your DVD drive and boot your system from the DVD. Even if you booted from alternative media, you can still install Red Hat Enterprise Linux from DVD media.

The installation program then probes your system and attempts to identify your DVD drive. It starts by looking for an IDE (also known as an ATAPI) DVD drive.

If your DVD drive is not detected, and it is a SCSI DVD, the installation program prompts you to choose a SCSI driver. Choose the driver that most closely resembles your adapter. You may specify options for the driver if necessary; however, most drivers detect your SCSI adapter automatically.

If the DVD drive is found and the driver loaded, the installer will present you with the option to perform a media check on the DVD. This will take some time, and you may opt to skip over this step. However, if you later encounter problems with the installer, you should reboot and perform the media check before calling for support. From the media check dialog, continue to the next stage of the installation process (refer to Section 16.5, “Welcome to Red Hat Enterprise Linux”).

15.3.2. Installing from a Hard Drive

The **Select Partition** screen applies only if you are installing from a disk partition (that is, you selected **Hard Drive** in the **Installation Method** dialog). This dialog allows you to name the disk partition and directory from which you are installing Red Hat Enterprise Linux. If you used the **repo=hd** boot option, you already specified a partition.

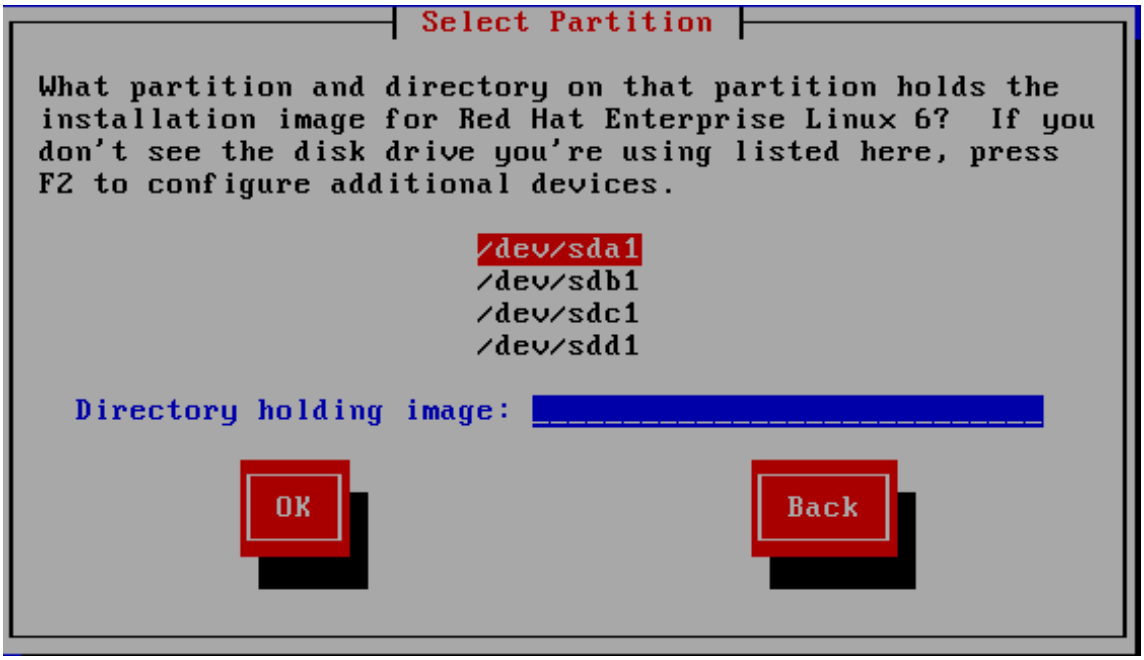


Figure 15.5. Selecting Partition Dialog for Hard Drive Installation

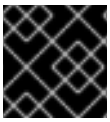
Select the partition containing the ISO files from the list of available partitions. Internal IDE, SATA, SCSI, and USB drive device names begin with **/dev/sd**. Each individual drive has its own letter, for example **/dev/sda**. Each partition on a drive is numbered, for example **/dev/sda1**.

Also specify the **Directory holding images**. Enter the full directory path from the drive that contains the ISO image files. The following table shows some examples of how to enter this information:

Table 15.1. Location of ISO images for different partition types

Partition type	Volume	Original path to files	Directory to use
VFAT	D:\	D:\Downloads\RHEL6.9	/Downloads/RHEL6.9
ext2, ext3, ext4	/home	/home/user1/RHEL6.9	/user1/RHEL6.9

If the ISO images are in the root (top-level) directory of a partition, enter a /. If the ISO images are located in a subdirectory of a mounted partition, enter the name of the directory holding the ISO images within that partition. For example, if the partition on which the ISO images is normally mounted as **/home/**, and the images are in **/home/new/**, you would enter **/new/**.



IMPORTANT

An entry without a leading slash may cause the installation to fail.

Select **OK** to continue. Proceed with Chapter 16, *Installing Using Anaconda*.

15.3.3. Performing a Network Installation

When you start an installation with the **askmethod** or **repo=** options, you can install Red Hat Enterprise Linux from a network server using FTP, HTTP, HTTPS, or NFS protocols. **Anaconda** uses the same network connection to consult additional software repositories later in the installation process.

If your system has more than one network device, **anaconda** presents you with a list of all available devices and prompts you to select one to use during installation. If your system only has a single network device, **anaconda** automatically selects it and does not present this dialog.

Figure 15.6. Networking Device

If you are not sure which device in the list corresponds to which physical socket on the system, select a device in the list then press the **Identify** button. The **Identify NIC** dialog appears.

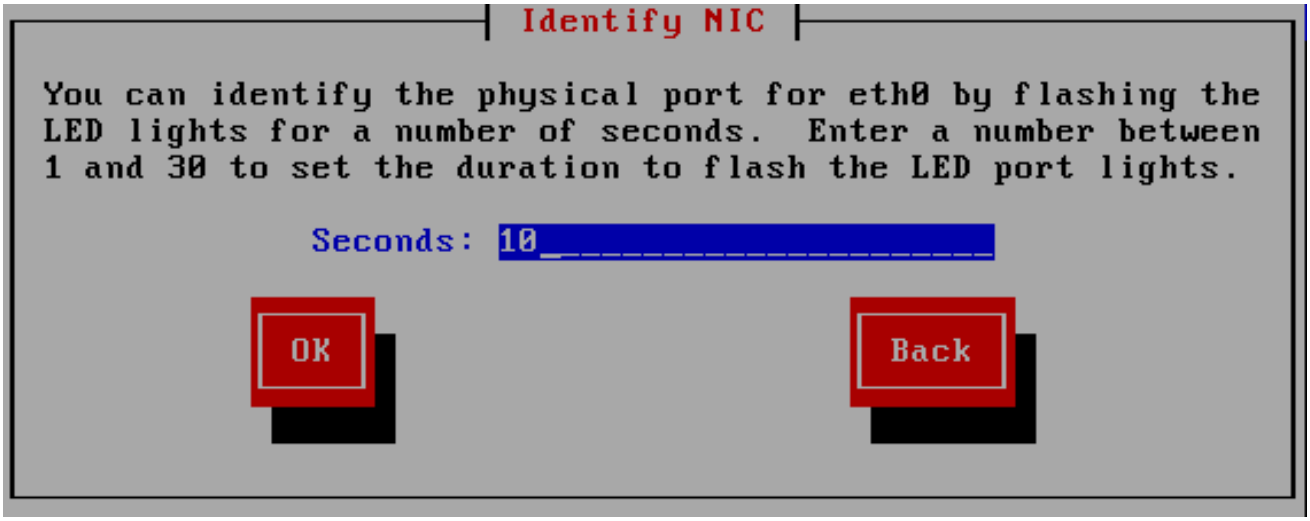


Figure 15.7. Identify NIC

The sockets of most network devices feature an *activity light* (also called a *link light*) — an LED that flashes to indicate that data is flowing through the socket. **Anaconda** can flash the activity light of the network device that you selected in the **Networking Device** dialog for up to 30 seconds. Enter the number of seconds that you require, then press **OK**. When **anaconda** finishes flashing the light, it returns you to the **Networking Device** dialog.

When you select a network device, **anaconda** prompts you to choose how to configure TCP/IP:

IPv4 options

Dynamic IP configuration (DHCP)

Anaconda uses DHCP running on the network to supply the network configuration automatically.

Manual configuration

Anaconda prompts you to enter the network configuration manually, including the IP address for this system, the netmask, the gateway address, and the DNS address.

IPv6 options

Automatic

Anaconda uses *router advertisement* (RA) and DHCP for automatic configuration, based on the network environment. (Equivalent to the **Automatic** option in **NetworkManager**)

Automatic, DHCP only

Anaconda does not use RA, but requests information from DHCPv6 directly to create a stateful configuration. (Equivalent to the **Automatic, DHCP only** option in **NetworkManager**)

Manual configuration

Anaconda prompts you to enter the network configuration manually, including the IP address for this system, the netmask, the gateway address, and the DNS address.

Anaconda supports the IPv4 and IPv6 protocols. However, if you configure an interface to use both IPv4 and IPv6, the IPv4 connection must succeed or the interface will not work, even if the IPv6 connection succeeds.

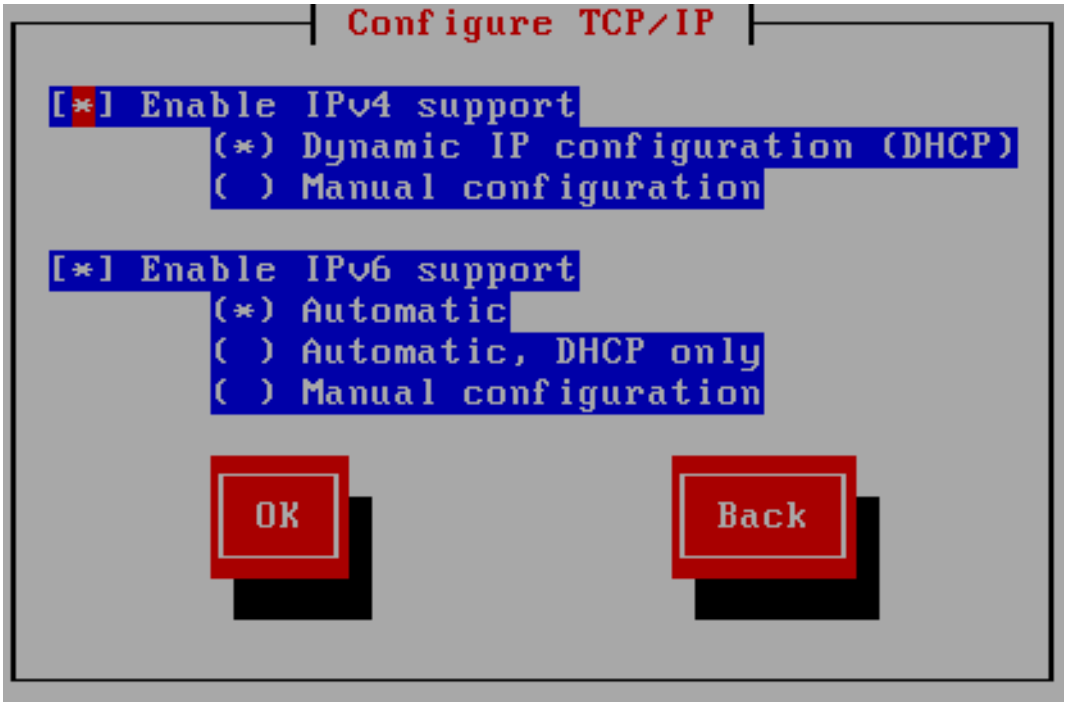


Figure 15.8. Configure TCP/IP

By default, **anaconda** uses DHCP to provide network settings automatically for IPv4 and automatic configuration to provide network settings for IPv6. If you choose to configure TCP/IP manually, **anaconda** prompts you to provide the details in the **Manual TCP/IP Configuration** dialog:

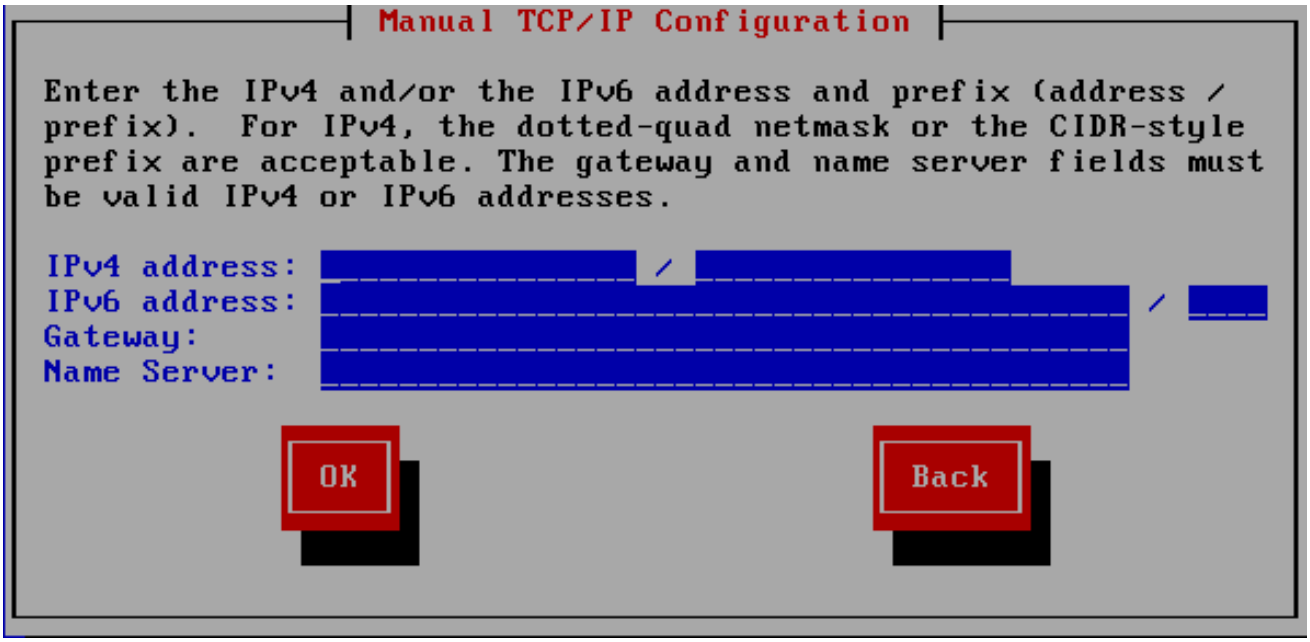


Figure 15.9. Manual TCP/IP Configuration

The dialog provides fields for IPv4 and IPv6 addresses and prefixes, depending on the protocols that you chose to configure manually, together with fields for the network gateway and name server. Enter the details for your network, then press **OK**.

When the installation process completes, it will transfer these settings to your system.

- If you are installing via NFS, proceed to Section 15.3.4, “Installing via NFS” .

- If you are installing via Web or FTP, proceed to Section 15.3.5, "Installing via FTP, HTTP, or HTTPS".

15.3.4. Installing via NFS

The NFS dialog applies only if you selected **NFS Image** in the **Installation Method** dialog. If you used the **repo=nfs** boot option, you already specified a server and path.

Figure 15.10. NFS Setup Dialog

1. Enter the domain name or IP address of your NFS server in the **NFS server name** field. For example, if you are installing from a host named **eastcoast** in the domain **example.com**, enter **eastcoast.example.com**.

2. Enter the name of the exported directory in the **Red Hat Enterprise Linux 6.9 directory** field:

   - If the NFS server is exporting a mirror of the Red Hat Enterprise Linux installation tree, enter the directory which contains the root of the installation tree. If everything was specified properly, a message appears indicating that the installation program for Red Hat Enterprise Linux is running.

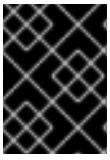
   - If the NFS server is exporting the ISO image of the Red Hat Enterprise Linux DVD, enter the directory which contains the ISO image.

   If you followed the setup described in Section 12.1.2, "Preparing for an NFS Installation" , the exported directory is the one that you specified as ***publicly_available_directory***.

3. Specify any NFS mount options that you require in the **NFS mount options** field. Refer to the man pages for **mount** and **nfs** for a comprehensive list of options. If you do not require any mount options, leave the field empty.

4. Proceed with Chapter 16, *Installing Using Anaconda*.

15.3.5. Installing via FTP, HTTP, or HTTPS



IMPORTANT

When you provide a URL to an installation source, you must explicitly specify **http://** or **https://** or **ftp://** as the protocol.

The URL dialog applies only if you are installing from a FTP, HTTP, or HTTPS server (if you selected **URL** in the **Installation Method** dialog). This dialog prompts you for information about the FTP, HTTP, or

HTTPS server from which you are installing Red Hat Enterprise Linux. If you used the **repo=ftp** or **repo=http** boot options, you already specified a server and path.

Enter the name or IP address of the FTP, HTTP, or HTTPS site from which you are installing, and the name of the directory that contains the **/images** directory for your architecture. For example:

/mirrors/redhat/rhel-6.9/Server/ppc64/

To install via a secure HTTPS connection, specify **https://** as the protocol.

Specify the address of a proxy server, and if necessary, provide a port number, username, and password. If everything was specified properly, a message box appears indicating that files are being retrieved from the server.

If your FTP, HTTP, or HTTPS server requires user authentication, specify user and password as part of the URL as follows:

{ftp|http|https}://<user>:<password>@<hostname>[:<port>]/<directory>/

For example:

http://install:rhel6.9pw@name.example.com/mirrors/redhat/rhel-6.9/Server/ppc64/

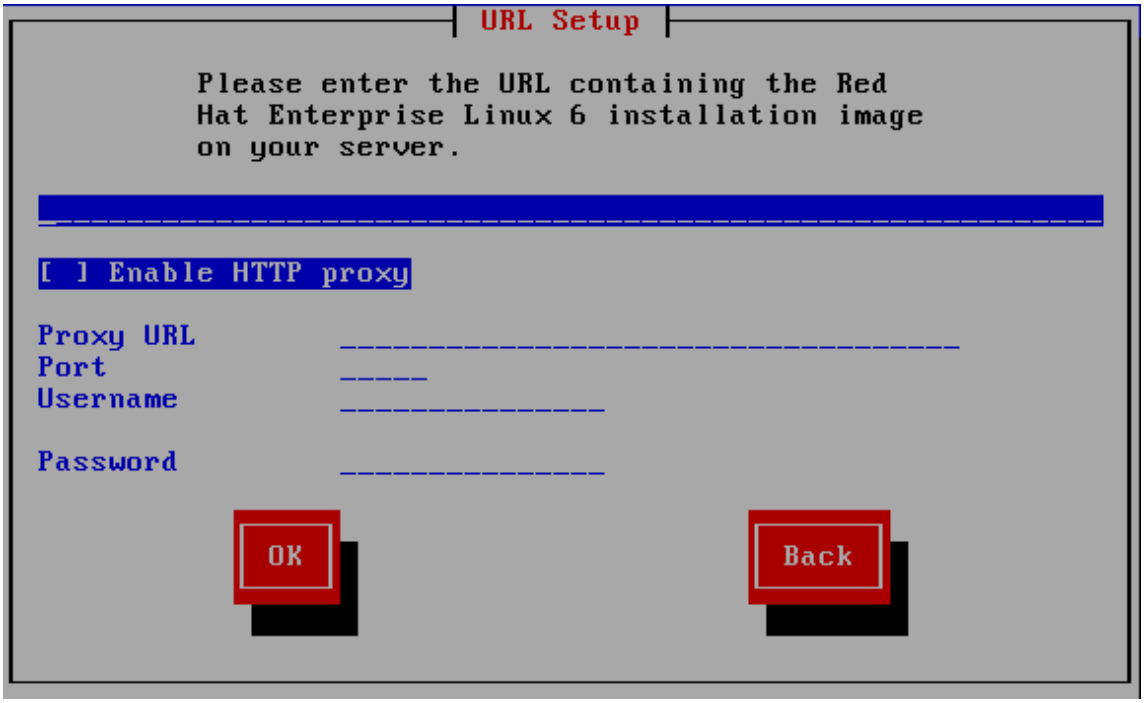


Figure 15.11. URL Setup Dialog

Proceed with Chapter 16, *Installing Using Anaconda*.

15.4. VERIFYING MEDIA

The DVD offers an option to verify the integrity of the media. Recording errors sometimes occur while producing DVD media. An error in the data for package chosen in the installation program can cause the installation to abort. To minimize the chances of data errors affecting the installation, verify the media before installing.

If the verification succeeds, the installation process proceeds normally. If the process fails, create a new DVD using the ISO image you downloaded earlier.

CHAPTER 16. INSTALLING USING ANACONDA

This chapter describes an installation using the graphical user interface of **anaconda**.

16.1. THE TEXT MODE INSTALLATION PROGRAM USER INTERFACE

While text mode installations are not explicitly documented, those using the text mode installation program can easily follow the GUI installation instructions. However, because text mode presents you with a simpler, more streamlined installation process, certain options that are available in graphical mode are not also available in text mode. These differences are noted in the description of the installation process in this guide, and include:

- configuring advanced storage methods such as LVM, RAID, FCoE, zFCP, and iSCSI.
- customizing the partition layout
- customizing the bootloader layout
- selecting packages during installation
- configuring the installed system with **firstboot**

16.2. THE GRAPHICAL INSTALLATION PROGRAM USER INTERFACE

If you have used a *graphical user interface (GUI)* before, you are already familiar with this process; use your mouse to navigate the screens, click buttons, or enter text fields.

You can also navigate through the installation using the keyboard. The **Tab** key allows you to move around the screen, the Up and Down arrow keys to scroll through lists, **+** and **-** keys expand and collapse lists, while **Space** and **Enter** selects or removes from selection a highlighted item. You can also use the **Alt**+***X*** key command combination as a way of clicking on buttons or making other screen selections, where ***X*** is replaced with any underlined letter appearing within that screen.

If you would like to use a graphical installation with a system that does not have that capability, such as a partitioned system, you can use VNC or display forwarding. Both the VNC and display forwarding options require an active network during the installation and the use of boot time arguments. For more information on available boot time options, refer to Chapter 28, *Boot Options*

NOTE

If you do not wish to use the GUI installation program, the text mode installation program is also available. To start the text mode installation program, use the following command at the **yaboot:** prompt:

```
linux text
```

Refer to Section 14.1, "The Boot Menu" for a description of the Red Hat Enterprise Linux boot menu and to Section 15.1, "The Text Mode Installation Program User Interface" for a brief overview of text mode installation instructions.

It is highly recommended that installs be performed using the GUI installation program. The GUI installation program offers the full functionality of the Red Hat Enterprise Linux installation program, including LVM configuration which is not available during a text mode installation.

Users who must use the text mode installation program can follow the GUI installation instructions and obtain all needed information.

16.3. A NOTE ABOUT LINUX VIRTUAL CONSOLES

This information only applies to users of non-partitioned System p systems using a video card as their console. Users of partitioned System p systems should skip to Section 16.4, "Using the HMC vterm".

The Red Hat Enterprise Linux installation program offers more than the dialog boxes of the installation process. Several kinds of diagnostic messages are available to you, as well as a way to enter commands from a shell prompt. The installation program displays these messages on five *virtual consoles*, among which you can switch using a single keystroke combination.

A virtual console is a shell prompt in a non-graphical environment, accessed from the physical machine, not remotely. Multiple virtual consoles can be accessed simultaneously.

These virtual consoles can be helpful if you encounter a problem while installing Red Hat Enterprise Linux. Messages displayed on the installation or system consoles can help pinpoint a problem. Refer to Table 16.1, "Console, Keystrokes, and Contents" for a listing of the virtual consoles, keystrokes used to switch to them, and their contents.

Generally, there is no reason to leave the default console (virtual console #6) for graphical installations unless you are attempting to diagnose installation problems.

Table 16.1. Console, Keystrokes, and Contents

console	keystrokes	contents
1	**ctrl+alt+f1**	installation dialog
2	**ctrl+alt+f2**	shell prompt
3	**ctrl+alt+f3**	install log (messages from installation program)
4	**ctrl+alt+f4**	system-related messages

console	keystrokes	contents
5	**ctrl+alt+f5**	other messages
6	**ctrl+alt+f6**	x graphical display

16.4. USING THE HMC VTERM

The HMC vterm is the console for any partitioned IBM System p. This is opened by right clicking on the partition on the HMC, and then selecting **Open Terminal Window**. Only a single vterm can be connected to the console at one time and there is no console access for partitioned system besides the vterm. This often is referred to as a 'virtual console', but is different from the virtual consoles in Section 16.3, “A Note About Linux Virtual Consoles” .

16.5. WELCOME TO RED HAT ENTERPRISE LINUX

The **Welcome** screen does not prompt you for any input.

Figure 16.1. The Welcome screen

Click on the **Next** button to continue.

16.6. LANGUAGE SELECTION

Using your mouse, select the language (for example, U.S. English) you would prefer to use for the installation and as the system default (refer to the figure below).

Once you have made your selection, click **Next** to continue.

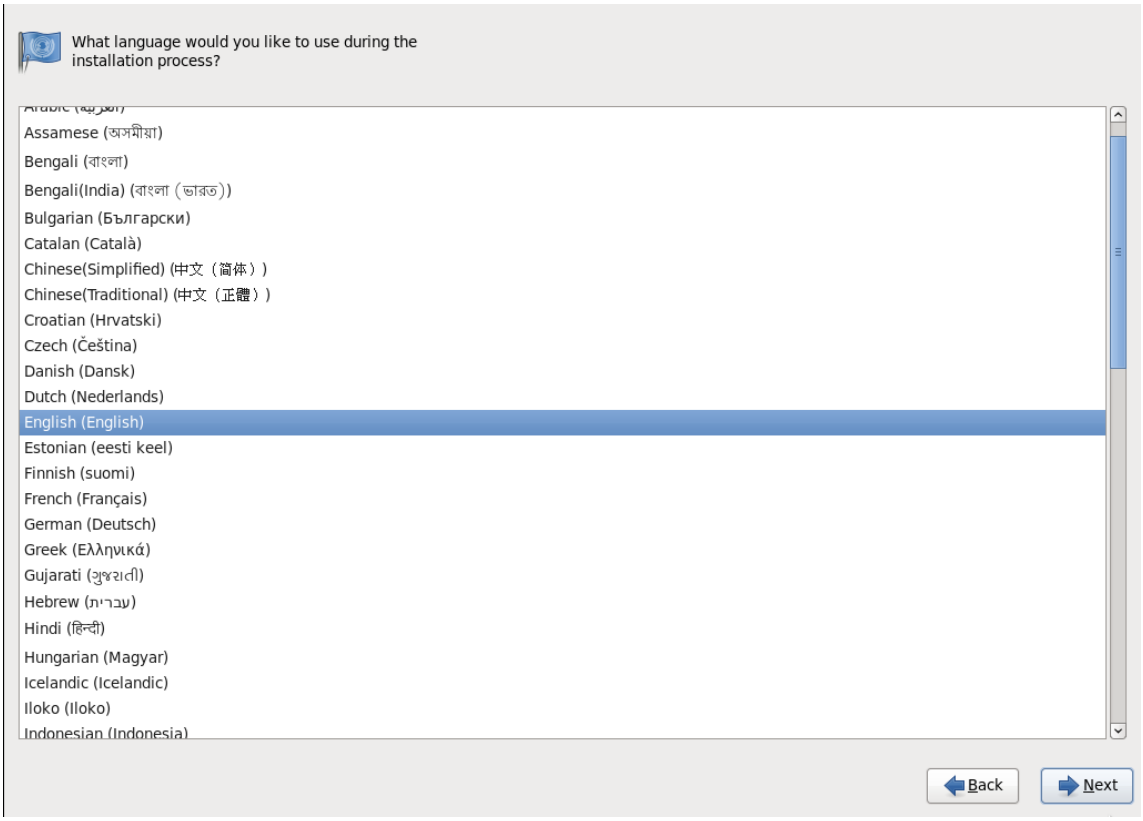


Figure 16.2. Language Configuration

16.7. KEYBOARD CONFIGURATION

Using your mouse, select the correct layout type (for example, U.S. English) for the keyboard you would prefer to use for the installation and as the system default (refer to Figure 16.3, "Keyboard Configuration").

Once you have made your selection, click **Next** to continue.

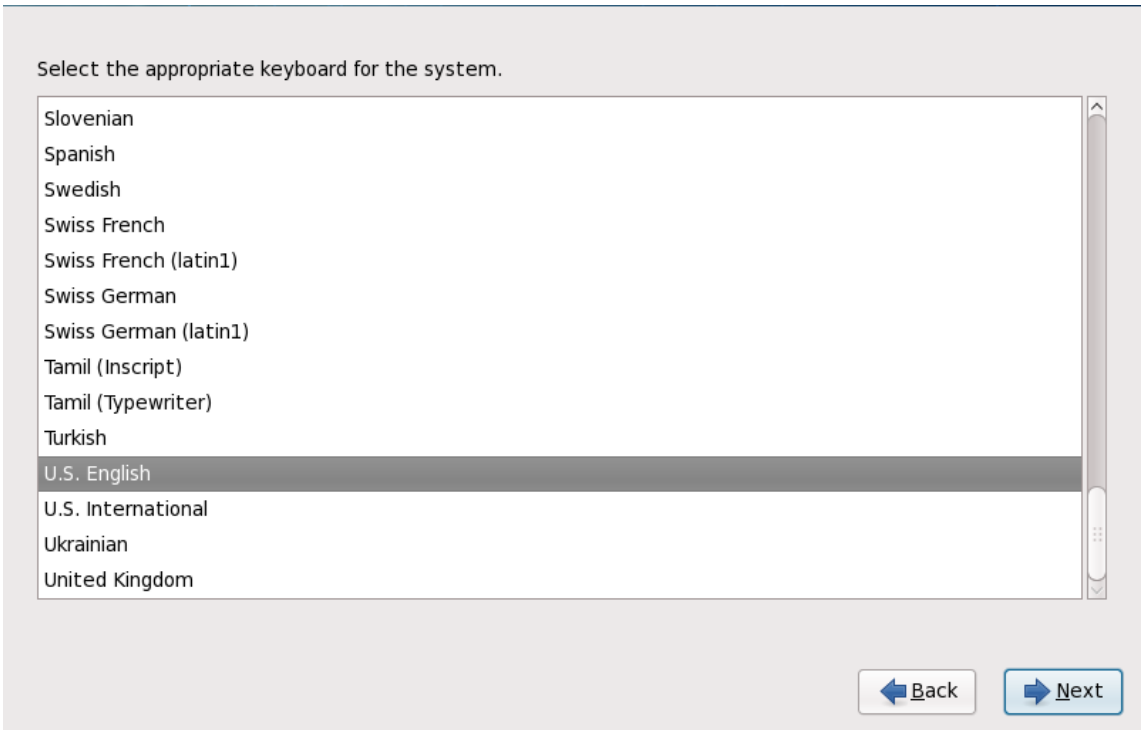
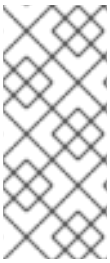


Figure 16.3. Keyboard Configuration



NOTE

To change your keyboard layout type after you have completed the installation, use the **Keyboard Configuration Tool**.

Type the **system-config-keyboard** command in a shell prompt to launch the **Keyboard Configuration Tool**. If you are not root, it prompts you for the root password to continue.

16.8. STORAGE DEVICES

You can install Red Hat Enterprise Linux on a large variety of storage devices. This screen allows you to select either basic or specialized storage devices.

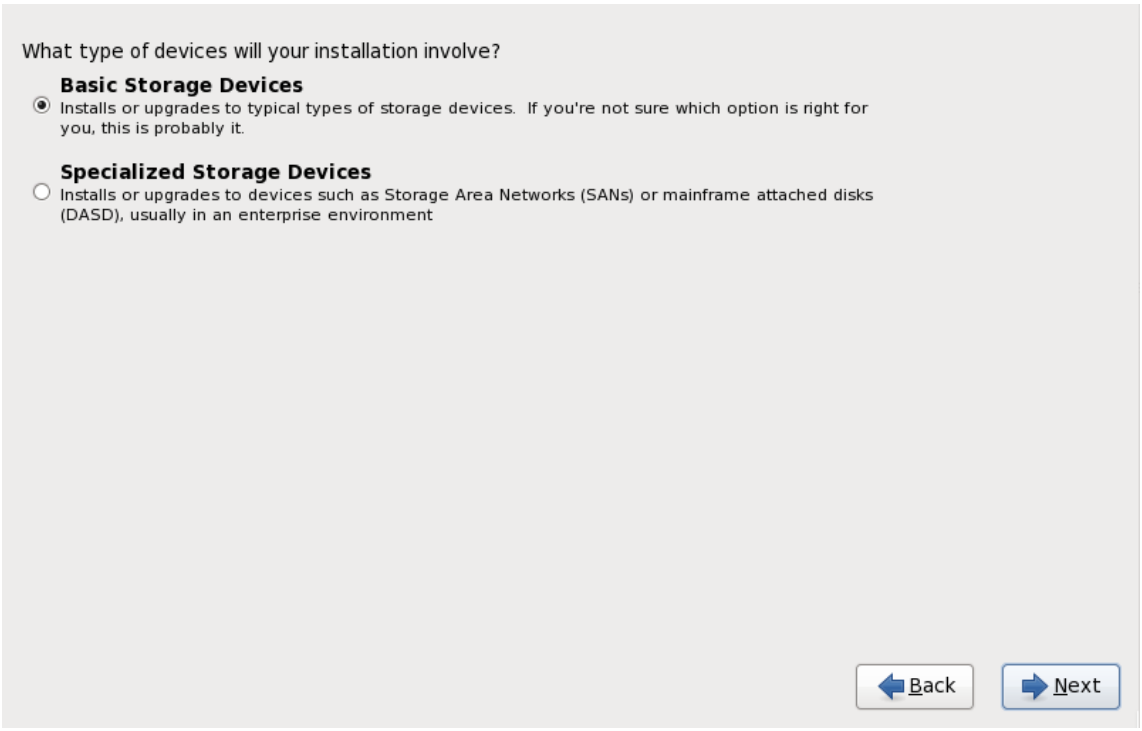


Figure 16.4. Storage devices

Basic Storage Devices

Select **Basic Storage Devices**to install Red Hat Enterprise Linux on the following storage devices:

- hard drives or solid-state drives connected directly to the local system.

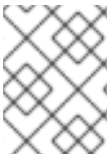
Specialized Storage Devices

Select **Specialized Storage Devices**to install Red Hat Enterprise Linux on the following storage devices:

- *Storage area networks* (SANs)
- *Direct access storage devices* (DASDs)
- Firmware RAID devices
- Multipath devices

Use the **Specialized Storage Devices**option to configure *Internet Small Computer System Interface* (iSCSI) and *FCoE* (Fiber Channel over Ethernet) connections.

If you select **Basic Storage Devices**, **anaconda** automatically detects the local storage attached to the system and does not require further input from you. Proceed to Section 16.9, "Setting the Hostname".



NOTE

Monitoring of LVM and software RAID devices by the **mdeventd** daemon is not performed during installation.

16.8.1. The Storage Devices Selection Screen

The storage devices selection screen displays all storage devices to which **anaconda** has access.

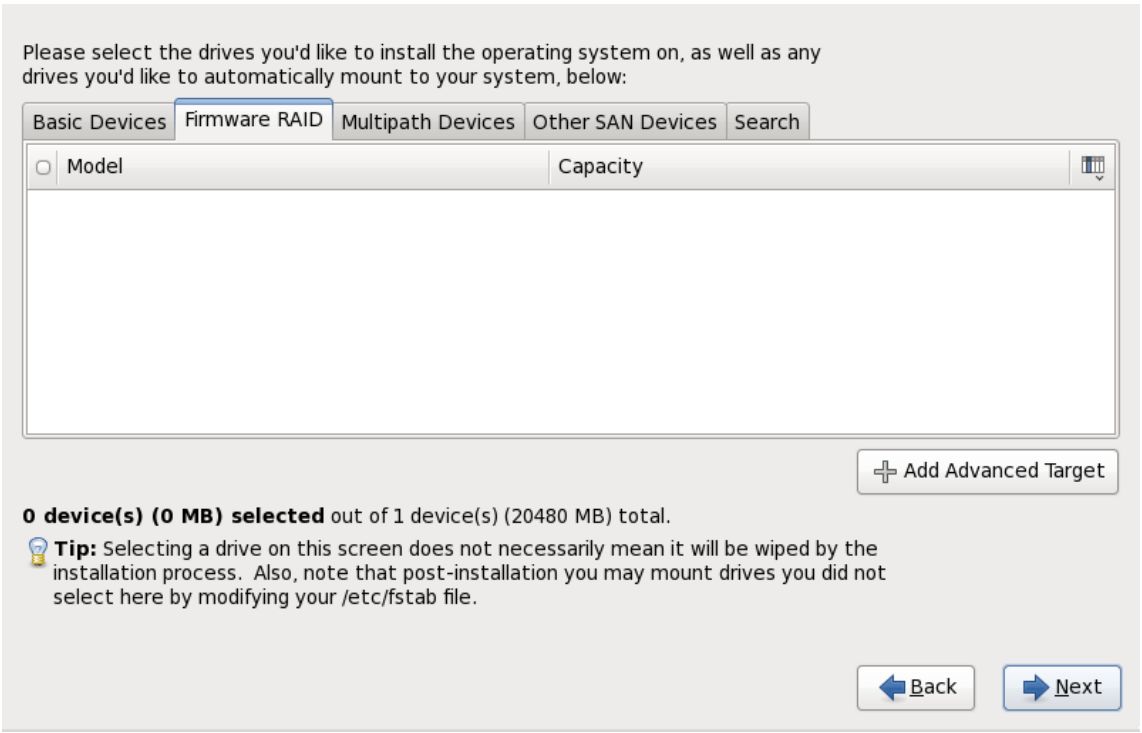


Figure 16.5. Select storage devices — Basic devices

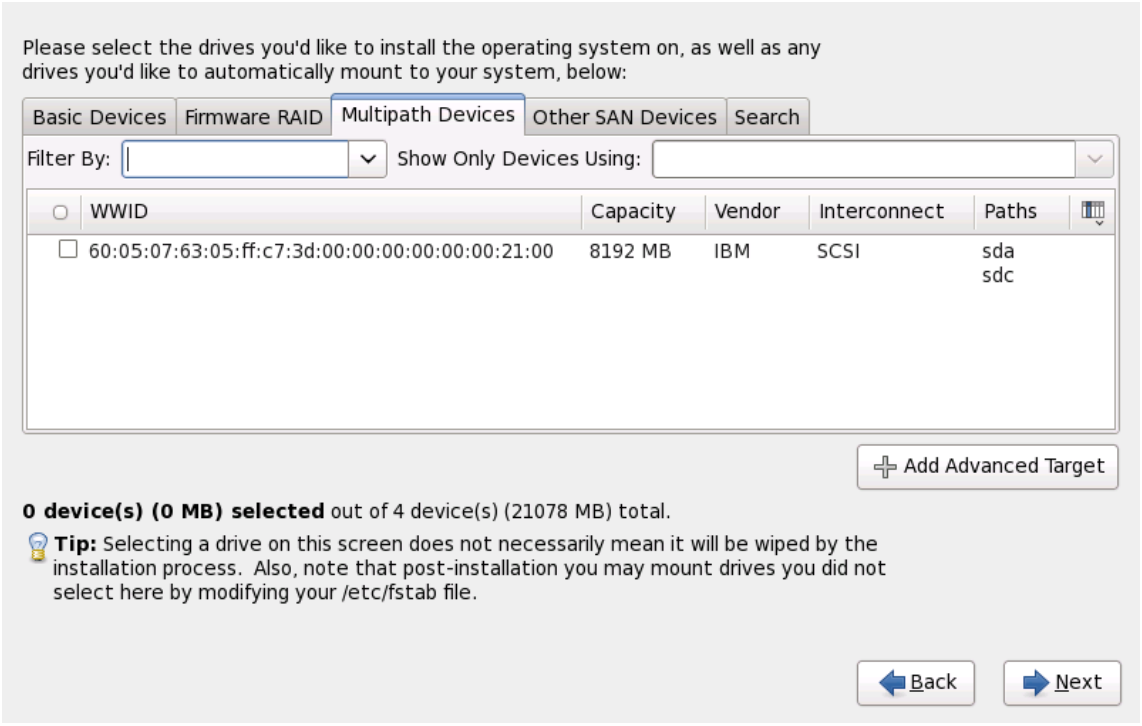


Figure 16.6. Select storage devices — Multipath Devices

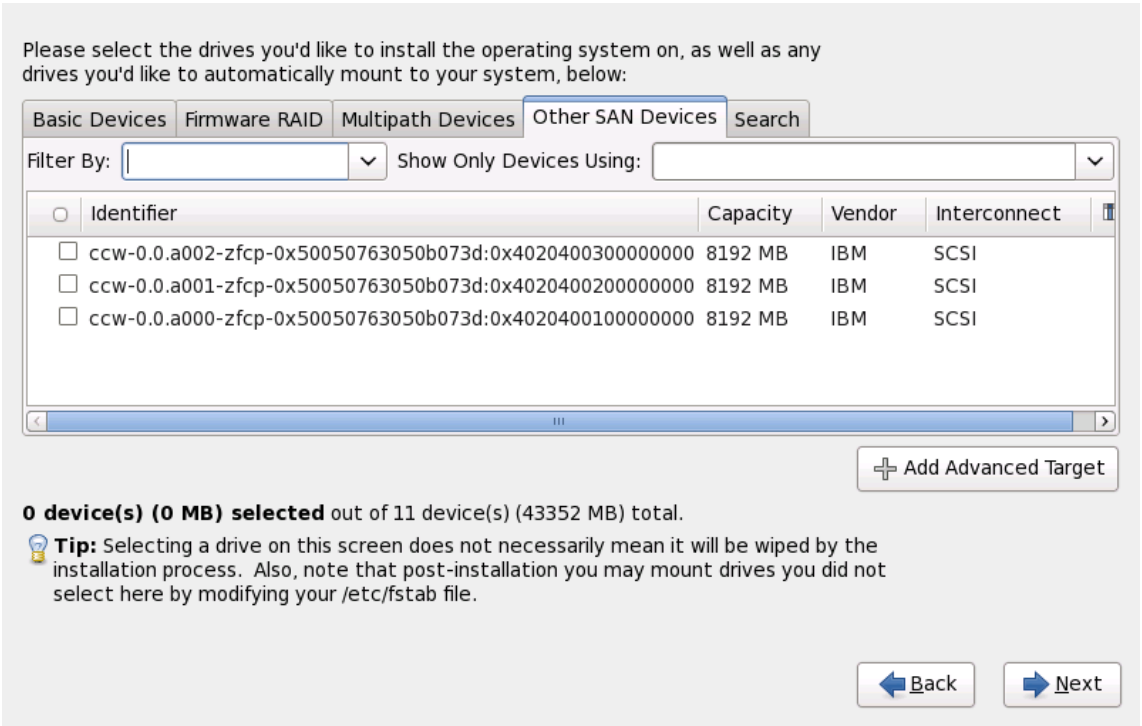


Figure 16.7. Select storage devices — Other SAN Devices

Devices are grouped under the following tabs:

Basic Devices

Basic storage devices directly connected to the local system, such as hard disk drives and solid-state drives.

Firmware RAID

Storage devices attached to a firmware RAID controller.

Multipath Devices

Storage devices accessible through more than one path, such as through multiple SCSI controllers or Fiber Channel ports on the same system.

IMPORTANT

The installer only detects multipath storage devices with serial numbers that are 16 or 32 characters in length.

Other SAN Devices

Any other devices available on a storage area network (SAN).

If you do need to configure iSCSI or FCoE storage, click **Add Advanced Target** and refer to Section 16.8.1.1, " Advanced Storage Options ".

The storage devices selection screen also contains a **Search** tab that allows you to filter storage devices either by their *World Wide Identifier* (WWID) or by the port, target, or *logical unit number* (LUN) at which they are accessed.

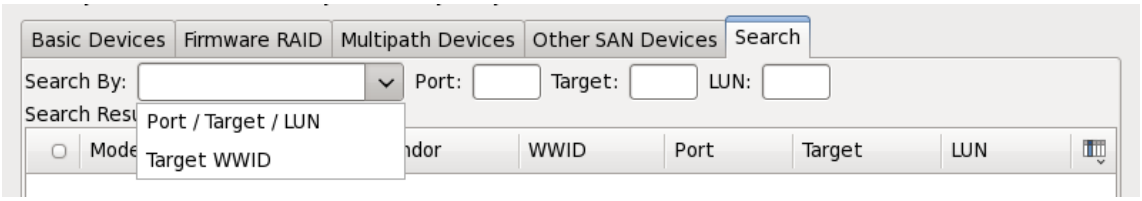


Figure 16.8. The Storage Devices Search Tab

The tab contains a drop-down menu to select searching by port, target, WWID, or LUN (with corresponding text boxes for these values). Searching by WWID or LUN requires additional values in the corresponding text box.

Each tab presents a list of devices detected by **anaconda**, with information about the device to help you to identify it. A small drop-down menu marked with an icon is located to the right of the column headings. This menu allows you to select the types of data presented on each device. For example, the menu on the **Multipath Devices** tab allows you to specify any of **WWID**, **Capacity**, **Vendor**, **Interconnect**, and **Paths** to include among the details presented for each device. Reducing or expanding the amount of information presented might help you to identify particular devices.

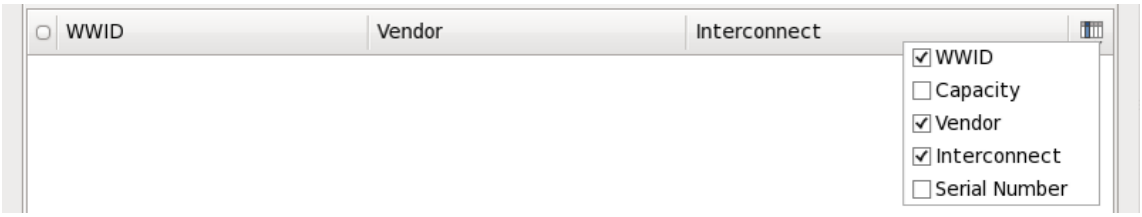
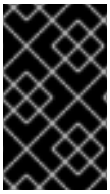


Figure 16.9. Selecting Columns

Each device is presented on a separate row, with a checkbox to its left. Click the checkbox to make a device available during the installation process, or click the *radio button* at the left of the column headings to select or deselect all the devices listed in a particular screen. Later in the installation process, you can choose to install Red Hat Enterprise Linux onto any of the devices selected here, and can choose to automatically mount any of the other devices selected here as part of the installed system.

Note that the devices that you select here are not automatically erased by the installation process. Selecting a device on this screen does not, in itself, place data stored on the device at risk. Note also that any devices that you do not select here to form part of the installed system can be added to the system after installation by modifying the **/etc/fstab** file.



IMPORTANT

Any storage devices that you do not select on this screen are hidden from **anaconda** entirely. To *chain load* the Red Hat Enterprise Linux boot loader from a different boot loader, select all the devices presented in this screen.

when you have selected the storage devices to make available during installation, click **Next** and proceed to Section 16.13, "Initializing the Hard Disk"

16.8.1.1. Advanced Storage Options

From this screen you can configure an *iSCSI* (SCSI over TCP/IP) target or *FCoE* (Fibre channel over ethernet) *SAN* (storage area network). Refer to Appendix B, *iSCSI Disks* for an introduction to iSCSI.

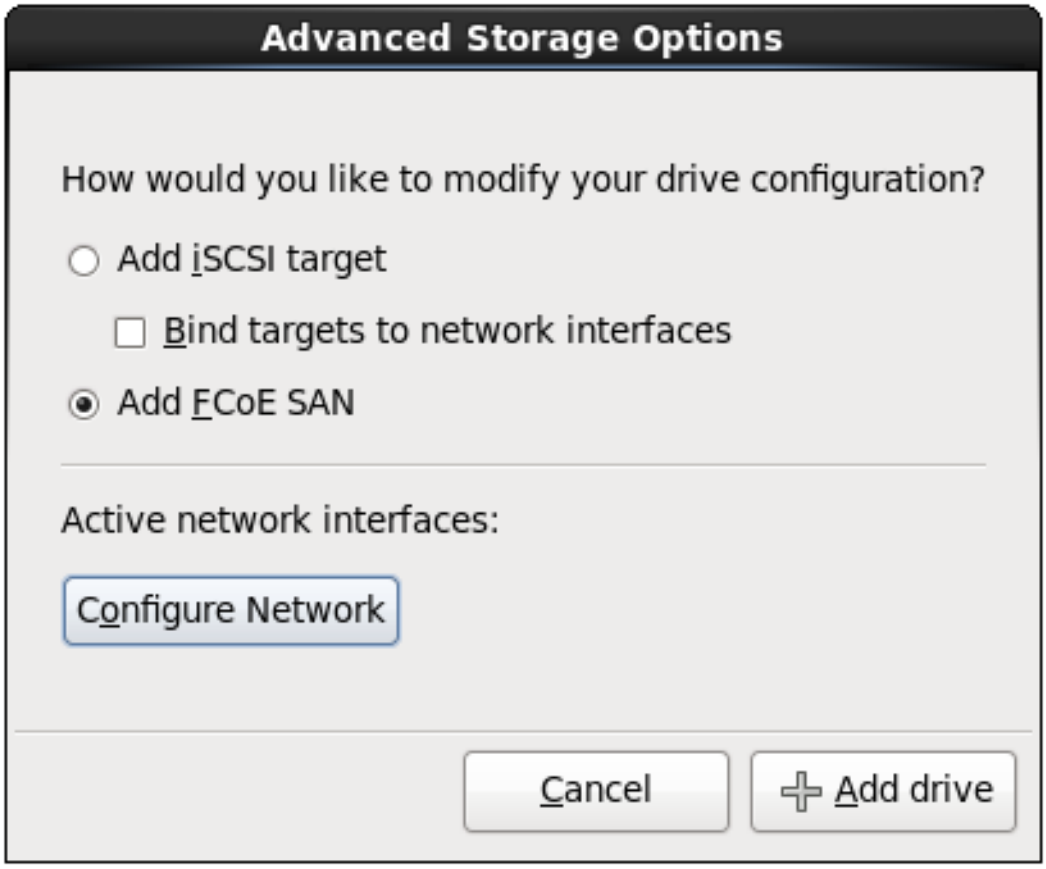


Figure 16.10. Advanced Storage Options

Select **Add iSCSI target** or **Add FCoE SAN** and click **Add drive**. If adding an iSCSI target, optionally check the box labeled **Bind targets to network interfaces**.

16.8.1.1.1. Select and configure a network interface

The **Advanced Storage Options** screen lists the active network interfaces **anaconda** has found on your system. If none are found, **anaconda** must activate an interface through which to connect to the storage devices.

Click **Configure Network** on the **Advanced Storage Options** screen to configure and activate one using **NetworkManager** to use during installation. Alternatively, **anaconda** will prompt you with the **Select network interface** dialog after you click **Add drive**.

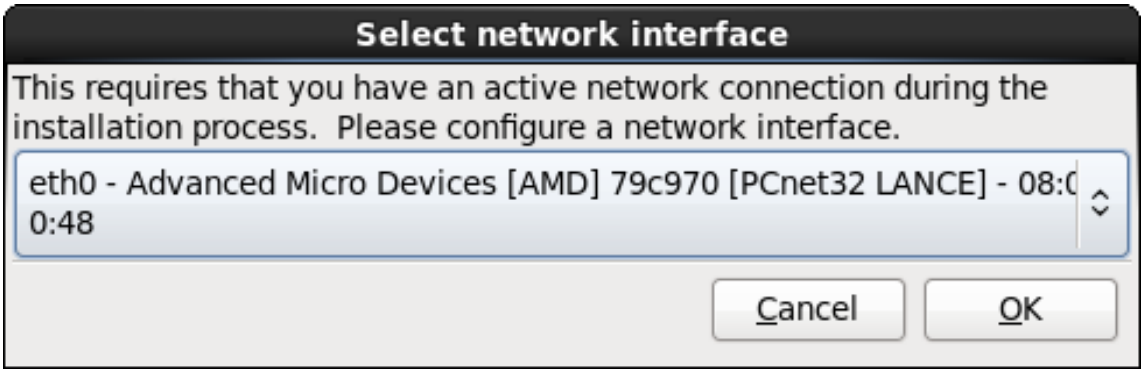


Figure 16.11. Select network interface

1. Select an interface from the drop-down menu.
2. Click **OK**.

Anaconda then starts **NetworkManager** to allow you to configure the interface.

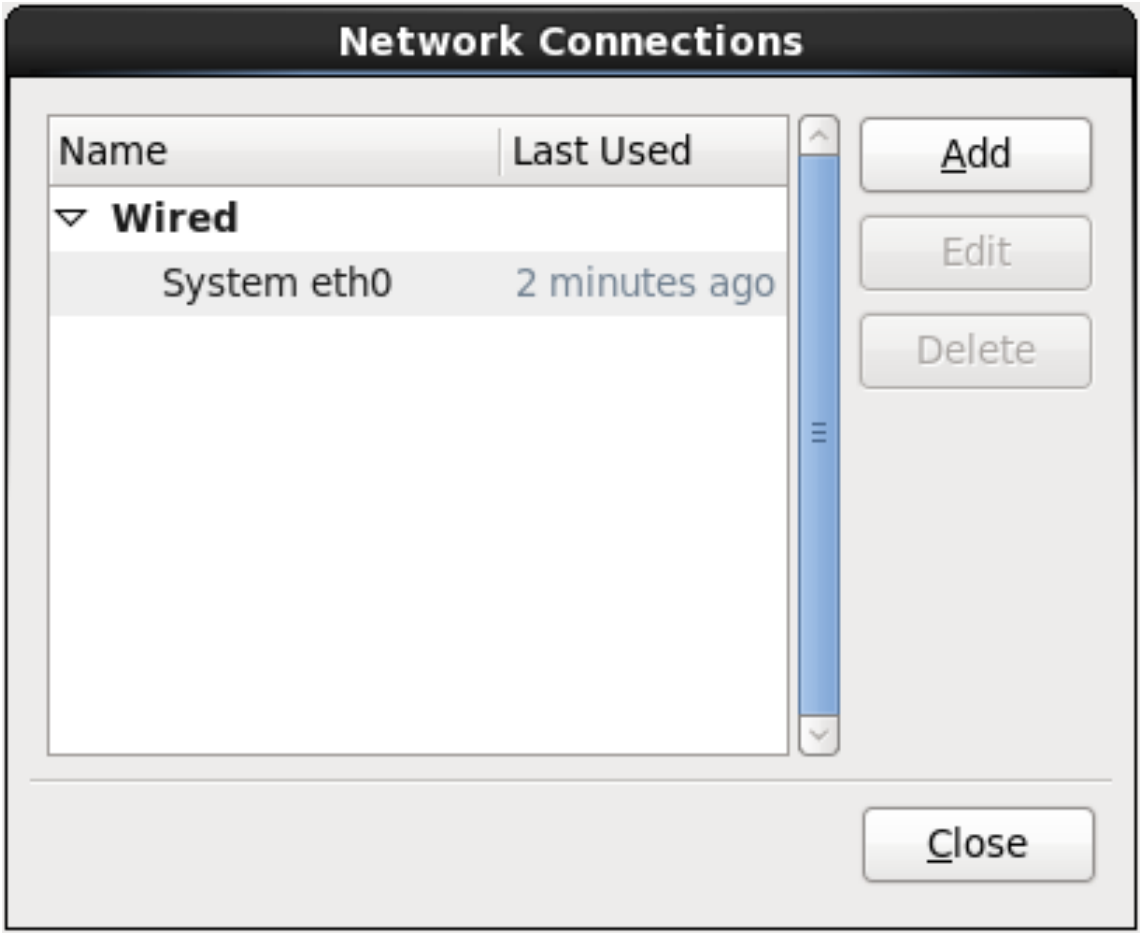


Figure 16.12. Network Connections

For details of how to use **NetworkManager**, refer to Section 16.9, "Setting the Hostname"

16.8.1.1.2. Configure iSCSI parameters

To add an iSCSI target, select **Add iSCSI target** and click **Add drive**.

To use iSCSI storage devices for the installation, **anaconda** 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 username and password for *CHAP* (Challenge Handshake Authentication Protocol) 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 username and password are different for CHAP authentication and reverse CHAP authentication.

Repeat the iSCSI discovery and iSCSI login steps as many times as necessary to add all required iSCSI storage. However, 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.

Procedure 16.1. iSCSI discovery

Use the **iSCSI Discovery Details** dialog to provide **anaconda** with the information that it needs to discover the iSCSI target.

Figure 16.13. The iSCSI Discovery Details dialog

1. Enter the IP address of the iSCSI target in the **Target IP Address** field.
2. Provide a name in the **iSCSI Initiator Name** field for the iSCSI initiator in *iSCSI qualified name* (IQN) format.

   A valid IQN contains:

   - 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 therefore resembles: **iqn.2010-09.storage.example.com:diskarrays-sn-a8675309**, and **anaconda** pre-populates the **iSCSI Initiator Name** field with a name in this format to help you with the structure.

For more information on IQNs, refer to *3.2.6. iSCSI Names* in *RFC 3720 - Internet Small Computer Systems Interface (iSCSI)* available from http://tools.ietf.org/html/rfc3720#section-3.2.6 and *1. iSCSI Names and Addresses* in *RFC 3721 - Internet Small Computer Systems Interface (iSCSI) Naming and Discovery* available from http://tools.ietf.org/html/rfc3721#section-1.

3. Use the drop-down menu to specify the type of authentication to use for iSCSI discovery:

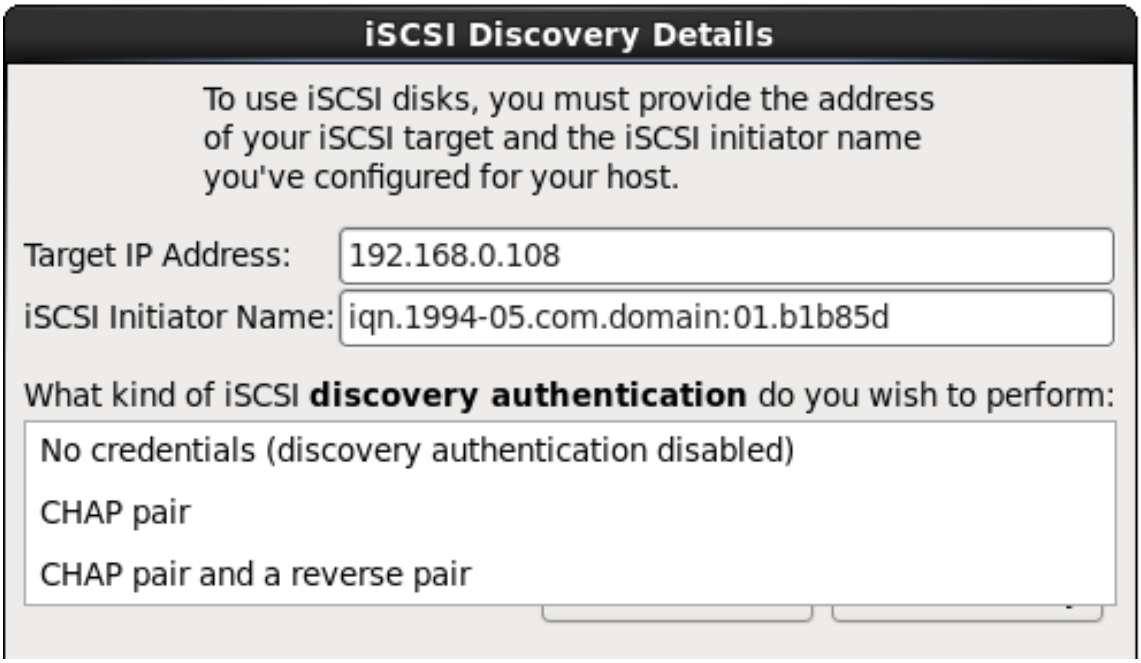


Figure 16.14. iSCSI discovery authentication

- **no credentials**

- **CHAP pair**

- **CHAP pair and a reverse pair**

4. - If you selected **CHAP pair** as the authentication type, provide the username and password for the iSCSI target in the **CHAP Username** and **CHAP Password** fields.

Figure 16.15. CHAP pair

- If you selected **CHAP pair and a reverse pair** as the authentication type, provide the username and password for the iSCSI target in the **CHAP Username** and **CHAP Password** field and the username and password for the iSCSI initiator in the **Reverse CHAP Username** and **Reverse CHAP Password** fields.

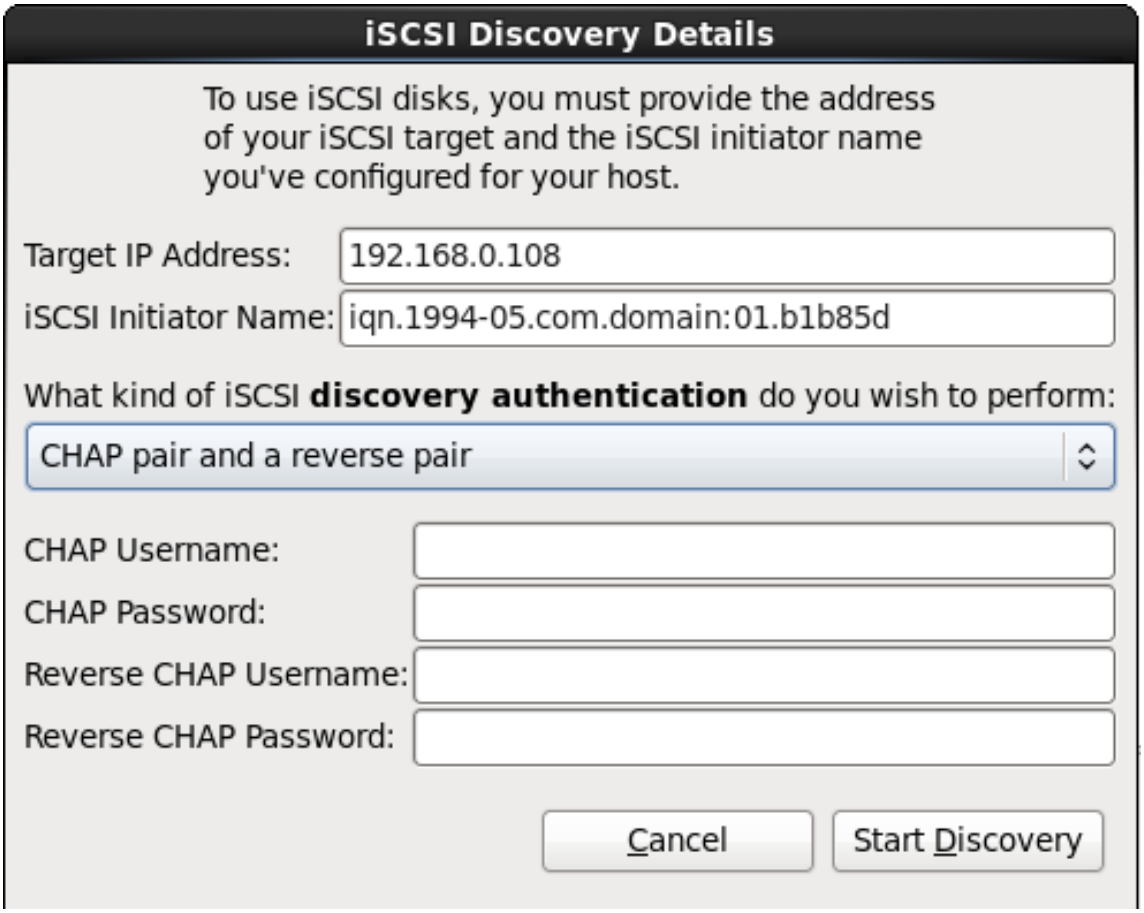


Figure 16.16. CHAP pair and a reverse pair

5. Click **Start Discovery**. **Anaconda** attempts to discover an iSCSI target based on the information that you provided. If discovery succeeds, the **iSCSI Discovered Nodes** dialog presents you with a list of all the iSCSI nodes discovered on the target.

6. Each node is presented with a checkbox beside it. Click the checkboxes to select the nodes to use for installation.

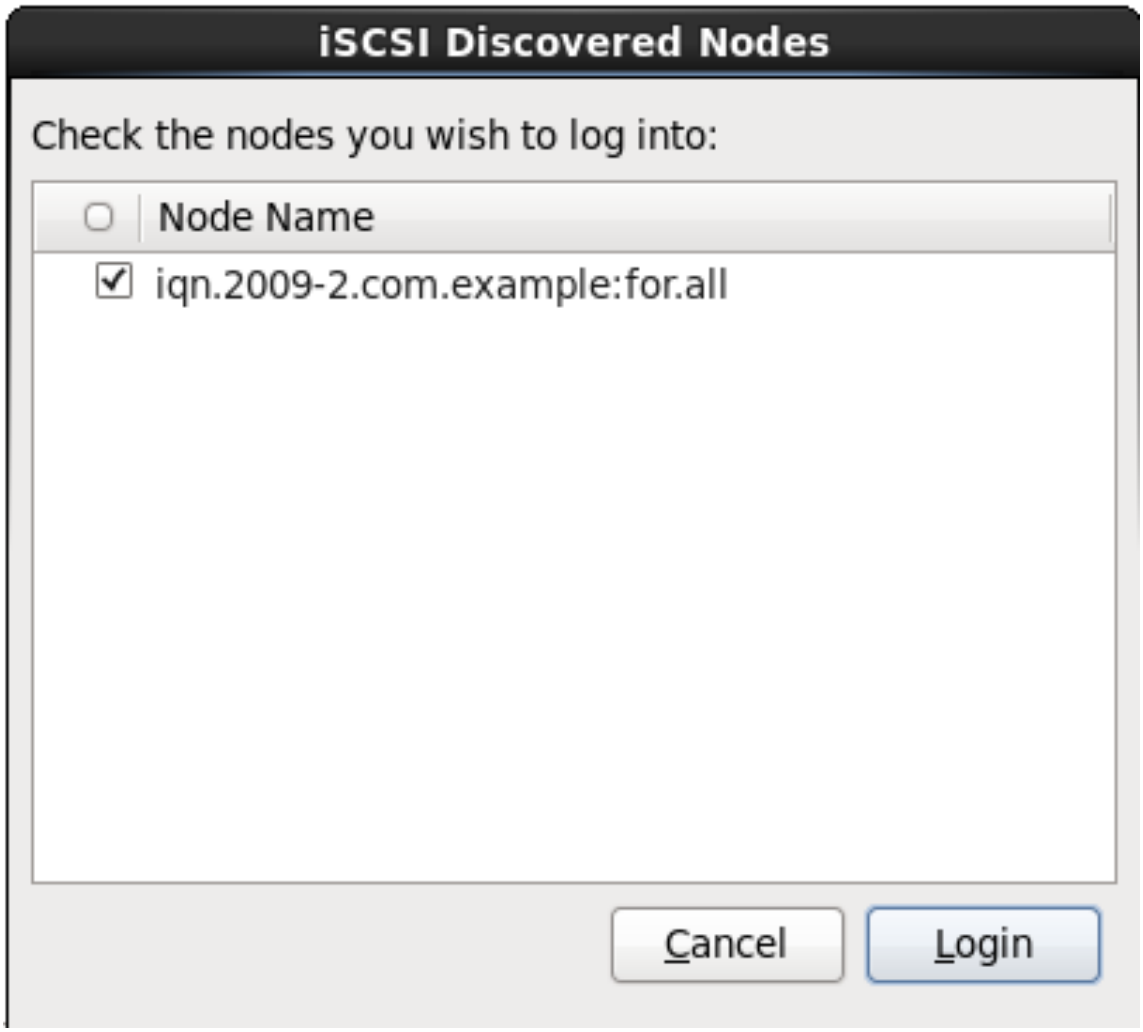


Figure 16.17. The iSCSI Discovered Nodes dialog

7. Click **Login** to initiate an iSCSI session.

Procedure 16.2. Starting an iSCSI session

Use the **iSCSI Nodes Login** dialog to provide **anaconda** with the information that it needs to log into the nodes on the iSCSI target and start an iSCSI session.

Figure 16.18. The iSCSI Nodes Login dialog

1. Use the drop-down menu to specify the type of authentication to use for the iSCSI session:

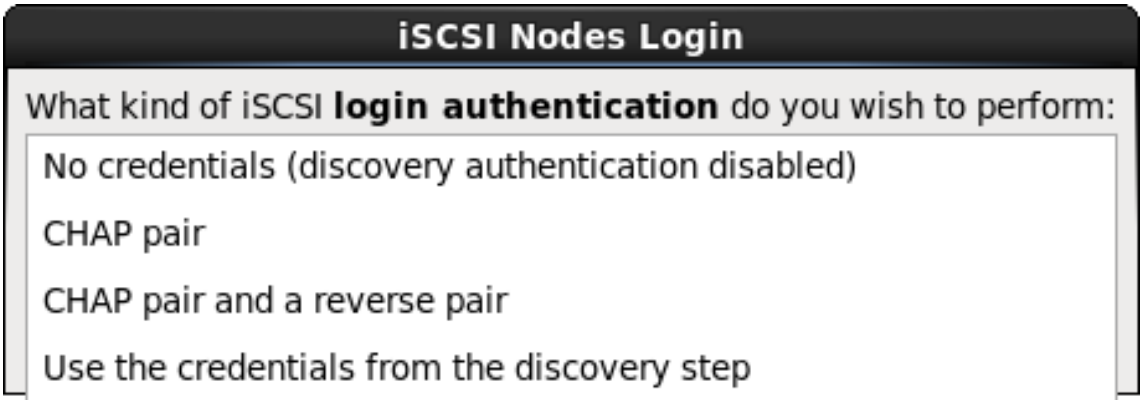


Figure 16.19. iSCSI session authentication

- **no credentials**
- **CHAP pair**
- **CHAP pair and a reverse pair**
- **Use the credentials from the discovery step**

If your environment uses the same type of authentication and same username and password for iSCSI discovery and for the iSCSI session, select **Use the credentials from the discovery step** to reuse these credentials.

2. - If you selected **CHAP pair** as the authentication type, provide the username and password for the iSCSI target in the **CHAP Username** and **CHAP Password** fields.

Figure 16.20. CHAP pair

- If you selected **CHAP pair and a reverse pair** as the authentication type, provide the username and password for the iSCSI target in the **CHAP Username** and **CHAP Password** fields and the username and password for the iSCSI initiator in the **Reverse CHAP Username** and **Reverse CHAP Password** fields.

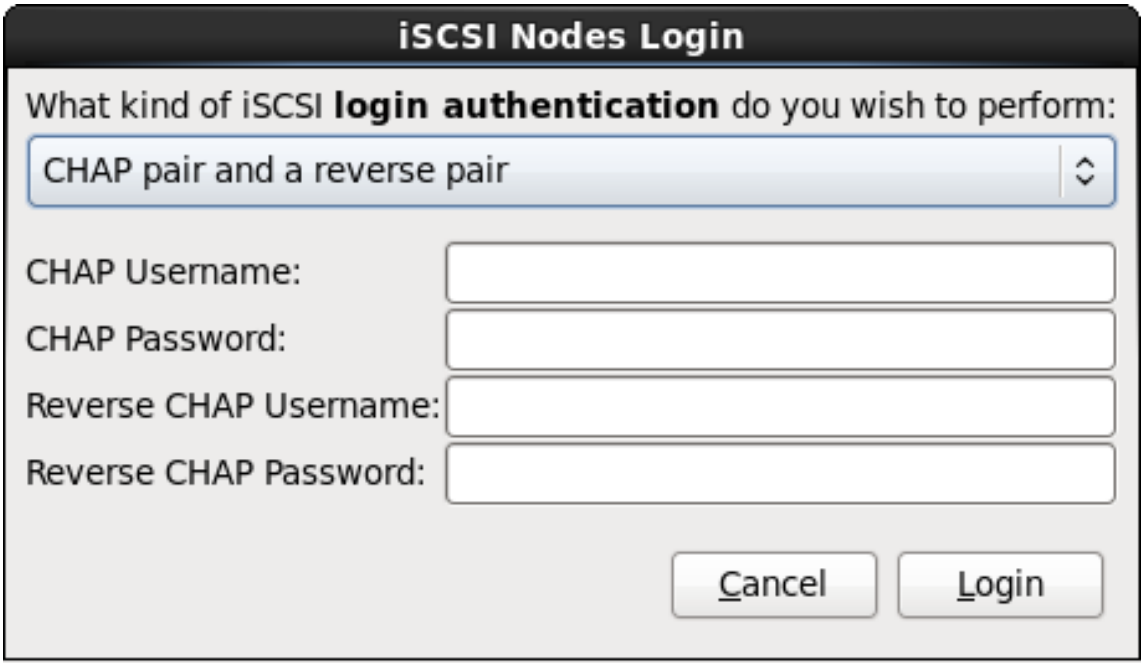


Figure 16.21. CHAP pair and a reverse pair

3. Click **Login**. **Anaconda** attempts to log into the nodes on the iSCSI target based on the information that you provided. The **iSCSI Login Results** dialog presents you with the results.

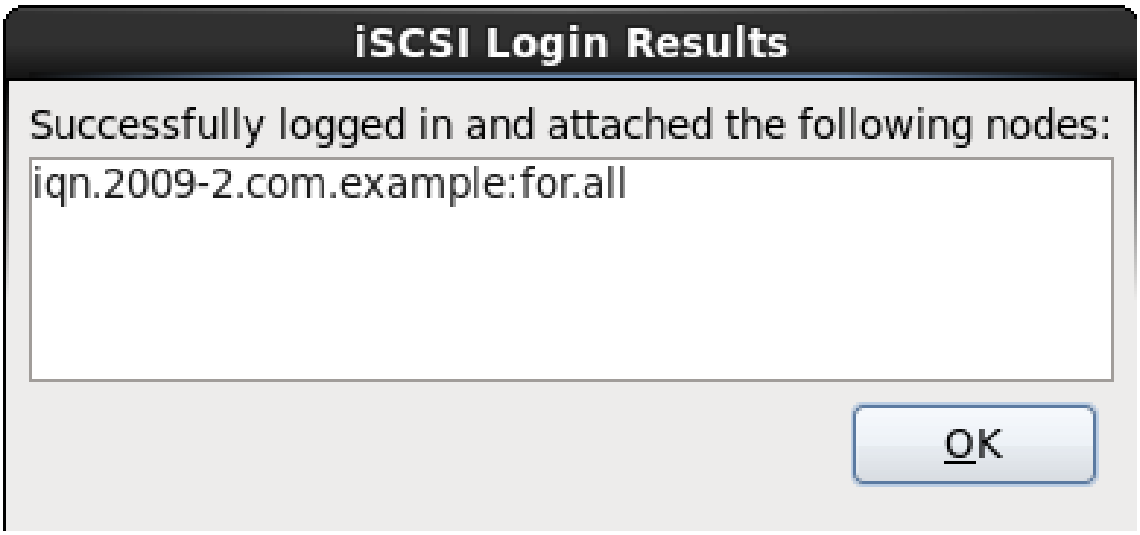


Figure 16.22. The iSCSI Login Results dialog

4. Click **OK** to continue.

16.8.1.1.3. Configure FCoE Parameters

To configure an FCoE SAN, select **Add FCoE SAN** and click **Add Drive**.

In the next dialog box that appears after you click **Add drive**, select the network interface that is connected to your FCoE switch and click **Add FCoE Disk(s)**.

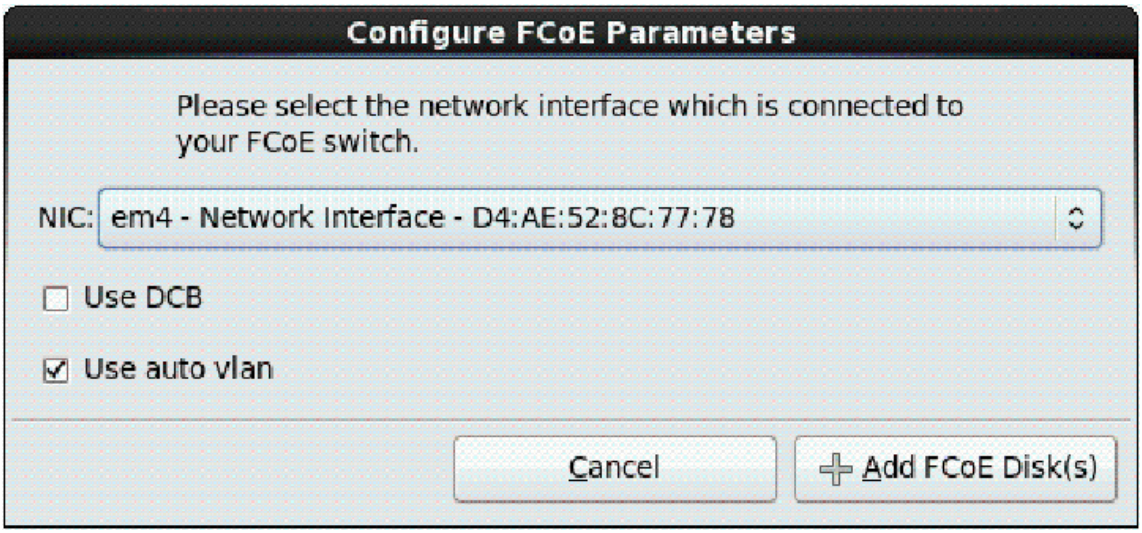


Figure 16.23. Configure FCoE Parameters

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. Enable or disable the installer's awareness of DCB with the checkbox in this dialog. This should only be set for networking interfaces that require a host-based DCBX client. Configurations on interfaces that implement a hardware DCBX client should leave this checkbox empty.

Auto VLAN indicates whether VLAN discovery should be performed. If this box is checked, then the FIP VLAN discovery protocol will run on the Ethernet interface once the link configuration has been validated. If they are not already configured, network interfaces for any discovered FCoE VLANs will be automatically created and FCoE instances will be created on the VLAN interfaces.

16.9. SETTING THE HOSTNAME

Setup prompts you to supply a host name for this computer, either as a *fully-qualified domain name* (FQDN) in the format *hostname.domainname* or as a *short host name* in the format *hostname*. 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 machine, specify the short host name only.

NOTE

You may give your system any name provided that the full hostname is unique. The hostname may include letters, numbers and hyphens.

Figure 16.24. Setting the hostname

If your Red Hat Enterprise Linux system is connected *directly* to the Internet, you must pay attention to additional considerations to avoid service interruptions or risk action by your upstream service provider. A full discussion of these issues is beyond the scope of this document.

NOTE

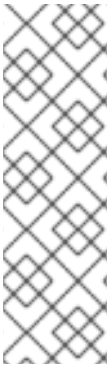
The installation program does not configure modems. Configure these devices after installation with the **Network** utility. The settings for your modem are specific to your particular Internet Service Provider (ISP).

16.9.1. Editing Network Connections

IMPORTANT

When a Red Hat Enterprise Linux 6.9 installation boots for the first time, it activates any network interfaces that you configured during the installation process. However, the installer does not prompt you to configure network interfaces on some common installation paths, for example, when you install Red Hat Enterprise Linux from a DVD to a local hard drive.

When you install Red Hat Enterprise Linux from a local installation source to a local storage device, be sure to configure at least one network interface manually if you require network access when the system boots for the first time. You will need to select the **Connect automatically** option manually when editing the connection.



NOTE

To change your network configuration after you have completed the installation, use the **Network Administration Tool**.

Type the **system-config-network** command in a shell prompt to launch the **Network Administration Tool**. If you are not root, it prompts you for the root password to continue.

The **Network Administration Tool** is now deprecated and will be replaced by **NetworkManager** during the lifetime of Red Hat Enterprise Linux 6.

To configure a network connection manually, click the button **Configure Network**. The **Network Connections** dialog appears that allows you to configure wired, wireless, mobile broadband, InfiniBand, VPN, DSL, VLAN, and bonded connections for the system using the **NetworkManager** tool. A full description of all configurations possible with **NetworkManager** is beyond the scope of this guide. This section only details the most typical scenario of how to configure wired connections during installation. Configuration of other types of network is broadly similar, although the specific parameters that you must configure are necessarily different.

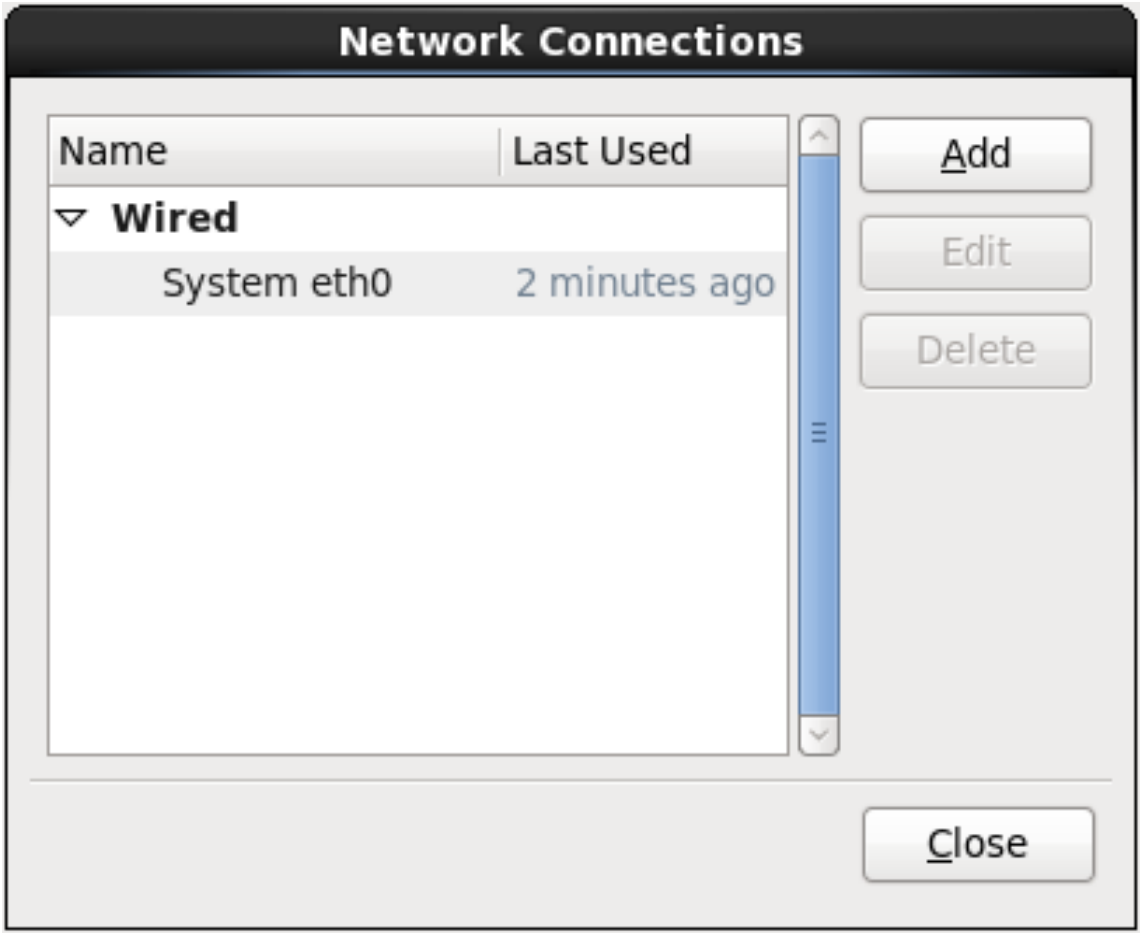


Figure 16.25. Network Connections

To add a new connection, click **Add** and select a connection type from the menu. To modify an existing connection, select it in the list and click **Edit**. In either case, a dialog box appears with a set of tabs that is appropriate to the particular connection type, as described below. To remove a connection, select it in the list and click **Delete**.

When you have finished editing network settings, click **Apply** to save the new configuration. If you reconfigured a device that was already active during installation, you must restart the device to use the new configuration – refer to Section 9.7.1.6, “Restart a network device” .

16.9.1.1. Options common to all types of connection

Certain configuration options are common to all connection types.

Specify a name for the connection in the **Connection name** name field.

Select **Connect automatically** to start the connection automatically when the system boots.

When **NetworkManager** runs on an installed system, the **Available to all users** option controls whether a network configuration is available system-wide or not. During installation, ensure that **Available to all users** remains selected for any network interface that you configure.

16.9.1.2. The Wired tab

Use the **Wired** tab to specify or change the *media access control* (MAC) address for the network adapter, and either set the *maximum transmission unit* (MTU, in bytes) that can pass through the interface.

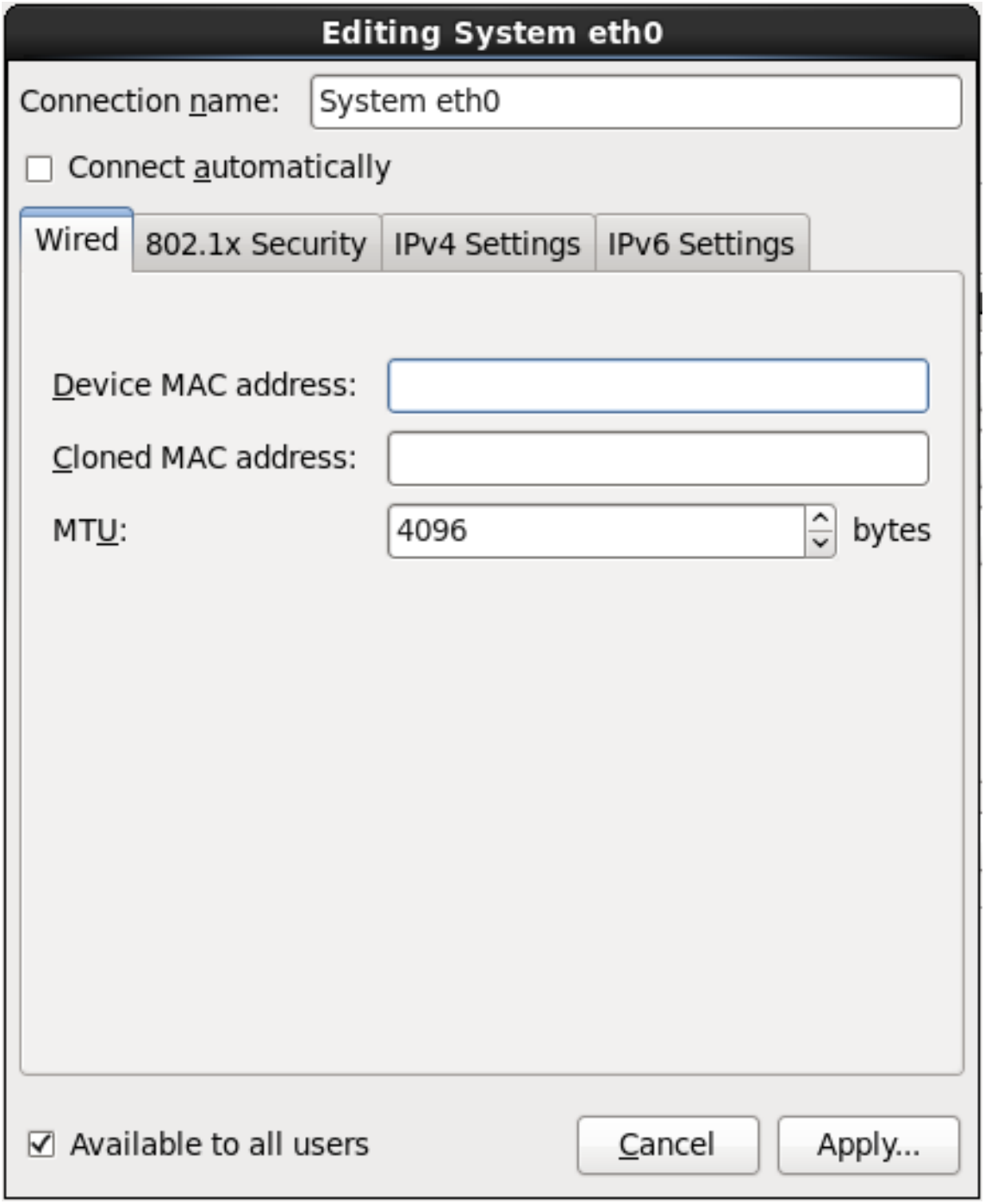


Figure 16.26. The Wired tab

16.9.1.3. The 802.1x Security tab

Use the **802.1x Security** tab to configure 802.1X *port-based network access control* (PNAC). Select **Use 802.1X security for this connection** to enable access control, then specify details of your network. The configuration options include:

Authentication

Choose one of the following methods of authentication:

- **TLS** for *Transport Layer Security*
- **Tunneled TLS** for *Tunneled Transport Layer Security*, otherwise known as TTLS, or EAP-TTLS

- **Protected EAP (PEAP)** for *Protected Extensible Authentication Protocol*

Identity

Provide the identity of this server.

User certificate

Browse to a personal X.509 certificate file encoded with *Distinguished Encoding Rules* (DER) or *Privacy Enhanced Mail* (PEM).

CA certificate

Browse to a X.509 *certificate authority* certificate file encoded with *Distinguished Encoding Rules* (DER) or *Privacy Enhanced Mail* (PEM).

Private key

Browse to a *private key* file encoded with *Distinguished Encoding Rules* (DER), *Privacy Enhanced Mail* (PEM), or the *Personal Information Exchange Syntax Standard* (PKCS#12).

Private key password

The password for the private key specified in the **Private key** field. Select **Show password** to make the password visible as you type it.

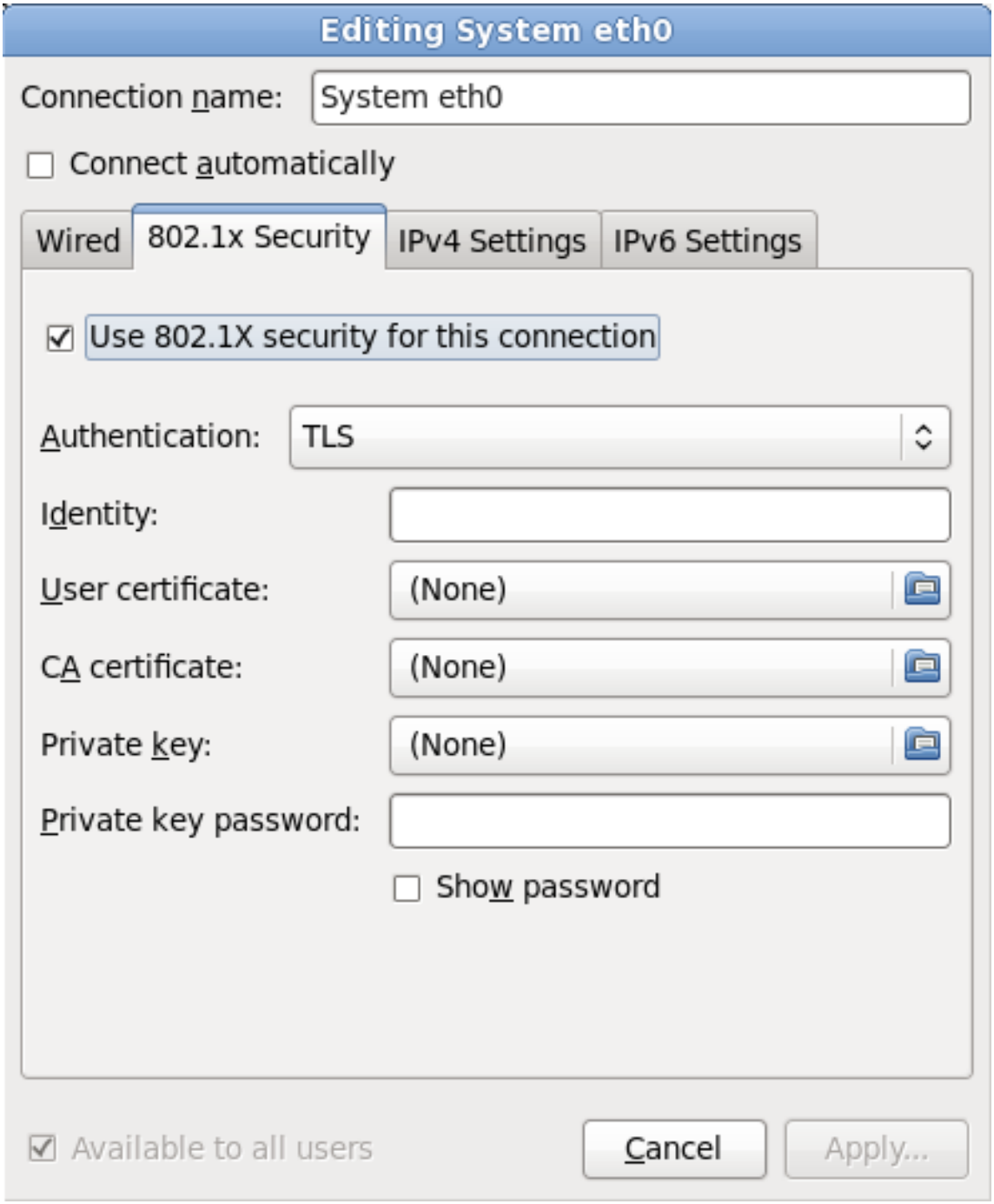


Figure 16.27. The 802.1x Security tab

16.9.1.4. The IPv4 Settings tab

Use the **IPv4 Settings tab** tab to configure the IPv4 parameters for the previously selected network connection.

Use the **Method** drop-down menu to specify which settings the system should attempt to obtain from a *Dynamic Host Configuration Protocol* (DHCP) service running on the network. Choose from the following options:

Automatic (DHCP)

IPv4 parameters are configured by the DHCP service on the network.

Automatic (DHCP) addresses only

The IPv4 address, netmask, and gateway address are configured by the DHCP service on the network, but DNS servers and search domains must be configured manually.

Manual

IPv4 parameters are configured manually for a static configuration.

Link-Local Only

A *link-local* address in the 169.254/16 range is assigned to the interface.

Shared to other computers

The system is configured to provide network access to other computers. The interface is assigned an address in the 10.42.x.1/24 range, a DHCP server and DNS server are started, and the interface is connected to the default network connection on the system with *network address translation* (NAT).

Disabled

IPv4 is disabled for this connection.

If you selected a method that requires you to supply manual parameters, enter details of the IP address for this interface, the netmask, and the gateway in the **Addresses** field. Use the **Add** and **Delete** buttons to add or remove addresses. Enter a comma-separated list of DNS servers in the **DNS servers** field, and a comma-separated list of domains in the **Search domains** field for any domains that you want to include in name server lookups.

Optionally, enter a name for this network connection in the **DHCP client ID** field. This name must be unique on the subnet. When you assign a meaningful DHCP client ID to a connection, it is easy to identify this connection when troubleshooting network problems.

Deselect the **Require IPv4 addressing for this connection to complete** check box to allow the system to make this connection on an IPv6-enabled network if IPv4 configuration fails but IPv6 configuration succeeds.

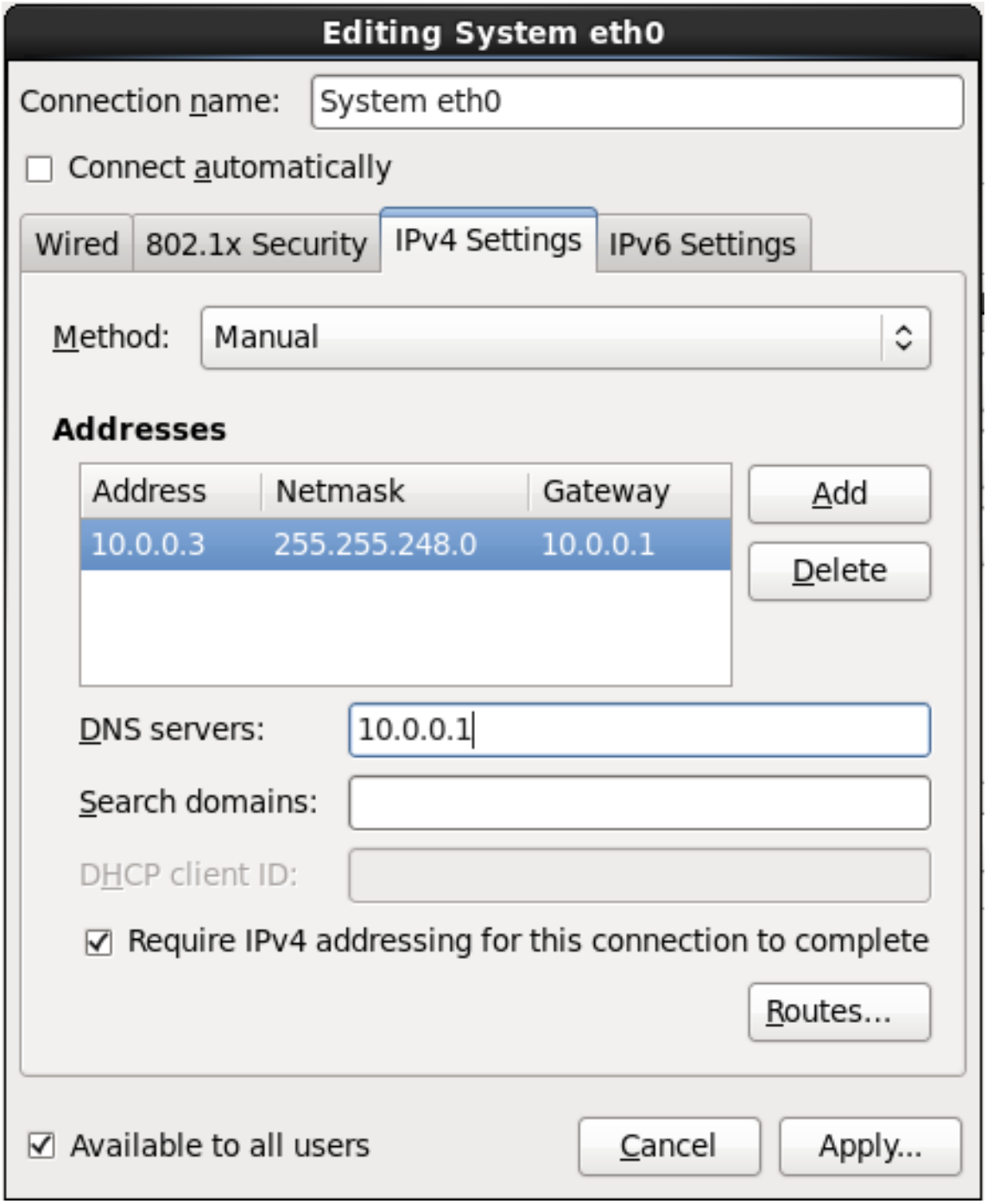


Figure 16.28. The IPv4 Settings tab

16.9.1.4.1. Editing IPv4 routes

Red Hat Enterprise Linux configures a number of routes automatically based on the IP addresses of a device. To edit additional routes, click the **Routes** button. The **Editing IPv4 routes** dialog appears.

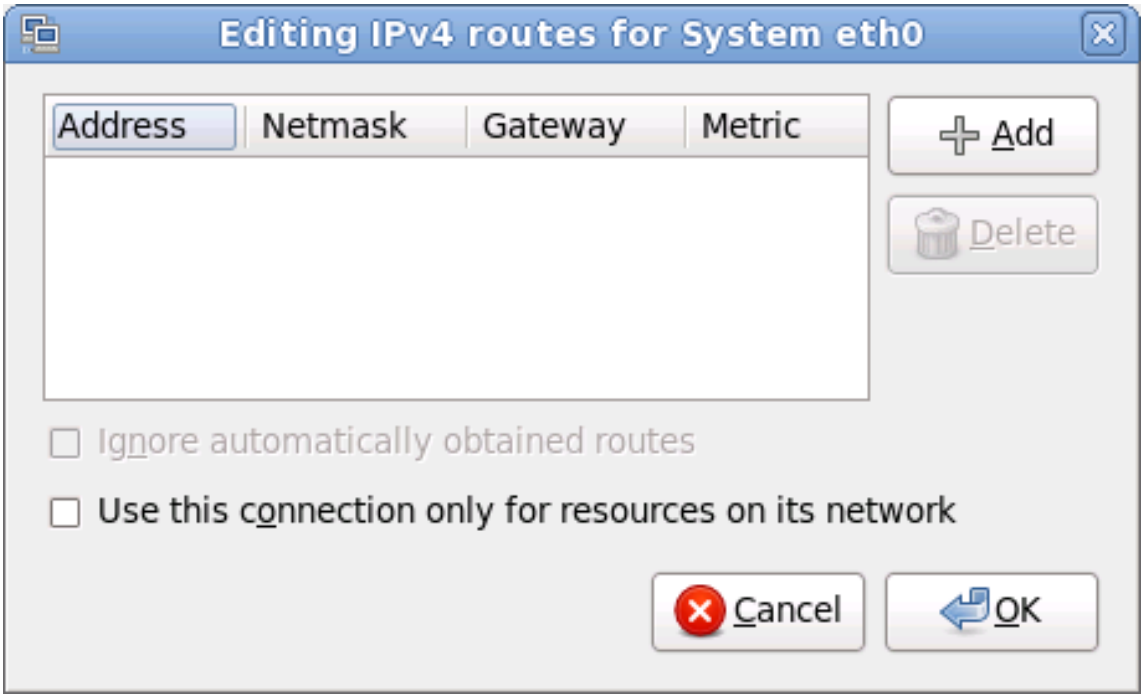


Figure 16.29. The Editing IPv4 Routes dialog

Click **Add** to add the IP address, netmask, gateway address, and metric for a new static route.

Select **Ignore automatically obtained routes** to make the interface use only the routes specified for it here.

Select **Use this connection only for resources on its network** to restrict connections only to the local network.

16.9.1.5. The IPv6 Settings tab

Use the **IPv6 Settings tab** tab to configure the IPv6 parameters for the previously selected network connection.

Use the **Method** drop-down menu to specify which settings the system should attempt to obtain from a *Dynamic Host Configuration Protocol* (DHCP) service running on the network. Choose from the following options:

Ignore

IPv6 is ignored for this connection.

Automatic

NetworkManager uses *router advertisement* (RA) to create an automatic, stateless configuration.

Automatic, addresses only

NetworkManager uses RA to create an automatic, stateless configuration, but DNS servers and search domains are ignored and must be configured manually.

Automatic, DHCP only

NetworkManager does not use RA, but requests information from DHCPv6 directly to create a stateful configuration.

Manual

IPv6 parameters are configured manually for a static configuration.

Link-Local Only

A *link-local* address with the fe80::/10 prefix is assigned to the interface.

If you selected a method that requires you to supply manual parameters, enter details of the IP address for this interface, the netmask, and the gateway in the **Addresses** field. Use the **Add** and **Delete** buttons to add or remove addresses. Enter a comma-separated list of DNS servers in the **DNS servers** field, and a comma-separated list of domains in the **Search domains** field for any domains that you want to include in name server lookups.

Optionally, enter a name for this network connection in the **DHCP client ID** field. This name must be unique on the subnet. When you assign a meaningful DHCP client ID to a connection, it is easy to identify this connection when troubleshooting network problems.

Deselect the **Require IPv6 addressing for this connection to complete** check box to allow the system to make this connection on an IPv4-enabled network if IPv6 configuration fails but IPv4 configuration succeeds.

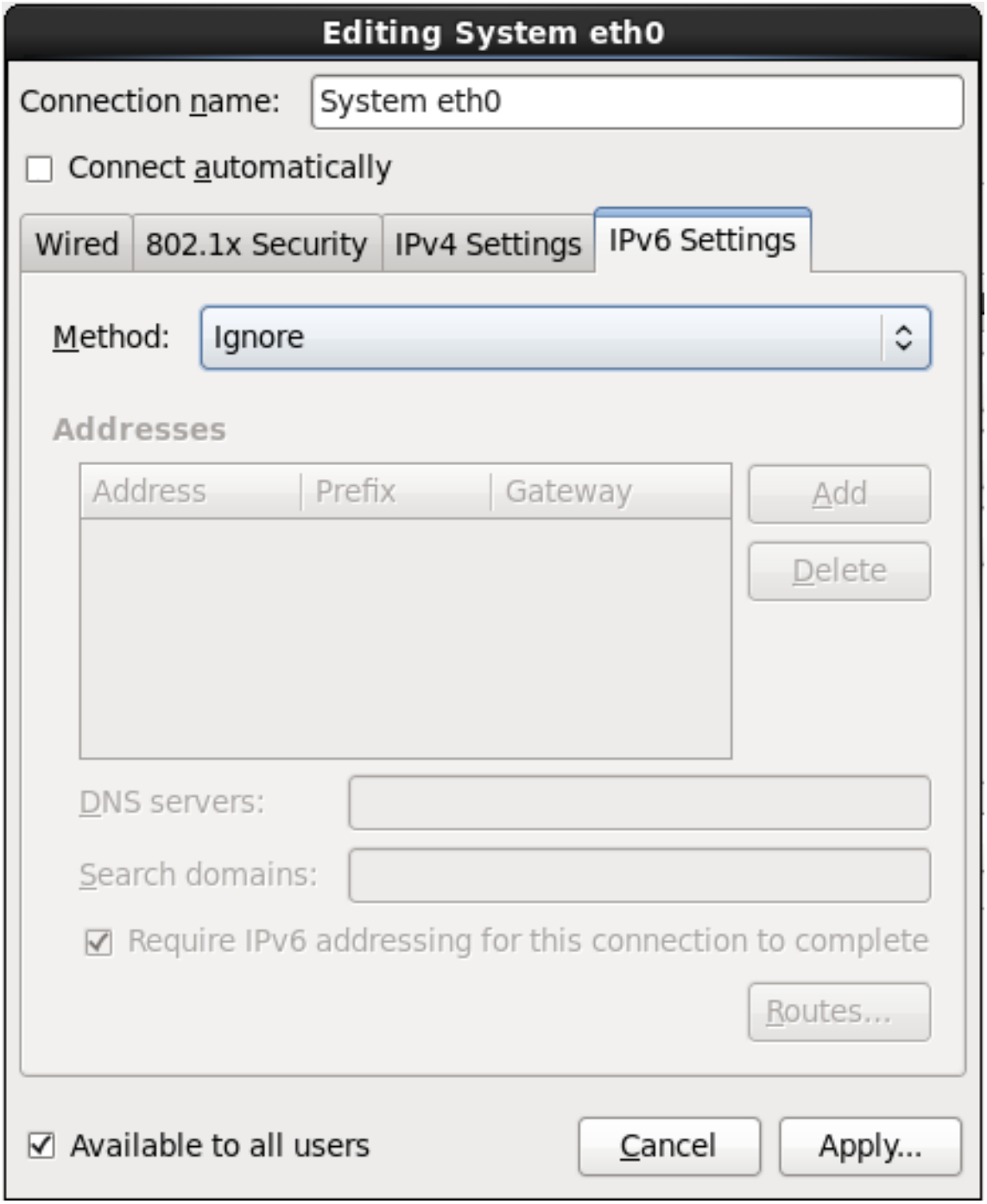


Figure 16.30. The IPv6 Settings tab

16.9.1.5.1. Editing IPv6 routes

Red Hat Enterprise Linux configures a number of routes automatically based on the IP addresses of a device. To edit additional routes, click the **Routes** button. The **Editing IPv6 routes** dialog appears.

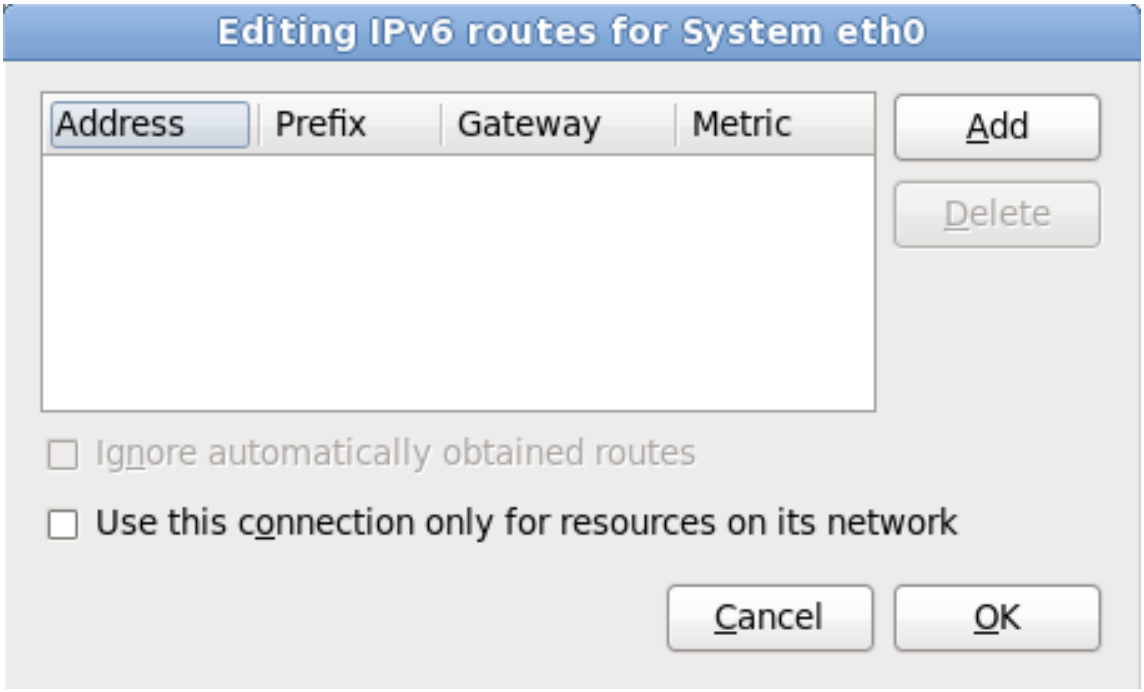


Figure 16.31. The Editing IPv6 Routes dialog

Click **Add** to add the IP address, netmask, gateway address, and metric for a new static route.

Select **Use this connection only for resources on its network** to restrict connections only to the local network.

16.9.1.6. Restart a network device

If you reconfigured a network that was already in use during installation, you must disconnect and reconnect the device in **anaconda** for the changes to take effect. **Anaconda** uses *interface configuration* (ifcfg) files to communicate with **NetworkManager**. A device becomes disconnected when its ifcfg file is removed, and becomes reconnected when its ifcfg file is restored, as long as **ONBOOT=yes** is set. Refer to the *Red Hat Enterprise Linux 6.9 Deployment Guide* available from https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/index.html for more information about interface configuration files.

1. Press **Ctrl**+**Alt**+**F2** to switch to virtual terminal **tty2**.

2. Move the interface configuration file to a temporary location:

   ```
   mv /etc/sysconfig/network-scripts/ifcfg-device_name /tmp
   ```

   where *device_name* is the device that you just reconfigured. For example, **ifcfg-eth0** is the ifcfg file for **eth0**.

   The device is now disconnected in **anaconda**.

3. Open the interface configuration file in the **vi** editor:

   ```
   vi /tmp/ifcfg-device_name
   ```

4. Verify that the interface configuration file contains the line **ONBOOT=yes**. If the file does not already contain the line, add it now and save the file.

5. Exit the **vi** editor.

6. Move the interface configuration file back to the **/etc/sysconfig/network-scripts/** directory:

   ```
   mv /tmp/ifcfg-device_name /etc/sysconfig/network-scripts/
   ```

   The device is now reconnected in **anaconda**.

7. Press **Ctrl**+**Alt**+**F6** to return to **anaconda**.

16.10. TIME ZONE CONFIGURATION

Set your time zone by selecting the city closest to your computer's physical location. Click on the map to zoom in to a particular geographical region of the world.

From here there are two ways for you to select your time zone:

- Using your mouse, click on the interactive map to select a specific city (represented by a yellow dot). A red **X** appears indicating your selection.
- You can also scroll through the list at the bottom of the screen to select your time zone. Using your mouse, click on a location to highlight your selection.

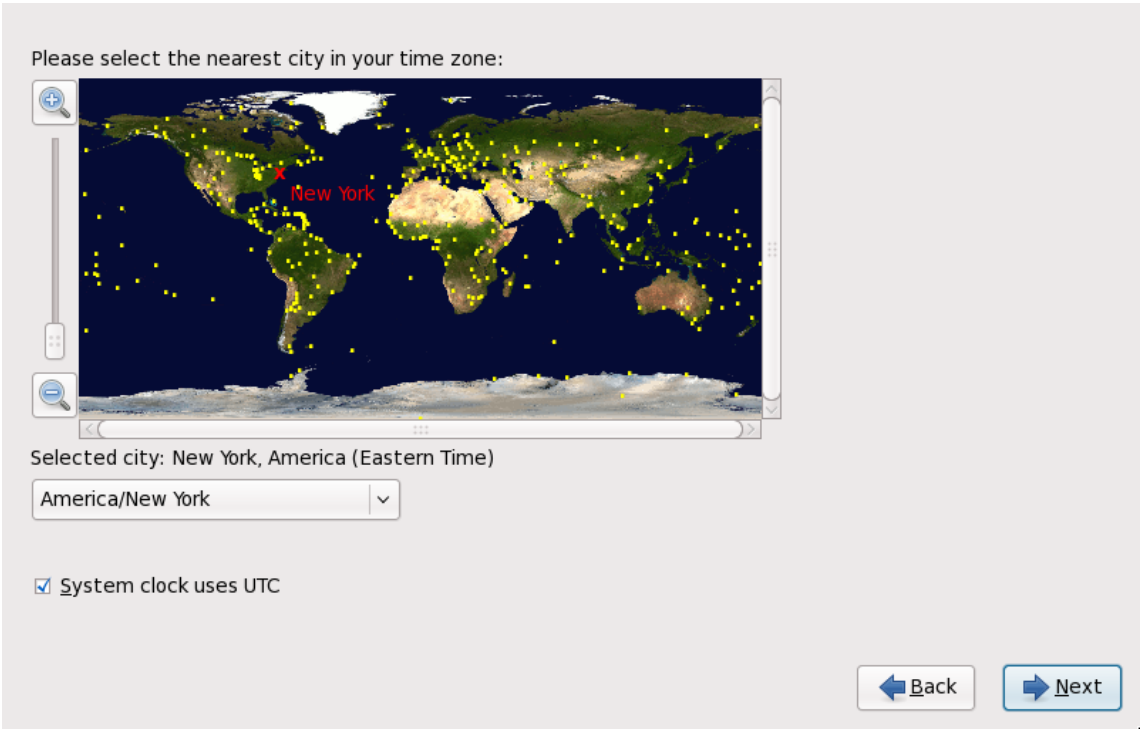
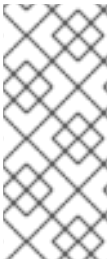


Figure 16.32. Configuring the Time Zone

If Red Hat Enterprise Linux is the only operating system on your computer, select **System clock uses UTC**. The system clock is a piece of hardware on your computer system. Red Hat Enterprise Linux uses the timezone setting to determine the offset between the local time and UTC on the system clock. This behavior is standard for systems that use UNIX, Linux, and similar operating systems.

Click **Next** to proceed.



NOTE

To change your time zone configuration after you have completed the installation, use the **Time and Date Properties Tool**.

Type the **system-config-date** command in a shell prompt to launch the **Time and Date Properties Tool**. If you are not root, it prompts you for the root password to continue.

16.11. SET THE ROOT PASSWORD

Setting up a root account and password is one of the most important steps during your installation. The root account is used to install packages, upgrade RPMs, and perform most system maintenance. Logging in as root gives you complete control over your system.

NOTE

The root user (also known as the superuser) has complete access to the entire system; for this reason, logging in as the root user is best done *only* to perform system maintenance or administration.

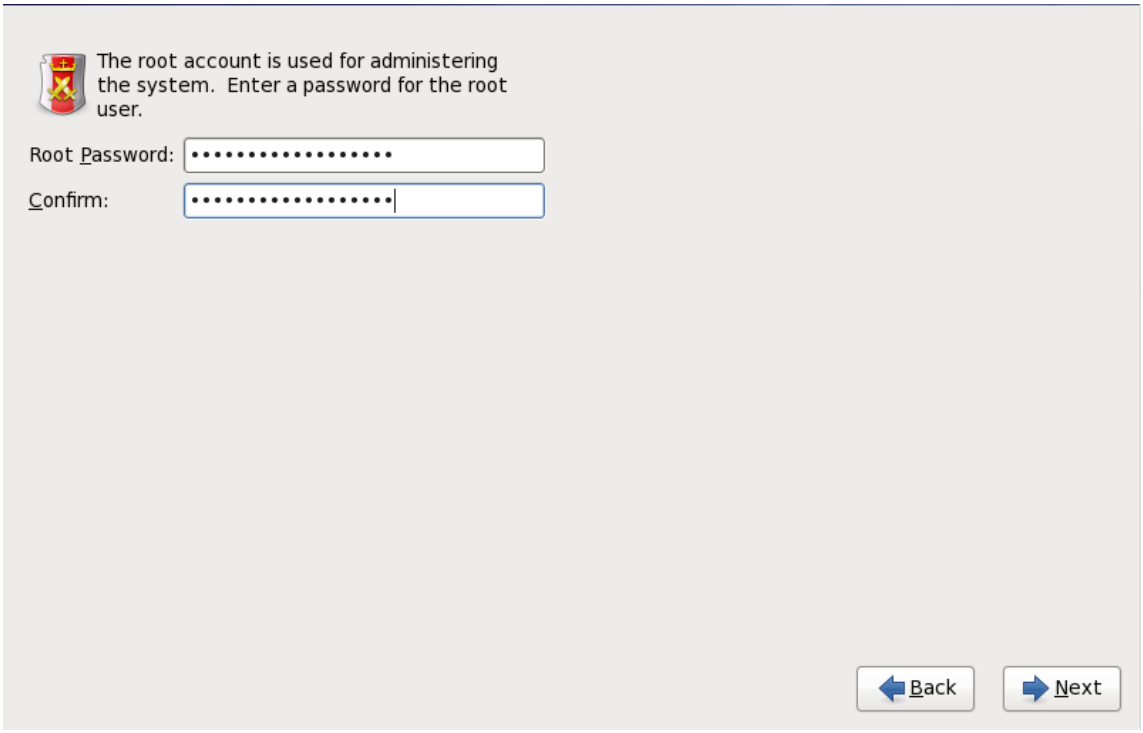
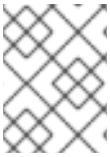


Figure 16.33. Root Password

Use the root account only for system administration. Create a non-root account for your general use and use the **su** command to change to root only when you need to perform tasks that require superuser authorization. These basic rules minimize the chances of a typo or an incorrect command doing damage to your system.



NOTE

To become root, type **su -** at the shell prompt in a terminal window and then press **Enter**. Then, enter the root password and press **Enter**.

The installation program prompts you to set a root password[7] for your system. . *You cannot proceed to the next stage of the installation process without entering a root password.*

The root password must be at least six characters long; the password you type is not echoed to the screen. You must enter the password twice; if the two passwords do not match, the installation program asks you to enter them again.

You should make the root password something you can remember, but not something that is easy for someone else to guess. Your name, your phone number, *qwerty*, *password, root*, *123456*, and *anteater* are all examples of bad passwords. Good passwords mix numerals with upper and lower case letters and do not contain dictionary words: *Aard387vark* or *420BMttNT*, for example. Remember that the password is case-sensitive. If you write down your password, keep it in a secure place. However, it is recommended that you do not write down this or any password you create.

WARNING

Do not use one of the example passwords offered in this manual. Using one of these passwords could be considered a security risk.

To change your root password after you have completed the installation, run the **passwd** command as **root**. If you forget the root password, see Resolving Problems in System Recovery Modes in the Red Hat Enterprise Linux 6 Deployment Guide for instructions on how to set a new one.

16.12. ASSIGN STORAGE DEVICES

If you selected more than one storage device on the storage devices selection screen (refer to Section 16.8, "Storage Devices"), **anaconda** asks you to select which of these devices should be available for installation of the operating system, and which should only be attached to the file system for data storage. If you selected only one storage device, **anaconda** does not present you with this screen.

During installation, the devices that you identify here as being for data storage only are mounted as part of the file system, but are not partitioned or formatted.

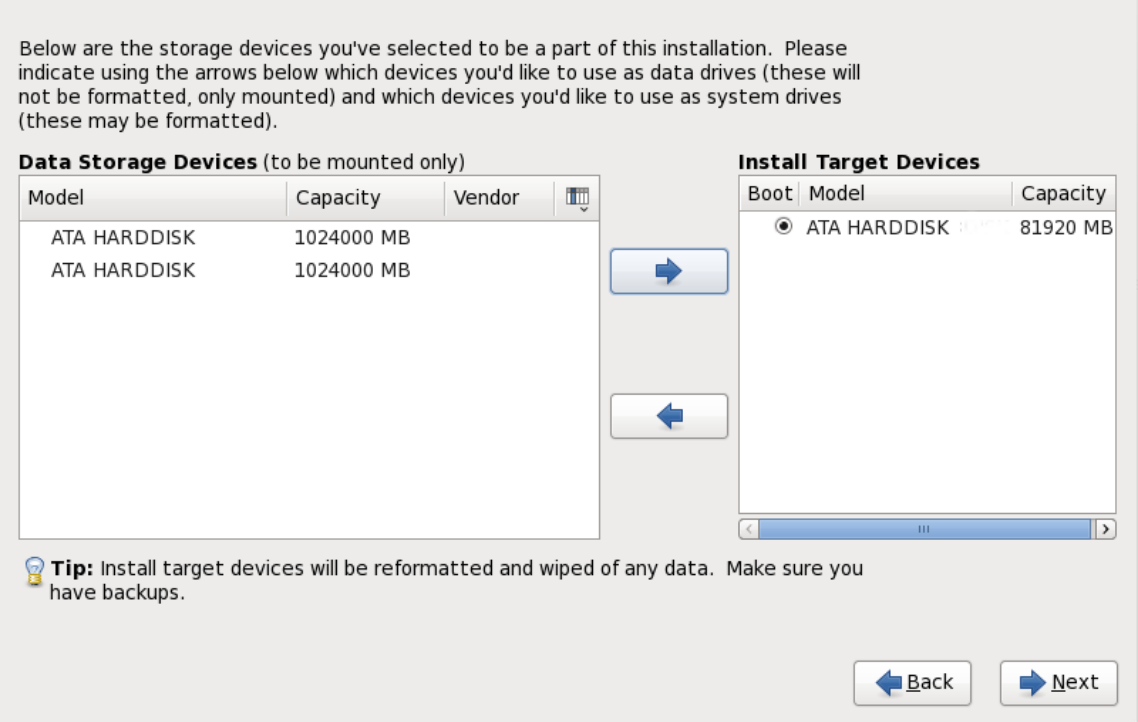


Figure 16.34. Assign storage devices

The screen is split into two panes. The left pane contains a list of devices to be used for data storage only. The right pane contains a list of devices that are to be available for installation of the operating system.

Each list contains information about the devices to help you to identify them. A small drop-down menu marked with an icon is located to the right of the column headings. This menu allows you to select the types of data presented on each device. Reducing or expanding the amount of information presented might help you to identify particular devices.

Move a device from one list to the other by clicking on the device, then clicking either the button labeled with a left-pointing arrow to move it to the list of data storage devices or the button labeled with a right-pointing arrow to move it to the list of devices available for installation of the operating system.

The list of devices available as installation targets also includes a radio button beside each device. Use this radio button to specify the device that you want to use as the boot device for the system.

IMPORTANT

If any storage device contains a boot loader that will chain load the Red Hat Enterprise Linux boot loader, include that storage device among the **Install Target Devices**. Storage devices that you identify as **Install Target Devices** remain visible to **anaconda** during boot loader configuration.

Storage devices that you identify as **Install Target Devices** on this screen are not automatically erased by the installation process unless you selected the **Use All Space** option on the partitioning screen (refer to Section 16.15, "Disk Partitioning Setup").

When you have finished identifying devices to be used for installation, click **Next** to continue.

16.13. INITIALIZING THE HARD DISK

If no readable partition tables are found on existing hard disks, the installation program asks to initialize

the hard disk. This operation makes any existing data on the hard disk unreadable. If your system has a brand new hard disk with no operating system installed, or you have removed all partitions on the hard disk, click **Re-initialize drive**.

The installation program presents you with a separate dialog for each disk on which it cannot read a valid partition table. Click the **Ignore all** button or **Re-initialize all** button to apply the same answer to all devices.

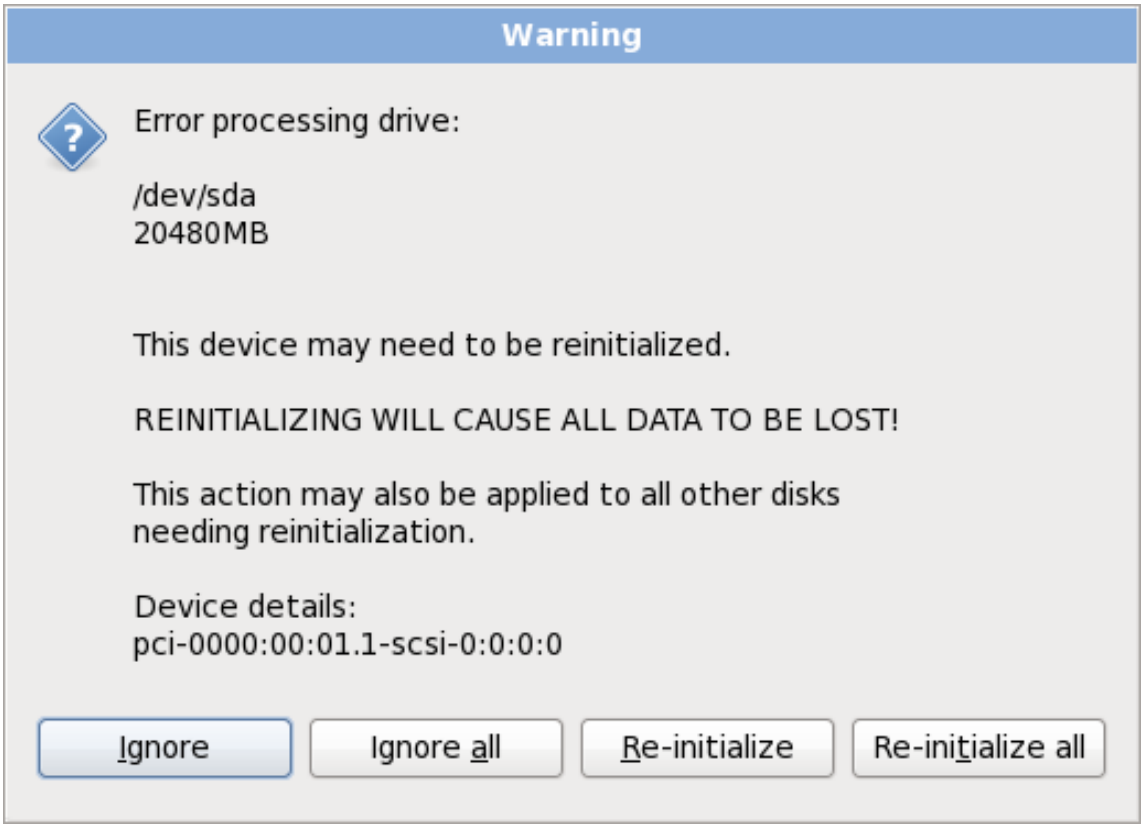


Figure 16.35. Warning screen – initializing hard drive

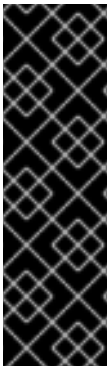
Certain RAID systems or other nonstandard configurations may be unreadable to the installation program and the prompt to initialize the hard disk may appear. The installation program responds to the physical disk structures it is able to detect.

To enable automatic initializing of hard disks for which it turns out to be necessary, use the kickstart command **zerombr** (refer to Chapter 32, *Kickstart Installations*). This command is required when performing an unattended installation on a system with previously initialized disks.

WARNING

If you have a nonstandard disk configuration that can be detached during installation and detected and configured afterward, power off the system, detach it, and restart the installation.

16.14. UPGRADING AN EXISTING SYSTEM



IMPORTANT

The following sections only apply to upgrading Red Hat Enterprise Linux between minor versions, for example, upgrading Red Hat Enterprise Linux 6.4 to Red Hat Enterprise Linux 6.5 or higher. This approach is not supported for upgrades between major versions, for example, upgrading Red Hat Enterprise Linux 6 to Red Hat Enterprise Linux 7.

In-place upgrades between major versions of Red Hat Enterprise Linux can be done, with certain limitations, using the **Red Hat Upgrade Tool** and **Preupgrade Assistant** tools. See Chapter 37, *Upgrading Your Current System* for more information.

The installation system automatically detects any existing installation of Red Hat Enterprise Linux. The upgrade process updates the existing system software with new versions, but does not remove any data from users' home directories. The existing partition structure on your hard drives does not change. Your system configuration changes only if a package upgrade demands it. Most package upgrades do not change system configuration, but rather install an additional configuration file for you to examine later.

Note that the installation medium that you are using might not contain all the software packages that you need to upgrade your computer.

16.14.1. The Upgrade Dialog

If your system contains a Red Hat Enterprise Linux installation, a dialog appears asking whether you want to upgrade that installation. To perform an upgrade of an existing system, choose the appropriate installation from the drop-down list and select **Next**.

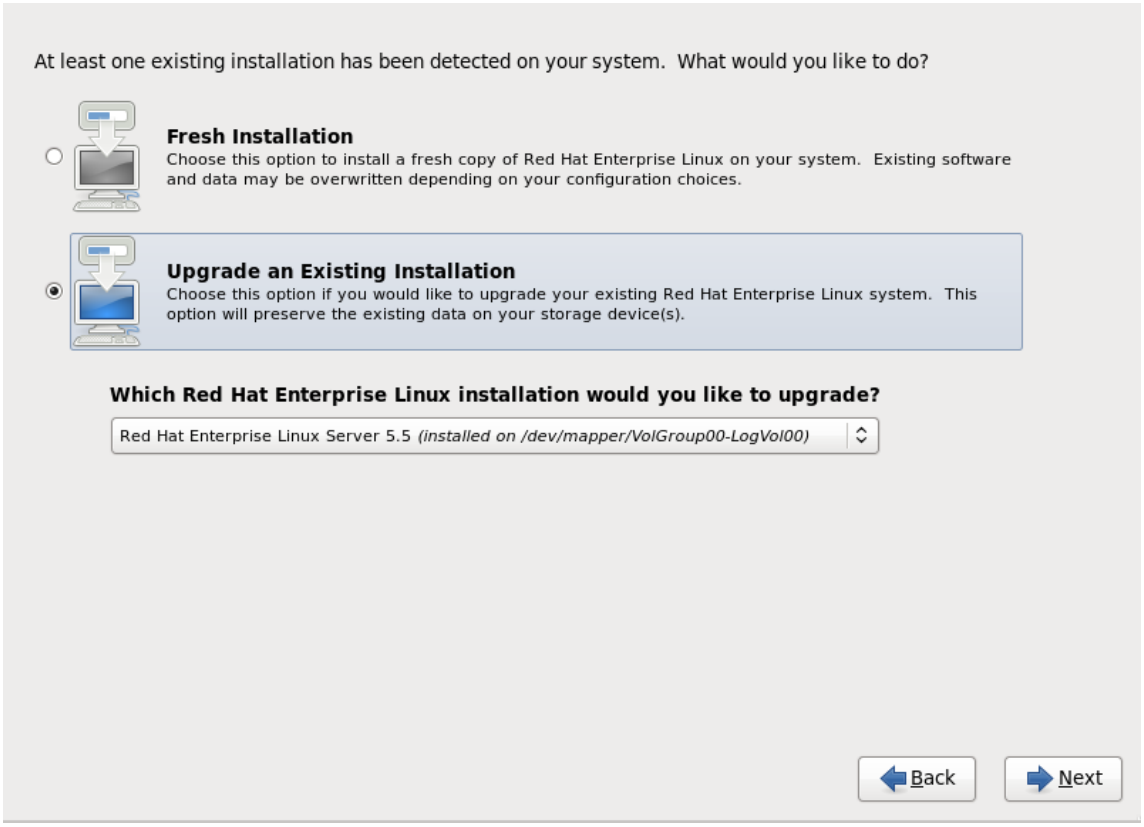
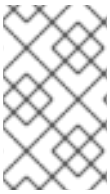


Figure 16.36. The Upgrade Dialog

NOTE

Software you have installed manually on your existing Red Hat Enterprise Linux system may behave differently after an upgrade. You may need to manually reinstall or recompile this software after an upgrade to ensure it performs correctly on the updated system.

16.14.2. Upgrading Using the Installer



NOTE

In general, Red Hat recommends that you keep user data on a separate /**home** partition and perform a fresh installation. For more information on partitions and how to set them up, refer to Section 9.13, "Disk Partitioning Setup" .

If you choose to upgrade your system using the installation program, any software not provided by Red Hat Enterprise Linux that conflicts with Red Hat Enterprise Linux software is overwritten. Before you begin an upgrade this way, make a list of your system's current packages for later reference:

```
rpm -qa --qf '%{NAME} %{VERSION}-%{RELEASE} %{ARCH}\n' > ~/old-pkglist.txt
```

After installation, consult this list to discover which packages you may need to rebuild or retrieve from sources other than Red Hat.

Next, make a backup of any system configuration data:

```
su -c 'tar czf /tmp/etc-`date +%F`.tar.gz /etc'
su -c 'mv /tmp/etc-*.tar.gz /home'
```

Make a complete backup of any important data before performing an upgrade. Important data may include the contents of your entire /**home** directory as well as content from services such as an Apache, FTP, or SQL server, or a source code management system. Although upgrades are not destructive, if you perform one improperly there is a small possibility of data loss.

WARNING

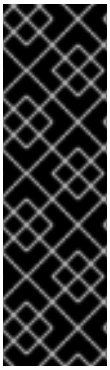
Note that the above examples store backup materials in a /**home** directory. If your /**home** directory is not a separate partition, *you should not follow these examples verbatim!* Store your backups on another device such as CD or DVD discs or an external hard disk.

For more information on completing the upgrade process later, refer to Section 35.2, "Finishing an Upgrade".

16.15. DISK PARTITIONING SETUP

WARNING

It is always a good idea to back up any data that you have on your systems. For example, if you are upgrading or creating a dual-boot system, you should back up any data you wish to keep on your storage devices. Mistakes do happen and can result in the loss of all your data.



IMPORTANT

If you install Red Hat Enterprise Linux in text mode, you can only use the default partitioning schemes described in this section. You cannot add or remove partitions or file systems beyond those that the installer automatically adds or removes. If you require a customized layout at installation time, you should perform a graphical installation over a VNC connection or a kickstart installation.

Furthermore, advanced options such as LVM, encrypted filesystems, and resizable filesystems are available only in graphical mode and kickstart.

IMPORTANT

If you have a RAID card, be aware that some BIOS types do not support booting from the RAID card. In cases such as these, the /**boot**/ partition must be created on a partition outside of the RAID array, such as on a separate hard drive. An internal hard drive is necessary to use for partition creation with problematic RAID cards.

A /**boot**/ partition is also necessary for software RAID setups.

If you have chosen to automatically partition your system, you should select **Review** and manually edit your /**boot**/ partition.

Partitioning allows you to divide your hard drive into isolated sections, where each section behaves as its own hard drive. Partitioning is particularly useful if you run multiple operating systems. If you are not sure how you want your system to be partitioned, read Appendix A, *An Introduction to Disk Partitions* for more information.

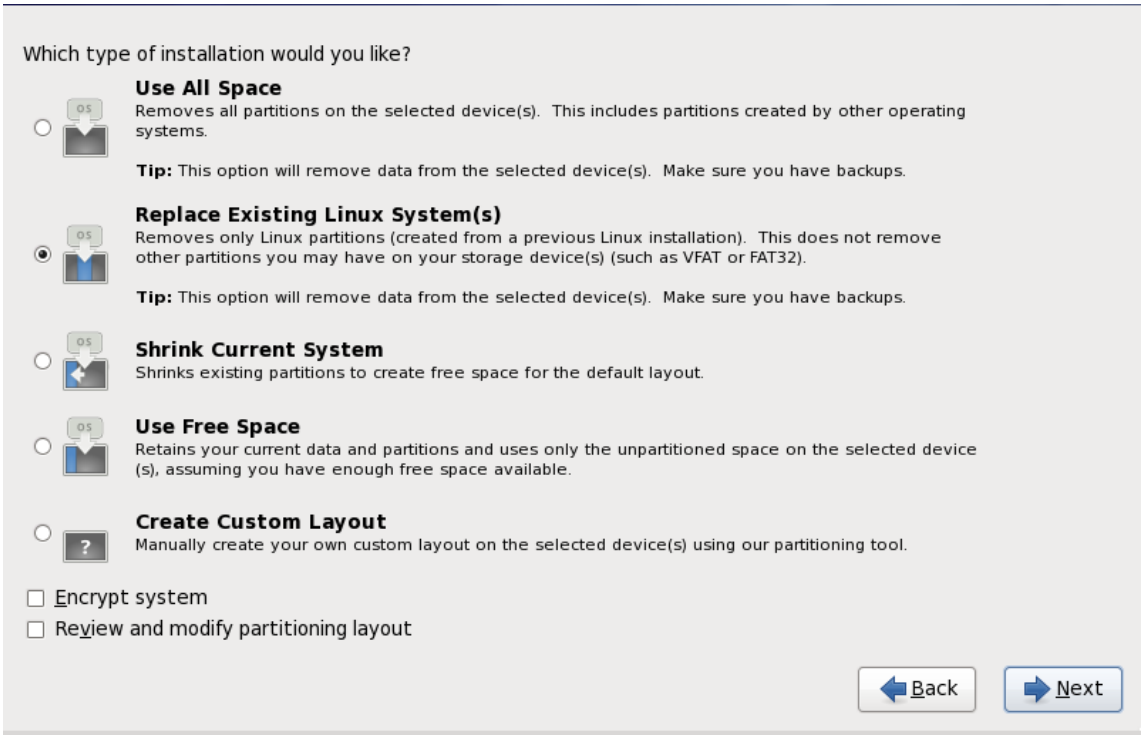


Figure 16.37. Disk Partitioning Setup

On this screen you can choose to create the default partition layout in one of four different ways, or choose to partition storage devices manually to create a custom layout.

The first four options allow you to perform an automated installation without having to partition your storage devices yourself. If you do not feel comfortable with partitioning your system, choose one of these options and let the installation program partition the storage devices for you. Depending on the option that you choose, you can still control what data (if any) is removed from the system.

Your options are:

Use All Space

Select this option to remove all partitions on your hard drives (this includes partitions created by other operating systems such as Windows VFAT or NTFS partitions).

WARNING

If you select this option, all data on the selected hard drives is removed by the installation program. Do not select this option if you have information that you want to keep on the hard drives where you are installing Red Hat Enterprise Linux.

In particular, do not select this option when you configure a system to chain load the Red Hat Enterprise Linux boot loader from another boot loader.

Replace Existing Linux System(s)

Select this option to remove only partitions created by a previous Linux installation. This does not remove other partitions you may have on your hard drives (such as VFAT or FAT32 partitions).

Shrink Current System

Select this option to resize your current data and partitions manually and install a default Red Hat Enterprise Linux layout in the space that is freed.

WARNING

If you shrink partitions on which other operating systems are installed, you might not be able to use those operating systems. Although this partitioning option does not destroy data, operating systems typically require some free space in their partitions. Before you resize a partition that holds an operating system that you might want to use again, find out how much space you need to leave free.

Use Free Space

Select this option to retain your current data and partitions and install Red Hat Enterprise Linux in the unused space available on the storage drives. Ensure that there is sufficient space available on the storage drives before you select this option – refer to Section 11.6, "Do You Have Enough Disk Space?".

Create Custom Layout

Select this option to partition storage devices manually and create customized layouts. Refer to Section 16.17, " Creating a Custom Layout or Modifying the Default Layout "

Choose your preferred partitioning method by clicking the radio button to the left of its description in the dialog box.

Select **Encrypt system** to encrypt all partitions except the **/boot** partition. Refer to Appendix C, *Disk Encryption* for information on encryption.

To review and make any necessary changes to the partitions created by automatic partitioning, select the **Review** option. After selecting **Review** and clicking **Next** to move forward, the partitions created for you by **anaconda** appear. You can make modifications to these partitions if they do not meet your needs.

IMPORTANT

To configure the Red Hat Enterprise Linux boot loader to *chain load* from a different boot loader, you must specify the boot drive manually. If you chose any of the automatic partitioning options, you must now select the **Review and modify partitioning layout** option before you click **Next** or you cannot specify the correct boot drive.

IMPORTANT

When you install Red Hat Enterprise Linux 6 on a system with multipath and non-multipath storage devices, the automatic partitioning layout in the installer might create volume groups that contain a mix of multipath and non-multipath devices. This defeats the purpose of multipath storage.

We advise that you select only multipath or only non-multipath devices on the disk selection screen that appears after selecting automatic partitioning. Alternatively, select custom partitioning.

Click **Next** once you have made your selections to proceed.

16.16. CHOOSING A DISK ENCRYPTION PASSPHRASE

If you selected the **Encrypt System** option, the installer prompts you for a passphrase with which to encrypt the partitions on the system.

Partitions are encrypted using the *Linux Unified Key Setup* — refer to Appendix C, *Disk Encryption* for more information.

Figure 16.38. Enter passphrase for encrypted partition

Choose a passphrase and type it into each of the two fields in the dialog box. You must provide this passphrase every time that the system boots.

WARNING

If you lose this passphrase, any encrypted partitions and the data on them will become completely inaccessible. There is no way to recover a lost passphrase.

Note that if you perform a kickstart installation of Red Hat Enterprise Linux, you can save encryption passphrases and create backup encryption passphrases during installation. Refer to Section C.3.2, "Saving Passphrases" and Section C.3.3, "Creating and Saving Backup Passphrases".

16.17. CREATING A CUSTOM LAYOUT OR MODIFYING THE DEFAULT LAYOUT

If you chose one of the four automatic partitioning options and did not select **Review**, skip ahead to Section 16.18, "Write Changes to Disk" .

If you chose one of the automatic partitioning options and selected **Review**, you can either accept the current partition settings (click **Next**), or modify the setup manually in the partitioning screen.

If you chose to create a custom layout, you must tell the installation program where to install Red Hat Enterprise Linux. This is done by defining mount points for one or more disk partitions in which Red Hat Enterprise Linux is installed.

If you have not yet planned how to set up your partitions, refer to Appendix A, *An Introduction to Disk Partitions* and Section 16.17.5, "Recommended Partitioning Scheme". At a bare minimum, you need an appropriately-sized root (/) partition, a /**boot**/ partition, PReP boot partition, and usually a swap partition appropriate to the amount of RAM you have on the system.

Anaconda can handle the partitioning requirements for a typical installation.

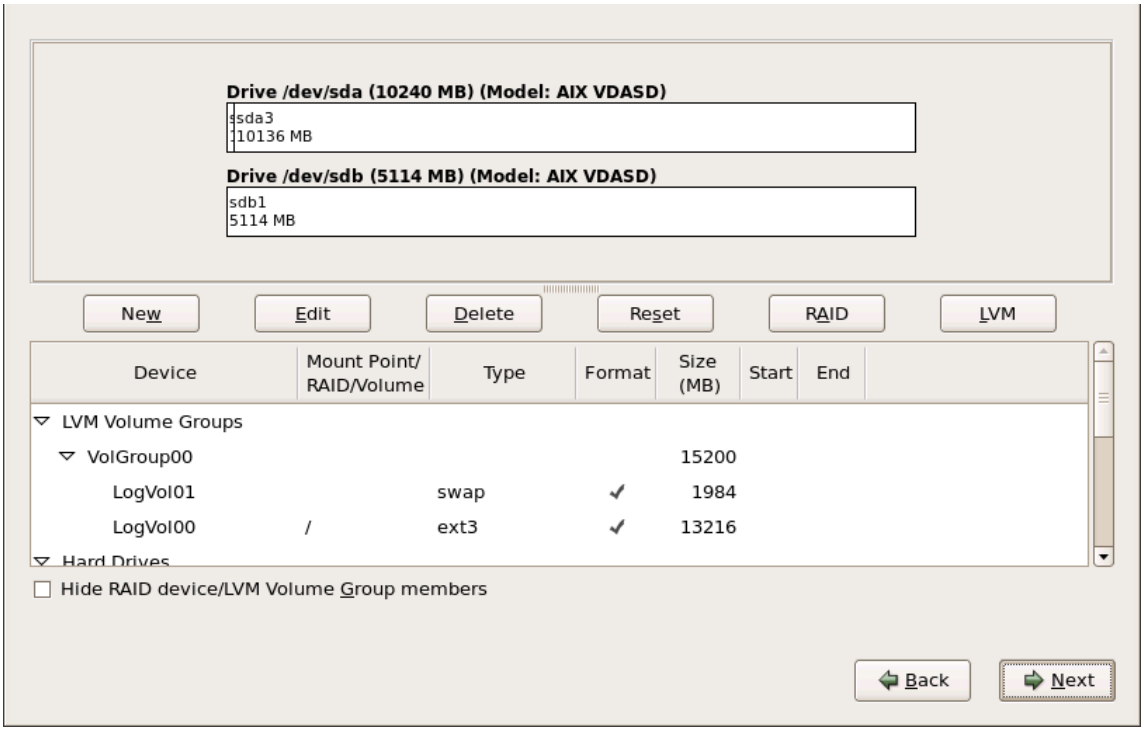


Figure 16.39. Partitioning on IBM System p

The partitioning screen contains two panes. The top pane contains a graphical representation of the hard drive, logical volume, or RAID device selected in the lower pane.

Above the graphical representation of the device, you can review the name of the drive (such as /**dev**/**sda** or **LogVol00**), its size (in MB), and its model as detected by the installation program.

Using your mouse, click once to highlight a particular field in the graphical display. Double-click to edit an existing partition or to create a partition out of existing free space.

The lower pane contains a list of all drives, logical volumes, and RAID devices to be used during installation, as specified earlier in the installation process – refer to Section 16.12, " Assign Storage Devices "

Devices are grouped by type. Click on the small triangles to the left of each device type to view or hide devices of that type.

Anaconda displays several details for each device listed:

Device
the name of the device, logical volume, or partition

Size (MB)
the size of the device, logical volume, or partition (in MB)

Mount Point/RAID/Volume
the *mount point* (location within a file system) on which a partition is to be mounted, or the name of the RAID or logical volume group of which it is a part

Type
the type of partition. If the partition is a standard partition, this field displays the type of file system on the partition (for example, ext4). Otherwise, it indicates that the partition is a **physical volume (LVM)**, or part of a **software RAID**

Format
A check mark in this column indicates that the partition will be formatted during installation.

Beneath the lower pane are four buttons: **Create**, **Edit**, **Delete**, and **Reset**.

Select a device or partition by clicking on it in either the graphical representation in the upper pane of in the list in the lower pane, then click one of the four buttons to carry out the following actions:

Create
create a new partition, logical volume, or software RAID

Edit
change an existing partition, logical volume, or software RAID. Note that you can only shrink partitions with the **Resize** button, not enlarge partitions.

Delete
remove a partition, logical volume, or software RAID

Reset
undo all changes made in this screen

16.17.1. Create Storage

The **Create Storage** dialog allows you to create new storage partitions, logical volumes, and software RAIDs. **Anaconda** presents options as available or unavailable depending on the storage already present on the system or configured to transfer to the system.

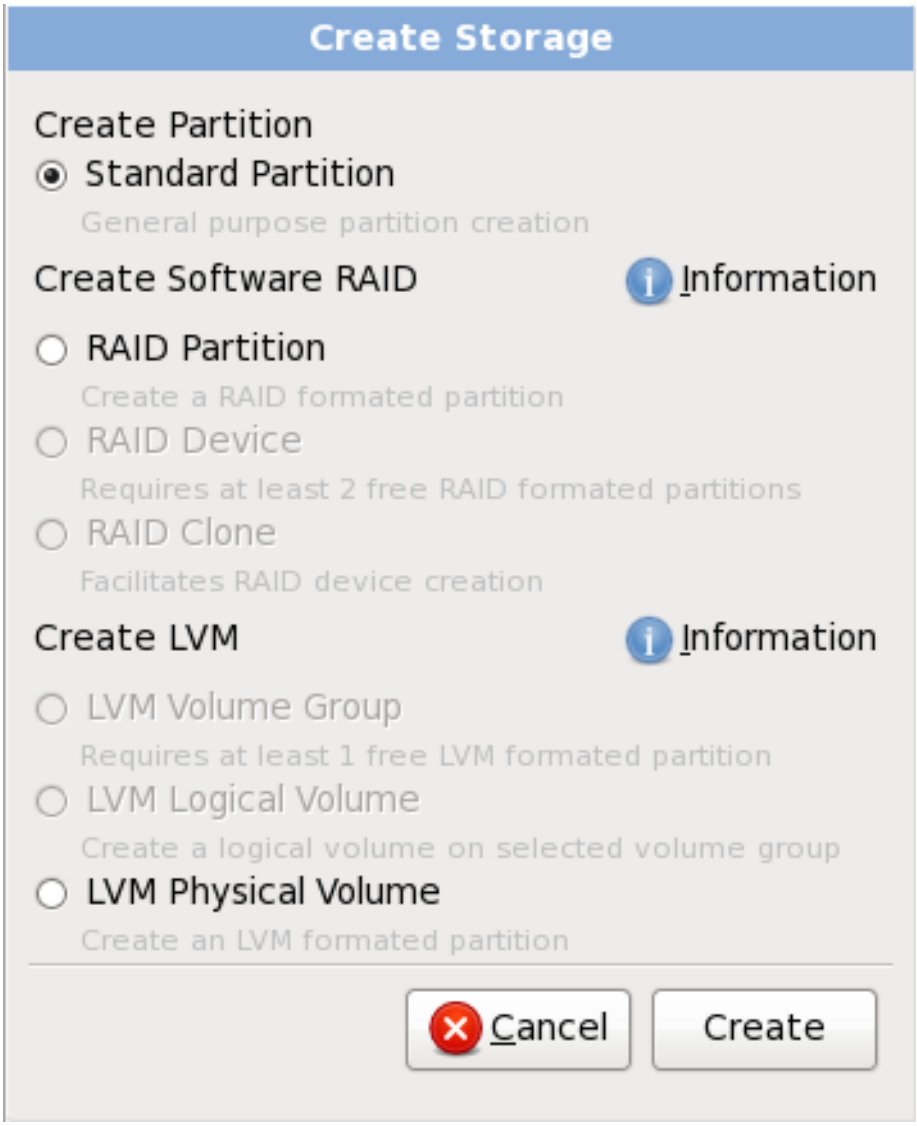


Figure 16.40. Creating Storage

Options are grouped under **Create Partition**, **Create Software RAID** and **Create LVM** as follows:

Create Partition

Refer to Section 9.15.2, "Adding Partitions" for details of the **Add Partition** dialog.

- **Standard Partition** – create a standard disk partition (as described in Appendix A, *An Introduction to Disk Partitions*) in unallocated space.

Create Software RAID

Refer to Section 23.15.3, " Create Software RAID " for more detail.

- **RAID Partition** – create a partition in unallocated space to form part of a software RAID device. To form a software RAID device, two or more RAID partitions must be available on the system.
- **RAID Device** – combine two or more RAID partitions into a software RAID device. When you choose this option, you can specify the type of RAID device to create (the *RAID level*). This option is only available when two or more RAID partitions are available on the system.

Create LVM Logical Volume

Refer to Section 16.17.4, " Create LVM Logical Volume " for more detail.

- **LVM Physical Volume** — create a *physical volume* in unallocated space.
- **LVM Volume Group** — create a *volume group* from one or more physical volumes. This option is only available when at least one physical volume is available on the system.
- **LVM Logical Volume** — create a *logical volume* on a volume group. This option is only available when at least one volume group is available on the system.

16.17.2. Adding Partitions

To add a new partition, select the **Create** button. A dialog box appears (refer to Figure 16.41, "Creating a New Partition").

NOTE

You must dedicate at least one partition for this installation, and optionally more. For more information, refer to Appendix A, *An Introduction to Disk Partitions*.

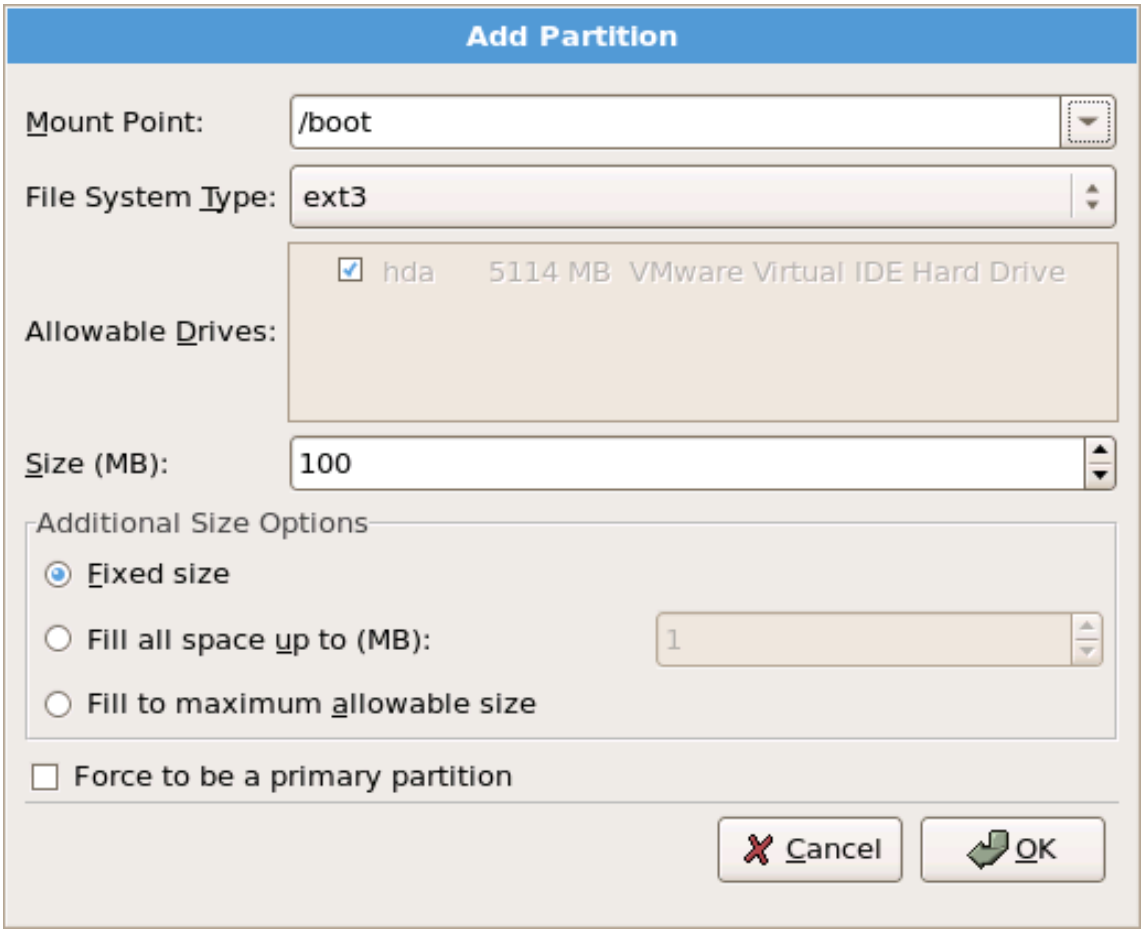


Figure 16.41. Creating a New Partition

- **Mount Point**: Enter the partition's mount point. For example, if this partition should be the root partition, enter /; enter **/boot** for the **/boot** partition, and so on. You can also use the pull-down menu to choose the correct mount point for your partition. For a swap partition the mount point should not be set — setting the filesystem type to **swap** is sufficient.

- **File System Type**: Using the pull-down menu, select the appropriate file system type for this partition. For more information on file system types, refer to Section 16.17.2.1, "File System Types".

- **Allowable Drives**: This field contains a list of the hard disks installed on your system. If a hard disk's box is highlighted, then a desired partition can be created on that hard disk. If the box is *not* checked, then the partition will *never* be created on that hard disk. By using different checkbox settings, you can have **anaconda** place partitions where you need them, or let **anaconda** decide where partitions should go.

- **Size (MB)**: Enter the size (in megabytes) of the partition. Note, this field starts with 200 MB; unless changed, only a 200 MB partition will be created.

- **Additional Size Options**: Choose whether to keep this partition at a fixed size, to allow it to "grow" (fill up the available hard drive space) to a certain point, or to allow it to grow to fill any remaining hard drive space available.

  If you choose **Fill all space up to (MB)**, you must give size constraints in the field to the right of this option. This allows you to keep a certain amount of space free on your hard drive for future use.

- **Force to be a primary partition**: Select whether the partition you are creating should be one of the first four partitions on the hard drive. If unselected, the partition is created as a logical partition. Refer to Section A.1.3, "Partitions Within Partitions — An Overview of Extended Partitions", for more information.

- **Encrypt**: Choose whether to encrypt the partition so that the data stored on it cannot be accessed without a passphrase, even if the storage device is connected to another system. Refer to Appendix C, *Disk Encryption* for information on encryption of storage devices. If you select this option, the installer prompts you to provide a passphrase before it writes the partition to the disk.

- **OK**: Select **OK** once you are satisfied with the settings and wish to create the partition.

- **Cancel**: Select **Cancel** if you do not want to create the partition.

16.17.2.1. File System Types

Red Hat Enterprise Linux allows you to create different partition types and file systems. The following is a brief description of the different partition types and file systems available, and how they can be used.

Partition types

- **standard partition** — A standard partition can contain a file system or swap space, or it can provide a container for software RAID or an LVM physical volume.

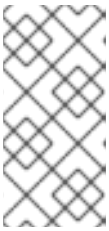
- **swap** — Swap partitions are used to support virtual memory. In other words, data is written to a swap partition when there is not enough RAM to store the data your system is processing. Refer to the Red Hat Enterprise Linux Deployment Guide for additional information.

- **software RAID** — Creating two or more software RAID partitions allows you to create a RAID device. For more information regarding RAID, refer to the chapter *RAID (Redundant Array of Independent Disks)* in the Red Hat Enterprise Linux Deployment Guide .

- **physical volume (LVM)** — Creating one or more physical volume (LVM) partitions allows you to create an LVM logical volume. LVM can improve performance when using physical disks. For more information regarding LVM, refer to the Red Hat Enterprise Linux Deployment Guide .

File systems

- **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. A maximum file system size of 16TB is supported for ext4. The ext4 file system is selected by default and is highly recommended.

  NOTE

  The **user_xattr** and **acl** mount options are automatically set on ext4 systems by the installation system. These options enable extended attributes and access control lists, respectively. More information about mount options can be found in the Red Hat Enterprise Linux Storage Administration Guide .

- **ext3** — The ext3 file system is based on the ext2 file system and has one main advantage — journaling. Using a journaling file system reduces time spent recovering a file system after a crash as there is no need to **fsck** [8] the file system. A maximum file system size of 16TB is supported for ext3.

- **ext2** — An ext2 file system supports standard Unix file types (regular files, directories, symbolic links, etc). It provides the ability to assign long file names, up to 255 characters.

- **xfs** — XFS is a highly scalable, high-performance file system that supports filesystems 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 supports metadata journaling, which facilitates quicker crash recovery. The XFS file system can also be defragmented and resized while mounted and active.

  NOTE

  The maximum size of an XFS partition the installer can create is *100 TB*.

- **vfat** — The VFAT file system is a Linux file system that is compatible with Microsoft Windows long filenames on the FAT file system.

- **Btrfs** — Btrfs is under development as a file system capable of addressing and managing more files, larger files, and larger volumes than the ext2, ext3, and ext4 file systems. Btrfs is designed to make the file system tolerant of errors, and to facilitate the detection and repair of errors when they occur. It uses checksums to ensure the validity of data and metadata, and maintains snapshots of the file system that can be used for backup or repair.

  Because Btrfs is still experimental and under development, the installation program does not offer it by default. If you want to create a Btrfs partition on a drive, you must commence the installation process with the boot option **btrfs**. Refer to Chapter 28, *Boot Options* for instructions.

WARNING

Red Hat Enterprise Linux 6.9 includes Btrfs as a technology preview to allow you to experiment with this file system. You should not choose Btrfs for partitions that will contain valuable data or that are essential for the operation of important systems.

16.17.3. Create Software RAID

Redundant arrays of independent disks (RAIDs) are constructed from multiple storage devices that are arranged to provide increased performance and – in some configurations – greater fault tolerance. Refer to the Red Hat Enterprise Linux Storage Administration Guide for a description of different kinds of RAIDs.

To make a RAID device, you must first create software RAID partitions. Once you have created two or more software RAID partitions, select **RAID** to join the software RAID partitions into a RAID device.

RAID Partition

Choose this option to configure a partition for software RAID. This option is the only choice available if your disk contains no software RAID partitions. This is the same dialog that appears when you add a standard partition – refer to Section 16.17.2, "Adding Partitions" for a description of the available options. Note, however, that **File System Type** must be set to **software RAID**

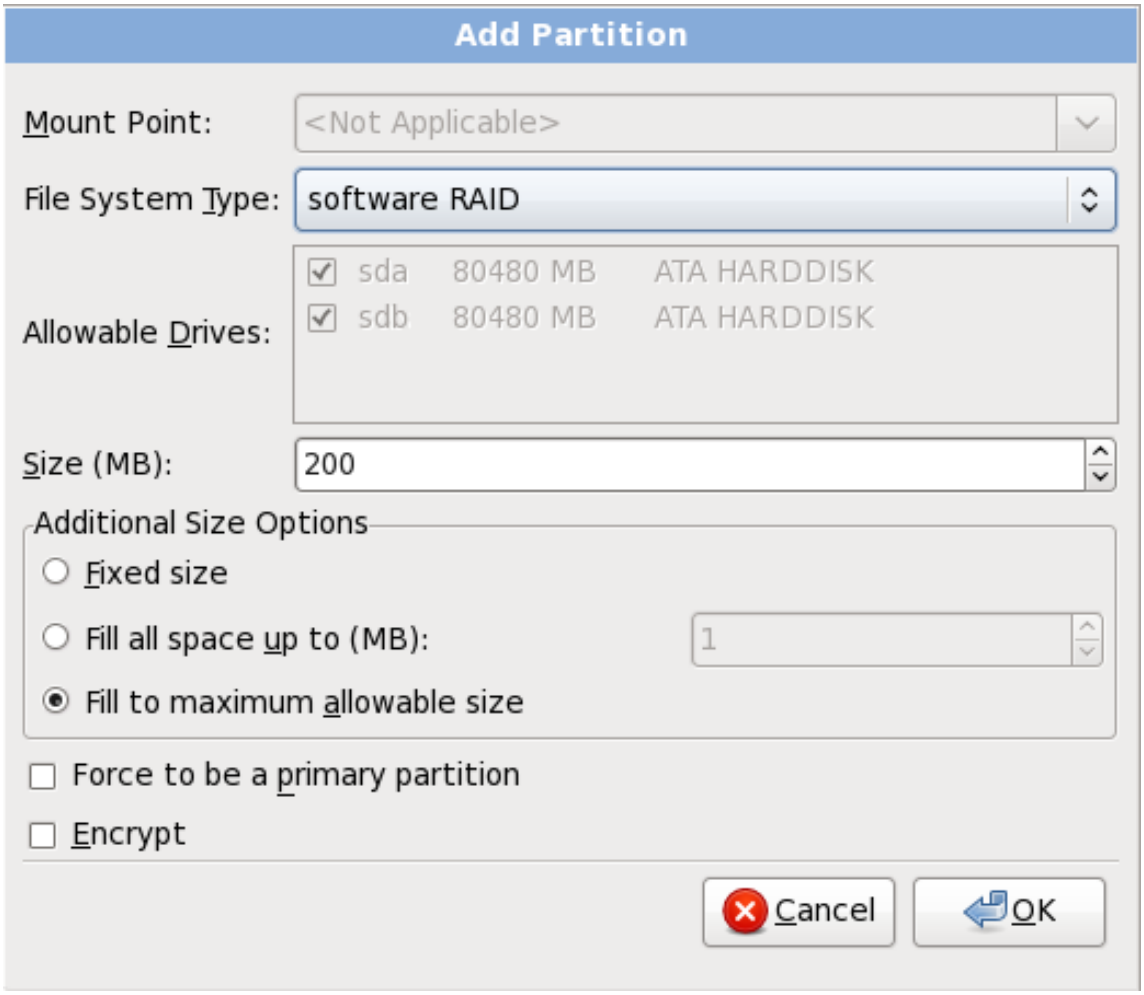


Figure 16.42. Create a software RAID partition

RAID Device

Choose this option to construct a RAID device from two or more existing software RAID partitions. This option is available if two or more software RAID partitions have been configured.

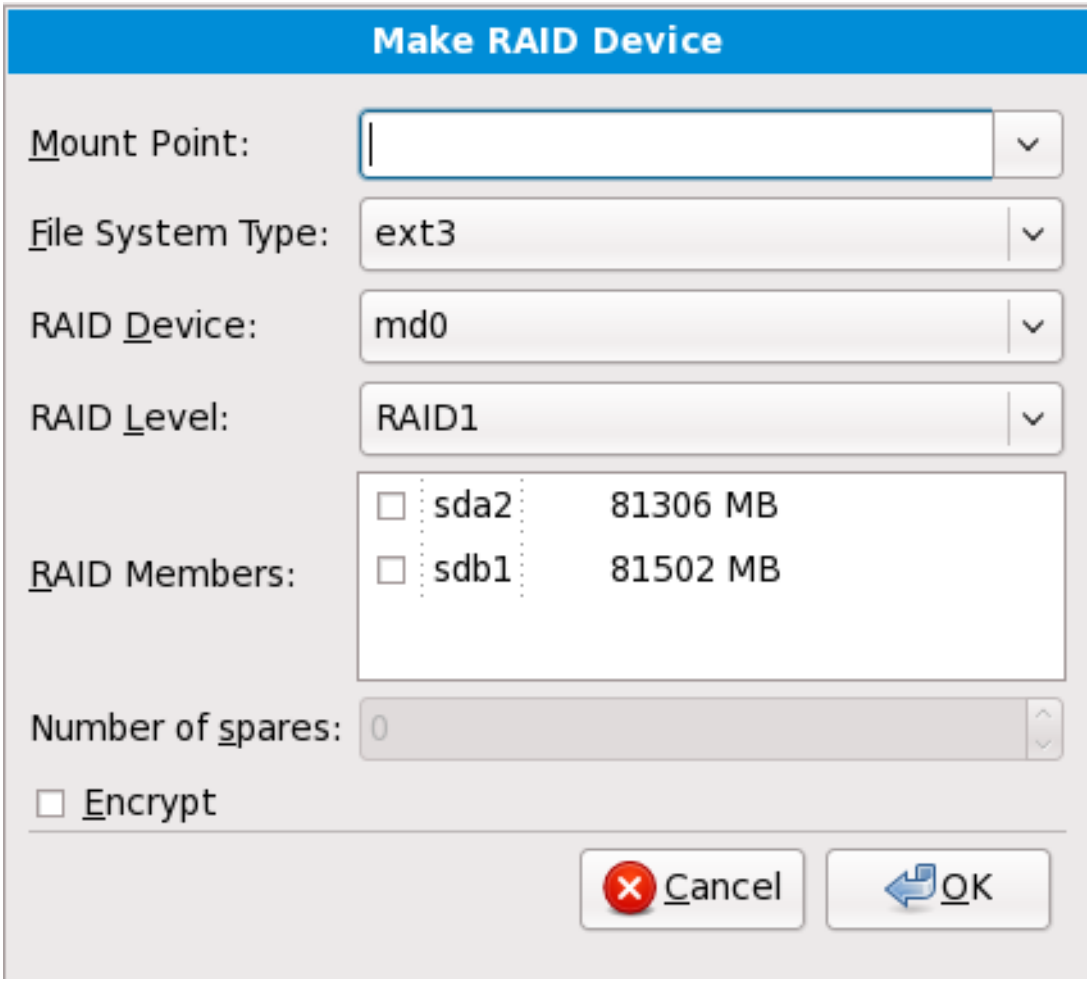


Figure 16.43. Create a RAID device

Select the file system type as for a standard partition.

Anaconda automatically suggests a name for the RAID device, but you can manually select names from **md0** to **md15**.

Click the checkboxes beside individual storage devices to include or remove them from this RAID.

The **RAID Level** corresponds to a particular type of RAID. Choose from the following options:

- **RAID 0** – distributes data across multiple storage devices. Level 0 RAIDs offer increased performance over standard partitions, and can be used to pool the storage of multiple devices into one large virtual device. Note that Level 0 RAIDS offer no redundancy and that the failure of one device in the array destroys the entire array. RAID 0 requires at least two RAID partitions.

- **RAID 1** – mirrors the data on one storage device onto one or more other storage devices. Additional devices in the array provide increasing levels of redundancy. RAID 1 requires at least two RAID partitions.

- **RAID 4** – distributes data across multiple storage devices, but uses one device in the array to store parity information that safeguards the array in case any device within the array fails. Because all parity information is stored on the one device, access to this device creates a bottleneck in the performance of the array. RAID 4 requires at least three RAID partitions.

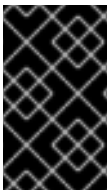
- **RAID 5** – distributes data and parity information across multiple storage devices. Level 5 RAIDs therefore offer the performance advantages of distributing data across multiple

devices, but do not share the performance bottleneck of level 4 RAIDs because the parity information is also distributed through the array. RAID 5 requires at least three RAID partitions.

- **RAID 6** – level 6 RAIDs are similar to level 5 RAIDs, but instead of storing only one set of parity data, they store two sets. RAID 6 requires at least four RAID partitions.

- **RAID 10** – level 10 RAIDs are *nested RAIDs* or *hybrid RAIDs*. Level 10 RAIDs are constructed by distributing data over mirrored sets of storage devices. For example, a level 10 RAID constructed from four RAID partitions consists of two pairs of partitions in which one partition mirrors the other. Data is then distributed across both pairs of storage devices, as in a level 0 RAID. RAID 10 requires at least four RAID partitions.

16.17.4. Create LVM Logical Volume



IMPORTANT

LVM initial set up is not available during text-mode installation. If you need to create an LVM configuration from scratch, press **Alt**+**F2** to use a different virtual console, and run the **lvm** command. To return to the text-mode installation, press **Alt**+**F1**.

Logical Volume Management (LVM) presents a simple logical view of underlying physical storage space, such as a hard drives or LUNs. Partitions on physical storage are represented as *physical volumes* that can be grouped together into *volume groups*. Each volume group can be divided into multiple *logical volumes*, each of which is analogous to a standard disk partition. Therefore, LVM logical volumes function as partitions that can span multiple physical disks.

To read more about LVM, refer to the Red Hat Enterprise Linux Deployment Guide . Note, LVM is only available in the graphical installation program.

LVM Physical Volume

Choose this option to configure a partition or device as an LVM physical volume. This option is the only choice available if your storage does not already contain LVM Volume Groups. This is the same dialog that appears when you add a standard partition – refer to Section 16.17.2, "Adding Partitions" for a description of the available options. Note, however, that **File System Type** must be set to **physical volume (LVM)**

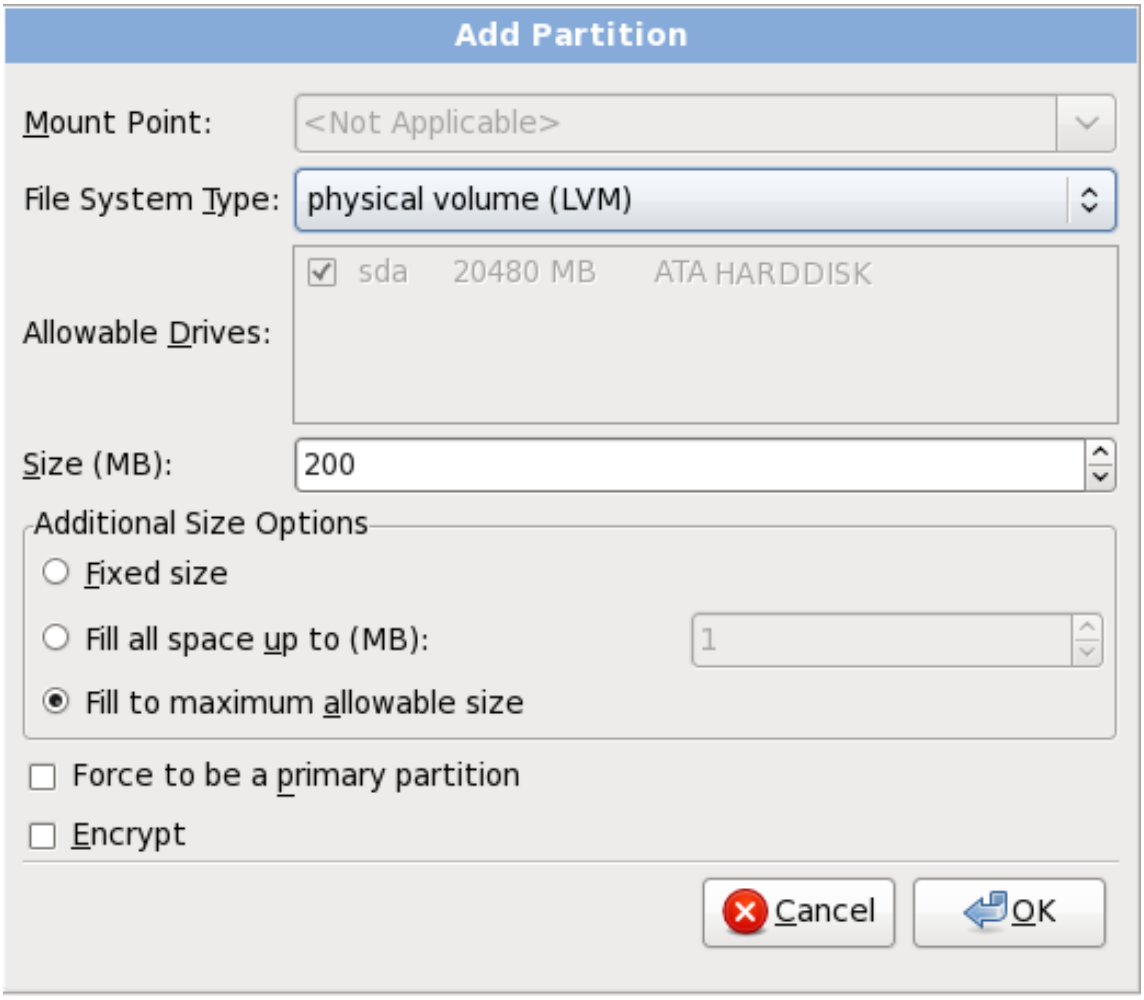


Figure 16.44. Create an LVM Physical Volume

Make LVM Volume Group

Choose this option to create LVM volume groups from the available LVM physical volumes, or to add existing logical volumes to a volume group.

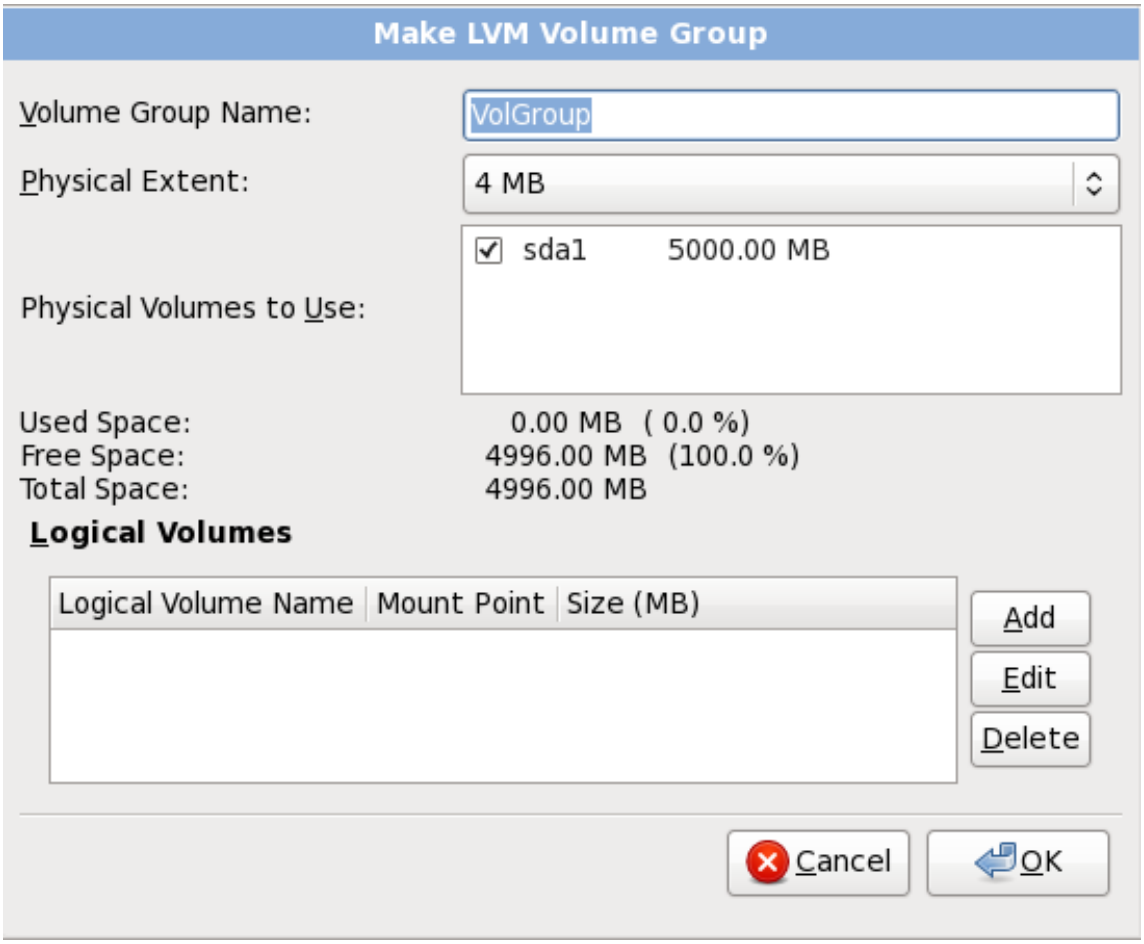


Figure 16.45. Make LVM Volume Group

To assign one or more physical volumes to a volume group, first name the volume group. Then select the physical volumes to be used in the volume group. Finally, configure logical volumes on any volume groups using the **Add**, **Edit** and **Delete** options.

You may not remove a physical volume from a volume group if doing so would leave insufficient space for that group's logical volumes. Take for example a volume group made up of two 5 GB LVM physical volume partitions, which contains an 8 GB logical volume. The installer would not allow you to remove either of the component physical volumes, since that would leave only 5 GB in the group for an 8 GB logical volume. If you reduce the total size of any logical volumes appropriately, you may then remove a physical volume from the volume group. In the example, reducing the size of the logical volume to 4 GB would allow you to remove one of the 5 GB physical volumes.

Make Logical Volume

Choose this option to create an LVM logical volume. Select a mount point, file system type, and size (in MB) just as if it were a standard disk partition. You can also choose a name for the logical volume and specify the volume group to which it will belong.

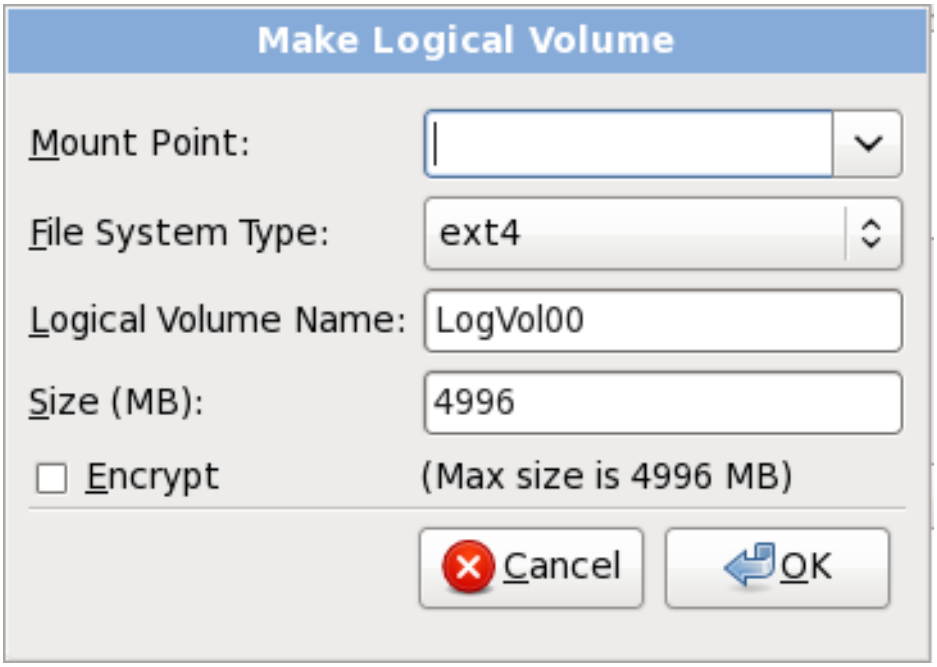


Figure 16.46. Make Logical Volume

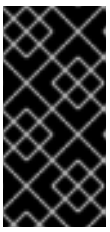
16.17.5. Recommended Partitioning Scheme

Unless you have a reason for doing otherwise, we recommend that you create the following partitions:

- A **swap** partition (at least 256 MB) – Swap partitions support virtual memory: data is written to a swap partition when there is not enough RAM to store the data your system is processing.

  In years past, the recommended amount of swap space increased linearly with the amount of RAM in the system. Modern systems often include hundreds of gigabytes of RAM, however. As a consequence, recommended swap space is considered a function of system memory workload, not system memory.

  The following table provides the recommended size of a swap partition depending on the amount of RAM in your system and whether you want sufficient memory for your system to hibernate. The recommended swap partition size is established automatically during installation. To allow for hibernation, however, you will need to edit the swap space in the custom partitioning stage.



IMPORTANT

Recommendations in the table below are especially important on systems with low memory (1 GB and less). Failure to allocate sufficient swap space on these systems may cause issues such as instability or even render the installed system unbootable.

Table 16.2. Recommended System Swap Space

Amount of RAM in the system	Recommended swap space	Recommended swap space if allowing for hibernation
⩽ 2GB	2 times the amount of RAM	3 times the amount of RAM

Amount of RAM in the system	Recommended swap space	Recommended swap space if allowing for hibernation
> 2GB – 8GB	Equal to the amount of RAM	2 times the amount of RAM
> 8GB – 64GB	At least 4 GB	1.5 times the amount of RAM
> 64GB	At least 4 GB	Hibernation not recommended

At the border between each range listed above (for example, a system with 2GB, 8GB, or 64GB 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 may lead to better performance.

Note that distributing swap space over multiple storage devices — particularly on systems with fast drives, controllers and interfaces — also improves swap space performance.

NOTE

Swap space size recommendations issued for Red Hat Enterprise Linux 6.0, 6.1, and 6.2 differed from the current recommendations, which were first issued with the release of Red Hat Enterprise Linux 6.3 in June 2012 and did not account for hibernation space. Automatic installations of these earlier versions of Red Hat Enterprise Linux 6 still generate a swap space in line with these superseded recommendations. However, manually selecting a swap space size in line with the newer recommendations issued for Red Hat Enterprise Linux 6.3 is advisable for optimal performance.

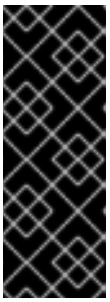
- A PReP boot partition on the first partition of the hard drive — the PReP boot partition contains the **Yaboot** boot loader (which allows other Power Systems servers to boot Red Hat Enterprise Linux). Unless you plan to boot from a network source, you must have a PReP boot partition to boot Red Hat Enterprise Linux.

  For IBM System p users: The PReP boot partition should be between 4-8 MB, not to exceed 10 MB.

- A /**boot**/ partition (250 MB) — the partition mounted on /**boot**/ contains the operating system kernel (which allows your system to boot Red Hat Enterprise Linux), along with files used during the bootstrap process. Due to the limitations of most PC firmware, creating a small partition to hold these is a good idea. For most users, a 250 MB boot partition is sufficient.

WARNING

If you have a RAID card, be aware that Red Hat Enterprise Linux 6.9 does not support setting up hardware RAID on an IPR card. You can boot the standalone diagnostics CD prior to installation to create a RAID array and then install to that RAID array.

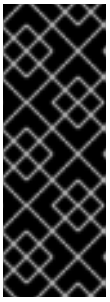


IMPORTANT

The **/boot** and / (root) partition in Red Hat Enterprise Linux 6.9 can only use the ext2, ext3, and ext4 (recommended) file systems. You cannot use any other file system for this partition, such as Btrfs, XFS, or VFAT. Other partitions, such as **/home**, can use any supported file system, including Btrfs and XFS (if available). See the following article on the Red Hat Customer Portal for additional information: https://access.redhat.com/solutions/667273.

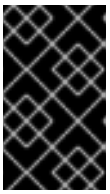
- A **root** partition (3.0 GB - 5.0 GB) — this is where " /" (the root directory) is located. In this setup, all files (except those stored in **/boot**) are on the root partition.

  A 3.0 GB partition allows you to install a minimal installation, while a 5.0 GB root partition lets you perform a full installation, choosing all package groups.



IMPORTANT

The **/boot** and / (root) partition in Red Hat Enterprise Linux 6.9 can only use the ext2, ext3, and ext4 (recommended) file systems. You cannot use any other file system for this partition, such as Btrfs, XFS, or VFAT. Other partitions, such as **/home**, can use any supported file system, including Btrfs and XFS (if available). See the following article on the Red Hat Customer Portal for additional information: https://access.redhat.com/solutions/667273.



IMPORTANT

The / (or root) partition is the top of the directory structure. The **/root** directory (sometimes pronounced "slash-root") is the home directory of the user account for system administration.

WARNING

The **PackageKit** update software downloads updated packages to **/var/cache/yum/** by default. If you partition the system manually, and create a separate **/var/** partition, be sure to create the partition large enough (3.0 GB or more) to download package updates.

16.18. WRITE CHANGES TO DISK

The installer prompts you to confirm the partitioning options that you selected. Click **Write changes to disk** to allow the installer to partition your hard drive and install Red Hat Enterprise Linux.

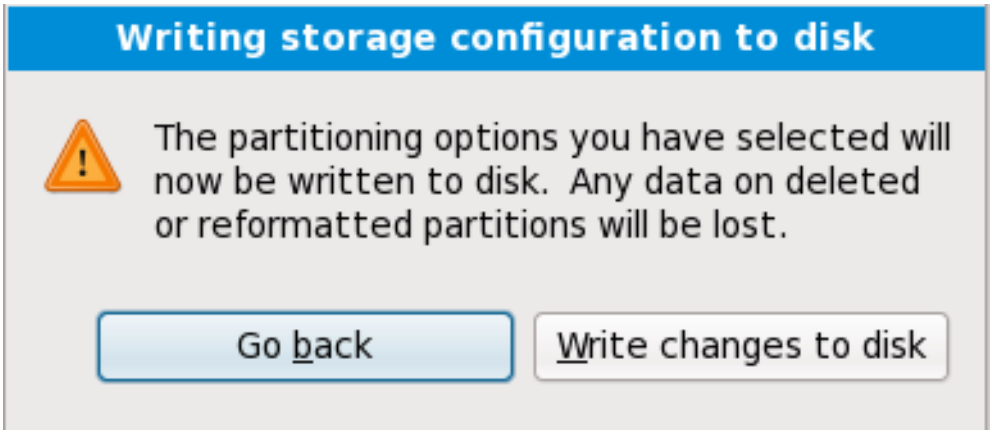


Figure 16.47. Writing storage configuration to disk

If you are certain that you want to proceed, click **Write changes to disk**.

WARNING

Up to this point in the installation process, the installer has made no lasting changes to your computer. When you click **Write changes to disk**, the installer will allocate space on your hard drive and start to transfer Red Hat Enterprise Linux into this space. Depending on the partitioning option that you chose, this process might include erasing data that already exists on your computer.

To revise any of the choices that you made up to this point, click **Go back**. To cancel installation completely, switch off your computer.

After you click **Write changes to disk**, allow the installation process to complete. If the process is interrupted (for example, by you switching off or resetting the computer, or by a power outage) you will probably not be able to use your computer until you restart and complete the Red Hat Enterprise Linux installation process, or install a different operating system.

16.19. PACKAGE GROUP SELECTION

Now that you have made most of the choices for your installation, you are ready to confirm the default package selection or customize packages for your system.

The **Package Installation Defaults** screen appears and details the default package set for your Red Hat Enterprise Linux installation. This screen varies depending on the version of Red Hat Enterprise Linux you are installing.

IMPORTANT

If you install Red Hat Enterprise Linux in text mode, you cannot make package selections. The installer automatically selects packages only from the base and core groups. These packages are sufficient to ensure that the system is operational at the end of the installation process, ready to install updates and new packages. To change the package selection, complete the installation, then use the **Add/Remove Software** application to make desired changes.

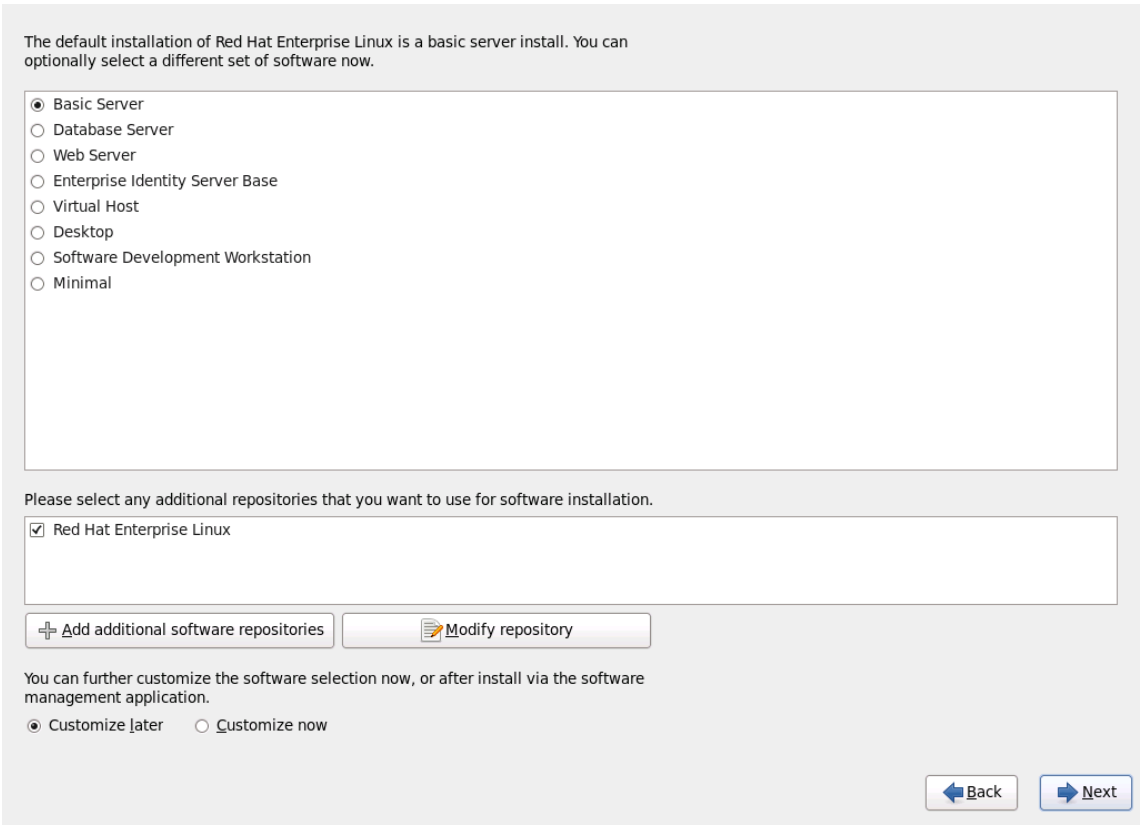


Figure 16.48. Package Group Selection

By default, the Red Hat Enterprise Linux installation process loads a selection of software that is suitable for a system deployed as a basic server. Note that this installation does not include a graphical environment. To include a selection of software suitable for other roles, click the radio button that corresponds to one of the following options:

Basic Server

This option provides a basic installation of Red Hat Enterprise Linux for use on a server.

Database Server

This option provides the **MySQL** and **PostgreSQL** databases.

Web server

This option provides the **Apache** web server.

Enterprise Identity Server Base

This option provides **OpenLDAP** and **Enterprise Identity Management** (IPA) to create an identity and authentication server.

Virtual Host

This option provides the **KVM** and **Virtual Machine Manager** tools to create a host for virtual machines.

Desktop

This option provides the **OpenOffice.org** productivity suite, graphical tools such as the **GIMP**, and multimedia applications.

Software Development Workstation

This option provides the necessary tools to compile software on your Red Hat Enterprise Linux system.

Minimal

This option provides only the packages essential to run Red Hat Enterprise Linux. A minimal installation provides the basis for a single-purpose server or desktop appliance and maximizes performance and security on such an installation.

WARNING

Minimal installation currently does not configure the firewall (**iptables**/**ip6tables**) by default because the authconfig and system-config-firewall-base packages are missing from the selection. To work around this issue, you can use a Kickstart file to add these packages to your selection. See the Red Hat Customer Portal for details about the workaround, and Chapter 32, *Kickstart Installations* for information about Kickstart files.

If you do not use the workaround, the installation will complete successfully, but no firewall will be configured, presenting a security risk.

If you choose to accept the current package list, skip ahead to Section 16.20, "Installing Packages".

To select a component, click on the checkbox beside it (refer to Figure 16.48, "Package Group Selection").

To customize your package set further, select the **Customize now** option on the screen. Clicking **Next** takes you to the **Package Group Selection** screen.

16.19.1. Installing from Additional Repositories

You can define additional *repositories* to increase the software available to your system during installation. A repository is a network location that stores software packages along with *metadata* that describes them. Many of the software packages used in Red Hat Enterprise Linux require other software to be installed. The installer uses the metadata to ensure that these requirements are met for every piece of software you select for installation.

The **Red Hat Enterprise Linux** repository is automatically selected for you. It contains the complete collection of software that was released as Red Hat Enterprise Linux 6.9, with the various pieces of software in their versions that were current at the time of release.

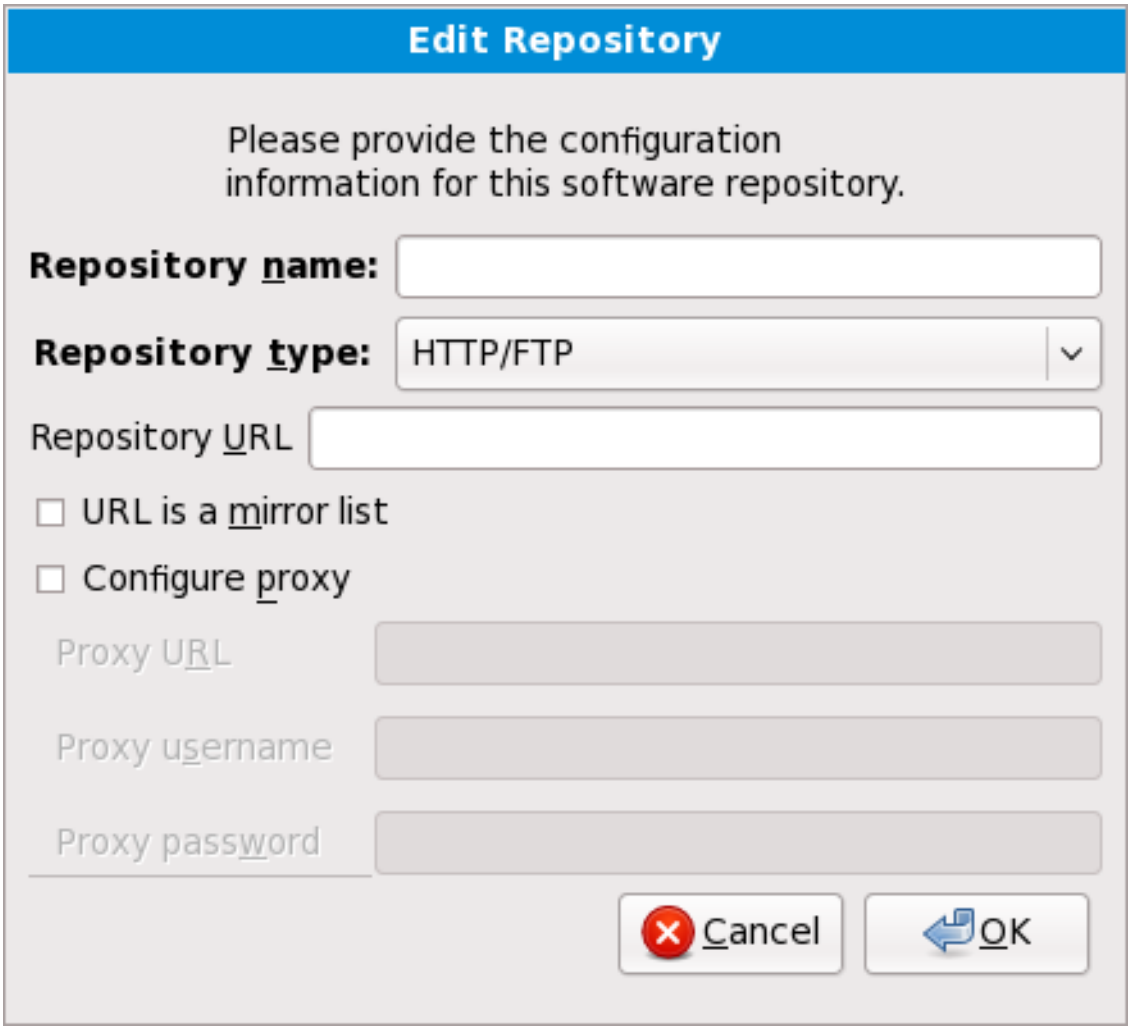


Figure 16.49. Adding a software repository

To include software from extra *repositories*, select **Add additional software repositories** and provide the location of the repository.

To edit an existing software repository location, select the repository in the list and then select **Modify repository**.

If you change the repository information during a non-network installation, such as from a Red Hat Enterprise Linux DVD, the installer prompts you for network configuration information.

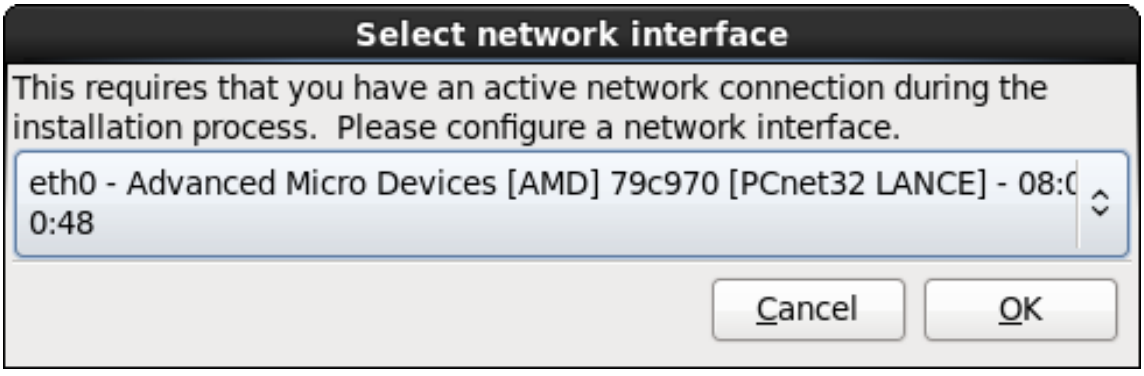


Figure 16.50. Select network interface

1. Select an interface from the drop-down menu.
2. Click **OK**.

Anaconda then starts **NetworkManager** to allow you to configure the interface.

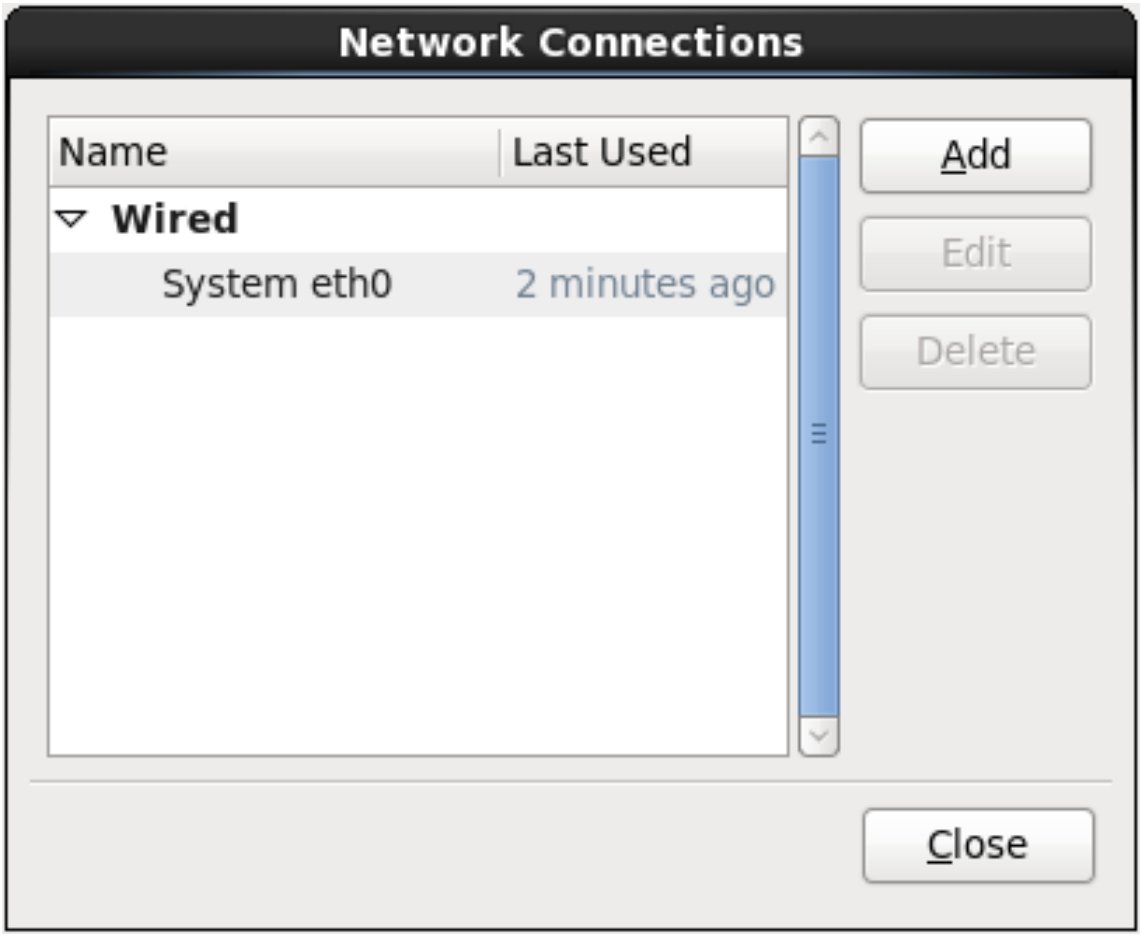


Figure 16.51. Network Connections

For details of how to use **NetworkManager**, refer to Section 16.9, "Setting the Hostname"

If you select **Add additional software repositories**, the **Edit repository** dialog appears. Provide a **Repository name** and the **Repository URL** for its location.

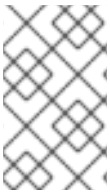
Once you have located a mirror, to determine the URL to use, find the directory on the mirror that *contains* a directory named **repodata**.

Once you provide information for an additional repository, the installer reads the package metadata over the network. Software that is specially marked is then included in the package group selection system.

WARNING

If you choose **Back** from the package selection screen, any extra repository data you may have entered is lost. This allows you to effectively cancel extra repositories. Currently there is no way to cancel only a single repository once entered.

16.19.2. Customizing the Software Selection



NOTE

Your Red Hat Enterprise Linux system automatically supports the language that you selected at the start of the installation process. To include support for additional languages, select the package group for those languages from the **Languages** category.

NOTE

Users who want support for developing or running 64-bit applications are encouraged to select the **Compatibility Arch Support** and **Compatibility Arch Development Support** packages to install architecture specific support for their systems.

Select **Customize now** to specify the software packages for your final system in more detail. This option causes the installation process to display an additional customization screen when you select **Next**.

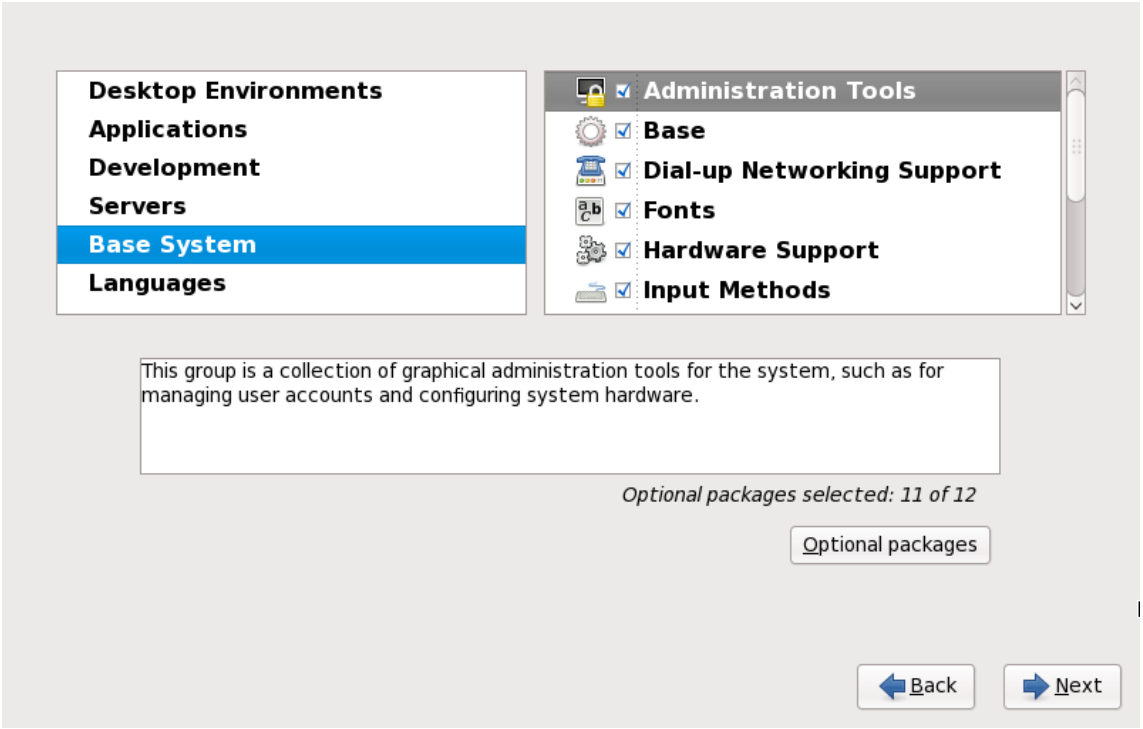


Figure 16.52. Package Group Details

Red Hat Enterprise Linux divides the included software into *package groups*. For ease of use, the package selection screen displays these groups as categories.

You can select package groups, which group components together according to function (for example, **X Window System** and **Editors**), individual packages, or a combination of the two.

To view the package groups for a category, select the category from the list on the left. The list on the right displays the package groups for the currently selected category.

To specify a package group for installation, select the check box next to the group. The box at the bottom of the screen displays the details of the package group that is currently highlighted. *None* of the packages from a group will be installed unless the check box for that group is selected.

If you select a package group, Red Hat Enterprise Linux automatically installs the base and mandatory packages for that group. To change which optional packages within a selected group will be installed, select the **Optional Packages** button under the description of the group. Then use the check box next to an individual package name to change its selection.

In the package selection list on the right, you can use the context menu as a shortcut to select or de-select base and mandatory packages or all optional packages.

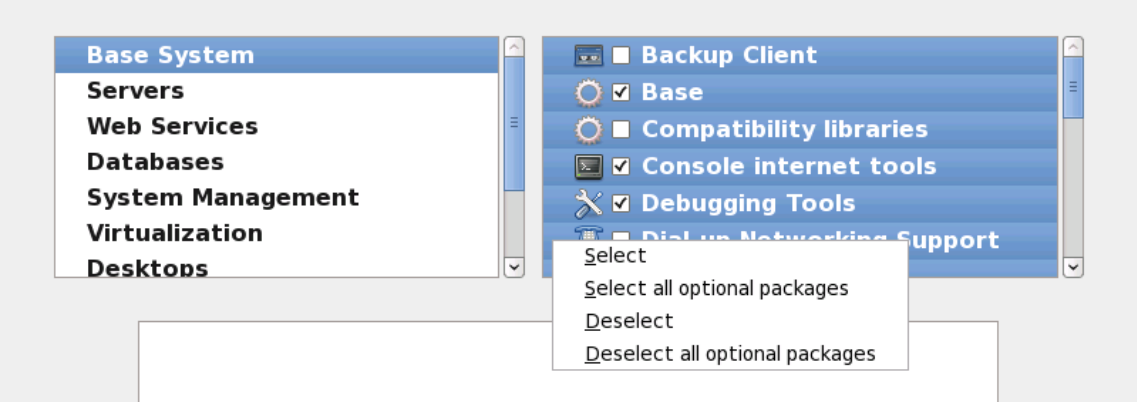


Figure 16.53. Package Selection List Context Menu

After you choose the desired packages, select **Next** to proceed. The installer checks your selection, and automatically adds any extra packages required to use the software you selected. When you have finished selecting packages, click **Close** to save your optional package selections and return to the main package selection screen.

The packages that you select are not permanent. After you boot your system, use the **Add/Remove Software** tool to either install new software or remove installed packages. To run this tool, from the main menu, select **System → Administration → Add/Remove Software**. The Red Hat Enterprise Linux software management system downloads the latest packages from network servers, rather than using those on the installation discs.

16.19.2.1. Core Network Services

All Red Hat Enterprise Linux installations include the following network services:

- centralized logging through syslog
- email through SMTP (Simple Mail Transfer Protocol)
- network file sharing through NFS (Network File System)
- remote access through SSH (Secure SHell)
- resource advertising through mDNS (multicast DNS)

The default installation also provides:

- network file transfer through HTTP (HyperText Transfer Protocol)
- printing through CUPS (Common UNIX Printing System)
- remote desktop access through VNC (Virtual Network Computing)

Some automated processes on your Red Hat Enterprise Linux system use the email service to send reports and messages to the system administrator. By default, the email, logging, and printing services

do not accept connections from other systems. Red Hat Enterprise Linux installs the NFS sharing, HTTP, and VNC components without enabling those services.

You may configure your Red Hat Enterprise Linux system after installation to offer email, file sharing, logging, printing and remote desktop access services. The SSH service is enabled by default. You may use NFS to access files on other systems without enabling the NFS sharing service.

16.20. INSTALLING PACKAGES

At this point there is nothing left for you to do until all the packages have been installed. How quickly this happens depends on the number of packages you have selected and your computer's speed.

Depending on the available resources, you might see the following progress bar while the installer resolves dependencies of the packages you selected for installation:

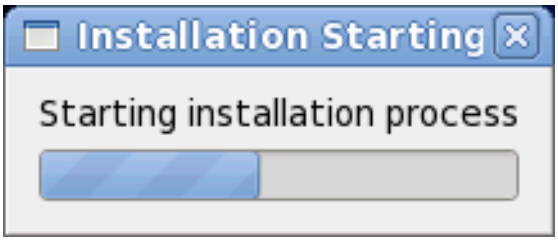


Figure 16.54. Starting installation

During installation of the selected packages and their dependencies, you see the following progress bar:

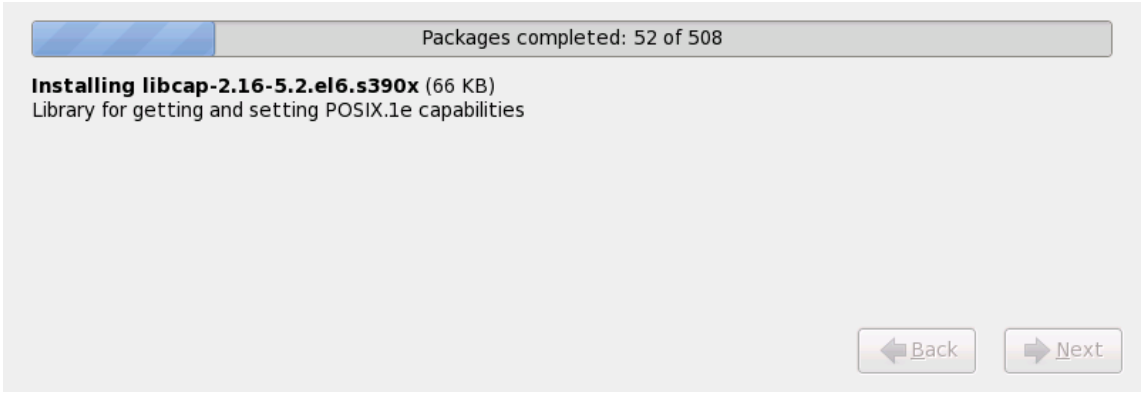


Figure 16.55. Packages completed

16.21. INSTALLATION COMPLETE

Congratulations! Your Red Hat Enterprise Linux installation is now complete!

The installation program prompts you to prepare your system for reboot. Remember to remove any installation media if it is not ejected automatically upon reboot.

After your computer's normal power-up sequence has completed, Red Hat Enterprise Linux loads and starts. By default, the start process is hidden behind a graphical screen that displays a progress bar. Eventually, a **login:** prompt or a GUI login screen (if you installed the X Window System and chose to start X automatically) appears.

The first time you start your Red Hat Enterprise Linux system in run level 5 (the graphical run level), the **FirstBoot** tool appears, which guides you through the Red Hat Enterprise Linux configuration. Using this tool, you can set your system time and date, install software, register your machine with Red Hat

Network, and more. **FirstBoot** lets you configure your environment at the beginning, so that you can get started using your Red Hat Enterprise Linux system quickly.

Chapter 34, *Firstboot* will guide you through the configuration process.

---

[7] A root password is the administrative password for your Red Hat Enterprise Linux system. You should only log in as root when needed for system maintenance. The root account does not operate within the restrictions placed on normal user accounts, so changes made as root can have implications for your entire system.

[8] The **fsck** application is used to check the file system for metadata consistency and optionally repair one or more Linux file systems.

CHAPTER 17. TROUBLESHOOTING INSTALLATION ON AN IBM POWER SYSTEMS SERVER

This section discusses some common installation problems and their solutions.

For debugging purposes, **anaconda** logs installation actions into files in the **/tmp** directory. These files include:

/tmp/anaconda.log

general **anaconda** messages

/tmp/program.log

all external programs run by **anaconda**

/tmp/storage.log

extensive storage module information

/tmp/yum.log

yum package installation messages

/tmp/syslog

hardware-related system messages

If the installation fails, the messages from these files are consolidated into **/tmp/anaconda-tb-*identifier***, where *identifier* is a random string.

You may also find the IBM Online Alert Section for System p useful. It is located at:

> http://www14.software.ibm.com/webapp/set2/sas/f/lopdiags/info/LinuxAlerts.html

All of the files above reside in the installer's ramdisk and are thus volatile. To make a permanent copy, copy those files to another system on the network using **scp** on the installation image (not the other way round).

17.1. YOU ARE UNABLE TO BOOT RED HAT ENTERPRISE LINUX

17.1.1. Is Your System Displaying Signal 11 Errors?

A signal 11 error, commonly known as a *segmentation fault*, means that the program accessed a memory location that was not assigned to it. A signal 11 error may be due to a bug in one of the software programs that is installed, or faulty hardware.

If you receive a fatal signal 11 error during your installation, it is probably due to a hardware error in memory on your system's bus. Like other operating systems, Red Hat Enterprise Linux places its own demands on your system's hardware. Some of this hardware may not be able to meet those demands, even if they work properly under another OS.

Ensure that you have the latest installation updates and images. Review the online errata to see if newer versions are available. If the latest images still fail, it may be due to a problem with your hardware. Commonly, these errors are in your memory or CPU-cache. A possible solution for this error is turning

off the CPU-cache in the BIOS, if your system supports this. You could also try to swap your memory around in the motherboard slots to check if the problem is either slot or memory related.

Another option is to perform a media check on your installation DVD. **Anaconda**, the installation program, has the ability to test the integrity of the installation media. It works with the DVD, hard drive ISO, and NFS ISO installation methods. Red Hat recommends that you test all installation media before starting the installation process, and before reporting any installation-related bugs (many of the bugs reported are actually due to improperly-burned DVDs). To use this test, type the following command at the **boot:** or **yaboot:** prompt:

```
linux mediacheck
```

For more information concerning signal 11 errors, refer to:

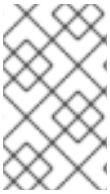
```
http://www.bitwizard.nl/sig11/
```

17.2. TROUBLE BEGINNING THE INSTALLATION

17.2.1. Problems with Booting into the Graphical Installation

There are some video cards that have trouble booting into the graphical installation program. If the installation program does not run using its default settings, it tries to run in a lower resolution mode. If that still fails, the installation program attempts to run in text mode.

One possible solution is to try using the **resolution=** boot option. Refer to Chapter 28, *Boot Options* for more information.



NOTE

To disable frame buffer support and allow the installation program to run in text mode, try using the **nofb** boot option. This command may be necessary for accessibility with some screen reading hardware.

17.3. TROUBLE DURING THE INSTALLATION

17.3.1. The "No devices found to install Red Hat Enterprise Linux" Error Message

If you receive an error message stating **No devices found to install Red Hat Enterprise Linux**, there is probably a SCSI controller that is not being recognized by the installation program.

Check your hardware vendor's website to determine if a driver disk image is available that fixes your problem. For more general information on driver disks, refer to Chapter 13, *Updating Drivers During Installation on IBM Power Systems Servers*.

You can also refer to the *Red Hat Hardware Compatibility List* , available online at:

https://hardware.redhat.com/

17.3.2. Saving Traceback Messages

If **anaconda** encounters an error during the graphical installation process, it presents you with a crash reporting dialog box:

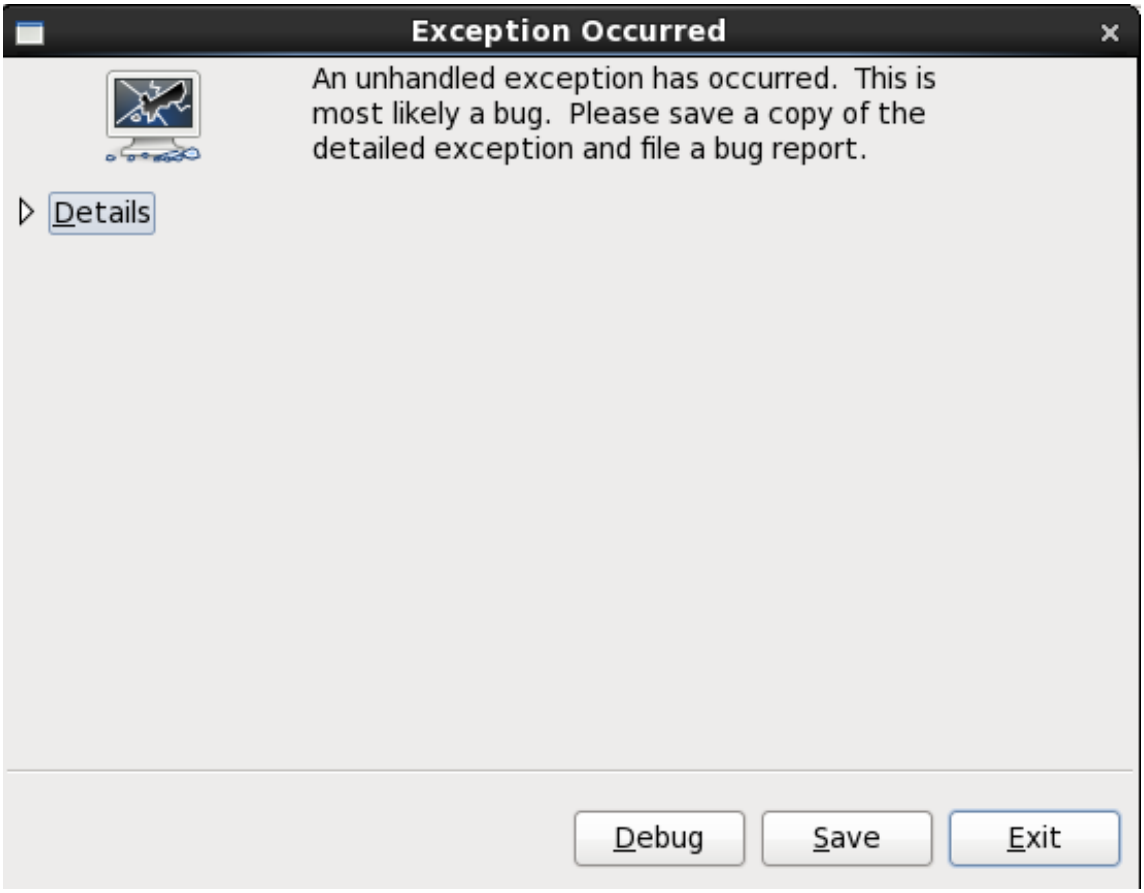


Figure 17.1. The Crash Reporting Dialog Box

Details

shows you the details of the error:

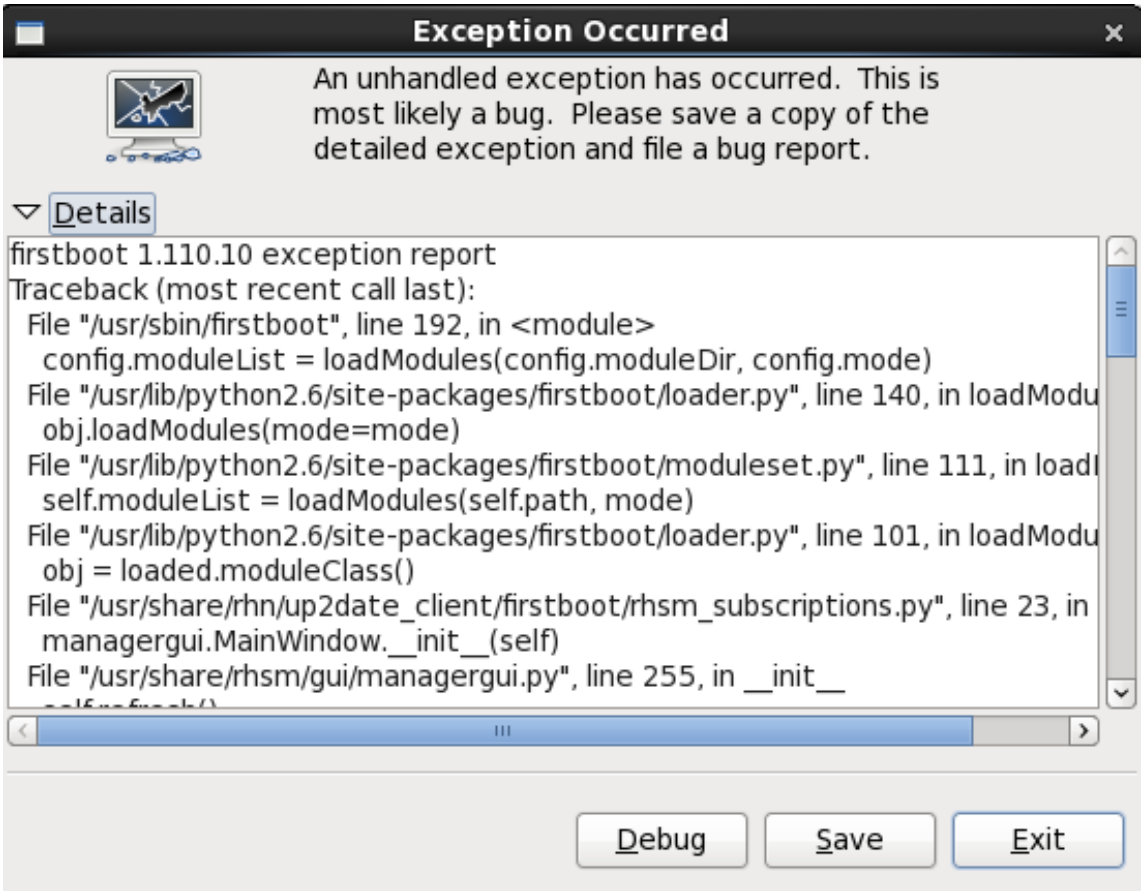


Figure 17.2. Details of the Crash

Save

saves details of the error locally or remotely:

Exit

exits the installation process.

If you select **Save** from the main dialog, you can choose from the following options:

Figure 17.3. Select reporter

Logger

saves details of the error as a log file to the local hard drive at a specified location.

Red Hat Customer Support

submits the crash report to Customer Support for assistance.

Report uploader

uploads a compressed version of the crash report to Bugzilla or a URL of your choice.

Before submitting the report, click **Preferences** to specify a destination or provide authentication details. Select the reporting method you need to configure and click **Configure Event**.

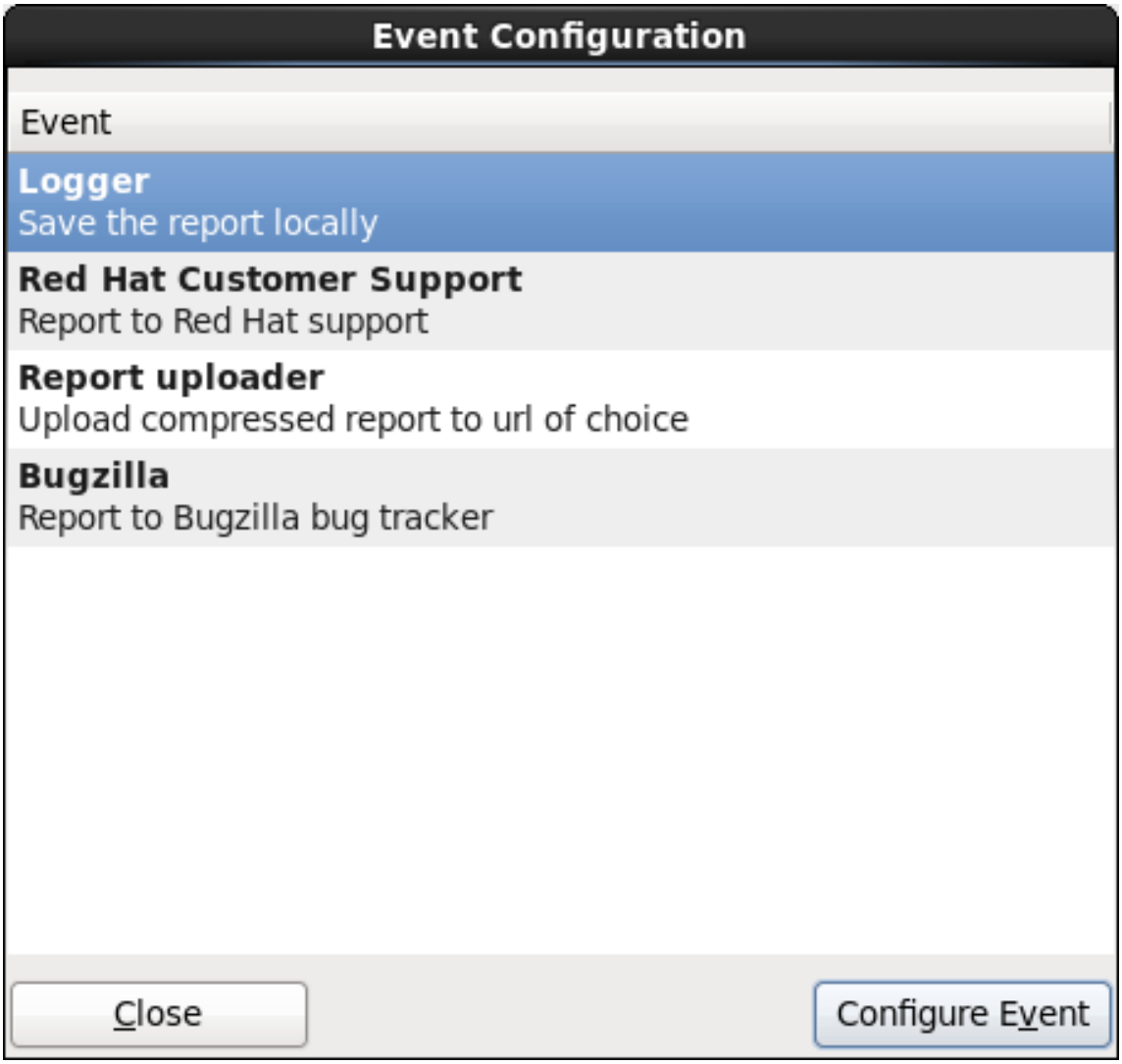


Figure 17.4. Configure reporter preferences

Logger

Specify a path and a filename for the log file. Check **Append** if you are adding to an existing log file.

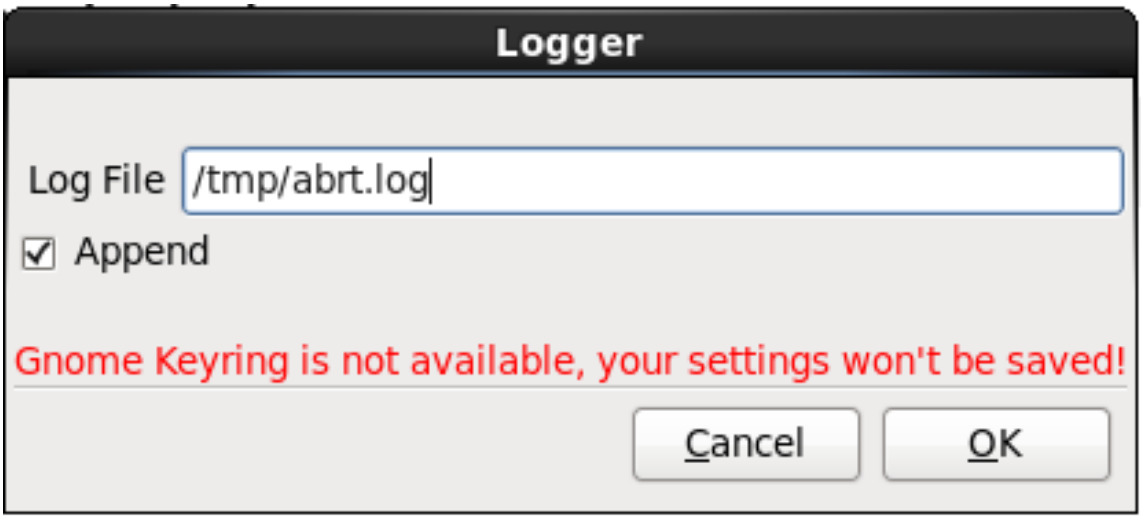


Figure 17.5. Specify local path for log file

Red Hat Customer Support

Enter your Red Hat Network username and password so your report reaches Customer Support and is linked with your account. The URL is prefilled and **Verify SSL** is checked by default.

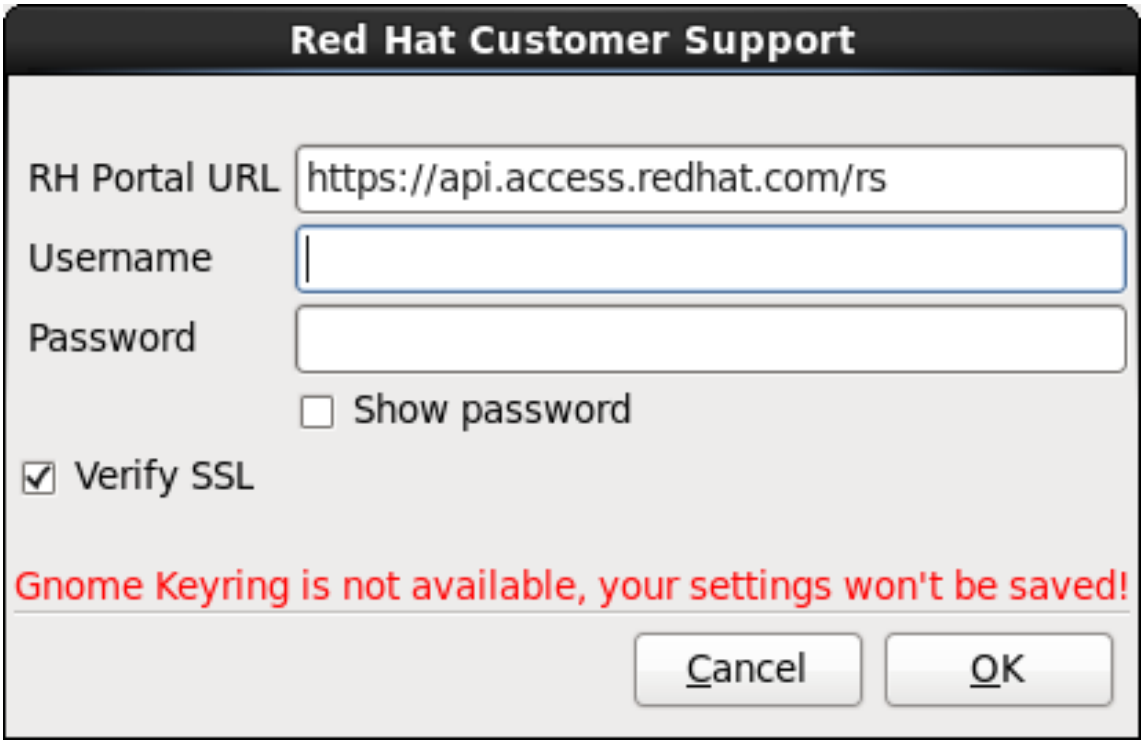


Figure 17.6. Enter Red Hat Network authentication details

Report uploader

Specify a URL for uploading a compressed version of the crash report.

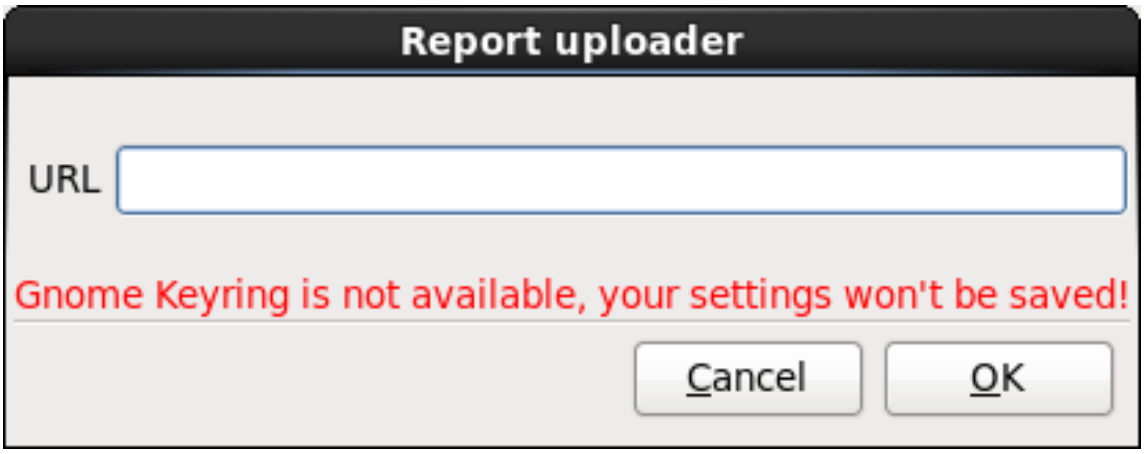


Figure 17.7. Enter URL for uploading crash report

Bugzilla

Enter your Bugzilla username and password to lodge a bug with Red Hat's bug-tracking system using the crash report. The URL is prefilled and **Verify SSL** is checked by default.

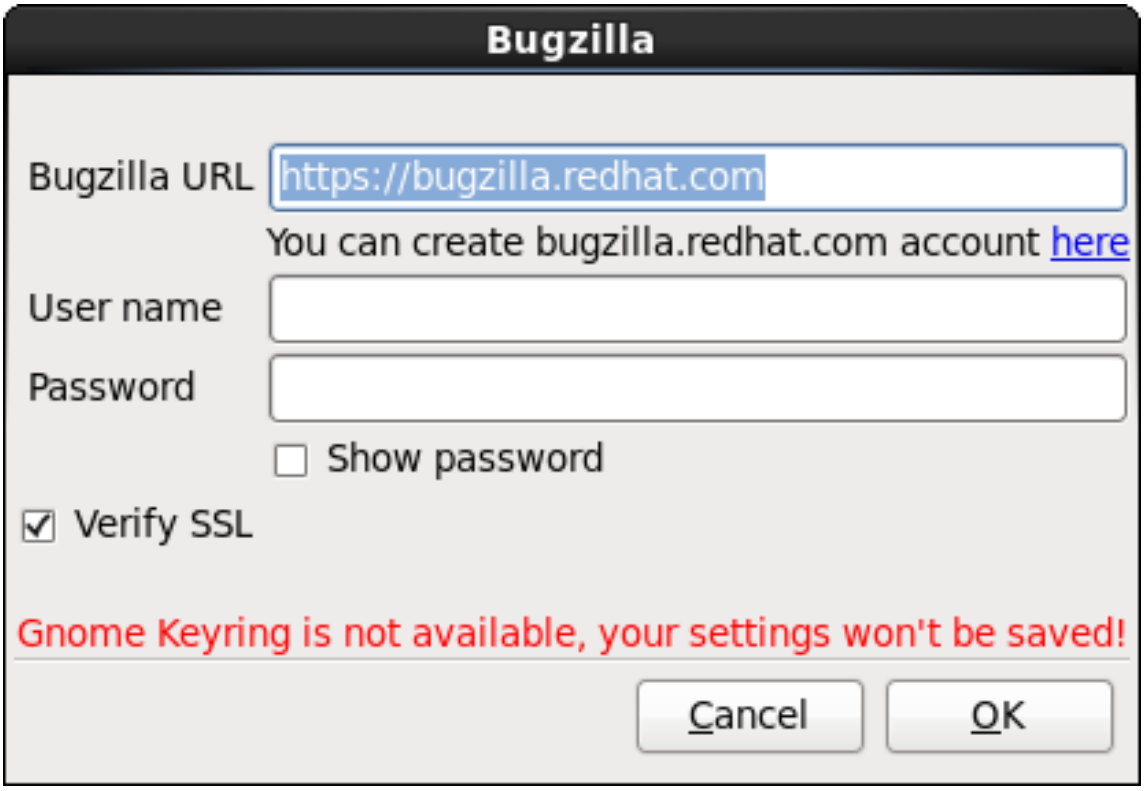


Figure 17.8. Enter Bugzilla authentication details

Once you have entered your preferences, click **OK** to return to the report selection dialog. Select how you would like to report the problem and then click **Forward**.

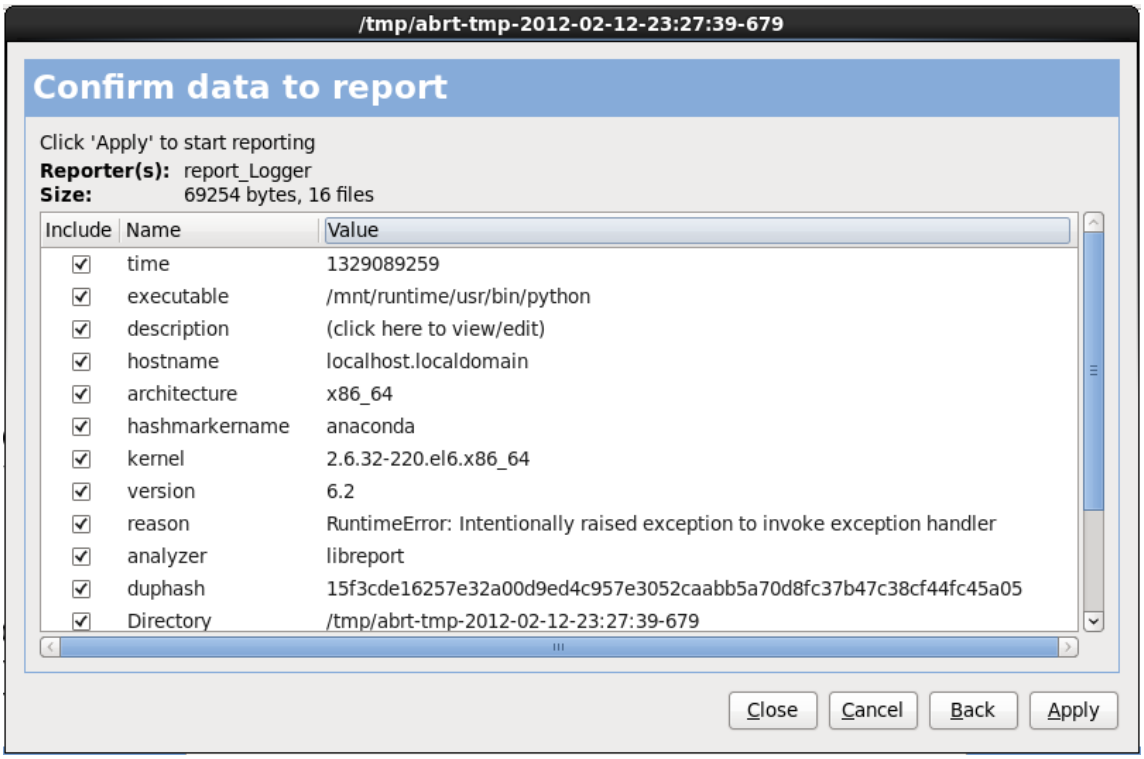


Figure 17.9. Confirm report data

You can now customize the report by checking and unchecking the issues that will be included. When finished, click **Apply**.

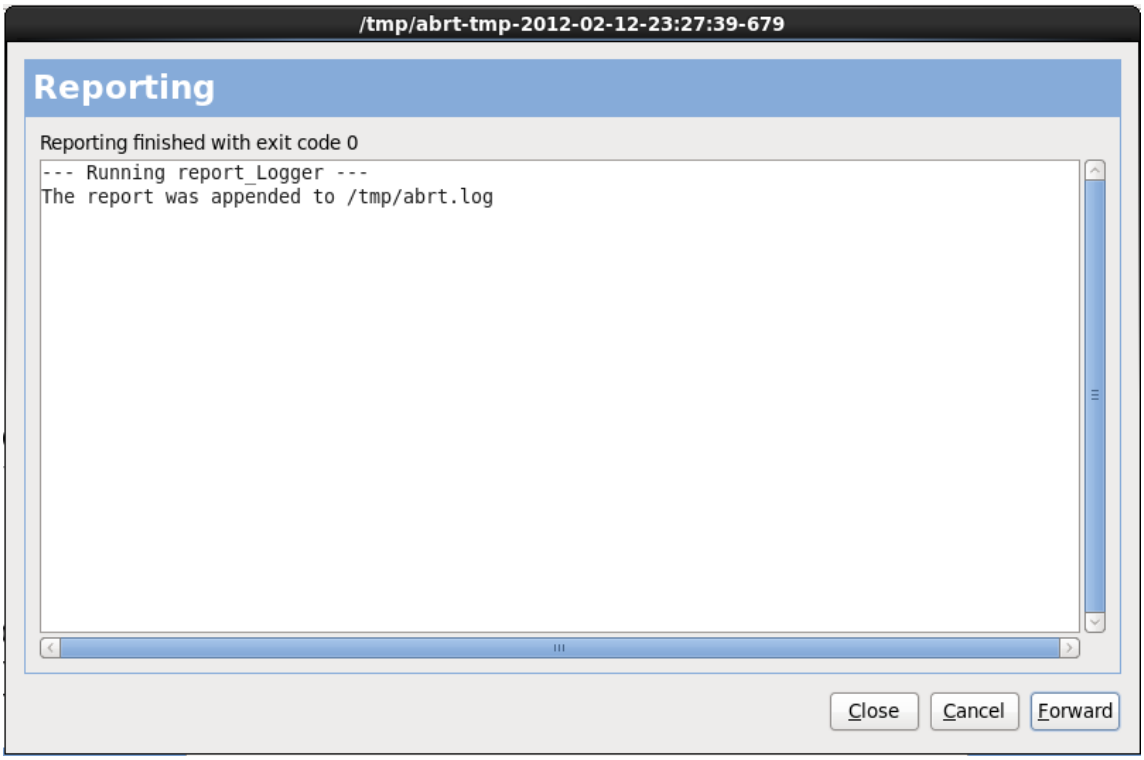


Figure 17.10. Report in progress

This screen displays the outcome of the report, including any errors in sending or saving the log. Click **Forward** to proceed.

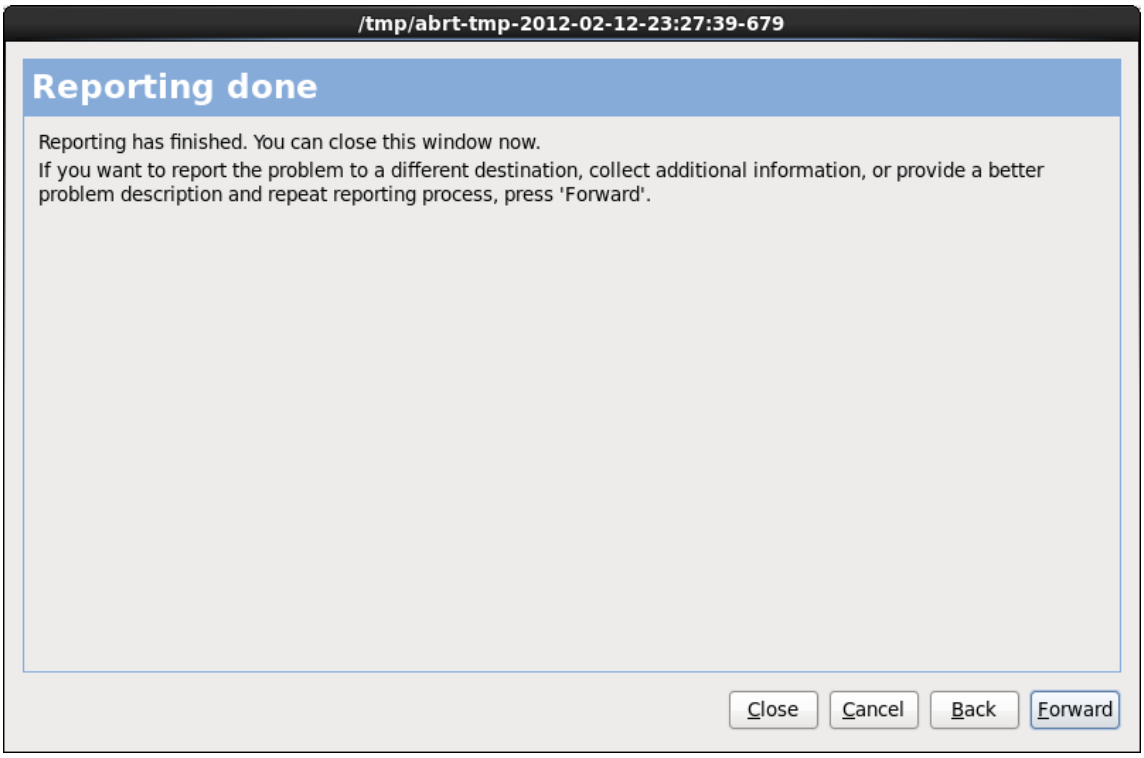
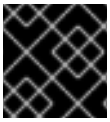


Figure 17.11. Reporting done

Reporting is now complete. Click **Forward** to return to the report selection dialog. You can now make another report, or click **Close** to exit the reporting utility and then **Exit** to close the installation process.



IMPORTANT

This information does not apply to users of headless IBM System p systems.

17.3.3. Trouble with Partition Tables

If you receive an error after the **Disk Partitioning Setup** (Section 16.15, "Disk Partitioning Setup") phase of the installation saying something similar to

> The partition table on device hda was unreadable. To create new partitions it must be initialized, causing the loss of ALL DATA on this drive.

you may not have a partition table on that drive or the partition table on the drive may not be recognizable by the partitioning software used in the installation program.

No matter what type of installation you are performing, backups of the existing data on your systems should always be made.

17.3.4. Other Partitioning Problems for IBM Power Systems Users

If you create partitions manually, but cannot move to the next screen, you probably have not created all the partitions necessary for installation to proceed.

You must have the following partitions as a bare minimum:

- A / (root) partition
- A <swap> partition of type swap
- A PReP Boot partition.
- A /boot/ partition.

Refer to Section 16.17.5, "Recommended Partitioning Scheme" for more information.

NOTE

When defining a partition's type as swap, do not assign it a mount point. **Anaconda** automatically assigns the mount point for you.

17.4. PROBLEMS AFTER INSTALLATION

17.4.1. Unable to IPL from *NWSSTG

If you are experiencing difficulties when trying to IPL from *NWSSTG, you may not have created a PReP Boot partition set as active.

17.4.2. Booting into a Graphical Environment

If you have installed the X Window System but are not seeing a graphical desktop environment once you log into your system, you can start the X Window System graphical interface using the command **startx**.

Once you enter this command and press **Enter**, the graphical desktop environment is displayed.

Note, however, that this is just a one-time fix and does not change the log in process for future log ins.

To set up your system so that you can log in at a graphical login screen, you must edit one file, **/etc/inittab**, by changing just one number in the runlevel section. When you are finished, reboot the computer. The next time you log in, you are presented with a graphical login prompt.

Open a shell prompt. If you are in your user account, become root by typing the **su** command.

Now, type the following to edit the file with **gedit**.

```
gedit /etc/inittab
```

The file **/etc/inittab** opens. Within the first screen, a section of the file which looks like the following appears:

```
# Default runlevel. The runlevels used are:
#   0 - halt (Do NOT set initdefault to this)
#   1 - Single user mode
#   2 - Multiuser, without NFS (The same as 3, if you do not have networking)
#   3 - Full multiuser mode
#   4 - unused
#   5 - X11
#   6 - reboot (Do NOT set initdefault to this)
#
id:3:initdefault:
```

To change from a console to a graphical login, you should change the number in the line **id:3:initdefault:** from a **3** to a **5**.

WARNING

Change *only* the number of the default runlevel from **3** to **5**.

Your changed line should look like the following:

```
id:5:initdefault:
```

When you are satisfied with your change, save and exit the file using the **Ctrl**+**Q** keys. A window appears and asks if you would like to save the changes. Click **Save**.

The next time you log in after rebooting your system, you are presented with a graphical login prompt.

17.4.3. Problems with the X Window System (GUI)

If you are having trouble getting X (the X Window System) to start, you may not have installed it during your installation.

If you want X, you can either install the packages from the Red Hat Enterprise Linux installation media or perform an upgrade.

If you elect to upgrade, select the X Window System packages, and choose GNOME, KDE, or both, during the upgrade package selection process.

Refer to Section 35.3, "Switching to a Graphical Login" for more detail on installing a desktop environment.

17.4.4. Problems with the X Server Crashing and Non-Root Users

If you are having trouble with the X server crashing when anyone logs in, you may have a full file system (or, a lack of available hard drive space).

To verify that this is the problem you are experiencing, run the following command:

```
df -h
```

The **df** command should help you diagnose which partition is full. For additional information about **df** and an explanation of the options available (such as the **-h** option used in this example), refer to the **df** man page by typing **man df** at a shell prompt.

A key indicator is 100% full or a percentage above 90% or 95% on a partition. The **/home/** and **/tmp/** partitions can sometimes fill up quickly with user files. You can make some room on that partition by removing old files. After you free up some disk space, try running X as the user that was unsuccessful before.

17.4.5. Problems When You Try to Log In

If you did not create a user account in the **firstboot** screens, switch to a console by pressing **Ctrl**+**Alt**+**F2**, log in as root and use the password you assigned to root.

If you cannot remember your root password, boot your system as **linux single**.

Once you have booted into single user mode and have access to the **#** prompt, you must type **passwd root**, which allows you to enter a new password for root. At this point you can type **shutdown -r now** to reboot the system with the new root password.

If you cannot remember your user account password, you must become root. To become root, type **su -** and enter your root password when prompted. Then, type **passwd <username>**. This allows you to enter a new password for the specified user account.

If the graphical login screen does not appear, check your hardware for compatibility issues. The *Hardware Compatibility List* can be found at:

```
https://hardware.redhat.com/
```

17.4.6. Your Printer Does Not Work

If you are not sure how to set up your printer or are having trouble getting it to work properly, try using the **Printer Configuration Tool**.

Type the **system-config-printer** command at a shell prompt to launch the **Printer Configuration Tool**. If you are not root, it prompts you for the root password to continue.

17.4.7. Apache HTTP Server or Sendmail Stops Responding During Startup

If **Apache HTTP Server** (**httpd**) or **Sendmail** stops responding during startup, make sure the following line is in the /**etc**/**hosts** file:

```
127.0.0.1  localhost.localdomain  localhost
```

PART III. IBM SYSTEM Z ARCHITECTURE – INSTALLATION AND BOOTING

This part of the *Red Hat Enterprise Linux Installation Guide* discusses installation and booting (or *initial program load*, IPL) of Red Hat Enterprise Linux on IBM System z.

CHAPTER 18. PLANNING FOR INSTALLATION ON SYSTEM Z

18.1. PRE-INSTALLATION

Red Hat Enterprise Linux 6.9 runs on System z9 or later IBM mainframe systems.

The installation process assumes that you are familiar with the IBM System z and can set up *logical partitions* (LPARs) and z/VM guest virtual machines. For additional information on System z, refer to http://www.ibm.com/systems/z.

For installation of Red Hat Enterprise Linux on System z, Red Hat supports DASD and FCP storage devices.

Before you install Red Hat Enterprise Linux, you must decide on the following:

- Decide whether you want to run the operating system on an LPAR or as a z/VM guest operating system.
- Decide if you need swap space and if so how much. Although it is possible (and recommended) to assign enough memory to z/VM guest virtual machine and let z/VM do the necessary swapping, there are cases where the amount of required RAM is hard to predict. Such instances should be examined on a case-by-case basis. Refer to Section 23.15.5, "Recommended Partitioning Scheme".
- Decide on a network configuration. Red Hat Enterprise Linux 6.9 for IBM System z supports the following network devices:
  - Real and virtual *Open Systems Adapter* (OSA)
  - Real and virtual HiperSockets
  - *LAN channel station* (LCS) for real OSA

You require the following hardware:

- Disk space. Calculate how much disk space you need and allocate sufficient disk space on DASDs[9] or SCSI[10] disks. You require at least 2 GB for a server installation, and 5 GB if you want to install all packages. You also require disk space for any application data. After the installation, more DASD or SCSI disk partitions may be added or deleted as necessary.

  The disk space used by the newly installed Red Hat Enterprise Linux system (the Linux instance) must be separate from the disk space used by other operating systems you may have installed on your system.

  For more information about disks and partition configuration, refer to Section 23.15.5, "Recommended Partitioning Scheme".
- RAM. Acquire 1 GB (recommended) for the Linux instance. With some tuning, an instance might run with as little as 512 MB RAM.

18.2. OVERVIEW OF THE SYSTEM Z INSTALLATION PROCEDURE

You can install Red Hat Enterprise Linux on System z interactively or in unattended mode. Installation on System z differs from installation on other architectures in that it is typically performed over a network and not from a local DVD. The installation can be summarized as follows:

1. **Booting (IPL) the installer**
   Connect with the mainframe, then perform an *initial program load* (IPL), or boot, from the medium containing the installation program.

2. **Installation Phase 1**
   Set up an initial network device. This network device is then used to connect to the installation system via SSH or VNC. This gets you a full-screen mode terminal or graphical display to continue installation as on other architectures.

3. **Installation Phase 2**
   Specify which language to use, and how and where the installation program and the software packages to be installed from the repository on the Red Hat installation medium can be found.

4. **Installation Phase 3**
   Use **anaconda** (the main part of the Red Hat installation program) to perform the rest of the installation.

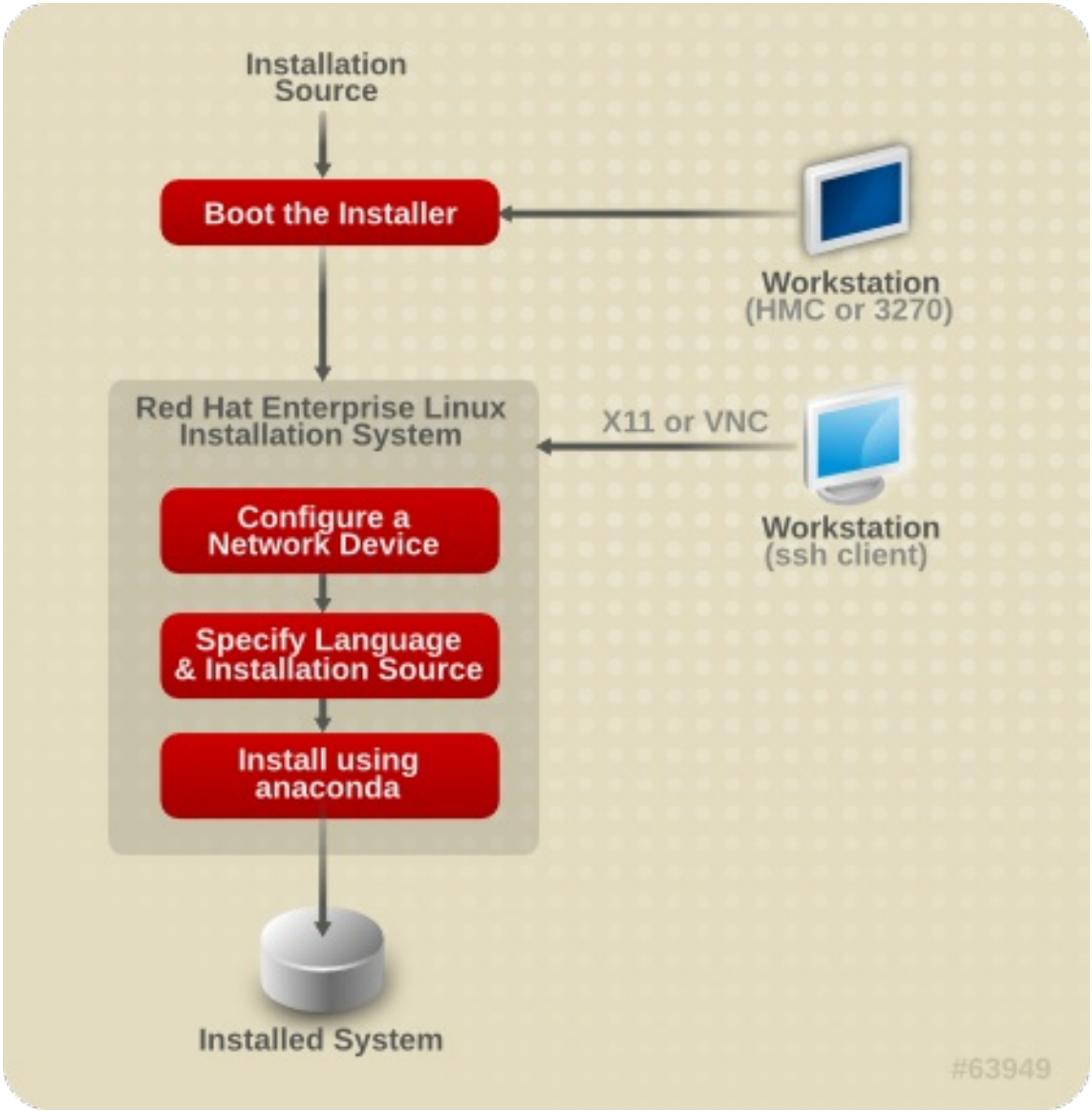


Figure 18.1. The Installation Process

18.2.1. Booting (IPL) the Installer

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. This document describes the most common methods of installing Red Hat Enterprise Linux 6.9 on System z. In general, you can use any method to boot the Linux installation system, which consists of a kernel (**kernel.img**) and initial ramdisk (**initrd.img**) with at least the parameters in **generic.prm**. The Linux installation system is also called the *installer* in this book.

The control point from where you can start the IPL process depends on the environment where your Linux is to run. If your Linux is to run as a z/VM guest operating system, the control point is the *control program* (CP) of the hosting z/VM. If your Linux is to run in LPAR mode, the control point is the mainframe's *Support Element* (SE) or an attached IBM System z *Hardware Management Console* (HMC).

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

- z/VM reader – refer to Section 20.1.1, "Using the z/VM Reader" for details.

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

- SE or HMC through a remote FTP server – refer to Section 20.2.1, "Using an FTP Server" for details.
- SE or HMC DVD – refer to Section 20.2.2, "Using the HMC or SE DVD Drive" for details

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

- DASD – refer to Section 20.1.2, "Using a Prepared DASD" for z/VM or Section 20.2.3, "Using a Prepared DASD" for LPAR
- SCSI device that is attached through an FCP channel – refer to Section 20.1.3, "Using a Prepared FCP-attached SCSI Disk" for z/VM or Section 20.2.4, "Using a Prepared FCP-attached SCSI Disk" for LPAR
- FCP-attached SCSI DVD – refer to Section 20.1.4, " Using an FCP-attached SCSI DVD Drive" for z/VM or Section 20.2.5, "Using an FCP-attached SCSI DVD Drive" for LPAR

If you use DASD and FCP-attached SCSI devices (except SCSI DVDs) as boot media, you must have a configured zipl boot loader. For more information, see the Chapter on zipl in *Linux on System z Device Drivers, Features, and Commands on Red Hat Enterprise Linux 6*.

18.2.2. Installation Phase 1

After the kernel boot, you will configure one network device. This network device is needed to complete the installation.

The interface you will use in installation phase 1 is the **linuxrc** interface, which is line-mode and text-based. (Refer to Chapter 21, *Installation Phase 1: Configuring a Network Device* .)

18.2.3. Installation Phase 2

In installation phase 2, you need to specify what language to use and where phase 3 of the installation program and the software packages to be installed from the repository on the Red Hat installation medium can be found. On System z, the installation sources are usually transferred from the DVD to a network server. Phase 3 of the installation program and the repository can be accessed in one of the following ways:

- Over the network using one of the FTP, HTTP, HTTPS, or NFS protocols. A separate network

server (FTP, HTTP, HTTPS, or NFS), which holds all the required installation sources, must be set up in advance. For details on how to set up a network server, refer to Section 19.1, "Preparing for a Network Installation".

- Hard disk (DASD or a SCSI device attached through an FCP channel). You need to set up a disk that holds the required installation sources in advance. For details, Refer to Section 19.2, "Preparing for a Hard Drive Installation".

- Through an FCP-attached SCSI DVD. This is handled automatically if booted from FCP-attached SCSI DVD.

The interface you will use in installation phase 2 is the loader, which provides a full-screen text-based interface with a blue background by default. For unattended installations in cmdline mode, the loader offers line-mode, text-based output. (Refer to Chapter 22, *Installation Phase 2: Configuring Language and Installation Source*.)

18.2.4. Installation Phase 3

In installation phase 3 you will use **anaconda** in graphical, text-based, or cmdline mode:

- **Graphical mode**

  This can be used through a VNC client (recommended) or through an X11 server. You can use your mouse and keyboard to navigate through the screens, click buttons, and enter text in fields.

- **Text-based mode**

  This interface does not offer all interface elements of the GUI and does not support all settings. Use this for interactive installations if you cannot use a VNC client or X11 server.

- **cmdline mode**

  This is intended for automated installations on System z. (Refer to Section 26.6, "Parameters for Kickstart Installations")

If you have a slow network connection or prefer a text-based installation, do not use X11 forwarding when logging in over the network and do not set the ***display=*** variable in the parameter file (refer to Section 26.4, "VNC and X11 Parameters" for details). In Red Hat Enterprise Linux 6.9 the text-based installation has been reduced to minimize user interaction. Features like installation on FCP-attached SCSI devices, changing partition layout, or package selection are only available with the graphical user interface installation. Use the graphical installation whenever possible. (Refer to Chapter 23, *Installation Phase 3: Installing Using Anaconda*.)

18.3. GRAPHICAL USER INTERFACE WITH X11 OR VNC

To run **anaconda** with the graphical user interface, use a workstation that has either an X Window System (X11) server or VNC client installed.

You can use X11 forwarding with an SSH client or X11 directly. If the installer on your workstation fails because the X11 server does not support required X11 extensions you might have to upgrade the X11 server or use VNC.

To use VNC, disable X11 forwarding in your SSH client prior to connecting to the Linux installation system on the mainframe or specify the vnc parameter in your parameter file. Using VNC is recommended for slow or long-distance network connections. Refer to Section 28.2, "Enabling Remote Access to the Installation System".

Table 18.1, “Parameters and SSH login types” shows how the parameters and SSH login type controls which **anaconda** user interface is used.

Table 18.1. Parameters and SSH login types

Parameter	SSH login	User interface
none	SSH without X11 forwarding	VNC or text
vnc	SSH with or without X11 forwarding	VNC
none	SSH with X11 forwarding	X11
display=IP/hostname:display	SSH without X11 forwarding	X11

18.3.1. Installation using X11 forwarding

You can connect a workstation to the Linux installation system on the mainframe and display the graphical installation program using SSH with X11 forwarding.

You require an SSH client that allows X11 forwarding. To open the connection, first start the X server on the workstation. Then connect to the Linux installation system. You can enable X11 forwarding in your SSH client when you connect.

For example, with OpenSSH enter the following in a terminal window on your workstation:

```
ssh -X install@linuxvm.example.com
```

Replace *linuxvm.example.com* with the hostname or IP address of the system you are installing. The **-X** option (the capital letter **X**) enables X11 forwarding.

18.3.2. Installation using X11

The direct connection from the X11 client to an X11 server on your local workstation requires an IP connection from your System z to your workstation. If the network and firewalls prevent such connections, use X11 forwarding or VNC instead.

The graphical installation program requires the DNS and hostname to be set correctly, and the Linux installation system must be allowed to open applications on your display. You can ensure this by setting the parameter ***display=workstationname:0.0*** in the parameter file, where workstationname is the hostname of the client workstation connecting to the Linux image. Alternatively, you can set the ***display*** environment variable and run loader manually after having logged in with SSH as user **root**. By default you log in as user **install**. This starts the loader automatically and does not allow overriding the ***display*** environment variable.

To permit X11 clients to open applications on the X11 server on your workstation, use the **xauth** command. To manage X11 authorization cookies with **xauth**, you must log in to the Linux installation system using SSH as user **root**. For details on **xauth** and how to manage authorization cookies, refer to the xauth manpage.

In contrast to setting up X11 authorizations with **xauth**, you can use **xhost** to permit the Linux installation system to connect to the X11 server:

```
xhost +linuxvm
```

Replace *linuxvm* with the hostname or IP address of the Linux installation system. This allows *linuxvm* to make connections to the X11 server.

If the graphical installation does not begin automatically, verify the ***display=*** variable settings in the parameter file. If performing an installation under z/VM, rerun the installation to load the new parameter file on the reader.

18.3.3. Installation using VNC

Using VNC is recommended for slow or long-distance network connections. To use VNC, disable X11 forwarding in your SSH client prior to connecting to the temporary Linux installation system. The loader will then provide a choice between text-mode and VNC; choose VNC here. Alternatively, provide the ***vnc*** variable and optionally the ***vncpassword*** variable in your parameter file (refer to Section 26.4, "VNC and X11 Parameters" for details).

A message on the workstation SSH terminal prompts you to start the VNC client viewer and provides details about the VNC display specifications. Enter the specifications from the SSH terminal into the VNC client viewer and connect to the temporary Linux installation system to begin the installation. Refer to Chapter 31, *Installing Through VNC* for details.

18.3.4. Installation using a VNC listener

To connect from your temporary Linux installation system to a VNC client running on your workstation in listening mode, use the ***vncconnect*** option in your parameter file, in addition to the options ***vnc*** and optionally ***vncpassword***. The network and firewalls must allow an IP connection from your temporary Linux installation to your workstation.

To have the temporary Linux installation system automatically connect to a VNC client, first start the client in listening mode. On Red Hat Enterprise Linux systems, use the **-listen** option to run **vncviewer** as a listener. In a terminal window, enter the command:

```
vncviewer -listen
```

Refer to Chapter 31, *Installing Through VNC* for details.

18.3.5. Automating the Installation with Kickstart

You can allow an installation to run unattended by using Kickstart. A *Kickstart* file specifies settings for an installation. Once the installation system boots, it can read a Kickstart file and carry out the installation process without any further input from a user.

On System z, this also requires a parameter file (optionally an additional configuration file under z/VM). This parameter file must contain the required network options described in Section 26.3, "Installation Network Parameters" and specify a kickstart file using the ***ks=*** option. The kickstart file typically resides on the network. The parameter file often also contains the options ***cmdline*** and ***RUNKS=1*** to execute the loader without having to log in over the network with SSH (Refer to Section 26.6, "Parameters for Kickstart Installations").

For further information and details on how to set up a kickstart file, refer to Section 32.3, "Creating the Kickstart File".

18.3.5.1. Every Installation Produces a Kickstart File

The Red Hat Enterprise Linux installation process automatically writes a Kickstart file that contains the settings for the installed system. This file is always saved as **/root/anaconda-ks.cfg**. You may use this file to repeat the installation with identical settings, or modify copies to specify settings for other systems.

---

[9] *Direct Access Storage Devices* (DASDs) are hard disks that allow a maximum of three partitions per device. For example, **dasda** can have partitions **dasda1**, **dasda2**, and **dasda3**.

[10] Using the SCSI-over-Fibre Channel device driver (zfcp device driver) and a switch, SCSI LUNs can be presented to Linux on System z as if they were locally attached SCSI drives.

CHAPTER 19. PREPARING FOR INSTALLATION

19.1. PREPARING FOR A NETWORK INSTALLATION

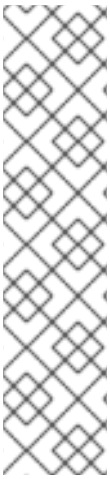
NOTE

Make sure no installation DVD (or any other type of DVD or CD) is in your hosting partition's drive if you are performing a network-based installation. Having a DVD or CD in the drive might cause unexpected errors.

Ensure that you have boot media available as described in Chapter 20, *Booting (IPL) the Installer*.

The Red Hat Enterprise Linux installation medium must be available for either a network installation (via NFS, FTP, HTTP, or HTTPS) or installation via local storage. Use the following steps if you are performing an NFS, FTP, HTTP, or HTTPS installation.

The NFS, FTP, HTTP, or HTTPS server to be used for installation over the network must be a separate, network-accessible server. The separate server can be a virtual machine, LPAR, or any other system (such as a Linux on Power Systems or x86 system). It must provide the complete contents of the installation DVD-ROM.



NOTE

The public directory used to access the installation files over FTP, NFS, HTTP, or HTTPS is mapped to local storage on the network server. For example, the local directory **/var/www/inst/rhel6.9** on the network server can be accessed as **http://network.server.com/inst/rhel6.9**.

In the following examples, the directory on the installation staging server that will contain the installation files will be specified as ***/location/of/disk/space***. The directory that will be made publicly available via FTP, NFS, HTTP, or HTTPS will be specified as */publicly_available_directory*. For example, ***/location/of/disk/space*** may be a directory you create called **/var/isos**. ***/publicly_available_directory*** might be **/var/www/html/rhel6.9**, for an HTTP install.

In the following, you will require an *ISO image*. An ISO image is a file containing an exact copy of the content of a DVD. To create an ISO image from a DVD use the following command:

```
dd if=/dev/dvd of=/path_to_image/name_of_image.iso
```

where *dvd* is your DVD drive device, *name_of_image* is the name you give to the resulting ISO image file, and *path_to_image* is the path to the location on your system where the resulting ISO image will be stored.

To copy the files from the installation DVD to a Linux instance, which acts as an installation staging server, continue with either Section 19.1.1, “Preparing for FTP, HTTP, and HTTPS Installation” or Section 19.1.2, “Preparing for an NFS Installation” .

19.1.1. Preparing for FTP, HTTP, and HTTPS Installation

WARNING

If your **Apache** web server or **tftp** FTP server configuration enables SSL security, make sure to only enable the **TLSv1** protocol, and disable **SSLv2** and **SSLv3**. This is due to the POODLE SSL vulnerability (CVE-2014-3566). See https://access.redhat.com/solutions/1232413 for details about securing **Apache**, and https://access.redhat.com/solutions/1234773 for information about securing **tftp**.

Extract the files from the ISO image of the installation DVD and place them in a directory that is shared over FTP, HTTP, or HTTPS.

Next, make sure that the directory is shared via FTP, HTTP, or HTTPS, and verify client access. Test to see whether the directory is accessible from the server itself, and then from another machine on the same subnet to which you will be installing.

19.1.2. Preparing for an NFS Installation

For NFS installation it is not necessary to extract all the files from the ISO image. It is sufficient to make the ISO image itself, the **install.img** file, and optionally the **product.img** file available on the network server via NFS.

1. Transfer the ISO image to the NFS exported directory. On a Linux system, run:

   ```
   mv /path_to_image/name_of_image.iso /publicly_available_directory/
   ```

   where *path_to_image* is the path to the ISO image file, *name_of_image* is the name of the ISO image file, and *publicly_available_directory* is a directory that is available over NFS or that you intend to make available over NFS.

2. Use a SHA256 checksum program to verify that the ISO image that you copied is intact. Many SHA256 checksum programs are available for various operating systems. On a Linux system, run:

   ```
   $ sha256sum name_of_image.iso
   ```

   where *name_of_image* is the name of the ISO image file. The SHA256 checksum program displays a string of 64 characters called a *hash*. Compare this hash to the hash displayed for this particular image on the **Downloads** page in the Red Hat Customer Portal (refer to Chapter 1, *Obtaining Red Hat Enterprise Linux*). The two hashes should be identical.

3. Copy the **images**/ directory from inside the ISO image to the same directory in which you stored the ISO image file itself. Enter the following commands:

   ```
   mount -t iso9660 /path_to_image/name_of_image.iso /mount_point -o loop,ro
   cp -pr /mount_point/images /publicly_available_directory/
   umount /mount_point
   ```

where ***path_to_image*** is the path to the ISO image file, ***name_of_image*** is the name of the ISO image file, and ***mount_point*** is a mount point on which to mount the image while you copy files from the image. For example:

```
mount -t iso9660 /var/isos/RHEL6.iso /mnt/tmp -o loop,ro
cp -pr /mnt/tmp/images /var/isos/
umount /mnt/tmp
```

The ISO image file and an **images/** directory are now present, side-by-side, in the same directory.

4. Verify that the **images/** directory contains at least the **install.img** file, without which installation cannot proceed. Optionally, the **images/** directory should contain the **product.img** file, without which only the packages for a **Minimal** installation will be available during the package group selection stage (refer to Section 23.17, "Package Group Selection").

5. Ensure that an entry for the publicly available directory exists in the **/etc/exports** file on the network server so that the directory is available via NFS.

   To export a directory read-only to a specific system, use:

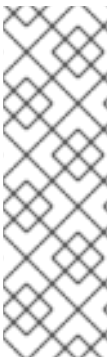
   ```
   /publicly_available_directory client.ip.address (ro)
   ```

   To export a directory read-only to all systems, use:

   ```
   /publicly_available_directory * (ro)
   ```

6. On the network server, start the NFS daemon (on a Red Hat Enterprise Linux system, use **/sbin/service nfs start**). If NFS is already running, reload the configuration file (on a Red Hat Enterprise Linux system use **/sbin/service nfs reload**).

7. Be sure to test the NFS share following the directions in the Red Hat Enterprise Linux Deployment Guide. Refer to your NFS documentation for details on starting and stopping the NFS server.



NOTE

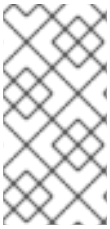
anaconda has the ability to test the integrity of the installation media. It works with the DVD, hard drive ISO, and NFS ISO installation methods. We recommend that you test all installation media before starting the installation process, and before reporting any installation-related bugs (many of the bugs reported are actually due to improperly-burned DVDs). To use this test, type the following command at the **boot:** prompt:

```
linux mediacheck
```

19.2. PREPARING FOR A HARD DRIVE INSTALLATION

Use this option to install Red Hat Enterprise Linux on hardware systems without a DVD drive and if you do not want to access installation phase 3 and the package repository over a network.

19.2.1. Accessing Installation Phase 3 and the Package Repository on a Hard Drive



NOTE

Hard drive installations using DASD or FCP-attached SCSI storage only work from native ext2, ext3, or ext4 partitions. If you have a file system based on devices other than native ext2, ext3, or ext4 (particularly a file system based on RAID or LVM partitions) you will not be able to use it as a source to perform a hard drive installation.

Hard drive installations use an *ISO image* of the installation DVD (a file that contains an exact copy of the content of the DVD), and an **install.img** file extracted from the ISO image. With these files present on a hard drive, you can choose **Hard drive** as the installation source when you boot the installation program.

Hard drive installations use the following files:

- an *ISO image* of the installation DVD. An ISO image is a file that contains an exact copy of the content of a DVD.
- an **install.img** file extracted from the ISO image.
- optionally, a **product.img** file extracted from the ISO image.

With these files present on a hard drive, you can choose **Hard drive** as the installation source when you boot the installation program (refer to Section 22.4, "Installation Method").

Ensure that you have boot media available as described in Chapter 20, *Booting (IPL) the Installer*.

To prepare a DASD or FCP-attached device as an installation source, follow these steps:

1. Obtain an ISO image of the Red Hat Enterprise Linux installation DVD (refer to Chapter 1, *Obtaining Red Hat Enterprise Linux*). Alternatively, if you have the DVD on physical media, you can create an image of it with the following command on a Linux system:

   ```
   dd if=/dev/dvd of=/path_to_image/name_of_image.iso
   ```

   where *dvd* is your DVD drive device, *name_of_image* is the name you give to the resulting ISO image file, and *path_to_image* is the path to the location on your system where the resulting ISO image will be stored.

2. Transfer the ISO images to the DASD or SCSI device.

   The ISO files must be located on a hard drive that is activated in installation phase 1 (refer to Chapter 21, *Installation Phase 1: Configuring a Network Device*) or in installation phase 2 (refer to Chapter 22, *Installation Phase 2: Configuring Language and Installation Source*). This is automatically possible with DASDs.

   For an FCP LUN, you must either boot (IPL) from the same FCP LUN or use the rescue shell provided by the installation phase 1 menus to manually activate the FCP LUN holding the ISOs as described in Section 25.2.1, "Dynamically Activating an FCP LUN" .

3. Use a SHA256 checksum program to verify that the ISO image that you copied is intact. Many SHA256 checksum programs are available for various operating systems. On a Linux system, run:

   ```
   $ sha256sum name_of_image.iso
   ```

   where *name_of_image* is the name of the ISO image file. The SHA256 checksum program

displays a string of 64 characters called a *hash*. Compare this hash to the hash displayed for this particular image on the **Downloads** page in the Red Hat Customer Portal (refer to Chapter 1, *Obtaining Red Hat Enterprise Linux*). The two hashes should be identical.

4. Copy the **images**/ directory from inside the ISO image to the same directory in which you stored the ISO image file itself. Enter the following commands:

```
mount -t iso9660 /path_to_image/name_of_image.iso /mount_point -o loop,ro
cp -pr /mount_point/images /publicly_available_directory/
umount /mount_point
```

where ***path_to_image*** is the path to the ISO image file, ***name_of_image*** is the name of the ISO image file, and ***mount_point*** is a mount point on which to mount the image while you copy files from the image. For example:

```
mount -t iso9660 /var/isos/RHEL6.iso /mnt/tmp -o loop,ro
cp -pr /mnt/tmp/images /var/isos/
umount /mnt/tmp
```

The ISO image file and an **images**/ directory are now present, side-by-side, in the same directory.

5. Verify that the **images**/ directory contains at least the **install.img** file, without which installation cannot proceed. Optionally, the **images**/ directory should contain the **product.img** file, without which only the packages for a **Minimal** installation will be available during the package group selection stage (refer to Section 23.17, "Package Group Selection").

IMPORTANT

install.img and **product.img** must be the only files in the **images**/ directory.

6. Make the DASD or SCSI LUN accessible to the new z/VM guest virtual machine or LPAR, and then proceed with installation. (Refer to Chapter 20, *Booting (IPL) the Installer*) or alternatively with Section 19.2.1.1, "Preparing for Booting the Installer from a Hard Drive" .

NOTE

The Red Hat Enterprise Linux installation program can test the integrity of the installation medium. It works with the DVD, hard drive ISO, and NFS ISO installation methods. We recommend that you test all installation media before starting the installation process, and before reporting any installation-related bugs. To use this test, add the ***mediacheck*** parameter to your parameter file (refer to Section 26.7, "Miscellaneous Parameters").

19.2.1.1. Preparing for Booting the Installer from a Hard Drive

If you would like to boot (IPL) the installer from a hard drive, in addition to accessing installation phase 3 and the package repository, you can optionally install the zipl boot loader on the same (or a different) disk. Be aware that zipl only supports one boot record per disk. If you have multiple partitions on a disk, they all "share" the disk's one boot record.

In the following, assume the hard drive is prepared as described in Section 19.2.1, "Accessing Installation Phase 3 and the Package Repository on a Hard Drive", mounted under **/mnt**, and you do not need to preserve an existing boot record.

To prepare a hard drive to boot the installer, install the zipl boot loader on the hard drive by entering the following command:

```
zipl -V -t /mnt/ -i /mnt/images/kernel.img -r /mnt/images/initrd.img -p /mnt/images/generic.prm
```

For more details on zipl.conf, refer to the chapter on zipl in *Linux on System z Device Drivers, Features, and Commands on Red Hat Enterprise Linux 6*.

WARNING

If you have an operating system installed on the disk, and you still plan to access it later on, refer the chapter on zipl in *Linux on System z Device Drivers, Features, and Commands on Red Hat Enterprise Linux 6* for how to add a new entry in the zipl boot loader (that is, in **zipl.conf**).

CHAPTER 20. BOOTING (IPL) THE INSTALLER

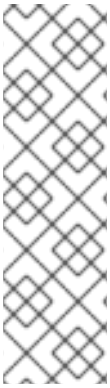
The steps to perform the initial boot (IPL) of the installer depend on the environment (either z/VM or LPAR) in which Red Hat Enterprise Linux will run. For more information on booting, see the *Booting Linux* chapter in *Linux on System z Device Drivers, Features, and Commands on Red Hat Enterprise Linux 6*.

20.1. INSTALLING UNDER Z/VM

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

- the z/VM virtual reader
- a DASD or an FCP-attached SCSI device prepared with the zipl boot loader
- an FCP-attached SCSI DVD drive

Log on to the z/VM guest virtual machine chosen for the Linux installation. You can use **x3270** or **c3270** (from the x3270-text package in Red Hat Enterprise Linux) to log in to z/VM from other Linux systems. Alternatively, use the 3270 terminal emulator on the IBM System z Hardware Management Console (HMC). If you are working from a machine with a Windows operating system, Jolly Giant (http://www.jollygiant.com/) offers an SSL-enabled 3270 emulator. A free native Windows port of **c3270** called **wc3270** also exists.



NOTE

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.

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
```

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
```

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:

- Query the available main memory, which is called *storage* in System z terminology. Your guest should have at least 512 megabytes of main memory.

  ```
  cp query virtual storage
  ```

- Query available network devices of type:

  osa

  OSA (CHPID type OSD, real or virtual (VSWITCH or GuestLAN type QDIO), 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:

  ```
  cp query virtual osa
  ```

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

  ```
  cp query virtual dasd
  ```

- Query available FCP channels:

  ```
  cp query virtual fcp
  ```

20.1.1. Using the z/VM Reader

Perform the following steps to boot from the z/VM reader:

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. Execute the command:

   ```
   ftp host
   ```

   Where *host* is the hostname 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. Optionally 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. Finally execute the REXX script **redhat.exec** to boot (IPL) the installer:

```
redhat
```

20.1.2. Using a Prepared DASD

Boot from the prepared DASD and select the zipl boot menu entry referring to the Red Hat Enterprise Linux installer. 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
```

20.1.3. Using a Prepared FCP-attached SCSI Disk

Perform the following steps to boot from a prepared FCP-attached SCSI disk:

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 installer. Use a command of the following form:

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

   Replace *WWPN* with the WWPN of the storage system and *LUN* with the LUN 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. Optionally, confirm your settings with the command:

```
query loaddev
```

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

```
cp ipl FCP_device
```

For example:

```
cp ipl fc00
```

20.1.4. Using an FCP-attached SCSI DVD Drive

This requires a SCSI DVD drive attached to an FCP-to-SCSI bridge which is in turn connected to an FCP adapter in your System z. The FCP adapter must be configured and available under z/VM.

1. Insert your Red Hat Enterprise Linux for System z DVD into the DVD drive.
2. Configure the SCSI boot loader of z/VM to access the DVD drive in the FCP storage area network and specify **1** for the boot entry on the Red Hat Enterprise Linux for System z DVD. Use a command of the following form:

```
cp set loaddev portname WWPN lun FCP_LUN bootprog 1
```

Replace *WWPN* with the WWPN of the FCP-to-SCSI bridge and *FCP_LUN* with the LUN of the DVD drive. The 16-digit hexadecimal numbers must be split into two pairs of eight characters each. For example:

```
cp set loaddev portname 20010060 eb1c0103 lun 00010000 00000000 bootprog 1
```

3. Optionally, confirm your settings with the command:

```
cp query loaddev
```

4. IPL on the FCP device connected with the FCP-to-SCSI bridge.

```
cp ipl FCP_device
```

For example:

```
cp ipl fc00
```

20.2. INSTALLING IN AN LPAR

When installing in a *logical partition* (LPAR), you can boot from:

- an FTP server
- the DVD drive of the HMC or SE
- a DASD or an FCP-attached SCSI drive prepared with the zipl boot loader

- an FCP-attached SCSI DVD drive

Perform these common steps first:

1. Log in on the IBM System 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. The **SYSPROG** user is recommended.

2. Select **Images**, then select the LPAR to which you wish to install. Use the arrows in the frame on the right side to navigate to the **CPC Recovery** menu.

3. Double-click **Operating System Messages** to show the text console on which Linux boot messages will appear and potentially user input will be required. Refer to the chapter on booting Linux in *Linux on System z Device Drivers, Features, and Commands on Red Hat Enterprise Linux 6* and the *Hardware Management Console Operations Guide* , order number [SC28-6857], for details.

Continue with the procedure for your installation source.

20.2.1. Using an FTP Server

1. Double-click **Load from CD-ROM, DVD, or Server**.

2. In the dialog box that follows, select **FTP Source**, and enter the following information: Host Computer: Hostname or IP address of the FTP server you wish to install from (for example, ftp.redhat.com) User ID: Your user name on the FTP server (or anonymous) Password: Your password (use your email address if you are logging in as anonymous) Account (optional): Leave this field empty File location (optional): Directory on the FTP server holding Red Hat Enterprise Linux for System z (for example, /rhel/s390x/)

3. Click **Continue**.

4. In the dialog that follows, keep the default selection of **generic.ins** and click **Continue**.

20.2.2. Using the HMC or SE DVD Drive

1. Double-click **Load from CD-ROM, DVD, or Server**.

2. In the dialog box that follows, select **Local CD-ROM / DVD** then click **Continue**.

3. In the dialog that follows, keep the default selection of **generic.ins** then click **Continue**.

20.2.3. Using a Prepared DASD

1. Double-click **Load**.

2. In the dialog box that follows, select **Normal** as the **Load type**.

3. As **Load address** fill in the device number of the DASD.

4. As **Load parameter** fill in the number corresponding the zipl boot menu entry that you prepared for booting the Red Hat Enterprise Linux installer.

5. Click the **OK** button.

20.2.4. Using a Prepared FCP-attached SCSI Disk

1. Double-click **Load**.
2. In the dialog box that follows, select **SCSI** as the **Load type**.
3. As **Load address** fill in the device number of the FCP channel connected with the SCSI disk.
4. As **World wide port name** fill in the WWPN of the storage system containing the disk as a 16-digit hexadecimal number.
5. As **Logical unit number** fill in the LUN of the disk as a 16-digit hexadecimal number.
6. As **Boot program selector** fill in the number corresponding the zipl boot menu entry that you prepared for booting the Red Hat Enterprise Linux installer.
7. Leave the **Boot record logical block address** as **0** and the **Operating system specific load parameters** empty.
8. Click the **OK** button.

20.2.5. Using an FCP-attached SCSI DVD Drive

This requires to have a SCSI DVD drive attached to an FCP-to-SCSI bridge which is in turn connected to an FCP adapter in your System z machine. The FCP adapter has to be configured and available in your LPAR.

1. Insert your Red Hat Enterprise Linux for System z DVD into the DVD drive.
2. Double-click **Load**.
3. In the dialog box that follows, select **SCSI** as the **Load type**.
4. As **Load address** fill in the device number of the FCP channel connected with the FCP-to-SCSI bridge.
5. As **World wide port name** fill in the WWPN of the FCP-to-SCSI bridge as a 16-digit hexadecimal number.
6. As **Logical unit number** fill in the LUN of the DVD drive as a 16-digit hexadecimal number.
7. As **Boot program selector** fill in the number **1** to select the boot entry on the Red Hat Enterprise Linux for System z DVD.
8. Leave the **Boot record logical block address** as **0** and the **Operating system specific load parameters** empty.
9. Click the **OK** button.

CHAPTER 21. INSTALLATION PHASE 1: CONFIGURING A NETWORK DEVICE

After the kernel boot, you will configure one network device using the **linuxrc** program. This network device is needed to complete the installation. If you are installing interactively (with the default parameter file **generic.prm**), you will be asked questions about your network. It is a good idea to have your data ready in the form of a datasheet or similar. If you want to automate this step, supply the information for each option in your parameter file or CMS configuration file.

As an example, let us look at how to configure an OSA network adapter under z/VM. When **linuxrc** starts, you see the following message:

```
Starting the zSeries initrd to configure networking. Version is 1.2
Starting udev...
```

Network devices are sensed and listed. The list of devices depends on the ***cio_ignore*** kernel parameter used. If no devices are found because of ***cio_ignore***, as in the example below, you can clear the list of ignored devices. Note that this might take some time and result in a long list when there are many devices, such as on an LPAR.

```
Scanning for available network devices...
Autodetection found 0 devices.
Note: There is a device blacklist active! (Clearing might take long)
c) clear blacklist, m) manual config, r) rescan, s) shell:
c
Clearing device blacklist...
Scanning for available network devices...
Autodetection found 14 devices.

NUM CARD CU CHPID TYPE DRIVER IF DEVICES
 1 OSA (QDIO) 1731/01 00 OSD qeth eth 0.0.f500,0.0.f501,0.0.f502
 2 OSA (QDIO) 1731/01 01 OSD qeth eth 0.0.f503,0.0.f504,0.0.f505
 3 OSA (QDIO) 1731/01 02 OSD qeth eth 0.0.1010,0.0.1011,0.0.1012
 4 HiperSockets 1731/05 03 IQD qeth hsi 0.0.1013,0.0.1014,0.0.1015
 5 OSA (QDIO) 1731/01 04 OSD qeth eth 0.0.1017,0.0.1018,0.0.1019
 6 CTC adapter 3088/08 12 ? ctcm ctc 0.0.1000,0.0.1001
 7 escon channel 3088/1f 12 ? ctcm ctc 0.0.1002,0.0.1003
 8 ficon channel 3088/1e 12 ? ctcm ctc 0.0.1004,0.0.1005
 9 OSA (QDIO) 1731/01 76 OSD qeth eth 0.0.f5f0,0.0.f5f1,0.0.f5f2
 10 LCS OSA 3088/60 8a OSE lcs eth 0.0.1240,0.0.1241
 11 HiperSockets 1731/05 fb IQD qeth hsi 0.0.8024,0.0.8025,0.0.8026
 12 HiperSockets 1731/05 fc IQD qeth hsi 0.0.8124,0.0.8125,0.0.8126
 13 HiperSockets 1731/05 fd IQD qeth hsi 0.0.8224,0.0.8225,0.0.8226
 14 HiperSockets 1731/05 fe IQD qeth hsi 0.0.8324,0.0.8325,0.0.8326

<num>) use config, m) manual config, r) rescan, s) shell:
```

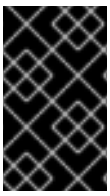
Enter the number of the configuration you want to use, for example **9**. Selecting from the table provides the installer with information for the type of network device and the device addresses for its subchannels. Alternatively, you can enter **m** and proceed to enter the network type (qeth), the read, write, data channels, and the OSA port. Accept defaults by pressing **Enter**; under z/VM you might need to press **Enter** twice.

```
m
```

```
* NOTE: To enter default or empty values press enter twice. *


Network type (qeth, lcs, ctc, ? for help). Default is qeth:
qeth

Read,write,data channel (e.g. 0.0.0300,0.0.0301,0.0.0302 or ? for help).
0.0.f5f0,0.0.f5f1,0.0.f5f2

Portname (1..8 characters, or ? for help). Default is no portname:


Relative port number for OSA (0, 1, or ? for help). Default is 0:



Activating network device...
Detected: OSA card in OSD mode, Gigabit Ethernet
```

Then questions pertaining to your Linux instance are displayed:

```
 Hostname of your new Linux guest (FQDN e.g. s390.redhat.com or ? for help):
host.subdomain.domain

IPv4 address / IPv6 addr. (e.g. 10.0.0.2 / 2001:0DB8:: or ? for help)
10.0.0.42

IPv4 netmask or CIDR prefix (e.g. 255.255.255.0 or 1..32 or ? for help). Default is 255.0.0.0:
24

IPv4 address of your default gateway or ? for help:
10.0.0.1
Trying to reach gateway 10.0.0.1...

IPv4 addresses of DNS servers (separated by colons ':' or ? for help):
10.1.2.3:10.3.2.1
Trying to reach DNS servers...

DNS search domains (separated by colons ':' or ? for help):
subdomain.domain:domain

DASD range (e.g. 200-203,205 or ? for help). Default is autoprobing:
eb1c
Activated DASDs:
0.0.eb1c(ECKD) dasda : active, blocksize: 4096, 1803060 blocks, 7043 MB
```



IMPORTANT

The installer requires the definition of a DASD. For a SCSI-only installation, enter **none**. This satisfies the requirement for a defined DASD parameter, while resulting in a SCSI-only environment.

If you make a mistake, the dialog either notices the error and asks you to re-enter the parameter, or you can go back later to restart the dialog:

```
 Incorrect ... (<OPTION-NAME>):
0) redo this parameter, 1) continue, 2) restart dialog, 3) halt, 4) shell
```

When you restart the dialog, it remembers what you entered before:

```
 Network type
0) default is previous "qeth", 1) new value, ?) help
```

At the end of the configuration, you see the message **Initial configuration completed**:

```
 Initial configuration completed.

c) continue, p) parm file/configuration, n) network state, r) restart, s) shell
```

You can now check your network configuration by entering **n**:

```
 n
eth0 Link encap:Ethernet HWaddr 02:00:00:AB:C9:81
 inet addr:10.0.0.42 Bcast:10.0.0.255 Mask:255.255.255.0
 UP BROADCAST RUNNING MULTICAST MTU:1492 Metric:1
 RX packets:64 errors:0 dropped:0 overruns:0 frame:0
 TX packets:4 errors:0 dropped:0 overruns:0 carrier:0
 collisions:0 txqueuelen:1000
 RX bytes:3334 (3.2 KiB) TX bytes:336 (336.0 b)

lo Link encap:Local Loopback
 inet addr:127.0.0.1 Mask:255.0.0.0
 UP LOOPBACK RUNNING MTU:16436 Metric:1
 RX packets:0 errors:0 dropped:0 overruns:0 frame:0
 TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
 collisions:0 txqueuelen:0
 RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)

Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
127.0.0.1 0.0.0.0 255.255.255.255 UH 0 0 0 lo
10.0.0.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0
0.0.0.0 10.0.0.1 0.0.0.0 UG 0 0 0 eth0

c) continue, p) parm file/configuration, n) network state, r) restart, s) shell
```

If you want to change something, enter **r** to restart the dialog. To show the parameters as specified in a parameter or configuration file or interactively enter **p**. You can then copy the output from your terminal and paste it into an editor to save it to disk on your local workstation. You can use the copy as a template for a parameter or configuration file for future installations:

```
 p

NETTYPE=qeth
IPADDR=10.0.0.42
NETMASK=255.255.255.0
```

```
GATEWAY=10.0.0.1
HOSTNAME=host.subdomain.domain
SUBCHANNELS=0.0.f5f0,0.0.f5f1,0.0.f5f2
LAYER2=1
MACADDR=02:00:00:AB:C9:81
PORTNAME=OSAPORT
DNS=10.1.2.3:10.3.2.1
SEARCHDNS=subdomain.domain:domain
DASD=eb1c

c) continue, p) parm file/configuration, n) network state, r) restart, s) shell
```

Again, to change something, restart the dialog with **r**. Finally, if all is in order, enter **c** to continue:

```
 c

Starting sshd to allow login over the network.

Connect now to 10.0.0.42 and log in as user 'install' to start the
installation.
E.g. using: ssh -x install@10.0.0.42
For VNC or text mode, disable X11 forwarding (recommended) with 'ssh -x'.
For X11, enable X11 forwarding with 'ssh -X'.

You may log in as the root user to start an interactive shell.
```

The preliminary network setup is now complete and the installer starts an SSH daemon. You can log into your Linux instance over SSH. If you are using **RUNKS=1** with kickstart and cmdline mode, **linuxrc** automatically starts the loader.

21.1. A NOTE ON TERMINALS

During the installation, the installation program displays messages on a line-mode terminal. This is the HMC **Operating System Messages** applet if you install under LPAR, or a 3270 terminal if you install under z/VM.

Linuxrc provides a rescue shell on the line-mode terminal. Press the **Enter** key (twice under z/VM) to start the shell. You cannot use full-screen applications such as the **vi** editor on the line-mode terminal. Switch to line-mode based editors such as **ed**, **ex**, or **sed** to edit text files if necessary.

Be aware that long-running commands might not be interruptible with the escape sequence **Ctrl**+**C**. Call commands with options that make them return in time voluntarily. The shell on the 3270 terminal is available throughout the whole installation process until the point where the system needs to reboot.

Once the shell has been provided, you may exit with an error level of zero to get a new shell instance replacing the old one, or you may exit with an error level different from zero to force a shutdown of the installation system.

Connect to the installed system using user **root** to get a root shell without automatically starting the installer. For problem determination, you might connect with many ssh sessions.

CHAPTER 22. INSTALLATION PHASE 2: CONFIGURING LANGUAGE AND INSTALLATION SOURCE

Before the graphical installation program starts, you need to configure the language and installation source.

By default, if you are installing interactively (with the default parameter file **generic.prm**) the loader program to select language and installation source starts in text mode. In your new ssh session, the following message is displayed:

> Welcome to the anaconda install environment 1.2 for zSeries

22.1. NON-INTERACTIVE LINE-MODE INSTALLATION

If the **cmdline** option was specified as boot option in your parameter file (Section 26.6, "Parameters for Kickstart Installations") or in your kickstart file (refer to Section 32.3, "Creating the Kickstart File" , the loader starts up with line-mode oriented text output. In this mode, all necessary information must be provided in the kickstart file. The installer does not allow user interaction and stops if there is unspecified installation information.

22.2. THE TEXT MODE INSTALLATION PROGRAM USER INTERFACE

Both the loader and later **anaconda** use a screen-based interface that includes most of the on-screen *widgets* commonly found on graphical user interfaces. Figure 22.1, "Installation Program Widgets as seen in **URL Setup**", and Figure 22.2, "Installation Program Widgets as seen in **Choose a Language**", illustrate widgets that appear on screens during the installation process.

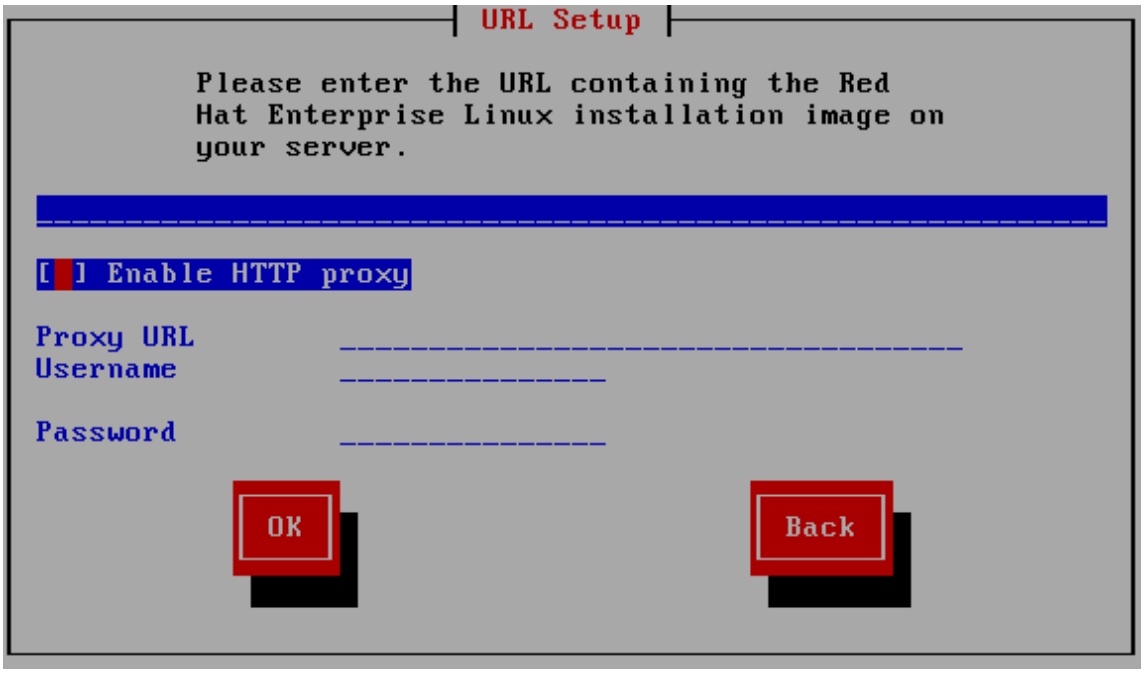


Figure 22.1. Installation Program Widgets as seen in**URL Setup**

Figure 22.2. Installation Program Widgets as seen in Choose a Language

Here is a list of the most important widgets shown in Figure 22.1, "Installation Program Widgets as seen in **URL Setup**" and Figure 22.2, "Installation Program Widgets as seen in **Choose a Language**":

- Window – Windows (usually referred to as *dialogs* in this manual) appear on your screen throughout the installation process. At times, one window may overlay another; in these cases, you can only interact with the window on top. When you are finished in that window, it disappears, allowing you to continue working in the window underneath.

- Checkbox – Checkboxes allow you to select or deselect a feature. The box displays either an asterisk (selected) or a space (unselected). When the cursor is within a checkbox, press **Space** to select or deselect a feature.

- Text Input – Text input lines are regions where you can enter information required by the installation program. When the cursor rests on a text input line, you may enter and/or edit information on that line.

- Text Widget – Text widgets are regions of the screen for the display of text. At times, text widgets may also contain other widgets, such as checkboxes. If a text widget contains more information than can be displayed in the space reserved for it, a scroll bar appears; if you position the cursor within the text widget, you can then use the **Up** and **Down** arrow keys to scroll through all the information available. Your current position is shown on the scroll bar by a **#** character, which moves up and down the scroll bar as you scroll.

- Scroll Bar – Scroll bars appear on the side or bottom of a window to control which part of a list or document is currently in the window's frame. The scroll bar makes it easy to move to any part of a file.

- Button Widget – Button widgets are the primary method of interacting with the installation program. You progress through the windows of the installation program by navigating these buttons, using the **Tab** and **Enter** keys. Buttons can be selected when they are highlighted.

- Cursor — Although not a widget, the cursor is used to select (and interact with) a particular widget. As the cursor is moved from widget to widget, it may cause the widget to change color, or the cursor itself may only appear positioned in or next to the widget. In Figure 22.1, "Installation Program Widgets as seen in **URL Setup**", the cursor is positioned on the **Enable HTTP proxy** checkbox. Figure 8.2, "Installation Program Widgets as seen in **Choose a Language**", shows the cursor on the **OK** button.

22.2.1. Using the Keyboard to Navigate

Navigation through the installation dialogs is performed through a simple set of keystrokes. To move the cursor, use the **Left**, **Right**, **Up**, and **Down** arrow keys. Use **Tab**, and **Shift**-**Tab** to cycle forward or backward through each widget on the screen. Along the bottom, most screens display a summary of available cursor positioning keys.

To "press" a button, position the cursor over the button (using **Tab**, for example) and press **Space** or **Enter**. To select an item from a list of items, move the cursor to the item you wish to select and press **Enter**. To select an item with a checkbox, move the cursor to the checkbox and press **Space** to select an item. To deselect, press **Space** a second time.

Pressing **F12** accepts the current values and proceeds to the next dialog; it is equivalent to pressing the **OK** button.

WARNING

Unless a dialog box is waiting for your input, do not press any keys during the installation process (doing so may result in unpredictable behavior).

22.3. LANGUAGE SELECTION

Use the arrow keys on your keyboard to select a language to use during the installation process (refer to Figure 22.3, "Language Selection"). With your selected language highlighted, press the **Tab** key to move to the **OK** button and press the **Enter** key to confirm your choice. You can automate this choice in the parameter file with the parameter ***lang=*** (refer to Section 26.5, "Loader Parameters") or with the kickstart command **lang** (refer to Section 28.4, "Automating the Installation with Kickstart").

The language you select here will become the default language for the operating system once it is installed. Selecting the appropriate language also helps target your time zone configuration later in the installation. The installation program tries to define the appropriate time zone based on what you specify on this screen.

To add support for additional languages, customize the installation at the package selection stage. For more information, refer to Section 23.17.2, " Customizing the Software Selection ".

Figure 22.3. Language Selection

Once you select the appropriate language, click **Next** to continue.

22.4. INSTALLATION METHOD

Use the arrow keys on your keyboard to select an installation method (refer to Figure 22.4, "Installation Method"). With your selected method highlighted, press the **Tab** key to move to the **OK** button and press the **Enter** key to confirm your choice.

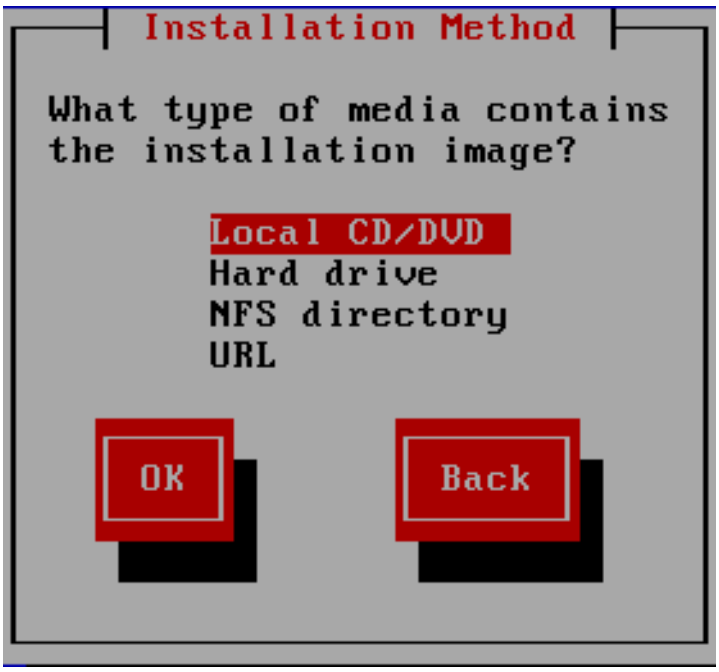


Figure 22.4. Installation Method

22.4.1. Installing from a DVD

To install Red Hat Enterprise Linux from a DVD, place the DVD in your DVD drive and boot your system from the DVD as described in Section 20.1.4, " Using an FCP-attached SCSI DVD Drive" for z/VM or Section 20.2.5, "Using an FCP-attached SCSI DVD Drive" for LPAR.

The installation program then probes your system and attempts to identify your DVD-ROM drive. It starts by looking for a SCSI DVD-ROM drive.

NOTE

To abort the installation process at this time, reboot your machine and then eject the boot media. You can safely cancel the installation at any point before the **Write changes to disk** screen. Refer to Section 23.16, "Write Changes to Disk" for more information.

If the DVD drive is found and the driver loaded, the installer presents you with the option to perform a media check on the DVD. This takes some time, and you may opt to skip over this step. However, if you later encounter problems with the installer, you should reboot and perform the media check before calling for support. From the media check dialog, continue to the next stage of the installation process (refer to Section 23.5, "Welcome to Red Hat Enterprise Linux").

22.4.2. Installing from a Hard Drive

The **Select Partition** screen applies only if you are installing from a disk partition (that is, you selected **Hard Drive** in the **Installation Method** dialog). This dialog allows you to name the disk partition and directory from which you are installing Red Hat Enterprise Linux. If you used the **repo=hd** boot option, you already specified a partition.

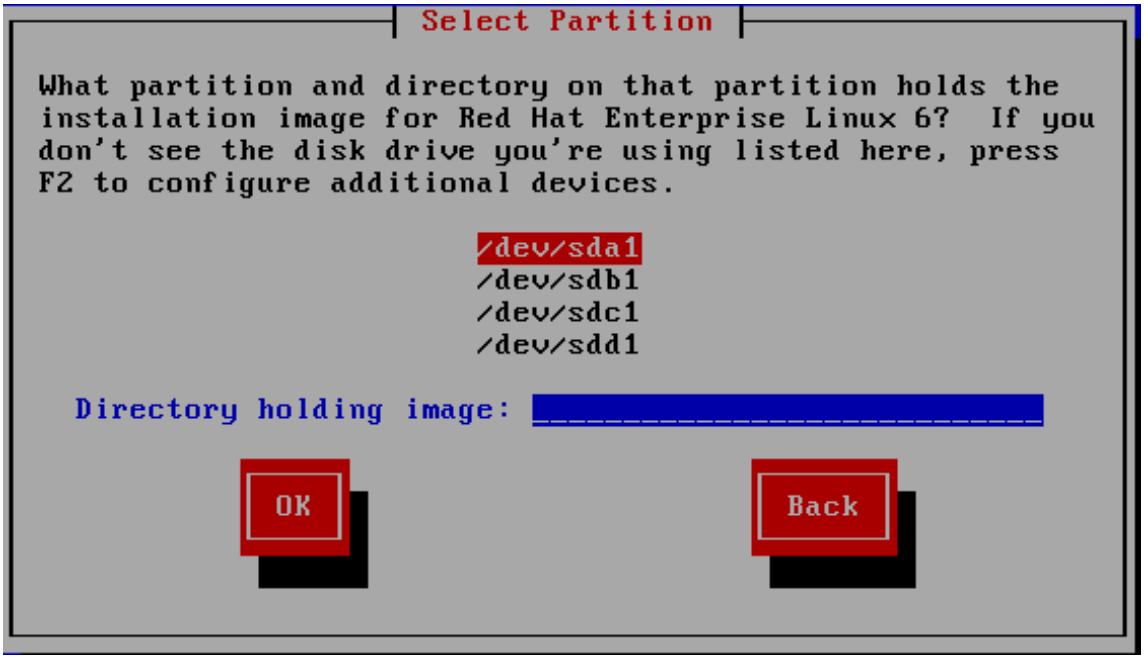


Figure 22.5. Selecting Partition Dialog for Hard Drive Installation

Select the partition containing the ISO files from the list of available partitions. DASD names begin with **/dev/dasd**. Each individual drive has its own letter, for example **/dev/dasda** or **/dev/sda**. Each partition on a drive is numbered, for example **/dev/dasda1** or **/dev/sda1**.

For an FCP LUN, you would have to either boot (IPL) from the same FCP LUN or use the rescue shell provided by the **linuxrc** menus to manually activate the FCP LUN holding the ISOs as described in Section 25.2.1, "Dynamically Activating an FCP LUN" .

Also specify the **Directory holding images**. Enter the full directory path from the drive that contains the ISO image files. The following table shows some examples of how to enter this information:

Table 22.1. Location of ISO images for different partition types

File system	Mount point	Original path to files	Directory to use
ext2, ext3, ext4	/home	/home/user1/RHEL6.9	/user1/RHEL6.9

If the ISO images are in the root (top-level) directory of a partition, enter a /. If the ISO images are located in a subdirectory of a mounted partition, enter the name of the directory holding the ISO images within that partition. For example, if the partition on which the ISO images is normally mounted as **/home**/, and the images are in **/home**/**new**/, you would enter /**new**/.

IMPORTANT

An entry without a leading slash may cause the installation to fail.

Select **OK** to continue. Proceed with Chapter 23, *Installation Phase 3: Installing Using Anaconda* .

22.4.3. Performing a Network Installation

The installation program is network-aware and can use network settings for a number of functions. On System z, installation phases 2 and 3 take over the network configuration values specified previously either interactively or by means of a parameter or configuration file in installation phase 1. You can also instruct the installation program to consult additional software repositories later in the process.

- If you are installing via NFS, proceed to Section 22.4.4, "Installing via NFS" .
- If you are installing via Web or FTP, proceed to Section 22.4.5, "Installing via FTP, HTTP, or HTTPS".

22.4.4. Installing via NFS

The NFS dialog applies only if you selected **NFS Image** in the **Installation Method** dialog. If you used the **repo=nfs** boot option, you already specified a server and path.

Figure 22.6. NFS Setup Dialog

1. Enter the domain name or IP address of your NFS server in the **NFS server name** field. For example, if you are installing from a host named **eastcoast** in the domain **example.com**, enter **eastcoast.example.com**.

2. Enter the name of the exported directory in the **Red Hat Enterprise Linux 6.9 directory** field:

   - If the NFS server is exporting a mirror of the Red Hat Enterprise Linux installation tree, enter the directory which contains the root of the installation tree. If everything was specified properly, a message appears indicating that the installation program for Red Hat Enterprise Linux is running.

   - If the NFS server is exporting the ISO image of the Red Hat Enterprise Linux DVD, enter the directory which contains the ISO image.

   If you followed the setup described in Section 19.1.2, "Preparing for an NFS Installation" , the exported directory is the one that you specified as ***publicly_available_directory***.

3. Specify any NFS mount options that you require in the **NFS mount options** field. Refer to the man pages for **mount** and **nfs** for a comprehensive list of options. If you do not require any mount options, leave the field empty.

4. Proceed with Chapter 23, *Installation Phase 3: Installing Using Anaconda* .

22.4.5. Installing via FTP, HTTP, or HTTPS

IMPORTANT

When you provide a URL to an installation source, you must explicitly specify **http://** or **https://** or **ftp://** as the protocol.

The URL dialog applies only if you are installing from a FTP, HTTP, or HTTPS server (if you selected **URL** in the **Installation Method** dialog). This dialog prompts you for information about the FTP, HTTP, or HTTPS server from which you are installing Red Hat Enterprise Linux. If you used the **repo=ftp** or **repo=http** boot options, you already specified a server and path.

Enter the name or IP address of the FTP, HTTP, or HTTPS site from which you are installing, and the name of the directory that contains the **/images** directory for your architecture. For example:

/mirrors/redhat/rhel-6.9/Server/s390x/

To install via a secure HTTPS connection, specify **https://** as the protocol.

Specify the address of a proxy server, and if necessary, provide a port number, username, and password. If everything was specified properly, a message box appears indicating that files are being retrieved from the server.

If your FTP, HTTP, or HTTPS server requires user authentication, specify user and password as part of the URL as follows:

{ftp|http|https}://<user>:<password>@<hostname>[:<port>]/<directory>/

For example:

http://install:rhel6.9pw@name.example.com/mirrors/redhat/rhel-6.9/Server/s390x/

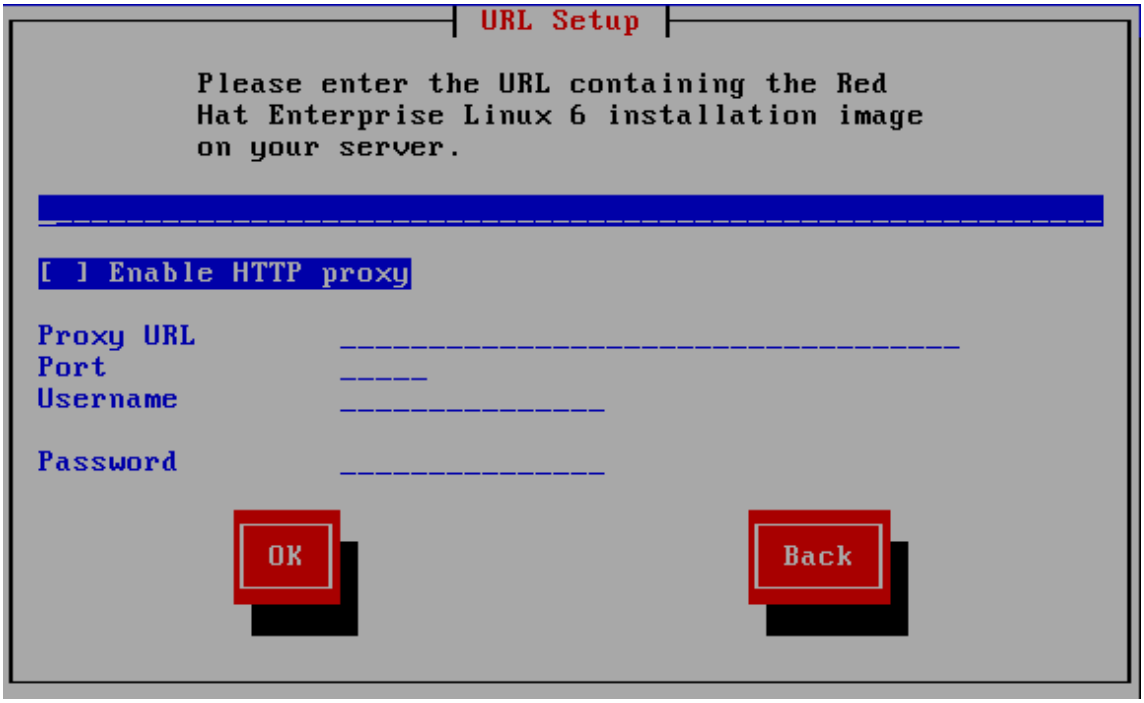


Figure 22.7. URL Setup Dialog

Proceed with Chapter 23, *Installation Phase 3: Installing Using Anaconda* .

22.5. VERIFYING MEDIA

The DVD offers an option to verify the integrity of the media. Recording errors sometimes occur while producing DVD media. An error in the data for package chosen in the installation program can cause the installation to abort. To minimize the chances of data errors affecting the installation, verify the media before installing.

If the verification succeeds, the installation process proceeds normally. If the process fails, create a new DVD using the ISO image you downloaded earlier.

22.6. RETRIEVING PHASE 3 OF THE INSTALLATION PROGRAM

The loader then retrieves phase 3 of the installation program from the network into its RAM disk. This may take some time.

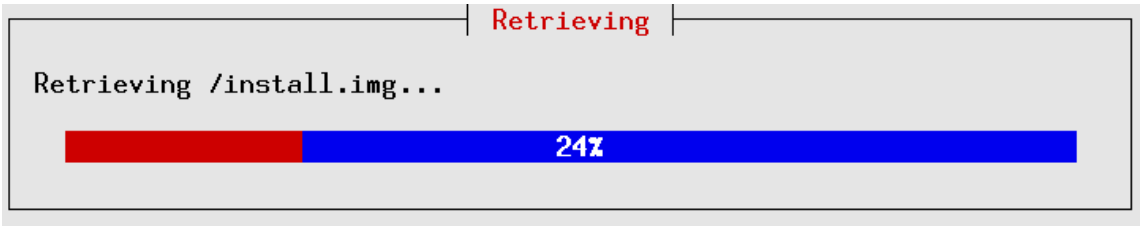


Figure 22.8. Retrieving phase 3 of the installation program

CHAPTER 23. INSTALLATION PHASE 3: INSTALLING USING ANACONDA

This chapter describes an installation using the graphical user interface of **anaconda**.

23.1. THE NON-INTERACTIVE LINE-MODE TEXT INSTALLATION PROGRAM OUTPUT

If the **cmdline** option was specified as boot option in your parameter file (Refer to Section 26.6, "Parameters for Kickstart Installations") or in your kickstart file (refer to Chapter 32, *Kickstart Installations*), **anaconda** starts with line-mode oriented text output. In this mode, all necessary information must be provided in the kickstart file. The installer will not allow user interaction and stops if there is unspecified installation information.

23.2. THE TEXT MODE INSTALLATION PROGRAM USER INTERFACE

While text mode installations are not explicitly documented, those using the text mode installation program can easily follow the GUI installation instructions. However, because text mode presents you with a simpler, more streamlined installation process, certain options that are available in graphical mode are not also available in text mode. These differences are noted in the description of the installation process in this guide, and include:

- Interactively activating FCP LUNs
- configuring advanced storage methods such as LVM, RAID, FCoE, zFCP, and iSCSI.
- customizing the partition layout
- customizing the bootloader layout
- selecting packages during installation
- configuring the installed system with **firstboot**

23.3. THE GRAPHICAL INSTALLATION PROGRAM USER INTERFACE

If you have used a *graphical user interface (GUI)* before, you are already familiar with this process; use your mouse to navigate the screens, click buttons, or enter text fields.

You can also navigate through the installation using the keyboard. The **Tab** key allows you to move around the screen, the Up and Down arrow keys to scroll through lists, **+** and **-** keys expand and collapse lists, while **Space** and **Enter** selects or removes from selection a highlighted item. You can also use the **Alt**+***X*** key command combination as a way of clicking on buttons or making other screen selections, where ***X*** is replaced with any underlined letter appearing within that screen.

23.4. CONFIGURE THE INSTALL TERMINAL

If you logged in with ssh and X11 forwarding, **anaconda** starts immediately with its graphical user interface.

If you did not set the ***display=*** variable and do not use X11 forwarding, **anaconda** gives you the choice of starting VNC or text mode.

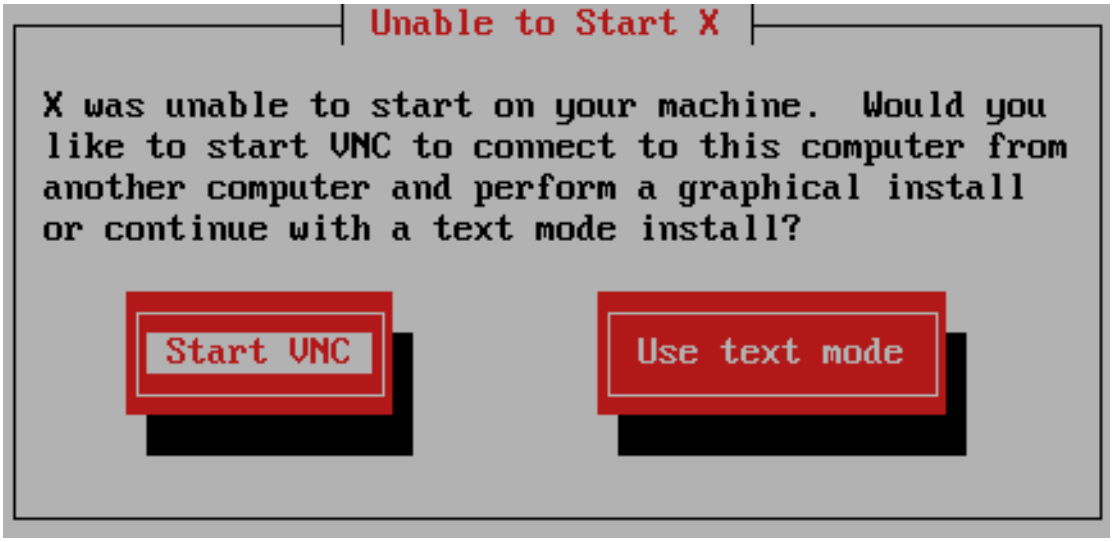


Figure 23.1. Choosing VNC or text mode

If you choose VNC, you will be asked for a password or you can choose to use VNC without a password. If you use a password, make a note of the password for future reference. The VNC server then starts.

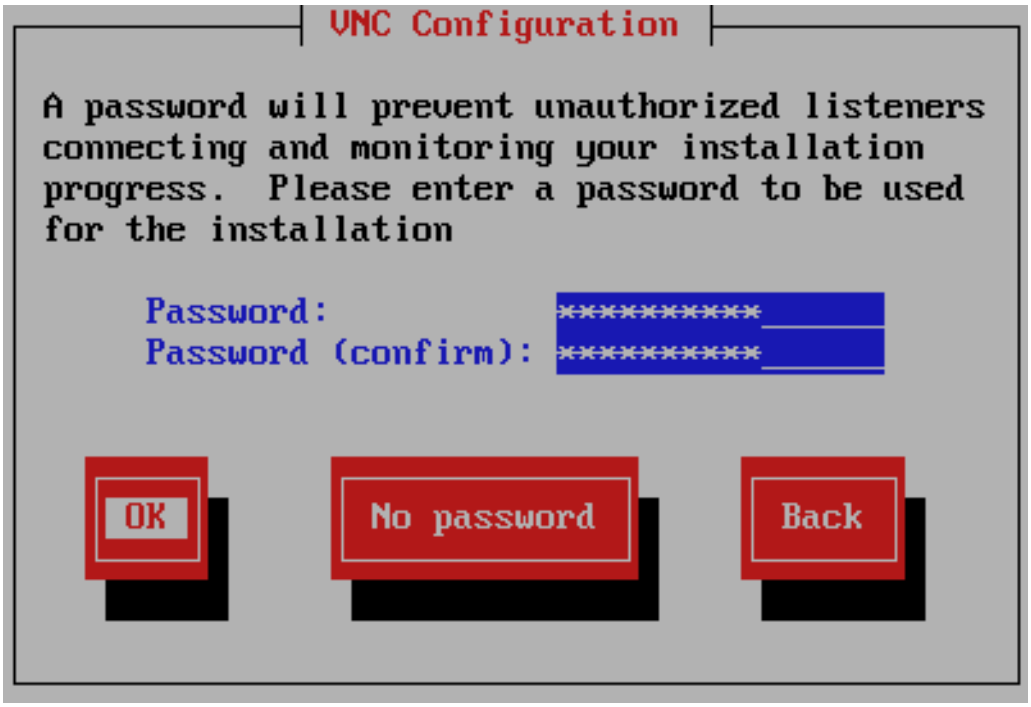


Figure 23.2. The VNC server starts

Now open a connection to the IP address of your z/VM guest virtual machine using a VNC client. Authenticate to the VNC server with the previously entered password.

23.5. WELCOME TO RED HAT ENTERPRISE LINUX

The **Welcome** screen does not prompt you for any input.

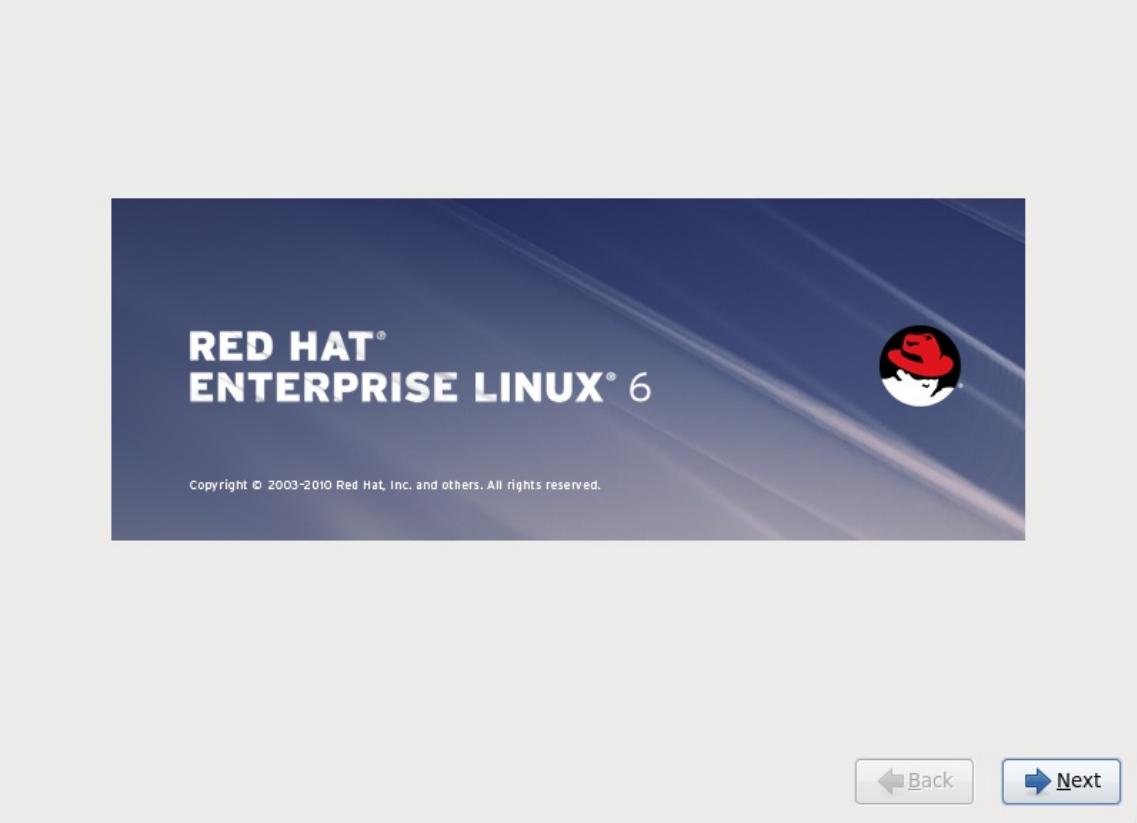


Figure 23.3. The Welcome screen

Click on the **Next** button to continue.

23.6. STORAGE DEVICES

You can install Red Hat Enterprise Linux on a large variety of storage devices. For System z, select **Specialized Storage Devices**

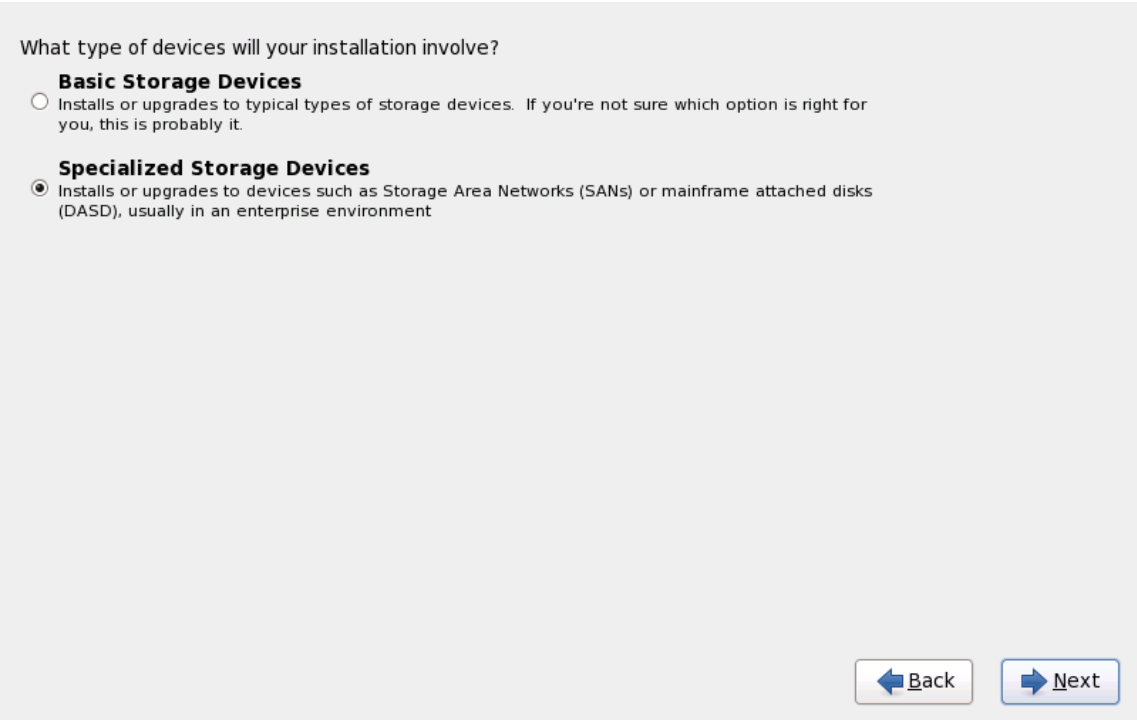


Figure 23.4. Storage devices

Basic Storage Devices

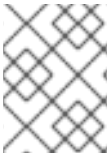
This option does not apply to System z.

Specialized Storage Devices

Select **Specialized Storage Devices**to install Red Hat Enterprise Linux on the following storage devices:

- *Direct access storage devices* (DASDs)
- Multipath devices such as FCP-attachable SCSI LUN with multiple paths
- *Storage area networks* (SANs) such as FCP-attachable SCSI LUNs with a single path

Use the **Specialized Storage Devices**option to configure *Internet Small Computer System Interface* (iSCSI) connections. You cannot use the *FCoE* (Fiber Channel over Ethernet) option on System z; this option is grayed out.



NOTE

Monitoring of LVM and software RAID devices by the **mdeventd** daemon is not performed during installation.

23.6.1. The Storage Devices Selection Screen

The storage devices selection screen displays all storage devices to which **anaconda** has access.

Devices are grouped under the following tabs:

Basic Devices

Basic storage devices directly connected to the local system, such as hard disk drives and solid-state drives. On System z, this contains activated DASDs.

Firmware RAID

Storage devices attached to a firmware RAID controller. This does not apply to System z.

Multipath Devices

Storage devices accessible through more than one path, such as through multiple SCSI controllers or Fiber Channel ports on the same system.

IMPORTANT

The installer only detects multipath storage devices with serial numbers that are 16 or 32 characters in length.

Other SAN Devices

Any other devices available on a storage area network (SAN) such as FCP LUNs attached over one single path.

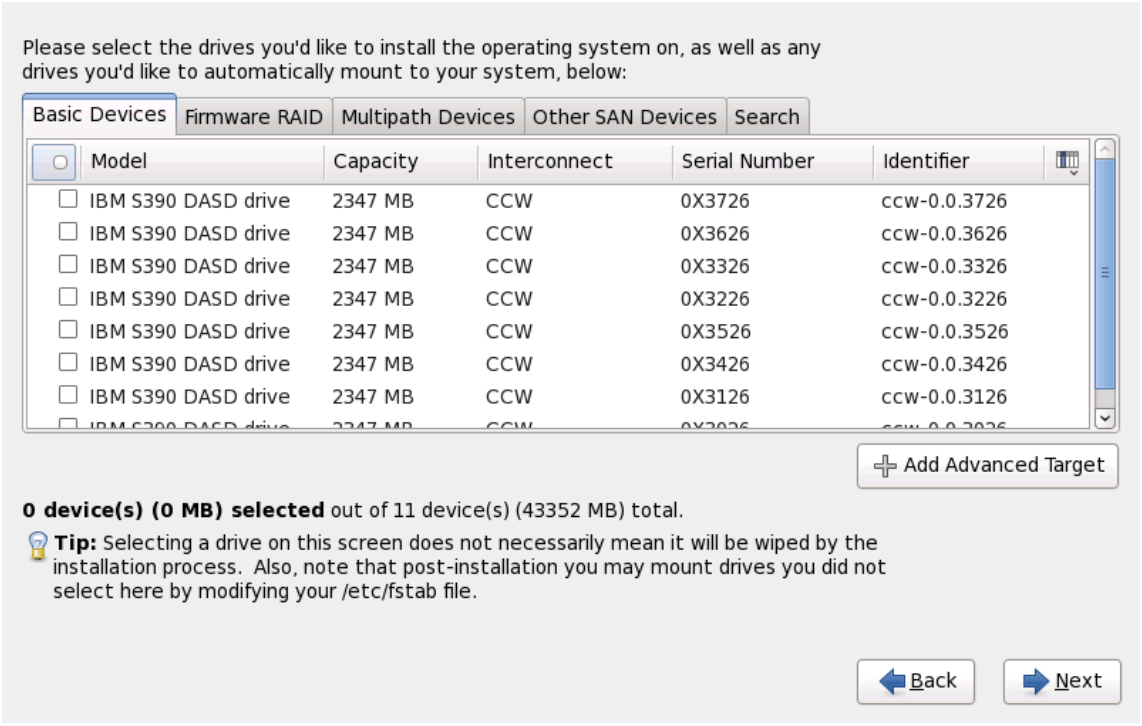


Figure 23.5. Select storage devices – Basic Devices

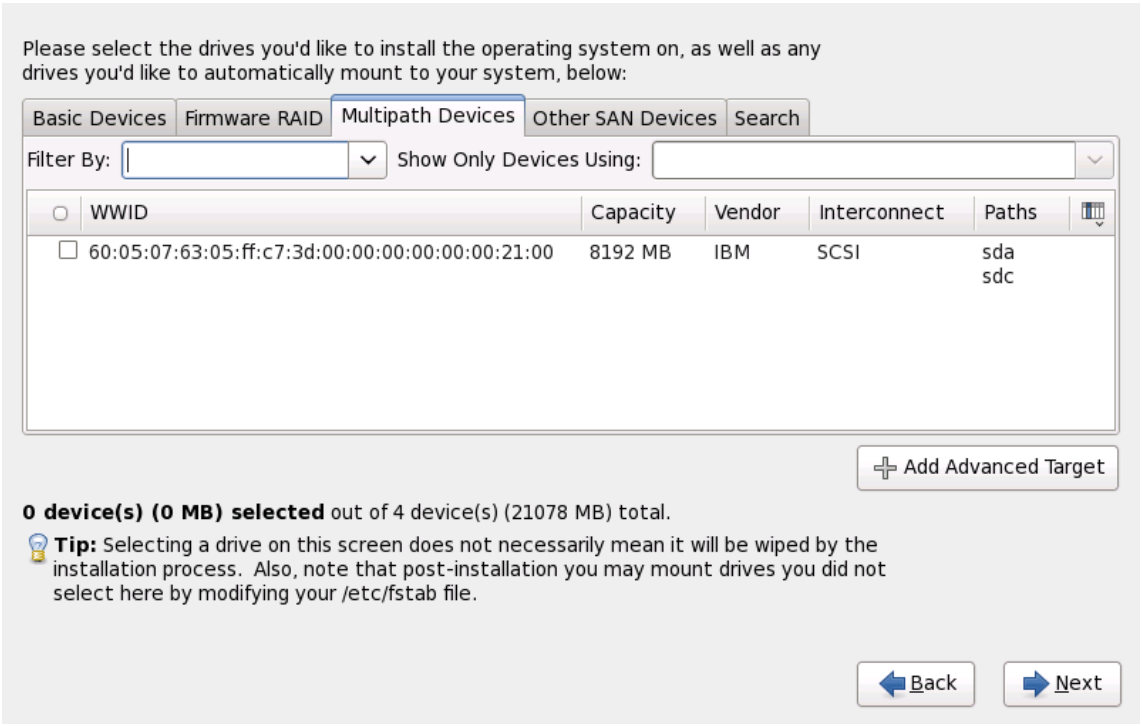


Figure 23.6. Select storage devices – Multipath Devices

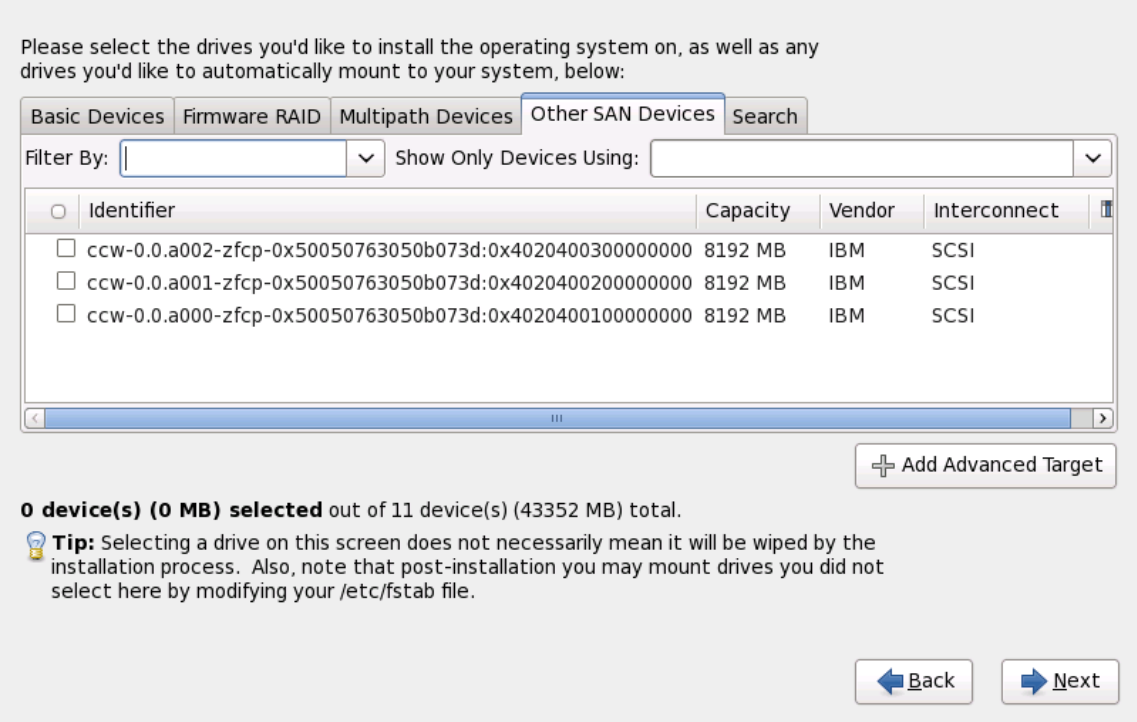


Figure 23.7. Select storage devices — Other SAN Devices

The storage devices selection screen also contains a **Search** tab that allows you to filter storage devices either by their *World Wide Identifier* (WWID) or by the port, target, or *logical unit number* (LUN) at which they are accessed.

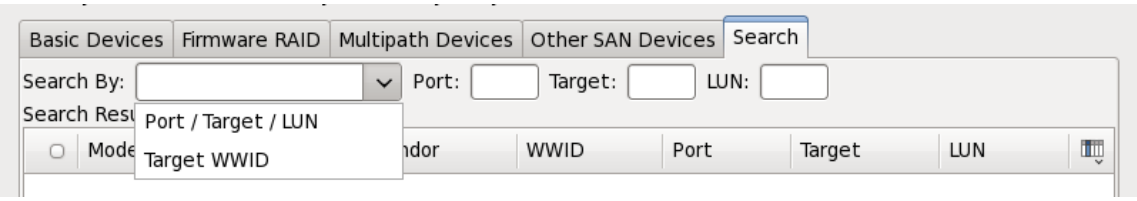


Figure 23.8. The Storage Devices Search Tab

The tab contains a drop-down menu to select searching by port, target, WWID, or LUN (with corresponding text boxes for these values). Searching by WWID or LUN requires additional values in the corresponding text box.

Each tab presents a list of devices detected by **anaconda**, with information about the device to help you to identify it. A small drop-down menu marked with an icon is located to the right of the column headings. This menu allows you to select the types of data presented on each device. For example, the menu on the **Multipath Devices** tab allows you to specify any of **WWID**, **Capacity**, **Vendor**, **Interconnect**, and **Paths** to include among the details presented for each device. Reducing or expanding the amount of information presented might help you to identify particular devices.

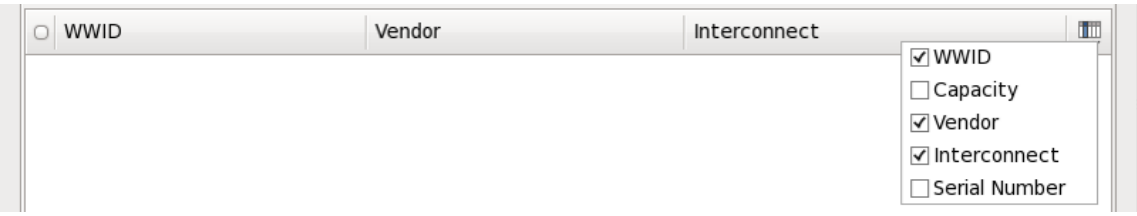


Figure 23.9. Selecting Columns

Each device is presented on a separate row, with a checkbox to its left. Click the checkbox to make a

device available during the installation process, or click the *radio button* at the left of the column headings to select or deselect all the devices listed in a particular screen. Later in the installation process, you can choose to install Red Hat Enterprise Linux onto any of the devices selected here, and can choose to automatically mount any of the other devices selected here as part of the installed system.

Note that the devices that you select here are not automatically erased by the installation process. Selecting a device on this screen does not, in itself, place data stored on the device at risk. Note also that any devices that you do not select here to form part of the installed system can be added to the system after installation by modifying the **/etc/fstab** file.

when you have selected the storage devices to make available during installation, click **Next** and proceed to Section 23.7, "Setting the Hostname"

23.6.1.1. DASD low-level formatting

Any DASDs used must be low-level formatted. The installer detects this and lists the DASDs that need formatting.

If any of the DASDs specified interactively in **linuxrc** or in a parameter or configuration file are not yet low-level formatted, the following confirmation dialog appears:

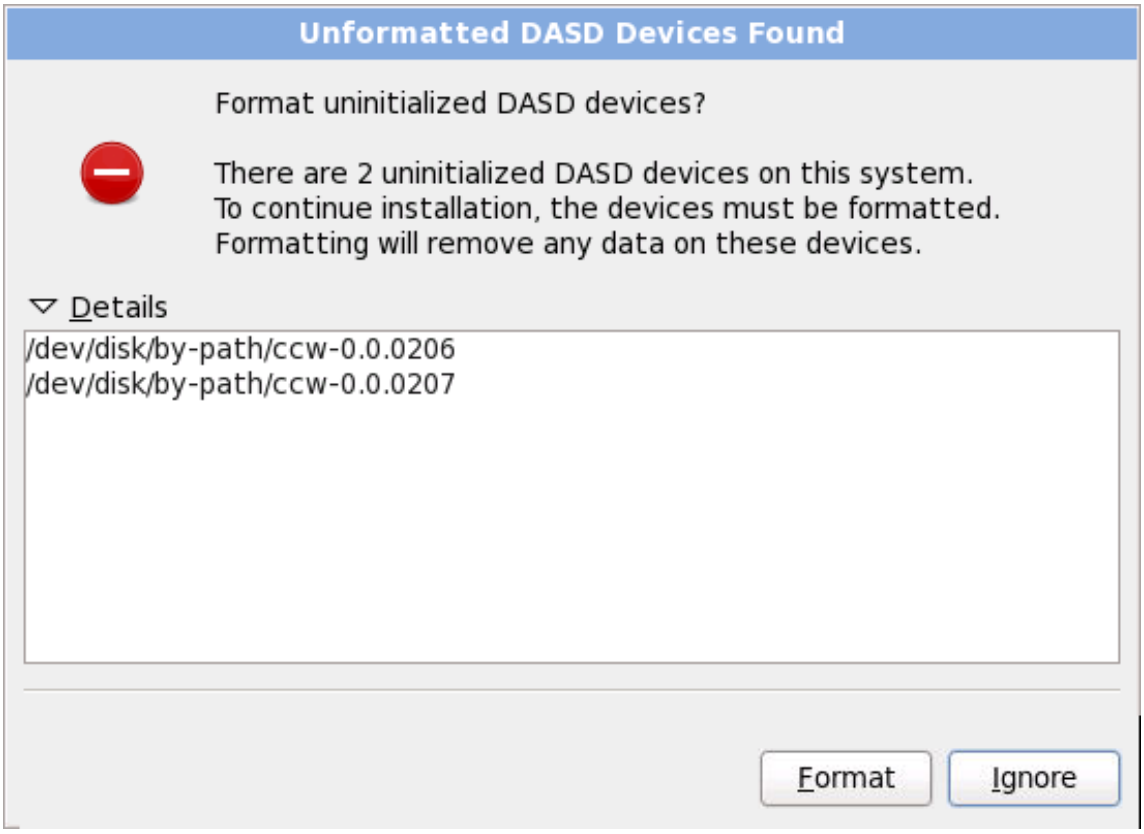


Figure 23.10. Unformatted DASD Devices Found

To automatically allow low-level formatting of unformatted online DASDs specify the kickstart command **zerombr**. Refer to Chapter 32, *Kickstart Installations* for more details.

23.6.1.2. Advanced Storage Options

From this screen you can configure an *iSCSI* (SCSI over TCP/IP) target or FCP LUNs. Refer to Appendix B, *iSCSI Disks* for an introduction to iSCSI.

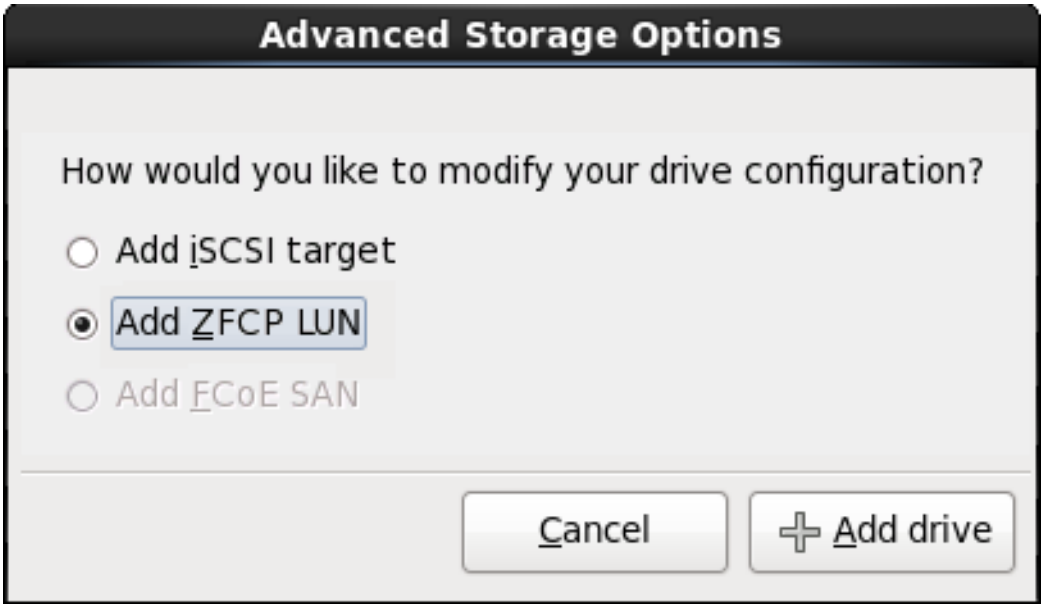


Figure 23.11. Advanced Storage Options

23.6.1.2.1. Configure iSCSI parameters

To use iSCSI storage devices for the installation, **anaconda** 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 username and password for *CHAP* (Challenge Handshake Authentication Protocol) 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 username and password are different for CHAP authentication and reverse CHAP authentication.

Repeat the iSCSI discovery and iSCSI login steps as many times as necessary to add all required iSCSI storage. However, 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.

Procedure 23.1. iSCSI discovery

Use the **iSCSI Discovery Details** dialog to provide **anaconda** with the information that it needs to discover the iSCSI target.

Figure 23.12. The iSCSI Discovery Details dialog

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

2. Provide a name in the **iSCSI Initiator Name** field for the iSCSI initiator in *iSCSI qualified name* (IQN) format.

   A valid IQN contains:

   - 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 therefore resembles: **iqn.2010-09.storage.example.com:diskarrays-sn-a8675309**, and **anaconda** pre-populates the **iSCSI Initiator Name** field with a name in this format to help you with the structure.

   For more information on IQNs, refer to *3.2.6. iSCSI Names* in *RFC 3720 - Internet Small Computer Systems Interface (iSCSI)* available from http://tools.ietf.org/html/rfc3720#section-3.2.6 and *1. iSCSI Names and Addresses* in *RFC 3721 - Internet Small Computer Systems Interface (iSCSI) Naming and Discovery* available from http://tools.ietf.org/html/rfc3721#section-1.

3. Use the drop-down menu to specify the type of authentication to use for iSCSI discovery:

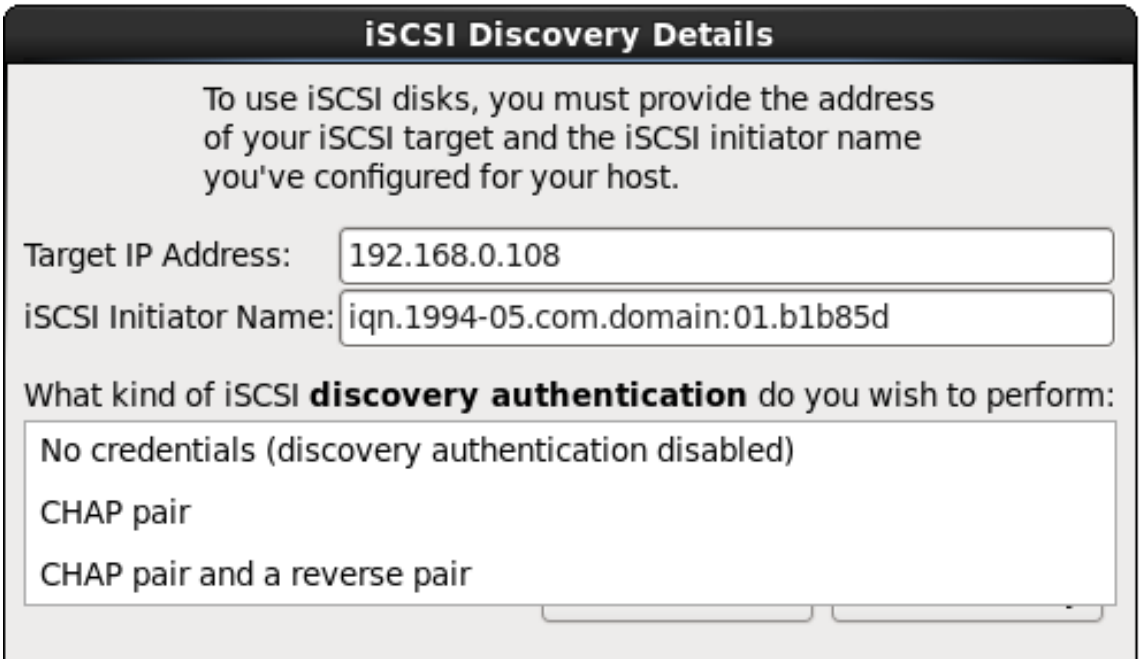


Figure 23.13. iSCSI discovery authentication

- no credentials
- CHAP pair
- CHAP pair and a reverse pair

4. - If you selected **CHAP pair** as the authentication type, provide the username and password for the iSCSI target in the **CHAP Username** and **CHAP Password** fields.

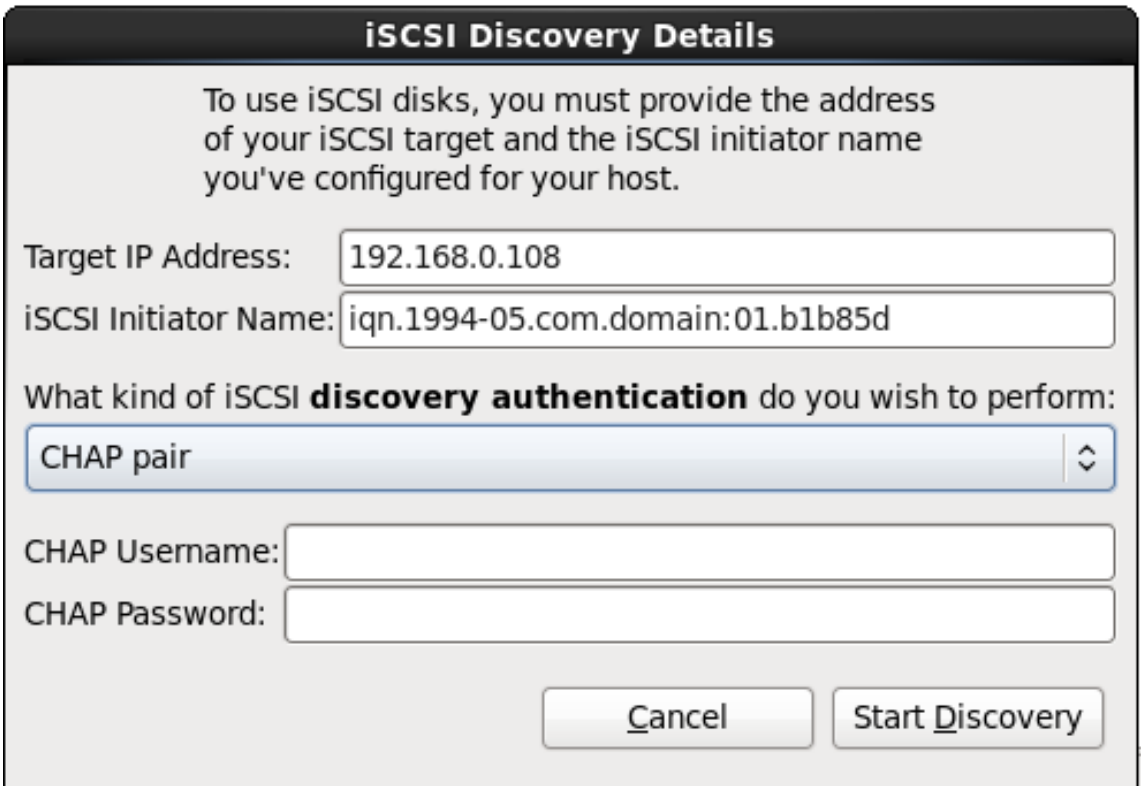


Figure 23.14. CHAP pair

- If you selected **CHAP pair and a reverse pair** as the authentication type, provide the

username and password for the iSCSI target in the **CHAP Username** and **CHAP Password** field and the username and password for the iSCSI initiator in the **Reverse CHAP Username** and **Reverse CHAP Password** fields.

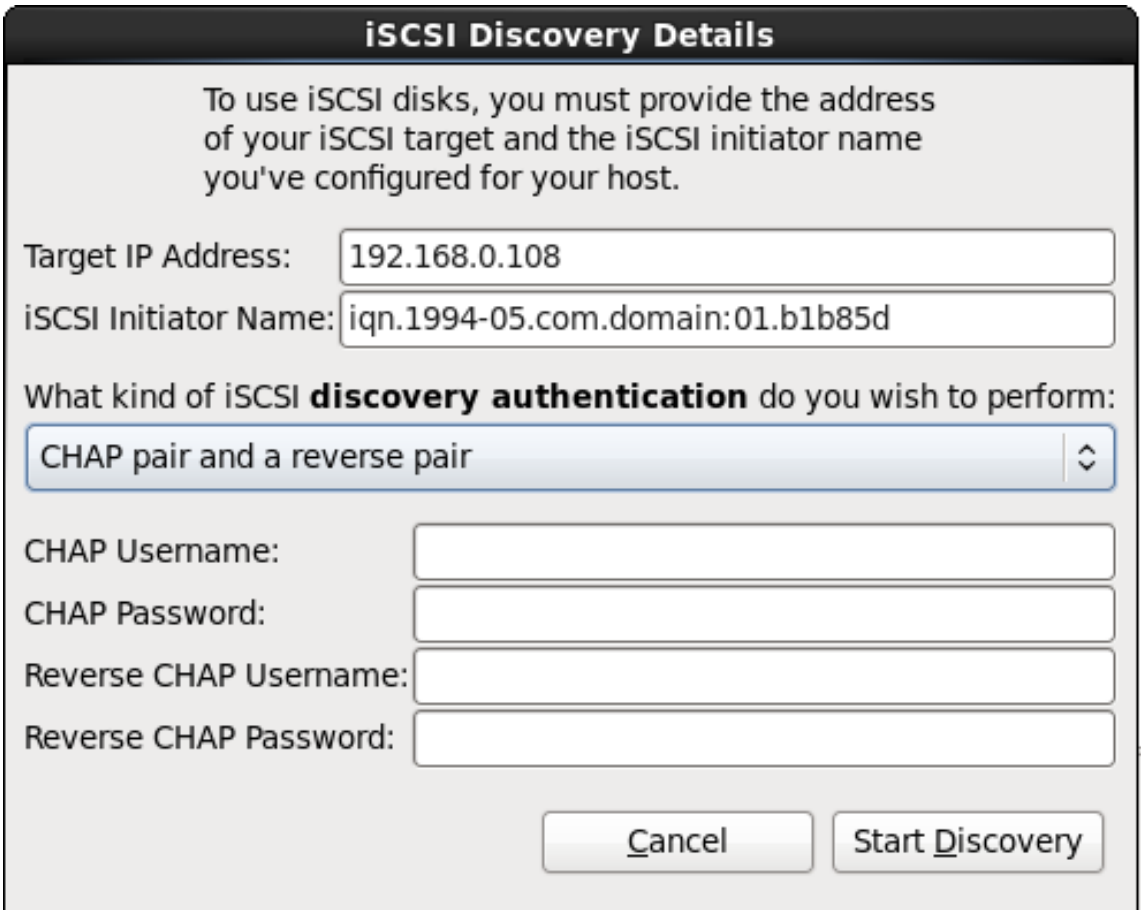


Figure 23.15. CHAP pair and a reverse pair

5. Click **Start Discovery**. **Anaconda** attempts to discover an iSCSI target based on the information that you provided. If discovery succeeds, the **iSCSI Discovered Nodes** dialog presents you with a list of all the iSCSI nodes discovered on the target.

6. Each node is presented with a checkbox beside it. Click the checkboxes to select the nodes to use for installation.

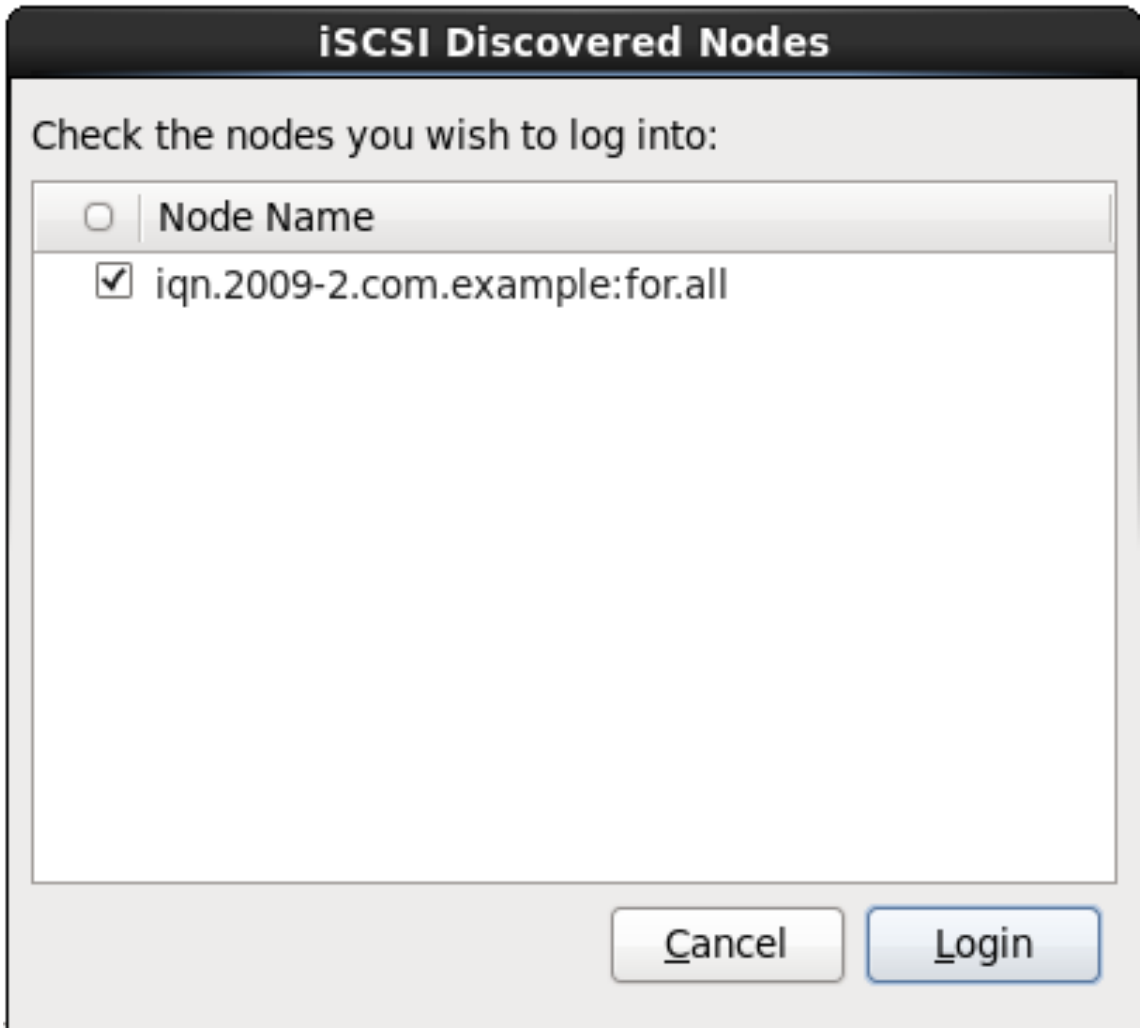


Figure 23.16. The iSCSI Discovered Nodes dialog

7. Click **Login** to initiate an iSCSI session.

Procedure 23.2. Starting an iSCSI session

Use the **iSCSI Nodes Login** dialog to provide **anaconda** with the information that it needs to log into the nodes on the iSCSI target and start an iSCSI session.

Figure 23.17. The iSCSI Nodes Login dialog

1. Use the drop-down menu to specify the type of authentication to use for the iSCSI session:

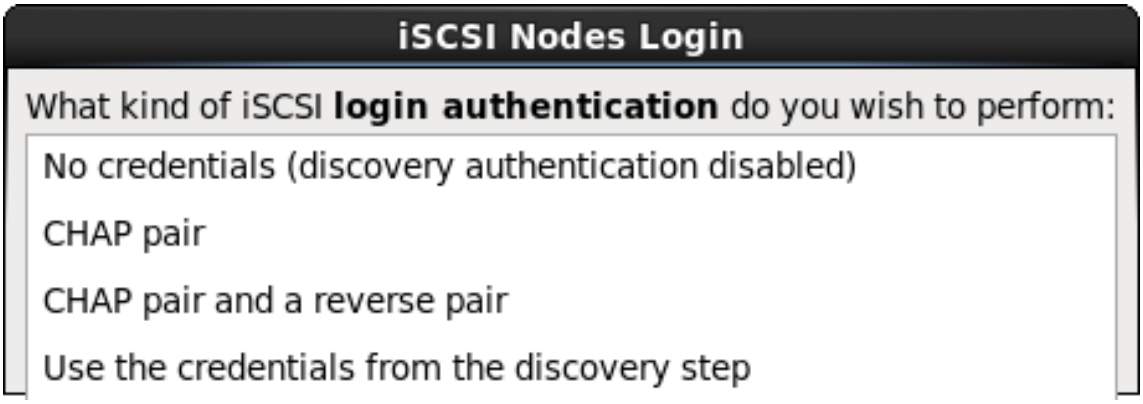


Figure 23.18. iSCSI session authentication

- **no credentials**
- **CHAP pair**
- **CHAP pair and a reverse pair**
- **Use the credentials from the discovery step**

If your environment uses the same type of authentication and same username and password for iSCSI discovery and for the iSCSI session, select **Use the credentials from the discovery step** to reuse these credentials.

2. - If you selected **CHAP pair** as the authentication type, provide the username and password for the iSCSI target in the **CHAP Username** and **CHAP Password** fields.

Figure 23.19. CHAP pair

- If you selected **CHAP pair and a reverse pair** as the authentication type, provide the username and password for the iSCSI target in the **CHAP Username** and **CHAP Password** fields and the username and password for the iSCSI initiator in the **Reverse CHAP Username** and **Reverse CHAP Password** fields.

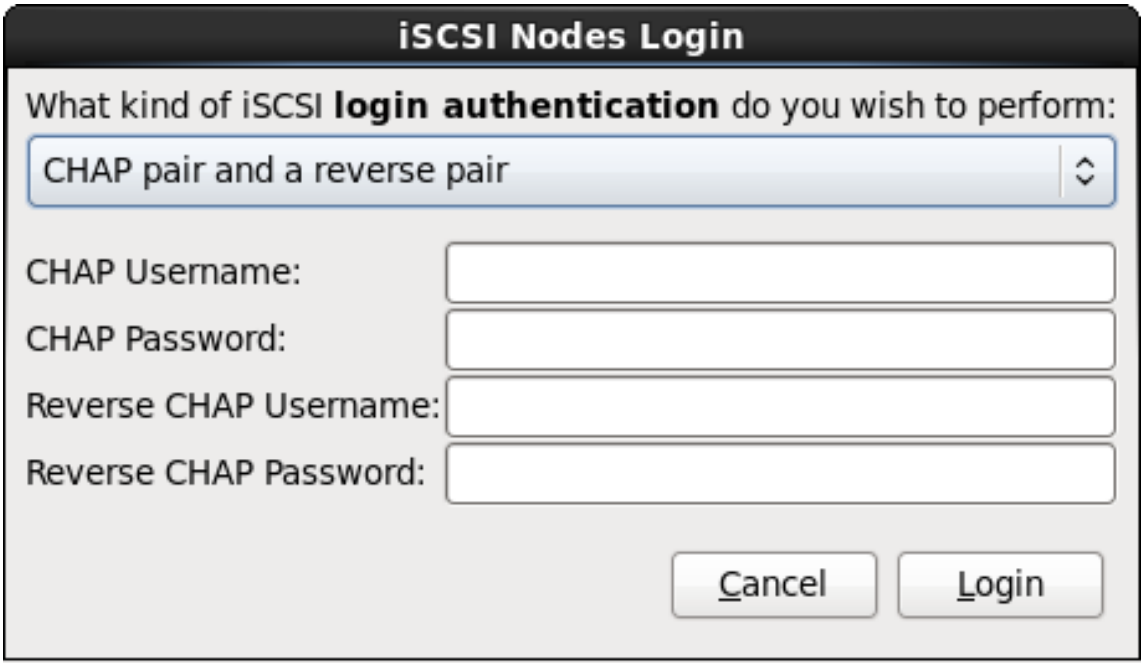


Figure 23.20. CHAP pair and a reverse pair

3. Click **Login**. **Anaconda** attempts to log into the nodes on the iSCSI target based on the information that you provided. The **iSCSI Login Results** dialog presents you with the results.

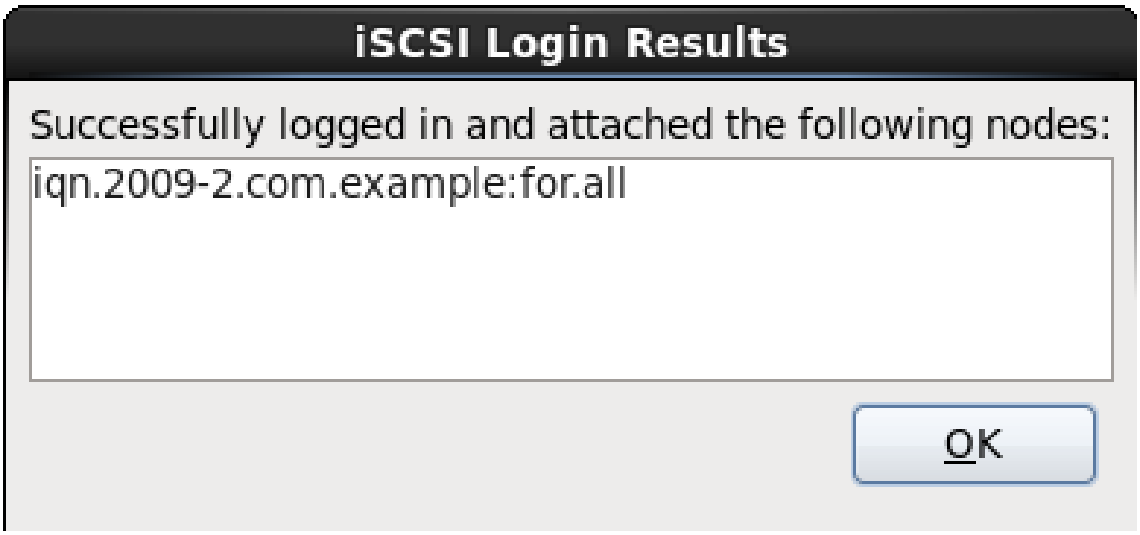


Figure 23.21. The iSCSI Login Results dialog

4. Click **OK** to continue.

23.6.1.2.2. FCP Devices

FCP devices enable IBM System z to use SCSI devices rather than, or in addition to, DASD devices. FCP devices provide a switched fabric topology that enables System z systems to use SCSI LUNs as disk devices in addition to traditional DASD devices.

IBM System z requires that any FCP device be entered manually (either in the installation program interactively, or specified as unique parameter entries in the parameter or CMS configuration file) for the installation program to activate FCP LUNs. The values entered here are unique to each site in which they are set up.

Notes

- Interactive creation of an FCP device is only possible in graphical mode. It is not possible to interactively configure an FCP device in a text-only install.
- Each value entered should be verified as correct, as any mistakes made may cause the system not to operate properly. Use only lower-case letters in hex values.
- For more information on these values, refer to the hardware documentation check with the system administrator who set up the network for this system.

To configure a Fiber Channel Protocol SCSI device, select **Add ZFCP LUN** and click **Add Drive**. In the **Add FCP device** dialog, fill in the details for the 16-bit device number, 64-bit World Wide Port Number (WWPN) and 64-bit FCP LUN. Click the **Add** button to connect to the FCP device using this information.

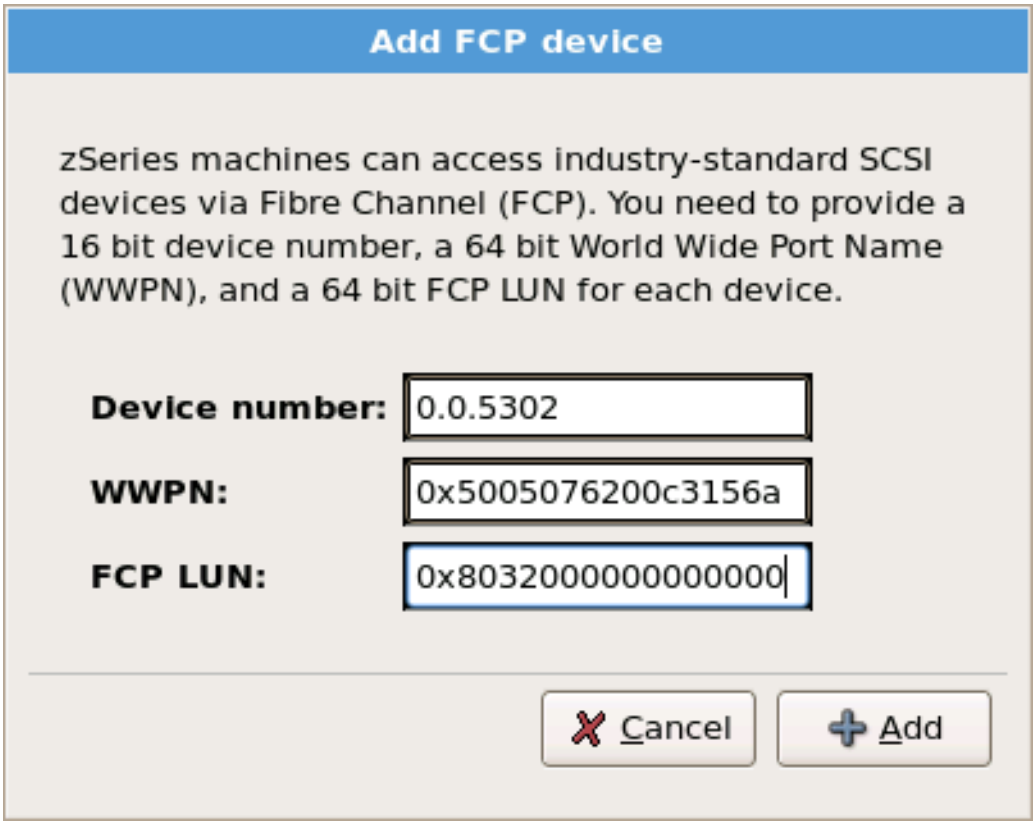


Figure 23.22. Add FCP Device

The newly added device should then be present and usable in the storage device selection screen on the **Multipath Devices** tab, if you have activated more than one path to the same LUN, or on **Other SAN Devices**, if you have activated only one path to the LUN.

IMPORTANT

The installer requires the definition of a DASD. For a SCSI-only installation, enter **none** as the parameter interactively during phase 1 of an interactive installation, or add **DASD=none** in the parameter or CMS configuration file. This satisfies the requirement for a defined DASD parameter, while resulting in a SCSI-only environment.

23.7. SETTING THE HOSTNAME

Setup prompts you to supply a host name for this computer, either as a *fully-qualified domain name*

(FQDN) in the format *hostname.domainname* or as a *short host name* in the format *hostname*. 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 machine, specify the short host name only.

NOTE

You may give your system any name provided that the full hostname is unique. The hostname may include letters, numbers and hyphens.

Change the default setting *localhost.localdomain* to a unique hostname for each of your Linux instances.

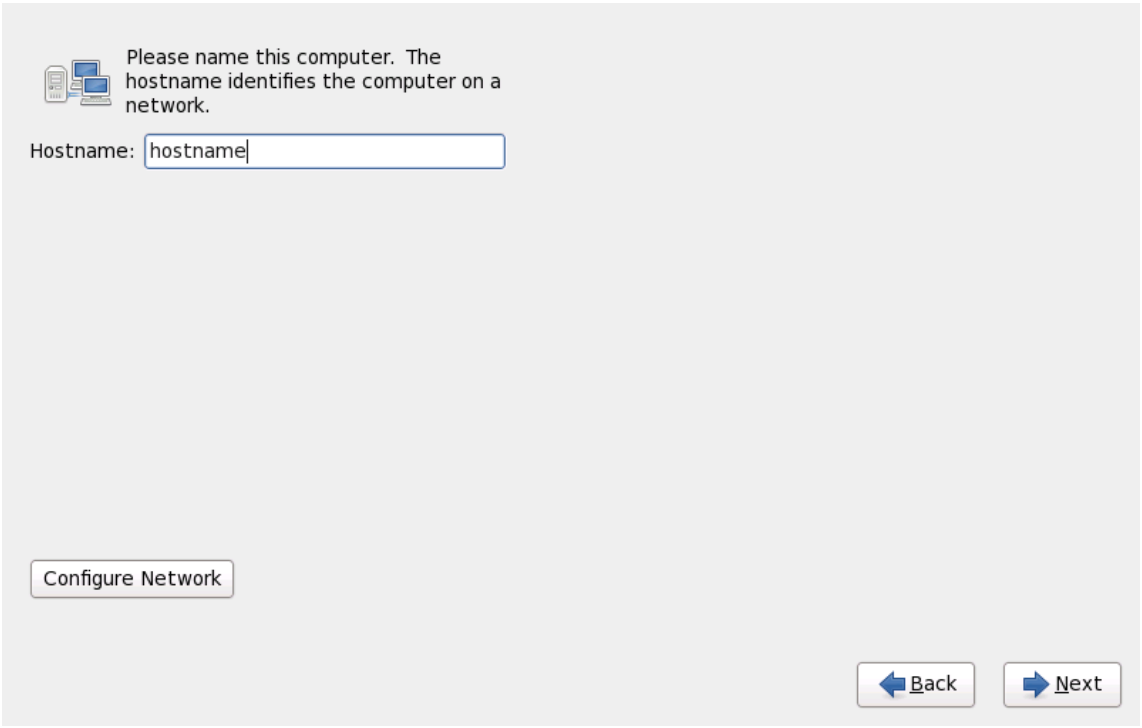
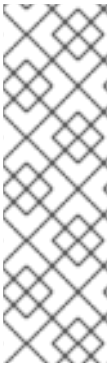


Figure 23.23. Setting the hostname

23.7.1. Editing Network Connections



NOTE

To change your network configuration after you have completed the installation, use the **Network Administration Tool**.

Type the **system-config-network** command in a shell prompt to launch the **Network Administration Tool**. If you are not root, it prompts you for the root password to continue.

The **Network Administration Tool** is now deprecated and will be replaced by **NetworkManager** during the lifetime of Red Hat Enterprise Linux 6.

Usually, the network connection configured earlier in installation phase 1 does not need to be modified during the rest of the installation. You cannot add a new connection on System z because the network subchannels need to be grouped and set online beforehand, and this is currently only done in installation

phase 1. To change the existing network connection, click the button **Configure Network**. The **Network Connections** dialog appears that allows you to configure network connections for the system, not all of which are relevant to System z.

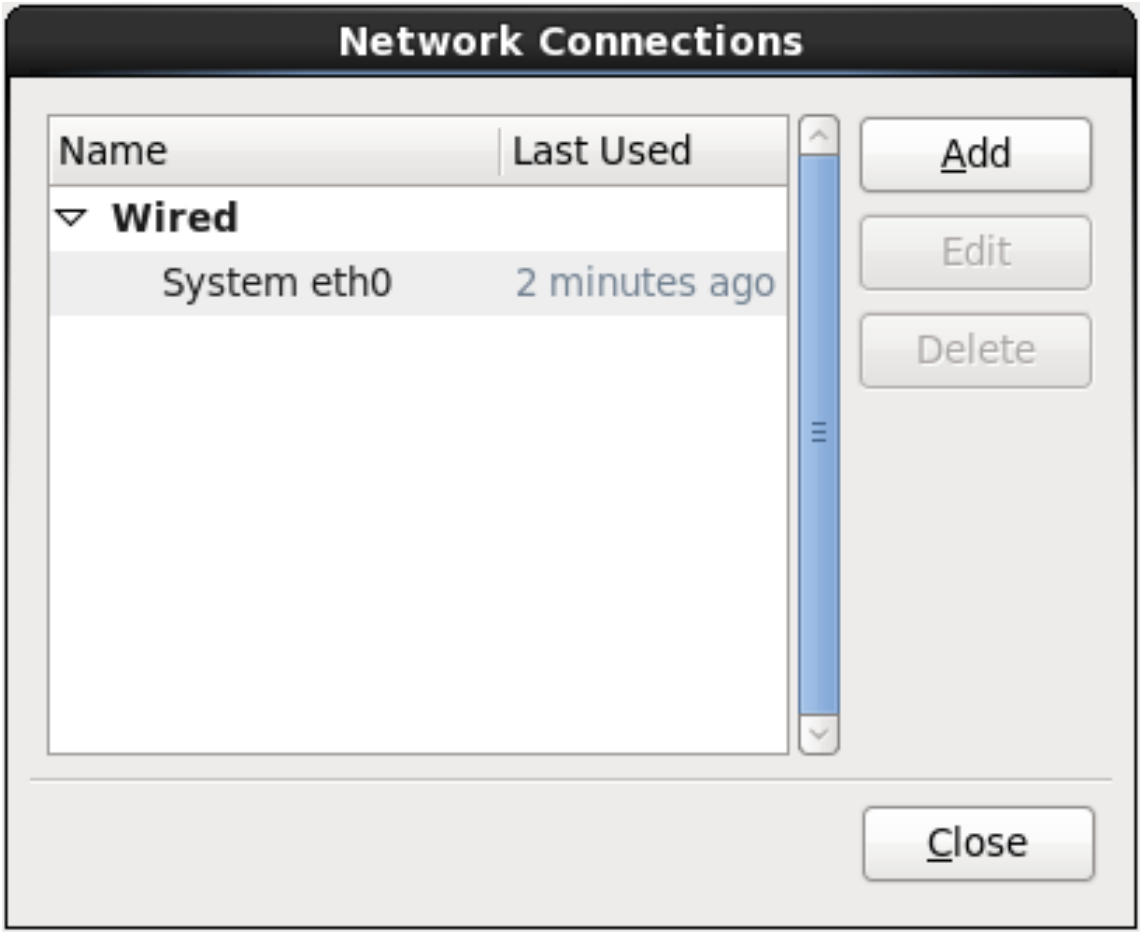


Figure 23.24. Network Connections

All network connections on System z are listed in the **Wired** tab. By default this contains the connection configured earlier in installation phase 1 and is either **eth0** (OSA, LCS), or **hsi0** (HiperSockets). Note that on System z you cannot add a new connection here. To modify an existing connection, select a row in the list and click the **Edit** button. A dialog box appears with a set of tabs appropriate to wired connections, as described below.

The most important tabs on System z are **Wired** and **IPv4 Settings**.

When you have finished editing network settings, click **Apply** to save the new configuration. If you reconfigured a device that was already active during installation, you must restart the device to use the new configuration – refer to Section 9.7.1.6, “Restart a network device” .

23.7.1.1. Options common to all types of connection

Certain configuration options are common to all connection types.

Specify a name for the connection in the **Connection name** name field.

Select **Connect automatically** to start the connection automatically when the system boots.

When **NetworkManager** runs on an installed system, the **Available to all users** option controls whether a network configuration is available system-wide or not. During installation, ensure that **Available to all users** remains selected for any network interface that you configure.

23.7.1.2. The Wired tab

Use the **Wired** tab to specify or change the *media access control* (MAC) address for the network adapter, and either set the *maximum transmission unit* (MTU, in bytes) that can pass through the interface.

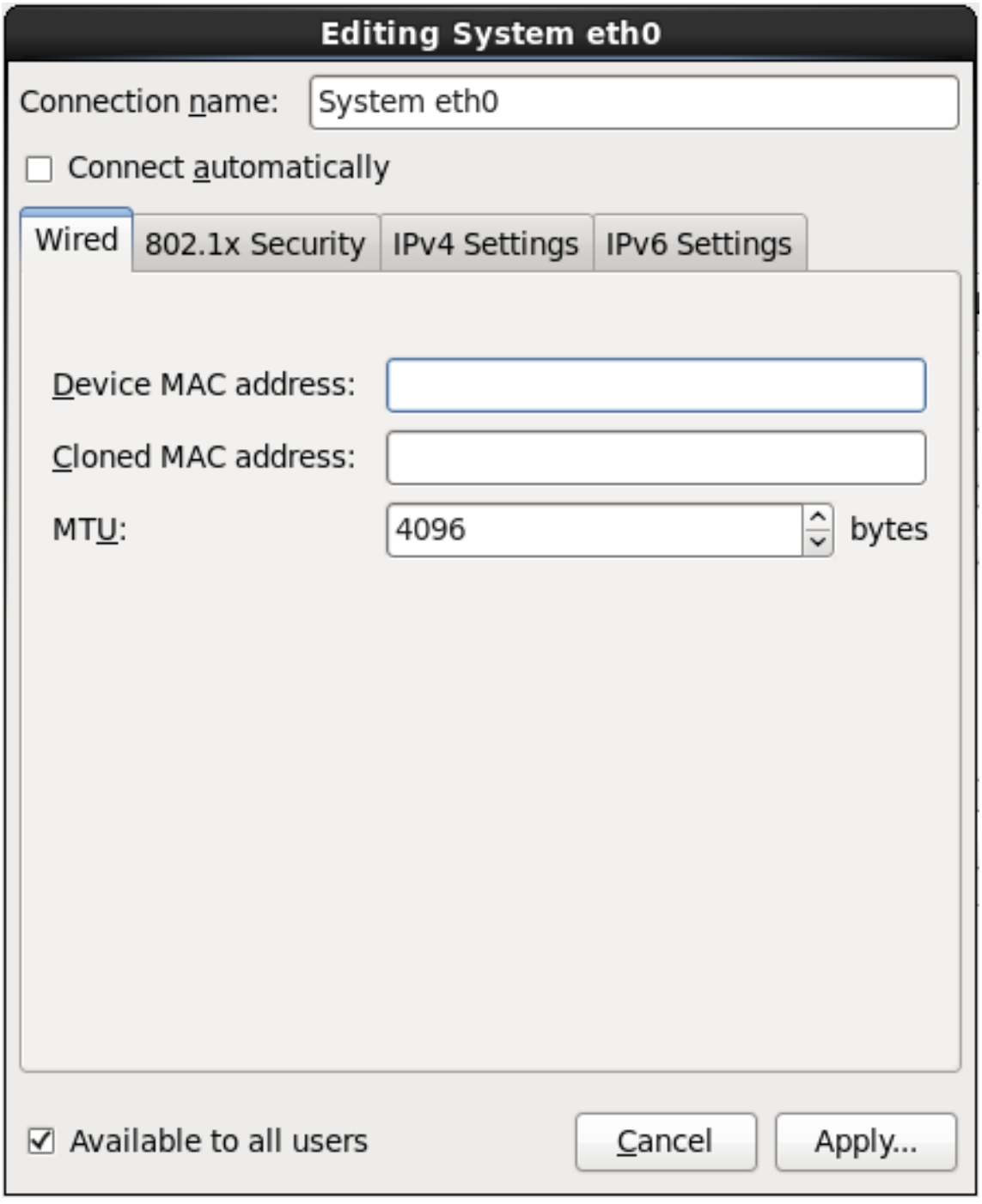


Figure 23.25. The Wired tab

23.7.1.3. The 802.1x Security tab

Use the **802.1x Security** tab to configure 802.1X *port-based network access control* (PNAC). Select **Use 802.1X security for this connection** to enable access control, then specify details of your network. The configuration options include:

Authentication

Choose one of the following methods of authentication:

- **TLS** for *Transport Layer Security*
- **Tunneled TLS** for *Tunneled Transport Layer Security*, otherwise known as TTLS, or EAP-TTLS
- **Protected EAP (PEAP)** for *Protected Extensible Authentication Protocol*

Identity

Provide the identity of this server.

User certificate

Browse to a personal X.509 certificate file encoded with *Distinguished Encoding Rules* (DER) or *Privacy Enhanced Mail* (PEM).

CA certificate

Browse to a X.509 *certificate authority* certificate file encoded with *Distinguished Encoding Rules* (DER) or *Privacy Enhanced Mail* (PEM).

Private key

Browse to a *private key* file encoded with *Distinguished Encoding Rules* (DER), *Privacy Enhanced Mail* (PEM), or the *Personal Information Exchange Syntax Standard* (PKCS#12).

Private key password

The password for the private key specified in the **Private key** field. Select **Show password** to make the password visible as you type it.

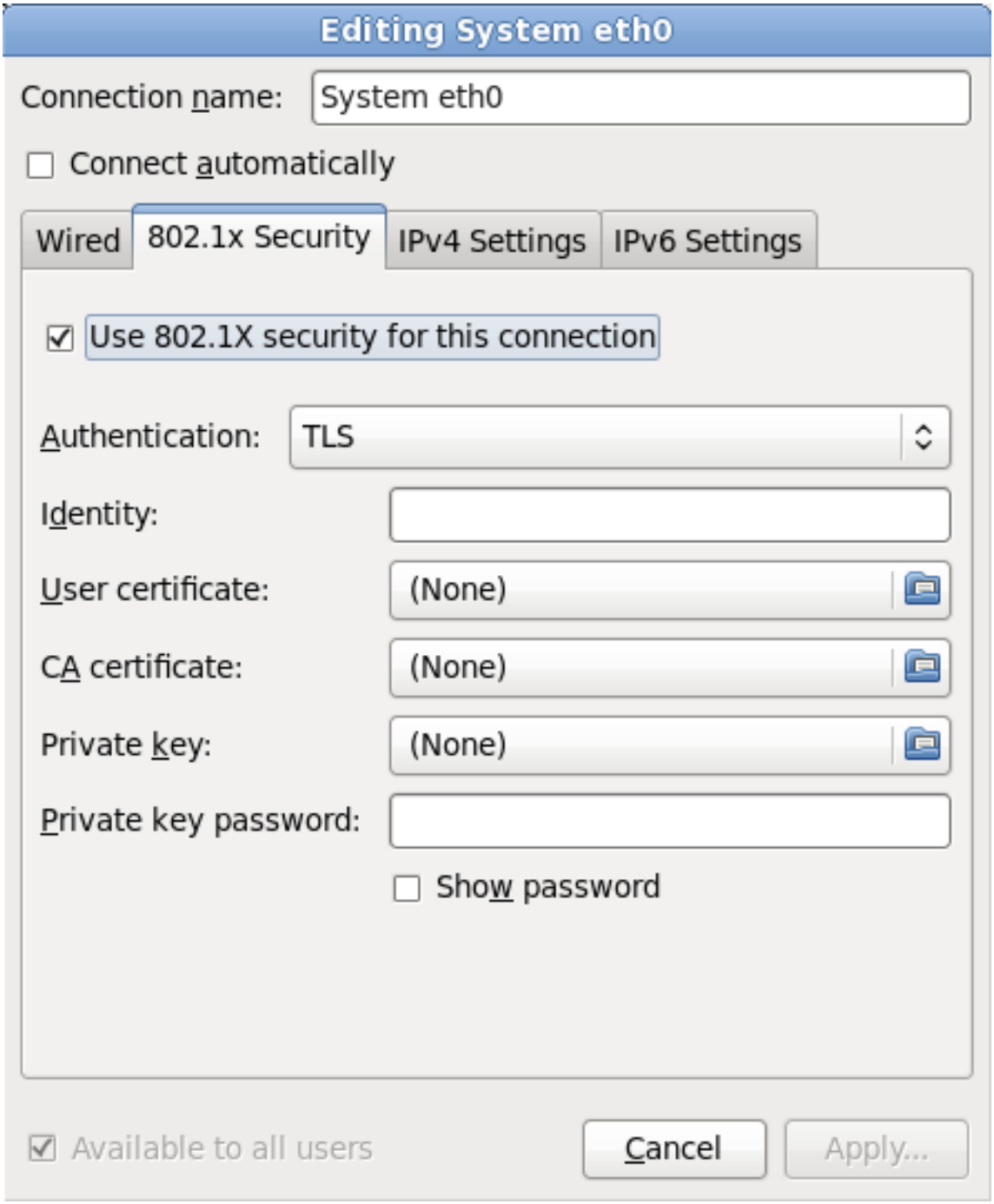


Figure 23.26. The 802.1x Security tab

23.7.1.4. The IPv4 Settings tab

Use the **IPv4 Settings tab** tab to configure the IPv4 parameters for the previously selected network connection.

The address, netmask, gateway, DNS servers and DNS search suffix for an IPv4 connection were configured during installation phase 1 or reflect the following parameters in the parameter file or configuration file: ***IPADDR***, ***NETMASK***, ***GATEWAY***, ***DNS***, ***SEARCHDNS*** (Refer to Section 26.3, "Installation Network Parameters").

Use the **Method** drop-down menu to specify which settings the system should attempt to obtain from a *Dynamic Host Configuration Protocol* (DHCP) service running on the network. Choose from the following options:

Automatic (DHCP)

IPv4 parameters are configured by the DHCP service on the network.

Automatic (DHCP) addresses only

The IPv4 address, netmask, and gateway address are configured by the DHCP service on the network, but DNS servers and search domains must be configured manually.

Manual

IPv4 parameters are configured manually for a static configuration.

Link-Local Only

A *link-local* address in the 169.254/16 range is assigned to the interface.

Shared to other computers

The system is configured to provide network access to other computers. The interface is assigned an address in the 10.42.x.1/24 range, a DHCP server and DNS server are started, and the interface is connected to the default network connection on the system with *network address translation* (NAT).

Disabled

IPv4 is disabled for this connection.

If you selected a method that requires you to supply manual parameters, enter details of the IP address for this interface, the netmask, and the gateway in the **Addresses** field. Use the **Add** and **Delete** buttons to add or remove addresses. Enter a comma-separated list of DNS servers in the **DNS servers** field, and a comma-separated list of domains in the **Search domains** field for any domains that you want to include in name server lookups.

Optionally, enter a name for this network connection in the **DHCP client ID** field. This name must be unique on the subnet. When you assign a meaningful DHCP client ID to a connection, it is easy to identify this connection when troubleshooting network problems.

Deselect the **Require IPv4 addressing for this connection to complete** check box to allow the system to make this connection on an IPv6-enabled network if IPv4 configuration fails but IPv6 configuration succeeds.

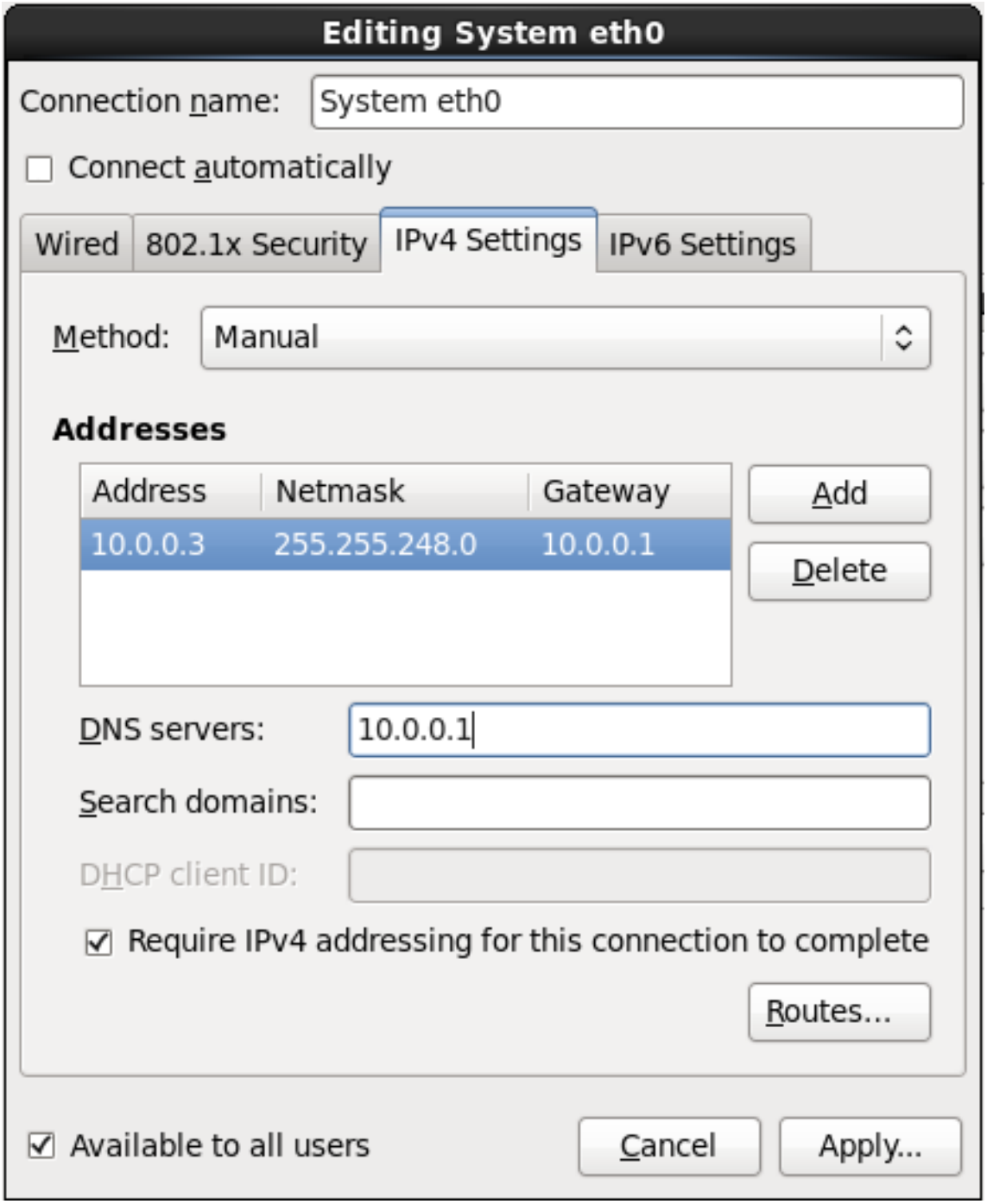


Figure 23.27. The IPv4 Settings tab

23.7.1.4.1. Editing IPv4 routes

Red Hat Enterprise Linux configures a number of routes automatically based on the IP addresses of a device. To edit additional routes, click the **Routes** button. The **Editing IPv4 routes** dialog appears.

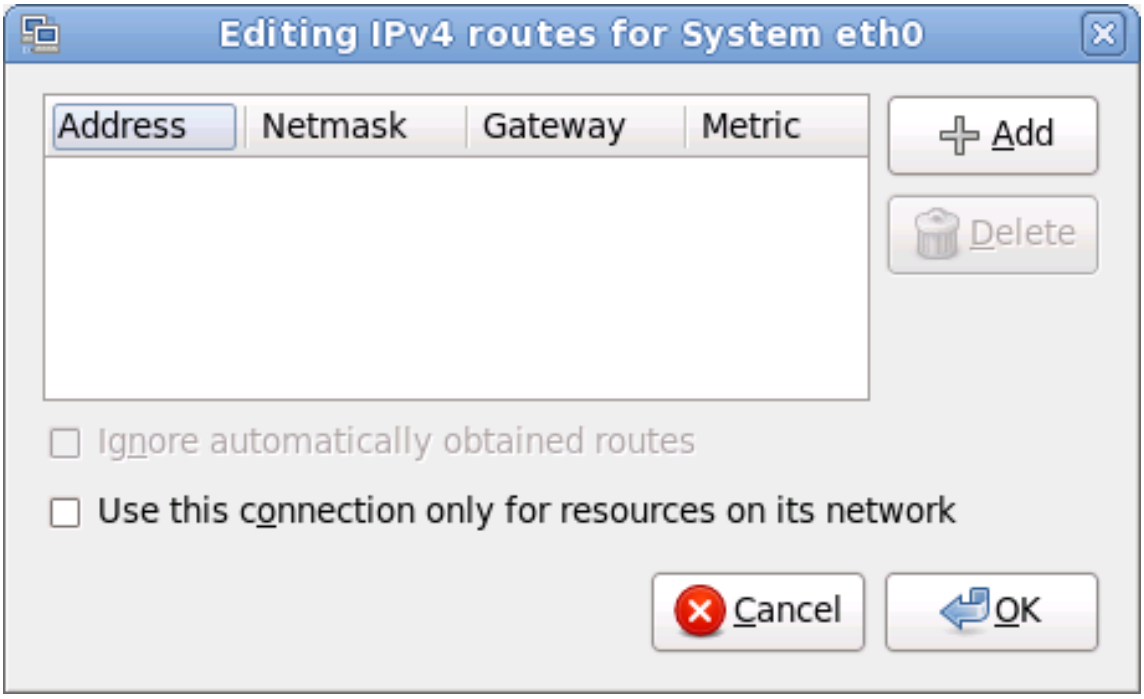


Figure 23.28. The Editing IPv4 Routes dialog

Click **Add** to add the IP address, netmask, gateway address, and metric for a new static route.

Select **Ignore automatically obtained routes** to make the interface use only the routes specified for it here.

Select **Use this connection only for resources on its network** to restrict connections only to the local network.

23.7.1.5. The IPv6 Settings tab

Use the **IPv6 Settings tab** tab to configure the IPv6 parameters for the previously selected network connection.

Use the **Method** drop-down menu to specify which settings the system should attempt to obtain from a *Dynamic Host Configuration Protocol* (DHCP) service running on the network. Choose from the following options:

Ignore

IPv6 is ignored for this connection.

Automatic

NetworkManager uses *router advertisement* (RA) to create an automatic, stateless configuration.

Automatic, addresses only

NetworkManager uses RA to create an automatic, stateless configuration, but DNS servers and search domains are ignored and must be configured manually.

Automatic, DHCP only

NetworkManager does not use RA, but requests information from DHCPv6 directly to create a stateful configuration.

Manual

IPv6 parameters are configured manually for a static configuration.

Link-Local Only

A *link-local* address with the fe80::/10 prefix is assigned to the interface.

If you selected a method that requires you to supply manual parameters, enter details of the IP address for this interface, the netmask, and the gateway in the **Addresses** field. Use the **Add** and **Delete** buttons to add or remove addresses. Enter a comma-separated list of DNS servers in the **DNS servers** field, and a comma-separated list of domains in the **Search domains** field for any domains that you want to include in name server lookups.

Optionally, enter a name for this network connection in the **DHCP client ID** field. This name must be unique on the subnet. When you assign a meaningful DHCP client ID to a connection, it is easy to identify this connection when troubleshooting network problems.

Deselect the **Require IPv6 addressing for this connection to complete** check box to allow the system to make this connection on an IPv4-enabled network if IPv6 configuration fails but IPv4 configuration succeeds.

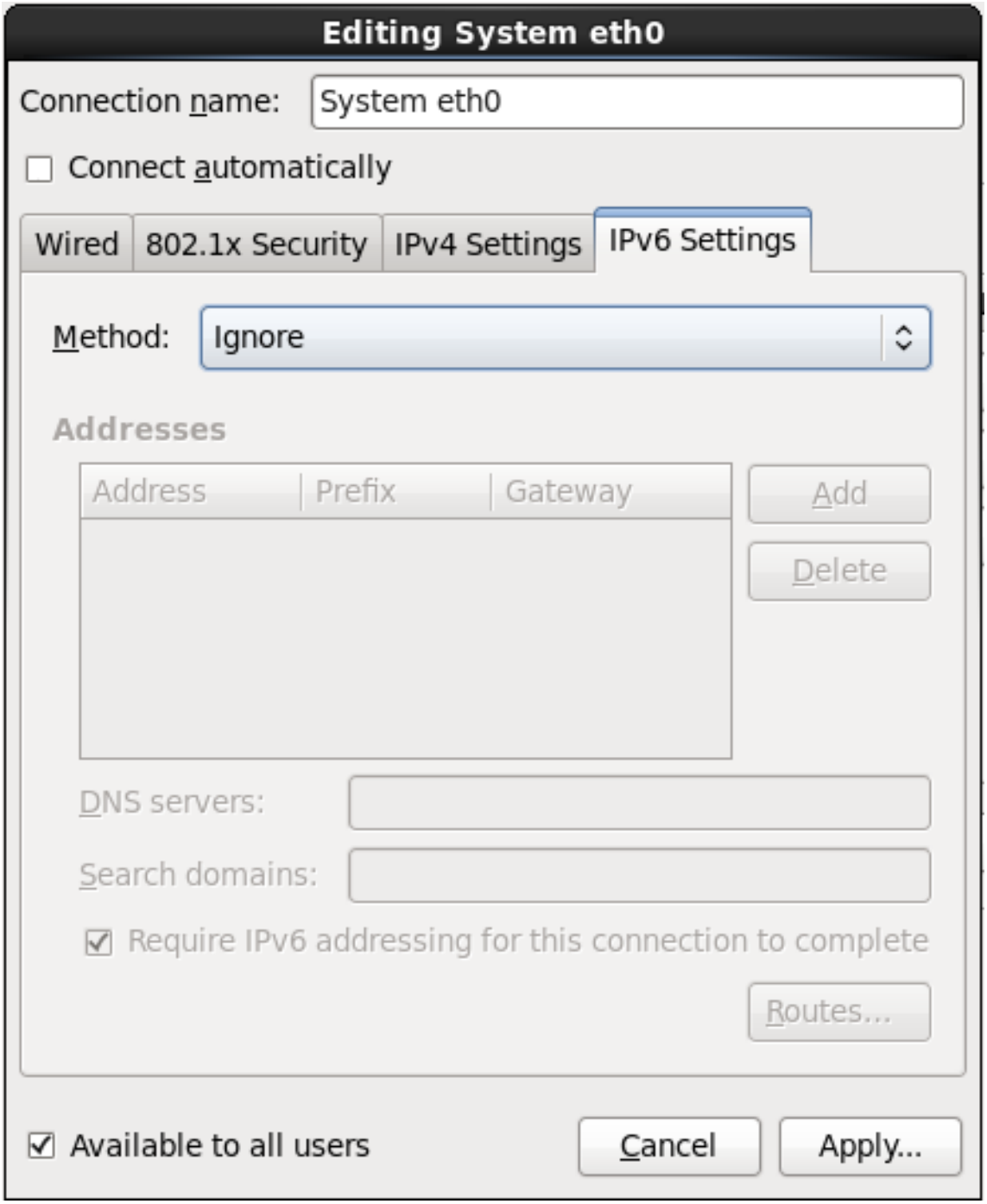


Figure 23.29. The IPv6 Settings tab

23.7.1.5.1. Editing IPv6 routes

Red Hat Enterprise Linux configures a number of routes automatically based on the IP addresses of a device. To edit additional routes, click the **Routes** button. The **Editing IPv6 routes** dialog appears.

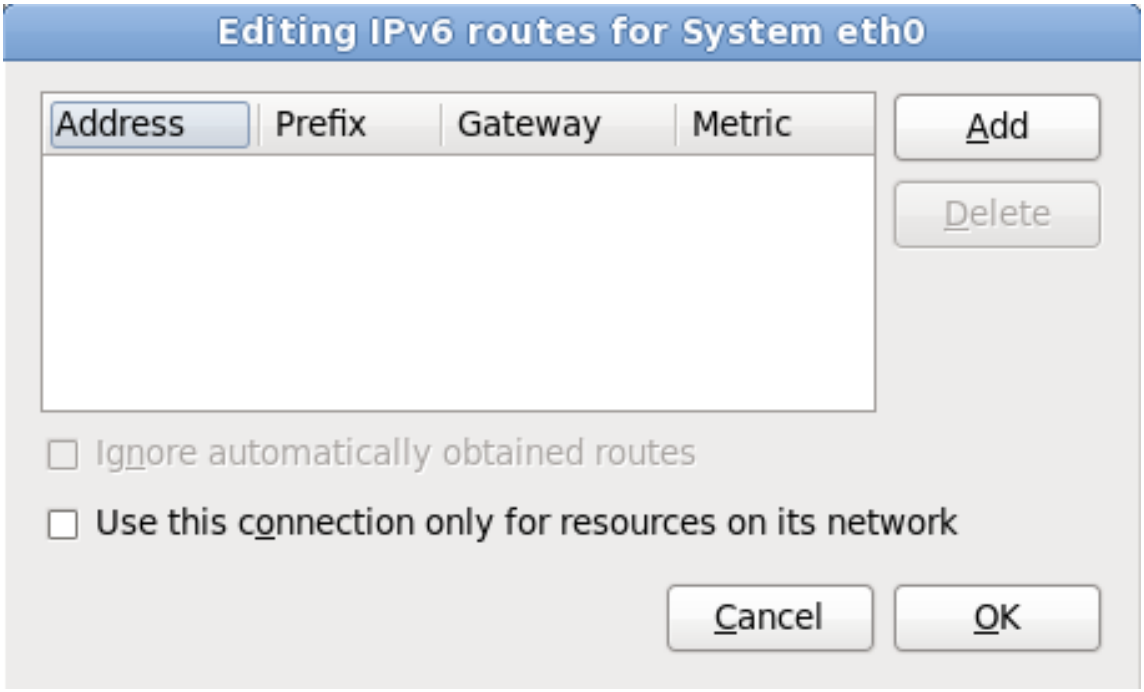


Figure 23.30. The Editing IPv6 Routes dialog

Click **Add** to add the IP address, netmask, gateway address, and metric for a new static route.

Select **Use this connection only for resources on its network** to restrict connections only to the local network.

23.7.1.6. Restart a network device

If you reconfigured a network that was already in use during installation, you must disconnect and reconnect the device in **anaconda** for the changes to take effect. **Anaconda** uses *interface configuration* (ifcfg) files to communicate with **NetworkManager**. A device becomes disconnected when its ifcfg file is removed, and becomes reconnected when its ifcfg file is restored, as long as **ONBOOT=yes** is set. Refer to the *Red Hat Enterprise Linux 6.9 Deployment Guide* available from https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/index.html for more information about interface configuration files.

1. Press **Ctrl**+**Alt**+**F2** to switch to virtual terminal **tty2**.

2. Move the interface configuration file to a temporary location:

   ```
   mv /etc/sysconfig/network-scripts/ifcfg-device_name /tmp
   ```

   where *device_name* is the device that you just reconfigured. For example, **ifcfg-eth0** is the ifcfg file for **eth0**.

   The device is now disconnected in **anaconda**.

3. Open the interface configuration file in the **vi** editor:

   ```
   vi /tmp/ifcfg-device_name
   ```

4. Verify that the interface configuration file contains the line **ONBOOT=yes**. If the file does not already contain the line, add it now and save the file.

5. Exit the **vi** editor.

6. Move the interface configuration file back to the **/etc/sysconfig/network-scripts/** directory:

   ```
   mv /tmp/ifcfg-device_name /etc/sysconfig/network-scripts/
   ```

   The device is now reconnected in **anaconda**.

7. Press **Ctrl**+**Alt**+**F6** to return to **anaconda**.

23.8. TIME ZONE CONFIGURATION

Set your time zone by selecting the city closest to your computer's physical location. Click on the map to zoom in to a particular geographical region of the world.

Specify a time zone even if you plan to use NTP (Network Time Protocol) to maintain the accuracy of the system clock.

From here there are two ways for you to select your time zone:

- Using your mouse, click on the interactive map to select a specific city (represented by a yellow dot). A red **X** appears indicating your selection.
- You can also scroll through the list at the bottom of the screen to select your time zone. Using your mouse, click on a location to highlight your selection.

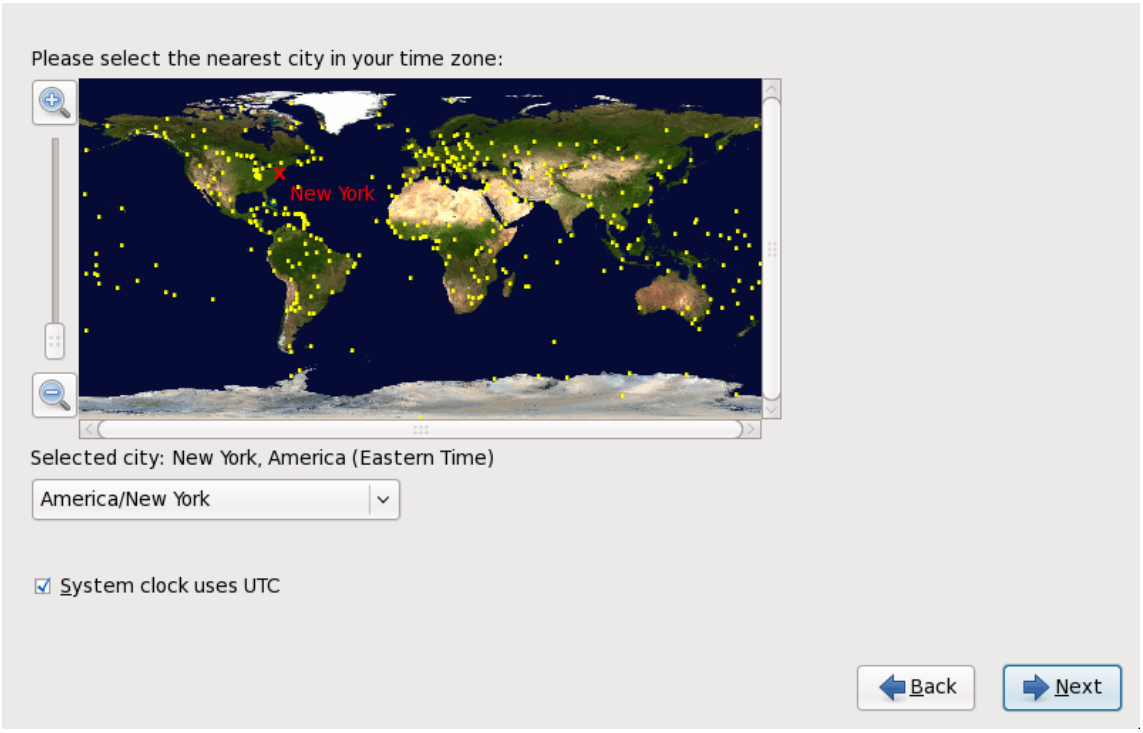


Figure 23.31. Configuring the Time Zone

Select **System clock uses UTC**. The system clock is a piece of hardware on your computer system. Red Hat Enterprise Linux uses the timezone setting to determine the offset between the local time and UTC on the system clock. This behavior is standard for systems that use UNIX, Linux, and similar operating systems.

Click **Next** to proceed.

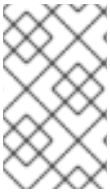
NOTE

To change your time zone configuration after you have completed the installation, use the **Time and Date Properties Tool**.

Type the **system-config-date** command in a shell prompt to launch the **Time and Date Properties Tool**. If you are not root, it prompts you for the root password to continue.

23.9. SET THE ROOT PASSWORD

Setting up a root account and password is one of the most important steps during your installation. The root account is used to install packages, upgrade RPMs, and perform most system maintenance. Logging in as root gives you complete control over your system.



NOTE

The root user (also known as the superuser) has complete access to the entire system; for this reason, logging in as the root user is best done *only* to perform system maintenance or administration.

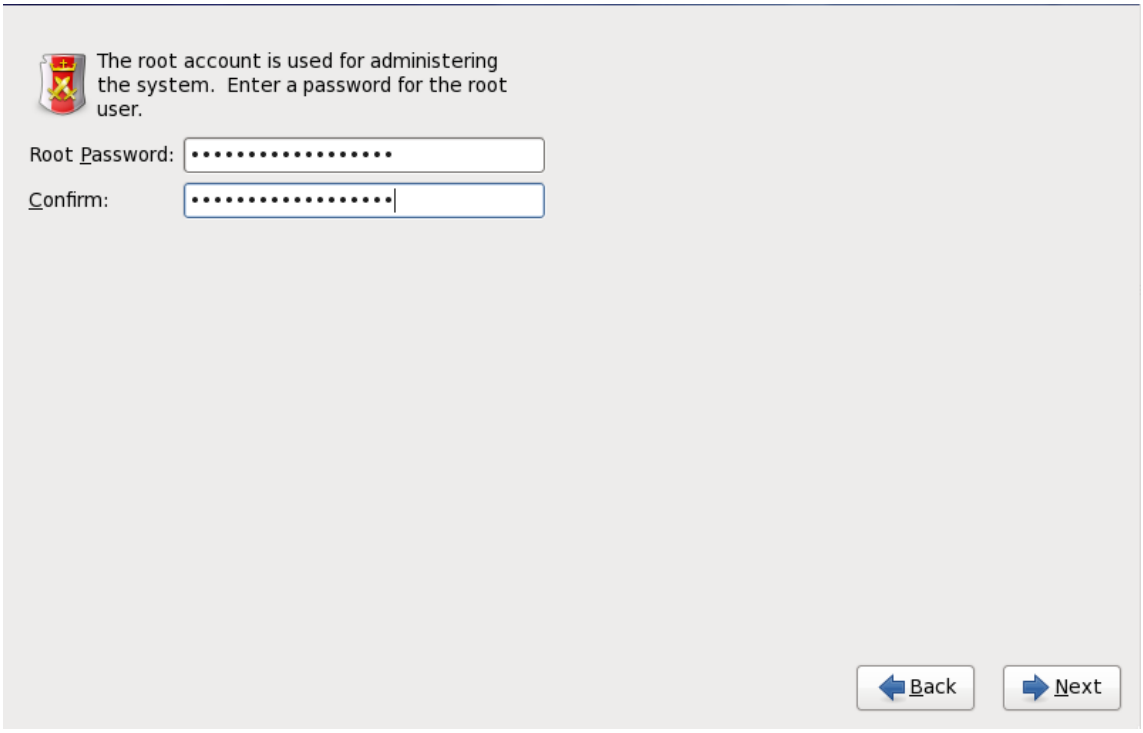


Figure 23.32. Root Password

Use the root account only for system administration. Create a non-root account for your general use and use the **su** command to change to root only when you need to perform tasks that require superuser authorization. These basic rules minimize the chances of a typo or an incorrect command doing damage to your system.

NOTE

To become root, type **su -** at the shell prompt in a terminal window and then press **Enter**. Then, enter the root password and press **Enter**.

The installation program prompts you to set a root password[11] for your system. . *You cannot proceed to the next stage of the installation process without entering a root password.*

The root password must be at least six characters long; the password you type is not echoed to the screen. You must enter the password twice; if the two passwords do not match, the installation program asks you to enter them again.

You should make the root password something you can remember, but not something that is easy for someone else to guess. Your name, your phone number, *qwerty*, *password, root*, *123456*, and *anteater* are all examples of bad passwords. Good passwords mix numerals with upper and lower case letters and do not contain dictionary words: *Aard387vark* or *420BMttNT*, for example. Remember that the password is case-sensitive. If you write down your password, keep it in a secure place. However, it is recommended that you do not write down this or any password you create.

WARNING

Do not use one of the example passwords offered in this manual. Using one of these passwords could be considered a security risk.

To change your root password after you have completed the installation, run the **passwd** command as **root**. If you forget the root password, see Resolving Problems in System Recovery Modes in the Red Hat Enterprise Linux 6 Deployment Guide for instructions on how to set a new one.

23.10. ASSIGN STORAGE DEVICES

If you selected more than one storage device on the storage devices selection screen (refer to Section 23.6, "Storage Devices"), **anaconda** asks you to select which of these devices should be available for installation of the operating system, and which should only be attached to the file system for data storage.

During installation, the devices that you identify here as being for data storage only are mounted as part of the file system, but are not partitioned or formatted.

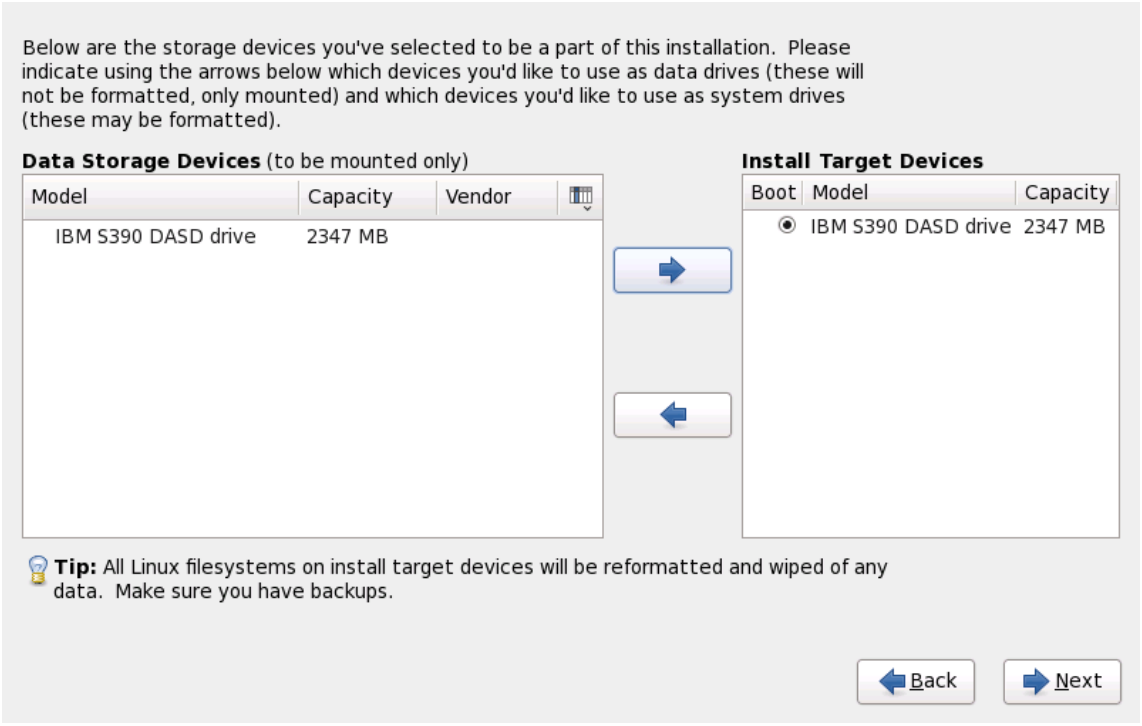


Figure 23.33. Assign storage devices

The screen is split into two panes. The left pane contains a list of devices to be used for data storage only. The right pane contains a list of devices that are to be available for installation of the operating system.

Each list contains information about the devices to help you to identify them. A small drop-down menu marked with an icon is located to the right of the column headings. This menu allows you to select the types of data presented on each device. Reducing or expanding the amount of information presented might help you to identify particular devices.

Move a device from one list to the other by clicking on the device, then clicking either the button labeled with a left-pointing arrow to move it to the list of data storage devices or the button labeled with a right-pointing arrow to move it to the list of devices available for installation of the operating system.

The list of devices available as installation targets also includes a radio button beside each device. On platforms other than System z, this radio button is used to specify the device to which you want to install the boot loader. On System z this choice does not have any effect. The **zipl** boot loader will be installed on the disk that contains the **/boot** directory, which is determined later on during partitioning.

When you have finished identifying devices to be used for installation, click **Next** to continue.

23.11. INITIALIZING THE HARD DISK

If no readable partition tables are found on existing hard disks, the installation program asks to initialize the hard disk. This operation makes any existing data on the hard disk unreadable. If your system has a brand new hard disk with no operating system installed, or you have removed all partitions on the hard disk, click **Re-initialize drive**.

The installation program presents you with a separate dialog for each disk on which it cannot read a valid partition table. Click the **Ignore all** button or **Re-initialize all** button to apply the same answer to all devices.

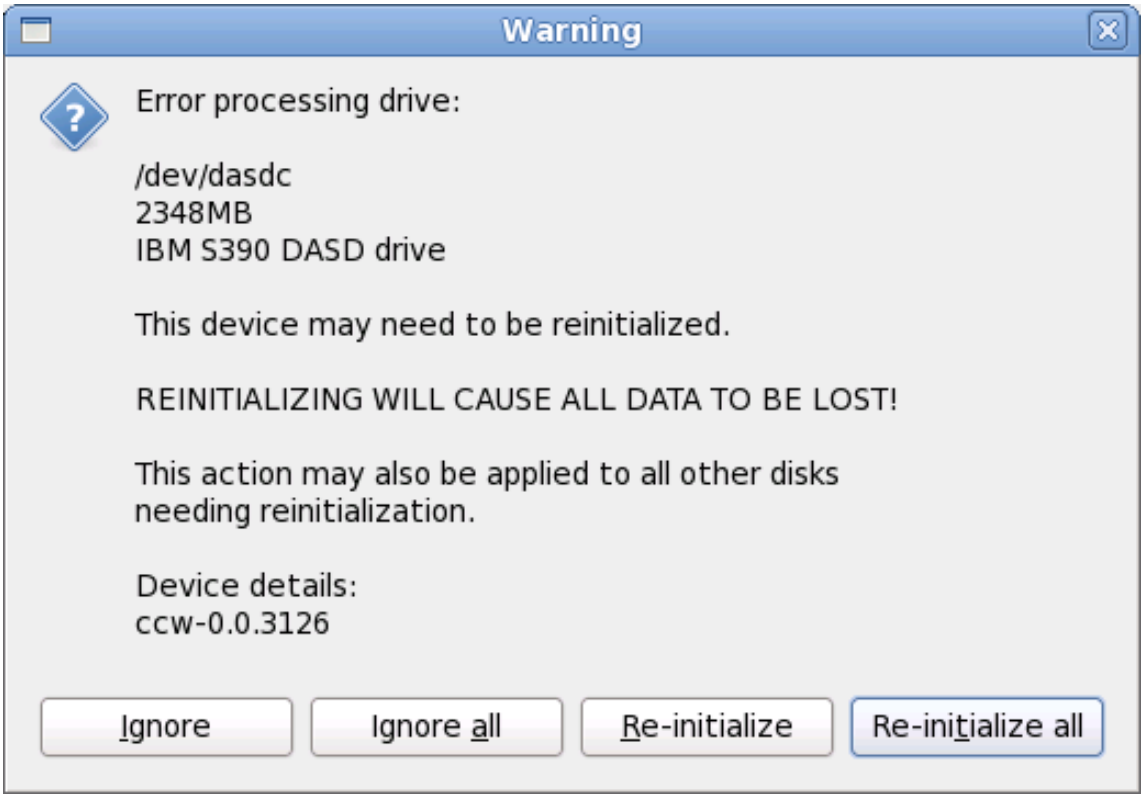


Figure 23.34. Warning screen – initializing DASD

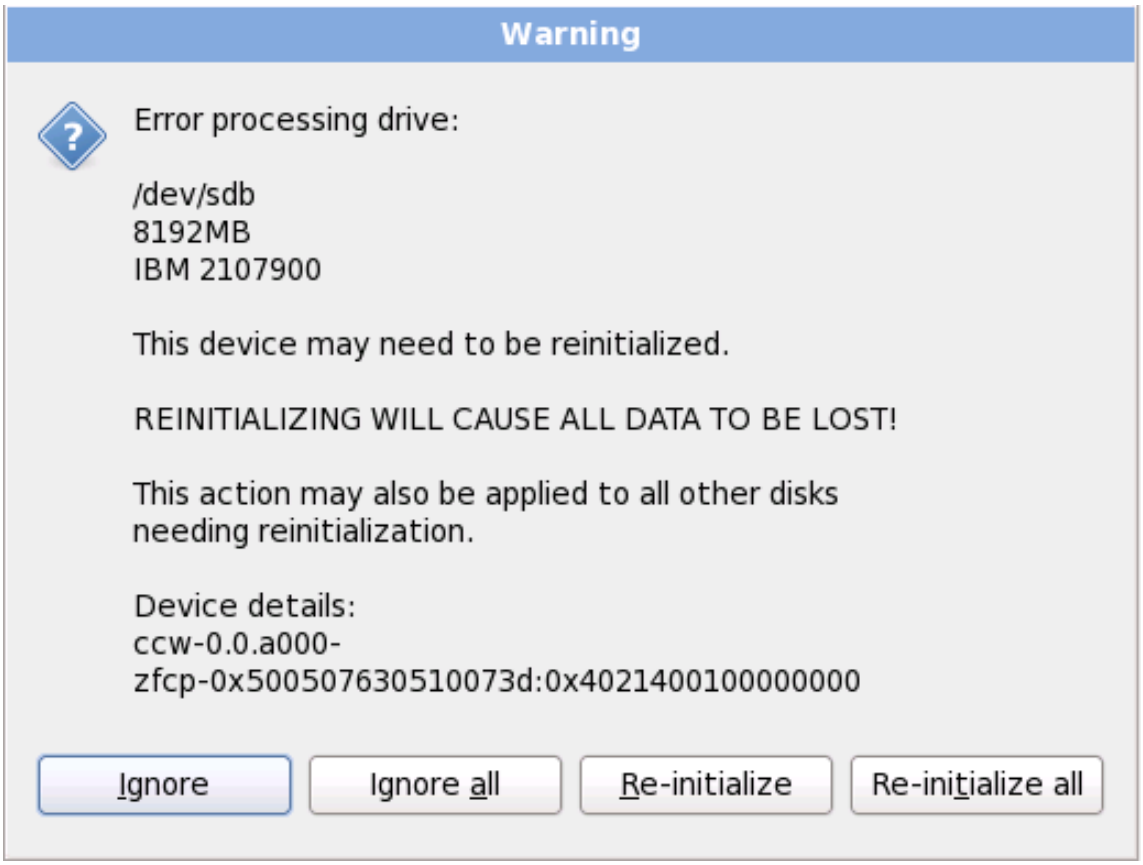


Figure 23.35. Warning screen – initializing FCP LUN

Certain RAID systems or other nonstandard configurations may be unreadable to the installation program and the prompt to initialize the hard disk may appear. The installation program responds to the physical disk structures it is able to detect.

To enable automatic initializing of hard disks for which it turns out to be necessary, use the kickstart command **zerombr** (refer to Chapter 32, *Kickstart Installations*). This command is required when performing an unattended installation on a system with previously initialized disks.

WARNING

If you have a nonstandard disk configuration that can be detached during installation and detected and configured afterward, power off the system, detach it, and restart the installation.

23.12. UPGRADING AN EXISTING SYSTEM

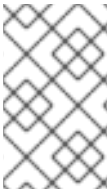
IMPORTANT

The following sections only apply to upgrading Red Hat Enterprise Linux between minor versions, for example, upgrading Red Hat Enterprise Linux 6.4 to Red Hat Enterprise Linux 6.5 or higher. This approach is not supported for upgrades between major versions, for example, upgrading Red Hat Enterprise Linux 6 to Red Hat Enterprise Linux 7.

In-place upgrades between major versions of Red Hat Enterprise Linux can be done, with certain limitations, using the **Red Hat Upgrade Tool** and **Preupgrade Assistant** tools. See Chapter 37, *Upgrading Your Current System* for more information.

The installation system automatically detects any existing installation of Red Hat Enterprise Linux. The upgrade process updates the existing system software with new versions, but does not remove any data from users' home directories. The existing partition structure on your hard drives does not change. Your system configuration changes only if a package upgrade demands it. Most package upgrades do not change system configuration, but rather install an additional configuration file for you to examine later.

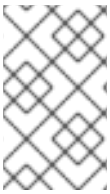
Note that the installation medium that you are using might not contain all the software packages that you need to upgrade your computer.



NOTE

Software you have installed manually on your existing Red Hat Enterprise Linux system may behave differently after an upgrade. You may need to manually reinstall or recompile this software after an upgrade to ensure it performs correctly on the updated system.

23.12.1. Upgrading Using the Installer



NOTE

In general, Red Hat recommends that you keep user data on a separate /**home** partition and perform a fresh installation. For more information on partitions and how to set them up, refer to Section 9.13, “Disk Partitioning Setup” .

If you choose to upgrade your system using the installation program, any software not provided by Red Hat Enterprise Linux that conflicts with Red Hat Enterprise Linux software is overwritten. Before you begin an upgrade this way, make a list of your system's current packages for later reference:

```
rpm -qa --qf '%{NAME} %{VERSION}-%{RELEASE} %{ARCH}\n' > ~/old-pkglist.txt
```

After installation, consult this list to discover which packages you may need to rebuild or retrieve from sources other than Red Hat.

Next, make a backup of any system configuration data:

```
su -c 'tar czf /tmp/etc-`date +%F`.tar.gz /etc'
su -c 'mv /tmp/etc-*.tar.gz /home'
```

Make a complete backup of any important data before performing an upgrade. Important data may include the contents of your entire /**home** directory as well as content from services such as an Apache, FTP, or SQL server, or a source code management system. Although upgrades are not destructive, if you perform one improperly there is a small possibility of data loss.

WARNING

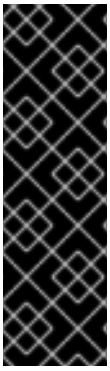
Note that the above examples store backup materials in a /**home** directory. If your /**home** directory is not a separate partition, *you should not follow these examples verbatim!* Store your backups on another device such as CD or DVD discs or an external hard disk.

For more information on completing the upgrade process later, refer to Section 35.2, “Finishing an Upgrade”.

23.13. DISK PARTITIONING SETUP

WARNING

It is always a good idea to back up any data that you have on your systems. For example, if you are upgrading or creating a dual-boot system, you should back up any data you wish to keep on your storage devices. Mistakes do happen and can result in the loss of all your data.



IMPORTANT

If you install Red Hat Enterprise Linux in text mode, you can only use the default partitioning schemes described in this section. You cannot add or remove partitions or file systems beyond those that the installer automatically adds or removes. If you require a customized layout at installation time, you should perform a graphical installation over a VNC connection or a kickstart installation.

Furthermore, advanced options such as LVM, encrypted filesystems, and resizable filesystems are available only in graphical mode and kickstart.

Partitioning allows you to divide your storage devices into isolated sections, where each section behaves as a separate Linux device. Partitioning is particularly useful if you run multiple operating systems, or wish to enforce a logical or functional distinction between your storage partitions (such as a **/home** partition that persistently contains user information).

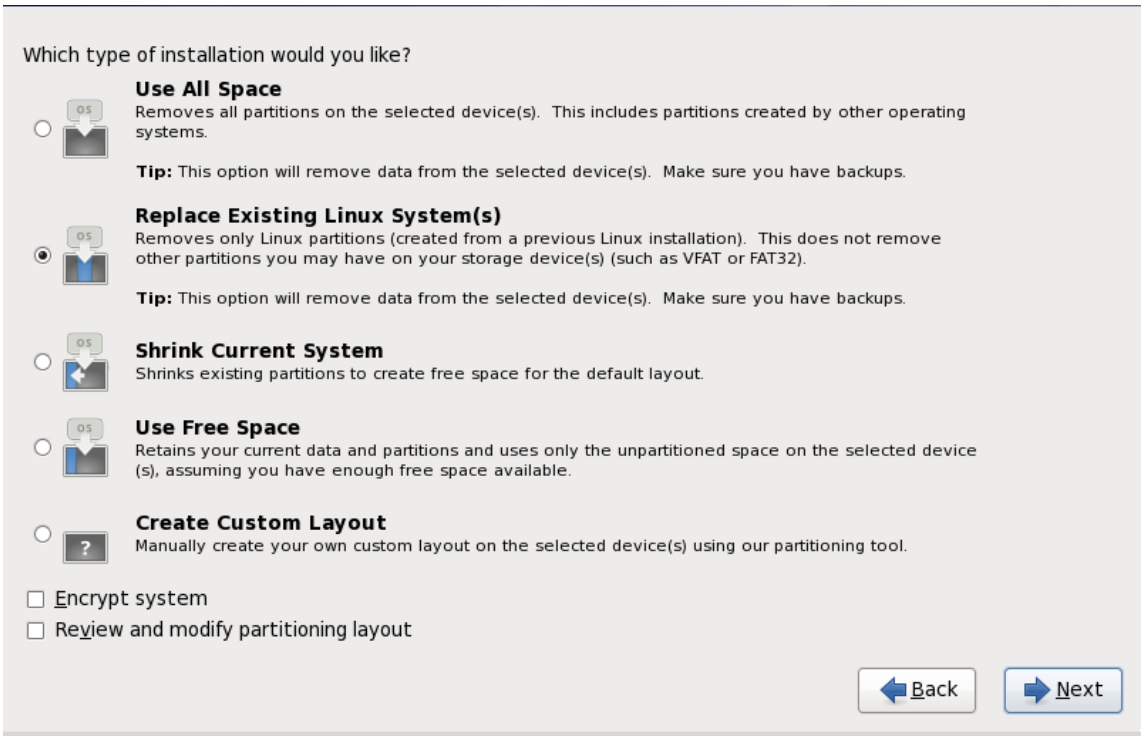


Figure 23.36. Disk Partitioning Setup

On this screen you can choose to create the default partition layout in one of four different ways, or choose to partition storage devices manually to create a custom layout.

The first four options allow you to perform an automated installation without having to partition your storage devices yourself. If you do not feel comfortable with partitioning your system, choose one of these options and let the installation program partition the storage devices for you. Depending on the option that you choose, you can still control what data (if any) is removed from the system.

IMPORTANT

To encrypt partitions, you will need to select the **Create Custom Layout**option. Partitions created with one of the four automated options cannot be encrypted.

Your options are:

Use All Space

Select this option to remove all partitions on your storage drives (this includes partitions created by other operating systems such as z/VM or z/OS).

WARNING

If you select this option, all data on the selected DASD and SCSI storage devices is removed by the installation program.

Replace Existing Linux System(s)

Select this option to remove only Linux partitions (partitions created from a previous Linux installation). This does not remove other partitions you may have on your storage devices (such as z/VM or z/OS partitions).

Shrink Current System

Select this option to resize your current data and partitions manually and install a default Red Hat Enterprise Linux layout in the space that is freed.

WARNING

If you shrink partitions on which other operating systems are installed, you might not be able to use those operating systems. Although this partitioning option does not destroy data, operating systems typically require some free space in their partitions. Before you resize a partition that holds an operating system that you might want to use again, find out how much space you need to leave free.

Use Free Space

Select this option to retain your current data and partitions and install Red Hat Enterprise Linux in the unused space available on the storage drives. Ensure that there is sufficient space available on the storage drives before you select this option – refer to Section 18.1, "Pre-Installation".

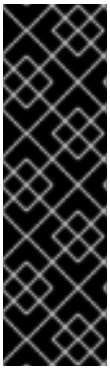
Create Custom Layout

Select this option to partition storage devices manually and create customized layouts. Refer to Section 23.15, " Creating a Custom Layout or Modifying the Default Layout "

Choose your preferred partitioning method by clicking the radio button to the left of its description in the dialog box.

Select **Encrypt system** to encrypt all partitions except the **/boot** partition. Refer to Appendix C, *Disk Encryption* for information on encryption.

To review and make any necessary changes to the partitions created by automatic partitioning, select the **Review** option. After selecting **Review** and clicking **Next** to move forward, the partitions created for you by **anaconda** appear. You can make modifications to these partitions if they do not meet your needs.



IMPORTANT

When you install Red Hat Enterprise Linux 6 on a system with multipath and non-multipath storage devices, the automatic partitioning layout in the installer might create volume groups that contain a mix of multipath and non-multipath devices. This defeats the purpose of multipath storage.

We advise that you select only multipath or only non-multipath devices on the disk selection screen that appears after selecting automatic partitioning. Alternatively, select custom partitioning.

Click **Next** once you have made your selections to proceed.

23.14. CHOOSING A DISK ENCRYPTION PASSPHRASE

If you selected the **Encrypt System** option, the installer prompts you for a passphrase with which to encrypt the partitions on the system.

Partitions are encrypted using the *Linux Unified Key Setup* — refer to Appendix C, *Disk Encryption* for more information.

Figure 23.37. Enter passphrase for encrypted partition

Choose a passphrase and type it into each of the two fields in the dialog box. You must provide this passphrase every time that the system boots.

WARNING

If you lose this passphrase, any encrypted partitions and the data on them will become completely inaccessible. There is no way to recover a lost passphrase.

Note that if you perform a kickstart installation of Red Hat Enterprise Linux, you can save encryption passphrases and create backup encryption passphrases during installation. Refer to Section C.3.2, "Saving Passphrases" and Section C.3.3, "Creating and Saving Backup Passphrases".

23.15. CREATING A CUSTOM LAYOUT OR MODIFYING THE DEFAULT LAYOUT

If you chose one of the four automatic partitioning options and did not select **Review**, skip ahead to Section 23.16, "Write Changes to Disk" .

If you chose to create a custom layout, you must tell the installation program where to install Red Hat Enterprise Linux. This is done by defining mount points for one or more disk partitions in which Red Hat Enterprise Linux is installed.

If you have not yet planned how to set up your partitions, refer to Appendix A, *An Introduction to Disk Partitions* and Section 23.15.5, "Recommended Partitioning Scheme". At a bare minimum, you need an appropriately-sized root partition, and usually a swap partition appropriate to the amount of RAM you have on the system.

Anaconda can handle the partitioning requirements for a typical installation.

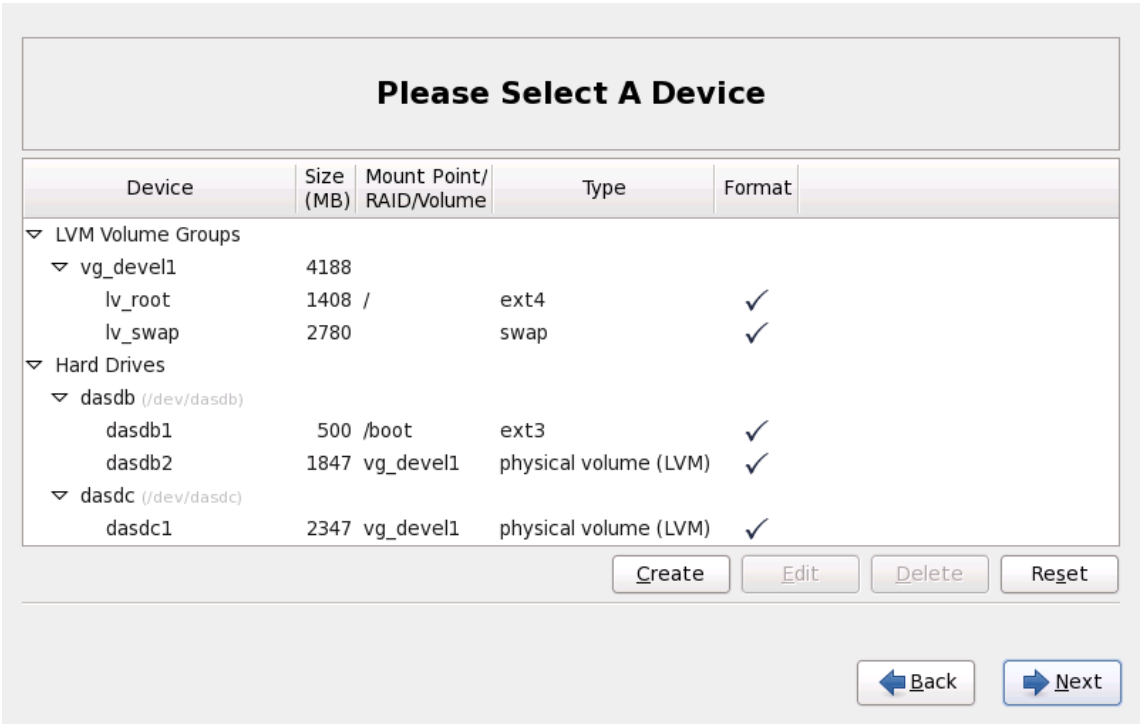


Figure 23.38. Partitioning on System z

The partitioning screen contains two panes. The top pane contains a graphical representation of the DASD, FCP LUN, or logical volume selected in the lower pane.

Above the display, you can review the **Drive** name (such as /dev/dasda), the **Geom** (which shows the hard disk's geometry and consists of three numbers representing the number of cylinders, heads, and sectors as reported by the hard disk), and the **Model** of the hard drive as detected by the installation program.

Using your mouse, click once to highlight a particular field in the graphical display. Double-click to edit an existing partition or to create a partition out of existing free space.

The lower pane contains a list of all DASDs, FCP LUNs, and logical volumes to be used during installation, as specified earlier in the installation process – refer to Section 23.10, " Assign Storage Devices ". Note that if you specified a CMSDASD in your parameter file, DASD names begin at **dasdb**; **dasda** was assigned to the CMSDASD and this name is no longer available at this point in the installation process.

Devices are grouped by type. Click on the small triangles to the left of each device type to view or hide devices of that type.

Anaconda displays several details for each device listed:

Device

the name of the device, logical volume, or partition

Size (MB)

the size of the device, logical volume, or partition (in MB)

Mount Point/RAID/Volume

the *mount point* (location within a file system) on which a partition is to be mounted, or the name of the RAID or logical volume group of which it is a part

Type

the type of partition. If the partition is a standard partition, this field displays the type of file system on the partition (for example, ext4). Otherwise, it indicates that the partition is a **physical volume (LVM)**, or part of a **software RAID**

Format

A check mark in this column indicates that the partition will be formatted during installation.

Beneath the lower pane are four buttons: **Create**, **Edit**, **Delete**, and **Reset**.

Select a device or partition by clicking on it in either the graphical representation in the upper pane of in the list in the lower pane, then click one of the four buttons to carry out the following actions:

Create

create a new partition, logical volume, or software RAID

Edit

change an existing partition, logical volume, or software RAID. Note that you can only shrink partitions with the **Resize** button, not enlarge partitions.

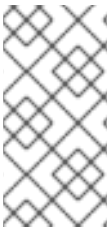
Delete

remove a partition, logical volume, or software RAID

Reset

undo all changes made in this screen

Finally, note which device is associated with /**boot**. The kernel files and bootloader sector will be associated with this device. The first DASD or SCSI LUN will be used, and the device number will be used when re-IPLing the post-installed system.



NOTE

The screenshots in the following subsections of this manual sometimes show hard disk types and device names that do not appear as such on System z. These screenshots are only intended to illustrate the installation interface itself and apply equally to DASDs and FCP-attached SCSI disks.

23.15.1. Create Storage

The **Create Storage** dialog allows you to create new storage partitions, logical volumes, and software RAIDs. **Anaconda** presents options as available or unavailable depending on the storage already present on the system or configured to transfer to the system.

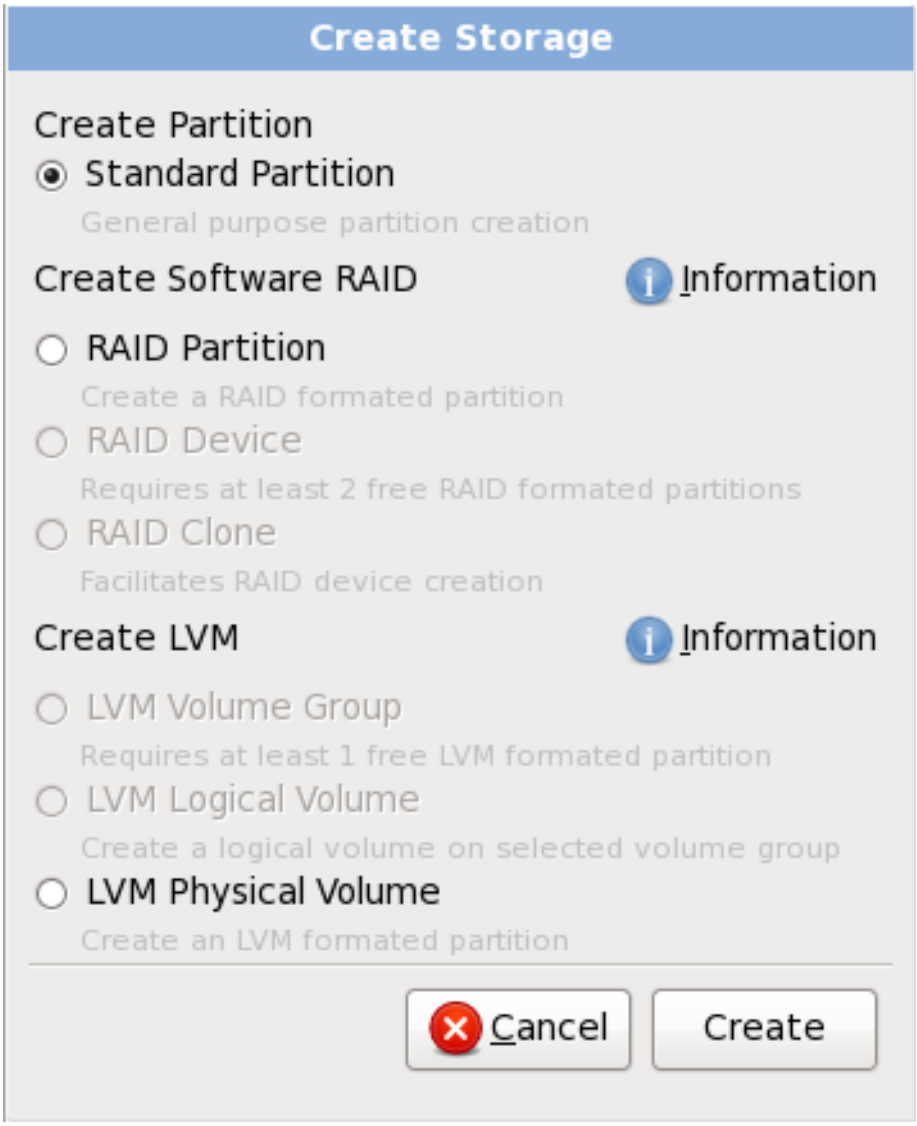


Figure 23.39. Creating Storage

Options are grouped under **Create Partition**, **Create Software RAID** and **Create LVM** as follows:

Create Partition

Refer to Section 23.15.2, "Adding Partitions" for details of the **Add Partition** dialog.

- **Standard Partition** – create a standard disk partition (as described in Appendix A, *An Introduction to Disk Partitions*) in unallocated space.

Create Software RAID

On System z, the storage subsystem uses RAID transparently, and you do not need to set it up.

Refer to Section 23.15.3, " Create Software RAID " for more detail.

- **RAID Partition** – create a partition in unallocated space to form part of a software RAID device. To form a software RAID device, two or more RAID partitions must be available on the system.
- **RAID Device** – combine two or more RAID partitions into a software RAID device. When you choose this option, you can specify the type of RAID device to create (the *RAID level*). This option is only available when two or more RAID partitions are available on the system.

Create LVM Logical Volume

Refer to Section 23.15.4, " Create LVM Logical Volume " for more detail.

- **LVM Physical Volume** — create a *physical volume* in unallocated space.
- **LVM Volume Group** — create a *volume group* from one or more physical volumes. This option is only available when at least one physical volume is available on the system.
- **LVM Logical Volume** — create a *logical volume* on a volume group. This option is only available when at least one volume group is available on the system.

23.15.2. Adding Partitions

To add a new partition, select the **Create** button. A dialog box appears (refer to Figure 23.40, "Creating a New Partition").

NOTE

You must dedicate at least one partition for this installation, and optionally more. For more information, refer to Appendix A, *An Introduction to Disk Partitions*.

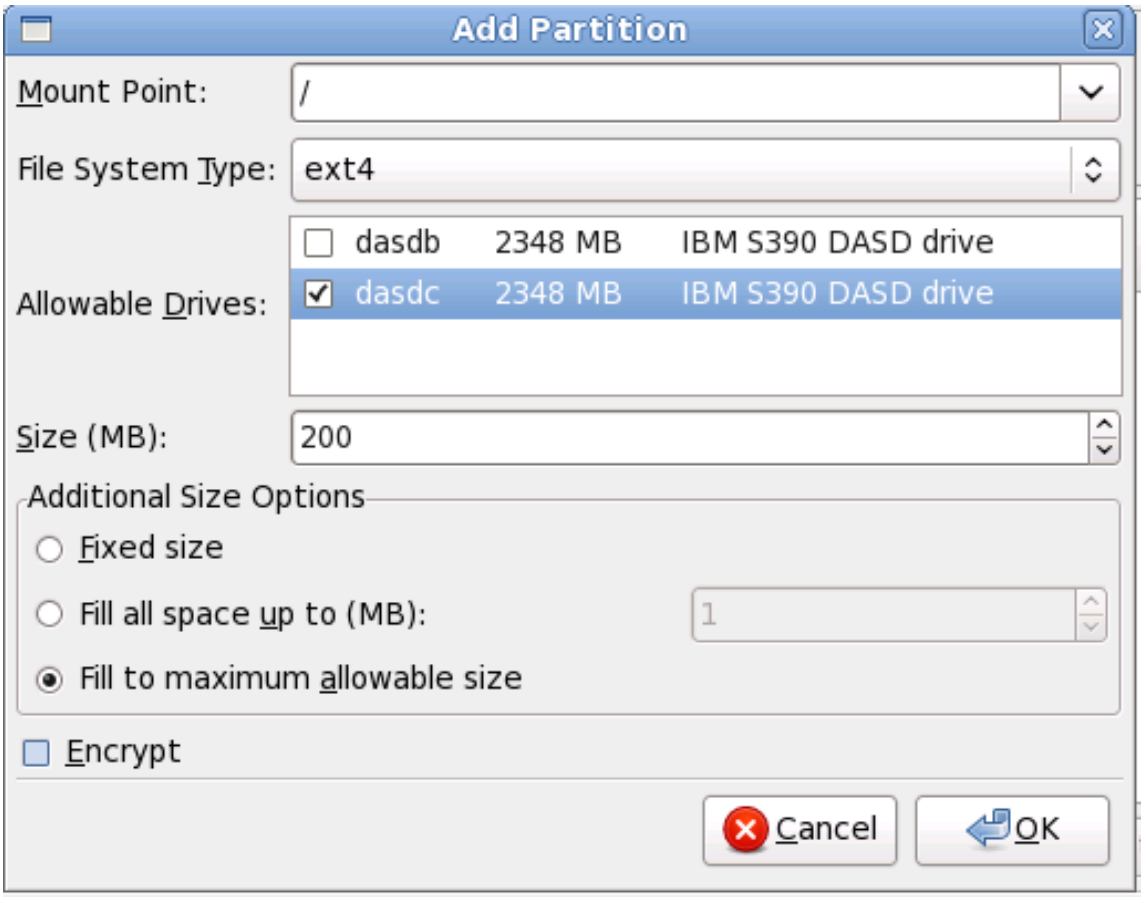


Figure 23.40. Creating a New Partition

- **Mount Point**: Enter the partition's mount point. For example, if this partition should be the root partition, enter /; enter **/boot** for the **/boot** partition, and so on. You can also use the pull-down menu to choose the correct mount point for your partition. For a swap partition the mount point should not be set — setting the filesystem type to **swap** is sufficient.

- **File System Type**: Using the pull-down menu, select the appropriate file system type for this partition. For more information on file system types, refer to Section 23.15.2.1, "File System Types".

- **Allowable Drives**: This field contains a list of the hard disks installed on your system. If a hard disk's box is highlighted, then a desired partition can be created on that hard disk. If the box is *not* checked, then the partition will *never* be created on that hard disk. By using different checkbox settings, you can have **anaconda** place partitions where you need them, or let **anaconda** decide where partitions should go.

- **Size (MB)**: Enter the size (in megabytes) of the partition. Note, this field starts with 200 MB; unless changed, only a 200 MB partition will be created.

- **Additional Size Options**: Choose whether to keep this partition at a fixed size, to allow it to "grow" (fill up the available hard drive space) to a certain point, or to allow it to grow to fill any remaining hard drive space available.

  If you choose **Fill all space up to (MB)**, you must give size constraints in the field to the right of this option. This allows you to keep a certain amount of space free on your hard drive for future use.

- **Force to be a primary partition**: Select whether the partition you are creating should be one of the first four partitions on the hard drive. If unselected, the partition is created as a logical partition. Refer to Section A.1.3, "Partitions Within Partitions — An Overview of Extended Partitions", for more information.

- **Encrypt**: Choose whether to encrypt the partition so that the data stored on it cannot be accessed without a passphrase, even if the storage device is connected to another system. Refer to Appendix C, *Disk Encryption* for information on encryption of storage devices. If you select this option, the installer prompts you to provide a passphrase before it writes the partition to the disk.

- **OK**: Select **OK** once you are satisfied with the settings and wish to create the partition.

- **Cancel**: Select **Cancel** if you do not want to create the partition.

23.15.2.1. File System Types

Red Hat Enterprise Linux allows you to create different partition types and file systems. The following is a brief description of the different partition types and file systems available, and how they can be used.

Partition types

- **standard partition** — A standard partition can contain a file system or swap space, or it can provide a container for software RAID or an LVM physical volume.

- **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. Refer to the Red Hat Enterprise Linux Deployment Guide for additional information.

- **software RAID** — Creating two or more software RAID partitions allows you to create a RAID device. For more information regarding RAID, refer to the chapter *RAID (Redundant Array of Independent Disks)* in the Red Hat Enterprise Linux Deployment Guide .

- **physical volume (LVM)** — Creating one or more physical volume (LVM) partitions allows you to create an LVM logical volume. LVM can improve performance when using physical disks. For more information regarding LVM, refer to the Red Hat Enterprise Linux Deployment Guide .

File systems

- **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. A maximum file system size of 16TB is supported for ext4. The ext4 file system is selected by default and is highly recommended.

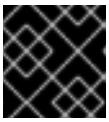
NOTE

The mount options **user_xattr** and **acl** are automatically set on ext4 systems by the installation system. These options enable extended attributes and access control lists, respectively. More information about mount options can be found in the Red Hat Enterprise Linux Storage Administration Guide .

- **ext3** — The ext3 file system is based on the ext2 file system and has one main advantage — journaling. Using a journaling file system reduces time spent recovering a file system after a crash as there is no need to **fsck** [12] the file system. A maximum file system size of 16TB is supported for ext3.

- **ext2** — An ext2 file system supports standard Unix file types (regular files, directories, symbolic links, etc). It provides the ability to assign long file names, up to 255 characters.

- **xfs** — XFS is a highly scalable, high-performance file system that supports filesystems 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 supports metadata journaling, which facilitates quicker crash recovery. The XFS file system can also be defragmented and resized while mounted and active.



IMPORTANT

Red Hat Enterprise Linux 6.9 does not support XFS on System z.

- **vfat** — The VFAT file system is a Linux file system that is compatible with Microsoft Windows long filenames on the FAT file system.

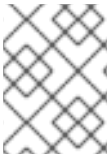
- **Btrfs** — Btrfs is under development as a file system capable of addressing and managing more files, larger files, and larger volumes than the ext2, ext3, and ext4 file systems. Btrfs is designed to make the file system tolerant of errors, and to facilitate the detection and repair of errors when they occur. It uses checksums to ensure the validity of data and metadata, and maintains snapshots of the file system that can be used for backup or repair.

  Because Btrfs is still experimental and under development, the installation program does not offer it by default. If you want to create a Btrfs partition on a drive, you must commence the installation process with the boot option **btrfs**. Refer to Chapter 28, *Boot Options* for instructions.

WARNING

Red Hat Enterprise Linux 6.9 includes Btrfs as a technology preview to allow you to experiment with this file system. You should not choose Btrfs for partitions that will contain valuable data or that are essential for the operation of important systems.

23.15.3. Create Software RAID



NOTE

On System z, the storage subsystem uses RAID transparently. There is no need to set up a software RAID.

Redundant arrays of independent disks (RAIDs) are constructed from multiple storage devices that are arranged to provide increased performance and – in some configurations – greater fault tolerance. Refer to the Red Hat Enterprise Linux Storage Administration Guide for a description of different kinds of RAIDs.

To make a RAID device, you must first create software RAID partitions. Once you have created two or more software RAID partitions, select **RAID** to join the software RAID partitions into a RAID device.

RAID Partition

Choose this option to configure a partition for software RAID. This option is the only choice available if your disk contains no software RAID partitions. This is the same dialog that appears when you add a standard partition – refer to Section 23.15.2, "Adding Partitions" for a description of the available options. Note, however, that **File System Type** must be set to **software RAID**

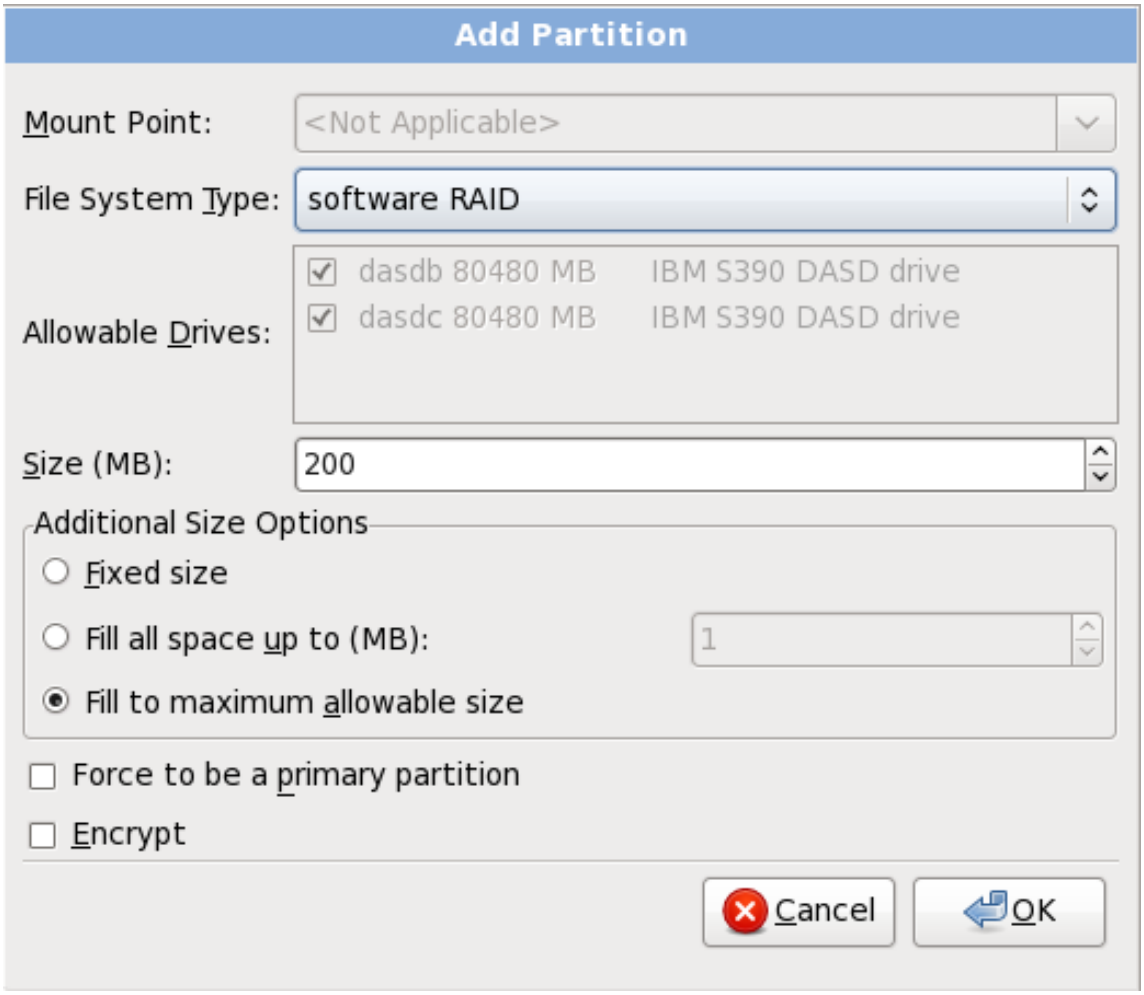


Figure 23.41. Create a software RAID partition

RAID Device

Choose this option to construct a RAID device from two or more existing software RAID partitions. This option is available if two or more software RAID partitions have been configured.

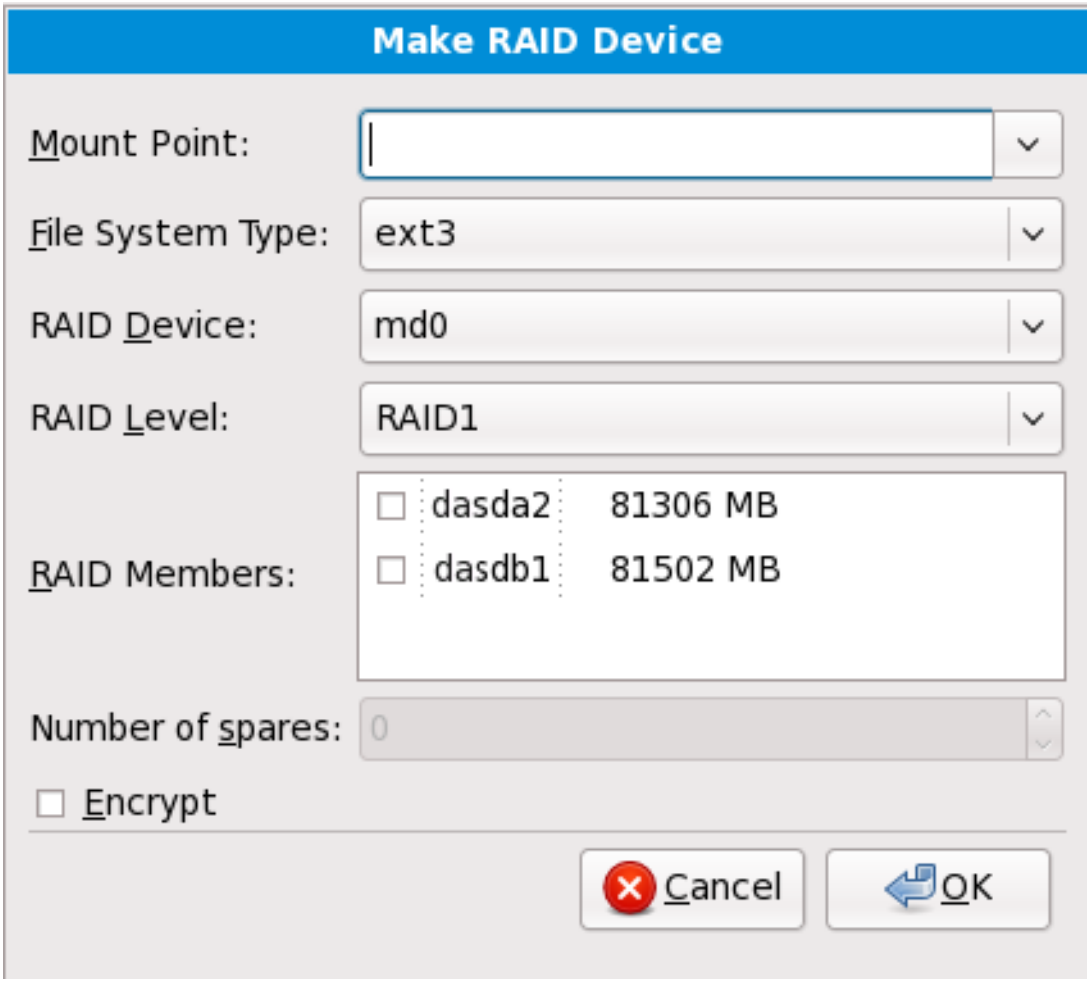


Figure 23.42. Create a RAID device

Select the file system type as for a standard partition.

Anaconda automatically suggests a name for the RAID device, but you can manually select names from **md0** to **md15**.

Click the checkboxes beside individual storage devices to include or remove them from this RAID.

The **RAID Level** corresponds to a particular type of RAID. Choose from the following options:

- **RAID 0** – distributes data across multiple storage devices. Level 0 RAIDs offer increased performance over standard partitions, and can be used to pool the storage of multiple devices into one large virtual device. Note that Level 0 RAIDS offer no redundancy and that the failure of one device in the array destroys the entire array. RAID 0 requires at least two RAID partitions.

- **RAID 1** – mirrors the data on one storage device onto one or more other storage devices. Additional devices in the array provide increasing levels of redundancy. RAID 1 requires at least two RAID partitions.

- **RAID 4** – distributes data across multiple storage devices, but uses one device in the array to store parity information that safeguards the array in case any device within the array fails. Because all parity information is stored on the one device, access to this device creates a bottleneck in the performance of the array. RAID 4 requires at least three RAID partitions.

- **RAID 5** – distributes data and parity information across multiple storage devices. Level 5 RAIDs therefore offer the performance advantages of distributing data across multiple

devices, but do not share the performance bottleneck of level 4 RAIDs because the parity information is also distributed through the array. RAID 5 requires at least three RAID partitions.

- **RAID 6** – level 6 RAIDs are similar to level 5 RAIDs, but instead of storing only one set of parity data, they store two sets. RAID 6 requires at least four RAID partitions.
- **RAID 10** – level 10 RAIDs are *nested RAIDs* or *hybrid RAIDs*. Level 10 RAIDs are constructed by distributing data over mirrored sets of storage devices. For example, a level 10 RAID constructed from four RAID partitions consists of two pairs of partitions in which one partition mirrors the other. Data is then distributed across both pairs of storage devices, as in a level 0 RAID. RAID 10 requires at least four RAID partitions.

23.15.4. Create LVM Logical Volume

IMPORTANT

LVM initial set up is not available during text-mode installation. If you need to create an LVM configuration from scratch, establish another SSH connection to the installation image with the root user and run the **lvm** command.

Logical Volume Management (LVM) presents a simple logical view of underlying physical storage space, such as a hard drives or LUNs. Partitions on physical storage are represented as *physical volumes* that can be grouped together into *volume groups*. Each volume group can be divided into multiple *logical volumes*, each of which is analogous to a standard disk partition. Therefore, LVM logical volumes function as partitions that can span multiple physical disks.

To read more about LVM, refer to the Red Hat Enterprise Linux Deployment Guide . Note, LVM is only available in the graphical installation program.

LVM Physical Volume

Choose this option to configure a partition or device as an LVM physical volume. This option is the only choice available if your storage does not already contain LVM Volume Groups. This is the same dialog that appears when you add a standard partition – refer to Section 23.15.2, "Adding Partitions" for a description of the available options. Note, however, that **File System Type** must be set to **physical volume (LVM)**

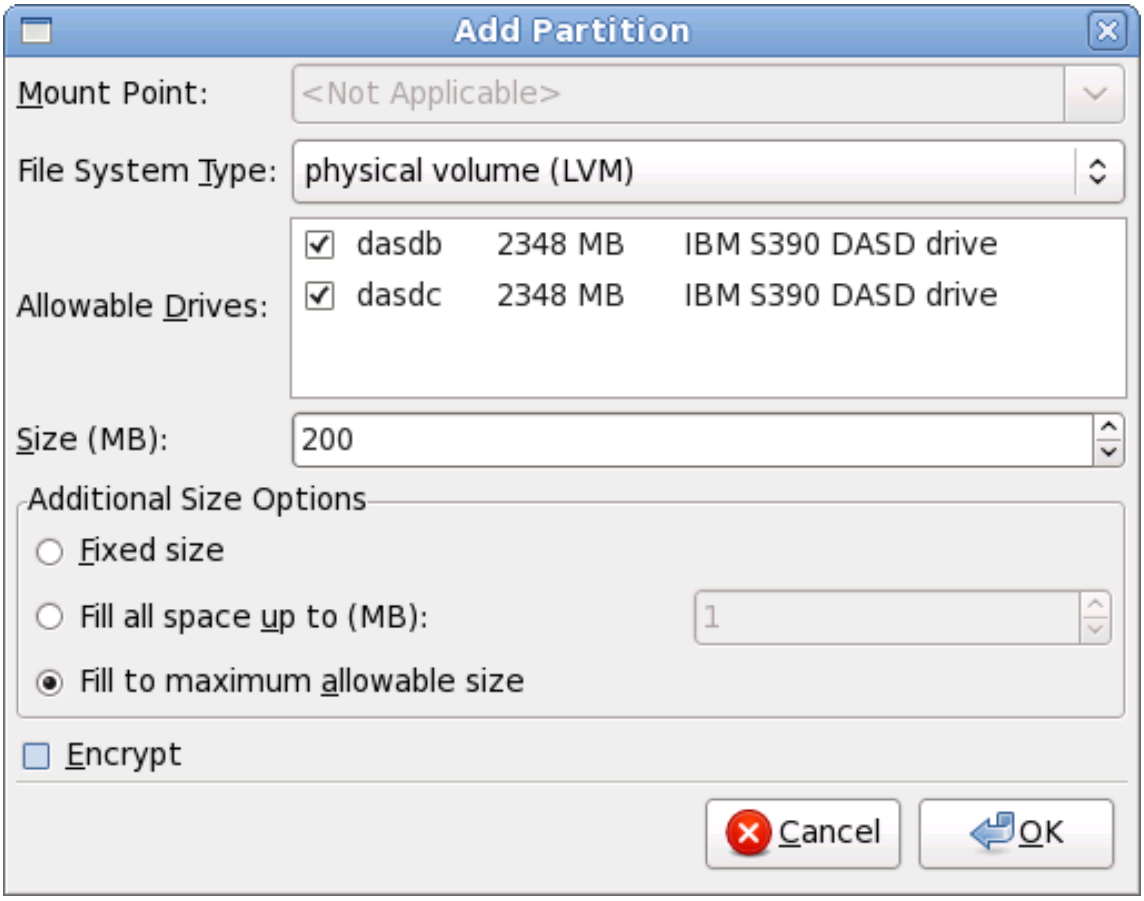


Figure 23.43. Create an LVM Physical Volume

Make LVM Volume Group

Choose this option to create LVM volume groups from the available LVM physical volumes, or to add existing logical volumes to a volume group.

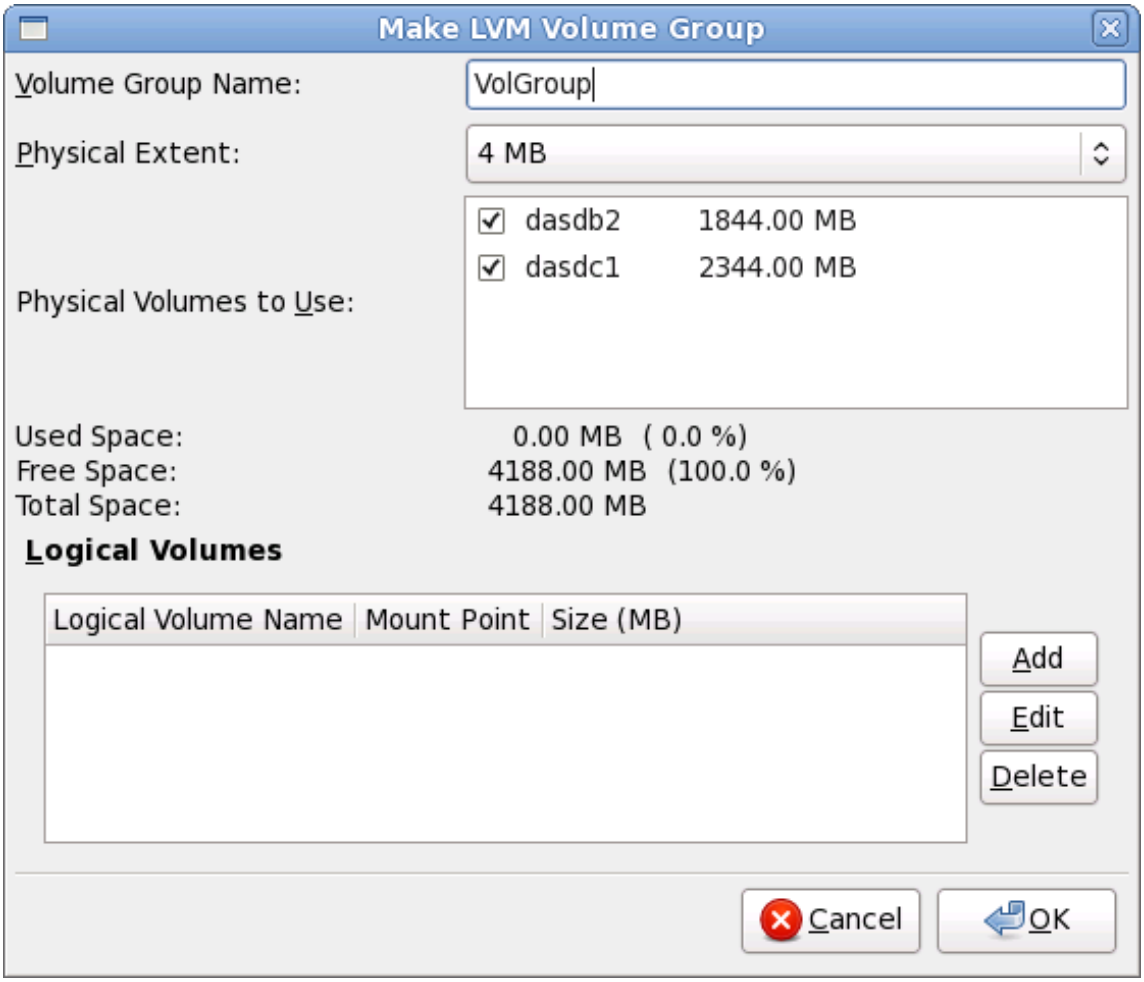


Figure 23.44. Make LVM Volume Group

To assign one or more physical volumes to a volume group, first name the volume group. Then select the physical volumes to be used in the volume group. Finally, configure logical volumes on any volume groups using the **Add**, **Edit** and **Delete** options.

You may not remove a physical volume from a volume group if doing so would leave insufficient space for that group's logical volumes. Take for example a volume group made up of two 5 GB LVM physical volume partitions, which contains an 8 GB logical volume. The installer would not allow you to remove either of the component physical volumes, since that would leave only 5 GB in the group for an 8 GB logical volume. If you reduce the total size of any logical volumes appropriately, you may then remove a physical volume from the volume group. In the example, reducing the size of the logical volume to 4 GB would allow you to remove one of the 5 GB physical volumes.

Make Logical Volume

Choose this option to create an LVM logical volume. Select a mount point, file system type, and size (in MB) just as if it were a standard disk partition. You can also choose a name for the logical volume and specify the volume group to which it will belong.

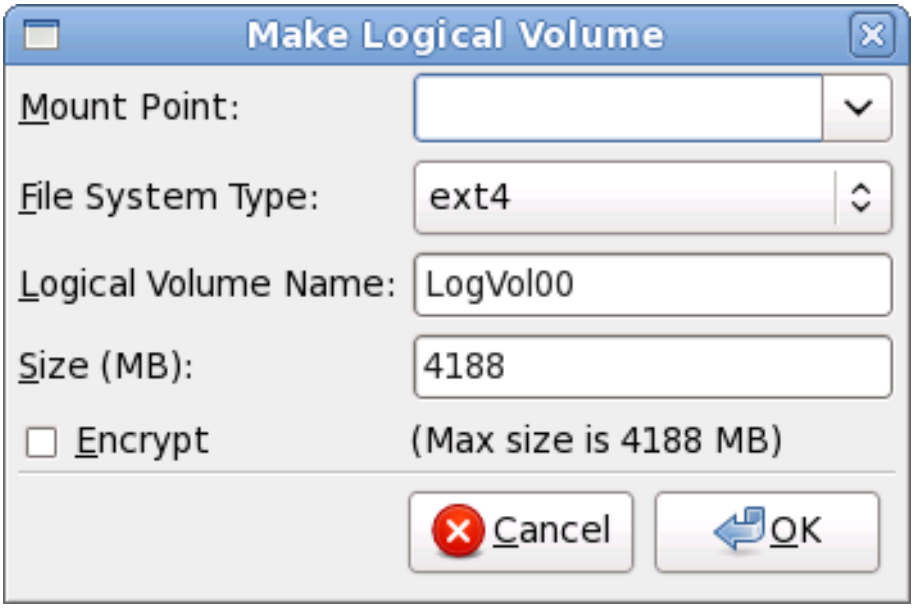


Figure 23.45. Make Logical Volume

23.15.5. Recommended Partitioning Scheme

Configuring efficient swap space for Linux on System z is a complex task. It very much depends on the specific environment and should be tuned to the actual system load.

Refer to the following resources for more information and to guide your decision:

- 'Chapter 7. Linux Swapping' in the IBM Redbooks publication *Linux on IBM System z: Performance Measurement and Tuning* [IBM Form Number SG24-6926-01], [ISBN 0738485586], available from http://www.redbooks.ibm.com/abstracts/sg246926.html
- *Linux Performance when running under VM*, available from http://www.vm.ibm.com/perf/tips/linuxper.html

23.16. WRITE CHANGES TO DISK

The installer prompts you to confirm the partitioning options that you selected. Click **Write changes to disk** to allow the installer to partition your hard drive and install Red Hat Enterprise Linux.

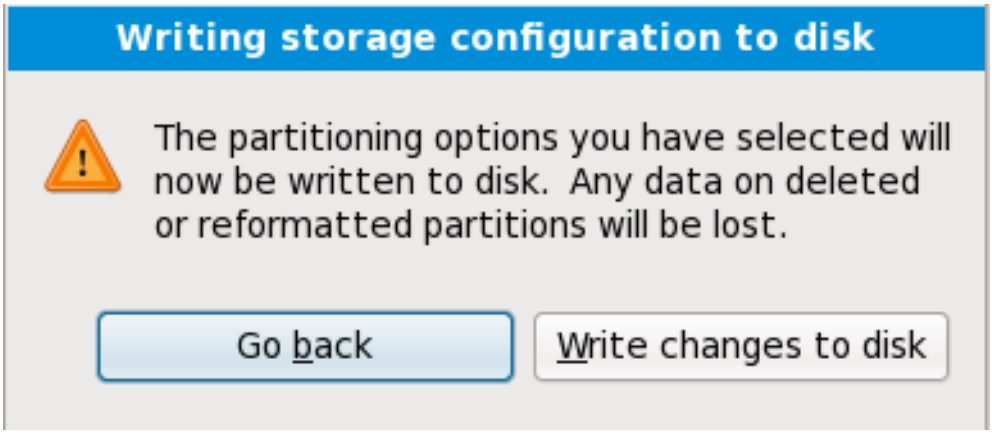


Figure 23.46. Writing storage configuration to disk

If you are certain that you want to proceed, click **Write changes to disk**.

WARNING

Up to this point in the installation process, the installer has made no lasting changes to your computer. When you click **Write changes to disk**, the installer will allocate space on your hard drive and start to transfer Red Hat Enterprise Linux into this space. Depending on the partitioning option that you chose, this process might include erasing data that already exists on your computer.

To revise any of the choices that you made up to this point, click **Go back**. To cancel installation completely, switch off your computer.

After you click **Write changes to disk**, allow the installation process to complete. If the process is interrupted (for example, by you switching off or resetting the computer, or by a power outage) you will probably not be able to use your computer until you restart and complete the Red Hat Enterprise Linux installation process, or install a different operating system.

23.17. PACKAGE GROUP SELECTION

Now that you have made most of the choices for your installation, you are ready to confirm the default package selection or customize packages for your system.

The **Package Installation Defaults** screen appears and details the default package set for your Red Hat Enterprise Linux installation. This screen varies depending on the version of Red Hat Enterprise Linux you are installing.

IMPORTANT

If you install Red Hat Enterprise Linux in text mode, you cannot make package selections. The installer automatically selects packages only from the base and core groups. These packages are sufficient to ensure that the system is operational at the end of the installation process, ready to install updates and new packages. To change the package selection, complete the installation, then use the **Add/Remove Software** application to make desired changes.

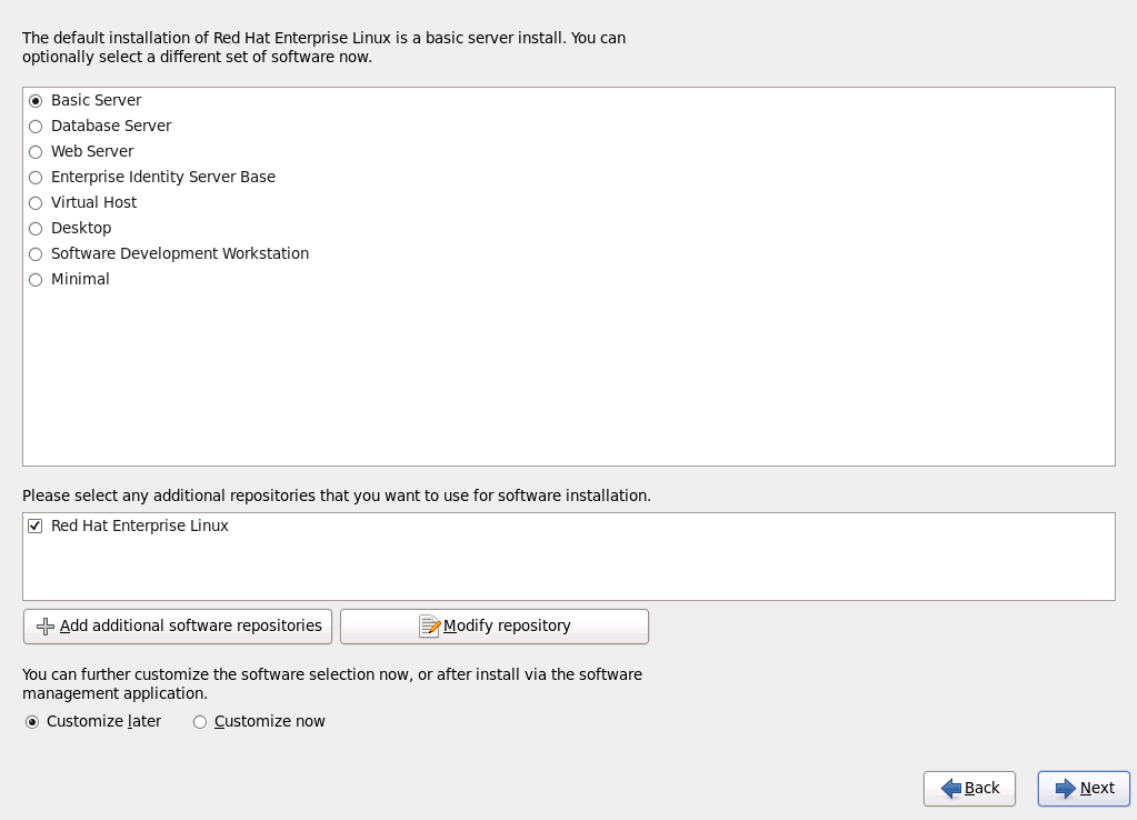


Figure 23.47. Package Group Selection

By default, the Red Hat Enterprise Linux installation process loads a selection of software that is suitable for a system deployed as a basic server. Note that this installation does not include a graphical environment. To include a selection of software suitable for other roles, click the radio button that corresponds to one of the following options:

Basic Server

This option provides a basic installation of Red Hat Enterprise Linux for use on a server.

Database Server

This option provides the **MySQL** and **PostgreSQL** databases.

Web server

This option provides the **Apache** web server.

Enterprise Identity Server Base

This option provides **OpenLDAP** and **Enterprise Identity Management** (IPA) to create an identity and authentication server.

Virtual Host

This option provides the **KVM** and **Virtual Machine Manager** tools to create a host for virtual machines.

Desktop

This option provides the **OpenOffice.org** productivity suite, graphical tools such as the **GIMP**, and multimedia applications.

Software Development Workstation

This option provides the necessary tools to compile software on your Red Hat Enterprise Linux system.

Minimal

This option provides only the packages essential to run Red Hat Enterprise Linux. A minimal installation provides the basis for a single-purpose server or desktop appliance and maximizes performance and security on such an installation.

WARNING

Minimal installation currently does not configure the firewall (**iptables**/**ip6tables**) by default because the authconfig and system-config-firewall-base packages are missing from the selection. To work around this issue, you can use a Kickstart file to add these packages to your selection. See the Red Hat Customer Portal for details about the workaround, and Chapter 32, *Kickstart Installations* for information about Kickstart files.

If you do not use the workaround, the installation will complete successfully, but no firewall will be configured, presenting a security risk.

If you choose to accept the current package list, skip ahead to Section 23.18, "Installing Packages".

To select a component, click on the checkbox beside it (refer to Figure 23.47, "Package Group Selection").

To customize your package set further, select the **Customize now** option on the screen. Clicking **Next** takes you to the **Package Group Selection** screen.

23.17.1. Installing from Additional Repositories

You can define additional *repositories* to increase the software available to your system during installation. A repository is a network location that stores software packages along with *metadata* that describes them. Many of the software packages used in Red Hat Enterprise Linux require other software to be installed. The installer uses the metadata to ensure that these requirements are met for every piece of software you select for installation.

The **Red Hat Enterprise Linux** repository is automatically selected for you. It contains the complete collection of software that was released as Red Hat Enterprise Linux 6.9, with the various pieces of software in their versions that were current at the time of release.

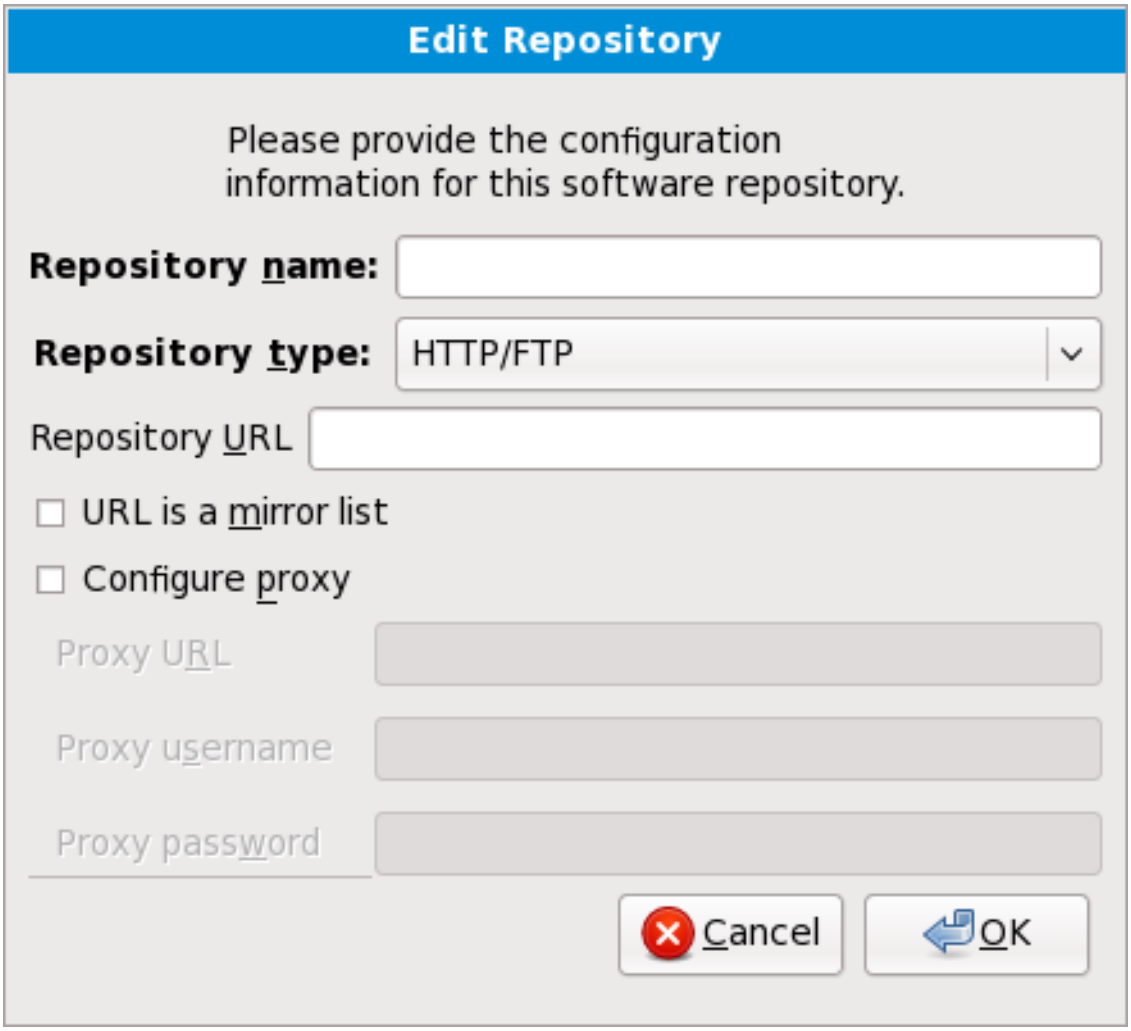


Figure 23.48. Adding a software repository

To include software from extra *repositories*, select **Add additional software repositories** and provide the location of the repository.

To edit an existing software repository location, select the repository in the list and then select **Modify repository**.

If you change the repository information during a non-network installation, such as from a Red Hat Enterprise Linux DVD, the installer prompts you for network configuration information.

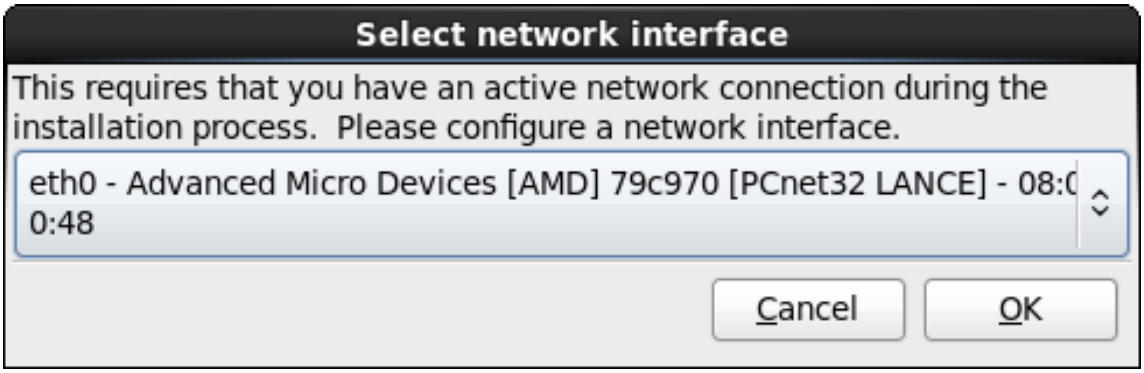


Figure 23.49. Select network interface

1. Select an interface from the drop-down menu.
2. Click **OK**.

Anaconda then starts **NetworkManager** to allow you to configure the interface.

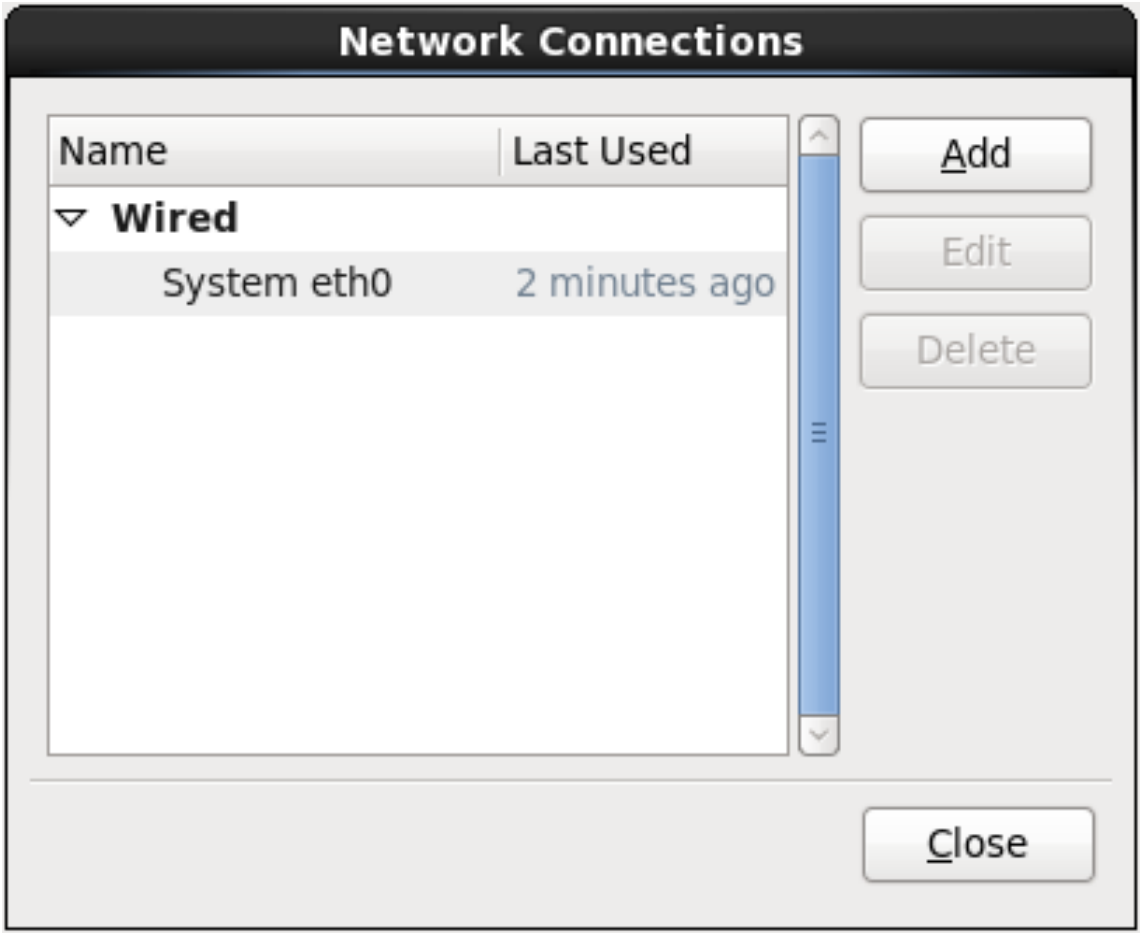


Figure 23.50. Network Connections

For details of how to use **NetworkManager**, refer to Section 23.7, "Setting the Hostname"

If you select **Add additional software repositories**, the **Edit repository** dialog appears. Provide a **Repository name** and the **Repository URL** for its location.

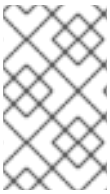
Once you have located a mirror, to determine the URL to use, find the directory on the mirror that *contains* a directory named **repodata**.

Once you provide information for an additional repository, the installer reads the package metadata over the network. Software that is specially marked is then included in the package group selection system.

WARNING

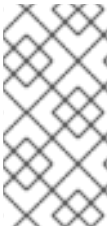
If you choose **Back** from the package selection screen, any extra repository data you may have entered is lost. This allows you to effectively cancel extra repositories. Currently there is no way to cancel only a single repository once entered.

23.17.2. Customizing the Software Selection



NOTE

Your Red Hat Enterprise Linux system automatically supports the language that you selected at the start of the installation process. To include support for additional languages, select the package group for those languages from the **Languages** category.



NOTE

Users of IBM System z who want support for developing or running legacy 31-bit applications are encouraged to select the **Compatibility Arch Support** and **Compatibility Arch Development Support** packages to install architecture specific support for their systems.

Select **Customize now** to specify the software packages for your final system in more detail. This option causes the installation process to display an additional customization screen when you select **Next**.

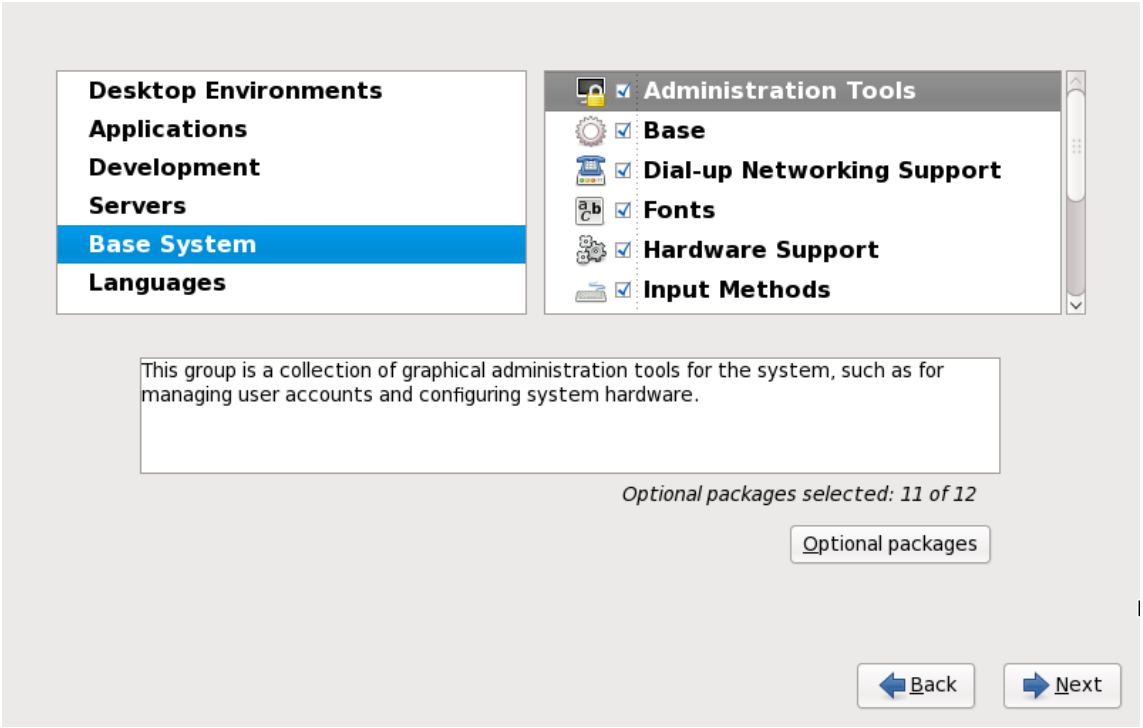


Figure 23.51. Package Group Details

Red Hat Enterprise Linux divides the included software into *package groups*. For ease of use, the package selection screen displays these groups as categories.

You can select package groups, which group components together according to function (for example, **X Window System** and **Editors**), individual packages, or a combination of the two.

To view the package groups for a category, select the category from the list on the left. The list on the right displays the package groups for the currently selected category.

To specify a package group for installation, select the check box next to the group. The box at the bottom of the screen displays the details of the package group that is currently highlighted. *None* of the packages from a group will be installed unless the check box for that group is selected.

If you select a package group, Red Hat Enterprise Linux automatically installs the base and mandatory packages for that group. To change which optional packages within a selected group will be installed,

select the **Optional Packages** button under the description of the group. Then use the check box next to an individual package name to change its selection.

In the package selection list on the right, you can use the context menu as a shortcut to select or de-select base and mandatory packages or all optional packages.

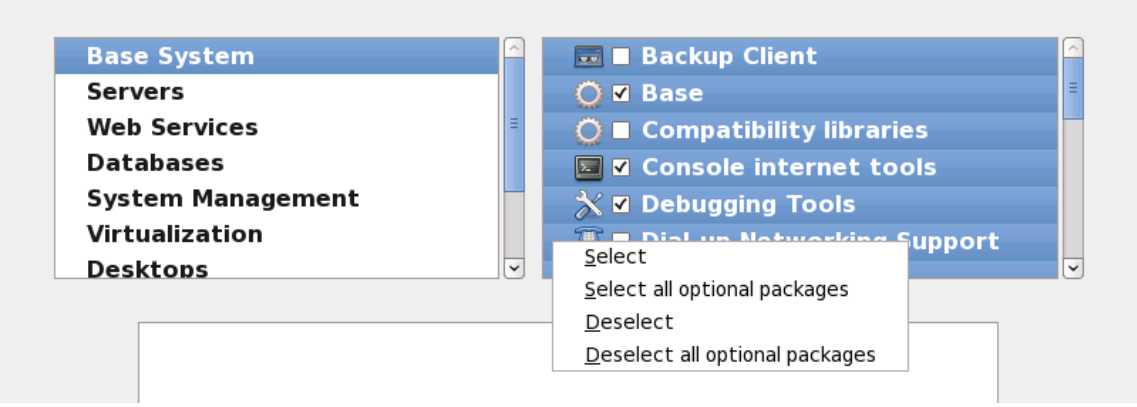


Figure 23.52. Package Selection List Context Menu

After you choose the desired packages, select **Next** to proceed. The installer checks your selection, and automatically adds any extra packages required to use the software you selected. When you have finished selecting packages, click **Close** to save your optional package selections and return to the main package selection screen.

The packages that you select are not permanent. After you boot your system, use the **Add/Remove Software** tool to either install new software or remove installed packages. To run this tool, from the main menu, select **System → Administration → Add/Remove Software**. The Red Hat Enterprise Linux software management system downloads the latest packages from network servers, rather than using those on the installation discs.

23.17.2.1. Core Network Services

All Red Hat Enterprise Linux installations include the following network services:

- centralized logging through syslog
- email through SMTP (Simple Mail Transfer Protocol)
- network file sharing through NFS (Network File System)
- remote access through SSH (Secure SHell)
- resource advertising through mDNS (multicast DNS)

The default installation also provides:

- network file transfer through HTTP (HyperText Transfer Protocol)
- printing through CUPS (Common UNIX Printing System)
- remote desktop access through VNC (Virtual Network Computing)

Some automated processes on your Red Hat Enterprise Linux system use the email service to send reports and messages to the system administrator. By default, the email, logging, and printing services do not accept connections from other systems. Red Hat Enterprise Linux installs the NFS sharing, HTTP,

and VNC components without enabling those services.

You may configure your Red Hat Enterprise Linux system after installation to offer email, file sharing, logging, printing and remote desktop access services. The SSH service is enabled by default. You may use NFS to access files on other systems without enabling the NFS sharing service.

23.18. INSTALLING PACKAGES

At this point there is nothing left for you to do until all the packages have been installed. How quickly this happens depends on the number of packages you have selected and your computer's speed.

Depending on the available resources, you might see the following progress bar while the installer resolves dependencies of the packages you selected for installation:

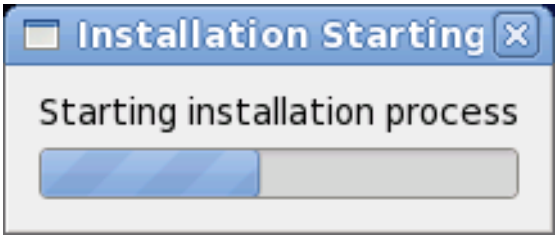


Figure 23.53. Starting installation

During installation of the selected packages and their dependencies, you see the following progress bar:

Figure 23.54. Packages completed

23.19. INSTALLATION COMPLETE

Congratulations! Your Red Hat Enterprise Linux installation is now complete!

The installation program prompts you to prepare your system for reboot.

The installation program automatically reboots into the installed system.

Should the installation program not reboot, the installation program shows information from which device to do an IPL (boot). Accept the shutdown option and after shutdown, IPL from the DASD or SCSI LUN where the **/boot** partition for Red Hat Enterprise Linux has been installed.

23.19.1. IPL Under z/VM

To IPL from a DASD, for example using the DASD device 200 on the 3270 console, issue the command:

-

```
#cp i 200
```

In DASD only environments where automatic partitioning (clearing data from all partitions) was used, the first activated DASD is where the **/boot** partition is typically located.

Using **/boot** on an FCP LUN, you must provide the WWPN and LUN for the FCP-attached device from which to IPL.

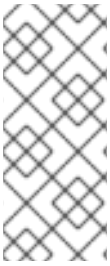
To IPL from an FCP-attached device:

1. Provide FCP routing information to an FCP-attached device, for example, where **0x50050763050B073D** is the WWPN, and **0x4020400100000000** is the FCP LUN:

   ```
   #cp set loaddev portname50050763 050B073D lun 40204001 00000000
   ```

2. IPL the FCP adapter, for example **FC00**:

   ```
   #cp ipl FC00
   ```



NOTE

To disconnect from the 3270 terminal without stopping the Linux running in your virtual machine, use **#cp disconnect** instead of **#cp logoff**. When your virtual machine is re-connected using the usual logon procedure, it might be placed in CP console function mode (**CP READ**). If so, to resume execution on your virtual machine, enter the **BEGIN** command.

23.19.2. IPL on an LPAR

For LPAR-based installations, on the HMC, issue a load command to the LPAR, specifying the particular DASD, or the FCP adapter, WWPN, and FCP LUN where the **/boot** partition is located.

23.19.3. Continuing After Reboot (re-IPL)

Following the automatic reboot or the manual IPL of the installed Red Hat Enterprise Linux operating system, you can log on to the system via **ssh**. Note that the only place from which you can log in as root is from the 3270 terminal or from other terminal devices listed in **/etc/securetty**.

The first time you start your Red Hat Enterprise Linux system in a graphical environment, you can use **FirstBoot** to guide you through Red Hat Enterprise Linux configuration. Using this tool, you can set your system time and date, install software, register your machine with Red Hat Network, and more. **FirstBoot** lets you configure your environment at the beginning, so that you can get started using your Red Hat Enterprise Linux system quickly.

Chapter 34, *Firstboot* will guide you through the configuration process.

---

[11] A root password is the administrative password for your Red Hat Enterprise Linux system. You should only log in as root when needed for system maintenance. The root account does not operate within the restrictions placed on normal user accounts, so changes made as root can have implications for your entire system.

[12] The **fsck** application is used to check the file system for metadata consistency and optionally repair one or more Linux file systems.

CHAPTER 24. TROUBLESHOOTING INSTALLATION ON IBM SYSTEM Z

This section discusses some common installation problems and their solutions.

For debugging purposes, **anaconda** logs installation actions into files in the **/tmp** directory. These files include:

/tmp/anaconda.log

general **anaconda** messages

/tmp/program.log

all external programs run by **anaconda**

/tmp/storage.log

extensive storage module information

/tmp/yum.log

yum package installation messages

/tmp/syslog

hardware-related system messages

If the installation fails, the messages from these files are consolidated into **/tmp/anaconda-tb-*identifier***, where *identifier* is a random string.

All of the files above reside in the installer's ramdisk and are thus volatile. To make a permanent copy, copy those files to another system on the network using **scp** on the installation image (not the other way round).

24.1. YOU ARE UNABLE TO BOOT RED HAT ENTERPRISE LINUX

24.1.1. Is Your System Displaying Signal 11 Errors?

A signal 11 error, commonly known as a *segmentation fault*, means that the program accessed a memory location that was not assigned to it. A signal 11 error may be due to a bug in one of the software programs that is installed, or faulty hardware.

Ensure that you have the latest installation updates and images from Red Hat. Review the online errata to see if newer versions are available.

24.2. TROUBLE DURING THE INSTALLATION

24.2.1. The "No devices found to install Red Hat Enterprise Linux" Error Message

If you receive an error message stating **No devices found to install Red Hat Enterprise Linux**, then there may be an issue with your DASD devices. If you encounter this error, add the **DASD=<*disks*>** parameter to your parameter file or CMS configuration file (where *disks* is the DASD range reserved for installation) and start the install again.

Additionally, make sure you format the DASDs using the **dasdfmt** command within a Linux root shell, instead of formatting the DASDs using CMS. **Anaconda** automatically detects any DASD devices that are not yet formatted and asks you whether to format the devices.

24.2.2. Saving Traceback Messages

If **anaconda** encounters an error during the graphical installation process, it presents you with a crash reporting dialog box:

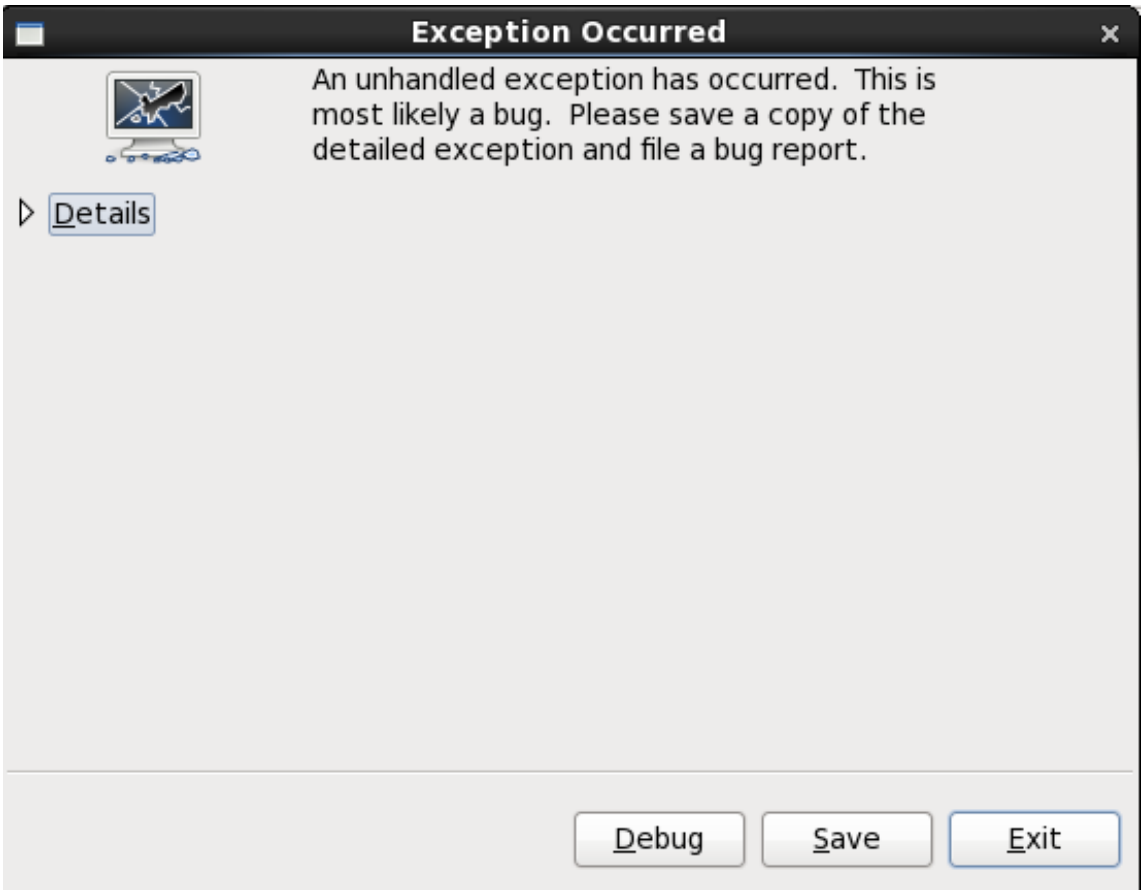


Figure 24.1. The Crash Reporting Dialog Box

Details

shows you the details of the error:

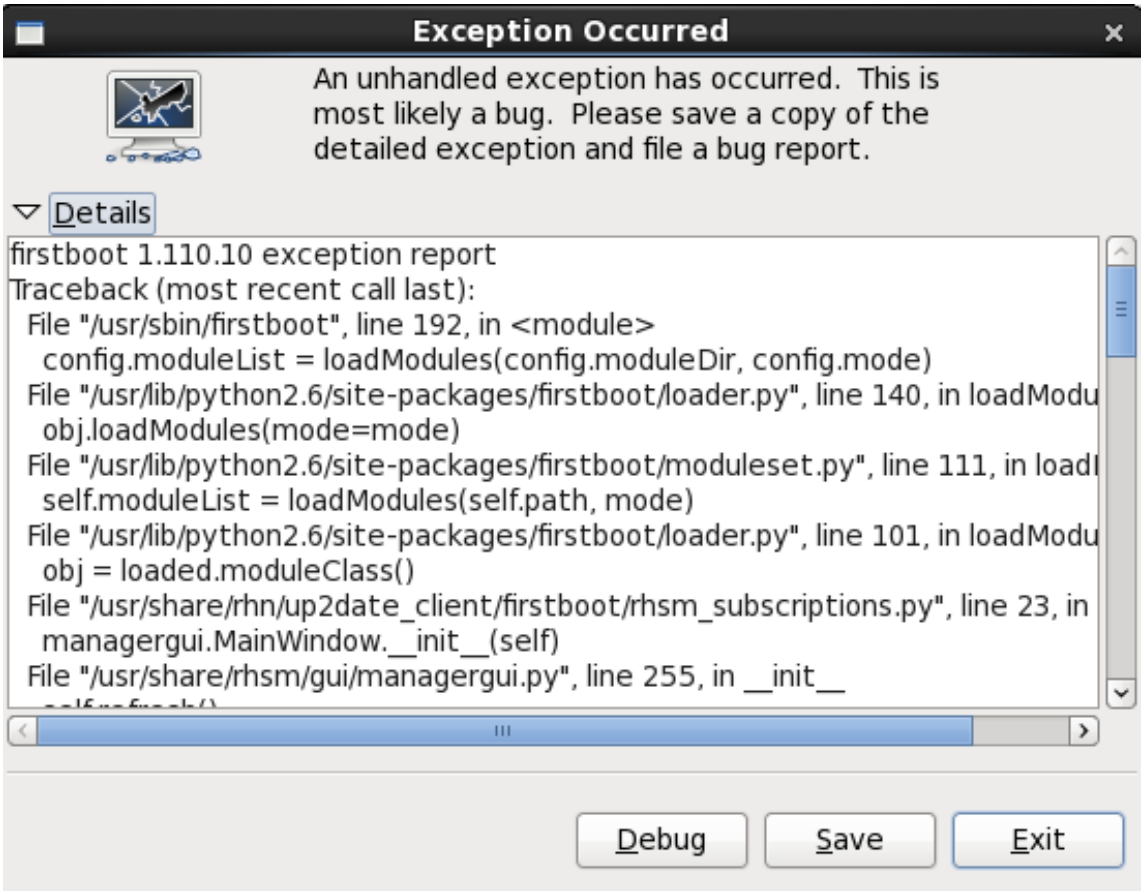


Figure 24.2. Details of the Crash

Save

saves details of the error locally or remotely:

Exit

exits the installation process.

If you select **Save** from the main dialog, you can choose from the following options:

Figure 24.3. Select reporter

Logger

saves details of the error as a log file to the local hard drive at a specified location.

Red Hat Customer Support

submits the crash report to Customer Support for assistance.

Report uploader

uploads a compressed version of the crash report to Bugzilla or a URL of your choice.

Before submitting the report, click **Preferences** to specify a destination or provide authentication details. Select the reporting method you need to configure and click **Configure Event**.

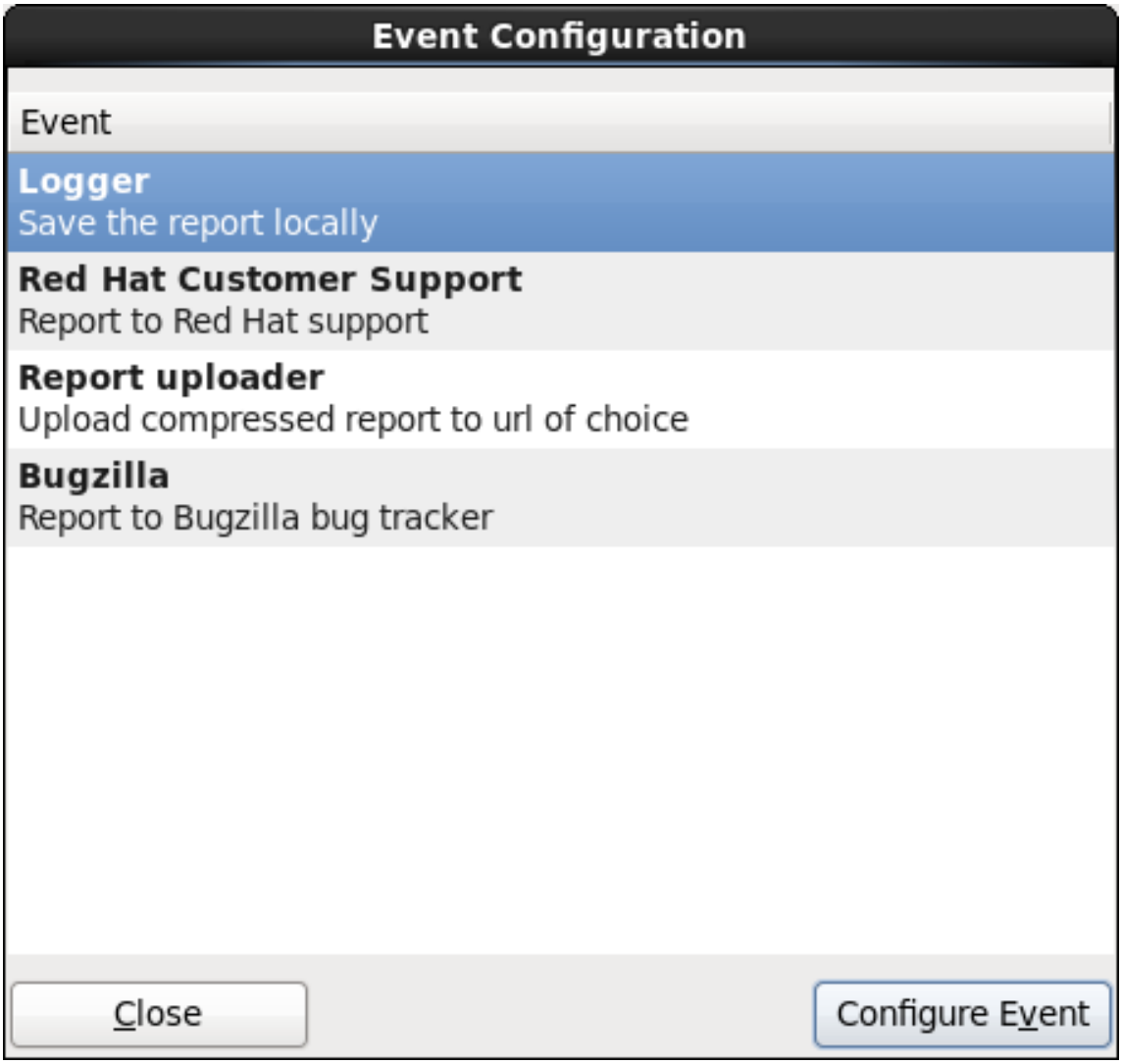


Figure 24.4. Configure reporter preferences

Logger

Specify a path and a filename for the log file. Check **Append** if you are adding to an existing log file.

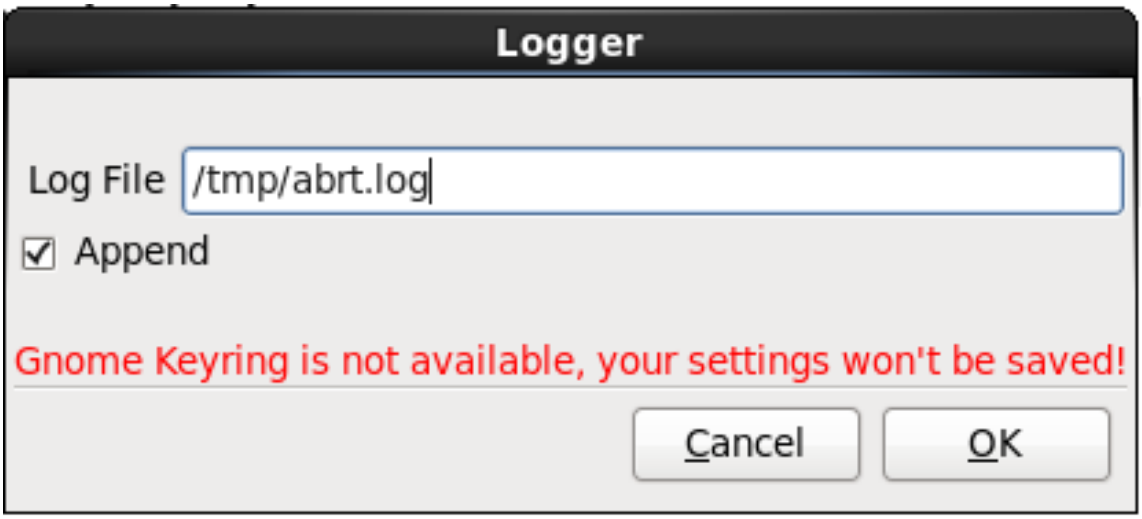


Figure 24.5. Specify local path for log file

Red Hat Customer Support

Enter your Red Hat Network username and password so your report reaches Customer Support and is linked with your account. The URL is prefilled and **Verify SSL** is checked by default.

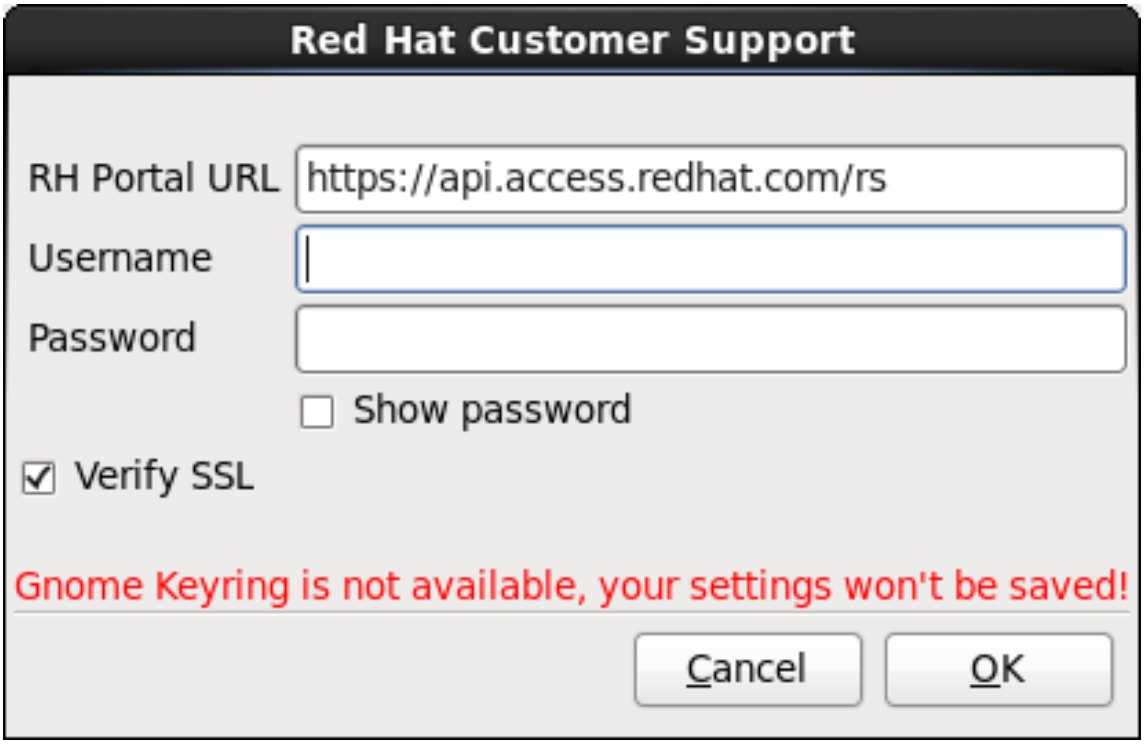


Figure 24.6. Enter Red Hat Network authentication details

Report uploader

Specify a URL for uploading a compressed version of the crash report.

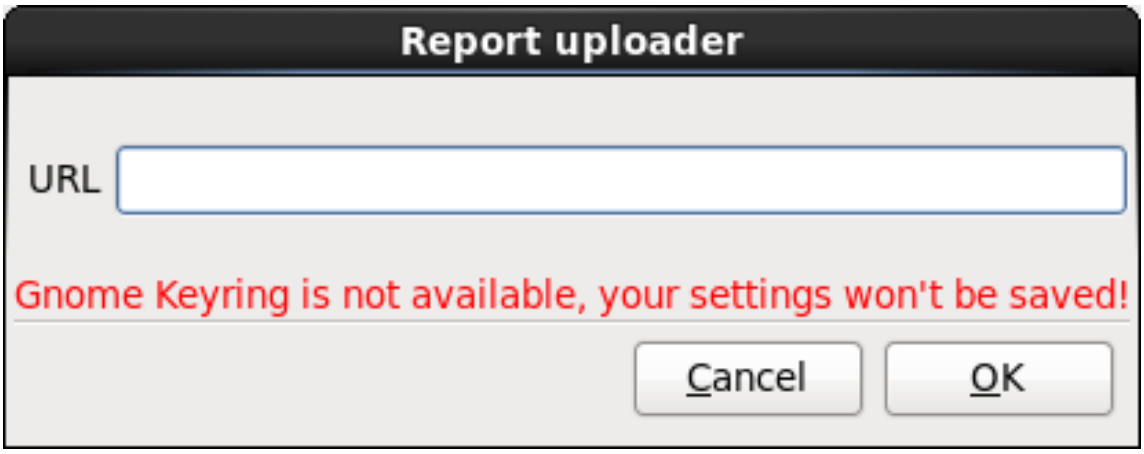


Figure 24.7. Enter URL for uploading crash report

Bugzilla

Enter your Bugzilla username and password to lodge a bug with Red Hat's bug-tracking system using the crash report. The URL is prefilled and **Verify SSL** is checked by default.

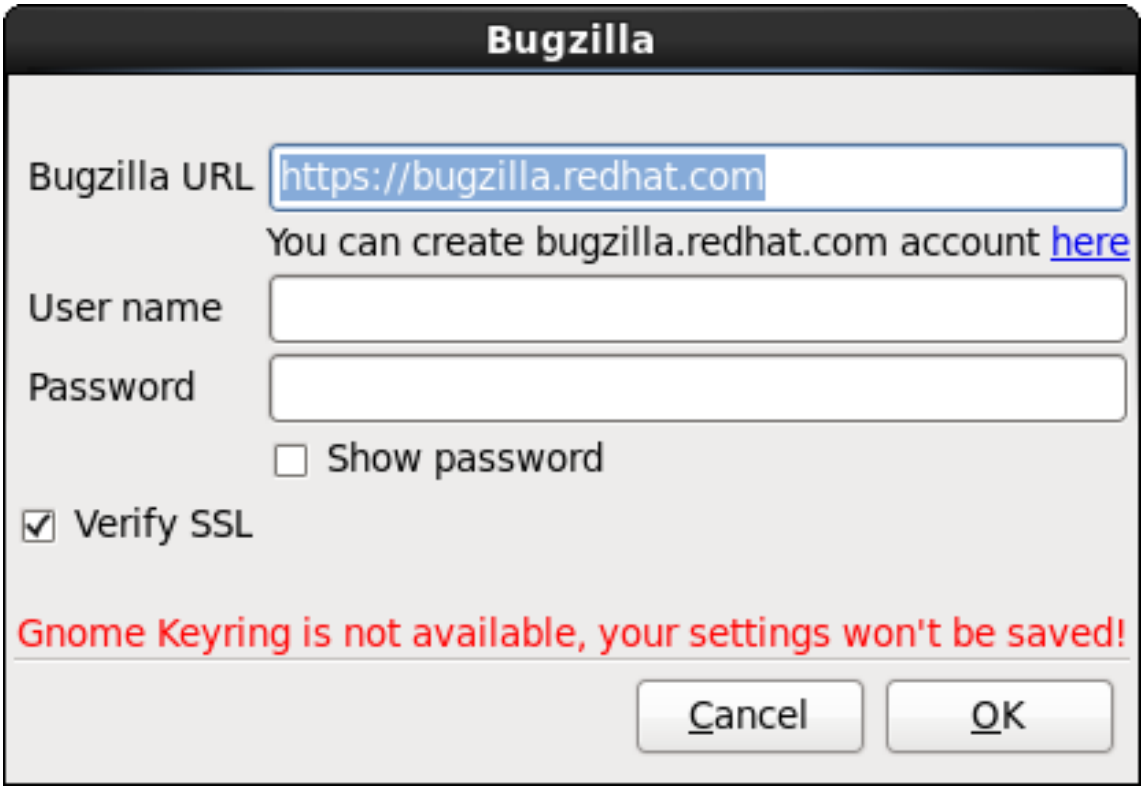


Figure 24.8. Enter Bugzilla authentication details

Once you have entered your preferences, click **OK** to return to the report selection dialog. Select how you would like to report the problem and then click **Forward**.

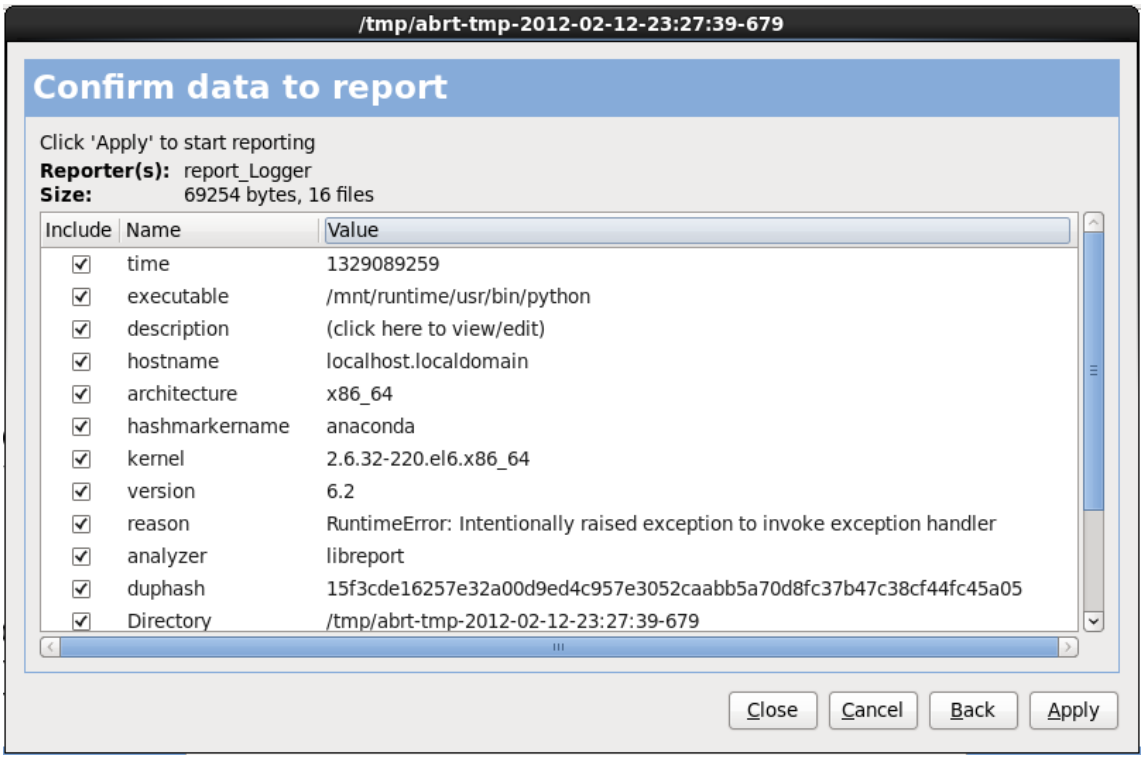


Figure 24.9. Confirm report data

You can now customize the report by checking and unchecking the issues that will be included. When finished, click **Apply**.

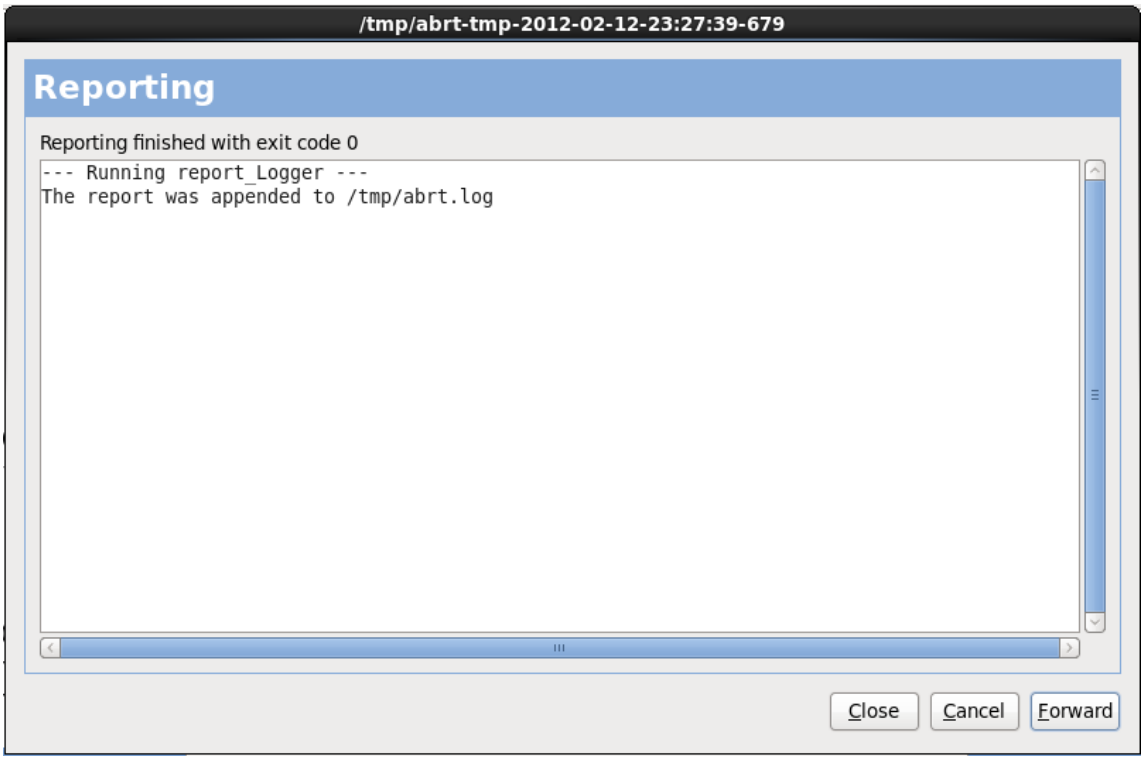


Figure 24.10. Report in progress

This screen displays the outcome of the report, including any errors in sending or saving the log. Click **Forward** to proceed.

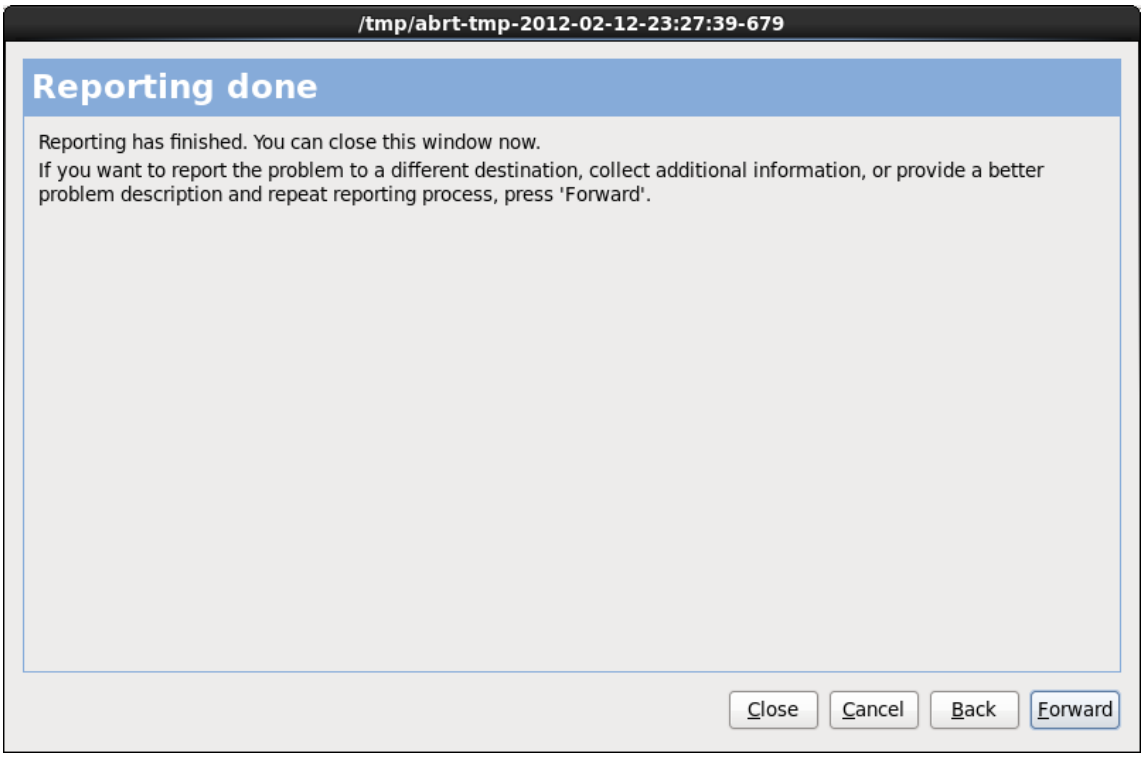


Figure 24.11. Reporting done

Reporting is now complete. Click **Forward** to return to the report selection dialog. You can now make another report, or click **Close** to exit the reporting utility and then **Exit** to close the installation process.

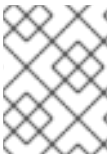
24.2.3. Other Partitioning Problems

If you create partitions manually, but cannot move to the next screen, you probably have not created all the partitions necessary for installation to proceed.

You must have the following partitions as a bare minimum:

- A / (root) partition
- A <swap> partition of type swap

Refer to Section 23.15.5, "Recommended Partitioning Scheme" for more information.



NOTE

When defining a partition's type as swap, do not assign it a mount point. **Anaconda** automatically assigns the mount point for you.

24.3. PROBLEMS AFTER INSTALLATION

24.3.1. Remote Graphical Desktops and XDMCP

If you have installed the X Window System and would like to log in to your Red Hat Enterprise Linux system using a graphical login manager, enable the *X Display Manager Control Protocol* (XDMCP). This protocol allows users to remotely log in to a desktop environment from any X Window System compatible client (such as a network-connected workstation or X11 terminal).

To enable remote login using XDMCP, edit the **/etc/gdm/custom.conf** file on the Red Hat Enterprise Linux system with a text editor such as **vi** or **nano**. In the **[xdcmp]** section, add the line **Enable=true**, save the file, and exit the text editor.

To enable this change, you will need to restart the X Window System. First, switch to runlevel 4:

```
/sbin/init 4
```

The graphical display will close, leaving only a terminal. When you reach the **login:** prompt, enter your username and password.

Then, as root in the terminal, switch to runlevel 5 to return to the graphical interface and start the X11 server:

```
/sbin/init 5
```

From the client machine, start a remote X11 session using **X**. For example:

```
X :1 -query s390vm.example.com
```

The command connects to the remote X11 server via XDMCP (replace *s390vm.example.com* with the hostname of the remote X11 server) and displays the remote graphical login screen on display **:1** of the X11 server system (usually accessible by using the **Ctrl**-**Alt**-**F8** key combination).

You can also access remote desktop sessions using a *nested* X11 server, which opens the remote desktop as a window in your current X11 session. **Xnest** allows users to open a remote desktop nested within their local X11 session. For example, run **Xnest** using the following command, replacing *s390vm.example.com* with the hostname of the remote X11 server:

```
Xnest :1 -query s390vm.example.com
```

24.3.2. Problems When You Try to Log In

If you did not create a user account in the **firstboot** screens, switch to a console by pressing **Ctrl**+**Alt**+**F2**, log in as root and use the password you assigned to root.

If you cannot remember your root password, boot your system into single user mode by appending the boot option **single** to the zipl boot menu or by any other means to append kernel command line options at IPL.

Once you have booted into single user mode and have access to the **#** prompt, you must type **passwd root**, which allows you to enter a new password for root. At this point you can type **shutdown -r now** to reboot the system with the new root password.

If you cannot remember your user account password, you must become root. To become root, type **su -** and enter your root password when prompted. Then, type **passwd <username>**. This allows you to enter a new password for the specified user account.

If the graphical login screen does not appear, check your hardware for compatibility issues. The *Hardware Compatibility List* can be found at:

```
https://hardware.redhat.com/
```

24.3.3. Your Printer Does Not Work

If you are not sure how to set up your printer or are having trouble getting it to work properly, try using the **Printer Configuration Tool**.

Type the **system-config-printer** command at a shell prompt to launch the **Printer Configuration Tool**. If you are not root, it prompts you for the root password to continue.

24.3.4. Apache HTTP Server or Sendmail Stops Responding During Startup

If **Apache HTTP Server** (**httpd**) or **Sendmail** stops responding during startup, make sure the following line is in the **/etc/hosts** file:

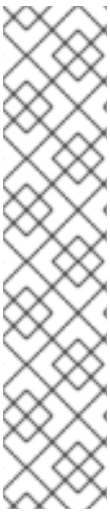
```
127.0.0.1  localhost.localdomain  localhost
```

CHAPTER 25. CONFIGURING AN INSTALLED LINUX ON SYSTEM Z INSTANCE

For more information about Linux on System z, see the publications listed in Chapter 27, *IBM System z References*. Some of the most common tasks are described here.

25.1. ADDING DASDS

This section explains how to set a *Direct Access Storage Device* (DASD) online, format it, and how to make sure it is attached to the system persistently, making it automatically available after a reboot.



NOTE

Make sure 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, issue, for example:

```
CP LINK RHEL6X 4B2E 4B2E MR
DASD 4B2E LINKED R/W
```

See *z/VM: CP Commands and Utilities Reference, SC24-6175* for details about these commands.

25.1.1. Dynamically Setting DASDs Online

The following procedure describes bringing a DASD online dynamically (not persistently). This is the first step when configuring a new DASD; later procedures will explain how to make it available persistently.

Procedure 25.1. Adding DASD Disks on IBM System z Using the VMCP Driver

1. Enable the **VMCP** driver:

   ```
   # modprobe vmcp
   ```

2. Use the **cio_ignore** command to remove the DASD from the list of ignored devices and make it visible to Linux:

   ```
   # cio_ignore -r DeviceNumber
   ```

   Replace *DeviceNumber* with the device number of the DASD. For example:

   ```
   # cio_ignore -r 0102
   ```

3. Link the disk to the virtual machine:

   ```
   # vmcp 'link * DeviceNumber DeviceNumber rw'
   ```

   Replace *DeviceNumber* with the device number of the DASD.

4. Set the device online. Use a command of the following form:

```
# # chccwdev -e DeviceNumber
```

Replace *DeviceNumber* with the device number of the DASD.

5. Verify that the disk is ready using the **lsdasd** command:

```
# lsdasd
Bus-ID     Status      Name      Device  Type  BlkSz  Size      Blocks
==============================================================================
===
0.0.0100   active      dasda     94:0    ECKD  4096   2347MB    600840
0.0.0301   active      dasdb     94:4    FBA   512    512MB     1048576
0.0.0300   active      dasdc     94:8    FBA   512    256MB     524288
0.0.0101   active      dasdd     94:12   ECKD  4096   2347MB    600840
0.0.0200   active      dasde     94:16   ECKD  4096   781MB     200160
0.0.0102   active      dasdf     94:20   ECKD  4096   2347MB    600840
```

In the above example, device 0102 (shown as **0.0.0102** in the **Bus-ID** column) is being accessed as **/dev/dasdf**.

If you followed the above procedure, the new DASD is attached for the current session only. This means that the DASD will not still be attached after you reboot the system. See Section 25.1.2, "Persistently setting DASDs online" for information about attaching the storage device permanently.

You can also find more information in the DASD Chapter in *Linux on System z Device Drivers, Features, and Commands on Red Hat Enterprise Linux 6*.

25.1.2. Persistently setting DASDs online

The instructions in Section 25.1.1, "Dynamically Setting DASDs Online" described how to activate DASDs dynamically in a running system. Such changes are not persistent; the DASDs will not be attached after the system reboots. Procedures described in this section assume that you have already attached the DASD dynamically.

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 early during the boot process by the **initramfs** to be able to mount the root file system. The DASDs which are not part of the root file system can be activated later, simplifying the configuration process.

The list of ignored devices (**cio_ignore**) is handled transparently for persistent device configurations. You do not need to free devices from the ignore list manually.

25.1.2.1. DASDs Which Are Part of the Root File System

If you are attaching a new DASD as part of the root file system, you will have to edit the **zipl** boot loader's configuration and then regenerate the **initramfs** so that your changes will take effect after the next reboot. The following procedure explains the steps you need to take.

Procedure 25.2. Permanently Attaching DASDs as Root Devices

1. Edit the **/etc/dasd.conf** configuration file using a plain text editor such as **Vim**, and append a line to this file with your DASD's configuration. You can use parts of the file that describe

previously configured devices for reference. A valid configuration line will look similar to the following:

```
0.0.0102 use_diag=0 readonly=0 erplog=0 failfast=0
```

2. Edit the **/etc/zipl.conf** configuration file. An example **zipl.conf** file will look similar to the following:

```
[defaultboot]
default=linux
target=/boot/
[linux]
  image=/boot/vmlinuz-2.6.32-19.el6.s390x
  ramdisk=/boot/initramfs-2.6.32-19.el6.s390x.img
  parameters="root=/dev/mapper/vg_devel1-lv_root
rd_DASD=0.0.0200,use_diag=0,readonly=0,erplog=0,failfast=0
rd_DASD=0.0.0207,use_diag=0,readonly=0,erplog=0,failfast=0
rd_LVM_LV=vg_devel1/lv_root rd_NO_LUKS rd_NO_MD rd_NO_DM LANG=en_US.UTF-8
SYSFONT=latarcyrheb-sun16 KEYTABLE=us cio_ignore=all,!0.0.0009"
```

Note the multiple **rd_DASD=** options on the **parameters=** line. You must add the new DASD to this line, using the same syntax - the **rd_DASD=** keyword, followed by the device ID and a comma-separated list of options. See the **dasd=** parameter description in the DASD device driver chapter in *Linux on System z Device Drivers, Features, and Commands on Red Hat Enterprise Linux 6* for details.

3. The next step is to rebuild the **initrd**:

```
# mkinitrd -f /boot/initramfs-2.6.32-71.el6.s390x.img `uname -r`
```

4. Then, rebuild the boot loader configuration using the **zipl** command. You can use the **-V** option for more detailed output:

```
# zipl -V
Using config file '/etc/zipl.conf'
Target device information
  Device.....: 5e:00
  Partition.....: 5e:01
  Device name.....: dasda
  DASD device number.....: 0201
  Type.....: disk partition
  Disk layout.....: ECKD/compatible disk layout
  Geometry - heads.....: 15
  Geometry - sectors.....: 12
  Geometry - cylinders.....: 3308
  Geometry - start.....: 24
  File system block size.....: 4096
  Physical block size.....: 4096
  Device size in physical blocks..: 595416
Building bootmap in '/boot/'
Building menu 'rh-automatic-menu'
Adding #1: IPL section 'linux' (default)
  kernel image.....: /boot/vmlinuz-2.6.32-19.el6.s390x
  kernel parmline...: 'root=/dev/mapper/vg_devel1-lv_root
```

```
rd_DASD=0.0.0200,use_diag=0,readonly=0,erplog=0,failfast=0
rd_DASD=0.0.0207,use_diag=0,readonly=0,erplog=0,failfast=0
rd_LVM_LV=vg_devel1/lv_root rd_NO_LUKS rd_NO_MD rd_NO_DM LANG=en_US.UTF-8
SYSFONT=latarcyrheb-sun16 KEYTABLE=us cio_ignore=all,!0.0.0009'
  initial ramdisk...: /boot/initramfs-2.6.32-19.el6.s390x.img
  component address:
    kernel image....: 0x00010000-0x00a70fff
    parmline........: 0x00001000-0x00001fff
    initial ramdisk.: 0x02000000-0x022d2fff
    internal loader.: 0x0000a000-0x0000afff
Preparing boot device: dasda (0201).
Preparing boot menu
  Interactive prompt......: enabled
  Menu timeout.............: 15 seconds
  Default configuration...: 'linux'
Syncing disks...
Done.
```

After completing this procedure, the new DASD is persistently attached and can be used as part of the root file system. However, the root file system still needs to be expanded to the new DASD. If your system uses an LVM logical volume for the root file system, you will also need to expand this volume (and the volume group which contains it) to the new DASD. This can be done using the built-in **pvcreate**, **vgextend** and **lvextend** commands to create a physical volume for LVM, expand the existing volume group and expand the root logical volume, respectively. See Section 25.1.5, “Expanding Existing LVM Volumes to New Storage Devices” for details.

25.1.3. DASDs Which Are Not Part of the Root File System

DASDs that are not part of the root file system, that is, *data disks*, are persistently configured in the file **/etc/dasd.conf**. It contains one DASD per line. Each line begins with the device bus ID of a DASD. Optionally, each line can continue with options separated by space or tab characters. Options consist of key-value-pairs, where the key and value are separated by an equals sign.

The key corresponds to any valid **sysfs** attribute a DASD may have. The value will be written to the key's **sysfs** attribute. Entries in **/etc/dasd.conf** are activated and configured by udev when a DASD is added to the system. At boot time, all DASDs visible to the system get added and trigger **udev**.

Example content of **/etc/dasd.conf**:

```
0.0.0207
0.0.0200 use_diag=1 readonly=1
```

Modifications of **/etc/dasd.conf** only become effective after a reboot of the system or after the dynamic addition of a new DASD by changing the system's I/O configuration (that is, the DASD is attached under z/VM). Alternatively, you can trigger the activation of a new entry in **/etc/dasd.conf** for a DASD which was previously not active, by executing the following commands:

Procedure 25.3. Permanently Attaching DASDs as Non-root Devices

- Trigger the activation by writing to the ***uevent*** attribute of the device:

  ```
  echo add > /sys/bus/ccw/devices/device.bus,ID/uevent
  ```

  For example:

  -

```
echo add > /sys/bus/ccw/devices/0.0.021a/uevent
```

25.1.4. Preparing a New DASD with Low-level Formatting

The next step after bringing the DASD online is to format it, if you need to do so. The following procedure explains the necessary steps.

WARNING

This procedure will wipe all existing data on the disk. Make sure to back up any data you want to keep before proceeding.

Procedure 25.4. Formatting a DASD

1. Wipe all existing data on the DASD using the **dasdfmt** command. Replace *DeviceNumber* with the device number of the DASD. When prompted for confirmation (as shown in the example below), type **yes** to proceed.

```
# dasdfmt -b 4096 -d cdl -p /dev/disk/by-path/ccw-0.0.DeviceNumber
Drive Geometry: 10017 Cylinders * 15 Heads =  150255 Tracks

I am going to format the device /dev/disk/by-path/ccw-0.0.0102 in the following way:
   Device number of device : 0x4b2e
   Labelling device        : yes
   Disk label              : VOL1
   Disk identifier         : 0X0102
   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%
```

   When the progress bar reaches the end and the format is complete, **dasdfmt** prints the following output:

```
Rereading the partition table...
Exiting...
```

   See the **dasdfmt(8)** man page for information about the syntax of the **dasdfmt** command.

2. Use the **fdasd** command to write a new Linux-compatible partition table to the DASD. Replace *DeviceNumber* with the device number of the DASD.

```
# fdasd -a /dev/disk/by-path/ccw-DeviceNumber
auto-creating one partition for the whole disk...
```

```
writing volume label...
writing VTOC...
checking !
wrote NATIVE!
rereading partition table...
```

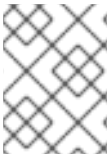
This example uses the **-a** option to create a single partition spanning the entire disk. Other layouts are possible; up to three partitions can be created on a single DASD. For information about the syntax of the **fdasd** command and available options, see the **fdasd(8)** man page.

3. Create a new partition with **fdisk**. Replace *DeviceName* with the device name of the DASD.

```
# fdisk /dev/DeviceName
```

After you execute **fdisk**, a series of prompts will appear in your terminal. These prompts can be used to manipulate the disk partition table, creating new partitions or editing existing one. For information about using **fdisk**, see the **fdisk(8)** man page.

After a (low-level formatted) DASD is online, it can be used like any other disk under Linux. For instance, 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.



NOTE

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/**.

25.1.5. Expanding Existing LVM Volumes to New Storage Devices

If your system uses LVM, you need to expand an existing volume group and one or more logical volumes so that they contain the new DASD which you attached by following the procedures described earlier in this chapter. Otherwise, the DASD will be attached to the system, but you will not be able to use it.

The following procedure explains how to use the entire capacity of the new DASD to expand an existing logical volume. If you want to use the new DASD for multiple logical volumes, you will need to create multiple LVM physical volumes on this partition, and repeat this procedure for each logical volume (and volume group) you want to expand. This procedure assumes you followed the steps in Section 25.1.1, "Dynamically Setting DASDs Online" to attach the new DASD dynamically, then Section 25.1.2.1, "DASDs Which Are Part of the Root File System" to attach it persistently and prepare it to be used for the root volume, and that you formatted it as described in Section 25.1.4, "Preparing a New DASD with Low-level Formatting" and created a single partition on it.

Procedure 25.5. Expanding Existing Logical Volume to Use a New DASD

1. Create a new physical volume for LVM on the DASD using the **pvcreate** command:

```
# pvcreate /dev/DeviceName
```

IMPORTANT

The device name must be specified as a *partition* - for example, **/dev/dasdf1**. Do not specify the entire block device.

2. List existing physical volumes using the **pvs** command to verify that the physical volume has been created:

```
# pvs
PV               VG            Fmt  Attr PSize   PFree
 /dev/dasda2     vg_local      lvm2 a--   1,29g      0
 /dev/dasdd1     vg_local      lvm2 a--   2,29g      0
 /dev/dasdf1                   lvm2 a--   2,29g   2,29g
 /dev/mapper/mpathb vgextnotshared lvm2 a--  200,00g 1020,00m
```

As you can see in the above example, **/dev/dasdf1** now contains an empty physical volume which is not assigned to any volume group.

3. Use the **vgextend** command to expand an existing volume group containing the volume you want to use the new DASD for:

```
# vgextend VolumeGroup PhysicalVolume
```

Replace *VolumeGroup* with the name of the volume group you are expanding, and *PhysicalVolume* with the name of the physical volume (for example, **/dev/dasdf1**).

4. Use the **lvextend** command to expand a logical volume you want to use the new DASD for:

```
# lvextend -L +Size /dev/mapper/VolumeGroup-LogicalVolume
```

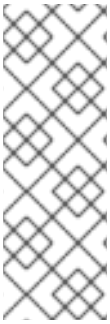
For example:

```
# lvextend -L +2G /dev/mapper/vg_local-lv_root
Extending logical volume lv_root to 2,58 GiB
Logical volume lv_root successfully resized
```

After you complete the procedure, an existing logical volume is expanded and contains the new DASD in addition to any previously assigned storage devices. You can also use the **pvs**, **vgs**, and **lvs** commands as **root** to view existing LVM physical volumes, volume groups and logical volumes at any point during the procedure.

25.2. ADDING FCP-ATTACHED LOGICAL UNITS (LUNS)

The following is an example of how to add an FCP LUN.



NOTE

If running under z/VM, make sure the FCP adapter is attached to the z/VM guest virtual machine. For multipathing in production environments there would be at least two FCP devices on two different physical adapters (CHPIDs). For example:

```
CP ATTACH FC00 TO *
CP ATTACH FCD0 TO *
```

25.2.1. Dynamically Activating an FCP LUN

Follow these steps to activate a LUN:

1. Use the **cio_ignore** command to remove the FCP adapter from the list of ignored devices and make it visible to Linux:

   ```
   # cio_ignore -r DeviceNumber
   ```

   Replace *DeviceNumber* with the device number of the FCP adapter. For example:

2. To bring the FCP adapter device online, use the following command:

   ```
   # chccwdev -e fc00
   ```

3. Verify that the required WWPN was found by the automatic port scanning of the zfcp device driver:

   ```
   # ls -l /sys/bus/ccw/drivers/zfcp/0.0.fc00/
   drwxr-xr-x.  3 root root    0 Apr 28 18:19 0x500507630040710b
   drwxr-xr-x.  3 root root    0 Apr 28 18:19 0x50050763050b073d
   drwxr-xr-x.  3 root root    0 Apr 28 18:19 0x500507630e060521
   drwxr-xr-x.  3 root root    0 Apr 28 18:19 0x500507630e860521
   -r--r--r--.  1 root root 4096 Apr 28 18:17 availability
   -r--r--r--.  1 root root 4096 Apr 28 18:19 card_version
   -rw-r--r--.  1 root root 4096 Apr 28 18:17 cmb_enable
   -r--r--r--.  1 root root 4096 Apr 28 18:17 cutype
   -r--r--r--.  1 root root 4096 Apr 28 18:17 devtype
   lrwxrwxrwx.  1 root root    0 Apr 28 18:17 driver ->  ../../../../bus/ccw/drivers/zfcp
   -rw-r--r--.  1 root root 4096 Apr 28 18:17 failed
   -r--r--r--.  1 root root 4096 Apr 28 18:19 hardware_version
   drwxr-xr-x. 35 root root    0 Apr 28 18:17 host0
   -r--r--r--.  1 root root 4096 Apr 28 18:17 in_recovery
   -r--r--r--.  1 root root 4096 Apr 28 18:19 lic_version
   -r--r--r--.  1 root root 4096 Apr 28 18:17 modalias
   -rw-r--r--.  1 root root 4096 Apr 28 18:17 online
   -r--r--r--.  1 root root 4096 Apr 28 18:19 peer_d_id
   -r--r--r--.  1 root root 4096 Apr 28 18:19 peer_wwnn
   -r--r--r--.  1 root root 4096 Apr 28 18:19 peer_wwpn
   --w-------.  1 root root 4096 Apr 28 18:19 port_remove
   --w-------.  1 root root 4096 Apr 28 18:19 port_rescan
   drwxr-xr-x.  2 root root    0 Apr 28 18:19 power
   -r--r--r--.  1 root root 4096 Apr 28 18:19 status
   lrwxrwxrwx.  1 root root    0 Apr 28 18:17 subsystem ->  ../../../../bus/ccw
   -rw-r--r--.  1 root root 4096 Apr 28 18:17 uevent
   ```

4. Activate the FCP LUN by adding it to the port (WWPN) through which you would like to access the LUN:

```
# echo 0x4020400100000000 >
/sys/bus/ccw/drivers/zfcp/0.0.fc00/0x50050763050b073d/unit_add
```

5. Find out the assigned SCSI device name:

```
# lszfcp -DV
/sys/devices/css0/0.0.0015/0.0.fc00/0x50050763050b073d/0x4020400100000000
/sys/bus/ccw/drivers/zfcp/0.0.fc00/host0/rport-0:0-21/target0:0:21/0:0:21:1089355792
```

For more information, refer to the chapter on SCSI-over-Fibre Channel in *Linux on System z Device Drivers, Features, and Commands on Red Hat Enterprise Linux 6*.

25.2.2. Persistently Activating FCP LUNs

The above instructions described how to activate FCP LUNs dynamically in a running system. However, such changes are not persistent and do not survive a reboot. How you make the changes to the FCP configuration persistent in your Linux system depends on whether the FCP LUNs belong to the root file system. Those 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. **cio_ignore** is handled transparently for persistent device configurations and you do not need to free devices from the ignore list manually.

25.2.2.1. 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 is **/etc/zipl.conf** followed by a run of the **zipl** boot loader tool. There is no more need to recreate the **initramfs**.

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 device bus ID, the 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 following example **zipl.conf** is for a system that uses physical volumes on partitions of two FCP LUNs for an LVM volume group **vg_devel1** that contains a logical volume **lv_root** for the root file system. For simplicity, the example shows a configuration without multipathing.

```
[defaultboot]
default=linux
target=/boot/
[linux]
image=/boot/vmlinuz-2.6.32-19.el6.s390x
ramdisk=/boot/initramfs-2.6.32-19.el6.s390x.img
parameters="root=/dev/mapper/vg_devel1-lv_root
rd_ZFCP=0.0.fc00,0x5105074308c212e9,0x401040a000000000
rd_ZFCP=0.0.fc00,0x5105074308c212e9,0x401040a100000000
rd_LVM_LV=vg_devel1/lv_root rd_NO_LUKS rd_NO_MD rd_NO_DM LANG=en_US.UTF-8
SYSFONT=latarcyrheb-sun16 KEYTABLE=us cio_ignore=all,!0.0.0009"
```

To add another physical volume on a partition of a third FCP LUN with device bus ID 0.0.fc00, WWPN 0x5105074308c212e9 and FCP LUN 0x401040a300000000, simply add **rd_ZFCP=0.0.fc00,0x5105074308c212e9,0x401040a300000000** to the parameters line of your boot

kernel in **zipl.conf**, for example:

```
[defaultboot]
default=linux
target=/boot/
[linux]
image=/boot/vmlinuz-2.6.32-19.el6.s390x
ramdisk=/boot/initramfs-2.6.32-19.el6.s390x.img
parameters="root=/dev/mapper/vg_devel1-lv_root
rd_ZFCP=0.0.fc00,0x5105074308c212e9,0x401040a000000000
rd_ZFCP=0.0.fc00,0x5105074308c212e9,0x401040a100000000
rd_ZFCP=0.0.fc00,0x5105074308c212e9,0x401040a300000000
rd_LVM_LV=vg_devel1/lv_root rd_NO_LUKS rd_NO_MD rd_NO_DM LANG=en_US.UTF-8
SYSFONT=latarcyrheb-sun16 KEYTABLE=us cio_ignore=all,!0.0.0009"
```

Run **zipl** to apply the changes of **/etc/zipl.conf** for the next IPL:

```
# zipl -V
Using config file '/etc/zipl.conf'
Target device information
Device.....: 08:00
Partition.....: 08:01
Device name.....: sda
Device driver name.....: sd
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 'rh-automatic-menu'
Adding #1: IPL section 'linux' (default)
kernel image.....: /boot/vmlinuz-2.6.32-19.el6.s390x
kernel parmline...: 'root=/dev/mapper/vg_devel1-lv_root
rd_ZFCP=0.0.fc00,0x5105074308c212e9,0x401040a000000000
rd_ZFCP=0.0.fc00,0x5105074308c212e9,0x401040a100000000
rd_ZFCP=0.0.fc00,0x5105074308c212e9,0x401040a300000000 rd_LVM_LV=vg_devel1/lv_root
rd_NO_LUKS rd_NO_MD rd_NO_DM LANG=en_US.UTF-8 SYSFONT=latarcyrheb-sun16
KEYTABLE=us cio_ignore=all,!0.0.0009'
initial ramdisk...: /boot/initramfs-2.6.32-19.el6.s390x.img
component address:
kernel image....: 0x00010000-0x007a21ff
parmline.....: 0x00001000-0x000011ff
initial ramdisk.: 0x02000000-0x028f63ff
internal loader.: 0x0000a000-0x0000a3ff
Preparing boot device: sda.
Detected SCSI PCBIOS disk layout.
Writing SCSI master boot record.
Syncing disks...
Done.
```

25.2.2.2. 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 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. 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
```

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 **cio_ignore** command to remove the FCP adapter from the list of ignored devices and make it visible to Linux:

   ```
   # cio_ignore -r DeviceNumber
   ```

   Replace *DeviceNumber* with the device number of the FCP adapter. For example:

   ```
   # cio_ignore -r fcfc
   ```

2. To trigger the uevent that activates the change, issue:

   ```
   echo add > /sys/bus/ccw/devices/Device.Bus.ID/uevent
   ```

   For example:

   ```
   echo add > /sys/bus/ccw/devices/0.0.fcfc/uevent
   ```

25.3. ADDING A NETWORK DEVICE

Network device driver modules are loaded automatically by **udev**.

You can add a network interface on IBM System z dynamically or persistently.

- Dynamically

  1. Load the device driver
  2. Remove the network devices from the list of ignored devices.
  3. Create the group device.
  4. Configure the device.
  5. Set the device online.

- Persistently

    1. Create a configuration script.

    2. Activate the interface.

The following sections provide basic information for each task of each IBM System z network device driver. Section 25.3.1, "Adding a qeth Device" describes how to add a qeth device to an existing instance of Red Hat Enterprise Linux. Section 25.3.2, "Adding an LCS Device" describes how to add an lcs device to an existing instance of Red Hat Enterprise Linux. Section 25.3.3, "Mapping Subchannels and Network Device Names" describes how persistent network device names work. Section 25.3.4, "Configuring a System z Network Device for Network Root File System" describes how to configure a network device to use with a root file system that is only accessible through the network.

25.3.1. Adding a qeth Device

The qeth network device driver supports System z OSA-Express features in QDIO mode, HiperSockets, z/VM guest LAN, and z/VM VSWITCH.

Based on the type of interface being added, the qeth device driver assigns one of the base interface names:

- hsi*n* for HiperSockets devices

- eth*n* for Ethernet features

The value *n* is an integer that uniquely identifies the device. *n* is **0** for the first device of that type, **1** for the second, and so on.

25.3.1.1. Dynamically Adding a qeth Device

To add a qeth device dynamically, follow these steps:

1. Determine whether the qeth device driver modules are loaded. The following example shows loaded qeth modules:

```
# lsmod | grep qeth
qeth_l3                127056  9
qeth_l2                 73008  3
ipv6                   492872  155ip6t_REJECT,nf_conntrack_ipv6,qeth_l3
qeth                   115808  2 qeth_l3,qeth_l2
qdio                    68240  1 qeth
ccwgroup                12112  2 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** command 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 command 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 (eth1)
```

5. Optionally, 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 (eth1)
```

Now you can continue to configure the network **eth1** interface.

Alternatively, you can use sysfs attributes to set the device online as follows:

1. 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
```

2. 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 may 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***

For information on additional parameters, refer to the chapter on the qeth device driver in *Linux on System z Device Drivers, Features, and Commands on Red Hat Enterprise Linux 6*.

3. Bring the device online by writing 1 to the online sysfs attribute:

```
# echo 1 > /sys/bus/ccwgroup/drivers/qeth/0.0.f500/online
```

4. 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.

5. Find the interface name that was assigned to the device:

```
# cat /sys/bus/ccwgroup/drivers/qeth/0.0.f500/if_name
eth1
```

Now you can continue to configure the network **eth1** interface.

The following command from the s390utils package shows the most important settings of your qeth device:

```
# lsqeth eth1
Device name                   : eth1
------------------------------------------------
        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
```

25.3.1.2. Dynamically Removing a qeth Device

To remove a qeth device, use the znetconf tool. For example:

1. Use the **znetconf** command to show you all configured network devices:

```
znetconf -c
Device IDs               Type    Card Type      CHPID Drv. Name       State
----------------------------------------------------------------------------
```

```
0.0.8036,0.0.8037,0.0.8038 1731/05 HiperSockets      FB qeth hsi1        online
0.0.f5f0,0.0.f5f1,0.0.f5f2 1731/01 OSD_1000          76 qeth eth0        online
0.0.f500,0.0.f501,0.0.f502 1731/01 GuestLAN QDIO     00 qeth eth1        online
```

2. Select the network device to be removed and trigger **znetconf** to set the device offline and ungroup the ccw group device.

```
# znetconf -r f500
Remove network device 0.0.f500 (0.0.f500,0.0.f501,0.0.f502)?
Warning: this may affect network connectivity!
Do you want to continue (y/n)?y
Successfully removed device 0.0.f500 (eth1)
```

3. Verify the success of the removal:

```
znetconf -c
Device IDs               Type    Card Type      CHPID Drv. Name        State
----------------------------------------------------------------------------
0.0.8036,0.0.8037,0.0.8038 1731/05 HiperSockets      FB qeth hsi1        online
0.0.f5f0,0.0.f5f1,0.0.f5f2 1731/01 OSD_1000          76 qeth eth0        online
```

25.3.1.3. Persistently Adding a qeth Device

To make your new qeth device persistent you need to create the configuration file for your new interface. The network interface configuration files are placed in **/etc/sysconfig/network-scripts/**.

The network configuration files use the naming convention **ifcfg-*device***, where *device* is the value found in the **if_name** file in the qeth group device that was created earlier. In this example it is **eth1**. **cio_ignore** is handled transparently for persistent device configurations and you do not need to free devices from the ignore list manually.

If a configuration file for another device of the same type already exists, the simplest solution is to copy it to the new name.

```
# cd /etc/sysconfig/network-scripts
# cp ifcfg-eth0 ifcfg-eth1
```

If you do not have a similar device defined you must create one. Use this example of **ifcfg-eth0** as a template:

/etc/sysconfig/network-scripts/ifcfg-eth0

```
# IBM QETH
DEVICE=eth0
BOOTPROTO=static
IPADDR=10.12.20.136
NETMASK=255.255.255.0
ONBOOT=yes
NETTYPE=qeth
SUBCHANNELS=0.0.09a0,0.0.09a1,0.0.09a2
PORTNAME=OSAPORT
OPTIONS='layer2=1 portno=0'
MACADDR=02:00:00:23:65:1a
TYPE=Ethernet
```

Edit the new ifcfg-eth1 file as follows:

1. Modify the ***DEVICE*** statement to reflect the contents of the **if_name** file from your ccwgroup.
2. Modify the ***IPADDR*** statement to reflect the IP address of your new interface.
3. Modify the ***NETMASK*** statement as needed.
4. If the new interface is to be activated at boot time, then make sure ***ONBOOT*** is set to **yes**.
5. Make sure the ***SUBCHANNELS*** statement matches the hardware addresses for your qeth device.
6. Modify the ***PORTNAME*** statement or leave it out if it is not necessary in your environment.
7. You may add any valid sysfs attribute and its value to the ***OPTIONS*** parameter. The Red Hat Enterprise Linux installer currently uses this to configure the layer mode (***layer2***) and the relative port number (***portno***) of qeth devices.

   The qeth device driver default for OSA devices is now layer 2 mode. To continue using old ifcfg definitions that rely on the previous default of layer 3 mode, add **layer2=0** to the ***OPTIONS*** parameter.

/etc/sysconfig/network-scripts/ifcfg-eth1

```
# IBM QETH
DEVICE=eth1
BOOTPROTO=static
IPADDR=192.168.70.87
NETMASK=255.255.255.0
ONBOOT=yes
NETTYPE=qeth
SUBCHANNELS=0.0.0600,0.0.0601,0.0.0602
PORTNAME=OSAPORT
OPTIONS='layer2=1 portno=0'
MACADDR=02:00:00:b3:84:ef
TYPE=Ethernet
```

Changes to an **ifcfg** file only become effective after rebooting the system or after the dynamic addition of new network device channels by changing the system's I/O configuration (for example, attaching under z/VM). Alternatively, you can trigger the activation of a **ifcfg** file for network channels which were previously not active yet, by executing the following commands:

1. Use the **cio_ignore** command 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
   ```

2. 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
```

3. Check the status of the network device:

```
# lsqeth
```

4. Now start the new interface:

```
# ifup eth1
```

5. Check the status of the interface:

```
# ifconfig eth1
eth1      Link encap:Ethernet  HWaddr 02:00:00:00:00:01
          inet addr:192.168.70.87  Bcast:192.168.70.255 Mask:255.255.255.0
          inet6 addr: fe80::ff:fe00:1/64 Scope:Link
          UP BROADCAST RUNNING NOARP MULTICAST  MTU:1492  Metric:1
          RX packets:23 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:644 (644.0 b)  TX bytes:264 (264.0 b)
```

6. Check the routing for the new interface:

```
# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref  Use Iface
192.168.70.0    *               255.255.255.0  U     0      0      0 eth1
10.1.20.0       *               255.255.255.0  U     0      0      0 eth0
default         10.1.20.1       0.0.0.0         UG    0      0      0 eth0
```

7. Verify your changes by using the ping command to ping the gateway or another host on the subnet of the new device:

```
# ping -c 1 192.168.70.8
PING 192.168.70.8 (192.168.70.8) 56(84) bytes of data.
64 bytes from 192.168.70.8: icmp_seq=0 ttl=63 time=8.07 ms
```

8. If the default route information has changed, you must also update **/etc/sysconfig/network** accordingly.

25.3.2. Adding an LCS Device

The *LAN channel station* (LCS) device driver supports 1000Base-T Ethernet on the OSA-Express2 and OSA-Express 3 features.

Based on the type of interface being added, the LCS driver assigns one base interface name:

- eth*n* for OSA-Express Fast Ethernet and Gigabit Ethernet

n is **0** for the first device of that type, **1** for the second, and so on.

25.3.2.1. Dynamically Adding an LCS Device

1. Load the device driver:

```
# modprobe lcs
```

2. Use the **cio_ignore** command 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
```

Replace *read_device_bus_id* and *write_device_bus_id* with the two device bus IDs representing a network device. For example:

```
# cio_ignore -r 0.0.09a0,0.0.09a1
```

3. Create the group device:

```
# echo read_device_bus_id,write_device_bus_id > /sys/bus/ccwgroup/drivers/lcs/group
```

4. Configure the device. OSA cards can provide up to 16 ports for a single CHPID. By default, the LCS group device uses port **0**. To use a different port, issue a command similar to the following:

```
# echo portno > /sys/bus/ccwgroup/drivers/lcs/device_bus_id/portno
```

Replace *portno* with the port number you want to use. For more information about configuration of the LCS driver, refer to the chapter on LCS in *Linux on System z Device Drivers, Features, and Commands on Red Hat Enterprise Linux 6*.

5. Set the device online:

```
# echo 1 > /sys/bus/ccwgroup/drivers/lcs/read_device_bus_id/online
```

6. To find out what network device name has been assigned, enter the command:

```
# ls -l /sys/bus/ccwgroup/drivers/lcs/read_device_bus_ID/net/
drwxr-xr-x 4 root root 0 2010-04-22 16:54 eth1
```

25.3.2.2. Persistently Adding an LCS Device

cio_ignore is handled transparently for persistent device configurations and you do not need to free devices from the ignore list manually.

To add an LCS device persistently, follow these steps:

1. Create a configuration script as file in **/etc/sysconfig/network-scripts/** with a name like **ifcfg-eth*n*** where *n* is an integer starting with **0**. The file should look similar to the following:

```
/etc/sysconfig/network-scripts/ifcfg-eth0
# IBM LCS
DEVICE=eth0
BOOTPROTO=static
IPADDR=10.12.20.136
NETMASK=255.255.255.0
ONBOOT=yes
NETTYPE=lcs
SUBCHANNELS=0.0.09a0,0.0.09a1
PORTNAME=0
OPTIONS=''
TYPE=Ethernet
```

2. Modify the value of ***PORTNAME*** to reflect the LCS port number (***portno***) you would like to use. You can add any valid lcs sysfs attribute and its value to the optional ***OPTIONS*** parameter. Refer to Section 25.3.1.3, "Persistently Adding a qeth Device" for the syntax.

3. Set the ***DEVICE*** parameter as follows:

```
DEVICE=ethn
```

4. Issue an **ifup** command to activate the device:

```
# ifup ethn
```

Changes to an **ifcfg** file only become effective after rebooting the system. You can trigger the activation of a **ifcfg** file for network channels by executing the following commands:

1. Use the **cio_ignore** command to remove the LCS device adapter from the list of ignored devices and make it visible to Linux:

```
# cio_ignore -r read_device_bus_id,write_device_bus_id
```

Replace *read_device_bus_id* and *write_device_bus_id* with the device bus IDs of the LCS device. For example:

```
# cio_ignore -r 0.0.09a0,0.0.09a1
```

2. 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/0.0.09a0/uevent
```

25.3.3. Mapping Subchannels and Network Device Names

The ***DEVICE=*** option in the **ifcfg** file does not determine the mapping of subchannels to network device names. Instead, the udev rules file **/etc/udev/rules.d/70-persistent-net.rules** determines which network device channel gets which network device name.

When configuring a new network device on System z, the system automatically adds a new rule to that file and assigns the next unused device name. You can then edit the values assigned to the ***NAME=*** variable for each device.

Example content of **/etc/udev/rules.d/70-persistent-net.rules**:

```
# This file was automatically generated by the /lib/udev/write_net_rules
# program run by the persistent-net-generator.rules rules file.
#
# You can modify it,as long as you keep each rule on a single line.
# S/390 qeth device at 0.0.f5f0
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="qeth", KERNELS=="0.0.f5f0",
ATTR{type}=="1", KERNEL=="eth*", NAME="eth0"
# S/390 ctcm device at 0.0.1000
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="ctcm", KERNELS=="0.0.1000",
ATTR{type}=="256", KERNEL=="ctc*", NAME="ctc0"
# S/390 qeth device at 0.0.8024
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="qeth", KERNELS=="0.0.8024",
ATTR{type}=="1", KERNEL=="hsi*", NAME="hsi0"
# S/390 qeth device at 0.0.8124
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="qeth", KERNELS=="0.0.8124",
ATTR{type}=="1", KERNEL=="hsi*", NAME="hsi1"
# S/390 qeth device at 0.0.1017
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="qeth", KERNELS=="0.0.1017",
ATTR{type}=="1", KERNEL=="eth*", NAME="eth3"
# S/390 qeth device at 0.0.8324
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="qeth", KERNELS=="0.0.8324",
ATTR{type}=="1", KERNEL=="hsi*", NAME="hsi3"
# S/390 qeth device at 0.0.8224
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="qeth", KERNELS=="0.0.8224",
ATTR{type}=="1", KERNEL=="hsi*", NAME="hsi2"
# S/390 qeth device at 0.0.1010
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="qeth", KERNELS=="0.0.1010",
ATTR{type}=="1", KERNEL=="eth*", NAME="eth2"
# S/390 lcs device at 0.0.1240
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="lcs", KERNELS=="0.0.1240",
ATTR{type}=="1", KERNEL=="eth*", NAME="eth1"
# S/390 qeth device at 0.0.1013
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="qeth", KERNELS=="0.0.1013",
ATTR{type}=="1", KERNEL=="hsi*", NAME="hsi4"
```

25.3.4. Configuring a System 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 (refer to Chapter 26, *Parameter and Configuration Files*) or part of a **zipl.conf** on a DASD or FCP-attached SCSI LUN prepared with the **zipl** boot loader. There is no need to recreate the initramfs.

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 System 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 System z

network hardware. The configuration of IP addresses and other network specifics works the same as for other platforms. Refer to the **dracut** documentation for more details.

cio_ignore for the network channels is handled transparently 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,!0.0.0009
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:eth0:non
e rd_NO_LUKS rd_NO_LVM rd_NO_MD rd_NO_DM LANG=en_US.UTF-8 SYSFONT=latarcyrheb-
sun16 KEYTABLE=us
```

CHAPTER 26. PARAMETER AND CONFIGURATION FILES

The IBM System z architecture can use a customized parameter file to pass boot parameters to the kernel and the installer. This section describes the contents of this parameter file.

You need only read this section if you intend to change the shipped parameter file. You need to change the parameter file if you want to:

- automate the user input for **linuxrc** or the loader (refer to Chapter 21, *Installation Phase 1: Configuring a Network Device* and Chapter 22, *Installation Phase 2: Configuring Language and Installation Source*).
- install unattended with kickstart.
- choose non-default installation settings that are not accessible through the installer"s interactive user interface, such as rescue mode.

The parameter file can be used to set up networking non-interactively before the installation program (loader and **anaconda**) starts.

The kernel parameter file is limited to 895 characters plus an end-of-line character. The parameter file can be variable or fixed record format. Fixed record format increases the file size by padding each line up to the record length. Should you encounter problems with the installer not recognizing all specified parameters in LPAR environments, you can try to put all parameters in one single line or start and end each line with a space character.

For more details on kernel parameters and different possibilities of specifying them, see the chapter on booting Linux and the chapter on kernel parameters in *Linux on System z Device Drivers, Features, and Commands on Red Hat Enterprise Linux 6*.

The parameter file contains kernel parameters, such as **root=/dev/ram0** or **ro**, and parameters for the installation process, such as **vncpassword=test** or **vnc**.

26.1. REQUIRED PARAMETERS

The following parameters are required and must be included in the parameter file. They are also provided in the file **generic.prm** in directory **images/** of the installation DVD:

root=file_system

where *file_system* represents the device on which the root file system can be found. For installation purposes, it must be set to **/dev/ram0**, which is the ramdisk containing the Red Hat Enterprise Linux installation program.

ro

mounts the root file system, which is a ramdisk, read-only.

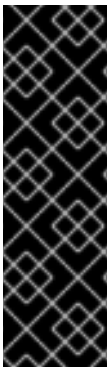
ip=off

disables automatic network configuration.

ramdisk_size=size

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

The file generic.prm also contains the additional parameter **cio_ignore=all,!0.0.0009**. This setting speeds up boot and device detection on systems with many devices. The installer transparently handles the activation of ignored devices.



IMPORTANT

To avoid installation problems arising from **cio_ignore** support not being implemented throughout the entire stack, adapt the ***cio_ignore=*** parameter value to your system or remove the parameter entirely from your parameter file used for booting (IPL) the installer.

When installing from an FCP-attached DVD drive, and you encounter a problem with ignored devices, select the menu option **clear blacklist** in **linuxrc** (refer to Chapter 21, *Installation Phase 1: Configuring a Network Device*) to remove the list of ignored devices.

26.2. THE Z/VM CONFIGURATION FILE

This applies only if installing under z/VM. 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 (refer to Section 26.3, "Installation Network Parameters").

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***

26.3. INSTALLATION NETWORK PARAMETERS

The following parameters can be used to set up the preliminary network automatically and can be defined in either the parameter file or the CMS configuration file. The parameters in this section 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 **lcs** for:

- OSA-2 Ethernet/Token Ring
- OSA-Express Fast Ethernet in non-QDIO mode
- OSA-Express High Speed Token Ring in non-QDIO mode
- Gigabit Ethernet in non-QDIO mode

Choose **qeth** for:

- OSA-Express Fast Ethernet
- Gigabit Ethernet (including 1000Base-T)
- High Speed Token Ring
- HiperSockets
- ATM (running Ethernet LAN emulation)

SUBCHANNELS="device_bus_IDs"

Where *bus_IDs* is a comma-separated list of two or three device bus IDs.

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"
```

PORTNAME="osa_portname" , PORTNAME="lcs_portnumber"

This variable supports OSA devices operating in qdio mode or in non-qdio mode.

When using qdio mode (**NETTYPE="qeth"**), *osa_portname* is the portname specified on the OSA device when operating in qeth mode.

When using non-qdio mode (**NETTYPE="lcs"**), *lcs_portnumber* is used to pass the relative port number as a decimal integer in the range of 0 through 15.

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) to the CMS configuration file to avoid being prompted for the mode.

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**. Note that 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 hostname 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. For a detailed description of the syntax, refer to the **dasd_mod** device driver module option described in the chapter on the DASD device driver in *Linux on System z Device Drivers, Features, and Commands on Red Hat Enterprise Linux 6*.

Linuxrc supports a comma-separated list of device bus IDs or of 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 (for example **/dev/disk/by-path/...**) to enable transparent addition of disks later. Other global options such as **probeonly**, **nopav**, or **nofcx** are not supported by linuxrc.

Only specify those DASDs that you really need to install your system. All unformatted DASDs specified here must be formatted after a confirmation later on in the installer (refer to Section 23.6.1.1, “DASD low-level formatting”). Add any data DASDs that are not needed for the root file system or the **/boot** partition after installation as described in Section 25.1.3, “DASDs Which Are Not Part of the Root File System”.

For FCP-only environments, specify **DASD="none"**.

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"

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**).

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. There is no interactive question for FCP in linuxrc. An example value may look similar to the

following:

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

IMPORTANT

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.

The installation program prompts you for any required parameters not specified in the parameter or configuration file except for FCP_n.

26.4. VNC AND X11 PARAMETERS

The following parameters can be defined in a parameter file but do not work in a CMS configuration file. With these parameters you control what interface will be used for **anaconda**.

To use an X11 user interface without X11 forwarding, specify the following X11 parameter:

display=IP/hostname:display

Sets the hostname or IP address and the X11 display where the installer should connect to and display its graphical user interface.

To use a VNC server instead of an X11 user interface, specify the following VNC parameters:

vnc

Specify ***vnc*** to use the VNC graphical user interface later in the installation process.

vncpassword=

This parameter sets the password used to connect to the VNC server. The password parameter is optional. If not used, the VNC server does not use a password and anybody can connect to the VNC server.

vncconnect=IP/hostname[:port]

When used in addition to ***vnc*** and ***vncpassword=***, this optional parameter specifies the hostname or IP address (and optionally, a TCP port) where a VNC client is running in listening mode. The installer connects to and displays its graphical user interface on this VNC client.

26.5. LOADER PARAMETERS

The following parameters can be defined in a parameter file but do not work in a CMS configuration file.

To automate the loader screens, specify the following parameters:

lang=language

Sets the language of the installer user interface, for example, **en** for English or **de** for German. This automates the response to **Choose a Language** (refer to Section 22.3, “Language Selection”).

repo=installation_source

Sets the installation source to access stage 2 as well as the repository with the packages to be installed. This automates the response to **Installation Method** (refer to Section 22.4, "Installation Method").

26.6. PARAMETERS FOR KICKSTART INSTALLATIONS

The following parameters can be defined in a parameter file but do not work in a CMS configuration file.

ks=URL

References a kickstart file, which usually resides on the network for Linux installations on System z. Replace *URL* with the full path including the file name of the kickstart file. This parameter activates automatic installation with kickstart. Refer to Section 28.4, "Automating the Installation with Kickstart" and Section 32.11, "Starting a Kickstart Installation" for more details.]

RUNKS=value

Where *value* is defined as *1* if you want to run the loader automatically on the Linux console without having to log in over the network with SSH. To use **RUNKS=1**, the console must either support full-screen or the ***cmdline*** option below should be used. The latter applies for the 3270 terminal under z/VM or the operating system messages console for LPAR. We recommend **RUNKS=1** for fully automatic installations with kickstart. When **RUNKS=1** is set, **linuxrc** automatically continues in case of parameter errors and does not interrupt unattended installations by prompting for user interaction.

Leave out the parameter or specify **RUNKS=0** otherwise.

cmdline

When ***cmdline*** is specified, output on line-mode terminals (such as 3270 under z/VM or operating system messages for LPAR) becomes readable, as the installer disables escape terminal sequences that are only applicable to UNIX-like consoles. This requires installation with a kickstart file that answers all questions, since the installer does not support interactive user input in cmdline mode.

Ensure that your kickstart file contains all required parameters before you use either the ***RUNKS*** or ***cmdline*** options. Refer to Chapter 32, *Kickstart Installations* for details.

26.7. MISCELLANEOUS PARAMETERS

The following parameters can be defined in a parameter file but do not work in a CMS configuration file.

askmethod

Do not use an automatically detected DVD as installation source but ask for the installation method to manually specify the installation source. This parameter is useful if you booted from an FCP-attached DVD but want to continue with another installation source, for example on the network or on a local hard disk.

mediacheck

Turns on testing of an ISO-based installation source; for example, when booted from an FCP-attached DVD or using ***repo=*** with an ISO on local hard disk or mounted with NFS.

nompath

Disables support for multipathing devices.

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

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

rescue

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

stage2=URL

Specifies a path to an **install.img** file instead of to an installation source. Otherwise, follows the same syntax as ***repo=***. If ***stage2*** is specified, it typically takes precedence over other methods of finding **install.img**. However, if **anaconda** finds **install.img** on local media, the ***stage2*** URL will be ignored.

If ***stage2*** is not specified and **install.img** cannot be found locally, **anaconda** looks to the location given by ***repo=*** or ***method=***.

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

syslog=IP/hostname[:port]

Makes the installer send log messages to a remote syslog server.

The boot parameters described here are the most useful for installations and trouble shooting on System z, but only a subset of those that influence the installer. Refer to Chapter 28, *Boot Options* for a more complete list of installer boot parameters.

26.8. SAMPLE PARAMETER FILE AND CMS CONFIGURATION FILE

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

Example of **generic.prm** file:

```
root="/dev/ram0" ro ip="off" ramdisk_size="40000" cio_ignore="all,!0.0.0009"
CMSDASD="191" CMSCONFFILE="redhat.conf"
vnc
```

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 27. IBM SYSTEM Z REFERENCES

27.1. IBM SYSTEM Z PUBLICATIONS

Current versions of the Linux on System z publications can be found at http://www.ibm.com/developerworks/linux/linux390/documentation_red_hat.html. They include:

Linux on System z – Device Drivers, Features, and Commands as available with Red Hat Enterprise Linux 6. IBM . 2010. SC34-2597.

Linux on System z – Using the Dump Tools on Red Hat Enterprise Linux 6 IBM . 2010. SC34-2607.

Linux on System z – How to use FC-attached SCSI devices with Linux on System z9 and zSeries IBM . 2008. SC33-8413.

How to use Execute-in-Place Technology with Linux on z/VM IBM . 2008. SC34-2594.

Linux on System z – How to Set up a Terminal Server Environment on z/VM IBM . 2009. SC34-2596.

Linux on System z – libica 2.0 Programmer's Reference IBM . 2009. SC34-2602.

Linux on System z – How to Improve Performance with PAV IBM . 2008. SC33-8414.

z/VM – Getting Started with Linux on System z IBM . 2009. SC24-6194.

27.2. IBM REDBOOKS PUBLICATIONS FOR SYSTEM Z

Current versions of IBM Redbooks publications can be found at http://www.redbooks.ibm.com/. They include:

Introductory publications

Introduction to the New Mainframe: z/VM Basics IBM Redbooks . 2007. SG24-7316.

z/VM and Linux on IBM System z The Virtualization Cookbook for Red Hat Enterprise Linux 5.2 IBM Redbooks . 2008. SG24-7492.

Practical Migration to Linux on System z. IBM Redbooks . 2009. SG24-7727.

Performance and high availability

Linux on IBM System z: Performance Measurement and Tuning IBM Redbooks . 2011. SG24-6926.

Achieving High Availability on Linux for System z with Linux-HA Release 2 IBM Redbooks . 2009. SG24-7711.

Security

Security for Linux on System z. IBM Redbooks . 2013. SG24-7728.

Using Cryptographic Adapters for Web Servers with Linux on IBM System z9 and zSeries IBM Redbooks . 2006. REDP-4131.

Networking

IBM System z Connectivity Handbook. IBM Redbooks . 2013. SG24-5444.

OSA Express Implementation Guide. IBM Redbooks . 2009. SG24-5948.

HiperSockets Implementation Guide. IBM Redbooks . 2007. SG24-6816.

Fibre Channel Protocol for Linux and z/VM on IBM System z IBM Redbooks . 2007. SG24-7266.

27.3. ONLINE RESOURCES

For z/VM publications, refer to http://www.vm.ibm.com/library/ .

For System z I/O connectivity information, refer to
http://www.ibm.com/systems/z/hardware/connectivity/index.html .

For System z cryptographic coprocessor information, refer to
http://www.ibm.com/security/cryptocards/ .

Sharing and maintaining RHEL 5.3 Linux under z/VM Brad Hinson and Mike MacIsaac.
http://www.linuxvm.org/Present/misc/ro-root-RH5.pdf .

PART IV. ADVANCED INSTALLATION OPTIONS

This part of the *Red Hat Enterprise Linux Installation Guide* covers more advanced or uncommon methods of installing Red Hat Enterprise Linux, including:

- boot options.
- installing without media.
- installing through VNC.
- using **kickstart** to automate the installation process.

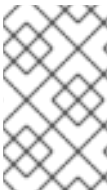
CHAPTER 28. BOOT OPTIONS

The Red Hat Enterprise Linux installation system includes a range of functions and options for administrators. To use boot options, enter **linux *option*** at the **boot:** prompt.

To access the **boot:** prompt on a system that displays a graphical boot screen, press the **Esc** key while the graphical boot screen is displayed.

If you specify more than one option, separate each of the options by a single space. For example:

```
linux option1 option2 option3
```



NOTE

The Red Hat Enterprise Linux installation and *rescue discs* may either boot with *rescue mode*, or load the installation system. For more information on rescue discs and rescue mode, refer to Section 28.6.2, “Booting Your Computer with the Rescue Mode” .

28.1. CONFIGURING THE INSTALLATION SYSTEM AT THE BOOT MENU

You can use the boot menu to specify a number of settings for the installation system, including:

- language
- display resolution
- interface type
- Installation method
- network settings

28.1.1. Specifying the Language

To set the language for both the installation process and the final system, specify the ISO code for that language with the **lang** option. Use the **keymap** option to configure the correct keyboard layout.

For example, the ISO codes **el_GR** and **gr** identify the Greek language and the Greek keyboard layout:

```
linux lang=el_GR keymap=gr
```

28.1.2. Configuring the Interface

To use a specific display resolution, enter **resolution=*setting*** as a boot option. For example, to set the display resolution to 1024×768, enter:

```
linux resolution=1024x768
```

To run the installation process in **text** mode, enter:

```
linux text
```

To enable support for a serial console, enter **serial** as an additional option.

Use **display=*ip*:0** to allow remote display forwarding. In this command, *ip* should be replaced with the IP address of the system on which you want the display to appear.

On the system you want the display to appear on, you must execute the command **xhost +*remotehostname***, where *remotehostname* is the name of the host from which you are running the original display. Using the command **xhost +*remotehostname*** limits access to the remote display terminal and does not allow access from anyone or any system not specifically authorized for remote access.

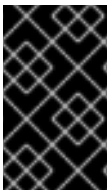
28.1.3. Updating anaconda

You can install Red Hat Enterprise Linux with a newer version of the **anaconda** installation program than the one supplied on your installation media.

The boot option

```
linux updates
```

presents you with a prompt that asks you for a disk image containing **anaconda** updates. You do not need to specify this option if you are performing a network installation and have already placed the updates image contents in **rhupdates**/ on the server.



IMPORTANT

The **rhupdates** directory should only contain **anaconda** updates. The installation may fail if you add other files (such as errata RPMs) or if you place too much content in the directory.

To load the **anaconda** updates from a network location instead, use:

```
linux updates=
```

followed by the URL for the location where the updates are stored.

28.1.4. Specifying the Installation Method

Use the **askmethod** option to display additional menus that enable you to specify the installation method and network settings. You may also configure the installation method and network settings at the **boot:** prompt itself.

To specify the installation method from the **boot:** prompt, use the **repo** option. Refer to Table 28.1, “Installation methods” for the supported installation methods.

Table 28.1. Installation methods

Installation method	Option format
DVD drive	**repo=cdrom:*device***
Hard Drive	**repo=hd:*device*/*path***

Installation method	Option format
HTTP Server	**repo=http://*host*/*path***
HTTPS Server	**repo=https://*host*/*path***
FTP Server	**repo=ftp://*username*:*password*@*host*/*path***
NFS Server	**repo=nfs:*server*:/*path***
ISO images on an NFS Server	**repo=nfsiso:*server*:/*path***

28.1.5. Specifying the Network Settings

Normally, **anaconda** prompts you to configure a network interface if one is needed during installation. However, you can provide network settings with options at the **boot:** prompt as follows:

ip

The system's IP address.

netmask

The system's netmask.

gateway

The IP address of the network gateway.

dns

The IP address of the DNS server.

ksdevice

The network device to use with these settings.

ifname

The name you wish to assign to the network device, followed by the device's MAC address.

Each of these settings is required even if you are only configuring a single interface.

The following settings are optional:

vlanid

The virtual LAN ID number (802.1q tag) for the specified network device.

nicdelay

The delay after which the network will be considered active. If you use this option, the system will wait after bringing up network interfaces until either the gateway is successfully pinged, or until the amount of seconds specified in this parameter passes. This is useful for some NICs which may report

that a link is available before it actually is, causing any operations which require network access (such as Kickstart file downloads) to fail. Maximum value of this parameter is 30 as defined by **NetworkManager**; specifying a value higher than 30 will cause the option to be ignored.

This example configures the network settings for an installation system that uses the IP address **192.168.1.10** for interface **eth0**. The interface is named **primary**, and the system will wait for 5 seconds or until it can successfully ping the gateway before continuing:

```
linux ip=192.168.1.10 netmask=255.255.255.0 gateway=192.168.1.1 dns=192.168.1.3 ksdevice=eth0
ifname=primary:01:23:45:67:89:ab nicdelay=5
```

If you specify the network configuration and network device at the **boot:** prompt, these settings are used for the installation process and the **Networking Devices** and **Configure TCP/IP** dialogs do not appear.

28.1.5.1. Configuring a Bonded Interface

To configure a bonded network interface, use the **bond** option. Name the bonded interface, specify which network connections will be bonded, and list any additional options in the following format:

```
linux bond=<bondname>:<bondslaves>:[:<options>]
```

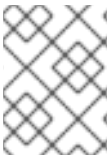
For example:

```
linux bond=bond0:eth0,eth1:mode=active-backup,primary=eth1
```

Available optional parameters are listed in the *Working with Kernel Modules* chapter of the Red Hat Enterprise Linux Deployment Guide.

28.2. ENABLING REMOTE ACCESS TO THE INSTALLATION SYSTEM

You may access either graphical or text interfaces for the installation system from any other system. Access to a text mode display requires **telnet**, which is installed by default on Red Hat Enterprise Linux systems. To remotely access the graphical display of an installation system, use client software that supports the VNC (Virtual Network Computing) display protocol.



NOTE

Red Hat Enterprise Linux includes the VNC client **vncviewer**. To obtain **vncviewer**, install the tigervnc package.

The installation system supports two methods of establishing a VNC connection. You may start the installation, and manually login to the graphical display with a VNC client on another system. Alternatively, you may configure the installation system to automatically connect to a VNC client on the network that is running in *listening mode*.

28.2.1. Enabling Remote Access with VNC

To enable remote graphical access to the installation system, enter two options at the prompt:

```
linux vnc vncpassword=qwerty
```

The **vnc** option enables the VNC service. The **vncpassword** option sets a password for remote access. The example shown above sets the password as **qwerty**.

NOTE

The VNC password must be at least six characters long.

Specify the language, keyboard layout and network settings for the installation system with the screens that follow. You may then access the graphical interface through a VNC client. The installation system displays the correct connection setting for the VNC client:

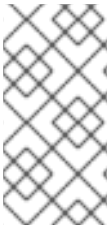
```
Starting VNC...
The VNC server is now running.
Please connect to computer.mydomain.com:1 to begin the install...
Starting graphical installation...
Press <enter> for a shell
```

You may then login to the installation system with a VNC client. To run the **vncviewer** client on Red Hat Enterprise Linux, choose **Applications → Accessories → VNC Viewer**, or type the command **vncviewer** in a terminal window. Enter the server and display number in the **VNC Server** dialog. For the example above, the **VNC Server** is **computer.mydomain.com:1**.

28.2.2. Connecting the Installation System to a VNC Listener

To have the installation system automatically connect to a VNC client, first start the client in listening mode. On Red Hat Enterprise Linux systems, use the **-listen** option to run **vncviewer** as a listener. In a terminal window, enter the command:

```
vncviewer -listen
```



NOTE

By default, **vncviewer** uses TCP port 5500 when in listening mode. The firewall must be configured to permit connections to this port from other systems. Choose **System → Administration → Firewall**. Select **Other ports**, and **Add**. Enter **5500** in the **Port(s)** field, and specify **tcp** as the **Protocol**.

Once the listening client is active, start the installation system and set the VNC options at the **boot:** prompt. In addition to **vnc** and **vncpassword** options, use the **vncconnect** option to specify the name or IP address of the system that has the listening client. To specify the TCP port for the listener, add a colon and the port number to the name of the system.

For example, to connect to a VNC client on the system **desktop.mydomain.com** on the port 5500, enter the following at the **boot:** prompt:

```
linux vnc vncpassword=qwerty vncconnect=desktop.mydomain.com:5500
```

28.2.3. Enabling Remote Access with ssh

To enable remote access to a text mode installation, use the **sshd=1** option at the **boot:** prompt:

```
linux sshd=1
```

You can then connect to the installation system with the **ssh** utility. The **ssh** command requires the name or IP address of the installation system, and a password if you specified one (for example, in a kickstart file).

28.2.4. Enabling Remote Access with Telnet

To enable remote access to a text mode installation, use the **telnet** option at the **boot:** prompt:

```
linux text telnet
```

You may then connect to the installation system with the **telnet** utility. The **telnet** command requires the name or IP address of the installation system:

```
telnet computer.mydomain.com
```

WARNING

To ensure the security of the installation process, only use the **telnet** option to install systems on networks with restricted access.

28.3. LOGGING TO A REMOTE SYSTEM DURING THE INSTALLATION

By default, the installation process sends log messages to the console as they are generated. You may specify that these messages go to a remote system that runs a *syslog* service.

To configure remote logging, add the **syslog** option. Specify the IP address of the logging system, and the UDP port number of the log service on that system. By default, syslog services that accept remote messages listen on UDP port 514.

For example, to connect to a syslog service on the system **192.168.1.20**, enter the following at the **boot:** prompt:

```
linux syslog=192.168.1.20:514
```

28.3.1. Configuring a Log Server

Red Hat Enterprise Linux uses **rsyslog** to provide a syslog service. The default configuration of **rsyslog** rejects messages from remote systems.

WARNING

Only enable remote syslog access on secured networks. The **rsyslog** configuration detailed below does not make use of any of the security measures available in **rsyslog** Crackers may slow or crash systems that permit access to the logging service, by sending large quantities of false log messages. In addition, hostile users may intercept or falsify messages sent to the logging service over the network.

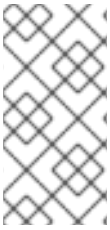
To configure a Red Hat Enterprise Linux system to accept log messages from other systems on the network, edit the file **/etc/rsyslog.conf**. You must use **root** privileges to edit the file **/etc/rsyslog.conf**. Uncomment the following lines by removing the hash preceding them:

```
$ModLoad imudp.so
 $UDPServerRun 514
```

Restart the **rsyslog** service to apply the change:

```
su -c '/sbin/service rsyslog restart'
```

Enter the **root** password when prompted.



NOTE

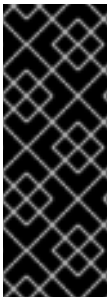
By default, the syslog service listens on UDP port 514. The firewall must be configured to permit connections to this port from other systems. Choose **System → Administration → Firewall**. Select **Other ports**, and **Add**. Enter **514** in the **Port(s)** field, and specify **udp** as the **Protocol**.

28.4. AUTOMATING THE INSTALLATION WITH KICKSTART

You can allow an installation to run unattended by using Kickstart. A *Kickstart* file specifies settings for an installation. Once the installation system boots, it can read a Kickstart file and carry out the installation process without any further input from a user.

NOTE

The Red Hat Enterprise Linux installation process automatically writes a Kickstart file that contains the settings for the installed system. This file is always saved as **/root/anaconda-ks.cfg**. You may use this file to repeat the installation with identical settings, or modify copies to specify settings for other systems.



IMPORTANT

Firstboot does not run after a system is installed from a Kickstart file unless a desktop and the X Window System were included in the installation and graphical login was enabled. Either specify a user with the **user** option in the Kickstart file before installing additional systems from it (refer to Section 32.4, "Kickstart Options" for details) or log into the installed system with a virtual console as root and add users with the **adduser** command.

Red Hat Enterprise Linux includes a graphical application to create and modify Kickstart files by selecting the options that you require. Use the package **system-config-kickstart** to install this utility. To load the Red Hat Enterprise Linux Kickstart editor, choose **Applications → System Tools → Kickstart**.

Kickstart files list installation settings in plain text, with one option per line. This format lets you modify your Kickstart files with any text editor, and write scripts or applications that generate custom Kickstart files for your systems.

To automate the installation process with a Kickstart file, use the **ks** option to specify the name and location of the file:

```
linux ks=location/kickstart-file.cfg
```

You may use Kickstart files that are held on either removable storage, a hard drive, or a network server. Refer to Table 28.2, "Kickstart sources" for the supported Kickstart sources.

Table 28.2. Kickstart sources

Kickstart source	Option format
DVD drive	**ks=*cdrom:/directory/ks.cfg***
Hard Drive	**ks=*hd:/device/directory/ks.cfg***
Other Device	**ks=*file:/device/directory/ks.cfg***
HTTP Server	**ks=*http://server.mydomain.com/directory/ks.cfg***
HTTPS Server	**ks=*https://server.mydomain.com/directory/ks.cfg***
FTP Server	**ks=*ftp://server.mydomain.com/directory/ks.cfg***
NFS Server	**ks=*nfs:server.mydomain.com:/directory/ks.cfg***

IMPORTANT

You can use a device name such as /**dev**/**sdb** to identify a hard drive or a USB drive containing a Kickstart file. However, there is no guarantee that the device identifier will remain the same on multiple systems. Therefore, the recommended method for specifying a hard drive or a USB drive in Kickstart installations is by UUID. For example:

```
ks=hd:UUID=ede47e6c-8b5f-49ad-9509-774fa7119281:ks.cfg
```

You can determine a device's UUID by using the **blkid** command as **root**:

```
# blkid /dev/sdb1
/dev/sdb1: UUID="2c3a072a-3d0c-4f3a-a4a1-ab5f24f59266" TYPE="ext4"
```

To obtain a Kickstart file from a script or application on a Web server, specify the URL of the application with the **ks=** option. If you add the option **kssendmac**, the request also sends HTTP headers to the Web application. Your application can use these headers to identify the computer. This line sends a request with headers to the application *http://server.mydomain.com/kickstart.cgi*:

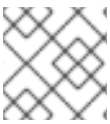
```
linux ks=http://server.mydomain.com/kickstart.cgi kssendmac
```

28.5. ENHANCING HARDWARE SUPPORT

By default, Red Hat Enterprise Linux attempts to automatically detect and configure support for all of the components of your computer. Red Hat Enterprise Linux supports the majority of hardware in common use with the software *drivers* that are included with the operating system. To support other devices you may supply additional drivers during the installation process, or at a later time.

28.5.1. Overriding Automatic Hardware Detection

For some models of device automatic hardware configuration may fail, or cause instability. In these cases, you may need to disable automatic configuration for that type of device, and take additional steps to manually configure the device after the installation process is complete.



NOTE

Refer to the Release Notes for information on known issues with specific devices.

To override the automatic hardware detection, use one or more of the following options:

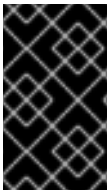
Table 28.3. Hardware Options

Compatibility	Option
Disable all hardware detection	**noprobe**
Disable graphics, keyboard, and mouse detection	**headless**
Disable passing keyboard and mouse information to stage 2 of the installation program	**nopass**

Compatibility	Option
Use basic VESA driver for video	**xdriver=vesa**
Disable shell access on virtual console 2 during installation	**noshell**
Disable advanced configuration and power interface (ACPI)	**acpi=off**
Disable machine check exception (MCE) CPU self-diagnosis.	**nomce**
Disable non-uniform memory access on the AMD64 architecture	**numa-off**
Force kernel to detect a specific amount of memory, where *xxx* is a value in megabytes	**mem=*xxx*m**
Enable DMA only for IDE and SATA drives	**libata.dma=1**
Disable BIOS-assisted RAID	**nodmraid**
Disable Firewire device detection	**nofirewire**
Disable parallel port detection	**noparport**
Disable PC Card (PCMCIA) device detection	**nopcmcia**
Disable all probing of network hardware	**nonet**

NOTE

The **isa** option causes the system to display an additional text screen at the beginning of the installation process. Use this screen to configure the ISA devices on your computer.



IMPORTANT

Other kernel boot options have no particular meaning for **anaconda** and do not affect the installation process. However, if you use these options to boot the installation system, **anaconda** will preserve them in the bootloader configuration.

28.6. USING THE MAINTENANCE BOOT MODES

28.6.1. Verifying Boot Media

You can test the integrity of an ISO-based installation source before using it to install Red Hat Enterprise Linux. These sources include DVD, and ISO images stored on a hard drive or NFS server. Verifying that the ISO images are intact before you attempt an installation helps to avoid problems that are often encountered during installation.

Red Hat Enterprise Linux offers you two ways to test installation ISOs:

- select **OK** at the prompt to test the media before installation when booting from the Red Hat Enterprise Linux DVD
- boot Red Hat Enterprise Linux with the option **mediacheck** option.

28.6.2. Booting Your Computer with the Rescue Mode

You may boot a command-line Linux system from either a rescue disc or an installation disc, without installing Red Hat Enterprise Linux on the computer. This enables you to use the utilities and functions of a running Linux system to modify or repair systems that are already installed on your computer.

The rescue disc starts the rescue mode system by default. To load the rescue system with the installation disc, choose **Rescue installed system** from the boot menu.

Specify the language, keyboard layout and network settings for the rescue system with the screens that follow. The final setup screen configures access to the existing system on your computer.

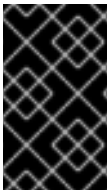
By default, rescue mode attaches an existing operating system to the rescue system under the directory **/mnt/sysimage/**.

28.6.3. Upgrading Your Computer

A previous boot option, **upgrade**, has been superceded by a stage in the installation process where the installation program prompts you to upgrade or reinstall earlier versions of Red Hat Enterprise Linux that it detects on your system.

However, the installation program may not correctly detect a previous version of Red Hat Enterprise Linux if the contents of the **/etc/redhat-release** file have changed. The boot option **upgradeany** relaxes the test that the installation program performs and allows you to upgrade a Red Hat Enterprise Linux installation that the installation program has not correctly identified.

CHAPTER 29. INSTALLING WITHOUT MEDIA



IMPORTANT

This procedure assumes you are already using Red Hat Enterprise Linux or another relatively modern Linux distribution, and the **GRUB** boot loader. It also assumes you are a somewhat experienced Linux user.

This section discusses how to install Red Hat Enterprise Linux on your system without making any additional physical media. Instead, you can use your existing **GRUB** boot loader to start the installation program.

29.1. RETRIEVING BOOT FILES

To perform an installation without media or a PXE server, your system must have two files stored locally, a kernel and an initial RAM disk.

Copy the **vmlinuz** and **initrd.img** files from a Red Hat Enterprise Linux DVD (or DVD image) to the **/boot/** directory, renaming them to **vmlinuz-install** and **initrd.img-install**. You must have **root** privileges to write files into the **/boot/** directory.

29.2. EDITING THE GRUB CONFIGURATION

The **GRUB** boot loader uses the configuration file **/boot/grub/grub.conf**. To configure **GRUB** to boot from the new files, add a boot stanza to **/boot/grub/grub.conf** that refers to them.

A minimal boot stanza looks like the following listing:

```
title Installation
        root (hd0,0)
        kernel /vmlinuz-install
        initrd /initrd.img-install
```

You may wish to add options to the end of the **kernel** line of the boot stanza. These options set preliminary options in **Anaconda** which the user normally sets interactively. For a list of available installer boot options, refer to Chapter 28, *Boot Options*.

The following options are generally useful for medialess installations:

- **ip=**
- **repo=**
- **lang=**
- **keymap=**
- **ksdevice=** (if installation requires an interface other than eth0)
- **vnc** and **vncpassword=** for a remote installation

When you are finished, change the **default** option in **/boot/grub/grub.conf** to point to the new first stanza you added:

```
default 0
```

29.3. BOOTING TO INSTALLATION

Reboot the system. **GRUB** boots the installation kernel and RAM disk, including any options you set. You may now refer to the appropriate chapter in this guide for the next step. If you chose to install remotely using VNC, refer to Section 28.2, "Enabling Remote Access to the Installation System" for assistance in connecting to the remote system.

CHAPTER 30. SETTING UP AN INSTALLATION SERVER

The following steps must be performed to prepare for a network installation:

1. Configure the network (NFS, FTP, HTTP, HTTPS) server to export the installation tree.
2. Configure the files on the **tftp** server necessary for network booting.
3. Configure which hosts are allowed to boot from the network configuration.
4. Start the **tftp** service.
5. Configure DHCP.
6. Boot the client, and start the installation.

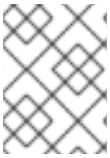
30.1. SETTING UP THE NETWORK SERVER

First, configure an NFS, FTP, HTTP, or HTTPS server to export the entire installation tree for the version and variant of Red Hat Enterprise Linux to be installed. Refer to Section 4.1, "Preparing for a Network Installation" for detailed instructions.

30.2. NETWORK BOOT CONFIGURATION

The next step is to copy the files necessary to start the installation to the **tftp** server so they can be found when the client requests them. The **tftp** server is usually the same server as the network server exporting the installation tree.

The PXE boot configuration procedure differs for BIOS and EFI. A separate **yaboot** configuration procedure is provided for Power Systems servers.



NOTE

Red Hat Satellite has the ability to automate the setup of a PXE server. See the Red Hat Satellite User Guide for more information.

30.2.1. Configuring PXE Boot for BIOS

1. If **tftp-server** is not yet installed, run **yum install tftp-server**.
2. In the **tftp-server** config file at **/etc/xinetd.d/tftp**, change the ***disabled*** parameter from **yes** to **no**.
3. Configure your DHCP server to use the boot images packaged with SYSLINUX. (If you do not have a DHCP server installed, refer to the *DHCP Servers* chapter in the Red Hat Enterprise Linux Deployment Guide.)

   A sample configuration in **/etc/dhcp/dhcpd.conf** might look like:

```
option space pxelinux;
option pxelinux.magic code 208 = string;
option pxelinux.configfile code 209 = text;
option pxelinux.pathprefix code 210 = text;
option pxelinux.reboottime code 211 = unsigned integer 32;
```

```
subnet 10.0.0.0 netmask 255.255.255.0 {
        option routers 10.0.0.254;
        range 10.0.0.2 10.0.0.253;

        class "pxeclients" {
                match if substring (option vendor-class-identifier, 0, 9) = "PXEClient";
                next-server 10.0.0.1;

                if option arch = 00:06 {
                        filename "pxelinux/bootia32.efi";
                } else if option arch = 00:07 {
                        filename "pxelinux/bootx64.efi";
                } else {
                        filename "pxelinux/pxelinux.0";
                }
        }

        host example-ia32 {
                hardware ethernet XX:YY:ZZ:11:22:33;
                fixed-address 10.0.0.2;
        }
}
```

4. You now need the **pxelinux.0** file from the syslinux-nolinux package in the ISO image file. To access it, run the following commands as root:

```
mount -t iso9660 /path_to_image/name_of_image.iso /mount_point -o loop,ro
cp -pr /mount_point/Packages/syslinux-nolinux-version-architecture.rpm
/publicly_available_directory
umount /mount_point
```

   Extract the package:

```
rpm2cpio syslinux-nolinux-version-architecture.rpm | cpio -dimv
```

5. Create a **pxelinux** directory within **tftpboot** and copy **pxelinux.0** into it:

```
mkdir /var/lib/tftpboot/pxelinux
cp publicly_available_directory/usr/share/syslinux/pxelinux.0 /var/lib/tftpboot/pxelinux
```

6. Create a **pxelinux.cfg** directory within **pxelinux**:

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

7. Add a config file to this directory. The file should either be named **default** or named after the IP address, converted into hexadecimal format without delimiters. For example, if your machine's IP address is 10.0.0.1, the filename would be **0A000001**.

   A sample config file at **/var/lib/tftpboot/pxelinux/pxelinux.cfg/default** might look like:

```
default vesamenu.c32
prompt 1
timeout 600
```

```
display boot.msg

label linux
  menu label ^Install or upgrade an existing system
  menu default
  kernel vmlinuz
  append initrd=initrd.img
label vesa
  menu label Install system with ^basic video driver
  kernel vmlinuz
  append initrd=initrd.img xdriver=vesa nomodeset
label rescue
  menu label ^Rescue installed system
  kernel vmlinuz
  append initrd=initrd.img rescue
label local
  menu label Boot from ^local drive
  localboot 0xffff
label memtest86
  menu label ^Memory test
  kernel memtest
  append -
```

For instructions on how to specify the installation source, refer to Section 7.1.3, "Additional Boot Options"

8. Copy the splash image into your **tftp** root directory:

```
cp /boot/grub/splash.xpm.gz /var/lib/tftpboot/pxelinux/splash.xpm.gz
```

9. Copy the boot images into your **tftp** root directory:

```
cp /path/to/x86_64/os/images/pxeboot/{vmlinuz,initrd.img} /var/lib/tftpboot/pxelinux/rhel6/
```

10. Boot the client system, and select the network device as your boot device when prompted.

30.2.2. Configuring PXE Boot for EFI

1. If **tftp-server** is not yet installed, run **yum install tftp-server**.

2. In the **tftp-server** config file at **/etc/xinetd.d/tftp**, change the ***disable*** parameter from **yes** to **no**.

3. Create a directory within **tftpboot** for the EFI boot images, and then copy them from your boot directory. In these examples we will name the subdirectory **pxelinux**, but any other name could be used.

```
mkdir /var/lib/tftpboot/pxelinux
cp /boot/efi/EFI/redhat/grub.efi /var/lib/tftpboot/pxelinux/bootx64.efi
```

4. Configure your DHCP server to use the EFI boot images packaged with GRUB. (If you do not have a DHCP server installed, refer to the *DHCP Servers* chapter in the Red Hat Enterprise Linux Deployment Guide.)

A sample configuration in **/etc/dhcp/dhcpd.conf** might look like:

```
option space PXE;
option PXE.mtftp-ip    code 1 = ip-address;
option PXE.mtftp-cport code 2 = unsigned integer 16;
option PXE.mtftp-sport code 3 = unsigned integer 16;
option PXE.mtftp-tmout code 4 = unsigned integer 8;
option PXE.mtftp-delay code 5 = unsigned integer 8;
option arch code 93 = unsigned integer 16; # RFC4578

subnet 10.0.0.0 netmask 255.255.255.0 {
        option routers 10.0.0.254;
        range 10.0.0.2 10.0.0.253;

        class "pxeclients" {
                match if substring (option vendor-class-identifier, 0, 9) = "PXEClient";
                next-server 10.0.0.1;

                if option arch = 00:06 {
                        filename "pxelinux/bootia32.efi";
                } else if option arch = 00:07 {
                        filename "pxelinux/bootx64.efi";
                } else {
                        filename "pxelinux/pxelinux.0";
                }
        }

        host example-ia32 {
                hardware ethernet XX:YY:ZZ:11:22:33;
                fixed-address 10.0.0.2;
        }
}
```

5. Add a config file to **/var/lib/tftpboot/pxelinux**. The file should either be named **efidefault** or named after the IP address, converted into hexadecimal format without delimiters. For example, if your machine's IP address is 10.0.0.1, the filename would be **0A000001**.

   A sample config file at **/var/lib/tftpboot/pxelinux/efidefault** might look like:

   ```
   default=0
   timeout=1
   splashimage=(nd)/splash.xpm.gz
   hiddenmenu
   title RHEL
           root (nd)
           kernel /rhel6/vmlinuz
           initrd /rhel6/initrd.img
   ```

   For instructions on how to specify the installation source, refer to Section 7.1.3, “Additional Boot Options”

6. Copy the splash image into your **tftp** root directory:

   ```
   cp /boot/grub/splash.xpm.gz /var/lib/tftpboot/pxelinux/splash.xpm.gz
   ```

7. Copy the boot images into your **tftp** root directory:

```
cp /path/to/x86_64/os/images/pxeboot/{vmlinuz,initrd.img} /var/lib/tftpboot/pxelinux/rhel6/
```

8. Boot the client system, and select the network device as your boot device when prompted.

30.2.3. Configuring for Power Systems Servers

1. If **tftp-server** is not yet installed, run **yum install tftp-server**.
2. In the **tftp-server** config file at **/etc/xinetd.d/tftp**, change the ***disabled*** parameter from **yes** to **no**.
3. Configure your DHCP server to use the boot images packaged with **yaboot**. (If you do not have a DHCP server installed, refer to the *DHCP Servers* chapter in the Red Hat Enterprise Linux Deployment Guide.)

   A sample configuration in **/etc/dhcp/dhcpd.conf** might look like:

```
host bonn {
    filename "yaboot";
    next-server          10.32.5.1;
    hardware ethernet 00:0e:91:51:6a:26;
    fixed-address 10.32.5.144;
}
```

4. You now need the **yaboot** binary file from the **yaboot** package in the ISO image file. To access it, run the following commands as root:

```
mkdir /publicly_available_directory/yaboot-unpack
mount -t iso9660 /path_to_image/name_of_image.iso /mount_point -o loop,ro
cp -pr /mount_point/Packages/yaboot-version.ppc.rpm /publicly_available_directory/yaboot-unpack
```

   Extract the package:

```
cd /publicly_available_directory/yaboot-unpack
rpm2cpio yaboot-version.ppc.rpm | cpio -dimv
```

5. Create a **yaboot** directory within **tftpboot** and copy the **yaboot** binary file into it:

```
mkdir /var/lib/tftpboot/yaboot
cp publicly_available_directory/yaboot-unpack/usr/lib/yaboot/yaboot /var/lib/tftpboot/yaboot
```

6. Add a config file named **yaboot.conf** to this directory. A sample config file might look like:

```
init-message = "\nWelcome to the Red Hat Enterprise Linux 6 installer!\n\n"
timeout=60
default=rhel6
image=/rhel6/vmlinuz-RHEL6
        label=linux
        alias=rhel6
        initrd=/rhel6/initrd-RHEL6.img
```

```
        append="repo=http://10.32.5.1/mnt/archive/redhat/released/RHEL-
6/6.x/Server/ppc64/os/"
        read-only
```

For instructions on how to specify the installation source, refer to Section 7.1.3, "Additional Boot Options"

7. Copy the boot images from the extracted ISO into your **tftp** root directory:

```
cp /mount_point/images/ppc/ppc64/vmlinuz /var/lib/tftpboot/yaboot/rhel6/vmlinuz-RHEL6
cp /mount_point/images/ppc/ppc64/initrd.img /var/lib/tftpboot/yaboot/rhel6/initrd-RHEL6.img
```

8. Clean up by removing the **yaboot-unpack** directory and unmounting the ISO:

```
rm -rf /publicly_available_directory/yaboot-unpack
umount /mount_point
```

9. Boot the client system, and select the network device as your boot device when prompted.

30.3. STARTING THE TFTP SERVER

On the DHCP server, verify that the **tftp-server** package is installed with the command **rpm -q tftp-server**.

tftp is an xinetd-based service; start it with the following commands:

```
/sbin/chkconfig --level 345 xinetd on
/sbin/chkconfig --level 345 tftp on
```

These commands configure the **tftp** and **xinetd** services to start at boot time in runlevels 3, 4, and 5.

30.4. ADDING A CUSTOM BOOT MESSAGE

Optionally, modify **/var/lib/tftpboot/linux-install/msgs/boot.msg** to use a custom boot message.

30.5. PERFORMING THE INSTALLATION

For instructions on how to configure the network interface card to boot from the network, consult the documentation for the NIC. It varies slightly per card.

After the system boots the installation program, refer to the Chapter 9, *Installing Using Anaconda*.

CHAPTER 31. INSTALLING THROUGH VNC

The Red Hat Enterprise Linux installer (**anaconda**) offers you two interactive modes of operation. The original mode is a text-based interface. The newer mode uses GTK+ and runs in the X Window environment. This chapter explains how you can use the graphical installation mode in environments where the system lacks a proper display and input devices typically associated with a workstation. This scenario is typical of systems in datacenters, which are often installed in a rack environment and do not have a display, keyboard, or mouse. Additionally, a lot of these systems even lack the ability to connect a graphical display. Given that enterprise hardware rarely needs that ability at the physical system, this hardware configuration is acceptable.

Even in these environments, however, the graphical installer remains the recommended method of installation. The text mode environment lacks a lot of capabilities found in the graphical mode. Many users still feel that the text mode interface provides them with additional power or configuration ability not found in the graphical version. The opposite is true. Much less development effort is put in to the text-mode environment and specific things (for example, LVM configuration, partition layout, package selection, and bootloader configuration) are deliberately left out of the text mode environment. The reasons for this are:

- Less screen real estate for creating user interfaces similar to those found in the graphical mode.
- Difficult internationalization support.
- Desire to maintain a single interactive installation code path.

Anaconda therefore includes a **Virtual Network Computing** (VNC) mode that allows the graphical mode of the installer to run locally, but display on a system connected to the network. Installing in VNC mode provides you with the full range of installation options, even in situations where the system lacks a display or input devices.

31.1. VNC VIEWER

Performing a VNC installation requires a VNC viewer running on your workstation or other terminal computer. Locations where you might want a VNC viewer installed:

- Your workstation
- Laptop on a datacenter crash cart

VNC is open source software licensed under the GNU General Public License.

VNC clients are available in the repositories of most Linux distributions. Use your package manager to search for a client for your chosen distribution. For example, on Red Hat Enterprise Linux, install the tigervnc package:

```
# yum install tigervnc
```

Once you have verified you have a VNC viewer available, it's time to start the installation.

31.2. VNC MODES IN ANACONDA

Anaconda offers two modes for VNC installation. The mode you select will depend on the network configuration in your environment.

31.2.1. Direct Mode

Direct mode VNC in anaconda is when the client initiates a connection to the VNC server running in anaconda. Anaconda will tell you when to initiate this connection in the VNC viewer. Direct mode can be activated by either of the following commands:

- Specify **vnc** as a boot argument.
- Specify the **vnc** command in the kickstart file used for installation.

When you activate VNC mode, anaconda will complete the first stage of the installer and then start VNC to run the graphical installer. The installer will display a message on the console in the following format:

```
Running anaconda VERSION, the PRODUCT system installer - please wait...
```

Anaconda will also tell you the IP address and display number to use in your VNC viewer. At this point, you need to start the VNC viewer and connect to the target system to continue the installation. The VNC viewer will present anaconda to you in graphical mode.

There are some disadvantages to direct mode, including:

- Requires visual access to the system console to see the IP address and port to connect the VNC viewer to.
- Requires interactive access to the system console to complete the first stage of the installer.

If either of these disadvantages would prevent you from using direct mode VNC in anaconda, then connect mode is probably more suited to your environment.

31.2.2. Connect Mode

Certain firewall configurations or instances where the target system is configured to obtain a dynamic IP address may cause trouble with the direct VNC mode in anaconda. In addition, if you lack a console on the target system to see the message that tells you the IP address to connect to, then you will not be able to continue the installation.

The VNC connect mode changes how VNC is started. Rather than anaconda starting up and waiting for you to connect, the VNC connect mode allows anaconda to automatically connect to your view. You won't need to know the IP address of the target system in this case.

To activate the VNC connect mode, pass the **vncconnect** boot parameter:

```
boot: linux vncconnect=HOST
```

Replace HOST with your VNC viewer's IP address or DNS host name. Before starting the installation process on the target system, start up your VNC viewer and have it wait for an incoming connection.

Start the installation and when your VNC viewer displays the graphical installer, you are ready to go.

31.3. INSTALLATION USING VNC

Now that you have installed a VNC viewer application and selected a VNC mode for use in anaconda, you are ready to begin the installation.

31.3.1. Installation Example

The easiest way to perform an installation using VNC is to connect another computer directly to the network port on the target system. The laptop on a datacenter crash cart usually fills this role. If you are performing your installation this way, make sure you follow these steps:

1. Connect the laptop or other workstation to the target system using a crossover cable. If you are using regular patch cables, make sure you connect the two systems using a small hub or switch. Most recent Ethernet interfaces will automatically detect if they need to be crossover or not, so it may be possible to connect the two systems directly using a regular patch cable.

2. Configure the VNC viewer system to use a RFC 1918 address with no gateway. This private network connection will only be used for the purpose of installation. Configure the VNC viewer system to be 192.168.100.1/24. If that address is in use, just pick something else in the RFC 1918 address space that is available to you.

3. Start the installation on the target system.

    1. Booting the installation DVD.

        If booting the installation DVD, make sure **vnc** is passed as a boot parameter. To add the **vnc** parameter, you will need a console attached to the target system that allows you to interact with the boot process. Enter the following at the prompt:

        ```
        boot: linux vnc
        ```

    2. Boot over the network.

        If the target system is configured with a static IP address, add the **vnc** command to the kickstart file. If the target system is using DHCP, add **vncconnect=*HOST*** to the boot arguments for the target system. HOST is the IP address or DNS host name of the VNC viewer system. Enter the following at the prompt:

        ```
        boot: linux vncconnect=HOST
        ```

4. When prompted for the network configuration on the target system, assign it an available RFC 1918 address in the same network you used for the VNC viewer system. For example, 192.168.100.2/24.

    NOTE

    This IP address is only used during installation. You will have an opportunity to configure the final network settings, if any, later in the installer.

5. Once the installer indicates it is starting anaconda, you will be instructed to connect to the system using the VNC viewer. Connect to the viewer and follow the graphical installation mode instructions found in the product documentation.

31.3.2. Kickstart Considerations

If your target system will be booting over the network, VNC is still available. Just add the **vnc** command to the kickstart file for the system. You will be able to connect to the target system using your VNC viewer and monitor the installation progress. The address to use is the one the system is configured with via the kickstart file.

If you are using DHCP for the target system, the reverse **vncconnect** method may work better for you.

Rather than adding the **vnc** boot parameter to the kickstart file, add the **vncconnect=*HOST*** parameter to the list of boot arguments for the target system. For HOST, put the IP address or DNS host name of the VNC viewer system. See the next section for more details on using the vncconnect mode.

31.3.3. Firewall Considerations

If you are performing the installation where the VNC viewer system is a workstation on a different subnet from the target system, you may run in to network routing problems. VNC works fine so long as your viewer system has a route to the target system and ports 5900 and 5901 are open. If your environment has a firewall, make sure ports 5900 and 5901 are open between your workstation and the target system.

In addition to passing the **vnc** boot parameter, you may also want to pass the **vncpassword** parameter in these scenarios. While the password is sent in plain text over the network, it does provide an extra step before a viewer can connect to a system. Once the viewer connects to the target system over VNC, no other connections are permitted. These limitations are usually sufficient for installation purposes.

IMPORTANT

Be sure to use a temporary password for the **vncpassword** option. It should not be a password you use on any systems, especially a real root password.

If you continue to have trouble, consider using the **vncconnect** parameter. In this mode of operation, you start the viewer on your system first telling it to listen for an incoming connection. Pass **vncconnect=*HOST*** at the boot prompt and the installer will attempt to connect to the specified HOST (either a hostname or IP address).

31.4. REFERENCES

- TigerVNC: http://tigervnc.sourceforge.net/
- RFC 1918 - Address Allocation for Private Networks: http://www.ietf.org/rfc/rfc1918.txt

CHAPTER 32. KICKSTART INSTALLATIONS

32.1. WHAT ARE KICKSTART INSTALLATIONS?

Many system administrators would prefer to use an automated installation method to install Red Hat Enterprise Linux on their machines. To answer this need, Red Hat created the kickstart installation method. Using kickstart, a system administrator can create a single file containing the answers to all the questions that would normally be asked during a typical installation.

Kickstart files can be kept on a single server system and read by individual computers during the installation. This installation method can support the use of a single kickstart file to install Red Hat Enterprise Linux on multiple machines, making it ideal for network and system administrators.

Kickstart provides a way for users to automate a Red Hat Enterprise Linux installation.

All kickstart scriptlets and the log files of their execution are stored in the **/tmp** directory to assist with debugging installation failures.

NOTE

Anaconda now configures network interfaces with **NetworkManager**. Consequently, kickstart users that referenced the network settings located in **/tmp/netinfo** in previous versions of Red Hat Enterprise Linux must now source the **ifcfg** files in **/etc/sysconfig/network-scripts**.

32.2. HOW DO YOU PERFORM A KICKSTART INSTALLATION?

Kickstart installations can be performed using a local DVD, a local hard drive, or via NFS, FTP, HTTP, or HTTPS.

To use kickstart, you must:

1. Create a kickstart file.
2. Create a boot media with the kickstart file or make the kickstart file available on the network.
3. Make the installation tree available.
4. Start the kickstart installation.

This chapter explains these steps in detail.

32.3. CREATING THE KICKSTART FILE

The kickstart file is a simple text file, containing a list of items, each identified by a keyword. You can create it by using the **Kickstart Configurator** application, or writing it from scratch. The Red Hat Enterprise Linux installation program also creates a sample kickstart file based on the options that you selected during installation. It is written to the file **/root/anaconda-ks.cfg**. You should be able to edit it with any text editor or word processor that can save files as ASCII text.

First, be aware of the following issues when you are creating your kickstart file:

- Sections must be specified *in order*. Items within the sections do not have to be in a specific order unless otherwise specified. The section order is:

- Command section — Refer to Section 32.4, "Kickstart Options" for a list of kickstart options. You must include the required options.
- The **%packages** section — Refer to Section 32.5, "Package Selection" for details.
- The **%pre** and **%post** sections — These two sections can be in any order and are not required. Refer to Section 32.6, "Pre-installation Script" and Section 32.7, "Post-installation Script" for details.

NOTE

Each section should end with **%end** to avoid logged warnings.

- Items that are not required can be omitted.
- Omitting any required item results in the installation program prompting the user for an answer to the related item, just as the user would be prompted during a typical installation. Once the answer is given, the installation continues unattended (unless it finds another missing item).
- Lines starting with a pound (also known as hash) sign (#) are treated as comments and are ignored.
- For kickstart *upgrades*, the following items are required:
  - Language
  - Installation method
  - Device specification (if device is needed to perform the installation)
  - Keyboard setup
  - The **upgrade** keyword
  - Boot loader configuration

  If any other items are specified for an upgrade, those items are ignored (note that this includes package selection).

32.4. KICKSTART OPTIONS

The following options can be placed in a kickstart file. If you prefer to use a graphical interface for creating your kickstart file, use the **Kickstart Configurator** application. Refer to Chapter 33, ***Kickstart Configurator*** for details.

NOTE

If the option is followed by an equals mark (=), a value must be specified after it. In the example commands, options in square brackets (**[ ]**) are optional arguments for the command.

IMPORTANT

Device names are not guaranteed to be consistent across reboots, which can complicate usage in kickstart scripts. When a kickstart option calls for a device node name (such as **sda**), you can instead use any item from **/dev/disk**. For example, instead of:

```
part / --fstype=ext4 --onpart=sda1
```

You could use an entry similar to one of the following:

```
part / --fstype=ext4 --onpart=/dev/disk/by-path/pci-0000:00:05.0-scsi-0:0:0:0-part1
part / --fstype=ext4 --onpart=/dev/disk/by-id/ata-ST3160815AS_6RA0C882-part1
```

This provides a consistent way to refer to disks that is more meaningful than just **sda**. This is especially useful in large storage environments.

auth or **authconfig** (required)

Sets up the authentication options for the system. It is similar to the **authconfig** command, which can be run after the installation - see the **authconfig(8)** man page for more information.

Passwords are shadowed by default.

WARNING

The **authconfig** command requires the authconfig package, which is not included when using the minimal package group. Add **authconfig** to the **%packages** section as described in Section 32.5, "Package Selection", if you are using the minimal package group and want to use this command in your Kickstart file.

WARNING

When using OpenLDAP with the **SSL** protocol for security, make sure that the **SSLv2** and **SSLv3** protols are disabled in the server configuration. This is due to the POODLE SSL vulnerability (CVE-2014-3566). See https://access.redhat.com/solutions/1234843 for details.

- **--enablenis** – Turns on NIS support. By default, **--enablenis** uses whatever domain it finds on the network. A domain should almost always be set by hand with the **--nisdomain=** option.
- **--nisdomain=** – NIS domain name to use for NIS services.
- **--nisserver=** – Server to use for NIS services (broadcasts by default).

- **--useshadow** or **--enableshadow** – Use shadow passwords. This option is enabled by default.

- **--enableldap** – Turns on LDAP support in **/etc/nsswitch.conf**, allowing your system to retrieve information about users (for example, their UIDs, home directories, and shells) from an LDAP directory. To use this option, you must install the **nss-pam-ldapd** package. You must also specify a server and a base *DN* (distinguished name) with **--ldapserver=** and **--ldapbasedn=**.

- **--enableldapauth** – Use LDAP as an authentication method. This enables the **pam_ldap** module for authentication and changing passwords, using an LDAP directory. To use this option, you must have the **nss-pam-ldapd** package installed. You must also specify a server and a base DN with **--ldapserver=** and **--ldapbasedn=**. If your environment does not use *TLS* (Transport Layer Security), use the **--disableldaptls** switch to ensure that the resulting configuration file works.

- **--ldapserver=** – If you specified either **--enableldap** or **--enableldapauth**, use this option to specify the name of the LDAP server to use. This option is set in the **/etc/ldap.conf** file.

- **--ldapbasedn=** – If you specified either **--enableldap** or **--enableldapauth**, use this option to specify the DN in your LDAP directory tree under which user information is stored. This option is set in the **/etc/ldap.conf** file.

- **--enableldaptls** – Use TLS (Transport Layer Security) lookups. This option allows LDAP to send encrypted usernames and passwords to an LDAP server before authentication.

- **--disableldaptls** – Do not use TLS (Transport Layer Security) lookups in an environment that uses LDAP for authentication.

- **--enablekrb5** – Use Kerberos 5 for authenticating users. Kerberos itself does not know about home directories, UIDs, or shells. If you enable Kerberos, you must make users' accounts known to this workstation by enabling LDAP, NIS, or Hesiod or by using the **/usr/sbin/useradd** command. If you use this option, you must have the **pam_krb5** package installed.

- **--krb5realm=** – The Kerberos 5 realm to which your workstation belongs.

- **--krb5kdc=** – The KDC (or KDCs) that serve requests for the realm. If you have multiple KDCs in your realm, separate their names with commas (,).

- **--krb5adminserver=** – The KDC in your realm that is also running kadmind. This server handles password changing and other administrative requests. This server must be run on the master KDC if you have more than one KDC.

- **--enablehesiod** – Enable Hesiod support for looking up user home directories, UIDs, and shells. More information on setting up and using Hesiod on your network is in **/usr/share/doc/glibc-2.x.x/README.hesiod**, which is included in the **glibc** package. Hesiod is an extension of DNS that uses DNS records to store information about users, groups, and various other items.

- **--hesiodlhs** and **--hesiodrhs** – The **Hesiod** LHS (left-hand side) and RHS (right-hand side) values, set in **/etc/hesiod.conf**. The **Hesiod** library uses these values to search DNS for a name, similar to the way that **LDAP** uses a base DN.

  To look up user information for the username **jim**, the Hesiod library looks up **jim.passwd<*LHS*><*RHS*>**, which should resolve to a TXT record that contains a string identical to an entry for that user in the **passwd** file: **jim:*:501:501:Jungle**

Jim:/home/jim:/bin/bash. To look up groups, the Hesiod library looks up **jim.group<*LHS*><*RHS*>** instead.

To look up users and groups by number, make **501.uid** a CNAME for **jim.passwd**, and **501.gid** a CNAME for **jim.group**. Note that the library does not place a period (**.**) in front of the LHS and RHS values when performing a search. Therefore, if the LHS and RHS values need to have a period placed in front of them, you must include the period in the values you set for **--hesiodlhs** and **--hesiodrhs**.

- **--enablesmbauth** – Enables authentication of users against an SMB server (typically a Samba or Windows server). SMB authentication support does not know about home directories, UIDs, or shells. If you enable SMB, you must make users' accounts known to the workstation by enabling LDAP, NIS, or Hesiod or by using the **/usr/sbin/useradd** command.

- **--smbservers=** – The name of the servers to use for SMB authentication. To specify more than one server, separate the names with commas (,).

- **--smbworkgroup=** – The name of the workgroup for the SMB servers.

- **--enablecache** – Enables the **nscd** service. The **nscd** service caches information about users, groups, and various other types of information. Caching is especially helpful if you choose to distribute information about users and groups over your network using NIS, LDAP, or Hesiod.

- **--passalgo=** – specify **sha256** to set up the SHA-256 hashing algorithm or **sha512** to set up the SHA-512 hashing algorithm.

autopart (optional)

Automatically creates partitions – a root (/) partition (1 GB or bigger), a swap partition, and an appropriate boot partition for the architecture.

NOTE

Note that the **autopart** option cannot be used together with the **part/partition**, **raid**, **logvol**, or **volgroup** options in the same kickstart file.

- **--encrypted** – Should all devices with support be encrypted by default? This is equivalent to checking the **Encrypt** checkbox on the initial partitioning screen.

- **--cipher=** – Specifies which type of encryption will be used if the **anaconda** default aes-xts-plain64 is not satisfactory. You must use this option together with the **--encrypted** option; by itself it has no effect. Available types of encryption are listed in the *Red Hat Enterprise Linux Security Guide*, but Red Hat strongly recommends using either aes-xts-plain64 or aes-cbc-essiv:sha256.

- **--passphrase=** – Provide a default system-wide passphrase for all encrypted devices.

- **--escrowcert=*URL_of_X.509_certificate*** – Store data encryption keys of all encrypted volumes as files in **/root**, encrypted using the X.509 certificate from the URL specified with *URL_of_X.509_certificate*. The keys are stored as a separate file for each encrypted volume. This option is only meaningful if **--encrypted** is specified.

- **--backuppassphrase=** – Add a randomly-generated passphrase to each encrypted volume. Store these passphrases in separate files in **/root**, encrypted using the X.509 certificate specified with **--escrowcert**. This option is only meaningful if **--escrowcert** is specified.

autostep (optional)

Similar to **interactive** except it goes to the next screen for you. It is used mostly for debugging and should not be used when deploying a system because it may disrupt package installation.

- **--autoscreenshot** – Take a screenshot at every step during installation and copy the images over to **/root/anaconda-screenshots** after installation is complete. This is most useful for documentation.

bootloader (required)

Specifies how the boot loader should be installed. This option is required for both installations and upgrades.

IMPORTANT

If you select text mode for a kickstart installation, make sure that you specify choices for the partitioning, bootloader, and package selection options. These steps are automated in text mode, and **anaconda** cannot prompt you for missing information. If you do not provide choices for these options, **anaconda** will stop the installation process.

IMPORTANT

It is highly recommended to set up a boot loader password on every machine. An unprotected boot loader can allow a potential attacker to modify the system's boot options and gain access to the system. See the chapter titled *Workstation Security* in the *Red Hat Enterprise Linux Security Guide* for more information on boot loader passwords and password security in general.

- **--append=** – Specifies kernel parameters. To specify multiple parameters, separate them with spaces. For example:

```
bootloader --location=mbr --append="hdd=ide-scsi ide=nodma"
```

- **--driveorder** – Specify which drive is first in the BIOS boot order. For example:

```
bootloader --driveorder=sda,hda
```

- **--disabled** – This option is a stronger version of **--location=none**. While **--location=none** simply disables bootloader installation, **--disabled** disables bootloader installation and also disables installation of the bootloader package, thus saving space.

- **--location=** – Specifies where the boot record is written. Valid values are the following: **mbr** (the default), **partition** (installs the boot loader on the first sector of the partition containing the kernel – necessary for UEFI), or **none** (do not install the boot loader).

IMPORTANT

64-bit AMD and Intel systems with UEFI firmware require the boot loader to be installed in an EFI system partition on a disk labeled with a GUID Partition Table (GPT). Using a disk with a Master Boot Record (MBR) label requires that the disk be relabeled using the **clearpart** and **zerombr** commands. Relabeling a disk will render all data on that disk inaccessible and it will require creating a new partition layout.

- **--password=** – If using GRUB, sets the GRUB boot loader password to the one specified with this option. This should be used to restrict access to the GRUB shell, where arbitrary kernel options can be passed.
- **--iscrypted** – If using GRUB, should be included if the password is already encrypted. The encryption method is detected automatically based on the password.

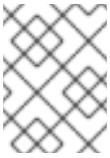
  To create an encrypted password, use the following command:

  ```
  python -c 'import crypt; print(crypt.crypt("My Password"))'
  ```

  This will create a sha512 crypt of your password.
- **--upgrade** – Upgrade the existing boot loader configuration, preserving the old entries. This option is only available for upgrades.

clearpart (optional)

Removes partitions from the system, prior to creation of new partitions. By default, no partitions are removed.



NOTE

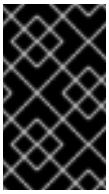
If the **clearpart** command is used, then the **--onpart** command cannot be used on a logical partition.

- **--all** – Erases all partitions from the system.

WARNING

This option will erase all disks which can be reached by the installer, including any attached network storage. Use this option with caution.

You can prevent **clearpart** from wiping storage you want to preserve by using the **--drives=** option and specifying only the drives you want to clear, by attaching network storage later (for example, in the **%post** section of the Kickstart file), or by blacklisting the kernel modules used to access network storage.



IMPORTANT

The **clearpart** cannot clear an existing BIOS RAID setup. For this, the command **wipefs -a** must be added to your **%pre** script. Note that this will wipe all metadata from the RAID.

- **--drives=** – Specifies which drives to clear partitions from. For example, the following clears all the partitions on the first two drives on the primary IDE controller:

```
clearpart --drives=hda,hdb --all
```

To clear a multipath device, use the format **disk/by-id/scsi-*WWID***, where *WWID* is the *world-wide identifier* for the device. For example, to clear a disk with WWID **58095BEC5510947BE8C0360F604351918**, use:

```
clearpart --drives=disk/by-id/scsi-58095BEC5510947BE8C0360F604351918
```

This format is preferable for all multipath devices, but if errors arise, multipath devices that do not use *logical volume management* (LVM) can also be cleared using the format **disk/by-id/dm-uuid-mpath-*WWID***, where *WWID* is the *world-wide identifier* for the device. For example, to clear a disk with WWID **2416CD96995134CA5D787F00A5AA11017**, use:

```
clearpart --drives=disk/by-id/dm-uuid-mpath-2416CD96995134CA5D787F00A5AA11017
```

WARNING

Never specify multipath devices by device names like **mpatha**. Device names like **mpatha** are not specific to a particular disk. The disk named **/dev/mpatha** during installation might not be the one that you expect it to be. Therefore, the **clearpart** command could target the wrong disk.

- **--initlabel** – Initializes a disk (or disks) by creating a default disk label for all disks in their respective architecture that have been designated for formatting (for example, msdos for x86). Because **--initlabel** can see all disks, it is important to ensure only those drives that are to be formatted are connected.

```
clearpart --initlabel --drives=names_of_disks
```

For example:

```
clearpart --initlabel --drives=dasda,dasdb,dasdc
```

- **--linux** – Erases all Linux partitions.
- **--none** (default) – Do not remove any partitions.
- **--cdl** – Reformat all detected LDL (*Linux Disk Layout*) disks to CDL (*Compatible Disk Layout*). Only available on IBM System z.

NOTE

Using the **clearpart --all** command in a Kickstart file to remove all existing partitions during the installation will cause **Anaconda** to pause and prompt you for a confirmation. If you need to perform the installation automatically with no interaction, add the **zerombr** command to your Kickstart file.

cmdline (optional)

Perform the installation in a completely non-interactive command line mode. Any prompts for interaction halts the install. This mode is useful on IBM System z systems with the 3270 terminal under z/VM and operating system messages applet on LPAR. The recommended use is in conjunction with **RUNKS=1** and **ks=**. Refer to Section 26.6, "Parameters for Kickstart Installations" .

device (optional)

On most PCI systems, the installation program autoprobes for Ethernet and SCSI cards properly. On older systems and some PCI systems, however, kickstart needs a hint to find the proper devices. The **device** command, which tells the installation program to install extra modules, is in this format:

```
device <moduleName> --opts=<options>
```

- *<moduleName>* – Replace with the name of the kernel module which should be installed.
- **--opts=** – Options to pass to the kernel module. For example:

```
--opts="aic152x=0x340 io=11"
```

driverdisk (optional)

Driver disks can be used during kickstart installations. You must copy the driver disks's contents to the root directory of a partition on the system's hard drive. Then you must use the **driverdisk** command to tell the installation program where to look for the driver disk.

```
driverdisk <partition> --source=<url> --biospart=<biospart> [--type=<fstype>]
```

Alternatively, a network location can be specified for the driver disk:

```
driverdisk --source=ftp://path/to/dd.img
driverdisk --source=http://path/to/dd.img
driverdisk --source=nfs:host:/path/to/img
```

- *<partition>* – Partition containing the driver disk.
- *<url>* – URL for the driver disk. NFS locations can be given in the form **nfs:*host*:/*path/to/img***.
- *<biospart>* – BIOS partition containing the driver disk (for example, **82p2**).
- **--type=** – File system type (for example, vfat or ext2).

fcoe (optional)

Specify which FCoE devices should be activated automatically in addition to those discovered by *Enhanced Disk Drive Services* (EDD).

- **--nic=** (mandatory) — The name of the device to be activated.
- **--dcb=** — Establish *Data Center Bridging* (DCB) settings.
- **--autovlan** — Discover VLANs automatically.

firewall (optional)

This option corresponds to the **Firewall Configuration** screen in the installer.

```
firewall --enabled|--disabled [--trust=] <device> <incoming> [--port=]
```

WARNING

The **firewall** command requires the system-config-firewall-base package, which is not included when using the minimal package group. Add **system-config-firewall-base** to the **%packages** section as described in Section 32.5, "Package Selection", if you are using the minimal package group and you want to use this command in your Kickstart file.

- **--enabled** or **--enable** — Reject incoming connections that are not in response to outbound requests, such as DNS replies or DHCP requests. If access to services running on this machine is needed, you can choose to allow specific services through the firewall.
- **--disabled** or **--disable** — Do not configure any iptables rules.
- **--trust=** — Listing a device here, such as eth0, allows all traffic coming to and from that device to go through the firewall. To list more than one device, use **--trust eth0 --trust eth1**. Do NOT use a comma-separated format such as **--trust eth0, eth1**.
- *<incoming>* — Replace with one or more of the following to allow the specified services through the firewall.
  - **--ssh**
  - **--telnet**
  - **--smtp**
  - **--http**
  - **--ftp**
- **--port=** — You can specify that ports be allowed through the firewall using the port:protocol format. For example, to allow IMAP access through your firewall, specify **imap:tcp**. Numeric ports can also be specified explicitly; for example, to allow UDP packets on port 1234 through, specify **1234:udp**. To specify multiple ports, separate them by commas.

firstboot (optional)

Determine whether the **firstboot** starts the first time the system is booted. If enabled, the firstboot package must be installed. If not specified, this option is disabled by default.

- **--enable** or **--enabled** — The **Setup Agent** is started the first time the system boots.
- **--disable** or **--disabled** — The **Setup Agent** is not started the first time the system boots.
- **--reconfig** — Enable the **Setup Agent** to start at boot time in reconfiguration mode. This mode enables the language, mouse, keyboard, root password, security level, and time zone configuration options in addition to the default ones.

graphical (optional)

Perform the kickstart installation in graphical mode. This is the default.

group (optional)

Creates a new user group on the system. If a group with the given name or GID already exists, this command will fail. In addition, the **user** command can be used to create a new group for the newly created user.

> **group --name=*name* [--gid=*gid*]**

- **--name=** - Provides the name of the group.
- **--gid=** - The group's GID. If not provided, defaults to the next available non-system GID.

halt (optional)

Halt the system after the installation has successfully completed. This is similar to a manual installation, where anaconda displays a message and waits for the user to press a key before rebooting. During a kickstart installation, if no completion method is specified, this option is used as the default.

The **halt** option is equivalent to the **shutdown -h** command.

For other completion methods, refer to the **poweroff**, **reboot**, and **shutdown** kickstart options.

ignoredisk (optional)

Causes the installer to ignore the specified disks. This is useful if you use autopartition and want to be sure that some disks are ignored. For example, without **ignoredisk**, attempting to deploy on a SAN-cluster the kickstart would fail, as the installer detects passive paths to the SAN that return no partition table.

The syntax is:

```
ignoredisk --drives=drive1,drive2,...
```

where *driveN* is one of **sda**, **sdb**,..., **hda**,... etc.

To ignore a multipath device that does not use *logical volume management* (LVM), use the format **disk/by-id/dm-uuid-mpath-*WWID***, where *WWID* is the *world-wide identifier* for the device. For example, to ignore a disk with WWID **2416CD96995134CA5D787F00A5AA11017**, use:

```
ignoredisk --drives=disk/by-id/dm-uuid-mpath-2416CD96995134CA5D787F00A5AA11017
```

Multipath devices that use LVM are not assembled until after **anaconda** has parsed the kickstart file. Therefore, you cannot specify these devices in the format **dm-uuid-mpath**. Instead, to ignore a multipath device that uses LVM, use the format **disk/by-id/scsi-*WWID***, where *WWID* is the *world-wide identifier* for the device. For example, to ignore a disk with WWID **58095BEC5510947BE8C0360F604351918**, use:

```
ignoredisk --drives=disk/by-id/scsi-58095BEC5510947BE8C0360F604351918
```

WARNING

Never specify multipath devices by device names like **mpatha**. Device names like **mpatha** are not specific to a particular disk. The disk named **/dev/mpatha** during installation might not be the one that you expect it to be. Therefore, the **clearpart** command could target the wrong disk.

- **--only-use** – specifies a list of disks for the installer to use. All other disks are ignored. For example, to use disk **sda** during installation and ignore all other disks:

  ```
  ignoredisk --only-use=sda
  ```

  To include a multipath device that does not use LVM:

  ```
  ignoredisk --only-use=disk/by-id/dm-uuid-mpath-
  2416CD96995134CA5D787F00A5AA11017
  ```

  To include a multipath device that uses LVM:

  ```
  ignoredisk --only-use=disk/by-id/scsi-58095BEC5510947BE8C0360F604351918
  ```

install (optional)

Tells the system to install a fresh system rather than upgrade an existing system. This is the default mode. For installation, you must specify the type of installation from **cdrom**, **harddrive**, **nfs**, or **url** (for FTP, HTTP, or HTTPS installations). The **install** command and the installation method command must be on separate lines.

- **cdrom** – Install from the first optical drive on the system.
- **harddrive** – Install from a Red Hat installation tree on a local drive, which must be either vfat or ext2.
  - **--biospart=**

    BIOS partition to install from (such as 82).
  - **--partition=**

    Partition to install from (such as sdb2).

- **--dir=**

  Directory containing the ***variant*** directory of the installation tree.

For example:

```
harddrive --partition=hdb2 --dir=/tmp/install-tree
```

- **nfs** – Install from the NFS server specified.
  - **--server=**

    Server from which to install (hostname or IP).

  - **--dir=**

    Directory containing the ***variant*** directory of the installation tree.

  - **--opts=**

    Mount options to use for mounting the NFS export. (optional)

  For example:

```
nfs --server=nfsserver.example.com --dir=/tmp/install-tree
```

- **url** – Install from an installation tree on a remote server using the FTP, HTTP, or HTTPS protocol. You can only specify one URL.

  For example:

```
url --url http://<server>/<dir>
```

  or:

```
url --url ftp://<username>:<password>@<server>/<dir>
```

interactive (optional)

Perform an interactive installation, but use the information in the kickstart file to provide defaults. During the installation, **anaconda** still prompts you at every stage. Either accept the values from the kickstart file by clicking **Next** or change the values and click **Next** to continue. Refer also to the **autostep** command.

iscsi (optional)

```
iscsi --ipaddr=<ipaddr> [options]
```

Specifies additional iSCSI storage to be attached during installation. If you use the ***iscsi*** parameter, you must also assign a name to the iSCSI node, using the ***iscsiname*** parameter *earlier* in the kickstart file.

We recommend that wherever possible you configure iSCSI storage in the system BIOS or firmware (iBFT for Intel systems) rather than use the ***iscsi*** parameter. **Anaconda** automatically detects and

uses disks configured in BIOS or firmware and no special configuration is necessary in the kickstart file.

If you must use the ***iscsi*** parameter, ensure that networking is activated at the beginning of the installation, and that the ***iscsi*** parameter appears in the kickstart file before you refer to iSCSI disks with parameters such as ***clearpart*** or ***ignoredisk***.

- **--port=** (mandatory) — the port number (typically, **--port=3260**)
- **--user=** — the username required to authenticate with the target
- **--password=** — the password that corresponds with the username specified for the target
- **--reverse-user=** — the username required to authenticate with the initiator from a target that uses reverse CHAP authentication
- **--reverse-password=** — the password that corresponds with the username specified for the initiator

iscsiname (optional)

Assigns a name to an iSCSI node specified by the iscsi parameter. If you use the ***iscsi*** parameter in your kickstart file, you must specify ***iscsiname*** *earlier* in the kickstart file.

keyboard (required)

Sets the default keyboard type for the system. The available keyboard types are:

- **be-latin1** — Belgian
- **bg_bds-utf8** — Bulgarian
- **bg_pho-utf8** — Bulgarian (Phonetic)
- **br-abnt2** — Brazilian (ABNT2)
- **cf** — French Canadian
- **croat** — Croatian
- **cz-us-qwertz** — Czech
- **cz-lat2** — Czech (qwerty)
- **de** — German
- **de-latin1** — German (latin1)
- **de-latin1-nodeadkeys** — German (latin1 without dead keys)
- **dvorak** — Dvorak
- **dk** — Danish
- **dk-latin1** — Danish (latin1)
- **es** — Spanish

- **et** – Estonian
- **fi** – Finnish
- **fi-latin1** – Finnish (latin1)
- **fr** – French
- **fr-latin9** – French (latin9)
- **fr-latin1** – French (latin1)
- **fr-pc** – French (pc)
- **fr_CH** – Swiss French
- **fr_CH-latin1** – Swiss French (latin1)
- **gr** – Greek
- **hu** – Hungarian
- **hu101** – Hungarian (101 key)
- **is-latin1** – Icelandic
- **it** – Italian
- **it-ibm** – Italian (IBM)
- **it2** – Italian (it2)
- **jp106** – Japanese
- **ko** – Korean
- **la-latin1** – Latin American
- **mk-utf** – Macedonian
- **nl** – Dutch
- **no** – Norwegian
- **pl2** – Polish
- **pt-latin1** – Portuguese
- **ro** – Romanian
- **ru** – Russian
- **sr-cy** – Serbian
- **sr-latin** – Serbian (latin)
- **sv-latin1** – Swedish

- **sg** – Swiss German
- **sg-latin1** – Swiss German (latin1)
- **sk-qwerty** – Slovak (qwerty)
- **slovene** – Slovenian
- **trq** – Turkish
- **uk** – United Kingdom
- **ua-utf** – Ukrainian
- **us-acentos** – U.S. International
- **us** – U.S. English

The file **/usr/lib/python2.6/site-packages/system_config_keyboard/keyboard_models.py** on 32-bit systems or **/usr/lib64/python2.6/site-packages/system_config_keyboard/keyboard_models.py** on 64-bit systems also contains this list and is part of the system-config-keyboard package.

lang (required)

Sets the language to use during installation and the default language to use on the installed system. For example, to set the language to English, the kickstart file should contain the following line:

```
lang en_US
```

The file **/usr/share/system-config-language/locale-list** provides a list of the valid language codes in the first column of each line and is part of the **system-config-language** package.

Certain languages (for example, Chinese, Japanese, Korean, and Indic languages) are not supported during text-mode installation. If you specify one of these languages with the **lang** command, the installation process continues in English, but the installed system uses your selection as its default language.

langsupport (deprecated)

The langsupport keyword is deprecated and its use will cause an error message to be printed to the screen and installation to halt. Instead of using the langsupport keyword, you should now list the support package groups for all languages you want supported in the **%packages** section of your kickstart file. For instance, adding support for French means you should add the following to **%packages**:

```
@french-support
```

logging (optional)

This command controls the error logging of anaconda during installation. It has no effect on the installed system.

```
logging [--host=<host>] [--port=<port>] [--level=debug|info|error|critical]
```

- **--host=** – Send logging information to the given remote host, which must be running a syslogd process configured to accept remote logging.

- **--port=** — If the remote syslogd process uses a port other than the default, it may be specified with this option.
- **--level=** — One of debug, info, warning, error, or critical.

  Specify the minimum level of messages that appear on tty3. All messages will still be sent to the log file regardless of this level, however.

logvol (optional)

Create a logical volume for Logical Volume Management (LVM) with the syntax:

```
logvol <mntpoint> --vgname=<name> --size=<size> --name=<name> [options]
```

IMPORTANT

Do not use the dash ("**-**") character in logical volume or volume group names when installing Red Hat Enterprise Linux using Kickstart. If you do, the installation will finish normally, but the character will be removed from all newly created volume and volume group names. For example, if you create a volume group named **volgrp-01**, its name will be changed to **volgrp01**.

This limitation only applies to new installations. If you are upgrading or reinstalling an existing installation and use the **--noformat** option described below, dashes used in volume and volume group names will be preserved.

- The *<mntpoint>* is where the partition is mounted and must be of one of the following forms:
  - /***<path>***

    For example, /, /**usr**, /**home**
  - **swap**

    The partition is used as swap space.

    To determine the size of the swap partition automatically, use the **--recommended** option:

    ```
    swap --recommended
    ```

    The size assigned will be effective but not precisely calibrated for your system.

    To determine the size of the swap partition automatically but also allow extra space for your system to hibernate, use the **--hibernation** option:

    ```
    swap --hibernation
    ```

    The size assigned will be equivalent to the swap space assigned by **--recommended** plus the amount of RAM on your system.

    For the swap sizes assigned by these commands, refer to Section 9.15.5, "Recommended Partitioning Scheme" for x86, AMD64, and Intel 64 Architecture and Section 16.17.5, "Recommended Partitioning Scheme" for IBM Power Systems servers.

IMPORTANT

Swap space recommendations were updated in Red Hat Enterprise Linux 6.3. Previously, systems with large amounts of RAM were assigned huge swap spaces. This delayed the Out-of-Memory Killer (**oom_kill**) in addressing critical memory shortages, even if a process was malfunctioning.

Consequently, if you are using an earlier version of Red Hat Enterprise Linux 6, **swap --recommended** will generate larger swap spaces than those described in the Recommended Partitioning Scheme, even on systems with large amounts of RAM. This may negate the need to allow extra space for hibernation.

However, these updated swap space values are nonetheless recommended for earlier versions of Red Hat Enterprise Linux 6 and can be set manually using the **swap --size=** option.

The options are as follows:

- **--noformat** – Use an existing logical volume and do not format it.
- **--useexisting** – Use an existing logical volume and reformat it.
- **--fstype=** – Sets the file system type for the logical volume. Valid values are **xfs**, **ext2**, **ext3**, **ext4**, **swap**, **vfat**, **hfs**, and **efi**.
- **--fsoptions=** – Specifies a free form string of options to be used when mounting the filesystem. This string will be copied into the **/etc/fstab** file of the installed system and should be enclosed in quotes.
- **--fsprofile** – Specifies a *usage type* to be passed to the program that makes a filesystem on this partition. A usage type defines a variety of tuning parameters to be used when making a filesystem. For this option to work, the filesystem must support the concept of usage types and there must be a configuration file that lists valid types. For ext2, ext3, and ext4, this configuration file is **/etc/mke2fs.conf**.
- **--grow=** – Tells the logical volume to grow to fill available space (if any), or up to the maximum size setting.
- **--maxsize=** – The maximum size in megabytes when the logical volume is set to grow. Specify an integer value here such as **500** (do not include the unit).
- **--recommended=** – Determine the size of the logical volume automatically.
- **--percent=** – Specify the amount by which to grow the logical volume, as a percentage of the free space in the volume group after any statically-sized logical volumes are taken into account. This option must be used in conjunction with the **--size** and **--grow** options for **logvol**.
- **--encrypted** – Specifies that this logical volume should be encrypted, using the passphrase provided in the **--passphrase** option. If you do not specify a passphrase, **anaconda** uses the default, system-wide passphrase set with the **autopart --passphrase** command, or stops the installation and prompts you to provide a passphrase if no default is set.
- **--cipher=** – Specifies which type of encryption will be used if the **anaconda** default aes-xts-

plain64 is not satisfactory. You must use this option together with the **--encrypted** option; by itself it has no effect. Available types of encryption are listed in the *Red Hat Enterprise Linux Security Guide*, but Red Hat strongly recommends using either aes-xts-plain64 or aes-cbc-essiv:sha256.

- **--passphrase=** — Specifies the passphrase to use when encrypting this logical volume. You must use this option together with the **--encrypted** option; by itself it has no effect.
- **--escrowcert=*URL_of_X.509_certificate*** — Store data encryption keys of all encrypted volumes as files in **/root**, encrypted using the X.509 certificate from the URL specified with *URL_of_X.509_certificate*. The keys are stored as a separate file for each encrypted volume. This option is only meaningful if **--encrypted** is specified.
- **--backuppassphrase=** — Add a randomly-generated passphrase to each encrypted volume. Store these passphrases in separate files in **/root**, encrypted using the X.509 certificate specified with **--escrowcert**. This option is only meaningful if **--escrowcert** is specified.

Create the partition first, create the logical volume group, and then create the logical volume. For example:

```
part pv.01 --size 3000
volgroup myvg pv.01
logvol / --vgname=myvg --size=2000 --name=rootvol
```

Create the partition first, create the logical volume group, and then create the logical volume to occupy 90% of the remaining space in the volume group. For example:

```
part pv.01 --size 1 --grow
volgroup myvg pv.01
logvol / --vgname=myvg --size=1 --name=rootvol --grow --percent=90
```

mediacheck (optional)

If given, this will force anaconda to run mediacheck on the installation media. This command requires that installs be attended, so it is disabled by default.

monitor (optional)

If the monitor command is not given, anaconda will use X to automatically detect your monitor settings. Please try this before manually configuring your monitor.

```
monitor --monitor=<monitorname>|--hsync|vsync=<frequency> [--noprobe]
```

- **--hsync=** — Specifies the horizontal sync frequency of the monitor.
- **--monitor=** — Use specified monitor; monitor name should be from the list of monitors in /usr/share/hwdata/MonitorsDB from the hwdata package. The list of monitors can also be found on the X Configuration screen of the Kickstart Configurator. This is ignored if --hsync or --vsync is provided. If no monitor information is provided, the installation program tries to probe for it automatically.
- **--noprobe=** — Do not try to probe the monitor.
- **--vsync=** — Specifies the vertical sync frequency of the monitor.

mouse (deprecated)

The mouse keyword is deprecated.

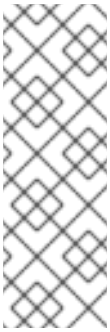
network (optional)

Configures network information for the target system and activates network devices in the installer environment. The device specified in the first **network** command is activated automatically if network access is required during installation, for example, during a network installation or installation over VNC. From Red Hat Enterprise Linux 6.1 onwards, you can also explicitly require device to activate in the installer environment with the **--activate** option.

IMPORTANT

If you need to manually specify network settings during an otherwise-automated kickstart installation, do not use **network**. Instead, boot the system with the **asknetwork** option (refer to Section 32.11, "Starting a Kickstart Installation"), which will prompt **anaconda** to ask you for network settings rather than use the default settings. **anaconda** will ask this before fetching the kickstart file.

Once the network connection is established, you can only reconfigure network settings with those specified in your kickstart file.



NOTE

You will only be prompted for information about your network:

- before fetching the kickstart file if you are using the **asknetwork** boot option
- when the network is first accessed once the kickstart file has been fetched, if the network was not used to fetch it and you have provided no kickstart network commands

- **--activate** – activate this device in the installer environment.

  If you use the **--activate** option on a device that has already been activated (for example, an interface you configured with boot options so that the system could retrieve the kickstart file) the device is reactivated to use the details specified in the kickstart file.

  Use the **--nodefroute** option to prevent the device from using the default route.

  The **activate** option is new in Red Hat Enterprise Linux 6.1.

- **--bootproto=** – One of **dhcp**, **bootp**, **ibft**, or **static**.

  The **ibft** option is new in Red Hat Enterprise Linux 6.1.

  The **bootproto** option defaults to **dhcp**. **bootp** and **dhcp** are treated the same.

  The DHCP method uses a DHCP server system to obtain its networking configuration. As you might guess, the BOOTP method is similar, requiring a BOOTP server to supply the networking configuration. To direct a system to use DHCP:

  ```
  network --bootproto=dhcp
  ```

  To direct a machine to use BOOTP to obtain its networking configuration, use the following line in the kickstart file:

```
network --bootproto=bootp
```

To direct a machine to use the configuration specified in iBFT, use:

```
network --bootproto=ibft
```

The static method requires that you specify the IP address, netmask, gateway, and nameserver in the kickstart file. As the name implies, this information is static and is used during and after the installation.

All static networking configuration information must be specified on *one* line; you cannot wrap lines using a backslash as you can on a command line. A line that specifies static networking in a kickstart file is therefore more complex than lines that specify DHCP, BOOTP, or iBFT. Note that the examples on this page have line breaks in them for presentation reasons; they would not work in an actual kickstart file.

```
network --bootproto=static --ip=10.0.2.15 --netmask=255.255.255.0
 --gateway=10.0.2.254 --nameserver=10.0.2.1
```

You can also configure multiple nameservers here. To do so, specify them as a comma-delimited list in the command line.

```
network --bootproto=static --ip=10.0.2.15 --netmask=255.255.255.0
 --gateway=10.0.2.254 --nameserver 192.168.2.1,192.168.3.1
```

- **--device=** – specifies the device to be configured (and eventually activated) with the **network** command. For the first **network** command, **--device=** defaults (in order of preference) to one of:
  - the device specified by the **ksdevice** boot option
  - the device activated automatically to fetch the kickstart file
  - the device selected in the **Networking Devices** dialog

  The behavior of any subsequent **network** command is unspecified if its **--device** option is missing. Take care to specify a **--device** option for any network command beyond the first.

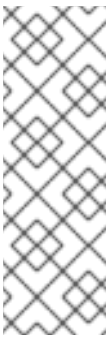
  You can specify a device in one of five ways:

  - the device name of the interface, for example, **eth0**
  - the MAC address of the interface, for example, **00:12:34:56:78:9a**
  - the keyword **link**, which specifies the first interface with its link in the **up** state
  - the keyword **bootif**, which uses the MAC address that **pxelinux** set in the ***BOOTIF*** variable. Set **IPAPPEND 2** in your **pxelinux.cfg** file to have **pxelinux** set the ***BOOTIF*** variable.
  - the keyword **ibft**, which uses the MAC address of the interface specified by iBFT

  ```
  network --bootproto=dhcp --device=eth0
  ```

- **--ip=** – IP address of the device.
- **--ipv6=** – IPv6 address of the device. Use **auto** for automatic configuration, or **dhcp** for DHCPv6 only configuration (no router advertisements).
- **--gateway=** – Default gateway as a single IPv4 address.
- **--ipv6gateway=** – Default gateway as a single IPv6 address.
- **--nameserver=** – Primary nameserver, as an IP address. Multiple nameservers must each be separated by a comma.
- **--nodefroute** – Prevents the interface being set as the default route. Use this option when you activate additional devices with the **--activate=** option, for example, a NIC on a separate subnet for an iSCSI target.

  The **nodefroute** option is new in Red Hat Enterprise Linux 6.1.
- **--nodns** – Do not configure any DNS server.
- **--netmask=** – Network mask of the device.
- **--hostname=** – Hostname for the installed system.
- **--ethtool=** – Specifies additional low-level settings for the network device which will be passed to the ethtool program.
- **--onboot=** – Whether or not to enable the device at boot time.
- **--dhcpclass=** – The DHCP class.
- **--mtu=** – The MTU of the device.
- **--noipv4** – Disable configuration of IPv4 on this device.
- **--noipv6** – Disable configuration of IPv6 on this device.



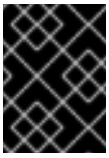
NOTE

The **--noipv6** kickstart option does not currently disable IPv6 configuration of individual devices, due to a bug. However, disabling ipv6 system-wide is possible by using the **--noipv6** option on every network device and using the **noipv6** boot parameter. See Section 32.11, "Starting a Kickstart Installation" for more information about the **noipv6** boot option, and the Knowledgebase article at https://access.redhat.com/solutions/1565723 for more information on disabling IPv6 system-wide.

- **--vlanid=** – Specifies virtual LAN ID number (802.1q tag).
- **--bondslaves=** – Specifies which network interfaces will be bonded as a comma-separated list.
- **--bondopts=** – a list of optional parameters for a bonded interface, which is specified using the **--bondslaves=** and **--device=** options. Options in this list must be separated by commas (",") or semicolons ("**;**"). If an option itself contains a comma, use a semicolon to separate the options. For example:

```
network --bondopts=mode=active-backup,balance-rr;primary=eth1
```

Available optional parameters are listed in the *Working with Kernel Modules* chapter of the Red Hat Enterprise Linux Deployment Guide .



IMPORTANT

The **--bondopts=mode=** parameter only supports full mode names such as **balance-rr** or **broadcast**, not their numerical representations such as **0** or **3**.

part or **partition** (required for installs, ignored for upgrades)

Creates a partition on the system.

If more than one Red Hat Enterprise Linux installation exists on the system on different partitions, the installation program prompts the user and asks which installation to upgrade.

WARNING

All partitions created are formatted as part of the installation process unless **--noformat** and **--onpart** are used.

IMPORTANT

If you select text mode for a kickstart installation, make sure that you specify choices for the partitioning, bootloader, and package selection options. These steps are automated in text mode, and **anaconda** cannot prompt you for missing information. If you do not provide choices for these options, **anaconda** will stop the installation process.

For a detailed example of **part** in action, refer to Section 32.4.1, “Advanced Partitioning Example”.

```
part|partition <mntpoint> --name=<name> --device=<device> --rule=<rule> [options]
```

- *<mntpoint>* – Where the partition is mounted. The value must be of one of the following forms:
  - **/*<path>***

    For example, /, **/usr**, **/home**
  - **swap**

    The partition is used as swap space.

    To determine the size of the swap partition automatically, use the **--recommended** option:

    ```
    swap --recommended
    ```

The size assigned will be effective but not precisely calibrated for your system.

To determine the size of the swap partition automatically but also allow extra space for your system to hibernate, use the **--hibernation** option:

```
swap --hibernation
```

The size assigned will be equivalent to the swap space assigned by **--recommended** plus the amount of RAM on your system.

For the swap sizes assigned by these commands, refer to Section 9.15.5, "Recommended Partitioning Scheme" for x86, AMD64, and Intel 64 Architecture and Section 16.17.5, "Recommended Partitioning Scheme" for IBM Power Systems servers.

IMPORTANT

Swap space recommendations were updated in Red Hat Enterprise Linux 6.3. Previously, systems with large amounts of RAM were assigned huge swap spaces. This delayed the Out-of-Memory Killer (**oom_kill**) in addressing critical memory shortages, even if a process was malfunctioning.

Consequently, if you are using an earlier version of Red Hat Enterprise Linux 6, **swap --recommended** will generate larger swap spaces than those described in the Recommended Partitioning Scheme, even on systems with large amounts of RAM. This may negate the need to allow extra space for hibernation.

However, these updated swap space values are nonetheless recommended for earlier versions of Red Hat Enterprise Linux 6 and can be set manually using the **swap --size=** option.

- **raid.*<id>***

  The partition is used for software RAID (refer to **raid**).

- **pv.*<id>***

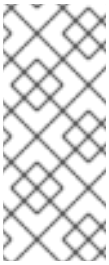
  The partition is used for LVM (refer to **logvol**).

- **--size=** – The minimum partition size in megabytes. Specify an integer value here such as **500** (do not include the unit).

IMPORTANT

If the **--size** value is too small, the installation will fail. Set the **--size** value as the minimum amount of space you require. For size recommendations, refer to Section 9.15.5, "Recommended Partitioning Scheme".

- **--grow** – Tells the partition to grow to fill available space (if any), or up to the maximum size setting.



NOTE

If you use **--grow=** without setting **--maxsize=** on a swap partition, **Anaconda** will limit the maximum size of the swap partition. For systems that have less than 2GB of physical memory, the imposed limit is twice the amount of physical memory. For systems with more than 2GB, the imposed limit is the size of physical memory plus 2GB.

- **--maxsize=** – The maximum partition size in megabytes when the partition is set to grow. Specify an integer value here such as **500** (do not include the unit).
- **--noformat** – Specifies that the partition should not be formatted, for use with the **--onpart** command.
- **--onpart=** or **--usepart=** – Specifies the device on which to place the partition. For example:

  ```
  partition /home --onpart=hda1
  ```

  puts **/home** on **/dev/hda1**.

  These options can also add a partition to a logical volume. For example:

  ```
  partition pv.1 --onpart=hda2
  ```

  The device must already exist on the system; the **--onpart** option will not create it.

- **--ondisk=** or **--ondrive=** – Forces the partition to be created on a particular disk. For example, **--ondisk=sdb** puts the partition on the second SCSI disk on the system.

  To specify a multipath device that does not use *logical volume management* (LVM), use the format **disk/by-id/dm-uuid-mpath-*WWID***, where *WWID* is the *world-wide identifier* for the device. For example, to specify a disk with WWID **2416CD96995134CA5D787F00A5AA11017**, use:

  ```
  part / --fstype=ext3 --grow --asprimary --size=100 --ondisk=disk/by-id/dm-uuid-mpath-
  2416CD96995134CA5D787F00A5AA11017
  ```

  Multipath devices that use LVM are not assembled until after **anaconda** has parsed the kickstart file. Therefore, you cannot specify these devices in the format **dm-uuid-mpath**. Instead, to specify a multipath device that uses LVM, use the format **disk/by-id/scsi-*WWID***, where *WWID* is the *world-wide identifier* for the device. For example, to specify a disk with WWID **58095BEC5510947BE8C0360F604351918**, use:

  ```
  part / --fstype=ext3 --grow --asprimary --size=100 --ondisk=disk/by-id/scsi-
  58095BEC5510947BE8C0360F604351918
  ```

WARNING

Never specify multipath devices by device names like **mpatha**. Device names like **mpatha** are not specific to a particular disk. The disk named **/dev/mpatha** during installation might not be the one that you expect it to be. Therefore, the **clearpart** command could target the wrong disk.

- **--asprimary** – Forces automatic allocation of the partition as a primary partition, or the partitioning fails.
- **--type=** (replaced by **fstype**) – This option is no longer available. Use **fstype**.
- **--fsoptions** – Specifies a free form string of options to be used when mounting the filesystem. This string will be copied into the **/etc/fstab** file of the installed system and should be enclosed in quotes.
- **--fsprofile** – Specifies a *usage type* to be passed to the program that makes a filesystem on this partition. A usage type defines a variety of tuning parameters to be used when making a filesystem. For this option to work, the filesystem must support the concept of usage types and there must be a configuration file that lists valid types. For ext2, ext3, and ext4, this configuration file is **/etc/mke2fs.conf**.
- **--fstype=** – Sets the file system type for the partition. Valid values are **xfs**, **ext2**, **ext3**, **ext4**, **swap**, **vfat**, **hfs**, and **efi**.
- **--recommended** – Determine the size of the partition automatically.
- **--onbiosdisk** – Forces the partition to be created on a particular disk as discovered by the BIOS.
- **--encrypted** – Specifies that this partition should be encrypted, using the passphrase provided in the **--passphrase** option. If you do not specify a passphrase, **anaconda** uses the default, system-wide passphrase set with the **autopart --passphrase** command, or stops the installation and prompts you to provide a passphrase if no default is set.
- **--cipher=** – Specifies which type of encryption will be used if the **anaconda** default aes-xts-plain64 is not satisfactory. You must use this option together with the **--encrypted** option; by itself it has no effect. Available types of encryption are listed in the *Red Hat Enterprise Linux Security Guide*, but Red Hat strongly recommends using either aes-xts-plain64 or aes-cbc-essiv:sha256.
- **--passphrase=** – Specifies the passphrase to use when encrypting this partition. You must use this option together with the **--encrypted** option; by itself it has no effect.
- **--escrowcert=*URL_of_X.509_certificate*** – Store data encryption keys of all encrypted partitions as files in **/root**, encrypted using the X.509 certificate from the URL specified with *URL_of_X.509_certificate*. The keys are stored as a separate file for each encrypted partition. This option is only meaningful if **--encrypted** is specified.
- **--backuppassphrase=** – Add a randomly-generated passphrase to each encrypted partition. Store these passphrases in separate files in **/root**, encrypted using the X.509 certificate specified with **--escrowcert**. This option is only meaningful if **--escrowcert** is

specified.

- **--label=** – assign a label to an individual partition.

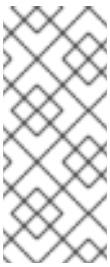
NOTE

If partitioning fails for any reason, diagnostic messages appear on virtual console 3.

poweroff (optional)

Shut down and power off the system after the installation has successfully completed. Normally during a manual installation, anaconda displays a message and waits for the user to press a key before rebooting. During a kickstart installation, if no completion method is specified, the **halt** option is used as default.

The **poweroff** option is equivalent to the **shutdown -p** command.



NOTE

The **poweroff** option is highly dependent on the system hardware in use. Specifically, certain hardware components such as the BIOS, APM (advanced power management), and ACPI (advanced configuration and power interface) must be able to interact with the system kernel. Contact your manufacturer for more information on you system's APM/ACPI abilities.

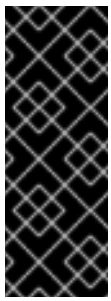
For other completion methods, refer to the **halt**, **reboot**, and **shutdown** kickstart options.

raid (optional)

Assembles a software RAID device. This command is of the form:

```
raid <mntpoint> --level=<level> --device=<mddevice> <partitions*>
```

- *<mntpoint>* – Location where the RAID file system is mounted. If it is /, the RAID level must be 1 unless a boot partition (/**boot**) is present. If a boot partition is present, the /**boot** partition must be level 1 and the root (/) partition can be any of the available types. The *<partitions*>* (which denotes that multiple partitions can be listed) lists the RAID identifiers to add to the RAID array.



IMPORTANT

If a RAID device has been prepared and has not been reformatted during installation, ensure that the RAID metadata version is **0.90** if you intend to put the /**boot** and **PReP** partitions on the RAID device.

The default Red Hat Enterprise Linux 6 **mdadm** metadata version is not supported for the boot device.

- **--level=** – RAID level to use (0, 1, or 5).
- **--device=** – Name of the RAID device to use (such as md0 or md1). RAID devices range from md0 to md15, and each may only be used once.

- **--spares=** – Specifies the number of spare drives allocated for the RAID array. Spare drives are used to rebuild the array in case of drive failure.

- **--fsprofile** – Specifies a *usage type* to be passed to the program that makes a filesystem on this partition. A usage type defines a variety of tuning parameters to be used when making a filesystem. For this option to work, the filesystem must support the concept of usage types and there must be a configuration file that lists valid types. For ext2, ext3, and ext4, this configuration file is **/etc/mke2fs.conf**.

- **--fstype=** – Sets the file system type for the RAID array. Valid values are **xfs**, **ext2**, **ext3**, **ext4**, **swap**, **vfat**, and **hfs**.

- **--fsoptions=** – Specifies a free form string of options to be used when mounting the filesystem. This string will be copied into the /etc/fstab file of the installed system and should be enclosed in quotes.

- **--noformat** – Use an existing RAID device and do not format the RAID array.

- **--useexisting** – Use an existing RAID device and reformat it.

- **--encrypted** – Specifies that this RAID device should be encrypted, using the passphrase provided in the **--passphrase** option. If you do not specify a passphrase, **anaconda** uses the default, system-wide passphrase set with the **autopart --passphrase** command, or stops the installation and prompts you to provide a passphrase if no default is set.

- **--cipher=** – Specifies which type of encryption will be used if the **anaconda** default aes-xts-plain64 is not satisfactory. You must use this option together with the **--encrypted** option; by itself it has no effect. Available types of encryption are listed in the *Red Hat Enterprise Linux Security Guide*, but Red Hat strongly recommends using either aes-xts-plain64 or aes-cbc-essiv:sha256.

- **--passphrase=** – Specifies the passphrase to use when encrypting this RAID device. You must use this option together with the **--encrypted** option; by itself it has no effect.

- **--escrowcert=*URL_of_X.509_certificate*** – Store the data encryption key for this device in a file in **/root**, encrypted using the X.509 certificate from the URL specified with *URL_of_X.509_certificate*. This option is only meaningful if **--encrypted** is specified.

- **--backuppassphrase=** – Add a randomly-generated passphrase to this device. Store the passphrase in a file in **/root**, encrypted using the X.509 certificate specified with **--escrowcert**. This option is only meaningful if **--escrowcert** is specified.

The following example shows how to create a RAID level 1 partition for /, and a RAID level 5 for **/usr**, assuming there are three SCSI disks on the system. It also creates three swap partitions, one on each drive.

```
part raid.01 --size=60 --ondisk=sda
part raid.02 --size=60 --ondisk=sdb
part raid.03 --size=60 --ondisk=sdc
```

```
part swap --size=128 --ondisk=sda
part swap --size=128 --ondisk=sdb
part swap --size=128 --ondisk=sdc
```

```
part raid.11 --size=1 --grow --ondisk=sda
part raid.12 --size=1 --grow --ondisk=sdb
part raid.13 --size=1 --grow --ondisk=sdc
```

```
raid / --level=1 --device=md0 raid.01 raid.02 raid.03
raid /usr --level=5 --device=md1 raid.11 raid.12 raid.13
```

For a detailed example of **raid** in action, refer to Section 32.4.1, "Advanced Partitioning Example".

reboot (optional)

Reboot after the installation is successfully completed (no arguments). Normally, kickstart displays a message and waits for the user to press a key before rebooting.

The **reboot** option is equivalent to the **shutdown -r** command.

Specify **reboot** to automate installation fully when installing in cmdline mode on System z.

For other completion methods, refer to the **halt**, **poweroff**, and **shutdown** kickstart options.

The **halt** option is the default completion method if no other methods are explicitly specified in the kickstart file.

NOTE

Use of the **reboot** option *may* result in an endless installation loop, depending on the installation media and method.

repo (optional)

Configures additional yum repositories that may be used as sources for package installation. Multiple repo lines may be specified.

```
repo --name=<repoid> [--baseurl=<url>| --mirrorlist=<url>]
```

- **--name=** – The repo id. This option is required.
- **--baseurl=** – The URL for the repository. The variables that may be used in yum repo config files are not supported here. You may use one of either this option or --mirrorlist, not both.
- **--mirrorlist=** – The URL pointing at a list of mirrors for the repository. The variables that may be used in yum repo config files are not supported here. You may use one of either this option or --baseurl, not both.

IMPORTANT

Repositories used for installation must be stable. The installation may fail if a repository is modified before the installation concludes.

rootpw (required)

Sets the system's root password to the *<password>* argument.

```
rootpw [--iscrypted] <password>
```

- **--iscrypted** – If this is present, the password argument is assumed to already be encrypted. To create an encrypted password, use the following command:

```
python -c 'import crypt; print(crypt.crypt("My Password"))'
```

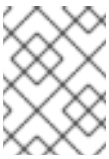
This will create a sha512 crypt of your password.

selinux (optional)

Sets the state of SELinux on the installed system. SELinux defaults to enforcing in anaconda.

```
selinux [--disabled|--enforcing|--permissive]
```

- **--enforcing** – Enables SELinux with the default targeted policy being enforced.



NOTE

If the **selinux** option is not present in the kickstart file, SELinux is enabled and set to **--enforcing** by default.

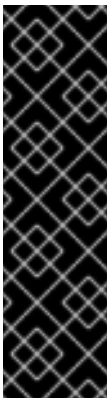
- **--permissive** – Outputs warnings based on the SELinux policy, but does not actually enforce the policy.
- **--disabled** – Disables SELinux completely on the system.

For more information regarding SELinux for Red Hat Enterprise Linux, refer to the *Red Hat Enterprise Linux 6.9 Deployment Guide*.

services (optional)

Modifies the default set of services that will run under the default runlevel. The list of disabled services is processed before the list of enabled services. Therefore, if a service appears on both lists, it is enabled.

- **--disabled** – Disable the services given in the comma separated list.
- **--enabled** – Enable the services given in the comma separated list.



IMPORTANT

Do not include spaces in the list of services. If you do, kickstart will enable or disable only the services up to the first space. For example:

services --disabled auditd, cups,smartd, nfslock

will disable only the **auditd** service. To disable all four services, this entry should include no spaces between services:

services --disabled auditd,cups,smartd,nfslock

shutdown (optional)

Shut down the system after the installation has successfully completed. During a kickstart installation, if no completion method is specified, the **halt** option is used as default.

The **shutdown** option is equivalent to the **shutdown** command.

For other completion methods, refer to the **halt**, **poweroff**, and **reboot** kickstart options.

skipx (optional)

If present, X is not configured on the installed system.

IMPORTANT

If you install a display manager among your package selection options, this package will create an X configuration, and the installed system will default to run level 5. The effect of the **skipx** option will be overridden.

sshpw (optional)

During installation, you can interact with **anaconda** and monitor its progress over an SSH connection. Use the **sshpw** command to create temporary accounts through which to log on. Each instance of the command creates a separate account that exists only in the installation environment. These accounts are not transferred to the installed system.

```
sshpw --username=<name> <password> [--iscrypted|--plaintext] [--lock]
```

- **--username** – Provides the name of the user. This option is required.
- **--iscrypted** – Specifies that the password is already encrypted.
- **--plaintext** – Specifies that the password is in plain text and not encrypted.
- **--lock** – If this is present, the new user account is locked by default. That is, the user will not be able to login from the console.

IMPORTANT

By default, the **ssh** server is not started during installation. To make **ssh** available during installation, boot the system with the kernel boot option ***sshd=1***. Refer to Section 28.2.3, "Enabling Remote Access with ssh" for details of how to specify this kernel option at boot time.

NOTE

If you want to disable root **ssh** access to your hardware during installation, run:

```
sshpw --username=root --lock
```

text (optional)

Perform the kickstart installation in text mode. Kickstart installations are performed in graphical mode by default.

IMPORTANT

If you select text mode for a kickstart installation, make sure that you specify choices for the partitioning, bootloader, and package selection options. These steps are automated in text mode, and **anaconda** cannot prompt you for missing information. If you do not provide choices for these options, **anaconda** will stop the installation process.

timezone (required)

Sets the system time zone to *<timezone>* which may be any of the time zones listed in the **/usr/share/zoneinfo** directory.

```
timezone [--utc] <timezone>
```

- **--utc** – If present, the system assumes the hardware clock is set to UTC (Greenwich Mean) time.

unsupported_hardware (optional)

Tells the installer to suppress the **Unsupported Hardware Detected** alert. If this command is not included and unsupported hardware is detected, the installation will stall at this alert.

upgrade (optional)

Tells the system to upgrade an existing system rather than install a fresh system. You must specify one of **cdrom**, **harddrive**, **nfs**, or **url** (for FTP, HTTP, and HTTPS) as the location of the installation tree. Refer to **install** for details.

user (optional)

Creates a new user on the system.

```
user --name=<username> [--groups=<list>] [--homedir=<homedir>] [--password=<password>] [--iscrypted] [--shell=<shell>] [--uid=<uid>]
```

- **--name=** – Provides the name of the user. This option is required.
- **--groups=** – In addition to the default group, a comma separated list of group names the user should belong to. The groups must exist before the user account is created.
- **--homedir=** – The home directory for the user. If not provided, this defaults to /home/*<username>*.
- **--password=** – The new user's password. If not provided, the account will be locked by default.
- **--iscrypted=** – Is the password provided by --password already encrypted or not?
- **--shell=** – The user's login shell. If not provided, this defaults to the system default.
- **--uid=** – The user's UID. If not provided, this defaults to the next available non-system UID.

vnc (optional)

Allows the graphical installation to be viewed remotely via VNC. This method is usually preferred over text mode, as there are some size and language limitations in text installs. With no options, this

command will start a VNC server on the machine with no password and will print out the command that needs to be run to connect a remote machine.

```
vnc [--host=<hostname>] [--port=<port>] [--password=<password>]
```

- **--host=** – Instead of starting a VNC server on the install machine, connect to the VNC viewer process listening on the given hostname.
- **--port=** – Provide a port that the remote VNC viewer process is listening on. If not provided, anaconda will use the VNC default.
- **--password=** – Set a password which must be provided to connect to the VNC session. This is optional, but recommended.

volgroup (optional)

Use to create a Logical Volume Management (LVM) group with the syntax:

```
volgroup <name> <partition> [options]
```

IMPORTANT

Do not use the dash ("**-**") character in logical volume or volume group names when installing Red Hat Enterprise Linux using Kickstart. If you do, the installation will finish normally, but the character will be removed from all newly created volume and volume group names. For example, if you create a volume group named **volgrp-01**, its name will be changed to **volgrp01**.

This limitation only applies to new installations. If you are upgrading or reinstalling an existing installation and use the **--noformat** option described below, dashes used in volume and volume group names will be preserved.

Create the partition first, create the logical volume group, and then create the logical volume. For example:

```
part pv.01 --size 3000
volgroup myvg pv.01
logvol / --vgname=myvg --size=2000 --name=rootvol
```

For a detailed example of **volgroup** in action, refer to Section 32.4.1, "Advanced Partitioning Example".

The options are as follows:

- **--noformat** – Use an existing volume group and do not format it.
- **--useexisting** - Use an existing volume group and reformat it. If you use this option, do not specify a *partition*. For example:

  ```
  volgroup rhel00 --useexisting --noformat
  ```

- **--pesize=** – Set the size of the physical extents. The default size for Kickstart installations is 4 MiB.

- **--reserved-space=** – Specify an amount of space to leave unused in a volume group, in megabytes. Only usable when creating a new volume group.
- **--reserved-percent=** – Specify a percentage of total volume group space to leave unused. Only usable when creating a new volume group.

NOTE

The **--reserved-space=** and **--reserved-percent=** options allow you to leave a part of the total volume group space unused by any volumes. This allows you to reserve space for LVM snapshots, even if the **logvol --grow** command is used during partitioning.

winbind (optional)

Configures the system to connect to a Windows Active Directory or a Windows domain controller. User information from the specified directory or domain controller can then be accessed and server authentication options can be configured.

- **--enablewinbind** – Enable winbind for user account configuration.
- **--disablewinbind** – Disable winbind for user account configuration.
- **--enablewinbindauth** – Enable windbindauth for authentication.
- **--disablewinbindauth** – Disable windbindauth for authentication.
- **--enablewinbindoffline** – Configures winbind to allow offline login.
- **--disablewinbindoffline** – Configures winbind to prevent offline login.
- **--enablewinbindusedefaultdomain** – Configures winbind to assume that users with no domain in their usernames are domain users.
- **--disablewinbindusedefaultdomain** – Configures winbind to assume that users with no domain in their usernames are not domain users.

xconfig (optional)

Configures the **X Window System**. If you install the **X Window System** with a kickstart file that does not include the **xconfig** command, you must provide the **X** configuration manually during installation.

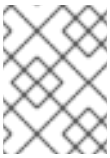
Do not use this command in a kickstart file that does not install the **X Window System**.

- **--driver** – Specify the X driver to use for the video hardware.
- **--videoram=** – Specifies the amount of video RAM the video card has.
- **--defaultdesktop=** – Specify either GNOME or KDE to set the default desktop (assumes that GNOME Desktop Environment and/or KDE Desktop Environment has been installed through **%packages**).
- **--startxonboot** – Use a graphical login on the installed system.

zerombr (optional)

If **zerombr** is specified any invalid partition tables found on disks are initialized. This destroys all of the contents of disks with invalid partition tables. This command is required when performing an unattended installation on a system with previously initialized disks.

Specific to System z: If **zerombr** is specified, any DASD visible to the installer which is not already low-level formatted gets automatically low-level formatted with **dasdfmt**. The command also prevents user choice during interactive installations. If **zerombr** is not specified and there is at least one unformatted DASD visible to the installer, a non-interactive kickstart installation will exit unsuccessfully. If **zerombr** is not specified and there is at least one unformatted DASD visible to the installer, an interactive installation exits if the user does not agree to format all visible and unformatted DASDs. To circumvent this, only activate those DASDs that you will use during installation. You can always add more DASDs after installation is complete.



NOTE

That this command was previously specified as **zerombr yes**. This form is now deprecated; you should now simply specify **zerombr** in your kickstart file instead.

zfcp (optional)

Define a Fiber channel device (IBM System z).

zfcp [--devnum=*<devnum>*] [--wwpn=*<wwpn>*] [--fcplun=*<fcplun>*]

%include (optional)

Use the **%include** ***/path/to/file*** command to include the contents of another file in the kickstart file as though the contents were at the location of the **%include** command in the kickstart file.

32.4.1. Advanced Partitioning Example

The following is a single, integrated example showing the **clearpart**, **raid**, **part**, **volgroup**, and **logvol** kickstart options in action:

```
clearpart --drives=hda,hdc
zerombr
# Raid 1 IDE config
part raid.11    --size 1000     --asprimary     --ondrive=hda
part raid.12    --size 1000     --asprimary     --ondrive=hda
part raid.13    --size 2000     --asprimary     --ondrive=hda
part raid.14    --size 8000                     --ondrive=hda
part raid.15    --size 16384 --grow             --ondrive=hda
part raid.21    --size 1000     --asprimary     --ondrive=hdc
part raid.22    --size 1000     --asprimary     --ondrive=hdc
part raid.23    --size 2000     --asprimary     --ondrive=hdc
part raid.24    --size 8000                     --ondrive=hdc
part raid.25    --size 16384 --grow             --ondrive=hdc

# You can add --spares=x
raid /          --fstype ext3 --device md0 --level=RAID1 raid.11 raid.21
raid /safe      --fstype ext3 --device md1 --level=RAID1 raid.12 raid.22
raid swap       --fstype swap --device md2 --level=RAID1 raid.13 raid.23
raid /usr       --fstype ext3 --device md3 --level=RAID1 raid.14 raid.24
raid pv.01      --fstype ext3 --device md4 --level=RAID1 raid.15 raid.25
```

```
# LVM configuration so that we can resize /var and /usr/local later
volgroup sysvg pv.01
logvol /var             --vgname=sysvg  --size=8000     --name=var
logvol /var/freespace   --vgname=sysvg  --size=8000     --name=freespacetouse
logvol /usr/local       --vgname=sysvg  --size=1 --grow --name=usrlocal
```

This advanced example implements LVM over RAID, as well as the ability to resize various directories for future growth.

32.5. PACKAGE SELECTION

WARNING

You can use a kickstart file to install every available package by specifying * in the **%packages** section. Red Hat does not support this type of installation.

In previous releases of Red Hat Enterprise Linux, this functionality was provided by **@Everything**, but this option is not included in Red Hat Enterprise Linux 6.

Use the **%packages** command to begin a kickstart file section that lists the packages you would like to install (this is for installations only, as package selection during upgrades is not supported).

You can specify packages by *group* or by their package names. The installation program defines several groups that contain related packages. Refer to the ***variant*/repodata/comps-*.xml** file on the Red Hat Enterprise Linux 6.9 Installation DVD for a list of groups. Each group has an id, user visibility value, name, description, and package list. If the group is selected for installation, the packages marked **mandatory** in the package list are always installed, the packages marked **default** are installed if they are not specifically excluded elsewhere, and the packages marked **optional** must be specifically included elsewhere even when the group is selected.

Specify groups, one entry to a line, starting with an **@** symbol, a space, and then the full group name or group id as given in the **comps.xml** file. For example:

```
%packages
@X Window System
@Desktop
@Sound and Video
```

Note that the **Core** and **Base** groups are always selected by default, so it is not necessary to specify them in the **%packages** section.

WARNING

When performing a minimal installation using the **@Core** group, the firewall (**iptables**/**ip6tables**) will not be configured on the installed system. This presents a security risk. To work around this issue, add the authconfig and system-config-firewall-base packages to your package selection as described below. The firewall will be configured properly if these packages are present.

A minimal installation's **%packages** section which will also configure the firewall will look similar to the following:

```
%packages
@Core
authconfig
system-config-firewall-base
```

See the Red Hat Customer Portal for details.

Specify individual packages by name, one entry to a line. You can use asterisks as wildcards to glob package names in entries. For example:

```
sqlite
curl
aspell
docbook*
```

The **docbook*** entry includes the packages docbook-dtds, docbook-simple, docbook-slides and others that match the pattern represented with the wildcard.

Use a leading dash to specify packages or groups to exclude from the installation. For example:

```
-@ Graphical Internet
-autofs
-ipa*fonts
```

IMPORTANT

To install a 32-bit package on a 64-bit system, you will need to append the package name with the 32-bit architecture the package was built for. For example:

```
glibc.i686
```

Using a kickstart file to install every available package by specifying * will introduce package and file conflicts onto the installed system. Packages known to cause such problems are assigned to the **@Conflicts (*variant*)** group, where *variant* is **Client**, **ComputeNode**, **Server** or **Workstation**. If you specify * in a kickstart file, be sure to exclude **@Conflicts (*variant*)** or the installation will fail:

```
*
-@Conflicts (Server)
```

Note that Red Hat does not support the use of ***** in a kickstart file, even if you exclude **@Conflicts (*variant*)**.

The section must end with the **%end** command.

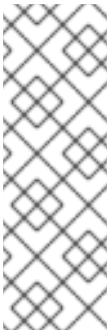
The following options are available for the **%packages** option:

--nobase

Do not install the **@Base** group. Use this option to perform a minimal installation, for example, for a single-purpose server or desktop appliance.

--nocore

Disables installation of the **@Core** package group which is otherwise always installed by default. Disabling the **@Core** package group with **--nocore** should be only used for creating lightweight containers; installing a desktop or server system with **--nocore** will result in an unusable system.



NOTE

- Using **-@Core** to exclude packages in the **@Core** package group does not work. The only way to exclude the **@Core** package group is with the **--nocore** option.
- The **@Core** package group is defined as a minimal set of packages needed for installing a working system. It is not related in any way to core packages as defined in the Package Manifest and Scope of Coverage Details.

--ignoredeps

The --ignoredeps option has been deprecated. Dependencies are resolved automatically every time now.

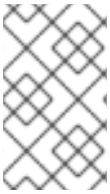
--ignoremissing

Ignore the missing packages and groups instead of halting the installation to ask if the installation should be aborted or continued. For example:

```
%packages --ignoremissing
```

32.6. PRE-INSTALLATION SCRIPT

You can add commands to run on the system immediately after the **ks.cfg** has been parsed. This section must be placed towards the end of the kickstart file, after the kickstart commands described in Section 32.4, "Kickstart Options", and must start with the **%pre** command and end with the **%end** command. If your kickstart file also includes a **%post** section, the order of the **%pre** and **%post** sections does not matter. See Section 32.8, "Kickstart Examples" for example configuration files.



NOTE

The pre-installation script section of kickstart *cannot* manage multiple install trees or source media. This information must be included for each created ks.cfg file, as the pre-installation script occurs during the second stage of the installation process.

You can access the network in the **%pre** section; however, *name service* has not been configured at this point, so only IP addresses work.

Only the most commonly used commands are available in the pre-installation environment:

arping, **awk**, **basename**, **bash**, **bunzip2**, **bzcat**, **cat**, **chattr**, **chgrp**, **chmod**, **chown**, **chroot**, **chvt**, **clear**, **cp**, **cpio**, **cut**, **date**, **dd**, **df**, **dirname**, **dmesg**, **du**, **e2fsck**, **e2label**, **echo**, **egrep**, **eject**, **env**, **expr**, **false**, **fdisk**, **fgrep**, **find**, **fsck**, **fsck.ext2**, **fsck.ext3**, **ftp**, **grep**, **gunzip**, **gzip**, **hdparm**, **head**, **hostname**, **hwclock**, **ifconfig**, **insmod**, **ip**, **ipcalc**, **kill**, **killall**, **less**, **ln**, **load_policy**, **login**, **losetup**, **ls**, **lsattr**, **lsmod**, **lvm**, **md5sum**, **mkdir**, **mke2fs**, **mkfs.ext2**, **mkfs.ext3**, **mknod**, **mkswap**, **mktemp**, **modprobe**, **more**, **mount**, **mt**, **mv**, **nslookup**, **openvt**, **pidof**, **ping**, **ps**, **pwd**, **readlink**, **rm**, **rmdir**, **rmmod**, **route**, **rpm**, **sed**, **sh**, **sha1sum**, **sleep**, **sort**, **swapoff**, **swapon**, **sync**, **tail**, **tar**, **tee**, **telnet**, **top**, **touch**, **true**, **tune2fs**, **umount**, **uniq**, **vconfig**, **vi**, **wc**, **wget**, **wipefs**, **xargs**, **zcat**.

NOTE

The pre-install script is not run in the change root environment.

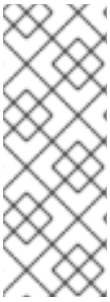
--interpreter ***/usr/bin/python***

Allows you to specify a different scripting language, such as Python. Replace */usr/bin/python* with the scripting language of your choice.

32.7. POST-INSTALLATION SCRIPT

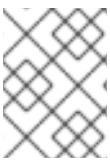
You have the option of adding commands to run on the system once the installation is complete. This section must be placed towards the end of the kickstart file, after the kickstart commands described in Section 32.4, "Kickstart Options", and must start with the **%post** command and end with the **%end** command. If your kickstart file also includes a **%pre** section, the order of the **%pre** and **%post** sections does not matter. See Section 32.8, "Kickstart Examples" for example configuration files.

This section is useful for functions such as installing additional software and configuring an additional nameserver.



NOTE

If you configured the network with static IP information, including a nameserver, you can access the network and resolve IP addresses in the **%post** section. If you configured the network for DHCP, the **/etc/resolv.conf** file has not been completed when the installation executes the **%post** section. You can access the network, but you can not resolve IP addresses. Thus, if you are using DHCP, you must specify IP addresses in the **%post** section.



NOTE

The post-install script is run in a chroot environment; therefore, performing tasks such as copying scripts or RPMs from the installation media do not work.

--nochroot

Allows you to specify commands that you would like to run outside of the chroot environment.

The following example copies the file **/etc/resolv.conf** to the file system that was just installed.

```
%post --nochroot
cp /etc/resolv.conf /mnt/sysimage/etc/resolv.conf
```

--interpreter *_/usr/bin/python_*

Allows you to specify a different scripting language, such as Python. Replace */usr/bin/python* with the scripting language of your choice.

--log *_/path/to/logfile_*

Logs the output of the post-install script. Note that the path of the log file must take into account whether or not you use the **--nochroot** option. For example, without **--nochroot**:

```
%post --log=/root/ks-post.log
```

with **--nochroot**:

```
%post --nochroot --log=/mnt/sysimage/root/ks-post.log
```

32.8. KICKSTART EXAMPLES

32.8.1. Set host name interactively during installation

The following example demonstrates how to interactively set the system's host name during installation. The **%pre** script asks to enter a host name for the installed system. The **%post** script configures the network according to the user's input.

```
%pre
chvt 3
exec </dev/tty3> /dev/tty3
clear
## Query for hostname, then write it to 'network' file
read -p "
What is my hostname (FQDN)? (This will be set on eth0)
" NAME /dev/tty3 2>&1
echo "NETWORKING=yes" > network
echo "HOSTNAME=${NAME}" >> network
echo "DEVICE=eth0" > ifcfg-eth0
echo "BOOTPROTO=dhcp" >> ifcfg-eth0
echo "ONBOOT=yes" >> ifcfg-eth0
echo "DHCP_HOSTNAME=${NAME} " >> ifcfg-eth0
cat ifcfg-eth0
chvt 1
exec < /dev/tty1 > /dev/tty1
%end

%post --nochroot
# bring in hostname collected from %pre, then source it
```

```
cp -Rvf network /mnt/sysimage/etc/sysconfig/network
# Set-up eth0 with hostname
cp ifcfg-eth0 /mnt/sysimage/etc/sysconfig/network-scripts/ifcfg-eth0
# force hostname change
/mnt/sysimage/bin/hostname $HOSTNAME
%end
```

32.8.2. Registering and Then Mounting an NFS Share

Register the system to a Red Hat Subscription Management server (in this example, a local Subscription Asset Manager server):

```
%post --log=/root/ks-post.log
/usr/sbin/subscription-manager register --username=admin@example.com --password=secret --
serverurl=sam-server.example.com --org="Admin Group" --environment="Dev"
%end
```

Run a script named **runme** from an NFS share:

```
mkdir /mnt/temp
mount -o nolock 10.10.0.2:/usr/new-machines /mnt/temp
openvt -s -w -- /mnt/temp/runme
umount /mnt/temp
```

NFS file locking is *not* supported while in kickstart mode, therefore **-o nolock** is required when mounting an NFS mount.

32.8.3. Registering a System in RHN Classic

The **rhnreg_ks** command is a utility for registering a system with the Red Hat Network. It is designed to be used in a non-interactive environment (a Kickstart style install, for example). All the information can be specified on the command line or standard input (stdin). This command should be used when you have created an activation key and you want to register a system using a key.

For details about using **rhnreg_ks** to automatically register your system, see the Knowledgebase article at https://access.redhat.com/solutions/876433.

32.8.4. Running subscription-manager as a Post-Install Script

The **subscription-manager** command-line script registers a system to a Red Hat Subscription Management server (Customer Portal Subscription Management, Subscription Asset Manager, or CloudForms System Engine). This script can also be used to assign or *attach* subscriptions automatically to the system that best-match that system.

When registering to the Customer Portal, use the Red Hat network login credentials. When registering to Subscription Asset Manager or CloudForms System Engine, use whatever user account was created by the local administrator.

Additional options can be used with the registration command to set a preferred service level for the system and to restrict updates and errata to a specific operating system version.

```
%post --log=/root/ks-post.log
/usr/sbin/subscription-manager register --username=admin@example.com --password=secret --
serverurl=sam-server.example.com --org="Admin Group" --environment="Dev" --
```

```
servicelevel=standard --release="6.6"
%end
```

For additional information about using **subscription-manager**, see the Knowledgebase article at https://access.redhat.com/solutions/748313.

32.8.5. Changing partition layout

The following example **%pre** script generates a different set of partitioning commands depending on whether the system has two drives or not.

```
%pre
#!/bin/sh
hds=""
mymedia=""
for file in /proc/ide/h* do
 mymedia=`cat $file/media`
 if [ $mymedia == "disk" ] ; then
  hds="$hds `basename $file`"
 fi
done
set $hds
numhd=`echo $#`
drive1=`echo $hds | cut -d' ' -f1`
drive2=`echo $hds | cut -d' ' -f2`
#Write out partition scheme based on whether there are 1 or 2 hard drives
if [ $numhd == "2" ] ; then
 #2 drives
 echo "#partitioning scheme generated in %pre for 2 drives" > /tmp/part-include
 echo "clearpart --all" >> /tmp/part-include
 echo "zerombr" >> /tmp/part-include
 echo "part /boot --fstype ext3 --size 75 --ondisk hda" >> /tmp/part-include
 echo "part / --fstype ext3 --size 1 --grow --ondisk hda" >> /tmp/part-include
 echo "part swap --recommended --ondisk $drive1" >> /tmp/part-include
 echo "part /home --fstype ext3 --size 1 --grow --ondisk hdb" >> /tmp/part-include
else
 #1 drive
 echo "#partitioning scheme generated in %pre for 1 drive" > /tmp/part-include
 echo "clearpart --all" >> /tmp/part-include
 echo "part /boot --fstype ext3 --size 75" >> /tmp/part-include
 echo "part swap --recommended" >> /tmp/part-include
 echo "part / --fstype ext3 --size 2048" >> /tmp/part-include
 echo "part /home --fstype ext3 --size 2048 --grow" >> /tmp/part-include
fi
%end
```

Adding the following line after the **%pre** script then instructs Kickstart to execute the commands that were generated by the script above:

```
%include /tmp/part-include
```

32.9. MAKING THE KICKSTART FILE AVAILABLE

A kickstart file must be placed in one of the following locations:

- On removable media, such as a floppy disk, optical disk, or USB flash drive
- On a hard drive
- On a network

Normally a kickstart file is copied to the removable media or hard drive, or made available on the network. The network-based approach is most commonly used, as most kickstart installations tend to be performed on networked computers.

The following section provides a more in-depth look at where the kickstart file may be placed.

32.9.1. Creating Kickstart Boot Media

If you want to modify boot media provided by Red Hat to include a Kickstart file and automatically load it during boot, follow the procedure below. Note that this procedure will only work on AMD and Intel systems (**x86** and **x86_64**). Additionally, this procedure requires the genisoimage and isomd5sum packages; these packages are available on Red Hat Enterprise Linux, but if you use a different system, you may need to adjust the commands used.

NOTE

Diskette-based booting is no longer supported in Red Hat Enterprise Linux. Installations must use CD-ROM or flash memory products for booting. However, the kickstart file may still reside on a diskette's top-level directory, and must be named **ks.cfg**. Separate boot media will be required.

Procedure 32.1. Including a Kickstart File on Boot Media

Before you start the procedure, make sure you have downloaded a boot ISO image (boot.iso or binary DVD) as described in Chapter 1, *Obtaining Red Hat Enterprise Linux* , and that you have created a working Kickstart file.

1. Mount the ISO image you have downloaded:

```
# mount /path/to/image.iso /mnt/iso
```

2. Extract the ISO image into a working directory somewhere in your system:

```
# cp -pRf /mnt/iso /tmp/workdir
```

3. Unmount the mounted image:

```
# umount /mnt/iso
```

4. The contents of the image is now placed in the **iso**/ directory in your working directory. Add your Kickstart file (**ks.cfg**) into the **iso**/ directory:

```
# cp /path/to/ks.cfg /tmp/workdir/iso
```

5. Open the **isolinux/isolinux.cfg** configuration file inside the **iso**/ directory. This file determines all the menu options which appear in the boot menu. A single menu entry is defined as the following:

```
label linux
  menu label ^Install or upgrade an existing system
  menu default
  kernel vmlinuz
  append initrd=initrd.img
```

Add the **ks=** boot option to the line beginning with **append**. The exact syntax depends on how you plan to boot the ISO image; for example, if you plan on booting from a CD or DVD, use **ks=cdrom:/ks.cfg**. A list of possible sources and the syntax used to configure them is available in Section 28.4, "Automating the Installation with Kickstart".

6. Use **genisoimage** in the **iso/** directory to create a new bootable ISO image with your changes included:

```
# genisoimage -U -r -v -T -J -joliet-long -V "RHEL-6.9" -volset "RHEL-6.9" -A "RHEL-6.9" -b
isolinux/isolinux.bin -c isolinux/boot.cat -no-emul-boot -boot-load-size 4 -boot-info-table -
eltorito-alt-boot -e images/efiboot.img -no-emul-boot -o ../NEWISO.iso .
```

This comand will create a file named **NEWISO.iso** in your working directory (one directory above the **iso/** directory).

IMPORTANT

If you use a disk label to refer to any device in your **isolinux.cfg** (e.g. **ks=hd:LABEL=RHEL-6.9/ks.cfg**, make sure that the label matches the label of the new ISO you are creating. Also note that in boot loader configuration, spaces in labels must be replaced with **\x20**.

7. Implant a md5 checksum into the new ISO image:

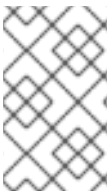
```
# implantisomd5 ../NEWISO.iso
```

After you finish the above procedure, your new image is ready to be turned into boot media. Refer to Chapter 2, *Making Media* for instructions.

To perform a pen-based flash memory kickstart installation, the kickstart file must be named **ks.cfg** and must be located in the flash memory's top-level directory. The kickstart file should be on a separate flash memory drive to the boot media.

To start the Kickstart installation, boot the system using the boot media you created, and use the **ks=** boot option to specify which device contains the USB drive. See Section 28.4, "Automating the Installation with Kickstart" for details about the **ks=** boot option.

See Section 2.2, "Making Minimal Boot Media" for instructions on creating boot USB media using the **rhel-*variant*-*version*-*architecture*-boot.iso** image file that you can download from the Software & Download Center of the Red Hat customer portal.



NOTE

Creation of USB flashdrives for booting is possible, but is heavily dependent on system hardware BIOS settings. Refer to your hardware manufacturer to see if your system supports booting to alternate devices.

32.9.2. Making the Kickstart File Available on the Network

Network installations using kickstart are quite common, because system administrators can quickly and easily automate the installation on many networked computers. In general, the approach most commonly used is for the administrator to have both a BOOTP/DHCP server and an NFS server on the local network. The BOOTP/DHCP server is used to give the client system its networking information, while the actual files used during the installation are served by the NFS server. Often, these two servers run on the same physical machine, but they are not required to.

Include the **ks** kernel boot option in the **append** line of a target in your **pxelinux.cfg/default** file to specify the location of a kickstart file on your network. The syntax of the **ks** option in a **pxelinux.cfg/default** file is identical to its syntax when used at the boot prompt. Refer to Section 32.11, "Starting a Kickstart Installation" for a description of the syntax and refer to Example 32.1, "Using the **ks** option in the **pxelinux.cfg/default** file" for an example of an **append** line.

If the **dhcpd.conf** file on the DHCP server is configured to point to **/var/lib/tftpboot/pxelinux.0** on the BOOTP server (whether on the same physical machine or not), systems configured to boot over the network can load the kickstart file and commence installation.

Example 32.1. Using the ks option in the pxelinux.cfg/default file

For example, if **foo.ks** is a kickstart file available on an NFS share at **192.168.0.200:/export/kickstart/**, part of your **pxelinux.cfg/default** file might include:

```
label 1
  kernel RHEL6/vmlinuz
  append initrd=RHEL6/initrd.img ramdisk_size=10000
ks=nfs:192.168.0.200:/export/kickstart/foo.ks
```

32.10. MAKING THE INSTALLATION TREE AVAILABLE

The kickstart installation must access an *installation tree*. An installation tree is a copy of the binary Red Hat Enterprise Linux DVD with the same directory structure.

If you are performing a DVD-based installation, insert the Red Hat Enterprise Linux installation DVD into the computer before starting the kickstart installation.

If you are performing a hard drive installation, make sure the ISO images of the binary Red Hat Enterprise Linux DVD are on a hard drive in the computer.

If you are performing a network-based (NFS, FTP or HTTP) installation, you must make the installation tree or ISO image available over the network. Refer to Section 4.1, "Preparing for a Network Installation" for details.

32.11. STARTING A KICKSTART INSTALLATION

IMPORTANT

Firstboot does not run after a system is installed from a Kickstart file unless a desktop and the X Window System were included in the installation and graphical login was enabled. Either specify a user with the **user** option in the Kickstart file before installing additional systems from it (refer to Section 32.4, "Kickstart Options" for details) or log into the installed system with a virtual console as root and add users with the **adduser** command.

To begin a kickstart installation, you must boot the system from boot media you have made or the Red Hat Enterprise Linux DVD, and enter a special boot command at the boot prompt. The installation program looks for a kickstart file if the **ks** command line argument is passed to the kernel.

DVD and local storage

The **linux ks=** command also works if the **ks.cfg** file is located on a vfat or ext2 file system on local storage and you boot from the Red Hat Enterprise Linux DVD.

With Driver Disk

If you need to use a driver disk with kickstart, specify the **dd** option as well. For example, if installation requires a kickstart file on a local hard drive and also requires a driver disk, boot the system with:

```
linux ks=hd:partition:/path/ks.cfg dd
```

Boot CD-ROM

If the kickstart file is on a boot CD-ROM as described in Section 32.9.1, "Creating Kickstart Boot Media", insert the CD-ROM into the system, boot the system, and enter the following command at the **boot:** prompt (where **ks.cfg** is the name of the kickstart file):

```
linux ks=cdrom:/ks.cfg
```

Other options to start a kickstart installation are as follows:

askmethod

Prompt the user to select an installation source, even if a Red Hat Enterprise Linux installation DVD is detected on the system.

asknetwork

Prompt for network configuration in the first stage of installation regardless of installation method.

autostep

Make kickstart non-interactive. Used for debugging and to generate screenshots. This option should not be used when deploying a system because it may disrupt package installation.

debug

Start up pdb immediately.

dd

Use a driver disk.

dhcpclass=*<class>*

Sends a custom DHCP vendor class identifier. ISC's dhcpcd can inspect this value using "option vendor-class-identifier".

dns=*<dns>*

Comma separated list of nameservers to use for a network installation.

driverdisk

Same as 'dd'.

expert

Turns on special features:

- allows partitioning of removable media
- prompts for a driver disk

gateway=*<gw>*

Gateway to use for a network installation.

graphical

Force graphical install. Required to have ftp/http use GUI.

isa

Prompt user for ISA devices configuration.

ip=*<ip>*

IP to use for a network installation, use 'dhcp' for DHCP.

ipv6=auto, ipv6=dhcp

IPv6 configuration for the device. Use **auto** for automatic configuration (SLAAC, SLAAC with DHCPv6), or **dhcp** for DHCPv6 only configuration (no router advertisements).

keymap=*<keymap>*

Keyboard layout to use. Valid layouts include:

- **be-latin1** – Belgian
- **bg_bds-utf8** – Bulgarian
- **bg_pho-utf8** – Bulgarian (Phonetic)
- **br-abnt2** – Brazilian (ABNT2)
- **cf** – French Canadian
- **croat** – Croatian
- **cz-us-qwertz** – Czech
- **cz-lat2** – Czech (qwerty)

- **de** – German
- **de-latin1** – German (latin1)
- **de-latin1-nodeadkeys** – German (latin1 without dead keys)
- **dvorak** – Dvorak
- **dk** – Danish
- **dk-latin1** – Danish (latin1)
- **es** – Spanish
- **et** – Estonian
- **fi** – Finnish
- **fi-latin1** – Finnish (latin1)
- **fr** – French
- **fr-latin9** – French (latin9)
- **fr-latin1** – French (latin1)
- **fr-pc** – French (pc)
- **fr_CH** – Swiss French
- **fr_CH-latin1** – Swiss French (latin1)
- **gr** – Greek
- **hu** – Hungarian
- **hu101** – Hungarian (101 key)
- **is-latin1** – Icelandic
- **it** – Italian
- **it-ibm** – Italian (IBM)
- **it2** – Italian (it2)
- **jp106** – Japanese
- **ko** – Korean
- **la-latin1** – Latin American
- **mk-utf** – Macedonian
- **nl** – Dutch
- **no** – Norwegian

- **pl2** – Polish
- **pt-latin1** – Portuguese
- **ro** – Romanian
- **ru** – Russian
- **sr-cy** – Serbian
- **sr-latin** – Serbian (latin)
- **sv-latin1** – Swedish
- **sg** – Swiss German
- **sg-latin1** – Swiss German (latin1)
- **sk-qwerty** – Slovak (qwerty)
- **slovene** – Slovenian
- **trq** – Turkish
- **uk** – United Kingdom
- **ua-utf** – Ukrainian
- **us-acentos** – U.S. International
- **us** – U.S. English

The file **/usr/lib/python2.6/site-packages/system_config_keyboard/keyboard_models.py** on 32-bit systems or **/usr/lib64/python2.6/site-packages/system_config_keyboard/keyboard_models.py** on 64-bit systems also contains this list and is part of the system-config-keyboard package.

ks=nfs:*<server>*:/*<path>*

The installation program looks for the kickstart file on the NFS server *<server>*, as file *<path>*. The installation program uses DHCP to configure the Ethernet card. For example, if your NFS server is server.example.com and the kickstart file is in the NFS share **/mydir/ks.cfg**, the correct boot command would be **ks=nfs:server.example.com:/mydir/ks.cfg**.

ks={http|https}://*<server>*/*<path>*

The installation program looks for the kickstart file on the HTTP or HTTPS server *<server>*, as file *<path>*. The installation program uses DHCP to configure the Ethernet card. For example, if your HTTP server is server.example.com and the kickstart file is in the HTTP directory **/mydir/ks.cfg**, the correct boot command would be **ks=http://server.example.com/mydir/ks.cfg**.

ks=hd:*<device>*:/*<file>*

The installation program mounts the file system on *<device>* (which must be vfat or ext2), and looks for the kickstart configuration file as *<file>* in that file system (for example, **ks=hd:sda3:/mydir/ks.cfg**).

ks=bd:*<biosdev>*:/*<path>*

The installation program mounts the file system on the specified partition on the specified BIOS device *<biosdev>*, and looks for the kickstart configuration file specified in *<path>* (for example, **ks=bd:80p3:/mydir/ks.cfg**). Note this does not work for BIOS RAID sets.

ks=file:/*<file>*

The installation program tries to read the file *<file>* from the file system; no mounts are done. This is normally used if the kickstart file is already on the **initrd** image.

ks=cdrom:/*<path>*

The installation program looks for the kickstart file on CD-ROM, as file *<path>*.

ks

If **ks** is used alone, the installation program configures the Ethernet card to use DHCP. The kickstart file is read from NFS server specified by DHCP option server-name. The name of the kickstart file is one of the following:

- If DHCP is specified and the boot file begins with a /, the boot file provided by DHCP is looked for on the NFS server.
- If DHCP is specified and the boot file begins with something other than a /, the boot file provided by DHCP is looked for in the **/kickstart** directory on the NFS server.
- If DHCP did not specify a boot file, then the installation program tries to read the file **/kickstart/1.2.3.4-kickstart**, where *1.2.3.4* is the numeric IP address of the machine being installed.

ksdevice=*<device>*

The installation program uses this network device to connect to the network. You can specify the device in one of five ways:

- the device name of the interface, for example, **eth0**
- the MAC address of the interface, for example, **00:12:34:56:78:9a**
- the keyword **link**, which specifies the first interface with its link in the **up** state
- the keyword **bootif**, which uses the MAC address that **pxelinux** set in the ***BOOTIF*** variable. Set **IPAPPEND 2** in your **pxelinux.cfg** file to have **pxelinux** set the ***BOOTIF*** variable.
- the keyword **ibft**, which uses the MAC address of the interface specified by iBFT

For example, consider a system connected to an NFS server through the eth1 device. To perform a kickstart installation on this system using a kickstart file from the NFS server, you would use the command **ks=nfs:*<server>*:/*<path>* ksdevice=eth1** at the **boot:** prompt.

kssendmac

Adds HTTP headers to ks=http:// request that can be helpful for provisioning systems. Includes MAC address of all nics in CGI environment variables of the form: "X-RHN-Provisioning-MAC-0: eth0 01:23:45:67:89:ab".

lang=*<lang>*

Language to use for the installation. This should be a language which is valid to be used with the 'lang' kickstart command.

loglevel=*<level>*

Set the minimum level required for messages to be logged. Values for <level> are debug, info, warning, error, and critical. The default value is info.

mediacheck

Activates loader code to give user option of testing integrity of install source (if an ISO-based method).

netmask=*<nm>*

Netmask to use for a network installation.

nofallback

If GUI fails, exit.

nofb

Do not load the VGA16 framebuffer required for doing text-mode installation in some languages.

nofirewire

Do not load support for firewire devices.

noipv4

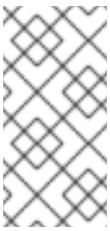
Disable IPv4 networking on the device specified by the **ksdevice=** boot option.

noipv6

Disable IPv6 networking on all network devices on the installed system, and during installation.

IMPORTANT

During installations from a PXE server, IPv6 networking might become active before **anaconda** processes the Kickstart file. If so, this option will have no effect during installation.



NOTE

To disable IPv6 on the installed system, the **--noipv6** kickstart option must be used on each network device, in addition to the **noipv6** boot option. See the Knowledgebase article at https://access.redhat.com/solutions/1565723 for more information about disabling IPv6 system-wide.

nomount

Don't automatically mount any installed Linux partitions in rescue mode.

nonet

Do not auto-probe network devices.

noparport

Do not attempt to load support for parallel ports.

nopass

Do not pass information about the keyboard and mouse from **anaconda** stage 1 (the loader) to stage 2 (the installer).

nopcmcia

Ignore PCMCIA controllers in the system.

noprobe

Do not automatically probe for hardware; prompt the user to allow **anaconda** to probe for particular categories of hardware.

noshell

Do not put a shell on tty2 during install.

repo=cdrom

Do a DVD based installation.

repo=ftp://*<path>*

Use <path> for an FTP installation.

repo=hd:*<dev>*:*<path>*

Use <path> on <dev> for a hard drive installation.

repo=http://*<path>*

Use <path> for an HTTP installation.

repo=https://*<path>*

Use <path> for an HTTPS installation.

repo=nfs:*<path>*

Use <path> for an NFS installation.

rescue

Run rescue environment.

resolution=*<mode>*

Run installer in mode specified, '1024x768' for example.

serial

Turns on serial console support.

skipddc

Do not probe the *Data Display Channel* (DDC) of the monitor. This option provides a workaround if the DDC probe causes the system to stop responding.

syslog=*<host>*[:*<port>*]

Once installation is up and running, send log messages to the syslog process on *<host>*, and optionally, on port *<port>*. Requires the remote syslog process to accept connections (the -r option).

text

Force text mode install.

IMPORTANT

If you select text mode for a kickstart installation, make sure that you specify choices for the partitioning, bootloader, and package selection options. These steps are automated in the text mode, and **anaconda** cannot prompt you for missing information. If you do not provide choices for these options, **anaconda** will stop the installation process.

updates

Prompt for storage device containing updates (bug fixes).

updates=ftp://*<path>*

Image containing updates over FTP.

updates=http://*<path>*

Image containing updates over HTTP.

updates=https://*<path>*

Image containing updates over HTTPS.

upgradeany

Offer to upgrade any Linux installation detected on the system, regardless of the contents or the existence of the **/etc/redhat-release** file.

vnc

Enable vnc-based installation. You will need to connect to the machine using a vnc client application.

vncconnect=*<host>*[:*<port>*]

Connect to the vnc client named *<host>*, and optionally use port *<port>*.

Requires 'vnc' option to be specified as well.

vncpassword=*<password>*

Enable a password for the vnc connection. This will prevent someone from inadvertently connecting to the vnc-based installation.

Requires 'vnc' option to be specified as well.

CHAPTER 33. KICKSTART CONFIGURATOR

Kickstart Configurator allows you to create or modify a kickstart file using a graphical user interface, so that you do not have to remember the correct syntax of the file.

Kickstart Configurator is not installed by default on Red Hat Enterprise Linux 6.9. Run **su - yum install system-config-kickstart** or use your graphical package manager to install the software.

To launch **Kickstart Configurator**, boot your system into a graphical environment, then run **system-config-kickstart**, or click **Applications → System Tools → Kickstart** on the GNOME desktop or **Kickoff Application Launcher**+**Applications → System → Kickstart** on the KDE desktop.

As you are creating a kickstart file, you can click **File → Preview** at any time to review your current selections.

To start with an existing kickstart file, select **File → Open** and select the existing file.

33.1. BASIC CONFIGURATION

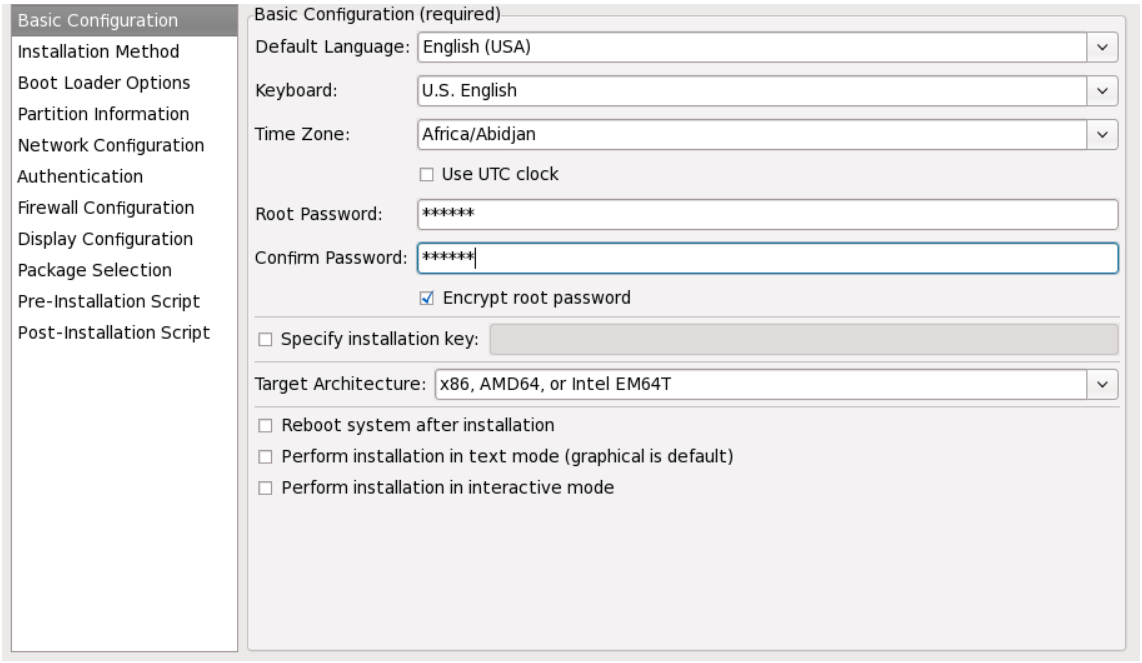


Figure 33.1. Basic Configuration

Choose the language to use during the installation and as the default language to be used after installation from the **Default Language** menu.

Select the system keyboard type from the **Keyboard** menu.

From the **Time Zone** menu, choose the time zone to use for the system. To configure the system to use UTC, select **Use UTC clock**.

Enter the desired root password for the system in the **Root Password** text entry box. Type the same password in the **Confirm Password** text box. The second field is to make sure you do not mistype the password and then realize you do not know what it is after you have completed the installation. To save the password as an encrypted password in the file, select **Encrypt root password**. If the encryption option is selected, when the file is saved, the plain text password that you typed is encrypted and written

to the kickstart file. Do not type an already encrypted password and select to encrypt it. Because a kickstart file is a plain text file that can be easily read, it is recommended that an encrypted password be used.

Choosing **Target Architecture** specifies which specific hardware architecture distribution is used during installation.

Choosing **Target Architecture** specifies which specific hardware architecture distribution is used during installation.

Choosing **Reboot system after installation** reboots your system automatically after the installation is finished.

Kickstart installations are performed in graphical mode by default. To override this default and use text mode instead, select the **Perform installation in text mode** option.

You can perform a kickstart installation in interactive mode. This means that the installation program uses all the options pre-configured in the kickstart file, but it allows you to preview the options in each screen before continuing to the next screen. To continue to the next screen, click the **Next** button after you have approved the settings or change them before continuing the installation. To select this type of installation, select the **Perform installation in interactive mode** option.

33.2. INSTALLATION METHOD

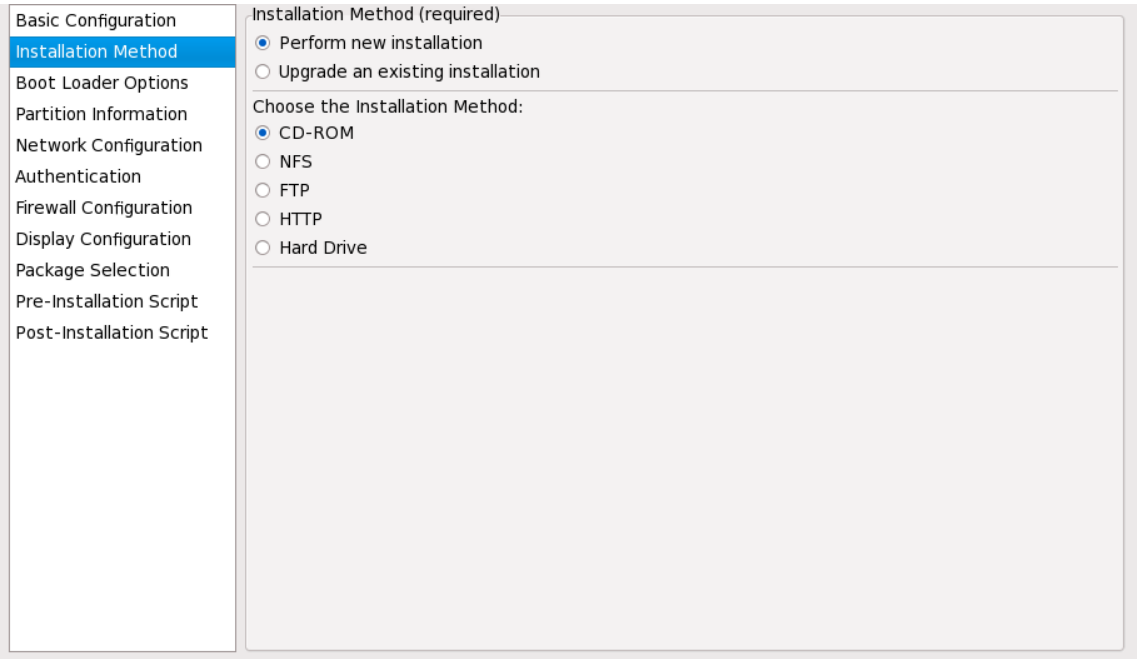


Figure 33.2. Installation Method

The **Installation Method** screen allows you to choose whether to perform a new installation or an upgrade. If you choose upgrade, the **Partition Information** and **Package Selection** options are disabled. They are not supported for kickstart upgrades.

Choose the type of kickstart installation or upgrade from the following options:

- **DVD** — Choose this option to install or upgrade from the Red Hat Enterprise Linux DVD.
- **NFS** — Choose this option to install or upgrade from an NFS shared directory. In the text field for the NFS server, enter a fully-qualified domain name or IP address. For the NFS directory, enter the name of the NFS directory that contains the ***variant*** directory of the installation tree. For

example, if the NFS server contains the directory **/mirrors/redhat/i386/Server/**, enter **/mirrors/redhat/i386/** for the NFS directory.

- **FTP** — Choose this option to install or upgrade from an FTP server. In the FTP server text field, enter a fully-qualified domain name or IP address. For the FTP directory, enter the name of the FTP directory that contains the ***variant*** directory. For example, if the FTP server contains the directory **/mirrors/redhat/i386/Server/**, enter **/mirrors/redhat/i386/Server/** for the FTP directory. If the FTP server requires a username and password, specify them as well.

- **HTTP** — Choose this option to install or upgrade from an HTTP server. In the text field for the HTTP server, enter the fully-qualified domain name or IP address. For the HTTP directory, enter the name of the HTTP directory that contains the ***variant*** directory. For example, if the HTTP server contains the directory **/mirrors/redhat/i386/Server/**, enter **/mirrors/redhat/i386/Server/** for the HTTP directory.

- **Hard Drive** — Choose this option to install or upgrade from a hard drive. Hard drive installations require the use of ISO images. Be sure to verify that the ISO images are intact before you start the installation. To verify them, use an **md5sum** program as well as the **linux mediacheck** boot option as discussed in Section 28.6.1, "Verifying Boot Media". Enter the hard drive partition that contains the ISO images (for example, **/dev/hda1**) in the **Hard Drive Partition** text box. Enter the directory that contains the ISO images in the **Hard Drive Directory** text box.

33.3. BOOT LOADER OPTIONS

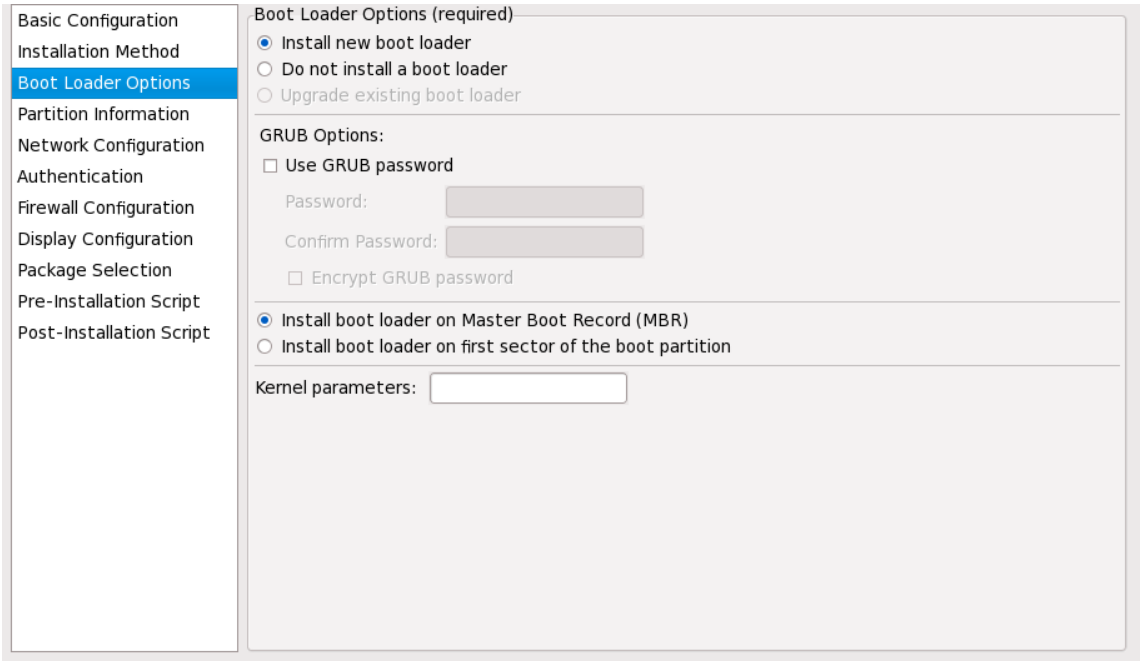


Figure 33.3. Boot Loader Options

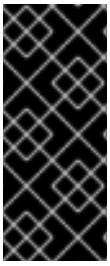
Please note that this screen will be disabled if you have specified a target architecture other than x86 / x86_64.

GRUB is the default boot loader for Red Hat Enterprise Linux on x86 / x86_64 architectures. If you do not want to install a boot loader, select **Do not install a boot loader**. If you choose not to install a boot loader, make sure you create a boot diskette or have another way to boot your system, such as a third-party boot loader.

You must choose where to install the boot loader (the Master Boot Record or the first sector of the **/boot** partition). Install the boot loader on the MBR if you plan to use it as your boot loader.

To pass any special parameters to the kernel to be used when the system boots, enter them in the **Kernel parameters** text field. For example, if you have an IDE CD-ROM Writer, you can tell the kernel to use the SCSI emulation driver that must be loaded before using **cdrecord** by configuring **hdd=ide-scsi** as a kernel parameter (where **hdd** is the CD-ROM device).

You can password protect the GRUB boot loader by configuring a GRUB password. Select **Use GRUB password**, and enter a password in the **Password** field. Type the same password in the **Confirm Password** text field. To save the password as an encrypted password in the file, select **Encrypt GRUB password**. If the encryption option is selected, when the file is saved, the plain text password that you typed is encrypted and written to the kickstart file. If the password you typed was already encrypted, uncheck the encryption option.



IMPORTANT

It is highly recommended to set up a boot loader password on every machine. An unprotected boot loader can allow a potential attacker to modify the system's boot options and gain access to the system. See the chapter titled *Workstation Security* in the *Red Hat Enterprise Linux Security Guide* for more information on boot loader passwords and password security in general.

If **Upgrade an existing installation** is selected on the **Installation Method** page, select **Upgrade existing boot loader** to upgrade the existing boot loader configuration, while preserving the old entries.

33.4. PARTITION INFORMATION

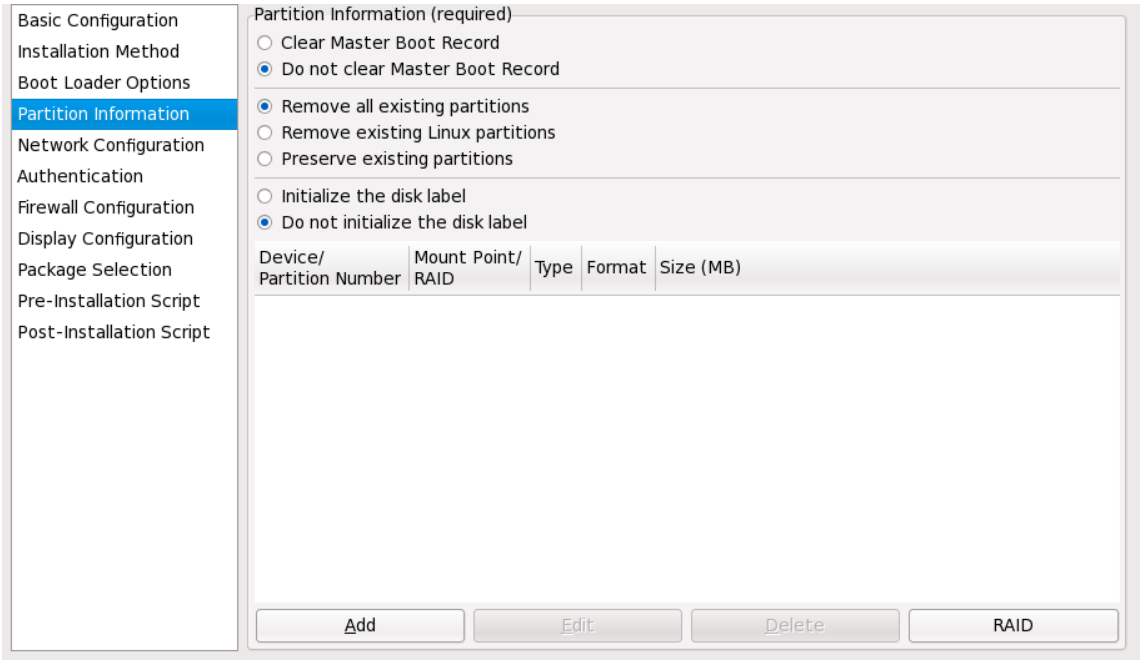


Figure 33.4. Partition Information

Select whether or not to clear the Master Boot Record (MBR). Choose to remove all existing partitions, remove all existing Linux partitions, or preserve existing partitions.

To initialize the disk label to the default for the architecture of the system (for example, **msdos** for x86), select **Initialize the disk label** if you are installing on a brand new hard drive.

NOTE

Although **anaconda** and **kickstart** support Logical Volume Management (LVM), at present there is no mechanism for configuring this using the **Kickstart Configurator**.

33.4.1. Creating Partitions

To create a partition, click the **Add** button. The **Partition Options** window shown in Figure 33.5, "Creating Partitions" appears. Choose the mount point, file system type, and partition size for the new partition. Optionally, you can also choose from the following:

- In the **Additional Size Options** section, choose to make the partition a fixed size, up to a chosen size, or fill the remaining space on the hard drive. If you selected swap as the file system type, you can select to have the installation program create the swap partition with the recommended size instead of specifying a size.
- Force the partition to be created as a primary partition.
- Create the partition on a specific hard drive. For example, to make the partition on the first IDE hard disk (**/dev/hda**), specify **hda** as the drive. Do not include **/dev** in the drive name.
- Use an existing partition. For example, to make the partition on the first partition on the first IDE hard disk (**/dev/hda1**), specify **hda1** as the partition. Do not include **/dev** in the partition name.
- Format the partition as the chosen file system type.

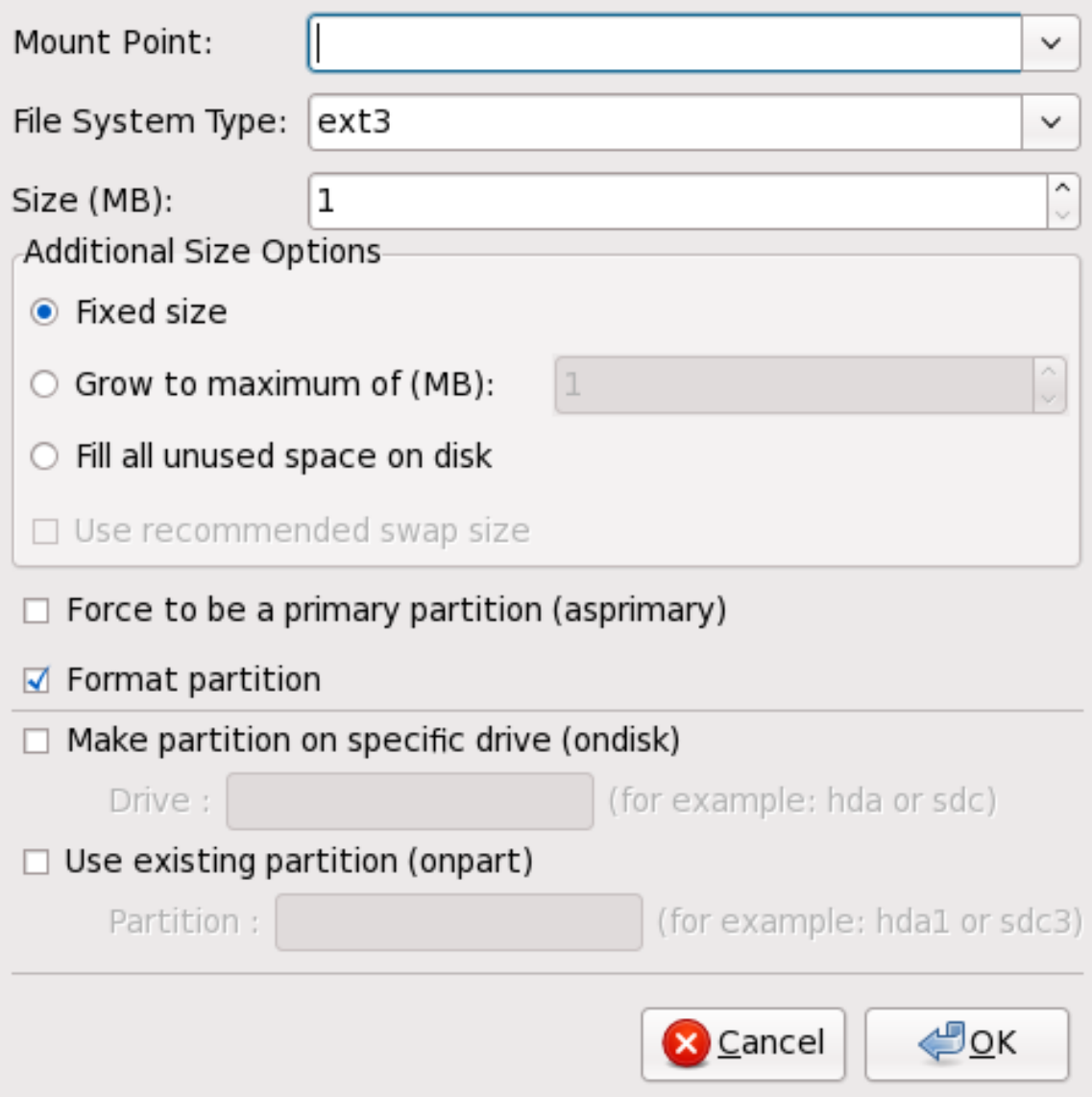


Figure 33.5. Creating Partitions

To edit an existing partition, select the partition from the list and click the **Edit** button. The same **Partition Options** window appears as when you chose to add a partition as shown in Figure 33.5, "Creating Partitions", except it reflects the values for the selected partition. Modify the partition options and click **OK**.

To delete an existing partition, select the partition from the list and click the **Delete** button.

33.4.1.1. Creating Software RAID Partitions

To create a software RAID partition, use the following steps:

1. Click the **RAID** button.
2. Select **Create a software RAID partition**.
3. Configure the partitions as previously described, except select **Software RAID** as the file system type. Also, you must specify a hard drive on which to make the partition or specify an existing partition to use.

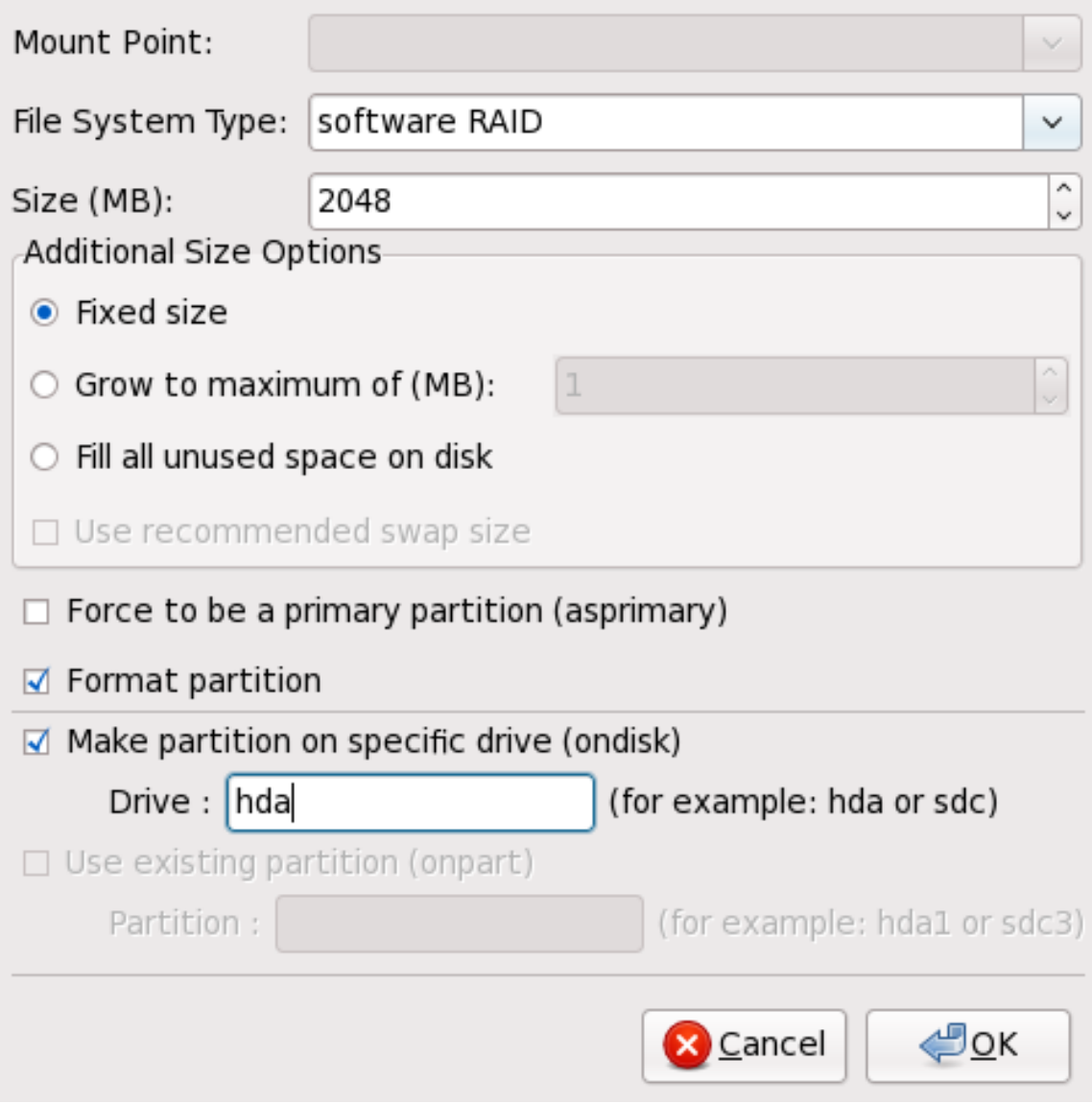


Figure 33.6. Creating a Software RAID Partition

Repeat these steps to create as many partitions as needed for your RAID setup. All of your partitions do not have to be RAID partitions.

After creating all the partitions needed to form a RAID device, follow these steps:

1. Click the **RAID** button.
2. Select **Create a RAID device**.
3. Select a mount point, file system type, RAID device name, RAID level, RAID members, number of spares for the software RAID device, and whether to format the RAID device.

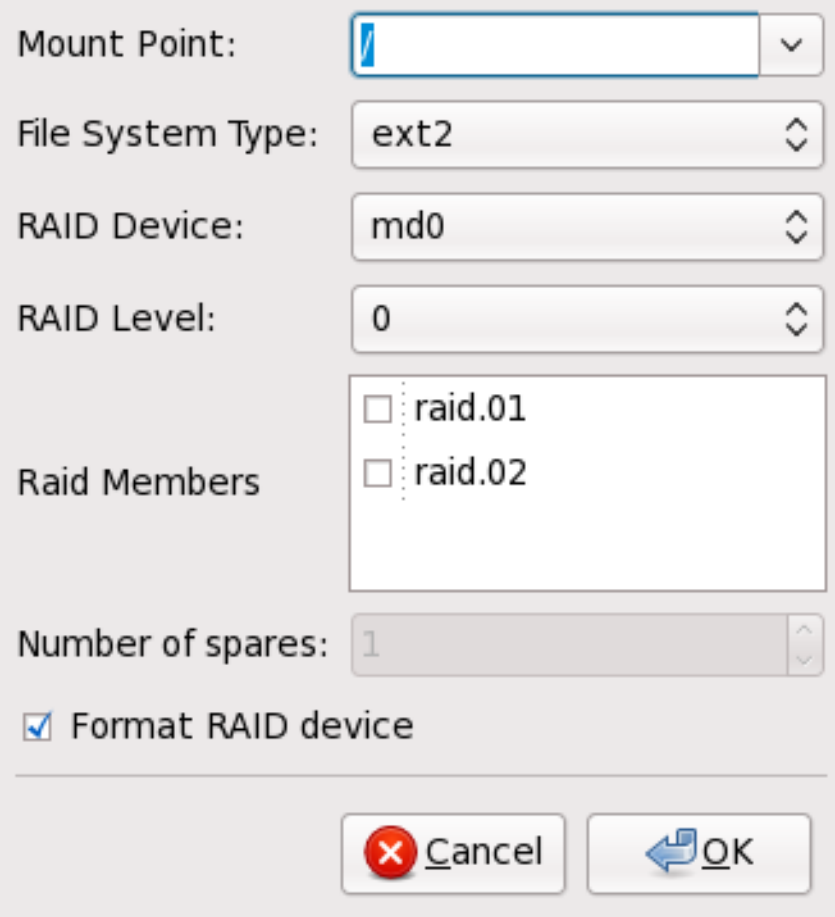


Figure 33.7. Creating a Software RAID Device

4. Click **OK** to add the device to the list.

33.5. NETWORK CONFIGURATION

Figure 33.8. Network Configuration

If the system to be installed via kickstart does not have an Ethernet card, do not configure one on the **Network Configuration** page.

Networking is only required if you choose a networking-based installation method (NFS, FTP, or HTTP). Networking can always be configured after installation with the **Network Administration Tool** (**system-config-network**). Refer to the Red Hat Enterprise Linux Deployment Guide for details.

For each Ethernet card on the system, click **Add Network Device** and select the network device and network type for the device. Select **eth0** to configure the first Ethernet card, **eth1** for the second Ethernet card, and so on.

33.6. AUTHENTICATION

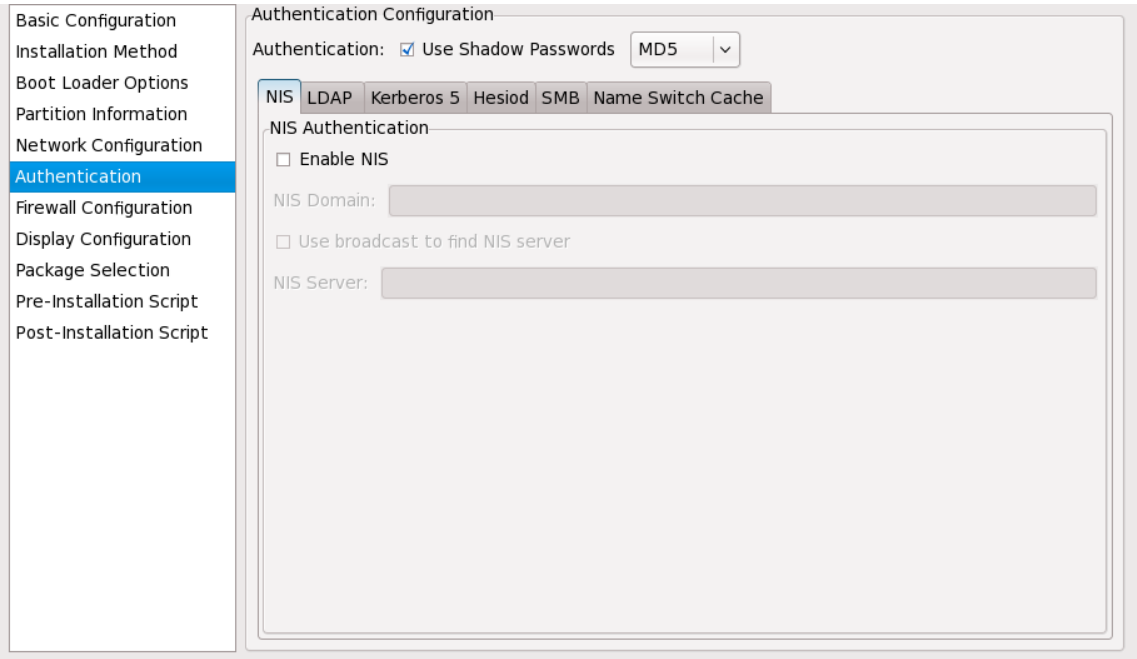


Figure 33.9. Authentication

In the **Authentication** section, select whether to use shadow passwords and MD5 encryption for user passwords. These options are highly recommended and chosen by default.

The **Authentication Configuration** options allow you to configure the following methods of authentication:

- NIS
- LDAP
- Kerberos 5
- Hesiod
- SMB
- Name Switch Cache

These methods are not enabled by default. To enable one or more of these methods, click the appropriate tab, click the checkbox next to **Enable**, and enter the appropriate information for the authentication method. Refer to the Red Hat Enterprise Linux Deployment Guide for more information about the options.

33.7. FIREWALL CONFIGURATION

The **Firewall Configuration** window allows you to configure firewall settings for the installed system.

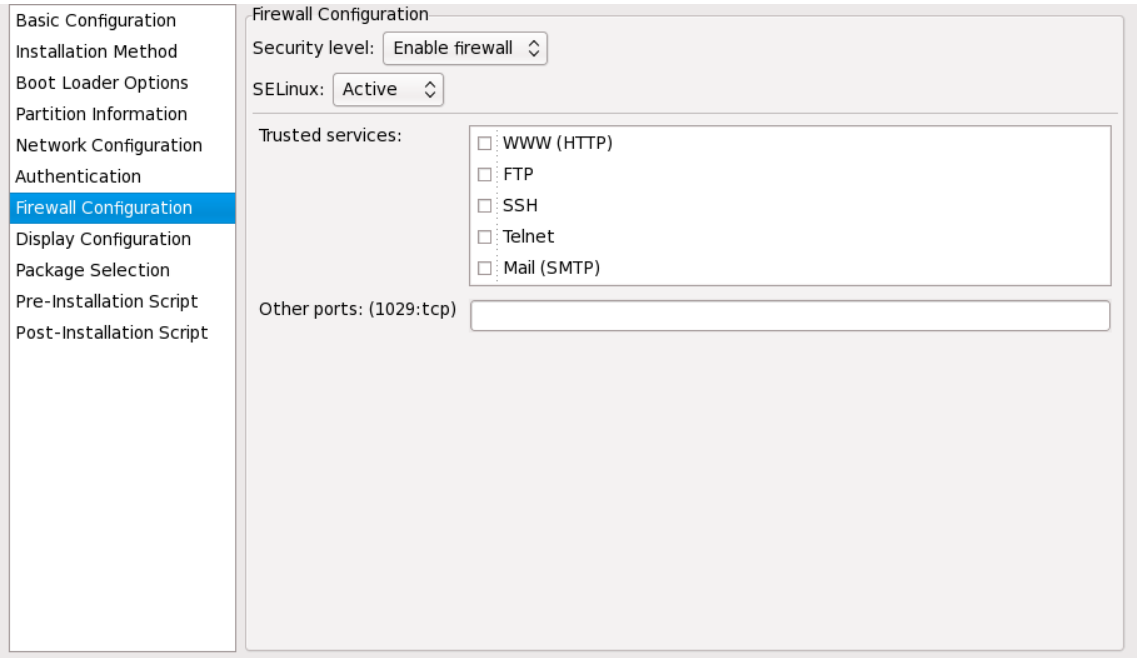


Figure 33.10. Firewall Configuration

If **Disable firewall** is selected, the system allows complete access to any active services and ports. No connections to the system are refused or denied.

Selecting **Enable firewall** configures the system to reject incoming connections that are not in response to outbound requests, such as DNS replies or DHCP requests. If access to services running on this machine is required, you can choose to allow specific services through the firewall.

Only devices configured in the **Network Configuration** section are listed as available **Trusted devices**. Connections from any devices selected in the list are accepted by the system. For example, if **eth1** only receives connections from internal system, you might want to allow connections from it.

If a service is selected in the **Trusted services** list, connections for the service are accepted and processed by the system.

In the **Other ports** text field, list any additional ports that should be opened for remote access. Use the following format: **port:protocol**. For example, to allow IMAP access through the firewall, specify **imap:tcp**. Numeric ports can also be specified explicitly; to allow UDP packets on port 1234 through the firewall, enter **1234:udp**. To specify multiple ports, separate them with commas.

33.7.1. SELinux Configuration

Kickstart can set SELinux to **enforcing**, **permissive** or **disabled** mode. Finer grained configuration is not possible at this time.

33.8. DISPLAY CONFIGURATION

If you are installing the X Window System, you can configure it during the kickstart installation by checking the **Configure the X Window System** option on the **Display Configuration** window as shown in Figure 33.11, “X Configuration”. If this option is not chosen, the X configuration options are disabled and the **skipx** option is written to the kickstart file.

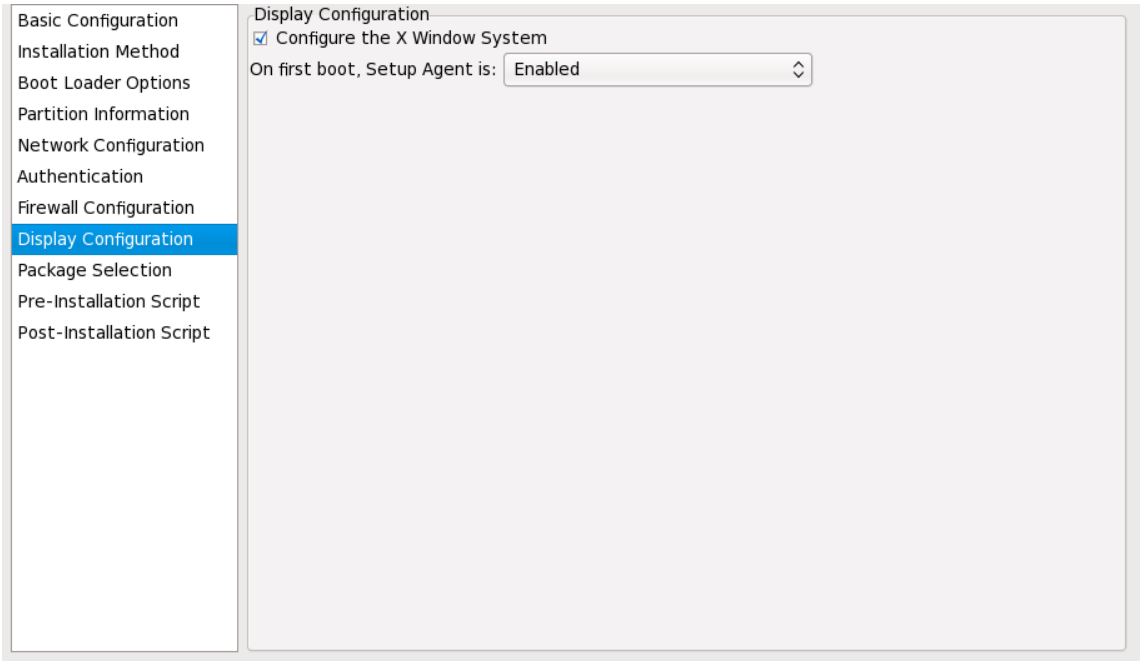


Figure 33.11. X Configuration

Select whether to start the Setup Agent the first time the installed system boots. The Setup Agent is disabled by default, but the setting can be changed to enabled or enabled in reconfiguration mode. Reconfiguration mode enables the language, mouse, keyboard, root password, security level, time zone, and networking configuration options in addition to the default ones.

33.9. PACKAGE SELECTION

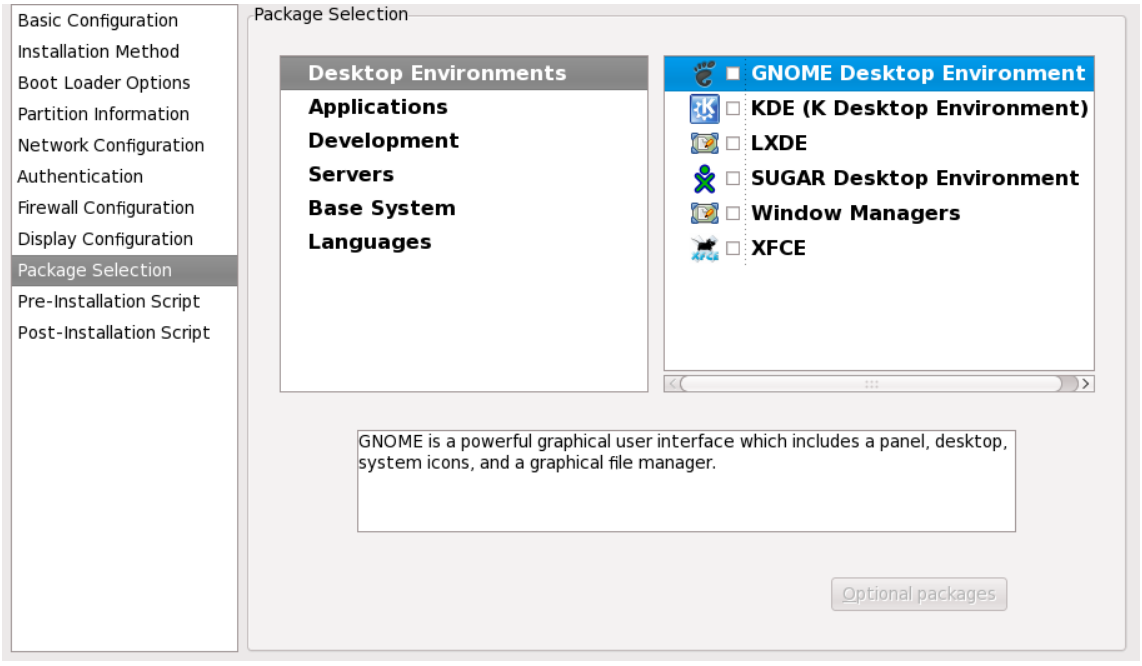


Figure 33.12. Package Selection

The **Package Selection** window allows you to choose which package groups to install.

Package resolution is carried out automatically.

Currently, **Kickstart Configurator** does not allow you to select individual packages. To install individual packages, modify the **%packages** section of the kickstart file after you save it. Refer to Section 32.5, "Package Selection" for details.

33.10. PRE-INSTALLATION SCRIPT

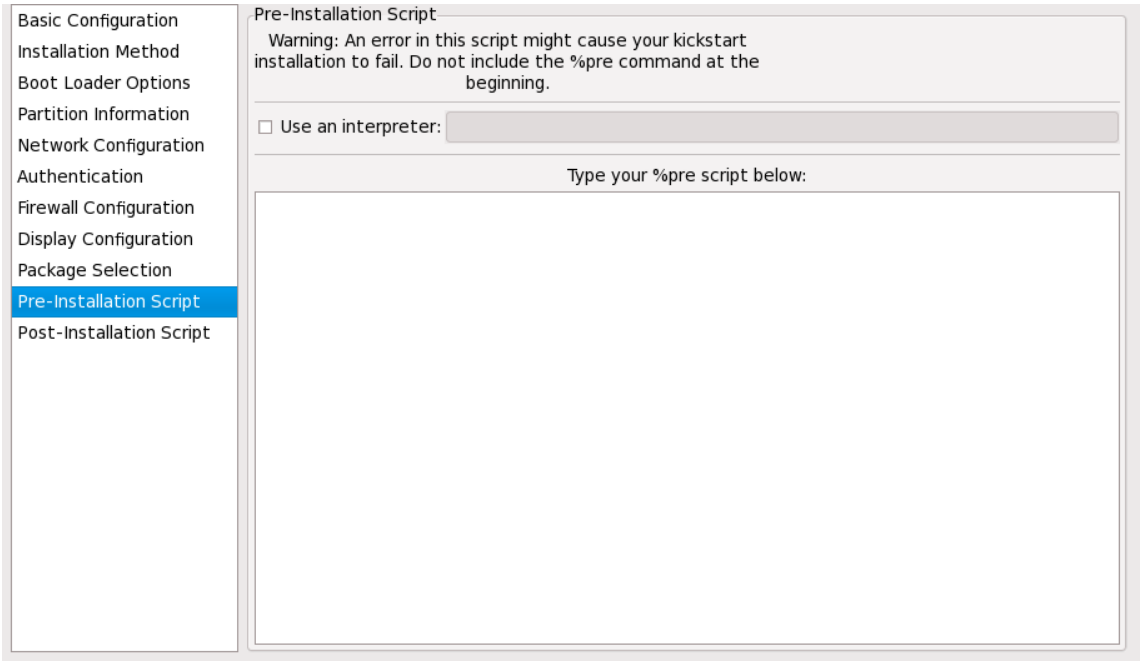
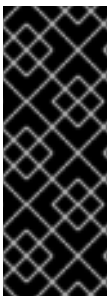


Figure 33.13. Pre-Installation Script

You can add commands to run on the system immediately after the kickstart file has been parsed and before the installation begins. If you have configured the network in the kickstart file, the network is enabled before this section is processed. To include a pre-installation script, type it in the text area.



IMPORTANT

The version of **anaconda** in previous releases of Red Hat Enterprise Linux included a version of **busybox** that provided shell commands in the pre-installation and post-installation environments. The version of **anaconda** in Red Hat Enterprise Linux 6 no longer includes **busybox**, and uses GNU **bash** commands instead.

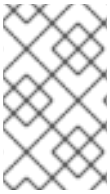
Refer to Appendix G, *Alternatives to busybox commands* for more information.

To specify a scripting language to use to execute the script, select the **Use an interpreter** option and enter the interpreter in the text box beside it. For example, **/usr/bin/python2.6** can be specified for a Python script. This option corresponds to using **%pre --interpreter *[/usr/bin/python2.6]*** in your kickstart file.

Only the most commonly used commands are available in the pre-installation environment. See Section 32.6, "Pre-installation Script" for a complete list.

IMPORTANT

Do not include the **%pre** command. It is added for you.



NOTE

The pre-installation script is run after the source media is mounted and stage 2 of the bootloader has been loaded. For this reason it is not possible to change the source media in the pre-installation script.

33.11. POST-INSTALLATION SCRIPT

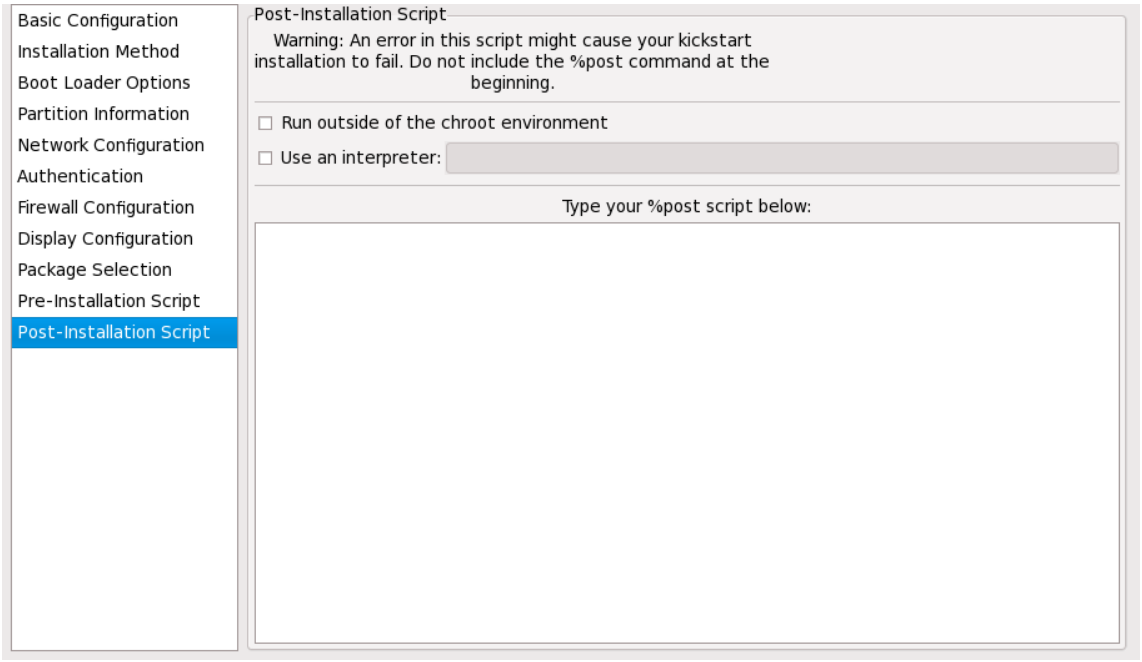


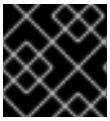
Figure 33.14. Post-Installation Script

You can also add commands to execute on the system after the installation is completed. If the network is properly configured in the kickstart file, the network is enabled, and the script can include commands to access resources on the network. To include a post-installation script, type it in the text area.

IMPORTANT

The version of **anaconda** in previous releases of Red Hat Enterprise Linux included a version of **busybox** that provided shell commands in the pre-installation and post-installation environments. The version of **anaconda** in Red Hat Enterprise Linux 6 no longer includes **busybox**, and uses GNU **bash** commands instead.

Refer to Appendix G, *Alternatives to busybox commands* for more information.



IMPORTANT

Do not include the **%post** command. It is added for you.

For example, to change the message of the day for the newly installed system, add the following command to the **%post** section:

```
echo "Welcome!" > /etc/motd
```

NOTE

More examples can be found in Section 32.8, “Kickstart Examples”.

33.11.1. Chroot Environment

To run the post-installation script outside of the chroot environment, click the checkbox next to this option on the top of the **Post-Installation** window. This is equivalent to using the **--nochroot** option in the **%post** section.

To make changes to the newly installed file system, within the post-installation section, but outside of the chroot environment, you must prepend the directory name with **/mnt/sysimage/**.

For example, if you select **Run outside of the chroot environment**, the previous example must be changed to the following:

```
echo "Welcome!" > /mnt/sysimage/etc/motd
```

33.11.2. Use an Interpreter

To specify a scripting language to use to execute the script, select the **Use an interpreter** option and enter the interpreter in the text box beside it. For example, **/usr/bin/python2.2** can be specified for a Python script. This option corresponds to using **%post --interpreter */usr/bin/python2.2*** in your kickstart file.

33.12. SAVING THE FILE

To review the contents of the kickstart file after you have finished choosing your kickstart options, select **File** => **Preview** from the pull-down menu.

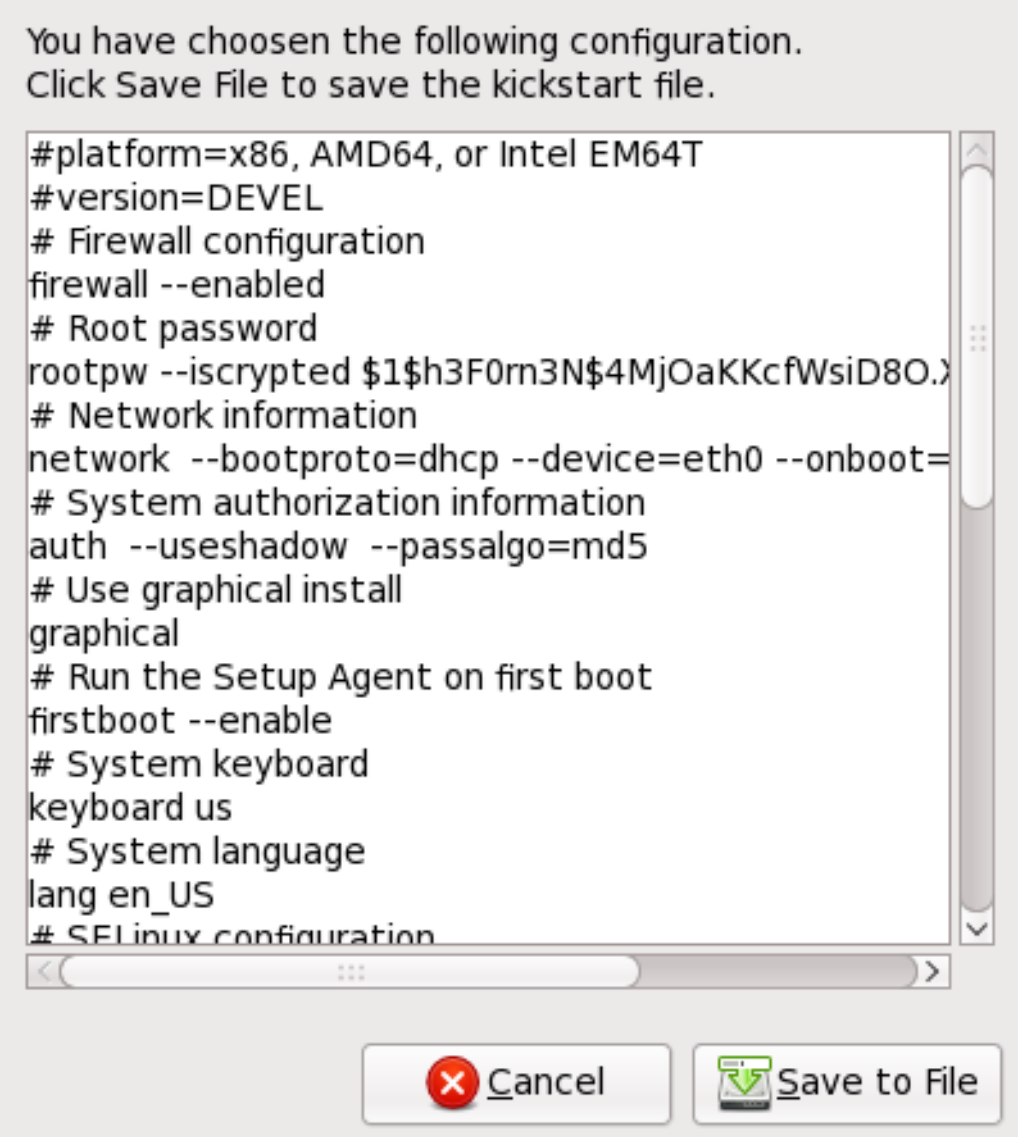


Figure 33.15. Preview

To save the kickstart file, click the **Save to File** button in the preview window. To save the file without previewing it, select **File** => **Save File** or press **Ctrl**+**S** . A dialog box appears. Select where to save the file.

After saving the file, refer to Section 32.11, "Starting a Kickstart Installation" for information on how to start the kickstart installation.

PART V. AFTER INSTALLATION

This part of the *Red Hat Enterprise Linux Installation Guide* covers finalizing the installation, as well as some installation-related tasks that you might perform at some time in the future. These include:

- using a Red Hat Enterprise Linux installation disc to rescue a damaged system.
- upgrading to a new version of Red Hat Enterprise Linux.
- removing Red Hat Enterprise Linux from your computer.

CHAPTER 34. FIRSTBOOT

IMPORTANT

Firstboot is only available on systems after a graphical installation or after a kickstart installation where a desktop and the X window system were installed and graphical login was enabled. If you performed a text-mode installation or a kickstart installation that did not include a desktop and the X window system, the **firstboot** configuration tool does not appear.

Firstboot launches the first time that you start a new Red Hat Enterprise Linux system. Use **firstboot** to configure the system for use before you log in.

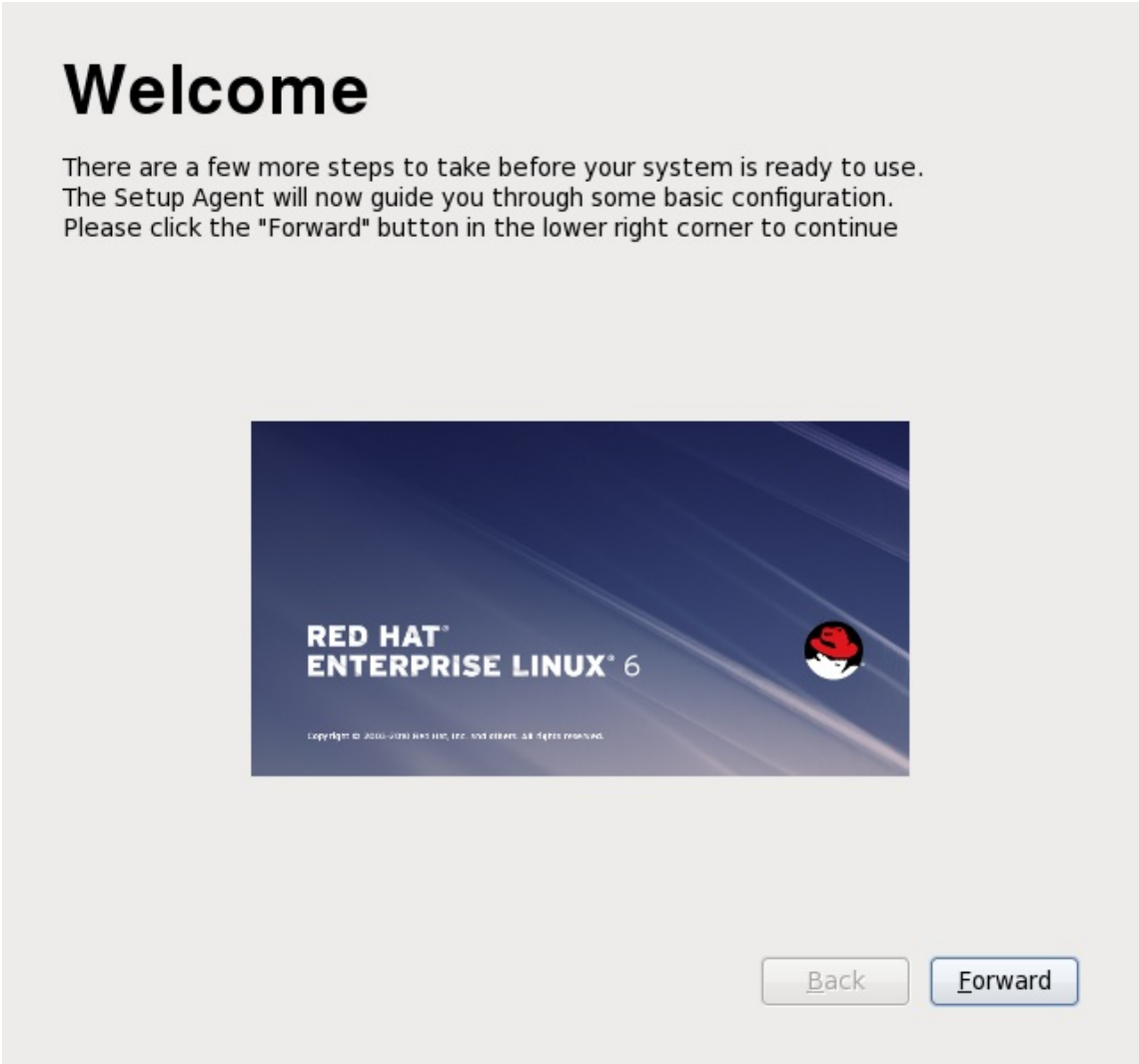


Figure 34.1. Firstboot welcome screen

Click **Forward** to start **firstboot**.

34.1. LICENSE INFORMATION

This screen displays the overall licensing terms for Red Hat Enterprise Linux.

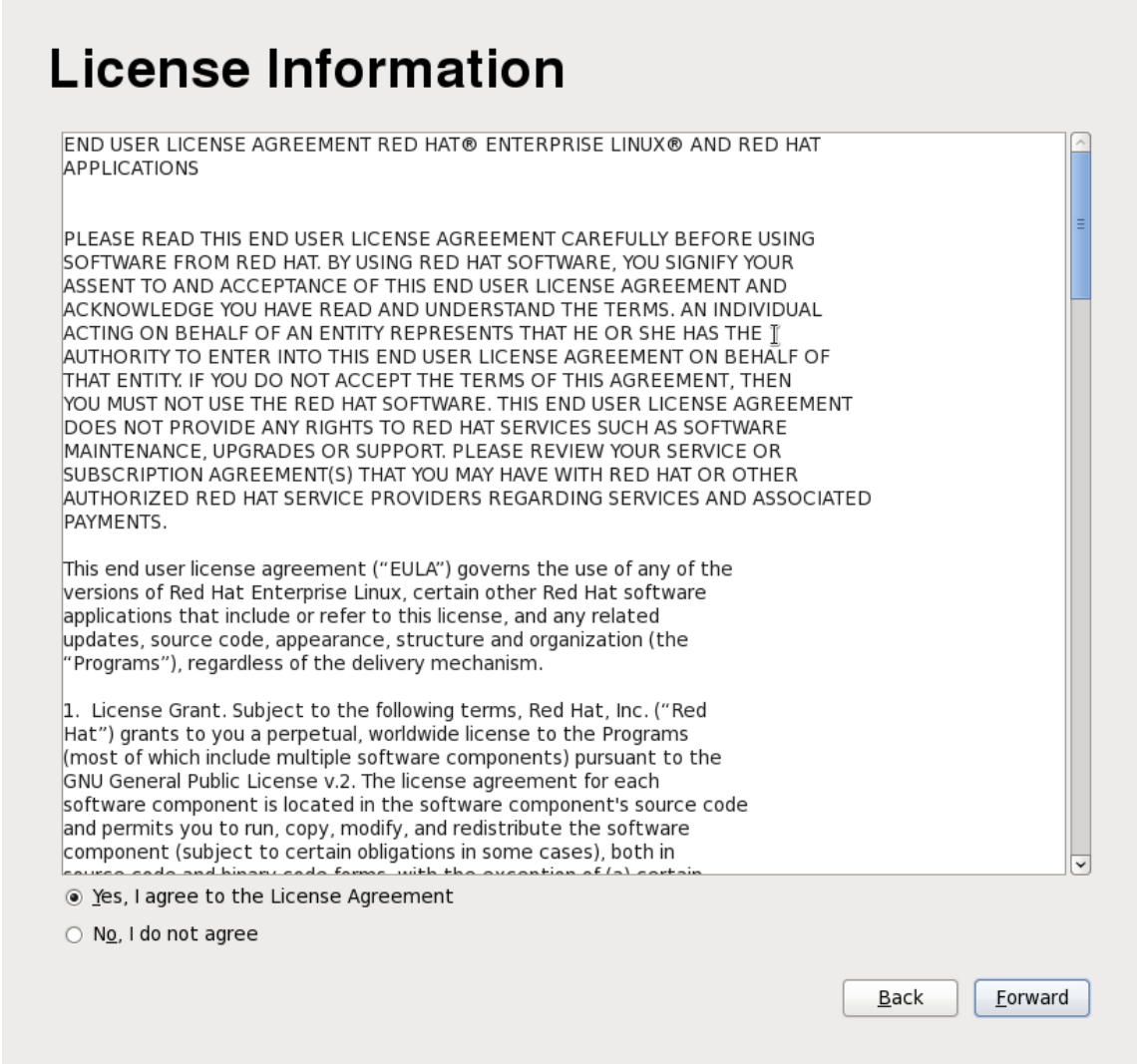


Figure 34.2. Firstboot license screen

If you agree to the terms of the license, select **Yes, I agree to the License Agreement** and click **Forward**.

34.2. CONFIGURING THE SUBSCRIPTION SERVICE

The products installed on a system (including the operating system itself) are covered by *subscriptions*. A subscription service is used to track registered systems, the products installed on those systems, and the subscriptions *attached* to the system to cover those products.

The **Subscription Management Registration** screens identify which subscription service to use and, by default, attach the best-matched subscriptions to the system.

More information about subscription management is available in the *Red Hat Subscription Management* guide.

34.2.1. Set Up Software Updates

The first step is to select whether to register the system immediately with a subscription service. To register the system, select **Yes, I'd like to register now**, and click **Forward**.

Figure 34.3. Set Up Software Updates

NOTE

Even if a system is not registered at firstboot, it can be registered with any of those three subscription services later, using the Red Hat Subscription Manager tools[13].

More information about the Red Hat Subscription Manager tools can be found in the *Red Hat Subscription Management Guide*.

34.2.2. Choose Service

Use the **Choose Service** screen to choose what kind of subscription service to register the system with. Click **Proxy Setup** to configure a proxy server if necessary. More information about subscription management with a proxy server can be found in the *Red Hat Subscription Management* guide.

Red Hat Subscription Management

Any subscription service which uses the proper X.509 certificates to identify the system, installed products, and attached subscriptions is part of *Red Hat Subscription Management*. This includes Customer Portal Subscription Management (hosted services), Subscription Asset Manager (on-premise subscription service and proxied content delivery), and CloudForms System Engine (on-premise subscription and content delivery services).

This option is the default. Red Hat Subscription Management is strongly recommended for organizations that do *not* run a local Satellite server.

Red Hat Network (RHN) Classic

Select the **Red Hat Network (RHN) Classic** option to use the legacy systems-management features of Red Hat Network. While RHN Classic can be used with Red Hat Enterprise Linux 6.x systems, it is intended primarily for existing, legacy systems. It is recommended that new installations use Red Hat Subscription Management.

An RHN Satellite or RHN Proxy

Use this option in environments with access to a local mirror of the Red Hat Network content.

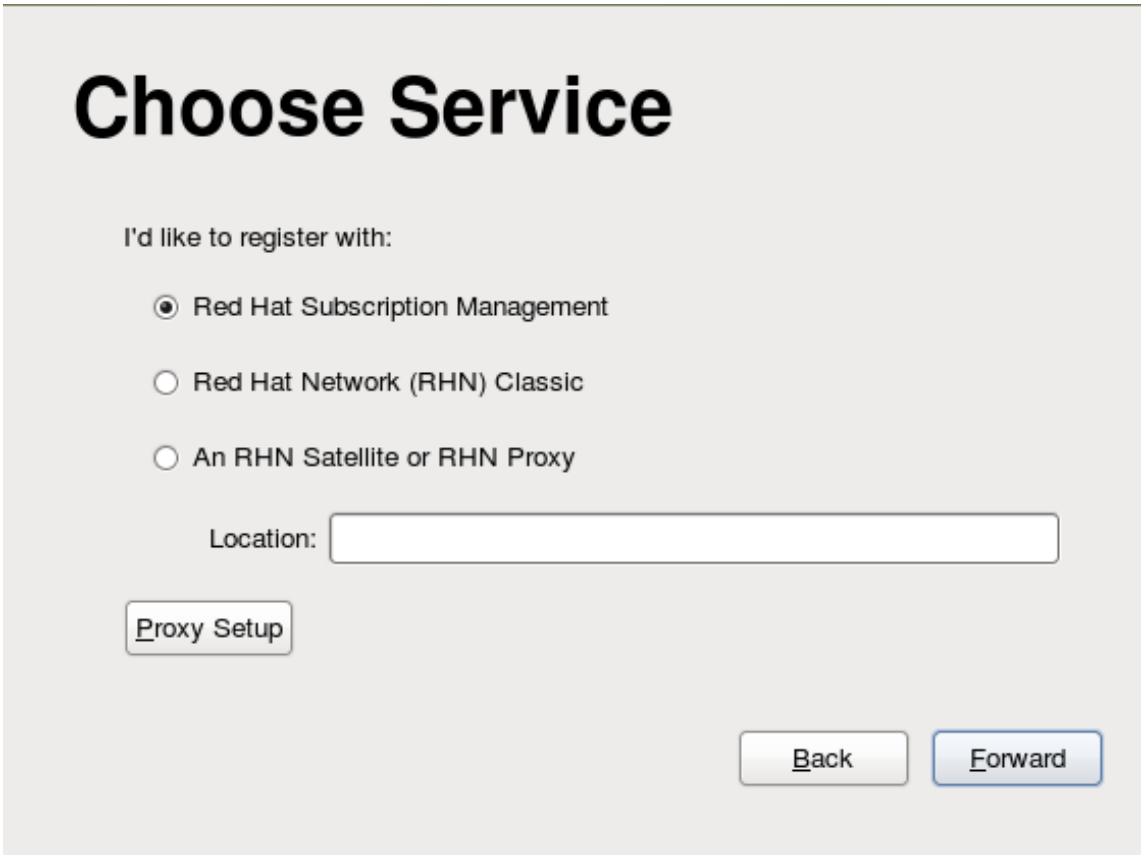


Figure 34.4. Choose Service

34.2.3. Subscription Management Registration

Red Hat uses *X.509 certificates* to identify installed products on a system, the subscriptions attached to a system, and the system itself within the subscription service inventory. There are several different subscription services which use and recognize certificate-base subscriptions, and a system can be registered with any of them in firstboot:

- Customer Portal Subscription Management, hosted services from Red Hat (the default)
- Subscription Asset Manager, an on-premise subscription server which proxies content delivery back to the Customer Portal's services
- CloudForms System Engine, an on-premise service which handles both subscription services and content delivery

The specific type of subscription/content service does not need to be selected; all three server types (Customer Portal Subscription Management, Subscription Asset Manager, and CloudForms System Engine) are within Red Hat Subscription Management and use the same types of service APIs. The only thing that needs to be identified is the hostname of the service to connect to and then the appropriate user credentials for that service.

1. To identify which subscription server to use for registration, enter the hostname of the service. The default service is Customer Portal Subscription Management, with the hostname **subscription.rhn.redhat.com**. To use a different subscription service, such as Subscription Asset Manager, enter the hostname of the local server.

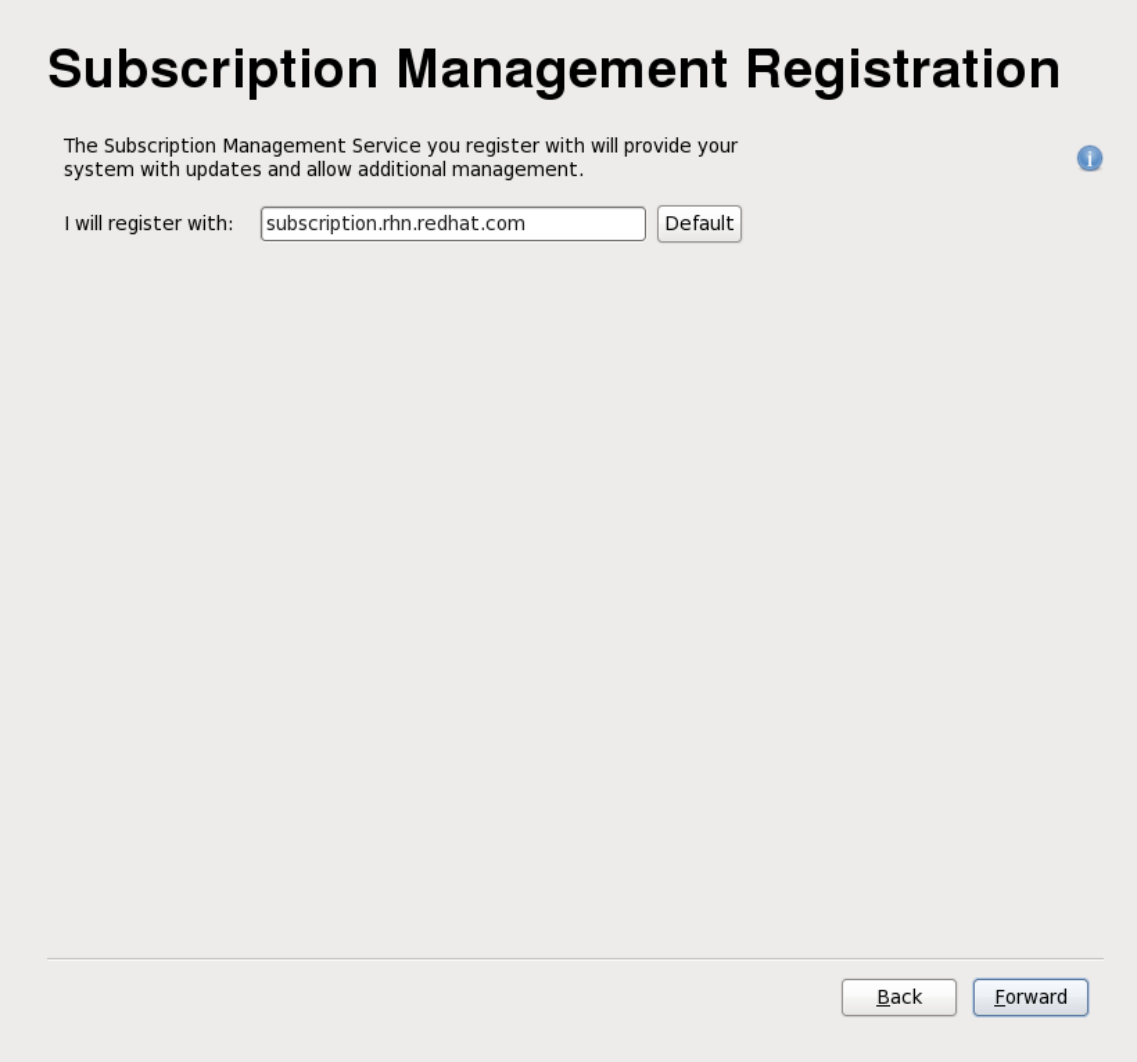


Figure 34.5. Subscription Service Selection

2. Click **Forward**.

3. Enter the user credentials **for the given subscription service**to log in.

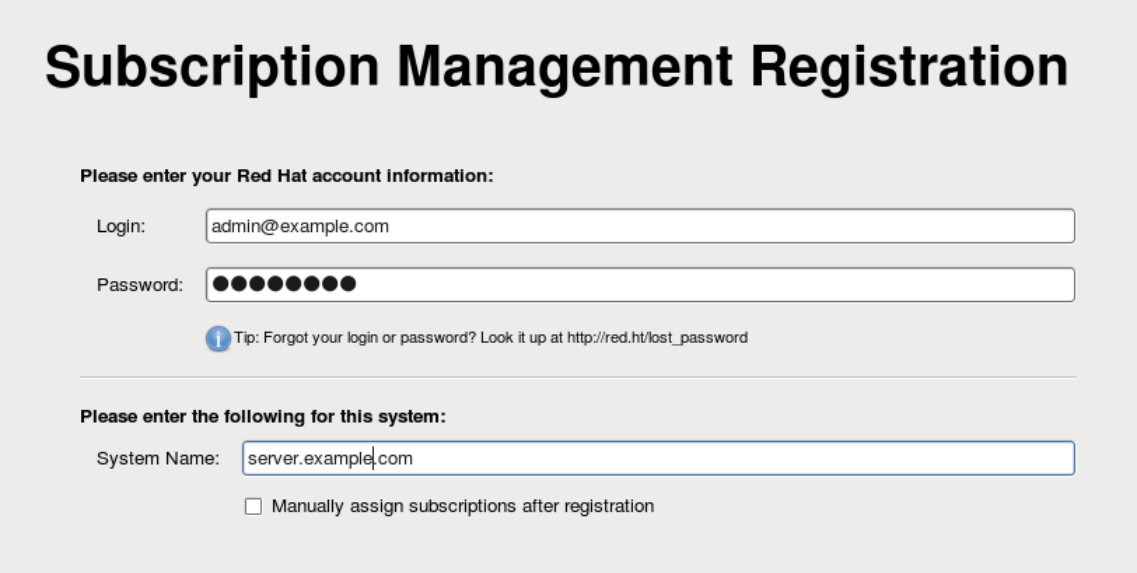
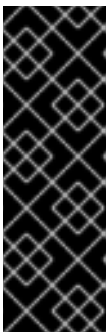


Figure 34.6. Subscription Management Registration



IMPORTANT

The user credentials to use depend on the subscription service. When registering with the Customer Portal, use the Red Hat Network credentials for the administrator or company account.

However, for Subscription Asset Manager or CloudForms System engine, the user account to use is created within the on-premise service and probably is not the same as the Customer Portal user account.

If you have lost your login or password for the Customer Portal, recover them from https://www.redhat.com/wapps/sso/lostPassword.html. For lost login or password information for Subscription Asset Manager or CloudForms System Engine, contact your local administrator.

4. Set the system name for the host. This is anything which uniquely and clearly identifies the system within the subscription service inventory. This is usually the hostname or fully-qualified domain name of the machine, but it can be any string.

5. *Optional.* Set whether subscriptions should be set manually after registration. By default, this checkbox is unchecked so that the best-matched subscriptions are automatically applied to the system. Selecting this checkbox means that subscriptions must be added to the system manually after firstboot registration is complete. (Even if subscriptions are auto-attached, additional subscriptions can be added to the system later using the local Subscription Manager tools.)

6. When registration begins, firstboot scans for organizations and environments (sub-domains within the organization) to which to register the system.

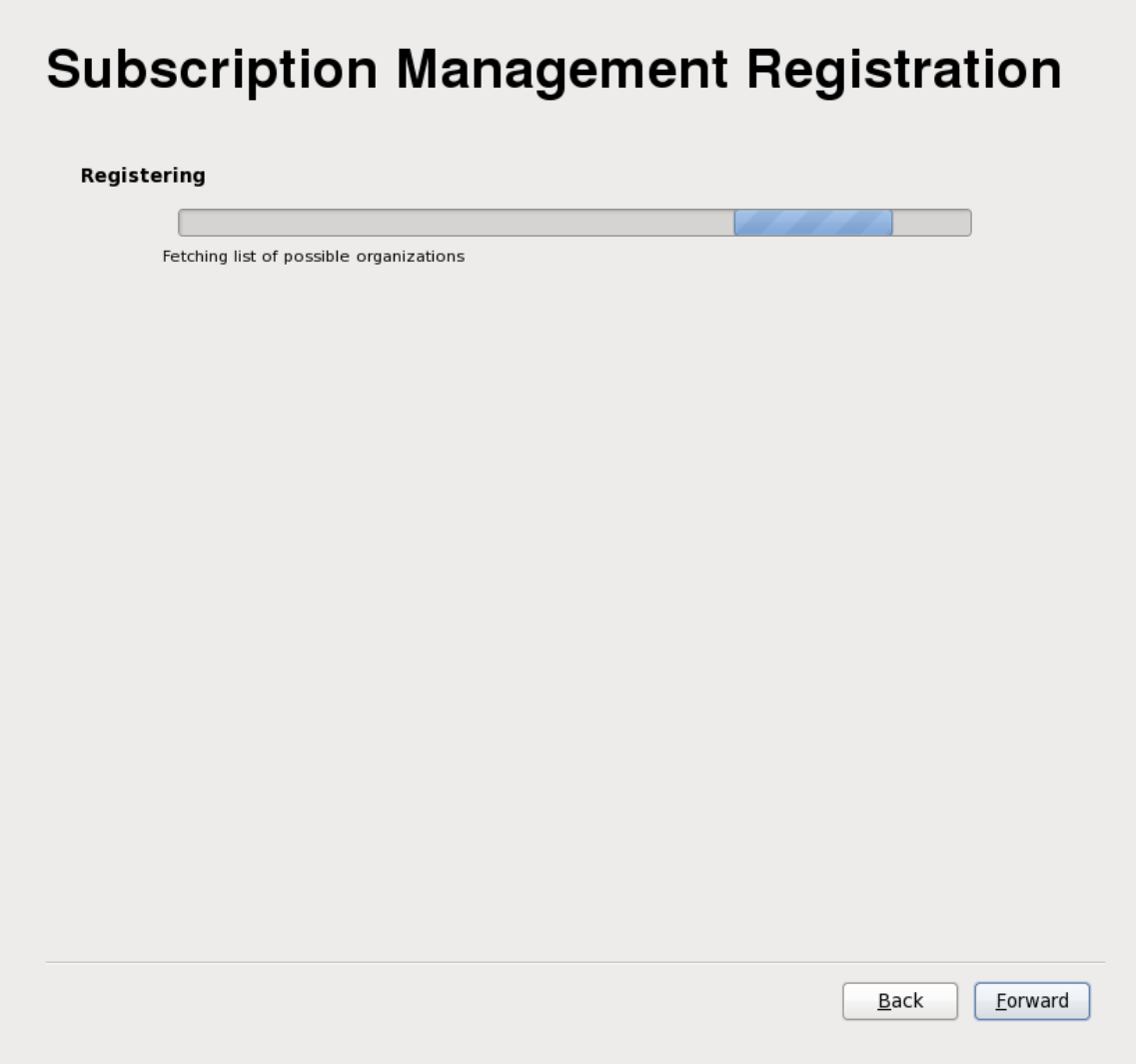


Figure 34.7. Organization Scan

IT environments that use Customer Portal Subscription Management have only a single organization, so no further configuration is necessary. IT infrastructures that use a local subscription service like Subscription Asset Manager might have multiple organizations configured, and those organizations may have multiple environments configured within them.

If multiple organizations are detected, Subscription Manager prompts to select the one to join.

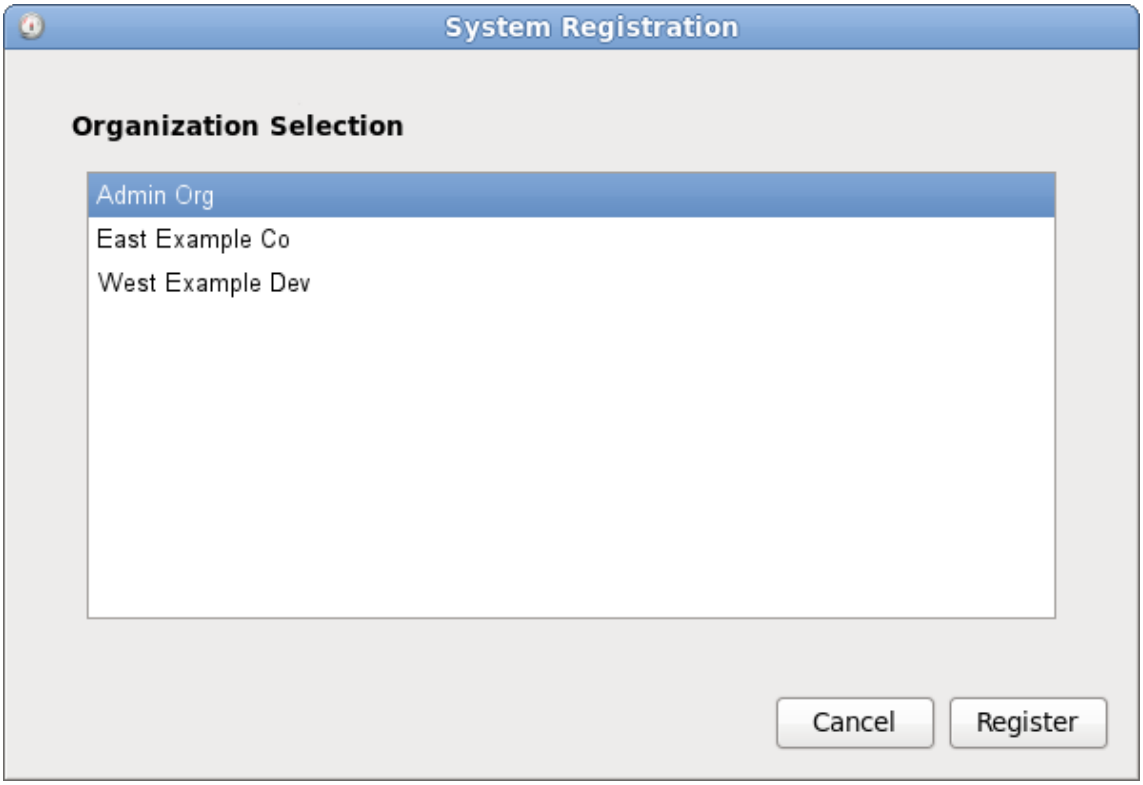


Figure 34.8. Organization Selection

7. If you decided to let Subscription Manager automatically attach subscriptions to the system (the default), then the system scans for the subscriptions to attach as part of the registration process.

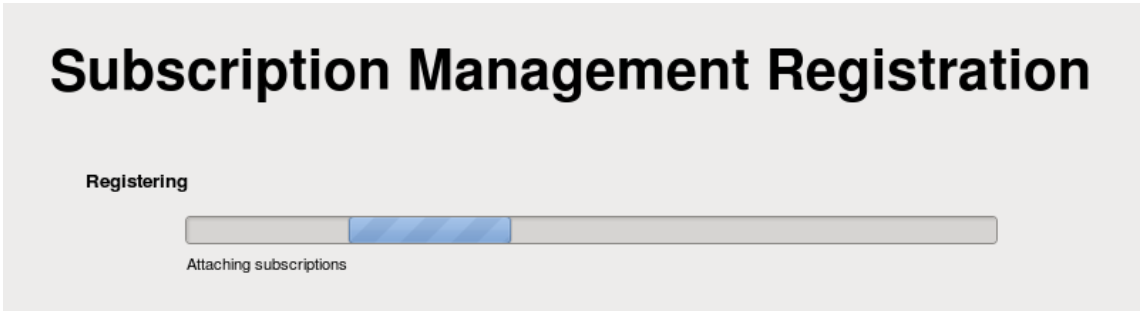


Figure 34.9. Auto-Selecting Subscriptions

When registration is complete, the Subscription Manager reports the applied service level for the system based on the information in the selected subscription and the specific subscription that has been attached to the new system. This subscription selection must be confirmed to complete the registration process.

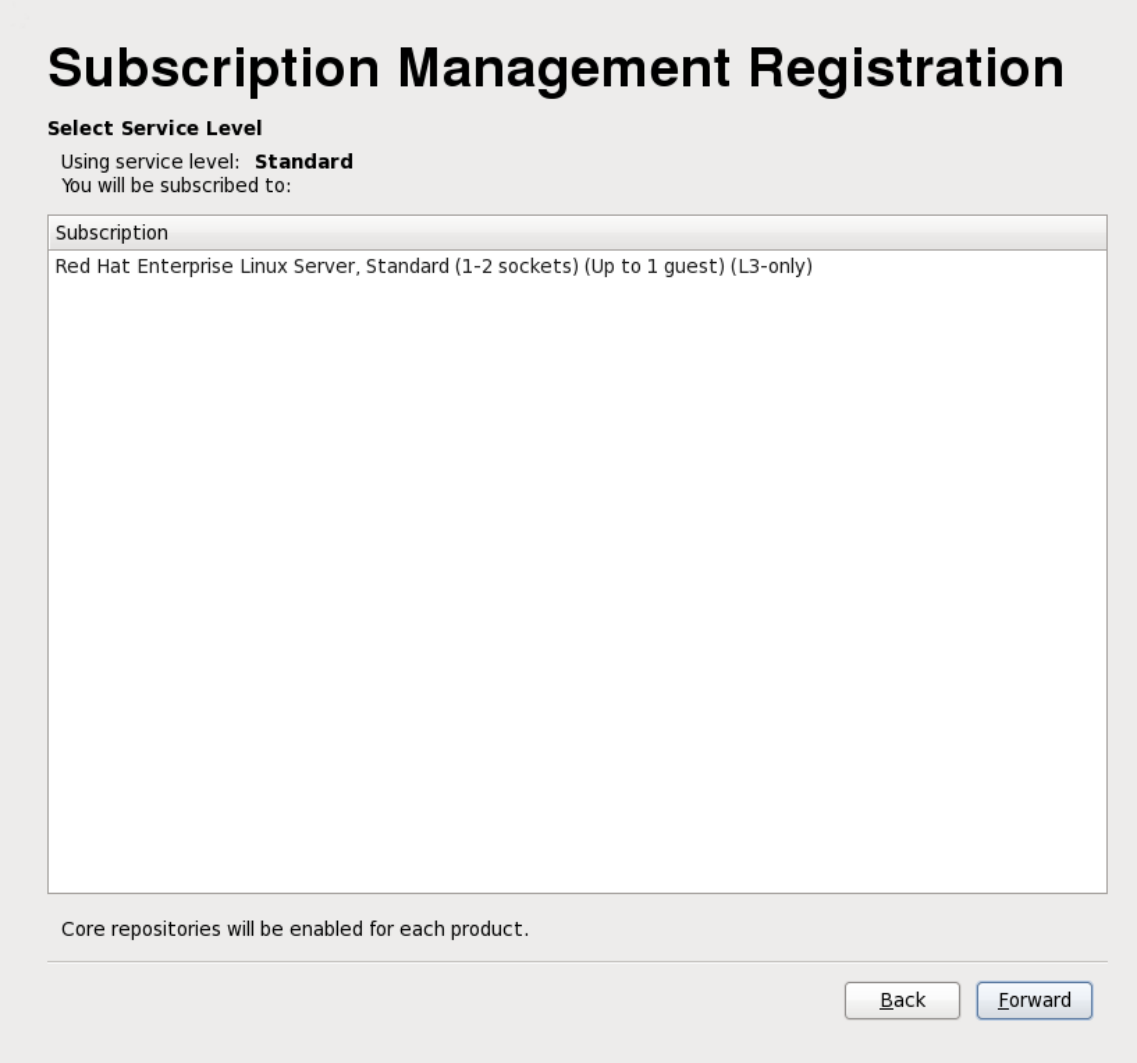


Figure 34.10. Confirm Subscription

If you selected to apply subscriptions later, then that part of the registration process is skipped, and the Subscription Manager screen in firstboot simply instructs you to attach subscriptions later.

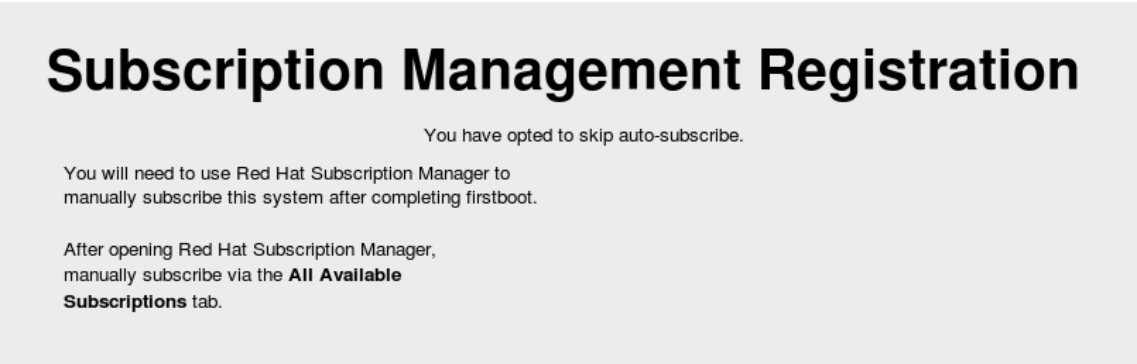


Figure 34.11. Note to Select Subscriptions Later

8. Click **Forward** to move to the next configuration area for firstboot, user setup.

34.3. CREATE USER

Create a user account for yourself with this screen. Always use this account to log in to your Red Hat Enterprise Linux system, rather than using the **root** account.

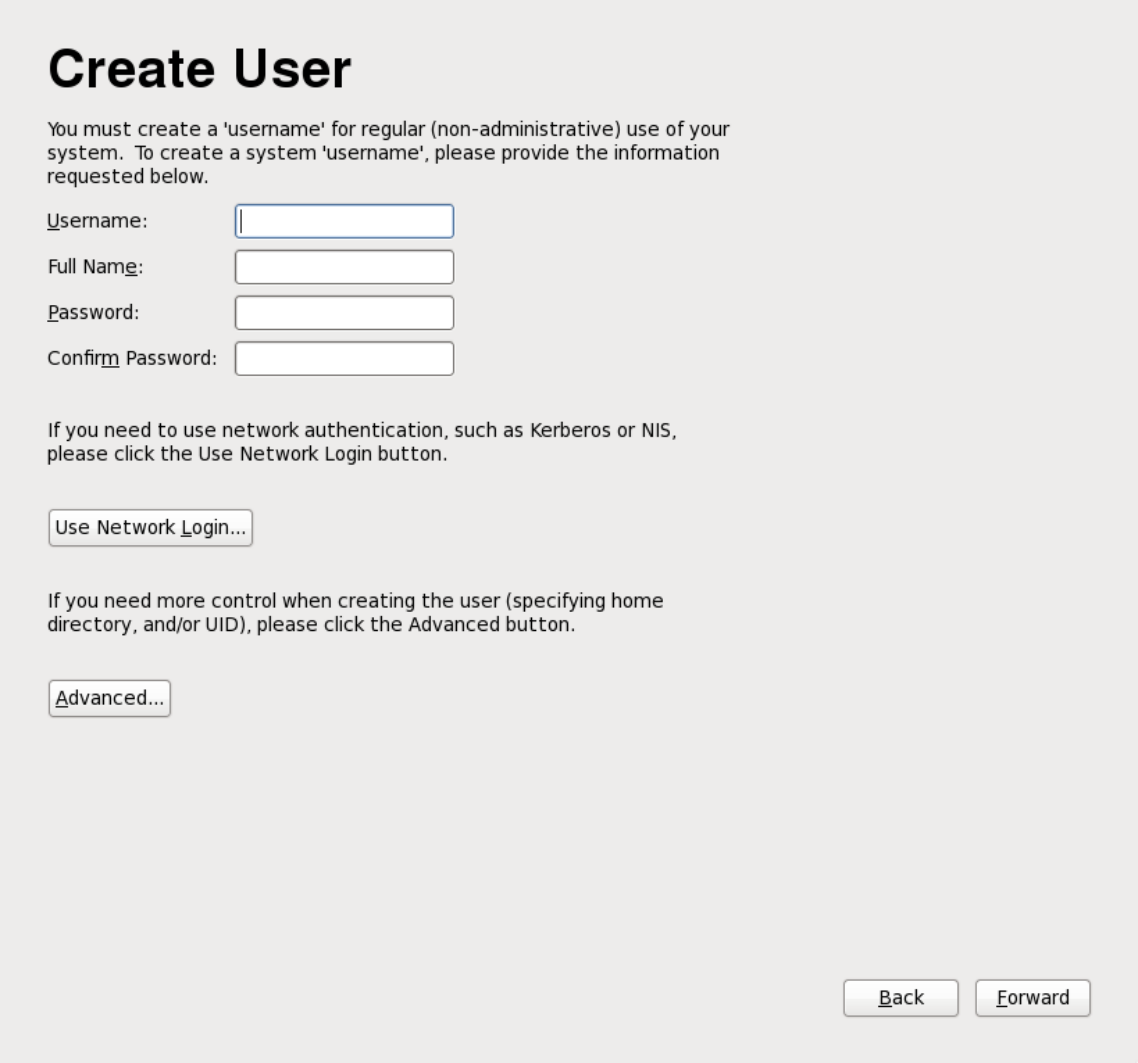


Figure 34.12. Firstboot create user screen

Enter a user name and your full name, and then enter your chosen password. Type your password once more in the **Confirm Password** box to ensure that it is correct.

To configure Red Hat Enterprise Linux to use network services for authentication of user information, click **Use Network Login**. Refer to Section 34.3.1, "Authentication Configuration" for further details.

IMPORTANT

If you do not create at least one user account in this step, you will not be able to log in to the Red Hat Enterprise Linux graphical environment. If you skipped this step during installation, refer to Section 10.4.2, "Booting into a Graphical Environment" .

NOTE

To add additional user accounts to your system after the installation is complete, choose **System → Administration → Users & Groups**

34.3.1. Authentication Configuration

If you clicked **Use Network Login** on the **Create User** screen, you must now specify how users are to be authenticated on the system. Use the drop-down menu to select from the following types of user database:

- **Local accounts only** (for use when the user database on the network is not accessible)
- **LDAP** (Lightweight Directory Access Protocol)
- **NIS** (Network Information Service)
- **Winbind** (for use with Microsoft Active Directory)

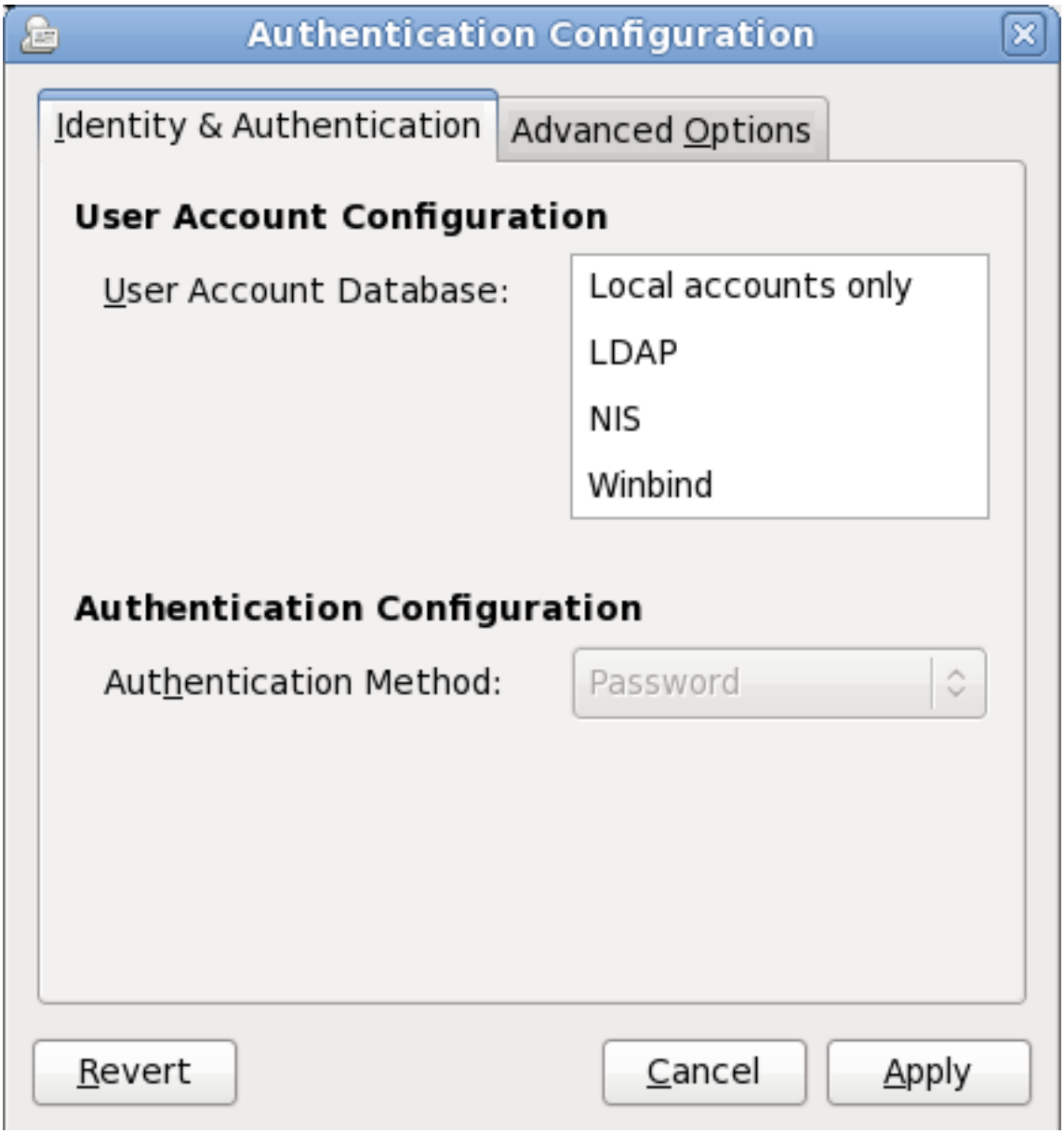


Figure 34.13. Firstboot Authentication Configuration screen

When you select the type of user database appropriate for your network, you must provide additional details relevant to that database type. For example, if you select **LDAP**, you must specify the *base distinguished name* for LDAP searches, and the address of the LDAP server. You must also select an **Authentication Method** relevant to the type of user database that you chose, for example, a Kerberos password, LDAP password, or NIS password.

The **Advanced Options** tab lets you enable other authentication mechanisms, including fingerprint readers, smart cards, and local access control in **/etc/security/access.conf**.

For more information, refer to *Authentication Configuration* in the Red Hat Enterprise Linux Deployment Guide.

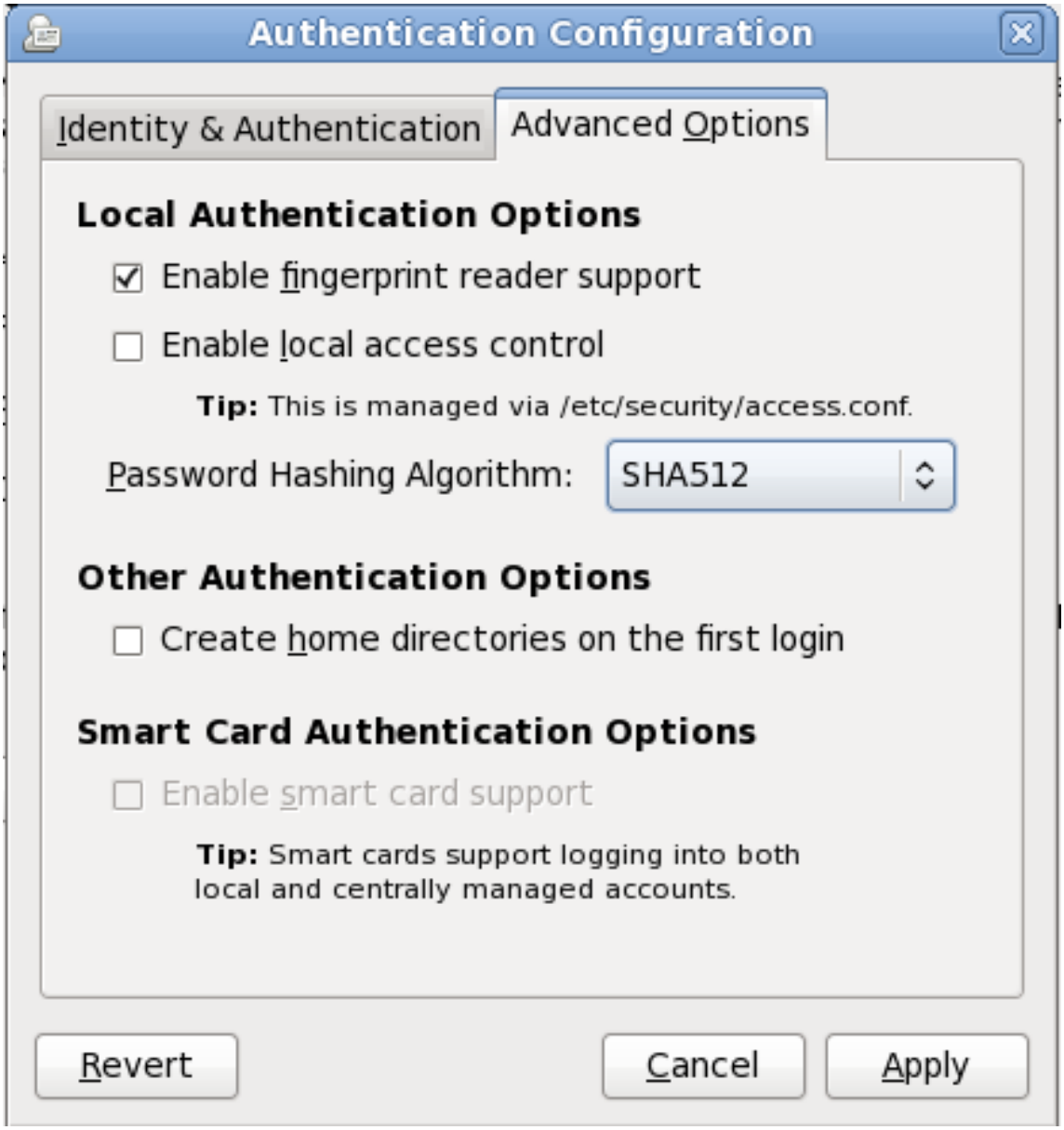


Figure 34.14. Firstboot authentication Advanced Options screen

34.4. DATE AND TIME

Use this screen to adjust the date and time of the system clock. To change these settings after installation, click **System → Administration → Date & Time**.

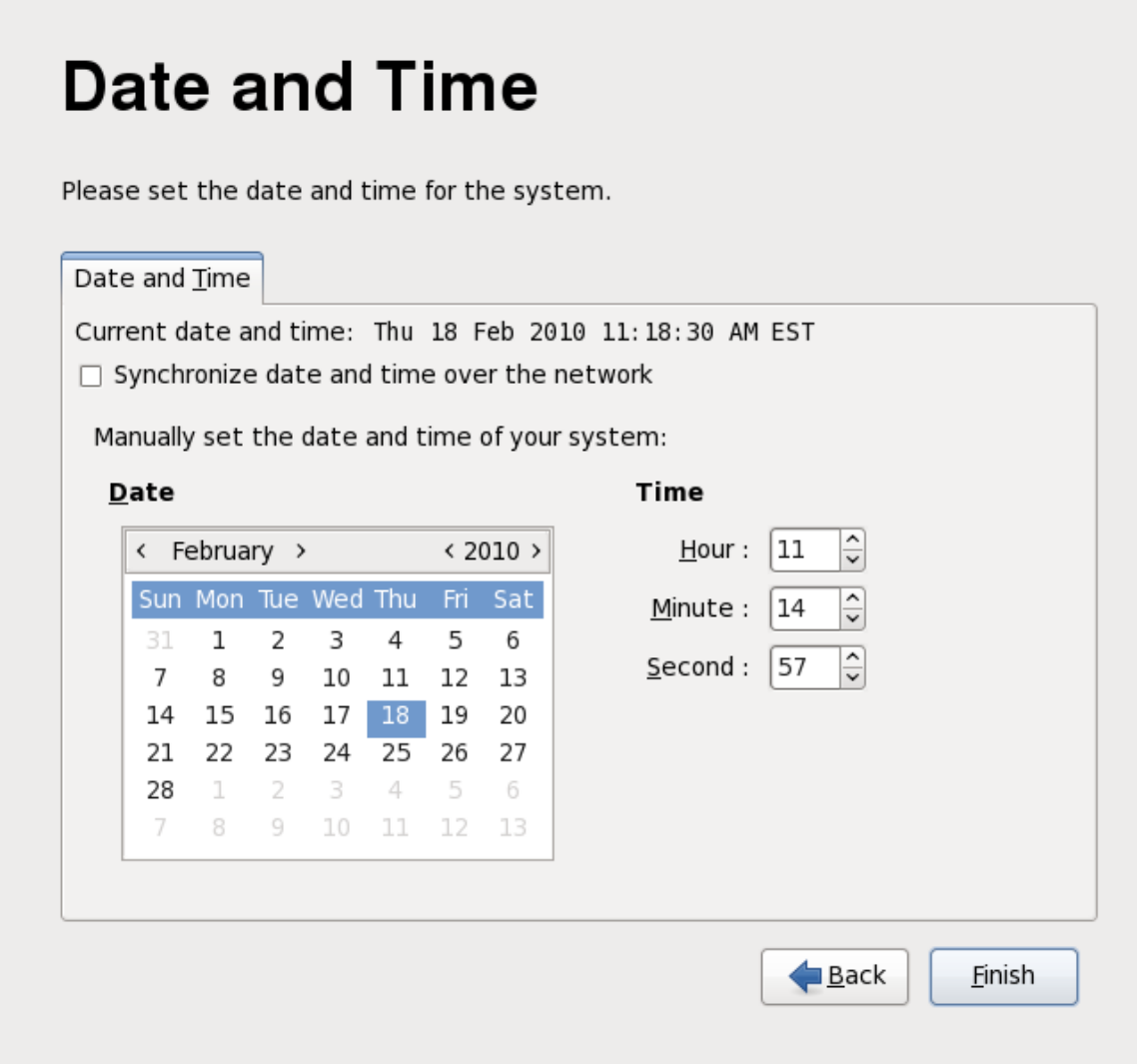


Figure 34.15. Firstboot date and time screen

Click the **Synchronize date and time over the network** checkbox to configure your system to use *Network Time Protocol* (NTP) servers to maintain the accuracy of the clock. NTP provides time synchronization service to computers on the same network. Many public NTP servers are available on the Internet.

34.5. KDUMP

Use this screen to select whether or not to use **Kdump** on this system. **Kdump** is a kernel crash dumping mechanism. In the event of a system crash, **Kdump** will capture information from your system that can be invaluable in determining the cause of the crash.

Note that if you select this option, you will need to reserve memory for **Kdump** and that this memory will not be available for any other purpose.

Figure 34.16. Kdump screen

If you do not want to use **Kdump** on this system, click **Finish**. If you want to use **Kdump**, select the **Enable kdump** option, then select an amount of memory to reserve for **Kdump** and click **Finish**.

Figure 34.17. Kdump enabled

[13] Systems can also be registered with Satellite or RHN Classic. For Satellite information, see the Satellite documentation. For information on using RHN Classic, see the appendix in the *Red Hat Subscription Management Guide*.

CHAPTER 35. YOUR NEXT STEPS

35.1. UPDATING YOUR SYSTEM

Red Hat releases updated software packages for Red Hat Enterprise Linux throughout the support period of each version. Updated packages add new features, improve reliability, resolve bugs, or remove security vulnerabilities. To ensure the security of your system, update regularly, and as soon as possible after Red Hat issues a security announcement.

35.1.1. Driver Update rpm Packages

Occasionally, when a new piece of hardware is not yet supported in the kernel that you have installed, Red Hat or a hardware vendor might make a driver update available. Although you can install driver updates during the installation process (refer to Chapter 6, *Updating Drivers During Installation on Intel and AMD Systems* for Intel and AMD systems and Chapter 13, *Updating Drivers During Installation on IBM Power Systems Servers* for IBM Power Systems servers) we recommend that you do this only for devices that are essential to carry out the installation. In all other cases, complete the installation first, and then add support for the device with a driver update rpm package as described in this section.

Do not install a driver update rpm unless you are certain that your system requires it. Installing a driver update on a system for which it was not intended can cause system difficulties.

To see a list of driver updates already installed on your system, click **System → Administration → Add/Remove Software** on your desktop, and enter the root password if prompted for it. Click the **Search** tab, enter the word **kmod-** (notice the final **-**) and click **Search**.

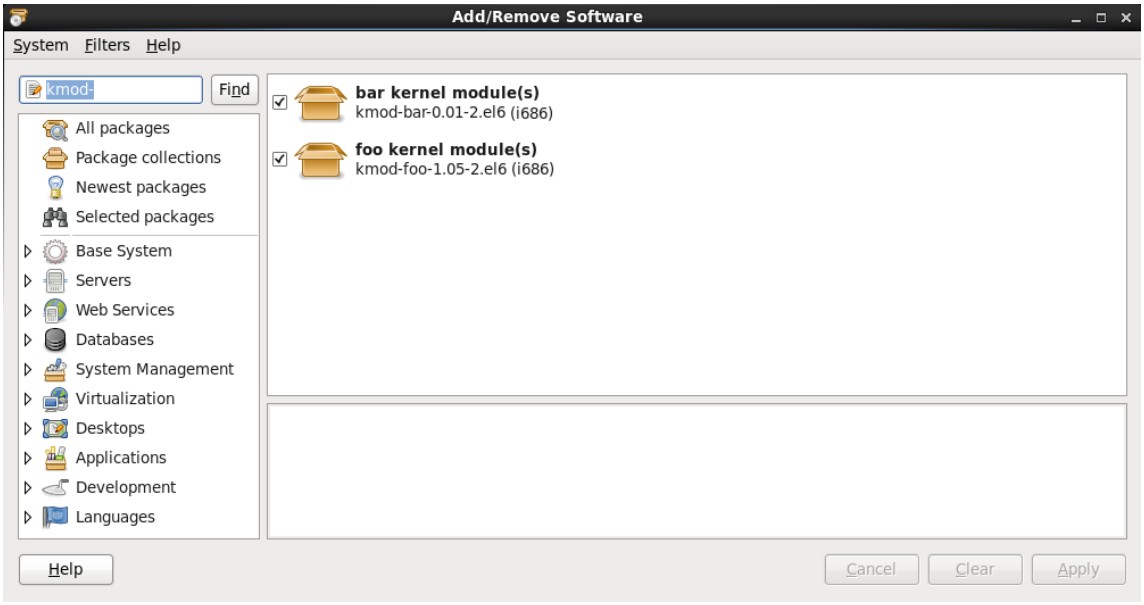


Figure 35.1. Listing Installed Driver Update RPM Packages

Alternatively, you can use the command line, as follows:

```
$ rpm -qa | egrep ^kmod-
```

Note the **-** on the end of **kmod**. This will list all installed packages that begin with **kmod-**, which should include all driver updates that are currently installed on your system. Additional drivers provided by third-party update software are not listed in this output. Contact the third-party vendor for details.

To install a new driver update rpm package:

1. Download the driver update rpm package from the location specified by Red Hat or your hardware vendor. The package file name will begin with **kmod** (short for *kernel module*) and have a form similar to this example:

   kmod-foo-1.05-2.el6.9.i686

   In the example, the driver update rpm package supplies a driver update named **foo** with version number 1.05-2 for Red Hat Enterprise Linux 6.9, on i686 systems.

   Driver update rpm packages are signed packages, and like all other software packages, they are automatically validated at install time. To perform this step manually, type the following at a command line:

   ```
   $ rpm --checksig -v filename.rpm
   ```

   where *filename.rpm* is the driver update rpm package file name. This verifies the package against using the standard Red Hat GPG package signing key that is already installed on any Red Hat Enterprise Linux 6.9 system. If you need this key for verification purposes on another system, you can can obtain it from: https://access.redhat.com/security/team/key/

2. Locate and double-click the file that you downloaded. The system might prompt you for the root password, after which it will present the following **Installing Packages** box:

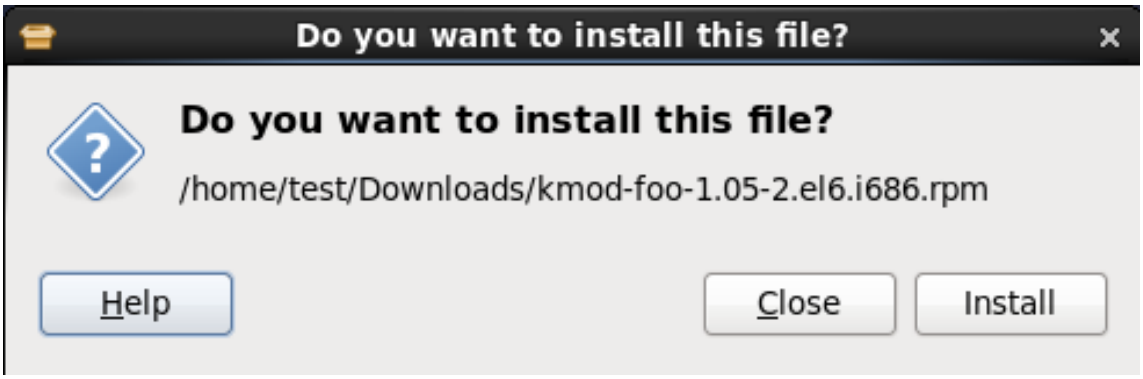
   

   Figure 35.2. The installing packages box

   Click **Apply** to complete the package installation.

   Alternatively, you can install a driver update manually on the command line:

   ```
   $ rpm -ivh kmod-foo-1.05-2.el6.9.i686
   ```

3. Whether you used a graphical install, or a command line install, reboot your system to ensure your system is using the new driver.

If Red Hat ships a kernel errata update before the next release of Red Hat Enterprise Linux, your system will continue to use the driver updates that you have installed. There is no need to re-install driver updates following an errata update. Generally, when Red Hat releases a new version of Red Hat Enterprise Linux, all driver updates for the previous version are incorporated in the new version. However, if it was not possible to include a particular driver, you will need to perform another driver update when you install the new version of Red Hat Enterprise Linux. In this case, Red Hat or your hardware party vendor will inform you of the location of the update.

35.2. FINISHING AN UPGRADE

IMPORTANT

Once you have rebooted your system after performing an upgrade, you should also perform a manual system update. Consult Section 35.1, “Updating Your System” for more information.

If you chose to upgrade your system from a previous release rather than perform a fresh installation, you may want to examine the differences in the package set. Section 9.12.2, “ Upgrading Using the Installer ” , Section 16.14.2, “ Upgrading Using the Installer ” , or Section 23.12.1, “ Upgrading Using the Installer ” (depending on your system architecture) advised you to create a package listing for your original system. You can now use that listing to determine how to bring your new system close to the original system state.

Most software repository configurations are stored in packages that end with the term **release**. Check the old package list for the repositories that were installed:

```
awk '{print $1}' ~/old-pkglist.txt | grep 'release$'
```

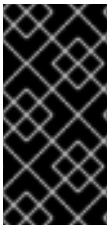
If necessary, retrieve and install these packages from their original sources on the Internet. Follow the instructions at the originating site to install the repository configuration packages for use by **yum** and other software management tools on your Red Hat Enterprise Linux system.

Then run the following commands to make a list of other missing software packages:

```
awk '{print $1}' ~/old-pkglist.txt | sort | uniq > ~/old-pkgnames.txt
rpm -qa --qf '%{NAME}\n' | sort | uniq > ~/new-pkgnames.txt
diff -u ~/old-pkgnames.txt ~/new-pkgnames.txt | grep '^-' | sed 's/^-//' > /tmp/pkgs-to-install.txt
```

Now use the file **/tmp/pkgs-to-install.txt** with the **yum** command to restore most or all of your old software:

```
su -c 'yum install `cat /tmp/pkgs-to-install.txt`'
```



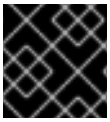
IMPORTANT

Due to changes in package complements between Red Hat Enterprise Linux releases, it is possible this method may not restore all the software on your system. You can use the routines above to again compare the software on your system, and remedy any problems you find.

35.3. SWITCHING TO A GRAPHICAL LOGIN

IMPORTANT

To switch to a graphical environment, you might need to install extra software from a *repository*. You can access Red Hat Enterprise Linux repositories with your Red Hat Network subscription through the Internet or use a Red Hat Enterprise Linux installation DVD as a repository. Refer to Section 35.3.1, “Enabling Access to Software Repositories from the Command Line”.



IMPORTANT

To use a graphical user interface on System z, use vncserver instead.

If you installed using a text login and wish to switch to a graphical login, follow this procedure.

1. If you are not already root, switch users to the **root** account:

   ```
   su -
   ```

   Provide the administrator password when prompted.

2. If you have not already done so, install the **X Window System** and a graphical desktop environment. For example, to install the GNOME desktop environment, use this command:

   ```
   yum groupinstall "X Window System" Desktop
   ```

   To install the KDE desktop environment, use:

   ```
   yum groupinstall "X Window System" "KDE Desktop"
   ```

   This step may take some time as your Red Hat Enterprise Linux system downloads and installs additional software. You may be asked to provide the installation media depending on your original installation source.

3. Run the following command to edit the **/etc/inittab** file:

   ```
   vi /etc/inittab
   ```

4. Press the **I** key to enter **insert** mode.

5. Find the line that includes the text **initdefault**. Change the numeral **3** to **5**.

6. Type **:wq** and press the **Enter** key to save the file and exit the **vi** text editor.

Reboot the system using the **reboot** command. Your system will restart and present a graphical login.

If you encounter any problems with the graphical login, refer to Chapter 10, *Troubleshooting Installation on an Intel or AMD System*.

35.3.1. Enabling Access to Software Repositories from the Command Line

The usual way to install new software on a Red Hat Enterprise Linux system is through a software repository. You can access Red Hat Enterprise Linux repositories through the Internet with your Red Hat Network subscription, or use a Red Hat Enterprise Linux installation DVD as a repository. The software that you access through online repositories is more up-to-date than what is available on an installation DVD. Furthermore, configuring a Red Hat Enterprise Linux system to access online repositories is generally easier than configuring the system to use an installation DVD as a repository, as long as you have an existing, wired network connection available.

35.3.1.1. Enabling Access to Software Repositories Through the Internet

If you supplied your Red Hat Network subscription number during the installation process, your system is already configured to access Red Hat Enterprise Linux repositories through the Internet. Therefore, all

you must do is ensure that the system can access the Internet. If you have an existing, wired network connection available, this process is straightforward:

1. If you are not already root, switch users to the **root** account:

   ```
   su -
   ```

2. Ensure that the system is plugged into your network. Note that your network might be as small as two devices — a computer and an external modem/router.

3. Run **system-config-network**. The network configuration tool starts and displays the **Select Action** screen.

4. Select **Device configuration** and press **Enter**. The network configuration tool displays the **Select A Device** screen with a list of network interfaces present on your system. The first interface is named **eth0** by default.

5. Select a network interface to configure and press **Enter**. The network configuration tool takes you to the **Network Configuration** screen.

6. You can manually configure a static IP, gateway, and DNS servers on this screen or leave these fields blank to accept the default values. When you have chosen a configuration, select **OK**, and press **Enter**. The network configuration tool takes you back to the **Select A Device** screen.

7. Select **Save** and press **Enter**. The network configuration tool takes you back to the **Select Action** screen.

8. Select **Save&Quit** and press **Enter**. The network configuration tool saves your settings and exits.

9. Run **ifup *interface***, where *interface* is the network interface that you configured with the network configuration tool. For example, run **ifup eth0** to start **eth0**.

Configuration of dial-up or wireless Internet connections is more complicated and beyond the scope of this guide.

35.3.1.2. Using a Red Hat Enterprise Linux Installation DVD as a Software Repository

To use a Red Hat Enterprise Linux installation DVD as a software repository, either in the form of a physical disc, or in the form of an ISO image file.

1. If you are using a physical DVD, insert the disc into your computer.

2. If you are not already root, switch users to the **root** account:

   ```
   su -
   ```

3. Create a *mount point* for the repository:

   ```
   mkdir -p /path/to/repo
   ```

   where */path/to/repo* is a location for the repository, for example, **/mnt/repo**

4. *Mount* the DVD on the mount point that you just created. If you are using a physical disc, you need to know the *device name* of your DVD drive. You can find the names of any CD or DVD

drives on your system with the command **cat /proc/sys/dev/cdrom/info**. The first CD or DVD drive on the system is typically named **sr0**. When you know the device name, mount the DVD:

```
mount -r -t iso9660 /dev/device_name /path/to/repo
```

For example: **mount -r -t iso9660 /dev/sr0 /mnt/repo**

If you are using an ISO image file of a disc, mount the image file like this:

```
mount -r -t iso9660 -o loop /path/to/image/file.iso /path/to/repo
```

For example: **mount -r -o loop /home/root/Downloads/RHEL6.9-Server-i386-DVD.iso /mnt/repo**

Note that you can only mount an image file if the storage device that holds the image file is itself mounted. For example, if the image file is stored on a hard drive that is not mounted automatically when the system boots, you must mount the hard drive before you mount an image file stored on that hard drive. Consider a hard drive named **/dev/sdb** that is not automatically mounted at boot time and which has an image file stored in a directory named **Downloads** on its first partition:

```
mkdir /mnt/temp
mount /dev/sdb1 /mnt/temp
mkdir /mnt/repo
mount -r -t iso9660 -o loop mount -r -o loop /mnt/temp/Downloads/RHEL6.9-Server-i386-DVD.iso /mnt/repo
```

If you are not sure whether a storage device is mounted, run the **mount** command to obtain a list of current mounts. If you are not sure of the device name or partition number of a storage device, run **fdisk -l** and try to identify it in the output.

5. Create a new *repo file* in the **/etc/yum.repos.d/** directory. The name of the file is not important, as long as it ends in **.repo**. For example, **dvd.repo** is an obvious choice.

    1. Choose a name for the repo file and open it as a new file with the **vi** text editor. For example:

        ```
        vi /etc/yum.repos.d/dvd.repo
        ```

    2. Press the **I** key to enter **insert** mode.

    3. Supply the details of the repository. For example:

        ```
        [dvd]
        baseurl=file:///mnt/repo/Server
        enabled=1
        gpgcheck=1
        gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
        ```

        The name of the repository is specified in square brackets – in this example, **[dvd]**. The name is not important, but you should choose something that is meaningful and recognizable.

The line that specifies the **baseurl** should contain the path to the mount point that you created previously, suffixed with /**Server** for a Red Hat Enterprise Linux server installation DVD, or with /**Client** for a Red Hat Enterprise Linux client installation DVD.

4. Press the **Esc** key to exit **insert** mode.
5. Type **:wq** and press the **Enter** key to save the file and exit the **vi** text editor.
6. After installing or upgrading software from the DVD, delete the repo file that you created.

35.4. INSTALLING PACKAGES WITH YUM

The **yum** utility allows you to install packages on your system.

To install a single package and all of its non-installed dependencies, enter a command in the following form:

```
yum install package_name
```

If you are installing packages on a *multilib* system, such as an AMD64 or Intel64 machine, you can specify the architecture of the package (as long as it is available in an enabled repository) by appending *.arch* to the package name. For example, to install the foobar package for **i686**, type:

```
~]# yum install foobar.i686
```

To install packages when your system cannot access a network or the Internet, consider enabling the installation DVD or ISO image file as an installation repository (refer to Section 35.3.1.2, "Using a Red Hat Enterprise Linux Installation DVD as a Software Repository"). Choose the appropriate installation media if you intend to install packages for a different architecture. For example, to install a 32-bit package on a 64-bit system, enable the 32-bit media as an installation repository.

For more information on installing packages, refer to the *Yum* chapter in the Red Hat Enterprise Linux Deployment Guide.

35.5. AUTOMATING THE INITIAL CONFIGURATION OF CLOUD INSTANCES USING CLOUD-INIT

For the initial configuration of cloud instances, you can use the cloud-init package. On a new cloud instance, **cloud-init** can automatically:

- set the default locale
- configure the host name
- configure network interfaces
- generate private SSH keys
- add SSH keys to the user's **.ssh/authorized_keys** directory
- set up ephemeral mount points

Cloud-init is used with Red Hat's cloud products. See documentation on using **cloud-init** with Red Hat products:

- Red Hat Enterprise Linux Atomic Host 7 Installation and Configuration Guide
- Red Hat OpenStack Platform 8 Instances and Images Guide
- Red Hat Enterprise Virtualization Virtual Machine Management Guide
- Red Hat CloudForms Provisioning Virtual Machines and Hosts Guide

See also upstream cloud-init documentation

CHAPTER 36. BASIC SYSTEM RECOVERY

When things go wrong, there are ways to fix problems. However, these methods require that you understand the system well. This chapter describes how to boot into rescue mode, single-user mode, and emergency mode, where you can use your own knowledge to repair the system.

36.1. RESCUE MODE

36.1.1. Common Problems

You might need to boot into one of these recovery modes for any of the following reasons:

- You are unable to boot normally into Red Hat Enterprise Linux (runlevel 3 or 5).
- You are having hardware or software problems, and you want to get a few important files off of your system's hard drive.
- You forgot the root password.

36.1.1.1. Unable to Boot into Red Hat Enterprise Linux

This problem is often caused by the installation of another operating system after you have installed Red Hat Enterprise Linux. Some other operating systems assume that you have no other operating system(s) on your computer. They overwrite the Master Boot Record (MBR) that originally contained the GRUB boot loader. If the boot loader is overwritten in this manner, you cannot boot Red Hat Enterprise Linux unless you can get into rescue mode and reconfigure the boot loader.

Another common problem occurs when using a partitioning tool to resize a partition or create a new partition from free space after installation, and it changes the order of your partitions. If the partition number of your / partition changes, the boot loader might not be able to find it to mount the partition. To fix this problem, boot in rescue mode and modify the **/boot/grub/grub.conf** file.

For instructions on how to reinstall the GRUB boot loader from a rescue environment, refer to Section 36.1.2.1, “Reinstalling the Boot Loader”.

36.1.1.2. Hardware/Software Problems

This category includes a wide variety of different situations. Two examples include failing hard drives and specifying an invalid root device or kernel in the boot loader configuration file. If either of these occur, you might not be able to reboot into Red Hat Enterprise Linux. However, if you boot into one of the system recovery modes, you might be able to resolve the problem or at least get copies of your most important files.

36.1.1.3. Root Password

What can you do if you forget your root password? To reset it to a different password, boot into rescue mode or single-user mode, and use the **passwd** command to reset the root password.

36.1.2. Booting into Rescue Mode

Rescue mode provides the ability to boot a small Red Hat Enterprise Linux environment entirely from CD-ROM, or some other boot method, instead of the system's hard drive.

As the name implies, rescue mode is provided to rescue you from something. During normal operation, your Red Hat Enterprise Linux system uses files located on your system's hard drive to do everything — run programs, store your files, and more.

However, there may be times when you are unable to get Red Hat Enterprise Linux running completely enough to access files on your system's hard drive. Using rescue mode, you can access the files stored on your system's hard drive, even if you cannot actually run Red Hat Enterprise Linux from that hard drive.

To boot into rescue mode, you must be able to boot the system using one of the following methods[14]:

- By booting the system from a boot CD-ROM or DVD.
- By booting the system from other installation boot media, such as USB flash devices.
- By booting the system from the Red Hat Enterprise Linux installation DVD.

Once you have booted using one of the described methods, add the keyword **rescue** as a kernel parameter. For example, for an x86 system, type the following command at the installation boot prompt:

```
linux rescue
```

If your system requires a third-party driver provided on a *driver disc* to boot, load the driver with the additional option **dd**:

```
linux rescue dd
```

For more information on using a driver disc at boot time, refer to Section 6.3.3, "Use a Boot Option to Specify a Driver Update Disk" for x86 systems or Section 13.3.3, "Use a Boot Option to Specify a Driver Update Disk" for Power Systems servers.

If a driver that is part of the Red Hat Enterprise Linux 6.9 distribution prevents the system from booting, blacklist that driver with the **rdblacklist** option. For example, to boot into rescue mode without the foobar driver, run:

```
linux rescue rdblacklist=foobar
```

You are prompted to answer a few basic questions, including which language to use. It also prompts you to select where a valid rescue image is located. Select from **Local CD-ROM**, **Hard Drive**, **NFS image**, **FTP**, or **HTTP**. The location selected must contain a valid installation tree, and the installation tree must be for the same version of Red Hat Enterprise Linux as the Red Hat Enterprise Linux disk from which you booted. If you used a boot CD-ROM or other media to start rescue mode, the installation tree must be from the same tree from which the media was created. For more information about how to setup an installation tree on a hard drive, NFS server, FTP server, or HTTP server, refer to the earlier section of this guide.

If you select a rescue image that does not require a network connection, you are asked whether or not you want to establish a network connection. A network connection is useful if you need to backup files to a different computer or install some RPM packages from a shared network location, for example.

The following message is displayed:

> The rescue environment will now attempt to find your Linux installation and mount it under the directory /mnt/sysimage. You can then make any changes required to your system. If you want to proceed with this step choose 'Continue'. You can also choose to

mount your file systems read-only instead of read-write by choosing 'Read-only'. If for some reason this process fails you can choose 'Skip' and this step will be skipped and you will go directly to a command shell.

If you select **Continue**, it attempts to mount your file system under the directory **/mnt/sysimage/**. If it fails to mount a partition, it notifies you. If you select **Read-Only**, it attempts to mount your file system under the directory **/mnt/sysimage/**, but in read-only mode. If you select **Skip**, your file system is not mounted. Choose **Skip** if you think your file system is corrupted.

Once you have your system 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-3.00b#
```

If you selected **Continue** to mount your partitions automatically and they were mounted successfully, you are in single-user mode.

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 (runlevel 3 or 5). 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:

```
chroot /mnt/sysimage
```

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.

If you selected **Skip**, you can still try to mount a partition or LVM2 logical volume manually inside rescue mode by creating a directory such as ***/foo***, and typing the following command:

```
mount -t ext4 /dev/mapper/VolGroup00-LogVol02 /foo
```

In the above command, ***/foo*** is a directory that you have created and ***/dev/mapper/VolGroup00-LogVol02*** is the LVM2 logical volume you want to mount. If the partition is of type **ext2** or **ext3** replace **ext4** with **ext2** or **ext3** respectively.

If you do not know the names of all physical partitions, use the following command to list them:

```
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, respectively.

From the prompt, you can run many useful commands, such as:

- **ssh**, **scp**, and **ping** if the network is started
- **dump** and **restore** for users with tape drives
- **parted** and **fdisk** for managing partitions
- **rpm** for installing or upgrading software
- **vi** for editing text files

36.1.2.1. Reinstalling the Boot Loader

In many cases, the GRUB boot loader can mistakenly be deleted, corrupted, or replaced by other operating systems.

The following steps detail the process on how GRUB is reinstalled on the master boot record:

- Boot the system from an installation boot medium.
- Type **linux rescue** at the installation boot prompt to enter the rescue environment.
- Type **chroot /mnt/sysimage** to mount the root partition.
- Type **/sbin/grub-install *bootpart*** to reinstall the GRUB boot loader, where *bootpart* is the boot partition (typically, /dev/sda).
- Review the **/boot/grub/grub.conf** file, as additional entries may be needed for GRUB to control additional operating systems.
- Reboot the system.

36.1.3. Booting into Single-User Mode

One of the advantages of single-user mode is that you do not need a boot CD-ROM; however, it does not give you the option to mount the file systems as read-only or not mount them at all.

If your system boots, but does not allow you to log in when it has completed booting, try single-user mode.

In single-user mode, your computer boots to runlevel 1. Your local file systems are mounted, but your network is not activated. You have a usable system maintenance shell. Unlike rescue mode, single-user mode automatically tries to mount your file system. *Do not use single-user mode if your file system cannot be mounted successfully.* You cannot use single-user mode if the runlevel 1 configuration on your system is corrupted.

On an x86 system using GRUB, use the following steps to boot into single-user mode:

1. At the GRUB splash screen at boot time, press any key to enter the GRUB interactive menu.
2. Select **Red Hat Enterprise Linux** with the version of the kernel that you wish to boot and type **a** to append the line.
3. Go to the end of the line and type **single** as a separate word (press the **Spacebar** and then type **single**). Press **Enter** to exit edit mode.

36.1.4. Booting into Emergency Mode

In emergency mode, you are booted into the most minimal environment possible. The root file system is mounted read-only and almost nothing is set up. The main advantage of emergency mode over single-user mode is that the **init** files are not loaded. If **init** is corrupted or not working, you can still mount file systems to recover data that could be lost during a re-installation.

To boot into emergency mode, use the same method as described for single-user mode in Section 36.1.3, “Booting into Single-User Mode” with one exception, replace the keyword **single** with the keyword **emergency**.

36.2. RESCUE MODE ON POWER SYSTEMS SERVERS

You can use the installation disks in rescue mode, in case your system does not boot. Rescue mode gives you access to the disk partitions on your system so you can make any changes necessary to rescue your installation.

After the Language Selection screen (Section 15.2, "Language Selection"), the installation program attempts to mount the disk partitions on your system. It then presents you with a shell prompt where you can make the changes you need. These changes may include storing the kernel and command line into the IPL source, as described in the Installation Complete section (Section 16.21, "Installation Complete").

When your changes are complete, you can exit the shell using **exit 0**. This causes a reboot from the C side. To reboot from the A or B side or from *NWSSTG, you should vary off the system instead of exiting the shell.

36.2.1. Special Considerations for Accessing the SCSI Utilities from Rescue Mode

If your system uses Native DASD disks, you may need access to the SCSI utilities from rescue mode. These utilities are located on the driver disc CD. The driver disc CD cannot be mounted from rescue mode unless special steps are taken. These steps are described below.

If you have a second CD-ROM drive assigned to your Linux system, you can mount the driver disc CD in the second drive.

If you have only one CD-ROM drive, you must set up an NFS boot, using the following steps:

1. Boot from the CD-ROM with the **linux rescue askmethod** command. This allows you to manually select NFS as the source of your rescue media instead of defaulting to the CD-ROM drive.

2. Copy the first installation disc onto a file system of another Linux system.

3. Make this copy of the installation disc available through NFS or FTP.

4. Vary off or power down the system you need to rescue. Set its IPL parameters as instructed for booting the Installation discs in rescue mode, except that the IPL source should point to the copy of **boot.img** on your IFS (from step 1, above).

5. *Make sure the installation disc is not in your DVD drive.*

6. IPL the Linux system.

7. Follow the prompts as described in Section 36.2, "Rescue Mode on Power Systems servers" . An additional prompt for the installation source appears. Select NFS or FTP (as appropriate) and complete the following network configuration screen.

8. When the Linux system has booted into rescue mode, the CD-ROM drive is available for use and you can mount the driver media to access the SCSI utilities.

36.3. USING RESCUE MODE TO FIX OR WORK AROUND DRIVER PROBLEMS

A malfunctioning or missing driver can prevent a system from booting normally. Rescue mode provides an environment in which you can add, remove, or replace a driver even when the system fails to boot. Wherever possible, we recommend that you use the **RPM** package manager to remove malfunctioning

drivers or to add updated or missing drivers. If you cannot remove a malfunctioning driver for some reason, you can instead *blacklist* the driver so that it does not load at boot time.

Note that 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.

36.3.1. Using RPM to Add, Remove, or Replace a Driver

In rescue mode, you can use **RPM** to install, remove, or update packages from the installed system, even though you did not boot the installed system. To remove a malfunctioning driver:

1. Boot the system into rescue mode with the **linux rescue** command at the boot prompt, or the **linux rescue dd** command if you need to load a third-party driver from a driver disc. Follow the instructions in Section 36.1.2, "Booting into Rescue Mode" and do *not* choose to mount the installed system as read only.

2. Change the root directory to **/mnt/sysimage/**:

```
chroot /mnt/sysimage/
```

3. Use the **rpm -e** command to remove the driver package. For example, to remove the kmod-foobar driver package, run:

```
rpm -e kmod-foobar
```

4. Exit the chroot environment:

```
exit
```

Installing a driver is a similar process, but the RPM package that contains the driver must be available on the system.

1. Boot the system into rescue mode with the **linux rescue** command at the boot prompt, or the **linux rescue dd** command if you need to load a third-party driver from a driver disc. Follow the instructions in Section 36.1.2, "Booting into Rescue Mode" and do *not* choose to mount the installed system as read only.

2. 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/sysimage/**, for example: **/mnt/sysimage/root/drivers/**.

3. Change the root directory to **/mnt/sysimage/**:

```
chroot /mnt/sysimage/
```

4. Use the **rpm -ivh** command to install the driver package. For example, to install the kmod-foobar driver package from **/root/drivers/**, run:

```
rpm -ivh /root/drivers/kmod-foobar-1.2.04.17.el6.i686
```

Note that **/root/drivers/** in this chroot environment is **/mnt/sysimage/root/drivers/** in the original rescue environment.

When you have finished removing and installing drivers, reboot the system.

36.3.2. Blacklisting a Driver

As described in Section 36.1.2, "Booting into Rescue Mode", the **rdblacklist** kernel option *blacklists* a driver at boot time. To continue to blacklist the driver on subsequent boots, add the **rdblacklist** option to the line in **/boot/grub/grub.conf** that describes your kernel. To blacklist the driver when the root device is mounted, add a blacklist entry in a file under **/etc/modprobe.d/**.

1. Boot the system into rescue mode with the command **linux rescue rdblacklist=*name_of_driver***, where *name_of_driver* is the driver that you need to blacklist. Follow the instructions in Section 36.1.2, "Booting into Rescue Mode" and do *not* choose to mount the installed system as read only.

2. Open the **/mnt/sysimage/boot/grub/grub.conf** file with the **vi** text editor:

```
vi /mnt/sysimage/boot/grub/grub.conf
```

3. Identify the default kernel used to boot the system. Each kernel is specified in the **grub.conf** file with a group of lines that begins **title**. The default kernel is specified by the ***default*** parameter near the start of the file. A value of **0** refers to the kernel described in the first group of lines, a value of **1** refers to the kernel described in the second group, and higher values refer to subsequent kernels in turn.

4. Edit the **kernel** line of the group to include the option **rdblacklist=*name_of_driver***, where *name_of_driver* is the driver that you need to blacklist. For example, to blacklist the driver named **foobar**:

```
kernel /vmlinuz-2.6.32-71.18-2.el6.i686 ro root=/dev/sda1 rhgb quiet rdblacklist=foobar
```

5. Save the file and exit **vi**.

6. Create a new file under **/etc/modprobe.d/** that contains the command **blacklist *name_of_driver***. Give the file a descriptive name that will help you find it in future, and use the filename extension **.conf**. For example, to continue to blacklist the driver **foobar** when the root device is mounted, run:

```
echo "blacklist foobar" >> /mnt/sysimage/etc/modprobe.d/blacklist-foobar.conf
```

7. Reboot the system. You no longer need to supply **rdblacklist** manually as a kernel option until you next update the default kernel. If you update the default kernel before the problem with the driver has been fixed, you must edit **grub.conf** again to ensure that the faulty driver is not loaded at boot time.

[14] Refer to the earlier sections of this guide for more details.

CHAPTER 37. UPGRADING YOUR CURRENT SYSTEM

The procedure for performing an in-place upgrade on your current system is handled by the following utilities:

- The **Preupgrade Assistant**, which is a diagnostics utility that assesses your current system and identifies potential problems you might encounter during and/or after the upgrade.
- The **Red Hat Upgrade Tool** utility, which is used to upgrade a system from Red Hat Enterprise Linux to version 7.

The current documentation for testing this procedure can be found in the following Red Hat Knowledgebase article: https://access.redhat.com/site/solutions/637583

CHAPTER 38. UNREGISTERING FROM RED HAT SUBSCRIPTION MANAGEMENT SERVICES

A system can only be registered with one subscription service. If you need to change which service your system is registered with or need to delete the registration in general, then the method to unregister depends on which type of subscription service the system was originally registered with.

38.1. SYSTEMS REGISTERED WITH RED HAT SUBSCRIPTION MANAGEMENT

Several different subscription services use the same, certificate-based framework to identify systems, installed products, and attached subscriptions. These services are Customer Portal Subscription Management (hosted), Subscription Asset Manager (on-premise subscription service), and CloudForms System Engine (on-premise subscription and content delivery services). These are all part of *Red Hat Subscription Management*.

For all services within Red Hat Subscription Management, the systems are managed with the Red Hat Subscription Manager client tools.

To unregister a system registered with a Red Hat Subscription Management server, use the **unregister** command.

```
[root@server ~]# subscription-manager unregister --username=name
```

NOTE

This command must be run as root.

38.2. SYSTEMS REGISTERED WITH RHN CLASSIC

There is no command to specifically unregister a system which is registered with RHN Classic. To delete the registration locally, remove the file with the system ID assigned to the system when it was registered:

```
[root@server ~]# rm -rf /etc/sysconfig/rhn/systemid
```

NOTE

If the system is being unregistered in order to register it with Red Hat Subscription Management (Customer Portal Subscription Management, Subscription Asset Manager, or CloudForms System Engine), then instead of unregistering the system, use the **rhn-migrate-classic-to-rhsm** script to migrate the system and all its attached subscriptions to the specified Red Hat Subscription Management server.

Using the migration scripts is covered in the *Subscription Management Guide*.

38.3. SYSTEMS REGISTERED WITH SATELLITE

For a Satellite registration on the server, locate the system in the **Systems** tab and delete the profile.

CHAPTER 39. REMOVING RED HAT ENTERPRISE LINUX FROM X86-BASED SYSTEMS

WARNING

If you have data from Red Hat Enterprise Linux that you want to keep, back it up before you proceed. Write your data to CD, DVD, external hard disk, or other storage device.

As a precaution, also back up data from any other operating systems that are installed on the same computer. Mistakes do happen and can result in the loss of all your data.

If you back up data from Red Hat Enterprise Linux to be used later in another operating system, make sure that the storage medium or device is readable by that other operating system. For example, without extra third-party software, Microsoft Windows cannot read an external hard drive that you have formatted with Red Hat Enterprise Linux to use the ext2, ext3, or ext4 file system.

To uninstall Red Hat Enterprise Linux from your x86-based system, you must remove the Red Hat Enterprise Linux boot loader information from your master boot record (MBR) and remove any partitions that contain the operating system. The method for removing Red Hat Enterprise Linux from your computer varies, depending on whether Red Hat Enterprise Linux is the only operating system installed on the computer, or whether the computer is configured to dual-boot Red Hat Enterprise Linux and another operating system.

These instructions cannot cover every possible computer configuration. If your computer is configured to boot three or more operating systems, or has a highly-customized partition scheme, use the following sections as a general guide to partition removal with the various tools described. In these situations, you will also need to learn to configure your chosen bootloader. See Appendix E, *The GRUB Boot Loader* for a general introduction to the subject, but detailed instructions are beyond the scope of this document.

IMPORTANT

Fdisk, the disk partitioning tool provided with MS-DOS and Microsoft Windows, is unable to remove the file systems used by Red Hat Enterprise Linux. MS-DOS and versions of Windows prior to Windows XP (except for Windows 2000) have no other means of removing or modifying partitions. Refer to Section 39.3, "Replacing Red Hat Enterprise Linux with MS-DOS or Legacy Versions of Microsoft Windows" for alternative removal methods for use with MS-DOS and these versions of Windows.

39.1. RED HAT ENTERPRISE LINUX IS THE ONLY OPERATING SYSTEM ON THE COMPUTER

If Red Hat Enterprise Linux is the only operating system on your computer, use the installation media for the replacement operating system to remove Red Hat Enterprise Linux. Examples of installation media include the Windows XP installation CD, Windows Vista installation DVD, or the installation CD, CDs, or DVD of another Linux distribution.

Note that some manufacturers of factory-built computers pre-installed with Microsoft Windows do not supply the Windows installation CD or DVD with the computer. The manufacturer may instead have supplied their own "system restore disc", or have included software with the computer that allowed you to create your own "system restore disc" when you first started the computer. In some cases, the system restore software is stored on a separate partition on the system's hard drive. If you cannot identify the installation media for an operating system that was pre-installed on your computer, consult the documentation supplied with the machine, or contact the manufacturer.

When you have located the installation media for your chosen operating system:

1. Back up any data that you want to keep.
2. Shut down the computer.
3. Boot your computer with the installation disc for the replacement operating system.
4. Follow the prompts presented during the installation process. Windows, OS X, and most Linux installation discs allow you to manually partition your hard drive during the installation process, or will offer you the option to remove all partitions and start with a fresh partition scheme. At this point, remove any existing partitions that the installation software detects or allow the installer to remove the partitions automatically. "System restore" media for computers pre-installed with Microsoft Windows might create a default partition layout automatically without input from you.

WARNING

If your computer has system restore software stored on a partition on a hard drive, take care when removing partitions while installing an operating system from other media. Under these circumstances, you could destroy the partition holding the system restore software.

39.2. YOUR COMPUTER DUAL-BOOTS RED HAT ENTERPRISE LINUX AND ANOTHER OPERATING SYSTEM

If your computer is configured to dual-boot Red Hat Enterprise Linux and another operating system, removing Red Hat Enterprise Linux without removing the partitions containing the other operating system and its data is more complicated. Specific instructions for a number of operating systems are set out below. To keep neither Red Hat Enterprise Linux nor the other operating system, follow the steps described for a computer with only Red Hat Enterprise Linux installed: Section 39.1, “Red Hat Enterprise Linux is the Only Operating System on the Computer”

39.2.1. Your Computer Dual-boots Red Hat Enterprise Linux and a Microsoft Windows Operating System

39.2.1.1. Windows 2000, Windows Server 2000, Windows XP, and Windows Server 2003

WARNING

Once you commence this process, your computer may be left in an unbootable state until you complete the entire set of instructions. Carefully read the steps below before beginning the removal process. Consider opening these instructions on another computer or printing them so that you have access to them at all times during the process.

This procedure relies on the **Windows Recovery Console** that loads from the Windows installation disk, so you will not be able to complete the procedure without access to this disk. If you start this procedure and do not complete it, you could leave your computer in a condition where you cannot boot it. The "system restore disk" supplied with some factory-built computers that are sold with Windows pre-installed on them might not include the **Windows Recovery Console**.

During the process outlined in these instructions, the **Windows Recovery Console** will prompt you for the Administrator password for your Windows system. Do not follow these instructions unless you know the Administrator password for your system or are certain that an Administrator password has never been created, even by the computer manufacturer.

1. Remove the Red Hat Enterprise Linux partitions

    1. Boot your computer into your Microsoft Windows environment.

    2. Click **Start**>**Run...**, type **diskmgmt.msc** and press **Enter**. The **Disk Management** tool opens.

        The tool displays a graphical representation of your disk, with bars representing each partition. The first partition is usually labeled **NTFS** and corresponds to your **C:** drive. At least two Red Hat Enterprise Linux partitions will be visible. Windows will not display a file system type for these partitions, but may allocate drive letters to some of them.

    3. Right-click on one of the Red Hat Enterprise Linux partitions, then click **Delete Partition** and click **Yes** to confirm the deletion. Repeat this process for the other Red Hat Enterprise Linux partitions on your system. As you delete partitions, Windows labels the space on the hard drive previously occupied by those partitions as **unallocated**.

2. Enable Windows to use the space on your hard drive vacated by Red Hat Enterprise Linux (optional)

    NOTE

    This step is not required to remove Red Hat Enterprise Linux from your computer. However, if you skip this step, you will leave part of your hard drive's storage capacity unusable by Windows. Depending on your configuration, this might be a significant portion of the storage capacity of the drive.

    Decide whether to extend an existing Windows partition to use the extra space, or create a new Windows partition in that space. If you create new a Windows partition, Windows will allocate a new drive letter to it and will interact with it as if it is a separate hard drive.

Extending an Existing Windows Partition

NOTE

The **diskpart** tool used in this step is installed as part of the Windows XP and Windows 2003 operating systems. If you are performing this step on a computer running Windows 2000 or Windows Server 2000, you can download a version of **diskpart** for your operating system from the Microsoft website.

1. Click **Start**>**Run...**, type **diskpart** and press **Enter**. A command window appears.
2. Type **list volume** and press **Enter**. **Diskpart** displays a list of the partitions on your system with a volume number, its drive letter, volume label, filesystem type, and size. Identify the Windows partition that you would like to use to occupy the space vacated on your hard drive by Red Hat Enterprise Linux and take note of its volume number (for example, your Windows **C:** drive might be "Volume 0").
3. Type **select volume *N*** (where *N* is the volume number for the Windows partition that you want to extend) and press **Enter**. Now type **extend** and press **Enter**. **Diskpart** now extends your chosen partition to fill the remaining space on your hard drive. It will notify you when the operation is complete.

Adding a New Windows Partition

1. In the **Disk Management** window, right-click on disk space that Windows labels as **unallocated** and select **New Partition** from the menu. The **New Partition Wizard** starts.
2. Follow the prompts presented by the **New Partition Wizard**. If you accept the default options, the tool will create a new partition that fills all available space on the hard drive, assigns it the next available drive letter, and formats it with the NTFS file system.

3. Restore the Windows bootloader

   1. Insert the Windows installation disk and restart your computer. As your computer starts, the following message will appear on the screen for a few seconds:

      > Press any key to boot from CD

      Press any key while the message is still showing and the Windows installation software will load.

   2. When the **Welcome to Setup** screen appears, you can start the **Windows Recovery Console**. The procedure is slightly different on different versions of Windows:

      - On Windows 2000 and Windows Server 2000, press the **R** key, then the **C** key.
      - On Windows XP and Windows Server 2003, press the **R** key.

   3. The **Windows Recovery Console** scans your hard drives for Windows installations, and assigns a number to each one. It displays a list of Windows installations and prompts you to select one. Type the number corresponding to the Windows installation that you want to restore.

4. The **Windows Recovery Console** prompts you for the Administrator password for your Windows installation. Type the Administrator password and press the **Enter** key. If there is no administrator password for this system, press only the **Enter** key.

5. At the prompt, type the command **fixmbr** and press the **Enter**. The **fixmbr** tool now restores the Master Boot Record for the system.

6. When the prompt reappears, type **exit** and press the **Enter** key.

7. Your computer will restart and boot your Windows operating system.

39.2.1.2. Windows Vista and Windows Server 2008

WARNING

Once you commence this process, your computer may be left in an unbootable state until you complete the entire set of instructions. Carefully read the steps below before beginning the removal process. Consider opening these instructions on another computer or printing them so that you have access to them at all times during the process.

This procedure relies on the **Windows Recovery Environment** that loads from the Windows installation disk and you will not be able to complete the procedure without access to this disk. If you start this procedure and do not complete it, you could leave your computer in a condition where you cannot boot it. The "system restore disk" supplied with some factory-built computers that are sold with Windows pre-installed on them might not include the **Windows Recovery Environment**

1. Remove the Red Hat Enterprise Linux partitions

    1. Boot your computer into your Microsoft Windows environment.

    2. Click **Start** then type **diskmgmt.msc** into the **Start Search** box and press **Enter**. The **Disk Management** tool opens.

       The tool displays a graphical representation of your disk, with bars representing each partition. The first partition is usually labeled **NTFS** and corresponds to your **C:** drive. At least two Red Hat Enterprise Linux partitions will be visible. Windows will not display a file system type for these partitions, but may allocate drive letters to some of them.

    3. Right-click on one of the Red Hat Enterprise Linux partitions, then click **Delete Partition** and click **Yes** to confirm the deletion. Repeat this process for the other Red Hat Enterprise Linux partitions on your system. As you delete partitions, Windows labels the space on the hard drive previously occupied by those partitions as **unallocated**.

2. Enable Windows to use the space on your hard drive vacated by Red Hat Enterprise Linux (optional)

NOTE

This step is not required to remove Red Hat Enterprise Linux from your computer. However, if you skip this step, you will leave part of your hard drive's storage capacity unusable by Windows. Depending on your configuration, this might be a significant portion of the storage capacity of the drive.

Decide whether to extend an existing Windows partition to use the extra space, or create a new Windows partition in that space. If you create new a Windows partition, Windows will allocate a new drive letter to it and will interact with it as if it is a separate hard drive.

Extending an Existing Windows Partition

1. In the **Disk Management** window, right-click on the Windows partition that you want to extend and select **Extend Volume** from the menu. The **Extend Volume Wizard** opens.

2. Follow the prompts presented by the **Extend Volume Wizard**. If you accept the defaults that it offers you, the tool will extend the selected volume to fill all available space on the hard drive.

Adding a New Windows Partition

1. In the **Disk Management** window, right-click on disk space that Windows labels as **unallocated** and select **New Simple Volume** from the menu. The **New Simple Volume Wizard** starts.

2. Follow the prompts presented by the **New Simple Volume Wizard** If you accept the default options, the tool will create a new partition that fills all available space on the hard drive, assigns it the next available drive letter, and formats it with the NTFS file system.

3. Restore the Windows bootloader

   1. Insert the Windows installation disk and restart your computer. As your computer starts, the following message will appear on the screen for a few seconds:

      > Press any key to boot from CD or DVD

      Press any key while the message is still showing and the Windows installation software will load.

   2. In the **Install Windows** dialog, select a language, time and currency format, and keyboard type. Click **Next**

   3. Click **Repair your computer**.

   4. The **Windows Recovery Environment** (WRE) shows you the Windows installations that it can detect on your system. Select the installation that you want to restore, then click **Next**.

   5. Click **Command prompt**. A command window will open.

   6. Type **bootrec /fixmbr** and press **Enter**.

   7. When the prompt reappears, close the command window, then click **Restart**.

   8. Your computer will restart and boot your Windows operating system.

39.2.2. Your computer dual-boots Red Hat Enterprise Linux and a different Linux distribution

Because of the differences between the many different Linux distributions, these instructions are a general guide only. Specific details vary according to the configuration of your particular system and the Linux distribution that dual-boots with Red Hat Enterprise Linux.

1. **Remove Red Hat Enterprise Linux partitions**

    1. Boot your Red Hat Enterprise Linux installation.

    2. As root or with **sudo**, run **mount**. Note the partitions that are mounted. In particular, note the partition that is mounted as the root of the filesystem. The output of **mount** on a system where the root of the filesystem is on a standard partition such as **/dev/sda2** might resemble:

        ```
        /dev/sda2 on / type ext4 (rw)
        proc on /proc type proc (rw)
        sysfs on /sys type sysfs (rw)
        devpts on /dev/pts type devpts (rw,gid=5,mode=620)
        tmpfs on /dev/shm type tmpfs (rw,rootcontext="system_u:object_r:tmpfs_t:s0")
        /dev/sda1 on /boot type ext4 (rw)
        none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
        sunrpc on /var/lib/nfs/rpc_pipefs type rpc_pipefs (rw)
        ```

        The output of **mount** on a system where the root of the filesystem is on a logical volume might resemble:

        ```
        /dev/mapper/VolGroup00-LogVol00 on / type ext4 (rw)
        proc on /proc type proc (rw)
        sysfs on /sys type sysfs (rw)
        devpts on /dev/pts type devpts (rw,gid=5,mode=620)
        tmpfs on /dev/shm type tmpfs (rw,rootcontext="system_u:object_r:tmpfs_t:s0")
        /dev/sda1 on /boot type ext4 (rw)
        none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
        sunrpc on /var/lib/nfs/rpc_pipefs type rpc_pipefs (rw)
        ```

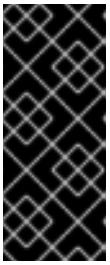
    3. Ensure that any data on this system that you still require is backed up to another system or storage location.

    4. Shut down the system and boot the Linux distribution that you want to keep on the system.

    5. As root or with **sudo**, run **mount**. If any of the partitions that you previously noted as used for Red Hat Enterprise Linux are mounted, review the contents of these partitions. If you no longer require the contents of these partitions, unmount them with the **umount** command.

    6. Remove any unwanted and unnecessary partitions, for example, with **fdisk** for standard partitions, or **lvremove** and **vgremove** to remove logical volumes and volume groups.

2. **Remove Red Hat Enterprise Linux entries from your bootloader**



IMPORTANT

These instructions assume that your system uses the **GRUB** bootloader. If you use a different bootloader (such as **LILO**) consult the documentation for that software to identify and remove Red Hat Enterprise Linux entries from its list of boot targets and to ensure that your default operating system is correctly specified.

1. At the command line, type **su -** and press **Enter**. When the system prompts you for the root password, type the password and press **Enter**.

2. Type **gedit /boot/grub/grub.conf** and press **Enter**. This opens the **grub.conf** file in the **gedit** text editor.

3. A typical Red Hat Enterprise Linux entry in the **grub.conf** file consists of four lines:

   Example 39.1. Example Red Hat Enterprise Linux entry in grub.conf

   ```
   title Red Hat Enterprise Linux (2.6.32.130.el6.i686)

   root (hd0,1)

   kernel /vmlinuz-2.6.32.130.el6.i686 ro root=UUID=04a07c13-e6bf-6d5a-b207-002689545705 rhgb quiet

   initrd /initrd-2.6.32.130.el6.i686.img
   ```

   Depending on the configuration of your system, there may be multiple Red Hat Enterprise Linux entries in **grub.conf**, each corresponding to a different version of the Linux kernel. Delete each of the Red Hat Enterprise Linux entries from the file.

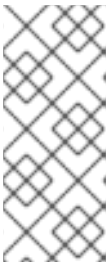
4. **Grub.conf** contains a line that specifies the default operating system to boot, in the format **default=*N*** where *N* is a number equal to or greater than 0. If *N* is set to 0, **GRUB** will boot the first operating system in the list. If *N* is set to 1, it will boot the second operating system, and so forth.

   Identify the entry for the operating system that you want **GRUB** to boot by default and note its place in the order within the list.

   Make sure that the **default=** line contains the number *one below* the number of your chosen default operating system in the list.

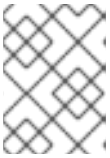
   Save the updated **grub.conf** file and close **gedit**

3. **Make space available to your operating system**



NOTE

This step is not required to remove Red Hat Enterprise Linux from your computer. However, if you skip this step, you will leave part of your hard drive's storage capacity unusable by your other Linux operating system. Depending on your configuration, this might be a significant portion of the storage capacity of the drive.



NOTE

To carry out this step, you require live media for a Linux distribution, for example, the Fedora Live CD or the Knoppix DVD.

The method to make the space freed by removing the Red Hat Enterprise Linux partitions available to your other Linux operating system differs, depending on whether your chosen operating system is installed on disk partitions configured to use Logical Volume Management (LVM) or not.

- **If you do not use LVM**

  1. Boot your computer from Linux live media, and install **parted** if it is not already present.
  2. As root or with **sudo**, run **parted** ***disk***, where *disk* is the device name of the disk that contains a partition that you want to resize, for example, /**dev**/**sda**.
  3. At the **(parted)** prompt, enter **print**. The **parted** tool displays information about the partitions on your system, including their partition numbers, their sizes, and their positions on the disk.
  4. At the **(parted)** prompt, enter **resize** ***number start end***, where *number* is the partition number, *start* is the location on the disk at which the partition begins, and *end* is the location on the disk at which you want the partition to end. Use the start position that you previously obtained with the **print** command, and refer to the **parted** documentation for different ways to specify the end parameter.
  5. When **parted** finishes resizing the partition, enter **quit** at the **(parted)** prompt.
  6. Run **e2fsck** ***partition***, where *partition* is the partition that you just resized. For example, if you just resized /**dev**/**sda3**, enter **e2fsck** /**dev**/**sda3**.

     Linux now checks the file system of the newly-resized partition.
  7. When the file system check finishes, type **resize2fs** ***partition*** at a command line and press **Enter**, where *partition* is the partition that you just resized. For example, if you just resized /**dev**/**sda3**, type **resize2fs** /**dev**/**sda3**.

     Linux now resizes your file system to fill the newly-resized partition.
  8. Restart your computer. The extra space is now available to your Linux installation.

- **If you use LVM**

  1. Boot your computer from Linux live media and install **fdisk** and **lvm2** if they are not already present.
  2. **Create a new partition in the free space on the disk**

     1. As root or with **sudo**, run **fdisk** ***disk***, where *disk* is the device name of the disk where you want to create new space, for example, /**dev**/**sda**.
     2. At the prompt **Command (m for help):**, enter **n** to create a new partition. Refer to the **fdisk** documentation for options.

3. Change the partition type identifier

    1. At the prompt **Command (m for help):**, enter **t** to change a partition type.

    2. At the prompt **Partition number (1-4):**, type the number of the partition that you just created. For example, if you just created partition **/dev/sda3**, type the number **3** and press **Enter**. This identifies the partition whose type **fdisk** will change.

    3. At the prompt **Hex code (type L to list codes):**, enter **8e** to create a Linux LVM partition.

    4. At the prompt **Command (m for help):**, enter **w** to write the changes to disk and exit **fdisk**.

4. Expand the volume group

    1. At the command prompt, type **lvm** and press **Enter** to start the **lvm2** tool.

    2. At the **lvm>** prompt, type **pvcreate *partition*** and press **Enter**, where *partition* is the partition that you recently created. For example, **pvcreate /dev/sda3**. This creates **/dev/sda3** as a physical volume in LVM.

    3. At the **lvm>** prompt, type **vgextend *VolumeGroup partition*** and press **Enter**, where *VolumeGroup* is the LVM volume group on which Linux is installed and *partition* is the partition that you recently created. For example, if Linux is installed on **/dev/VolumeGroup00**, you would type **vgextend /dev/VolumeGroup00 /dev/sda3** to extend that volume group to include the physical volume at **/dev/sda3**.

    4. At the **lvm>** prompt, type **lvextend -l +100%FREE *LogVol*** and press **Enter**, where *LogVol* is the logical volume that contains your Linux filesystem. For example, to extend **LogVol00** to fill the newly-available space in its volume group, **VolGroup00**, type **lvextend -l +100%FREE /dev/VolGroup00/LogVol00**.

    5. At the **lvm>** prompt, type **exit** and press **Enter** to exit **lvm2**

5. Type **e2fsck *LogVol*** at the command line and press **Enter**, where *LogVol* is the logical volume that you just resized. For example, if you just resized **/dev/VolumeGroup00/LogVol00**, you would type **e2fsck /dev/VolumeGroup00/LogVol00**.

    Linux now checks the file system of the newly-resized logical volume.

6. When the file system check finishes, type **resize2fs *LogVol*** at a command line and press **Enter**, where *LogVol* is the partition that you just resized. For example, if you just resized **/dev/VolumeGroup00/LogVol00**, you would type **resize2fs /dev/VolumeGroup00/LogVol00**.

    Linux now resizes your file system to fill the newly-resized logical volume.

7. Restart your computer. The extra space is now available to your Linux installation.

39.3. REPLACING RED HAT ENTERPRISE LINUX WITH MS-DOS OR LEGACY VERSIONS OF MICROSOFT WINDOWS

In DOS and Windows, use the Windows **fdisk** utility to create a new MBR with the *undocumented* flag **/mbr**. This *ONLY* rewrites the MBR to boot the primary DOS partition. The command should look like the following:

```
fdisk /mbr
```

If you need to remove Linux from a hard drive and have attempted to do this with the default DOS (Windows) **fdisk**, you will experience the *Partitions exist but they do not exist* problem. The best way to remove non-DOS partitions is with a tool that understands partitions other than DOS.

To begin, insert the Red Hat Enterprise Linux DVD and boot your system. When the boot prompt appears, type: **linux rescue**. This starts the rescue mode program.

You are prompted for your keyboard and language requirements. Enter these values as you would during the installation of Red Hat Enterprise Linux.

Next, a screen appears telling you that the program attempts to find a Red Hat Enterprise Linux install to rescue. Select **Skip** on this screen.

After selecting **Skip**, you are given a command prompt where you can access the partitions you would like to remove.

First, type the command **list-harddrives**. This command lists all hard drives on your system that are recognizable by the installation program, as well as their sizes in megabytes.

WARNING

Be careful to remove only the necessary Red Hat Enterprise Linux partitions. Removing other partitions could result in data loss or a corrupted system environment.

To remove partitions, use the partitioning utility **parted**. Start **parted**, where */dev/hda* is the device on which to remove the partition:

```
parted /dev/hda
```

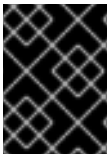
Using the **print** command, view the current partition table to determine the minor number of the partition to remove:

```
print
```

The **print** command also displays the partition's type (such as linux-swap, ext2, ext3, ext4 and so on). Knowing the type of the partition helps you in determining whether to remove the partition.

Remove the partition with the command **rm**. For example, to remove the partition with minor number 3:

```
rm 3
```



IMPORTANT

The changes start taking place as soon as you press [Enter], so review the command before committing to it.

After removing the partition, use the **print** command to confirm that it is removed from the partition table.

Once you have removed the Linux partitions and made all of the changes you need to make, type **quit** to quit **parted**.

After quitting **parted**, type **exit** at the boot prompt to exit rescue mode and reboot your system, instead of continuing with the installation. The system should reboot automatically. If it does not, you can reboot your computer using **Control**+**Alt**+**Delete** .

CHAPTER 40. REMOVING RED HAT ENTERPRISE LINUX FROM IBM SYSTEM Z

If you want to delete the existing operating system data, first, if any Linux disks contain sensitive data, ensure that you destroy the data according to your security policy. To proceed you can consider these options:

- Overwrite the disks with a new installation.
- Start a new installation and use the partitioning dialog (refer to Section 23.13, "Disk Partitioning Setup") to format the partitions where Linux was installed. After the **Write changes to disk** dialog described in Section 23.16, "Write Changes to Disk" , exit the installer.
- Make the DASD or SCSI disk where Linux was installed visible from another system, then delete the data. However, this might require special privileges. Ask your system administrator for advice. You can use Linux commands such as **dasdfmt** (DASD only), **parted**, **mke2fs** or **dd**. For more details about the commands, refer to the respective man pages.

40.1. RUNNING A DIFFERENT OPERATING SYSTEM ON YOUR Z/VM GUEST OR LPAR

If you want to boot from a DASD or SCSI disk different from where the currently installed system resides under a z/VM guest virtual machine or an LPAR, shut down the Red Hat Enterprise Linux installed and use the desired disk, where another Linux instance is installed, to boot from. This leaves the contents of the installed system unchanged.

PART VI. TECHNICAL APPENDICES

The appendices in this section do not contain instructions that tell you how to install Red Hat Enterprise Linux. Instead, they provide technical background that you might find helpful to understand the options that Red Hat Enterprise Linux offers you at various points in the installation process.

APPENDIX A. AN INTRODUCTION TO DISK PARTITIONS

NOTE

This appendix is not necessarily applicable to non-x86-based architectures. However, the general concepts mentioned here may apply.

This appendix is not necessarily applicable to non-x86-based architectures. However, the general concepts mentioned here may apply.

If you are reasonably comfortable with disk partitions, you could skip ahead to Section A.1.5, “Making Room For Red Hat Enterprise Linux”, for more information on the process of freeing up disk space to prepare for a Red Hat Enterprise Linux installation. This section also discusses the partition naming scheme used by Linux systems, sharing disk space with other operating systems, and related topics.

A.1. HARD DISK BASIC CONCEPTS

Hard disks perform a very simple function – they store data and reliably retrieve it on command.

When discussing issues such as disk partitioning, it is important to know a bit about the underlying hardware. Unfortunately, it is easy to become bogged down in details. Therefore, this appendix uses a simplified diagram of a disk drive to help explain what is really happening when a disk drive is partitioned. Figure A.1, “An Unused Disk Drive” , shows a brand-new, unused disk drive.

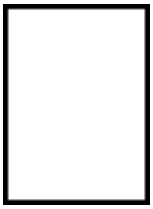


Figure A.1. An Unused Disk Drive

Not much to look at, is it? But if we are talking about disk drives on a basic level, it is adequate. Say that we would like to store some data on this drive. As things stand now, it will not work. There is something we need to do first.

A.1.1. It is Not What You Write, it is How You Write It

Experienced computer users probably got this one on the first try. We need to *format* the drive. Formatting (usually known as "making a *file system*") writes information to the drive, creating order out of the empty space in an unformatted drive.

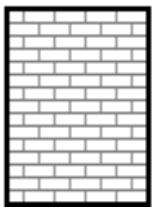


Figure A.2. Disk Drive with a File System

As Figure A.2, “Disk Drive with a File System” , implies, the order imposed by a file system involves some trade-offs:

- A small percentage of the drive's available space is used to store file system-related data and can be considered as overhead.
- A file system splits the remaining space into small, consistently-sized segments. For Linux, these segments are known as *blocks*. [15]

Given that file systems make things like directories and files possible, these trade-offs are usually seen as a small price to pay.

It is also worth noting that there is no single, universal file system. As Figure A.3, "Disk Drive with a Different File System", shows, a disk drive may have one of many different file systems written on it. As you might guess, different file systems tend to be incompatible; that is, an operating system that supports one file system (or a handful of related file system types) may not support another. This last statement is not a hard-and-fast rule, however. For example, Red Hat Enterprise Linux supports a wide variety of file systems (including many commonly used by other operating systems), making data interchange between different file systems easy.

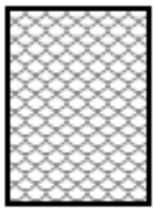


Figure A.3. Disk Drive with a Different File System

Of course, writing a file system to disk is only the beginning. The goal of this process is to actually *store* and *retrieve* data. Let us take a look at our drive after some files have been written to it.

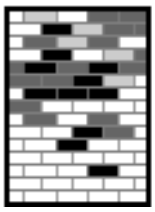


Figure A.4. Disk Drive with Data Written to It

As Figure A.4, "Disk Drive with Data Written to It" , shows, some of the previously-empty blocks are now holding data. However, by just looking at this picture, we cannot determine exactly how many files reside on this drive. There may only be one file or many, as all files use at least one block and some files use multiple blocks. Another important point to note is that the used blocks do not have to form a contiguous region; used and unused blocks may be interspersed. This is known as *fragmentation*. Fragmentation can play a part when attempting to resize an existing partition.

As with most computer-related technologies, disk drives changed over time after their introduction. In particular, they got bigger. Not larger in physical size, but bigger in their capacity to store information. And, this additional capacity drove a fundamental change in the way disk drives were used.

A.1.2. Partitions: Turning One Drive Into Many

Disk drives can be divided into *partitions*. Each partition can be accessed as if it was a separate disk. This is done through the addition of a *partition table*.

There are several reasons for allocating disk space into separate disk partitions, for example:

- Logical separation of the operating system data from the user data
- Ability to use different file systems
- Ability to run multiple operating systems on one machine

There are currently two partitioning layout standards for physical hard disks: Master Boot Record (MBR) and GUID Partition Table (GPT). MBR is an older method of disk partitioning used with BIOS-based computers. GPT is a newer partitioning layout that is a part of the Unified Extensible Firmware Interface (UEFI). This section and Section A.1.3, "Partitions Within Partitions – An Overview of Extended Partitions" mainly describe the *Master Boot Record* (MBR) disk partitioning scheme. For information about the *GUID Partition Table* (GPT) partitioning layout, see Section A.1.4, "GUID Partition Table (GPT)".

NOTE

While the diagrams in this chapter show the partition table as being separate from the actual disk drive, this is not entirely accurate. In reality, the partition table is stored at the very start of the disk, before any file system or user data. But for clarity, they are separate in our diagrams.

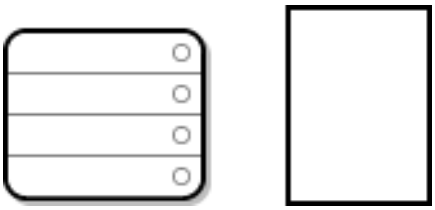


Figure A.5. Disk Drive with Partition Table

As Figure A.5, "Disk Drive with Partition Table" shows, the partition table is divided into four sections or four *primary* partitions. A primary partition is a partition on a hard drive that can contain only one logical drive (or section). Each section can hold the information necessary to define a single partition, meaning that the partition table can define no more than four partitions.

Each partition table entry contains several important characteristics of the partition:

- The points on the disk where the partition starts and ends
- Whether the partition is "active"
- The partition's type

Let us take a closer look at each of these characteristics. The starting and ending points actually define the partition's size and location on the disk. The "active" flag is used by some operating systems' boot loaders. In other words, the operating system in the partition that is marked "active" is booted.

The partition's type can be a bit confusing. The type is a number that identifies the partition's anticipated usage. If that statement sounds a bit vague, that is because the meaning of the partition type is a bit vague. Some operating systems use the partition type to denote a specific file system type, to flag the partition as being associated with a particular operating system, to indicate that the partition contains a bootable operating system, or some combination of the three.

By this point, you might be wondering how all this additional complexity is normally used. Refer to Figure A.6, "Disk Drive With Single Partition" , for an example.

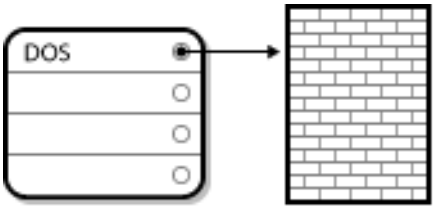


Figure A.6. Disk Drive With Single Partition

In many cases, there is only a single partition spanning the entire disk, essentially duplicating the method used before partitions. The partition table has only one entry used, and it points to the start of the partition.

We have labeled this partition as being of the "DOS" type. Although it is only one of several possible partition types listed in Table A.1, “Partition Types”, it is adequate for the purposes of this discussion.

Table A.1, “Partition Types”, contains a listing of some popular (and obscure) partition types, along with their hexadecimal numeric values.

Table A.1. Partition Types

Partition Type	Value	Partition Type	Value
Empty	00	Novell Netware 386	65
DOS 12-bit FAT	01	PIC/IX	75
XENIX root	02	Old MINIX	80
XENIX usr	03	Linux/MINUX	81
DOS 16-bit <=32M	04	Linux swap	82
Extended	05	Linux native	83
DOS 16-bit >=32	06	Linux extended	85
OS/2 HPFS	07	Amoeba	93
AIX	08	Amoeba BBT	94
AIX bootable	09	BSD/386	a5
OS/2 Boot Manager	0a	OpenBSD	a6
Win95 FAT32	0b	NEXTSTEP	a7
Win95 FAT32 (LBA)	0c	BSDI fs	b7
Win95 FAT16 (LBA)	0e	BSDI swap	b8

Partition Type	Value	Partition Type	Value
Win95 Extended (LBA)	0f	Syrinx	c7
Venix 80286	40	CP/M	db
Novell	51	DOS access	e1
PReP Boot	41	DOS R/O	e3
GNU HURD	63	DOS secondary	f2
Novell Netware 286	64	BBT	ff

A.1.3. Partitions Within Partitions – An Overview of Extended Partitions

Of course, over time it became obvious that four partitions would not be enough. As disk drives continued to grow, it became more and more likely that a person could configure four reasonably-sized partitions and still have disk space left over. There needed to be some way of creating more partitions.

Enter the extended partition. As you may have noticed in Table A.1, "Partition Types" , there is an "Extended" partition type. It is this partition type that is at the heart of extended partitions.

When a partition is created and its type is set to "Extended," an extended partition table is created. In essence, the extended partition is like a disk drive in its own right – it has a partition table that points to one or more partitions (now called *logical partitions*, as opposed to the four *primary partitions*) contained entirely within the extended partition itself. Figure A.7, "Disk Drive With Extended Partition" , shows a disk drive with one primary partition and one extended partition containing two logical partitions (along with some unpartitioned free space).

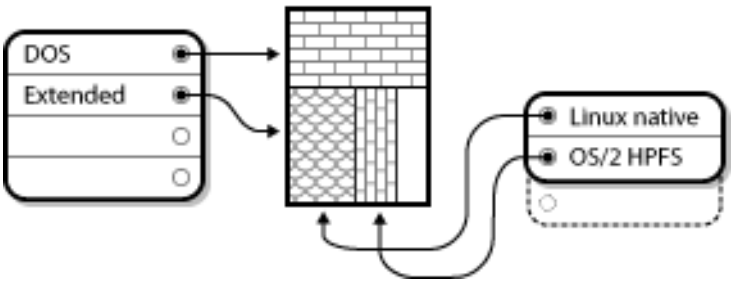


Figure A.7. Disk Drive With Extended Partition

As this figure implies, there is a difference between primary and logical partitions – there can only be four primary partitions, but there is no fixed limit to the number of logical partitions that can exist. However, due to the way in which partitions are accessed in Linux, you should avoid defining more than 12 logical partitions on a single disk drive.

Now that we have discussed partitions in general, let us review how to use this knowledge to install Red Hat Enterprise Linux.

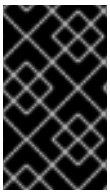
A.1.4. GUID Partition Table (GPT)

GUID Partition Table (GPT) is a newer partitioning scheme based on using Globally Unique Identifiers

(GUID). GPT was developed to cope with limitations of the MBR partition table, especially with the limited maximum addressable storage space of a disk. Unlike MBR, which is unable to address storage space larger than 2.2 terabytes, GPT can be used with hard disks larger than this; the maximum addressable disk size is 2.2 zettabytes. In addition, GPT by default supports creating up to 128 primary partitions. This number could be extended by allocating more space to the partition table.

GPT disks use logical block addressing (LBA) and the partition layout is as follows:

- To preserve backward compatibility with MBR disks, the first sector (LBA 0) of GPT is reserved for MBR data and it is called "protective MBR".
- The *primary GPT header* begins on the second logical block (LBA 1) of the device. The header contains the disk GUID, the location of the primary partition table, the location of the secondary GPT header, and CRC32 checksums of itself and the primary partition table. It also specifies the number of partition entries of the table.
- The *primary GPT table* includes, by default, 128 partition entries, each with an entry size 128 bytes, its partition type GUID and unique partition GUID.
- The *secondary GPT table* is identical to the primary GPT table. It is used mainly as a backup table for recovery in case the primary partition table is corrupted.
- The *secondary GPT header* is located on the last logical sector of the disk and it can be used to recover GPT information in case the primary header is corrupted. It contains the disk GUID, the location of the secondary partition table and the primary GPT header, CRC32 checksums of itself and the secondary partition table, and the number of possible partition entries.



IMPORTANT

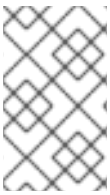
There must be a BIOS boot partition for the boot loader to be installed successfully onto a disk that contains a GPT (GUID Partition Table). This includes disks initialized by **Anaconda**. If the disk already contains a BIOS boot partition, it can be reused.

A.1.5. Making Room For Red Hat Enterprise Linux

The following list presents some possible scenarios you may face when attempting to repartition your hard disk:

- Unpartitioned free space is available
- An unused partition is available
- Free space in an actively used partition is available

Let us look at each scenario in order.



NOTE

Keep in mind that the following illustrations are simplified in the interest of clarity and do not reflect the exact partition layout that you encounter when actually installing Red Hat Enterprise Linux.

A.1.5.1. Using Unpartitioned Free Space

In this situation, the partitions already defined do not span the entire hard disk, leaving unallocated space that is not part of any defined partition. Figure A.8, "Disk Drive with Unpartitioned Free Space" , shows what this might look like.

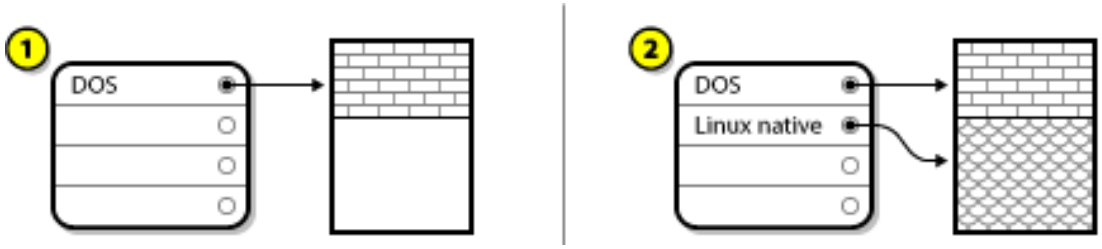


Figure A.8. Disk Drive with Unpartitioned Free Space

In Figure A.8, "Disk Drive with Unpartitioned Free Space" , *1* represents an undefined partition with unallocated space and *2* represents a defined partition with allocated space.

If you think about it, an unused hard disk also falls into this category. The only difference is that *all* the space is not part of any defined partition.

In any case, you can create the necessary partitions from the unused space. Unfortunately, this scenario, although very simple, is not very likely (unless you have just purchased a new disk just for Red Hat Enterprise Linux). Most pre-installed operating systems are configured to take up all available space on a disk drive (refer to Section A.1.5.3, "Using Free Space from an Active Partition").

Next, we will discuss a slightly more common situation.

A.1.5.2. Using Space from an Unused Partition

In this case, maybe you have one or more partitions that you do not use any longer. Perhaps you have dabbled with another operating system in the past, and the partition(s) you dedicated to it never seem to be used anymore. Figure A.9, "Disk Drive With an Unused Partition" , illustrates such a situation.

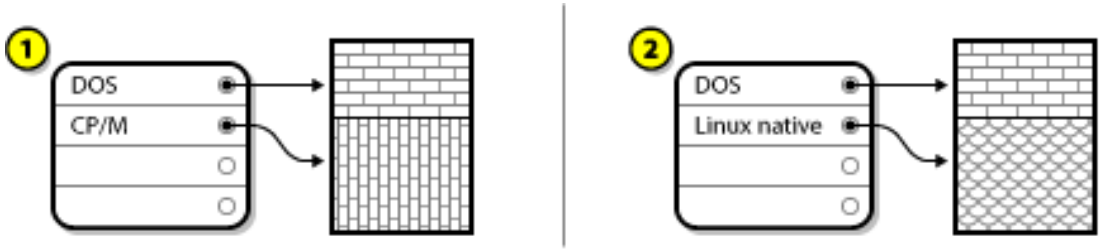


Figure A.9. Disk Drive With an Unused Partition

In Figure A.9, "Disk Drive With an Unused Partition" , *1* represents an unused partition and *2* represents reallocating an unused partition for Linux.

If you find yourself in this situation, you can use the space allocated to the unused partition. You first must delete the partition and then create the appropriate Linux partition(s) in its place. You can delete the unused partition and manually create new partitions during the installation process.

A.1.5.3. Using Free Space from an Active Partition

This is the most common situation. It is also, unfortunately, the hardest to handle. The main problem is that, even if you have enough free space, it is presently allocated to a partition that is already in use. If you purchased a computer with pre-installed software, the hard disk most likely has one massive partition holding the operating system and data.

Aside from adding a new hard drive to your system, you have two choices:

Destructive Repartitioning

Basically, you delete the single large partition and create several smaller ones. As you might imagine, any data you had in the original partition is destroyed. This means that making a complete backup is necessary. For your own sake, make two backups, use verification (if available in your backup software), and try to read data from your backup *before* you delete the partition.

WARNING

If there was an operating system of some type installed on that partition, it needs to be reinstalled as well. Be aware that some computers sold with pre-installed operating systems may not include the CD-ROM media to reinstall the original operating system. The best time to notice if this applies to your system is *before* you destroy your original partition and its operating system installation.

After creating a smaller partition for your existing operating system, you can reinstall any software, restore your data, and start your Red Hat Enterprise Linux installation. Figure A.10, "Disk Drive Being Destructively Repartitioned" shows this being done.

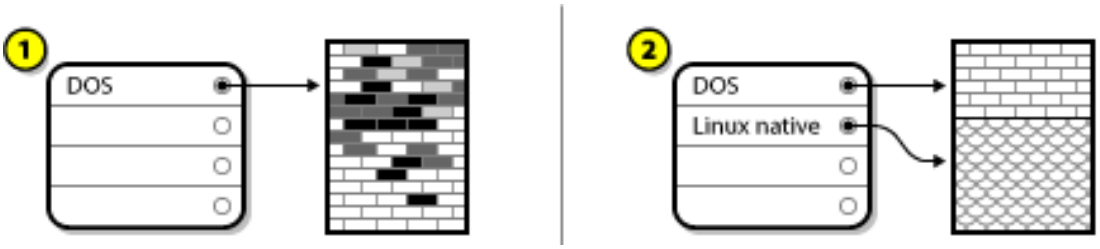


Figure A.10. Disk Drive Being Destructively Repartitioned

In Figure A.10, "Disk Drive Being Destructively Repartitioned" , *1* represents before and *2* represents after.

WARNING

As Figure A.10, "Disk Drive Being Destructively Repartitioned" , shows, any data present in the original partition is lost without proper backup!

Non-Destructive Repartitioning

Here, you run a program that does the seemingly impossible: it makes a big partition smaller without losing any of the files stored in that partition. Many people have found this method to be reliable and trouble-free. What software should you use to perform this feat? There are several disk management software products on the market. Do some research to find the one that is best for your situation.

While the process of non-destructive repartitioning is rather straightforward, there are a number of steps involved:

- Compress and backup existing data
- Resize the existing partition
- Create new partition(s)

Next we will look at each step in a bit more detail.

A.1.5.3.1. Compress existing data

As Figure A.11, "Disk Drive Being Compressed", shows, the first step is to compress the data in your existing partition. The reason for doing this is to rearrange the data such that it maximizes the available free space at the "end" of the partition.

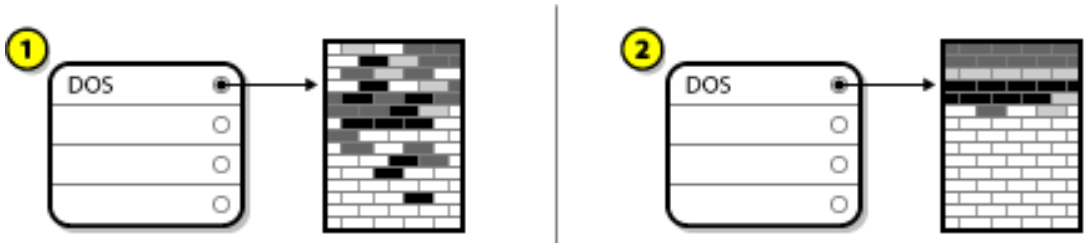


Figure A.11. Disk Drive Being Compressed

In Figure A.11, "Disk Drive Being Compressed", *1* represents before and *2* represents after.

This step is crucial. Without it, the location of your data could prevent the partition from being resized to the extent desired. Note also that, for one reason or another, some data cannot be moved. If this is the case (and it severely restricts the size of your new partition(s)), you may be forced to destructively repartition your disk.

A.1.5.3.2. Resize the existing partition

Figure A.12, "Disk Drive with Partition Resized", shows the actual resizing process. While the actual result of the resizing operation varies depending on the software used, in most cases the newly freed space is used to create an unformatted partition of the same type as the original partition.

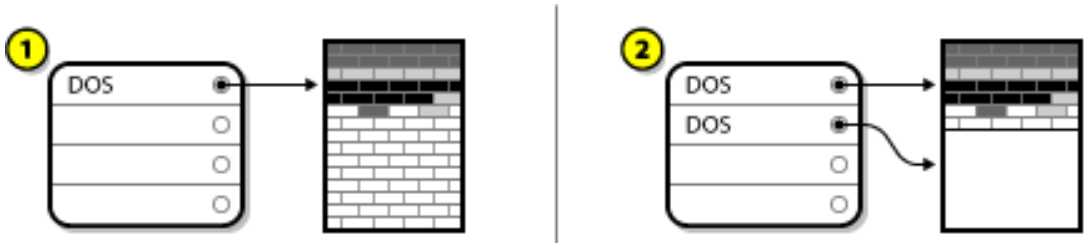


Figure A.12. Disk Drive with Partition Resized

In Figure A.12, "Disk Drive with Partition Resized", *1* represents before and *2* represents after.

It is important to understand what the resizing software you use does with the newly freed space, so that you can take the appropriate steps. In the case we have illustrated, it would be best to delete the new DOS partition and create the appropriate Linux partition(s).

A.1.5.3.3. Create new partition(s)

As the previous step implied, it may or may not be necessary to create new partitions. However, unless your resizing software is Linux-aware, it is likely that you must delete the partition that was created during the resizing process. Figure A.13, "Disk Drive with Final Partition Configuration", shows this being done.

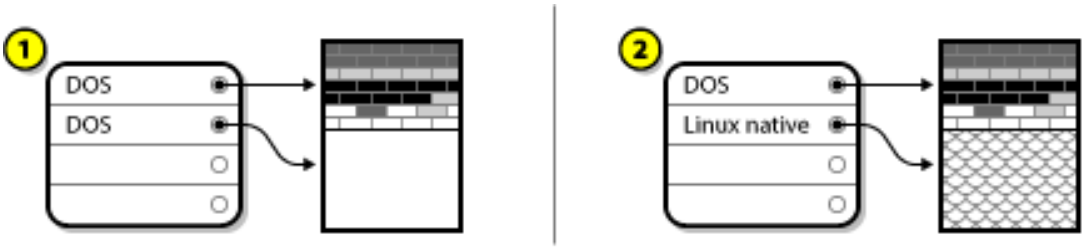
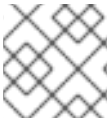


Figure A.13. Disk Drive with Final Partition Configuration

In Figure A.13, "Disk Drive with Final Partition Configuration", *1* represents before and *2* represents after.



NOTE

The following information is specific to x86-based computers only.

As a convenience to our customers, we provide the **parted** utility. This is a freely available program that can resize partitions.

If you decide to repartition your hard drive with **parted**, it is important that you be familiar with disk storage and that you perform a backup of your computer data. You should make two copies of all the important data on your computer. These copies should be to removable media (such as tape, CD-ROM, or diskettes), and you should make sure they are readable before proceeding.

Should you decide to use **parted**, be aware that after **parted** runs you are left with *two* partitions: the one you resized, and the one **parted** created out of the newly freed space. If your goal is to use that space to install Red Hat Enterprise Linux, you should delete the newly created partition, either by using the partitioning utility under your current operating system or while setting up partitions during installation.

A.1.6. Partition Naming Scheme

Linux refers to disk partitions using a combination of letters and numbers which may be confusing, particularly if you are used to the "C drive" way of referring to hard disks and their partitions. In the DOS/Windows world, partitions are named using the following method:

- Each partition's type is checked to determine if it can be read by DOS/Windows.
- If the partition's type is compatible, it is assigned a "drive letter." The drive letters start with a "C" and move on to the following letters, depending on the number of partitions to be labeled.
- The drive letter can then be used to refer to that partition as well as the file system contained on that partition.

Red Hat Enterprise Linux uses a naming scheme that is more flexible and conveys more information than the approach used by other operating systems. The naming scheme is file-based, with file names in the form of **/dev/*xxyN***.

Here is how to decipher the partition naming scheme:

/dev/

This is the name of the directory in which all device files reside. Since partitions reside on hard disks, and hard disks are devices, the files representing all possible partitions reside in **/dev/**.

xx

The first two letters of the partition name indicate the type of device on which the partition resides, usually either **hd** (for IDE disks) or **sd** (for SCSI disks).

y

This letter indicates which device the partition is on. For example, **/dev/hda** (the first IDE hard disk) or **/dev/sdb** (the second SCSI disk).

N

The final number denotes the partition. The first four (primary or extended) partitions are numbered **1** through **4**. Logical partitions start at **5**. So, for example, **/dev/hda3** is the third primary or extended partition on the first IDE hard disk, and **/dev/sdb6** is the second logical partition on the second SCSI hard disk.

NOTE

There is no part of this naming convention that is based on partition type; unlike DOS/Windows, *all* partitions can be identified under Red Hat Enterprise Linux. Of course, this does not mean that Red Hat Enterprise Linux can access data on every type of partition, but in many cases it is possible to access data on a partition dedicated to another operating system.

Keep this information in mind; it makes things easier to understand when you are setting up the partitions Red Hat Enterprise Linux requires.

A.1.7. Disk Partitions and Other Operating Systems

If your Red Hat Enterprise Linux partitions are sharing a hard disk with partitions used by other operating systems, most of the time you will have no problems. However, there are certain combinations of Linux and other operating systems that require extra care.

A.1.8. Disk Partitions and Mount Points

One area that many people new to Linux find confusing is the matter of how partitions are used and accessed by the Linux operating system. In DOS/Windows, it is relatively simple: Each partition gets a "drive letter." You then use the correct drive letter to refer to files and directories on its corresponding partition.

This is entirely different from how Linux deals with partitions and, for that matter, with disk storage in general. The main difference is that each partition is used to form part of the storage necessary to support a single set of files and directories. This is done by associating a partition with a directory through a process known as *mounting*. Mounting a partition makes its storage available starting at the specified directory (known as a *mount point*).

For example, if partition **/dev/hda5** is mounted on **/usr/**, that would mean that all files and directories under **/usr/** physically reside on **/dev/hda5**. So the file **/usr/share/doc/FAQ/txt/Linux-FAQ** would be stored on **/dev/hda5**, while the file **/etc/gdm/custom.conf** would not.

Continuing our example, it is also possible that one or more directories below **/usr/** would be mount points for other partitions. For instance, a partition (say, **/dev/hda7**) could be mounted on **/usr/local/**, meaning that **/usr/local/man/whatis** would then reside on **/dev/hda7** rather than **/dev/hda5**.

A.1.9. How Many Partitions?

At this point in the process of preparing to install Red Hat Enterprise Linux, you must give some consideration to the number and size of the partitions to be used by your new operating system. The question of "how many partitions" continues to spark debate within the Linux community and, without any end to the debate in sight, it is safe to say that there are probably as many partition layouts as there are people debating the issue.

Keeping this in mind, we recommend that, unless you have a reason for doing otherwise, you should at least create the following partitions: **swap**, **/boot/**, and / (root).

For more information, refer to Section 9.15.5, “Recommended Partitioning Scheme”.

---

[15] Blocks really *are* consistently sized, unlike our illustrations. Keep in mind, also, that an average disk drive contains thousands of blocks. But for the purposes of this discussion, please ignore these minor discrepancies.

APPENDIX B. ISCSI DISKS

Internet Small Computer System Interface (iSCSI) is a protocol that allows computers to communicate with storage devices by SCSI requests and responses carried over TCP/IP. Because iSCSI is based on the standard SCSI protocols, it uses some terminology from SCSI. The device on the SCSI bus to which requests get sent (and which answers these requests) is known as the *target* and the device issuing requests is known as the *initiator*. In other words, an iSCSI disk is a target and the iSCSI software equivalent of a SCSI controller or SCSI Host Bus Adapter (HBA) is called an initiator. This appendix only covers Linux as an iSCSI initiator: how Linux uses iSCSI disks, but not how Linux hosts iSCSI disks.

Linux has a software iSCSI initiator in the kernel that takes the place and form of a SCSI HBA driver and therefore allows Linux to use iSCSI disks. However, as iSCSI is a fully network-based protocol, iSCSI initiator support needs more than just the ability to send SCSI packets over the network. Before Linux can use an iSCSI target, Linux must find the target on the network and make a connection to it. In some cases, Linux must send authentication information to gain access to the target. Linux must also detect any failure of the network connection and must establish a new connection, including logging in again if necessary.

The discovery, connection, and logging in is handled in userspace by the **iscsiadm** utility, and the error handling is also handled in userspace by **iscsid**.

Both **iscsiadm** and **iscsid** are part of the **iscsi-initiator-utils** package under Red Hat Enterprise Linux.

B.1. ISCSI DISKS IN ANACONDA

Anaconda can discover (and then log in to) iSCSI disks in two ways:

1. When anaconda starts, it checks if the BIOS or add-on boot ROMs of the system support *iSCSI Boot Firmware Table* (iBFT), a BIOS extension for systems which can boot from iSCSI. If the BIOS supports iBFT, **anaconda** will read the iSCSI target information for the configured boot disk from the BIOS and log in to this target, making it available as an installation target.

2. If you select the **Specialized Storage Devices** option during installation, the storage device selection screen presents you with an **Add Advanced Target** button. If you click this button, you can add iSCSI target information like the discovery IP address. **Anaconda** probes the given IP address and logs in to any targets that it finds. See Section 9.6.1.1, " Advanced Storage Options " for the details that you can specify for iSCSI targets.

While **anaconda** uses **iscsiadm** to find and log into iSCSI targets, **iscsiadm** automatically stores any information about these targets in the iscsiadm iSCSI database. **Anaconda** then copies this database to the installed system and marks any iSCSI targets not used for / so that the system will automatically log in to them when it starts. If / is placed on an iSCSI target, **initrd** will log into this target and **anaconda** does not include this target in start up scripts to avoid multiple attempts to log into the same target.

If / is placed on an iSCSI target, **anaconda** sets **NetworkManager** to ignore any network interfaces that were active during the installation process. These interfaces will also be configured by **initrd** when the system starts. If **NetworkManager** were to reconfigure these interfaces, the system would lose its connection to /.

B.2. ISCSI DISKS DURING START UP

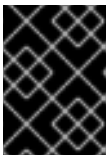
ISCSI-related events might occur at a number of points while the system starts:

1. The init script in the **initrd** will log in to iSCSI targets used for / (if any). This is done using the **iscsistart** utility (which can do this without requiring **iscsid** to run).

2. When the root filesystem has been mounted and the various service initscripts get run, the **iscsid** initscript will get called. This script will then start **iscsid** if any iSCSI targets are used for /, or if any targets in the iSCSI database are marked to be logged in to automatically.

3. After the classic network service script has been run (or would have been run if enabled) the iscsi initscript will run. If the network is accessible, this will log in to any targets in the iSCSI database which are marked to be logged in to automatically. If the network is not accessible, this script will exit quietly.

4. When using **NetworkManager** to access the network (instead of the classic network service script), **NetworkManager** will call the iscsi initscript. See: **/etc/NetworkManager/dispatcher.d/04-iscsi**



IMPORTANT

Because **NetworkManager** is installed in **/usr**, you cannot use it to configure network access if **/usr** is on network-attached storage such as an iSCSI target.

If **iscsid** is not needed as the system starts, it will not start automatically. If you start **iscsiadm**, **iscsiadm** will start **iscsid** in turn.

APPENDIX C. DISK ENCRYPTION

C.1. WHAT IS BLOCK DEVICE ENCRYPTION?

Block device encryption protects the data on a block device by encrypting it. To access the device's decrypted contents, a user must provide a passphrase or key as authentication. This provides additional security beyond existing OS security mechanisms in that it protects the device's contents even if it has been physically removed from the system.

C.2. ENCRYPTING BLOCK DEVICES USING DM-CRYPT/LUKS6TIT

Linux Unified Key Setup (LUKS) is a specification for block device encryption. It establishes an on-disk format for the data, as well as a passphrase/key management policy.

LUKS uses the kernel device mapper subsystem via the **dm-crypt** module. This arrangement provides a low-level mapping that handles encryption and decryption of the device's data. User-level operations, such as creating and accessing encrypted devices, are accomplished through the use of the **cryptsetup** utility.

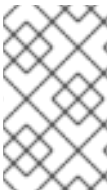
C.2.1. Overview of LUKS

- What LUKS does:
  - LUKS encrypts entire block devices
    - LUKS is thereby well-suited for protecting the contents of mobile devices such as:
      - Removable storage media
      - Laptop disk drives
  - The underlying contents of the encrypted block device are arbitrary.
    - This makes it useful for encrypting **swap** devices.
    - This can also be useful with certain databases that use specially formatted block devices for data storage.
  - LUKS uses the existing device mapper kernel subsystem.
    - This is the same subsystem used by LVM, so it is well tested.
  - LUKS provides passphrase strengthening.
    - This protects against dictionary attacks.
  - LUKS devices contain multiple key slots.
    - This allows users to add backup keys/passphrases.
- What LUKS does *not* do:
  - LUKS is not well-suited for applications requiring many (more than eight) users to have distinct access keys to the same device.
  - LUKS is not well-suited for applications requiring file-level encryption.

More detailed information about LUKS is available from the project website at http://code.google.com/p/cryptsetup/.

C.2.2. How Will I Access the Encrypted Devices After Installation? (System Startup)

During system startup you will be presented with a passphrase prompt. After the correct passphrase has been provided the system will continue to boot normally. If you used different passphrases for multiple encrypted devices you may need to enter more than one passphrase during the startup.



NOTE

Consider using the same passphrase for all encrypted block devices in a given system. This will simplify system startup and you will have fewer passphrases to remember. Just make sure you choose a good passphrase!

C.2.3. Choosing a Good Passphrase

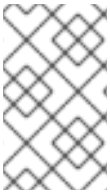
While dm-crypt/LUKS supports both keys and passphrases, the anaconda installer only supports the use of passphrases for creating and accessing encrypted block devices during installation.

LUKS does provide passphrase strengthening but it is still a good idea to choose a good (meaning "difficult to guess") passphrase. Note the use of the term "passphrase", as opposed to the term "password". This is intentional. Providing a phrase containing multiple words to increase the security of your data is important.

C.3. CREATING ENCRYPTED BLOCK DEVICES IN ANACONDA

You can create encrypted devices during system installation. This allows you to easily configure a system with encrypted partitions.

To enable block device encryption, check the "Encrypt System" checkbox when selecting automatic partitioning or the "Encrypt" checkbox when creating an individual partition, software RAID array, or logical volume. After you finish partitioning, you will be prompted for an encryption passphrase. This passphrase will be required to access the encrypted devices. If you have pre-existing LUKS devices and provided correct passphrases for them earlier in the install process the passphrase entry dialog will also contain a checkbox. Checking this checkbox indicates that you would like the new passphrase to be added to an available slot in each of the pre-existing encrypted block devices.



NOTE

Checking the "Encrypt System" checkbox on the "Automatic Partitioning" screen and then choosing "Create custom layout" does not cause any block devices to be encrypted automatically.

NOTE

You can use **kickstart** to set a separate passphrase for each new encrypted block device.

C.3.1. What Kinds of Block Devices Can Be Encrypted?

Most types of block devices can be encrypted using LUKS. From anaconda you can encrypt partitions, LVM physical volumes, LVM logical volumes, and software RAID arrays.

C.3.2. Saving Passphrases

If you use a kickstart file during installation, you can automatically save the passphrases used during installation to an encrypted file (an *escrow packet*) on the local file system. To use this feature, you must have an X.509 certificate available at a location that **anaconda** can access. To specify the URL of this certificate, add the ***--escrowcert*** parameter to any of the **autopart**, **logvol**, **part** or **raid** commands. During installation, the encryption keys for the specified devices are saved in files in /**root**, encrypted with the certificate.

You can save escrow packets during installation only with the use of a kickstart file – refer to Chapter 32, *Kickstart Installations* for more detail. You cannot save an escrow packet during an interactive installation, although you can create one on an installed system with the **volume_key** tool. The **volume_key** tool also allows you to use the information stored in an escrow packet to restore access to an encrypted volume. Refer to the **volume_key** manpage for more information.

C.3.3. Creating and Saving Backup Passphrases

If you use a kickstart file during installation, **anaconda** can add a randomly generated backup passphrase to each block device on the system and save each passphrase to an encrypted file on the local file system. Specify the URL of this certificate with the ***--escrowcert*** parameter as described in Section C.3.2, "Saving Passphrases", followed by the ***--backuppassphrase*** parameter for each of the kickstart commands that relate to the devices for which you want to create backup passphrases.

Note that this feature is available only while performing a kickstart installation. Refer to Chapter 32, *Kickstart Installations* for more detail.

C.4. CREATING ENCRYPTED BLOCK DEVICES ON THE INSTALLED SYSTEM AFTER INSTALLATION

Encrypted block devices can be created and configured after installation.

C.4.1. Create the Block Devices

Create the block devices you want to encrypt by using **parted**, **pvcreate**, **lvcreate** and **mdadm**.

C.4.2. Optional: Fill the Device with Random Data

Filling <device> (eg: /**dev**/**sda3**) with random data before encrypting it greatly increases the strength of the encryption. The downside is that it can take a very long time.

WARNING

The commands below will destroy any existing data on the device.

- The best way, which provides high quality random data but takes a long time (several minutes per gigabyte on most systems):

```
dd if=/dev/urandom of=<device>
```

- Fastest way, which provides lower quality random data:

```
badblocks -c 10240 -s -w -t random -v <device>
```

C.4.3. Format the Device as a dm-crypt/LUKS Encrypted Device

WARNING

The command below will destroy any existing data on the device.

```
cryptsetup luksFormat <device>
```

NOTE

For more information, read the **cryptsetup(8)** man page.

After supplying the passphrase twice the device will be formatted for use. To verify, use the following command:

```
cryptsetup isLuks <device> && echo Success
```

To see a summary of the encryption information for the device, use the following command:

```
cryptsetup luksDump <device>
```

C.4.4. Create a Mapping to Allow Access to the Device's Decrypted Contents

To access the device's decrypted contents, a mapping must be established using the kernel **device-mapper**.

It is useful to choose a meaningful name for this mapping. LUKS provides a UUID (Universally Unique Identifier) for each device. This, unlike the device name (eg: **/dev/sda3**), is guaranteed to remain constant as long as the LUKS header remains intact. To find a LUKS device's UUID, run the following command:

```
cryptsetup luksUUID <device>
```

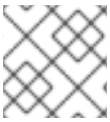
An example of a reliable, informative and unique mapping name would be **luks-<uuid>**, where <uuid> is replaced with the device's LUKS UUID (eg: **luks-50ec957a-5b5a-47ee-85e6-f8085bbc97a8**). This naming convention might seem unwieldy but is it not necessary to type it often.

```
cryptsetup luksOpen <device> <name>
```

There should now be a device node, **/dev/mapper/<name>**, which represents the decrypted device. This block device can be read from and written to like any other unencrypted block device.

To see some information about the mapped device, use the following command:

```
dmsetup info <name>
```



NOTE

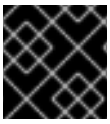
For more information, read the **dmsetup(8)** man page.

C.4.5. Create File Systems on the Mapped Device or Continue to Build Complex Storage Structures Using the Mapped Device

Use the mapped device node (**/dev/mapper/<name>**) as any other block device. To create an **ext2** filesystem on the mapped device, use the following command:

```
mke2fs /dev/mapper/<name>
```

To mount this filesystem on **/mnt/test**, use the following command:



IMPORTANT

The directory **/mnt/test** must exist before executing this command.

```
mount /dev/mapper/<name> /mnt/test
```

C.4.6. Add the Mapping Information to /etc/crypttab

In order for the system to set up a mapping for the device, an entry must be present in the **/etc/crypttab** file. If the file doesn't exist, create it and change the owner and group to root (**root:root**) and change the mode to **0744**. Add a line to the file with the following format:

```
<name>  <device>  none
```

The <device> field should be given in the form "UUID=<luks_uuid>", where <luks_uuid> is the LUKS uuid as given by the command **cryptsetup luksUUID <device>**. This ensures the correct device will be identified and used even if the device node (eg: **/dev/sda5**) changes.

NOTE

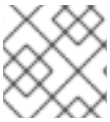
For details on the format of the **/etc/crypttab** file, read the **crypttab(5)** man page.

C.4.7. Add an Entry to /etc/fstab

Add an entry to /etc/fstab. This is only necessary if you want to establish a persistent association between the device and a mountpoint. Use the decrypted device, **/dev/mapper/<name>** in the **/etc/fstab** file.

In many cases it is desirable to list devices in **/etc/fstab** by UUID or by a filesystem label. The main purpose of this is to provide a constant identifier in the event that the device name (eg: **/dev/sda4**) changes. LUKS device names in the form of **/dev/mapper/luks-<luks_uuid>** are based only on the

device's LUKS UUID, and are therefore guaranteed to remain constant. This fact makes them suitable for use in **/etc/fstab**.



NOTE

For details on the format of the **/etc/fstab** file, read the **fstab(5)** man page.

C.5. COMMON POST-INSTALLATION TASKS

The following sections are about common post-installation tasks.

C.5.1. Set a Randomly Generated Key as an Additional Way to Access an Encrypted Block Device

The following sections are about generating keys and adding keys.

C.5.1.1. Generate a Key

This will generate a 256-bit key in the file **$HOME/keyfile**.

```
dd if=/dev/urandom of=$HOME/keyfile bs=32 count=1
chmod 600 $HOME/keyfile
```

C.5.1.2. Add the Key to an Available Keyslot on the Encrypted Device

```
cryptsetup luksAddKey <device> ~/keyfile
```

C.5.2. Add a New Passphrase to an Existing Device

```
cryptsetup luksAddKey <device>
```

After being prompted for any one of the existing passphrases for authentication, you will be prompted to enter the new passphrase.

C.5.3. Remove a Passphrase or Key from a Device

```
cryptsetup luksRemoveKey <device>
```

You will be prompted for the passphrase you wish to remove and then for any one of the remaining passphrases for authentication.

APPENDIX D. UNDERSTANDING LVM

LVM (Logical Volume Management) partitions provide a number of advantages over standard partitions. LVM partitions are formatted as *physical volumes*. One or more physical volumes are combined to form a *volume group*. Each volume group's total storage is then divided into one or more *logical volumes*. The logical volumes function much like standard partitions. They have a file system type, such as **ext4**, and a mount point.

NOTE

On most architectures, the boot loader cannot read LVM volumes. You must make a standard, non-LVM disk partition for your **/boot** partition.

However, on System z, the **zipl** boot loader supports **/boot** on LVM logical volumes with linear mapping.

To understand LVM better, imagine the physical volume as a pile of *blocks*. A block is simply a storage unit used to store data. Several piles of blocks can be combined to make a much larger pile, just as physical volumes are combined to make a volume group. The resulting pile can be subdivided into several smaller piles of arbitrary size, just as a volume group is allocated to several logical volumes.

An administrator may grow or shrink logical volumes without destroying data, unlike standard disk partitions. If the physical volumes in a volume group are on separate drives or RAID arrays then administrators may also spread a logical volume across the storage devices.

You may lose data if you shrink a logical volume to a smaller capacity than the data on the volume requires. To ensure maximum flexibility, create logical volumes to meet your current needs, and leave excess storage capacity unallocated. You may safely grow logical volumes to use unallocated space, as your needs dictate.

NOTE

By default, the installation process creates / and swap partitions within LVM volumes, with a separate **/boot** partition.

APPENDIX E. THE GRUB BOOT LOADER

When a computer running Linux is turned on, the operating system is loaded into memory by a special program called a *boot loader*. A boot loader usually exists on the system's primary hard drive (or other media device) and has the sole responsibility of loading the Linux kernel with its required files or (in some cases) other operating systems into memory.

E.1. BOOT LOADERS AND SYSTEM ARCHITECTURE

Each architecture capable of running Red Hat Enterprise Linux uses a different boot loader. The following table lists the boot loaders available for each architecture:

Table E.1. Boot Loaders by Architecture

Architecture	Boot Loaders
AMD AMD64	GRUB
IBM Power Systems	yaboot
IBM System z	z/IPL
x86	GRUB

This appendix discusses commands and configuration options for the GRUB boot loader included with Red Hat Enterprise Linux for the x86 architecture.

IMPORTANT

The /**boot** and / (root) partition in Red Hat Enterprise Linux 6.9 can only use the ext2, ext3, and ext4 (recommended) file systems. You cannot use any other file system for this partition, such as Btrfs, XFS, or VFAT. Other partitions, such as /**home**, can use any supported file system, including Btrfs and XFS (if available). See the following article on the Red Hat Customer Portal for additional information: https://access.redhat.com/solutions/667273.

E.2. GRUB

The *GNU GRand Unified Boot loader* (GRUB) is a program which enables the selection of the installed operating system or kernel to be loaded at system boot time. It also allows the user to pass arguments to the kernel.

E.2.1. GRUB and the Boot Process on BIOS-based x86 Systems

This section describes the specific role GRUB plays when booting a BIOS-based x86 system. For a look at the overall boot process, refer to Section F.2, “A Detailed Look at the Boot Process”.

GRUB loads itself into memory in the following stages:

1. *The Stage 1 or primary boot loader is read into memory by the BIOS from the MBR* [16]. The primary boot loader exists on less than 512 bytes of disk space within the MBR and is capable of loading either the Stage 1.5 or Stage 2 boot loader.

   BIOS cannot read partition tables or file systems. It initializes the hardware, reads the MBR, then depends entirely on the stage 1 bootloader to continue the boot process.

2. *The Stage 1.5 boot loader is read into memory by the Stage 1 boot loader, if necessary.* Some hardware requires an intermediate step to get to the Stage 2 boot loader. This is sometimes true when the **/boot/** partition is above the 1024 cylinder head of the hard drive or when using LBA mode. The Stage 1.5 boot loader is found either on the **/boot/** partition or on a small part of the MBR and the **/boot/** partition.

3. *The Stage 2 or secondary boot loader is read into memory.* The secondary boot loader displays the GRUB menu and command environment. This interface allows the user to select which kernel or operating system to boot, pass arguments to the kernel, or look at system parameters.

4. *The secondary boot loader reads the operating system or kernel as well as the contents of **/boot/sysroot/** into memory.* Once GRUB determines which operating system or kernel to start, it loads it into memory and transfers control of the machine to that operating system.

The method used to boot Linux is called *direct loading* because the boot loader loads the operating system directly. There is no intermediary between the boot loader and the kernel.

The boot process used by other operating systems may differ. For example, the Microsoft Windows operating system, as well as other operating systems, are loaded using *chain loading*. Under this method, the MBR points to the first sector of the partition holding the operating system, where it finds the files necessary to actually boot that operating system.

GRUB supports both direct and chain loading boot methods, allowing it to boot almost any operating system.

WARNING

During installation, Microsoft's DOS and Windows installation programs completely overwrite the MBR, destroying any existing boot loaders. If creating a dual-boot system, it is best to install the Microsoft operating system first.

E.2.2. GRUB and the Boot Process on UEFI-based x86 Systems

This section describes the specific role GRUB plays when booting a UEFI-based x86 system. For a look at the overall boot process, refer to Section F.2, “A Detailed Look at the Boot Process”.

GRUB loads itself into memory in the following stages:

1. The UEFI-based platform reads the partition table on the system storage and mounts the *EFI System Partition* (ESP), a VFAT partition labeled with a particular *globally unique identifier* (GUID). The ESP contains EFI applications such as bootloaders and utility software, stored in directories specific to software vendors. Viewed from within the Red Hat Enterprise Linux 6.9 file system, the ESP is **/boot/efi/**, and EFI software provided by Red Hat is stored in **/boot/efi/EFI/redhat/**.

2. The **/boot/efi/EFI/redhat/** directory contains **grub.efi**, a version of GRUB compiled for the EFI firmware architecture as an EFI application. In the simplest case, the EFI boot manager selects **grub.efi** as the default bootloader and reads it into memory.

   If the ESP contains other EFI applications, the EFI boot manager might prompt you to select an application to run, rather than load **grub.efi** automatically.

3. GRUB determines which operating system or kernel to start, loads it into memory, and transfers control of the machine to that operating system.

Because each vendor maintains its own directory of applications in the ESP, chain loading is not normally necessary on UEFI-based systems. The EFI boot manager can load any of the operating system bootloaders that are present in the ESP.

E.2.3. Features of GRUB

GRUB contains several features that make it preferable to other boot loaders available for the x86 architecture. Below is a partial list of some of the more important features:

- *GRUB provides a true command-based, pre-OS environment on x86 machines.* This feature affords the user maximum flexibility in loading operating systems with specified options or gathering information about the system. For years, many non-x86 architectures have employed pre-OS environments that allow system booting from a command line.

- *GRUB supports Logical Block Addressing (LBA) mode.* LBA places the addressing conversion used to find files in the hard drive's firmware, and is used on many IDE and all SCSI hard devices. Before LBA, boot loaders could encounter the 1024-cylinder BIOS limitation, where the BIOS could not find a file after the 1024 cylinder head of the disk. LBA support allows GRUB to boot operating systems from partitions beyond the 1024-cylinder limit, so long as the system BIOS supports LBA mode. Most modern BIOS revisions support LBA mode.

- *GRUB can read ext2 partitions.* This functionality allows GRUB to access its configuration file, **/boot/grub/grub.conf**, every time the system boots, eliminating the need for the user to write a new version of the first stage boot loader to the MBR when configuration changes are made. The only time a user needs to reinstall GRUB on the MBR is if the physical location of the **/boot/** partition is moved on the disk.

E.3. INSTALLING GRUB

In a vast majority of cases, **GRUB** is installed and configured by default during the installation of Red Hat Enterprise Linux. However, if for some reason **GRUB** is not installed, or if you need to install it again, it is possible to install grub manually.

On systems without UEFI firmware, a valid GRUB configuration file must be present at **/boot/grub/grub.conf**. You can use the **grub-install** script (part of the grub package) to install GRUB. For example:

```
# grub-install disk
```

Replace *disk* with the device name of your system's boot drive such as **/dev/sda**.

On systems with UEFI firmware, a valid GRUB configuration file must be present at **/boot/efi/EFI/redhat/grub.conf**. An image of GRUB's first-stage boot loader is available on the EFI System Partitition in the directory **EFI/redhat/** with the filename **grubx64.efi**, and you can use the **efibootmgr** command to install this image into your system's EFI System Partition. For example:

```
# efibootmgr -c -d disk -p partition_number -l /EFI/redhat/grubx64.efi -L "grub_uefi"
```

Replace *disk* with the name of the device containing the EFI System Partition (such as **/dev/sda**) and *partition_number* with the partition number of your EFI System Partition (the default value is 1, meaning the first partition on the disk).

IMPORTANT

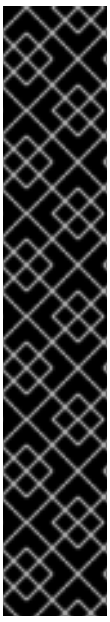
The grub package does not automatically update the system boot loader when the package is updated using **Yum** or **RPM**. Therefore, updating the package will not automatically update the actual boot loader on your system. Use the **grub-install** command manually every time after the package is updated.

For additional information about installing **GRUB**, see the GNU GRUB Manual and the **grub-install(8)** man page. For information about the EFI System Partition, see Section 9.18.1, "Advanced Boot Loader Configuration". For information about the **efibootmgr** tool, see the **efibootmgr(8)** man page.

E.4. TROUBLESHOOTING GRUB

In most cases, **GRUB** will be installed and configured during the initial installation process, unless you used a Kickstart file and specifically disabled this behavior. The installed system should therefore be prepared to boot into your desktop environment or a command line, depending on your package selection. However, in certain cases it is possible that the system's **GRUB** configuration becomes corrupted and the system will no longer be able to boot. This section describes how to fix such problems.

When troubleshooting **GRUB**, keep in mind thatthe grub package does not automatically update the system boot loader when the package is updated using **Yum** or **RPM**. Therefore, updating the package will not automatically update the actual boot loader on your system. To work around this problem, use the **grub-install** command manually every time after the package is updated. See Section E.3, "Installing GRUB" for details about the command.



IMPORTANT

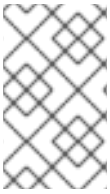
GRUB cannot construct a software RAID. Therefore, the **/boot** directory must reside on a single, specific disk partition. The **/boot** directory cannot be striped across multiple disks, as in a level 0 RAID. To use a level 0 RAID on your system, place **/boot** on a separate partition outside the RAID.

Similarly, because the **/boot** directory must reside on a single, specific disk partition, **GRUB** cannot boot the system if the disk holding that partition fails or is removed from the system. This is true even if the disk is mirrored in a level 1 RAID. The following Red Hat Knowledgebase article describes how to make the system bootable from another disk in the mirrored set: https://access.redhat.com/site/articles/7094

Note that these issues apply only to RAID that is implemented in software, where the individual disks that make up the array are still visible as individual disks on the system. These issues do not apply to hardware RAID where multiple disks are represented as a single device.

The exact steps to fix a broken **GRUB** configuration will vary depending on what kind of problem there is. The GNU GRUB Manual offers a list of all possible error messages displayed by **GRUB** in different stages and their underlying causes. Use the manual for reference.

Once you have determined the cause of the error, you can start fixing it. If you are encountering an error which only appears after you select an entry from the **GRUB** menu, then you can use the menu to fix the error temporarily, boot the system, and then fix the error permanently by running the **grub-install** command to reinstall the boot loader, or by editing the **/boot/grub/grub.conf** or **/boot/efi/EFI/redhat/grub.conf** with a plain text editor. For information about the configuration file structure, see Section E.8, “GRUB Menu Configuration File” .



NOTE

There are two identical files in the **GRUB** configuration directory: **grub.conf** and **menu.lst**. The **grub.conf** configuration file is loaded first; therefore you should make your changes there. The second file, **menu.lst**, will only be loaded if **grub.conf** is not found.

E.5. GRUB TERMINOLOGY

One of the most important things to understand before using GRUB is how the program refers to devices, such as hard drives and partitions. This information is particularly important when configuring GRUB to boot multiple operating systems.

E.5.1. Device Names

When referring to a specific device with GRUB, do so using the following format (note that the parentheses and comma are very important syntactically):

(*<type-of-device><bios-device-number>*,*<partition-number>*)

The *<type-of-device>* specifies the type of device from which GRUB boots. The two most common options are **hd** for a hard disk or **fd** for a 3.5 diskette. A lesser used device type is also available called **nd** for a network disk. Instructions on configuring GRUB to boot over the network are available online at http://www.gnu.org/software/grub/manual/.

The *<bios-device-number>* is the BIOS device number. The primary IDE hard drive is numbered **0** and a secondary IDE hard drive is numbered **1**. This syntax is roughly equivalent to that used for devices by the kernel. For example, the **a** in **hda** for the kernel is analogous to the **0** in **hd0** for GRUB, the **b** in **hdb** is analogous to the **1** in **hd1**, and so on.

The *<partition-number>* specifies the number of a partition on a device. Like the *<bios-device-number>*, most types of partitions are numbered starting at **0**. However, BSD partitions are specified using letters, with **a** corresponding to **0**, **b** corresponding to **1**, and so on.

NOTE

The numbering system for devices under GRUB always begins with **0**, not **1**. Failing to make this distinction is one of the most common mistakes made by new users.

To give an example, if a system has more than one hard drive, GRUB refers to the first hard drive as **(hd0)** and the second as **(hd1)**. Likewise, GRUB refers to the first partition on the first drive as **(hd0,0)** and the third partition on the second hard drive as **(hd1,2)**.

In general the following rules apply when naming devices and partitions under GRUB:

- It does not matter if system hard drives are IDE or SCSI, all hard drives begin with the letters **hd**. The letters **fd** are used to specify 3.5 diskettes.

- To specify an entire device without respect to partitions, leave off the comma and the partition number. This is important when telling GRUB to configure the MBR for a particular disk. For example, **(hd0)** specifies the MBR on the first device and **(hd3)** specifies the MBR on the fourth device.

- If a system has multiple drive devices, it is very important to know how the drive boot order is set in the BIOS. This is a simple task if a system has only IDE or SCSI drives, but if there is a mix of devices, it becomes critical that the type of drive with the boot partition be accessed first.

E.5.2. File Names and Blocklists

When typing commands to GRUB that reference a file, such as a menu list, it is necessary to specify an absolute file path immediately after the device and partition numbers.

The following illustrates the structure of such a command:

(<device-type><device-number>,<partition-number>)</path/to/file>

In this example, replace *<device-type>* with **hd**, **fd**, or **nd**. Replace *<device-number>* with the integer for the device. Replace *</path/to/file>* with an absolute path relative to the top-level of the device.

It is also possible to specify files to GRUB that do not actually appear in the file system, such as a chain loader that appears in the first few blocks of a partition. To load such files, provide a *blocklist* that specifies block by block where the file is located in the partition. Since a file is often comprised of several different sets of blocks, blocklists use a special syntax. Each block containing the file is specified by an offset number of blocks, followed by the number of blocks from that offset point. Block offsets are listed sequentially in a comma-delimited list.

The following is a sample blocklist:

```
0+50,100+25,200+1
```

This sample blocklist specifies a file that starts at the first block on the partition and uses blocks 0 through 49, 100 through 124, and 200.

Knowing how to write blocklists is useful when using GRUB to load operating systems which require chain loading. It is possible to leave off the offset number of blocks if starting at block 0. As an example, the chain loading file in the first partition of the first hard drive would have the following name:

```
(hd0,0)+1
```

The following shows the **chainloader** command with a similar blocklist designation at the GRUB command line after setting the correct device and partition as root:

```
chainloader +1
```

E.5.3. The Root File System and GRUB

The use of the term *root file system* has a different meaning in regard to GRUB. It is important to remember that GRUB's root file system has nothing to do with the Linux root file system.

The GRUB root file system is the top level of the specified device. For example, the image file **(hd0,0)/grub/splash.xpm.gz** is located within the **/grub/** directory at the top-level (or root) of the **(hd0,0)** partition (which is actually the **/boot/** partition for the system).

Next, the **kernel** command is executed with the location of the kernel file as an option. Once the Linux kernel boots, it sets up the root file system that Linux users are familiar with. The original GRUB root file system and its mounts are forgotten; they only existed to boot the kernel file.

Refer to the **root** and **kernel** commands in Section E.7, "GRUB Commands" for more information.

E.6. GRUB INTERFACES

GRUB features three interfaces which provide different levels of functionality. Each of these interfaces allows users to boot the Linux kernel or another operating system.

The interfaces are as follows:

NOTE

The following GRUB interfaces can only be accessed by pressing any key within the three seconds of the GRUB menu bypass screen.

Menu Interface

This is the default interface shown when GRUB is configured by the installation program. A menu of operating systems or preconfigured kernels are displayed as a list, ordered by name. Use the arrow keys to select an operating system or kernel version and press the **Enter** key to boot it. If you do nothing on this screen, then after the time out period expires GRUB will load the default option.

Press the **e** key to enter the entry editor interface or the **c** key to load a command line interface.

Refer to Section E.8, "GRUB Menu Configuration File" for more information on configuring this interface.

Menu Entry Editor Interface

To access the menu entry editor, press the **e** key from the boot loader menu. The GRUB commands for that entry are displayed here, and users may alter these command lines before booting the operating system by adding a command line (**o** inserts a new line after the current line and **O** inserts a new line before it), editing one (**e**), or deleting one (**d**).

After all changes are made, the **b** key executes the commands and boots the operating system. The **Esc** key discards any changes and reloads the standard menu interface. The **c** key loads the command line interface.

NOTE

For information about changing runlevels using the GRUB menu entry editor, refer to Section E.9, "Changing Runlevels at Boot Time".

Command Line Interface

The command line interface is the most basic GRUB interface, but it is also the one that grants the most control. The command line makes it possible to type any relevant GRUB commands followed by the **Enter** key to execute them. This interface features some advanced shell-like features, including **Tab** key completion based on context, and **Ctrl** key combinations when typing commands, such as **Ctrl**+**a** to move to the beginning of a line and **Ctrl**+**e** to move to the end of a line. In addition, the arrow, **Home**, **End**, and **Delete** keys work as they do in the **bash** shell.

Refer to Section E.7, "GRUB Commands" for a list of common commands.

E.6.1. Interfaces Load Order

When GRUB loads its second stage boot loader, it first searches for its configuration file. Once found, the menu interface bypass screen is displayed. If a key is pressed within three seconds, GRUB builds a menu list and displays the menu interface. If no key is pressed, the default kernel entry in the GRUB menu is used.

If the configuration file cannot be found, or if the configuration file is unreadable, GRUB loads the command line interface, allowing the user to type commands to complete the boot process.

If the configuration file is not valid, GRUB prints out the error and asks for input. This helps the user see precisely where the problem occurred. Pressing any key reloads the menu interface, where it is then possible to edit the menu option and correct the problem based on the error reported by GRUB. If the correction fails, GRUB reports an error and reloads the menu interface.

E.7. GRUB COMMANDS

GRUB allows a number of useful commands in its command line interface. Some of the commands accept options after their name; these options should be separated from the command and other options on that line by space characters.

The following is a list of useful commands:

- **boot** — Boots the operating system or chain loader that was last loaded.
- **chainloader** ***</path/to/file>*** — Loads the specified file as a chain loader. If the file is located on the first sector of the specified partition, use the blocklist notation, **+1**, instead of the file name.

  The following is an example **chainloader** command:

  ```
  chainloader +1
  ```

- **displaymem** — Displays the current use of memory, based on information from the BIOS. This is useful to determine how much RAM a system has prior to booting it.
- **initrd** ***</path/to/initrd>*** — Enables users to specify an initial RAM disk to use when booting. An **initrd** is necessary when the kernel needs certain modules in order to boot properly, such as when the root partition is formatted with the ext3 or ext4 file system.

  The following is an example **initrd** command:

  ```
  initrd /initrd-2.6.8-1.523.img
  ```

- **install** ***<stage-1> <install-disk> <stage-2>*** **p** ***config-file*** — Installs GRUB to the system MBR.
  - ***<stage-1>*** — Signifies a device, partition, and file where the first boot loader image can be found, such as **(hd0,0)/grub/stage1**.
  - ***<install-disk>*** — Specifies the disk where the stage 1 boot loader should be installed, such as **(hd0)**.

- ***<stage-2>*** – Passes the stage 2 boot loader location to the stage 1 boot loader, such as **(hd0,0)/grub/stage2**.
- **p *<config-file>*** – This option tells the **install** command to look for the menu configuration file specified by ***<config-file>***, such as **(hd0,0)/grub/grub.conf**.

WARNING

The **install** command overwrites any information already located on the MBR.

- **kernel *</path/to/kernel> <option-1> <option-N>*** ... – Specifies the kernel file to load when booting the operating system. Replace *</path/to/kernel>* with an absolute path from the partition specified by the root command. Replace *<option-1>* with options for the Linux kernel, such as **root=/dev/VolGroup00/LogVol00** to specify the device on which the root partition for the system is located. Multiple options can be passed to the kernel in a space separated list.

  The following is an example **kernel** command:

  ```
  kernel /vmlinuz-2.6.8-1.523 ro root=/dev/VolGroup00/LogVol00
  ```

  The option in the previous example specifies that the root file system for Linux is located on the **hda5** partition.

- **root (*<device-type><device-number>,<partition>*)** – Configures the root partition for GRUB, such as **(hd0,0)**, and mounts the partition.

  The following is an example **root** command:

  ```
  root (hd0,0)
  ```

- **rootnoverify (*<device-type><device-number>,<partition>*)** – Configures the root partition for GRUB, just like the **root** command, but does not mount the partition.

Other commands are also available; type **help --all** for a full list of commands. For a description of all GRUB commands, refer to the documentation available online at http://www.gnu.org/software/grub/manual/.

E.8. GRUB MENU CONFIGURATION FILE

The configuration file (**/boot/grub/grub.conf** on BIOS systems and **/boot/efi/EFI/redhat/grub.conf** on UEFI systems), which is used to create the list of operating systems to boot in GRUB's menu interface, essentially allows the user to select a pre-set group of commands to execute. The commands given in Section E.7, "GRUB Commands" can be used, as well as some special commands that are only available in the configuration file.

E.8.1. Configuration File Structure

The commands to set the global preferences for the menu interface are placed at the top of the GRUB configuration file, followed by stanzas for each operating kernel or operating system listed in the menu.

The following is a very basic GRUB menu configuration file designed to boot either Red Hat Enterprise Linux or Microsoft Windows:

```
default=0
timeout=10
splashimage=(hd0,0)/grub/splash.xpm.gz
hiddenmenu
title Red Hat Enterprise Linux Server (2.6.32.130.el6.i686)
root (hd0,0)
kernel /boot/vmlinuz-2.6.32.130.el6.i686 ro root=LABEL=/1 rhgb quiet
initrd /boot/initrd-2.6.32.130.el6.i686.img

# section to load Windows
title Windows
rootnoverify (hd0,0)
chainloader +1
```

This file configures GRUB to build a menu with Red Hat Enterprise Linux as the default operating system and sets it to autoboot after 10 seconds. Two sections are given, one for each operating system entry, with commands specific to the system disk partition table.

NOTE

Note that the default is specified as an integer. This refers to the first **title** line in the GRUB configuration file. For the **Windows** section to be set as the default in the previous example, change the **default=0** to **default=1**.

Configuring a GRUB menu configuration file to boot multiple operating systems is beyond the scope of this chapter. Consult Section E.10, "Additional Resources" for a list of additional resources.

E.8.2. Configuration File Directives

The following are directives commonly used in the GRUB menu configuration file:

- **chainloader *</path/to/file>*** – Loads the specified file as a chain loader. Replace *</path/to/file>* with the absolute path to the chain loader. If the file is located on the first sector of the specified partition, use the blocklist notation, **+1**.
- **color *<normal-color> <selected-color>*** – Allows specific colors to be used in the menu, where two colors are configured as the foreground and background. Use simple color names such as **red**/**black**. For example:

  ```
  color red/black green/blue
  ```

- **default=*<integer>*** – Replace *<integer>* with the default entry title number to be loaded if the menu interface times out.
- **fallback=*<integer>*** – Replace *<integer>* with the entry title number to try if the first attempt fails.

- **hiddenmenu** – Prevents the GRUB menu interface from being displayed, loading the **default** entry when the **timeout** period expires. The user can see the standard GRUB menu by pressing the **Esc** key.

- **initrd *</path/to/initrd>*** – Enables users to specify an initial RAM disk to use when booting. Replace *</path/to/initrd>* with the absolute path to the initial RAM disk.

- **kernel *</path/to/kernel> <option-1> <option-N>*** – Specifies the kernel file to load when booting the operating system. Replace *</path/to/kernel>* with an absolute path from the partition specified by the root directive. Multiple options can be passed to the kernel when it is loaded.

  These options include:

  - **rhgb** (*Red Hat graphical boot*) – displays an animation during the boot process, rather than lines of text.

  - **quiet** – suppresses all but the most important messages in the part of the boot sequence before the Red Hat graphical boot animation begins.

- **password=*<password>*** – Prevents a user who does not know the password from editing the entries for this menu option.

  Optionally, it is possible to specify an alternate menu configuration file after the **password=*<password>*** directive. In this case, GRUB restarts the second stage boot loader and uses the specified alternate configuration file to build the menu. If an alternate menu configuration file is left out of the command, a user who knows the password is allowed to edit the current configuration file.

  IMPORTANT

  It is highly recommended to set up a boot loader password on every machine. An unprotected boot loader can allow a potential attacker to modify the system's boot options and gain access to the system. See the chapter titled *Workstation Security* in the *Red Hat Enterprise Linux Security Guide* for more information on boot loader passwords and password security in general.

- **map** – Swaps the numbers assigned to two hard drives. For example:

```
map (hd0) (hd3)
map (hd3) (hd0)
```

  assigns the number **0** to the fourth hard drive, and the number **3** to the first hard drive. This option is especially useful if you configure your system with an option to boot a Windows operating system, because the Windows boot loader must find the Windows installation on the first hard drive.

  For example, if your Windows installation is on the fourth hard drive, the following entry in **grub.conf** will allow the Windows boot loader to load Windows correctly:

```
title Windows
map (hd0) (hd3)
map (hd3) (hd0)
rootnoverify (hd3,0)
chainloader +1
```

- **root (*<device-type><device-number>*,*<partition>*)** – Configures the root partition for GRUB, such as **(hd0,0)**, and mounts the partition. To specify the boot drive selected by the EFI boot manager, the syntax is *<device-type>*,*<partition>*, such as **(bd,1)**.

- **rootnoverify (*<device-type><device-number>*,*<partition>*)** – Configures the root partition for GRUB, just like the **root** command, but does not mount the partition.

- **timeout=*<integer>*** – Specifies the interval, in seconds, that GRUB waits before loading the entry designated in the **default** command.

- **splashimage=*<path-to-image>*** – Specifies the location of the splash screen image to be used when GRUB boots.

- **title *group-title*** – Specifies a title to be used with a particular group of commands used to load a kernel or operating system.

- **device *grub-device-name uefi-device-name*** – Assigns a GRUB device name to refer to a specific UEFI device. The argument *grub-device-name* should be replaced with a GRUB device name, for example **(hd0)**. The argument *uefi-device-name* should be replaced with a UEFI device name in the form of either **HD(*number*, *start*, *size*, *signature*)**, or **CD(*index*, *start*, *size*)**, where *number* is the partition number, starting at 1, *index* is the index of the CD's El Torito boot entry, *start* and *size* are the start position and size of the partition respectively, in sectors, in hexadecimal format, and *signature* is the partition's unique GUID.

To add human-readable comments to the menu configuration file, begin the line with the hash mark character (**#**).

E.9. CHANGING RUNLEVELS AT BOOT TIME

Under Red Hat Enterprise Linux, it is possible to change the default runlevel at boot time.

To change the runlevel of a single boot session, use the following instructions:

- When the GRUB menu bypass screen appears at boot time, press any key to enter the GRUB menu (within the first three seconds).

- Press the **a** key to append to the **kernel** command.

- Add **<space>*<runlevel>*** at the end of the boot options line to boot to the desired runlevel. For example, the following entry would initiate a boot process into runlevel 3:

```
grub append> ro root=/dev/VolGroup00/LogVol00 rhgb quiet 3
```

E.10. ADDITIONAL RESOURCES

This chapter is only intended as an introduction to GRUB. Consult the following resources to discover more about how GRUB works.

E.10.1. Installed Documentation

- **/usr/share/doc/grub-*<version-number>*/** – This directory contains good information about using and configuring GRUB, where *<version-number>* corresponds to the version of the GRUB package installed.

- **info grub** — The GRUB info page contains a tutorial, a user reference manual, a programmer reference manual, and a FAQ document about GRUB and its usage.

E.10.2. Useful Websites

- http://www.gnu.org/software/grub/ — The home page of the GNU GRUB project. This site contains information concerning the state of GRUB development and an FAQ.
- https://access.redhat.com/site/solutions/6863 — Details booting operating systems other than Linux.

---

[16] For more on the system BIOS and the MBR, refer to Section F.2.1.1, "BIOS-based x86 Systems".

APPENDIX F. BOOT PROCESS, INIT, AND SHUTDOWN

An important and powerful aspect of Red Hat Enterprise Linux is the open, user-configurable method it uses for starting the operating system. Users are free to configure many aspects of the boot process, including specifying the programs launched at boot-time. Similarly, system shutdown gracefully terminates processes in an organized and configurable way, although customization of this process is rarely required.

Understanding how the boot and shutdown processes work not only allows customization, but also makes it easier to troubleshoot problems related to starting or shutting down the system.

F.1. THE BOOT PROCESS

Below are the basic stages of the boot process:

1. The system loads and runs a boot loader. The specifics of this process depend on the system architecture. For example:
    - BIOS-based x86 systems run a first-stage boot loader from the MBR of the primary hard disk that, in turn, loads an additional boot loader, **GRUB**.
    - UEFI-based x86 systems mount an EFI System Partition that contains a version of the **GRUB** boot loader. The EFI boot manager loads and runs **GRUB** as an EFI application.
    - Power Systems servers mount a PPC PReP partition that contains the **Yaboot** boot loader. The System Management Services (SMS) boot manager loads and runs **yaboot**.
    - IBM System z runs the **z/IPL** boot loader from a DASD or FCP-connected device that you specify when you IPL the partition that contains Red Hat Enterprise Linux.
2. The boot loader loads the kernel into memory, which in turn loads any necessary modules and mounts the root partition read-only.
3. The kernel transfers control of the boot process to the **/sbin/init** program.
4. The **/sbin/init** program loads all services and user-space tools, and mounts all partitions listed in **/etc/fstab**.
5. The user is presented with a login screen for the freshly booted Linux system.

Because configuration of the boot process is more common than the customization of the shutdown process, the remainder of this chapter discusses in detail how the boot process works and how it can be customized to suite specific needs.

F.2. A DETAILED LOOK AT THE BOOT PROCESS

The beginning of the boot process varies depending on the hardware platform being used. However, once the kernel is found and loaded by the boot loader, the default boot process is identical across all architectures. This chapter focuses primarily on the x86 architecture.

F.2.1. The Firmware Interface

F.2.1.1. BIOS-based x86 Systems

The *Basic Input/Output System* (BIOS) is a firmware interface that controls not only the first step of the

boot process, but also provides the lowest level interface to peripheral devices. On x86 systems equipped with BIOS, the program is written into read-only, permanent memory and is always available for use. When the system boots, the processor looks at the end of system memory for the BIOS program, and runs it.

Once loaded, the BIOS tests the system, looks for and checks peripherals, and then locates a valid device with which to boot the system. Usually, it checks any optical drives or USB storage devices present for bootable media, then, failing that, looks to the system's hard drives. In most cases, the order of the drives searched while booting is controlled with a setting in the BIOS, and it looks on the master IDE on the primary IDE bus or for a SATA device with a boot flag set. The BIOS then loads into memory whatever program is residing in the first sector of this device, called the *Master Boot Record* (MBR). The MBR is only 512 bytes in size and contains machine code instructions for booting the machine, called a boot loader, along with the partition table. Once the BIOS finds and loads the boot loader program into memory, it yields control of the boot process to it.

This first-stage boot loader is a small machine code binary on the MBR. Its sole job is to locate the second stage boot loader (**GRUB**) and load the first part of it into memory.

F.2.1.2. UEFI-based x86 Systems

The *Unified Extensible Firmware Interface* (UEFI) is designed, like BIOS, to control the boot process (through *boot services*) and to provide an interface between system firmware and an operating system (through *runtime services*). Unlike BIOS, it features its own architecture, independent of the CPU, and its own device drivers. UEFI can mount partitions and read certain file systems.

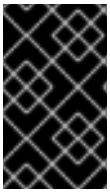
When an x86 computer equipped with UEFI boots, the interface searches the system storage for a partition labeled with a specific *globally unique identifier* (GUID) that marks it as the *EFI System Partition* (ESP). This partition contains applications compiled for the EFI architecture, which might include bootloaders for operating systems and utility software. UEFI systems include an *EFI boot manager* that can boot the system from a default configuration, or prompt a user to choose an operating system to boot. When a bootloader is selected, manually or automatically, UEFI reads it into memory and yields control of the boot process to it.

F.2.2. The Boot Loader

F.2.2.1. The GRUB boot loader for x86 systems

The system loads GRUB into memory, as directed by either a first-stage bootloader in the case of systems equipped with BIOS, or read directly from an EFI System Partition in the case of systems equipped with UEFI.

GRUB has the advantage of being able to read ext2, ext3, and ext4 [17] partitions and load its configuration file — **/boot/grub/grub.conf** (for BIOS) or **/boot/efi/EFI/redhat/grub.conf** (for UEFI) — at boot time. Refer to Section E.8, “GRUB Menu Configuration File” for information on how to edit this file.



IMPORTANT

The **GRUB** bootloader in Red Hat Enterprise Linux 6.9 supports ext2, ext3, and ext4 file systems. It does not support other file systems such as VFAT, Btrfs or XFS. Furthermore, **GRUB** does not support LVM.

Once the second stage boot loader is in memory, it presents the user with a graphical screen showing the different operating systems or kernels it has been configured to boot (when you update the kernel, the boot loader configuration file is updated automatically). On this screen a user can use the arrow

keys to choose which operating system or kernel they wish to boot and press **Enter**. If no key is pressed, the boot loader loads the default selection after a configurable period of time has passed.

Once the second stage boot loader has determined which kernel to boot, it locates the corresponding kernel binary in the **/boot/** directory. The kernel binary is named using the following format — **/boot/vmlinuz-*<kernel-version>*** file (where ***<kernel-version>*** corresponds to the kernel version specified in the boot loader's settings).

For instructions on using the boot loader to supply command line arguments to the kernel, refer to Appendix E, *The GRUB Boot Loader*. For information on changing the runlevel at the boot loader prompt, refer Section E.9, "Changing Runlevels at Boot Time".

The boot loader then places one or more appropriate *initramfs* images into memory. The **initramfs** is used by the kernel to load drivers and modules necessary to boot the system. This is particularly important if SCSI hard drives are present or if the systems use the ext3 or ext4 file system.

Once the kernel and the **initramfs** image(s) are loaded into memory, the boot loader hands control of the boot process to the kernel.

For a more detailed overview of the GRUB boot loader, refer to Appendix E, *The GRUB Boot Loader*.

F.2.2.2. Boot Loaders for Other Architectures

Once the kernel loads and hands off the boot process to the **init** command, the same sequence of events occurs on every architecture. So the main difference between each architecture's boot process is in the application used to find and load the kernel.

For example, the IBM eServer pSeries architecture uses **yaboot**, and the IBM System z systems use the z/IPL boot loader.

Consult the sections of this guide specific to these platforms for information on configuring their boot loaders.

F.2.3. The Kernel

When the kernel is loaded, it immediately initializes and configures the computer's memory and configures the various hardware attached to the system, including all processors, I/O subsystems, and storage devices. It then looks for the compressed **initramfs** image(s) in a predetermined location in memory, decompresses it directly to **/sysroot/**, and loads all necessary drivers. Next, it initializes virtual devices related to the file system, such as LVM or software RAID, before completing the **initramfs** processes and freeing up all the memory the disk image once occupied.

The kernel then creates a root device, mounts the root partition read-only, and frees any unused memory.

At this point, the kernel is loaded into memory and operational. However, since there are no user applications that allow meaningful input to the system, not much can be done with the system.

To set up the user environment, the kernel executes the **/sbin/init** program.

F.2.4. The /sbin/init Program

The **/sbin/init** program (also called **init**) coordinates the rest of the boot process and configures the environment for the user.

When the **init** command starts, it becomes the parent or grandparent of all of the processes that start up

automatically on the system. First, it runs the **/etc/rc.d/rc.sysinit** script, which sets the environment path, starts swap, checks the file systems, and executes all other steps required for system initialization. For example, most systems use a clock, so **rc.sysinit** reads the **/etc/sysconfig/clock** configuration file to initialize the hardware clock. Another example is if there are special serial port processes which must be initialized, **rc.sysinit** executes the **/etc/rc.serial** file.

The **init** command then processes the jobs in the **/etc/event.d** directory, which describe how the system should be set up in each *SysV init runlevel*. Runlevels are a state, or *mode*, defined by the services listed in the SysV **/etc/rc.d/rc*<x>*.d/** directory, where *<x>* is the number of the runlevel. For more information on SysV init runlevels, refer to Section F.4, "SysV Init Runlevels".

Next, the **init** command sets the source function library, **/etc/rc.d/init.d/functions**, for the system, which configures how to start, kill, and determine the PID of a program.

The **init** program starts all of the background processes by looking in the appropriate **rc** directory for the runlevel specified as the default in **/etc/inittab**. The **rc** directories are numbered to correspond to the runlevel they represent. For instance, **/etc/rc.d/rc5.d/** is the directory for runlevel 5.

When booting to runlevel 5, the **init** program looks in the **/etc/rc.d/rc5.d/** directory to determine which processes to start and stop.

Below is an example listing of the **/etc/rc.d/rc5.d/** directory:

```
K05innd -> ../init.d/innd
K05saslauthd -> ../init.d/saslauthd
K10dc_server -> ../init.d/dc_server
K10psacct -> ../init.d/psacct
K10radiusd -> ../init.d/radiusd
K12dc_client -> ../init.d/dc_client
K12FreeWnn -> ../init.d/FreeWnn
K12mailman -> ../init.d/mailman
K12mysqld -> ../init.d/mysqld
K15httpd -> ../init.d/httpd
K20netdump-server -> ../init.d/netdump-server
K20rstatd -> ../init.d/rstatd
K20rusersd -> ../init.d/rusersd
K20rwhod -> ../init.d/rwhod
K24irda -> ../init.d/irda
K25squid -> ../init.d/squid
K28amd -> ../init.d/amd
K30spamassassin -> ../init.d/spamassassin
K34dhcrelay -> ../init.d/dhcrelay
K34yppasswdd -> ../init.d/yppasswdd
K35dhcpd -> ../init.d/dhcpd
K35smb -> ../init.d/smb
K35vncserver -> ../init.d/vncserver
K36lisa -> ../init.d/lisa
K45arpwatch -> ../init.d/arpwatch
K45named -> ../init.d/named
K46radvd -> ../init.d/radvd
K50netdump -> ../init.d/netdump
K50snmpd -> ../init.d/snmpd
K50snmptrapd -> ../init.d/snmptrapd
K50tux -> ../init.d/tux
K50vsftpd -> ../init.d/vsftpd
K54dovecot -> ../init.d/dovecot
```

```
K61ldap -> ../init.d/ldap
K65kadmin -> ../init.d/kadmin
K65kprop -> ../init.d/kprop
K65krb524 -> ../init.d/krb524
K65krb5kdc -> ../init.d/krb5kdc
K70aep1000 -> ../init.d/aep1000
K70bcm5820 -> ../init.d/bcm5820
K74ypserv -> ../init.d/ypserv
K74ypxfrd -> ../init.d/ypxfrd
K85mdmpd -> ../init.d/mdmpd
K89netplugd -> ../init.d/netplugd
K99microcode_ctl -> ../init.d/microcode_ctl
S04readahead_early -> ../init.d/readahead_early
S05kudzu -> ../init.d/kudzu
S06cpuspeed -> ../init.d/cpuspeed
S08ip6tables -> ../init.d/ip6tables
S08iptables -> ../init.d/iptables
S09isdn -> ../init.d/isdn
S10network -> ../init.d/network
S12syslog -> ../init.d/syslog
S13irqbalance -> ../init.d/irqbalance
S13portmap -> ../init.d/portmap
S15mdmonitor -> ../init.d/mdmonitor
S15zebra -> ../init.d/zebra
S16bgpd -> ../init.d/bgpd
S16ospf6d -> ../init.d/ospf6d
S16ospfd -> ../init.d/ospfd
S16ripd -> ../init.d/ripd
S16ripngd -> ../init.d/ripngd
S20random -> ../init.d/random
S24pcmcia -> ../init.d/pcmcia
S25netfs -> ../init.d/netfs
S26apmd -> ../init.d/apmd
S27ypbind -> ../init.d/ypbind
S28autofs -> ../init.d/autofs
S40smartd -> ../init.d/smartd
S44acpid -> ../init.d/acpid
S54hpoj -> ../init.d/hpoj
S55cups -> ../init.d/cups
S55sshd -> ../init.d/sshd
S56rawdevices -> ../init.d/rawdevices
S56xinetd -> ../init.d/xinetd
S58ntpd -> ../init.d/ntpd
S75postgresql -> ../init.d/postgresql
S80sendmail -> ../init.d/sendmail
S85gpm -> ../init.d/gpm
S87iiim -> ../init.d/iiim
S90canna -> ../init.d/canna
S90crond -> ../init.d/crond
S90xfs -> ../init.d/xfs
S95atd -> ../init.d/atd
S96readahead -> ../init.d/readahead
S97messagebus -> ../init.d/messagebus
S97rhnsd -> ../init.d/rhnsd
S99local -> ../rc.local
```

As illustrated in this listing, none of the scripts that actually start and stop the services are located in the **/etc/rc.d/rc5.d/** directory. Rather, all of the files in **/etc/rc.d/rc5.d/** are *symbolic links* pointing to scripts located in the **/etc/rc.d/init.d/** directory. Symbolic links are used in each of the **rc** directories so that the runlevels can be reconfigured by creating, modifying, and deleting the symbolic links without affecting the actual scripts they reference.

The name of each symbolic link begins with either a **K** or an **S**. The **K** links are processes that are killed on that runlevel, while those beginning with an **S** are started.

The **init** command first stops all of the **K** symbolic links in the directory by issuing the **/etc/rc.d/init.d/*<command>* stop** command, where *<command>* is the process to be killed. It then starts all of the **S** symbolic links by issuing **/etc/rc.d/init.d/*<command>* start**.

NOTE

After the system is finished booting, it is possible to log in as root and execute these same scripts to start and stop services. For instance, the command **/etc/rc.d/init.d/httpd stop** stops the Apache HTTP Server.

Each of the symbolic links are numbered to dictate start order. The order in which the services are started or stopped can be altered by changing this number. The lower the number, the earlier it is started. Symbolic links with the same number are started alphabetically.

NOTE

One of the last things the **init** program executes is the **/etc/rc.d/rc.local** file. This file is useful for system customization. Refer to Section F.3, "Running Additional Programs at Boot Time" for more information about using the **rc.local** file.

After the **init** command has progressed through the appropriate **rc** directory for the runlevel, **Upstart** forks an **/sbin/mingetty** process for each virtual console (login prompt) allocated to the runlevel by the job definition in the **/etc/event.d** directory. Runlevels 2 through 5 have all six virtual consoles, while runlevel 1 (single user mode) has one, and runlevels 0 and 6 have none. The **/sbin/mingetty** process opens communication pathways to *tty* devices[18], sets their modes, prints the login prompt, accepts the user's username and password, and initiates the login process.

In runlevel 5, **Upstart** runs a script called **/etc/X11/prefdm**. The **prefdm** script executes the preferred X display manager[19] — **gdm**, **kdm**, or **xdm**, depending on the contents of the **/etc/sysconfig/desktop** file.

Once finished, the system operates on runlevel 5 and displays a login screen.

F.2.5. Job Definitions

Previously, the sysvinit package provided the **init** daemon for the default configuration. When the system started, this **init** daemon ran the **/etc/inittab** script to start system processes defined for each runlevel. The default configuration now uses an event-driven **init** daemon provided by the upstart package. Whenever particular *events* occur, the **init** daemon processes *jobs* stored in the **/etc/event.d** directory. The **init** daemon recognizes the start of the system as such an event.

Each job typically specifies a program, and the events that trigger **init** to run or to stop the program. Some jobs are constructed as *tasks*, which perform actions and then terminate until another event triggers the job again. Other jobs are constructed as *services*, which **init** keeps running until another event (or the user) stops it.

For example, the **/etc/events.d/tty2** job is a service to maintain a virtual terminal on **tty2** from the time that the system starts until the system shuts down, or another event (such as a change in runlevel) stops the job. The job is constructed so that **init** will restart the virtual terminal if it stops unexpectedly during that time:

```
# tty2 - getty
#
# This service maintains a getty on tty2 from the point the system is
# started until it is shut down again.

start on stopped rc2
start on stopped rc3
start on stopped rc4
start on started prefdm

stop on runlevel 0
stop on runlevel 1
stop on runlevel 6

respawn
exec /sbin/mingetty tty2
```

F.3. RUNNING ADDITIONAL PROGRAMS AT BOOT TIME

The **/etc/rc.d/rc.local** script is executed by the **init** command at boot time or when changing runlevels. Adding commands to the bottom of this script is an easy way to perform necessary tasks like starting special services or initialize devices without writing complex initialization scripts in the **/etc/rc.d/init.d/** directory and creating symbolic links.

The **/etc/rc.serial** script is used if serial ports must be setup at boot time. This script runs **setserial** commands to configure the system's serial ports. Refer to the **setserial** man page for more information.

F.4. SYSV INIT RUNLEVELS

The SysV init runlevel system provides a standard process for controlling which programs **init** launches or halts when initializing a runlevel. SysV init was chosen because it is easier to use and more flexible than the traditional BSD-style init process.

The configuration files for SysV init are located in the **/etc/rc.d/** directory. Within this directory, are the **rc**, **rc.local**, **rc.sysinit**, and, optionally, the **rc.serial** scripts as well as the following directories:

```
init.d/ rc0.d/ rc1.d/ rc2.d/ rc3.d/ rc4.d/ rc5.d/ rc6.d/
```

The **init.d/** directory contains the scripts used by the **/sbin/init** command when controlling services. Each of the numbered directories represent the six runlevels configured by default under Red Hat Enterprise Linux.

F.4.1. Runlevels

The idea behind SysV init runlevels revolves around the idea that different systems can be used in different ways. For example, a server runs more efficiently without the drag on system resources created by the X Window System. Or there may be times when a system administrator may need to operate the system at a lower runlevel to perform diagnostic tasks, like fixing disk corruption in runlevel 1.

The characteristics of a given runlevel determine which services are halted and started by **init**. For instance, runlevel 1 (single user mode) halts any network services, while runlevel 3 starts these services. By assigning specific services to be halted or started on a given runlevel, **init** can quickly change the mode of the machine without the user manually stopping and starting services.

The following runlevels are defined by default under Red Hat Enterprise Linux:

- **0** – Halt
- **1** – Single-user text mode
- **2** – Not used (user-definable)
- **3** – Full multi-user text mode
- **4** – Not used (user-definable)
- **5** – Full multi-user graphical mode (with an X-based login screen)
- **6** – Reboot

In general, users operate Red Hat Enterprise Linux at runlevel 3 or runlevel 5 – both full multi-user modes. Users sometimes customize runlevels 2 and 4 to meet specific needs, since they are not used.

The default runlevel for the system is listed in **/etc/inittab**. To find out the default runlevel for a system, look for the line similar to the following near the bottom of **/etc/inittab**:

```
id:5:initdefault:
```

The default runlevel listed in this example is five, as the number after the first colon indicates. To change it, edit **/etc/inittab** as root.

WARNING

Be very careful when editing **/etc/inittab**. Simple typos can cause the system to become unbootable. If this happens, either use a boot CD or DVD, enter single-user mode, or enter rescue mode to boot the computer and repair the file.

For more information on single-user and rescue mode, refer to Chapter 36, *Basic System Recovery*.

It is possible to change the default runlevel at boot time by modifying the arguments passed by the boot loader to the kernel. For information on changing the runlevel at boot time, refer to Section E.9, "Changing Runlevels at Boot Time".

F.4.2. Runlevel Utilities

One of the best ways to configure runlevels is to use an *initscript utility*. These tools are designed to simplify the task of maintaining files in the SysV init directory hierarchy and relieves system administrators from having to directly manipulate the numerous symbolic links in the subdirectories of **/etc/rc.d/**.

Red Hat Enterprise Linux provides three such utilities:

- **/sbin/chkconfig** — The **/sbin/chkconfig** utility is a simple command line tool for maintaining the **/etc/rc.d/init.d/** directory hierarchy.
- **/usr/sbin/ntsysv** — The ncurses-based **/sbin/ntsysv** utility provides an interactive text-based interface, which some find easier to use than **chkconfig**.
- **Services Configuration Tool** — The graphical **Services Configuration Tool** (**system-config-services**) program is a flexible utility for configuring runlevels.

Refer to the chapter titled *Services and Daemons* in the Red Hat Enterprise Linux Deployment Guide for more information regarding these tools.

F.5. SHUTTING DOWN

To shut down Red Hat Enterprise Linux, the root user may issue the **/sbin/shutdown** command. The **shutdown** man page has a complete list of options, but the two most common uses are:

```
/sbin/shutdown -h now
```

and

```
/sbin/shutdown -r now
```

After shutting everything down, the **-h** option halts the machine, and the **-r** option reboots.

PAM console users can use the **reboot** and **halt** commands to shut down the system while in runlevels 1 through 5. For more information about PAM console users, refer to the Red Hat Enterprise Linux Deployment Guide.

If the computer does not power itself down, be careful not to turn off the computer until a message appears indicating that the system is halted.

Failure to wait for this message can mean that not all the hard drive partitions are unmounted, which can lead to file system corruption.

---

[17] GRUB reads ext3 and ext4 file systems as ext2, disregarding the journal file.

[18] Refer to the Red Hat Enterprise Linux Deployment Guide for more information about **tty** devices.

[19] Refer to the Red Hat Enterprise Linux Deployment Guide for more information about display managers.

APPENDIX G. ALTERNATIVES TO BUSYBOX COMMANDS

Unlike previous releases of Red Hat Enterprise Linux, Red Hat Enterprise Linux 6 does not include a version of **busybox** to provide shell commands in the pre-installation and post-installation environments. Table G.1, "Alternatives to busybox commands" contains a list of **busybox** commands, equivalent ways to implement the same functionality in **bash**, and the availability of these alternatives in the %pre and %post environments. The table also indicates the exact path to the command, although you do not generally need to specify the path because the **PATH** environment variable is set in the installation environment.

If a command is only available in %post, the command is running on the target system and its availability therefore depends on whether the package that provides the command is installed. Every command that appears in the "New command or alternative" column of Table G.1, "Alternatives to busybox commands" is available for Red Hat Enterprise Linux 6, although not every command is available on every installed system.

Where a command is listed as unavailable, you might be able to create equivalent functionality with a Python script. The Python language is available to %pre and %post script authors, complete with a set of Python modules ready for use. Therefore, if a particular command is not available to you in the installation environment, we recommend that you use Python as the script language.

Table G.1. Alternatives to busybox commands

Busybox command	*%pre*	*%post*	New command or alternative
addgroup	no	yes	**/usr/sbin/groupadd**
adduser	no	yes	**/usr/sbin/useradd**
adjtimex	no	no	none
ar	no	yes	**/usr/bin/ar**
arping	yes	yes	**/sbin/arping** or **/usr/sbin/arping**
ash	yes	yes	**/bin/bash**
awk	yes	yes	**/sbin/awk**, **/sbin/gawk**, or **/usr/bin/gawk** [a]
basename	yes	yes	**/bin/bash** [b], **/usr/bin/basename**
bbconfig	no	no	none — this command is a specific to **Busybox**
bunzip2	yes	yes	**/usr/bin/bunzip2**, **/usr/bin/bzip2 -d**

Busybox command	*%pre*	*%post*	New command or alternative
busybox	no	no	none
bzcat	yes	yes	**/usr/bin/bzcat**, **/usr/bin/bzip2 -dc**
cal	no	yes	**/usr/bin/cal**
cat	yes	yes	**/usr/bin/cat**
catv	no	no	**cat -vET** or **cat -A**
chattr	yes	yes	**/usr/bin/chattr**
chgrp	yes	yes	**/usr/bin/chgrp**
chmod	yes	yes	**/usr/bin/chmod**
chown	yes	yes	**/usr/bin/chown**
chroot	yes	yes	**/usr/sbin/chroot**
chvt	yes	yes	**/usr/bin/chvt**
cksum	no	yes	**/usr/bin/cksum**
clear	yes	yes	**/usr/bin/clear**
cmp	no	yes	**/usr/bin/cmp**
comm	no	yes	**/usr/bin/comm**
cp	yes	yes	**/usr/bin/cp**
cpio	yes	yes	**/usr/bin/cpio**
crond	no	no	none — no daemons available to scriptlets
crontab	no	yes	**/usr/bin/crontab**
cut	yes	yes	**/usr/bin/cut**
date	yes	yes	**/usr/bin/date**

Busybox command	*%pre*	*%post*	New command or alternative
dc	no	yes	**/usr/bin/dc**
dd	yes	yes	**/usr/bin/dd**
deallocvt	no	yes	**/usr/bin/deallocvt**
delgroup	no	yes	**/usr/sbin/groupdel**
deluser	no	yes	**/usr/sbin/userdel**
devfsd	no	no	none — Red Hat Enterprise Linux does not use **devfs**
df	yes	yes	**/usr/bin/df**
diff	no	yes	**/usr/bin/diff**
dirname	yes	yes	**/bin/bash** [c], **/usr/bin/dirname**
dmesg	yes	yes	**/usr/bin/dmesg**
dnsd	no	no	none — no daemons available to scriptlets
dos2unix	no	no	**sed 's/.$//'**
dpkg	no	no	none — no support for Debian packages
dpkg-deb	no	no	none — no support for Debian packages
du	yes	yes	**/usr/bin/du**
dumpkmap	no	no	none
dumpleases	no	no	none
e2fsck	yes	yes	**/usr/sbin/e2fsck**
e2label	yes	yes	**/usr/sbin/e2label**
echo	yes	yes	**/usr/bin/echo**

Busybox command	*%pre*	*%post*	New command or alternative
ed	no	no	**/sbin/sed**, **/usr/bin/sed**
egrep	yes	yes	**/sbin/egrep**, **/usr/bin/egrep**
eject	yes	yes	**/usr/bin/eject**
env	yes	yes	**/usr/bin/env**
ether-wake	no	no	none
expr	yes	yes	**/usr/bin/expr**
fakeidentd	no	no	none — no daemons available to scriptlets
false	yes	yes	**/usr/bin/false**
fbset	no	yes	**/usr/sbin/fbset**
fdflush	no	no	none
fdformat	no	yes	**/usr/bin/fdformat**
fdisk	yes	yes	**/usr/sbin/fdisk**
fgrep	yes	yes	**/sbin/fgrep**, **/usr/bin/fgrep**
find	yes	yes	**/usr/bin/find**
findfs	no	no	none
fold	no	yes	**/usr/bin/fold**
free	no	yes	**/usr/bin/free**
freeramdisk	no	no	none
fsck	yes	yes	**/usr/sbin/fsck**
fsck.ext2	yes	yes	**/usr/sbin/fsck.ext2**, **/usr/sbin/e2fsck**

Busybox command	*%pre*	*%post*	New command or alternative
fsck.ext3	yes	yes	**/usr/sbin/fsck.ext3**, **/usr/sbin/e2fsck**
fsck.minix	no	no	none — no support for the Minix file system
ftpget	yes	yes	**/usr/bin/ftp** or Python **ftplib** module
ftpput	yes	yes	**/usr/bin/ftp** or Python **ftplib** module
fuser	no	yes	**/sbin/fuser**
getopt	no	yes	**/usr/bin/getopt**
getty	no	no	none
grep	yes	yes	**/sbin/grep**, **/usr/bin/grep**
gunzip	yes	yes	**/usr/bin/gunzip**, **/usr/bin/gzip -d**
gzip	yes	yes	**/usr/bin/gzip**
hdparm	yes	yes	**/usr/sbin/hdparm**
head	yes	yes	**/usr/bin/head**
hexdump	no	yes	**/usr/bin/hexdump**
hostid	no	yes	**/usr/bin/hostid or Python**
hostname	yes	yes	**/sbin/hostname**, **/usr/bin/hostname**
httpd	no	no	none — no daemons available to scriptlets
hush	no	no	none
hwclock	yes	yes	**/usr/sbin/hwclock**

Busybox command	*%pre*	*%post*	New command or alternative
id	no	yes	**/usr/bin/id** or Python
ifconfig	yes	yes	**/sbin/ifconfig**, **/usr/sbin/ifconfig**
ifdown	no	no	**ifconfig** ***device*** **down**
ifup	no	no	**ifconfig** ***device*** **up**
inetd	no	no	none — no daemons available to scriptlets
insmod	yes	yes	**/sbin/insmod**, **/usr/sbin/insmod**
install	no	yes	**/usr/bin/install or mkdir/cp/chmod/chown/chgrp**
ip	yes	yes	**/sbin/ip**, **/usr/sbin/ip**
ipaddr	no	no	**ifconfig** or **ip**
ipcalc	yes	yes	**/sbin/ipcalc**, **/usr/bin/ipcalc**
ipcrm	no	yes	**/usr/bin/ipcrm**
ipcs	no	yes	**/usr/bin/ipcs**
iplink	no	no	**ip**
iproute	no	no	**ip**
iptunnel	no	yes	**/sbin/iptunnel**
kill	yes	yes	**/sbin/kill**, **/usr/bin/kill**
killall	yes	yes	**/usr/bin/killall**
lash	no	no	none
last	no	yes	**/usr/bin/last**
length	no	no	Python or bash

Busybox command	*%pre*	*%post*	New command or alternative
less	yes	yes	**/usr/bin/less**
linux32	no	no	none
linux64	no	no	none
ln	yes	yes	**/sbin/ln**, **/usr/bin/ln**
load_policy	yes	yes	**/sbin/load_policy**, **/usr/sbin/load_policy**
loadfont	no	no	none
loadkmap	no	no	none
login	yes	yes	**/usr/bin/login**
logname	no	yes	**/usr/bin/logname**
losetup	yes	yes	**/usr/bin/losetup**
ls	yes	yes	**/usr/bin/ls**
lsattr	yes	yes	**/usr/bin/lsattr**
lsmod	yes	yes	**/usr/bin/lsmod**
lzmacat	no	yes	**/usr/bin/lzmadec**
makedevs	no	no	**/usr/bin/mknod**
md5sum	yes	yes	**/usr/bin/md5sum**
mdev	no	no	none
mesg	no	yes	**/usr/bin/mesg**
mkdir	yes	yes	**/sbin/mkdir**, **/usr/bin/mkdir**
mke2fs	yes	yes	**/usr/sbin/mke2fs**
mkfifo	no	yes	**/usr/bin/mkfifo**

Busybox command	*%pre*	*%post*	New command or alternative
mkfs.ext2	yes	yes	**/usr/sbin/mkfs.ext2**
mkfs.ext3	yes	yes	**/usr/sbin/mkfs.ext3**
mkfs.minix	no	no	none – no support for Minix filesystem
mknod	yes	yes	**/usr/bin/mknod**
mkswap	yes	yes	**/usr/sbin/mkswap**
mktemp	yes	yes	**/usr/bin/mktemp**
modprobe	yes	yes	**/sbin/modprobe**, **/usr/sbin/modprobe**
more	yes	yes	**/usr/bin/more**
mount	yes	yes	**/sbin/mount**, **/usr/bin/mount**
mountpoint	no	no	Look at the output of the mount command
msh	no	no	none
mt	yes	yes	**/usr/bin/mt**
mv	yes	yes	**/usr/bin/mv**
nameif	no	no	none
nc	no	yes	**/usr/bin/nc**
netstat	no	yes	**/bin/netstat**
nice	no	yes	**/bin/nice**
nohup	no	yes	**/usr/bin/nohup**
nslookup	yes	yes	**/usr/bin/nslookup**
od	no	yes	**/usr/bin/od**

Busybox command	*%pre*	*%post*	New command or alternative
openvt	yes	yes	**/usr/bin/openvt**
passwd	no	yes	**/usr/bin/passwd**
patch	no	yes	**/usr/bin/patch**
pidof	yes	yes	**/usr/sbin/pidof**
ping	yes	yes	**/usr/bin/ping**
ping6	no	yes	**/bin/ping6**
pipe_progress	no	no	none
pivot_root	no	yes	**/sbin/pivot_root**
printenv	no	yes	**/usr/bin/printenv**
printf	no	yes	**/usr/bin/printf**
ps	yes	yes	**/usr/bin/ps**
pwd	yes	yes	**/usr/bin/pwd**
rdate	no	yes	**/usr/bin/rdate**
readlink	yes	yes	**/sbin/readlink**, **/usr/bin/readlink**
readprofile	no	yes	**/usr/sbin/readprofile**
realpath	no	no	Python **os.path.realpath()**
renice	no	yes	**/usr/bin/renice**
reset	no	yes	**/usr/bin/reset**
rm	yes	yes	**/sbin/rm**, **/usr/bin/rm**
rmdir	yes	yes	**/sbin/rmdir**, **/usr/bin/rmdir**

Busybox command	*%pre*	*%post*	New command or alternative
rmmod	yes	yes	**/sbin/rmmod**, **/usr/bin/rmmod**
route	yes	yes	**/sbin/route**, **/usr/sbin/route**
rpm	yes	yes	**/usr/bin/rpm**
rpm2cpio	no	yes	**/usr/bin/rpm2cpio**
run-parts	no	no	none
runlevel	no	no	none
rx	no	no	none
sed	yes	yes	**/sbin/sed**, **/usr/bin/sed**
seq	no	yes	**/usr/bin/seq**
setarch	no	yes	**/usr/bin/setarch**
setconsole	no	no	none
setkeycodes	no	yes	**/usr/bin/setkeycodes**
setlogcons	no	no	none
setsid	no	yes	**/usr/bin/setsid**
sh	yes	yes	**/sbin/sh**, **/usr/bin/sh**
sha1sum	yes	yes	**/usr/bin/sha1sum**
sleep	yes	yes	**/sbin/sleep**, **/usr/bin/sleep**
sort	yes	yes	**/usr/bin/sort**
start-stop-daemon	no	no	none
stat	no	yes	**/usr/bin/stat** or Python **os.stat()**

Busybox command	*%pre*	*%post*	New command or alternative
strings	no	yes	**/usr/bin/strings**
stty	no	yes	**/bin/stty**
su	no	yes	**/bin/su**
sulogin	no	yes	**/sbin/sulogin**
sum	no	yes	**/usr/bin/sum**
swapoff	yes	yes	**/usr/sbin/swapoff**
swapon	yes	yes	**/usr/sbin/swapon**
switch_root	no	yes	**/sbin/switch_root**
sync	yes	yes	**/usr/bin/sync**
sysctl	no	yes	**/sbin/sysctl**
tail	yes	yes	**/usr/bin/tail**
tar	yes	yes	**/usr/bin/tar**
tee	yes	yes	**/usr/bin/tee**
telnet	yes	yes	**/usr/bin/telnet**
telnetd	no	no	none — no daemons available to scriptlets
test	no	yes	**/usr/bin/test** or **[** in bash
tftp	no	yes	**/usr/bin/tftp**
time	no	yes	**/usr/bin/time** or Python
top	yes	yes	**/usr/bin/top**
touch	yes	yes	**/sbin/touch**, **/usr/bin/touch**

Busybox command	*%pre*	*%post*	New command or alternative
tr	no	yes	**/usr/bin/tr** or Python
traceroute	no	yes	**/bin/traceroute**
true	yes	yes	**/usr/bin/true**
tty	no	yes	**/usr/bin/tty**
tune2fs	yes	yes	**/usr/sbin/tune2fs**
udhcpc	no	no	**/sbin/dhclient**
udhcpd	no	no	none — no daemons available to scriptlets
umount	yes	yes	**/sbin/umount**, **/usr/bin/umount**
uname	no	yes	**/bin/uname** or Python **os.uname()**
uncompress	no	no	none
uniq	yes	yes	**/usr/bin/uniq**
unix2dos	no	no	**sed 's/$//'**
unlzma	no	yes	**/usr/bin/unlzma**
unzip	no	yes	**/usr/bin/unzip**
uptime	no	yes	**/usr/bin/uptime** or Python reading **/proc/uptime**
usleep	no	yes	**/bin/usleep** or Python
uudecode	no	yes	**/usr/bin/uudecode** or Python
uuencode	no	yes	**/usr/bin/uuencode** or Python
vconfig	yes	yes	**/usr/sbin/vconfig**

Busybox command	*%pre*	*%post*	New command or alternative
vi	yes	yes	**/usr/bin/vi**
vlock	no	no	none
watch	no	yes	**/usr/bin/watch**
watchdog	no	no	none
wc	yes	yes	**/usr/bin/wc**
wget	yes	yes	**/sbin/wget**, **/usr/bin/wget**
which	no	yes	**/usr/bin/which**
who	no	yes	**/usr/bin/who**
whoami	no	yes	**/usr/bin/whoami**
xargs	yes	yes	**/usr/bin/xargs**
yes	no	yes	**/usr/bin/yes**
zcat	yes	yes	**/usr/bin/zcat**
zcip	no	no	NetworkManager should take care of this

[a] Red Hat Enterprise Linux 6 ships with GNU **awk** rather than the busybox **awk** in the installation environment.

[b] GNU bash can provide basename functionality using string manipulation. If **var="/usr/bin/command"**, then **echo ${var##*/}** gives **command**.

[c] GNU bash can provide dirname functionality using string manipulation. If **var="/usr/bin/command"**, then **echo ${var%/*}** gives **/usr/bin**.

APPENDIX H. OTHER TECHNICAL DOCUMENTATION

To learn more about **anaconda**, the Red Hat Enterprise Linux installation program, visit the project Web page: https://fedoraproject.org/wiki/Anaconda.

Both **anaconda** and Red Hat Enterprise Linux systems use a common set of software components. For detailed information on key technologies, refer to the Web sites listed below:

Boot Loader

Red Hat Enterprise Linux uses the **GRUB** boot loader. Refer to http://www.gnu.org/software/grub/ for more information.

Disk Partitioning

Red Hat Enterprise Linux uses **parted** to partition disks. Refer to http://www.gnu.org/software/parted/ for more information.

Storage Management

Logical Volume Management (LVM) provides administrators with a range of facilities to manage storage. By default, the Red Hat Enterprise Linux installation process formats drives as LVM volumes. Refer to http://www.tldp.org/HOWTO/LVM-HOWTO/ for more information.

Audio Support

The Linux kernel used by Red Hat Enterprise Linux incorporates PulseAudio audio server. For more information about PulseAudio, refer to the project documentation: http://www.freedesktop.org/wiki/Software/PulseAudio/Documentation/User/.

Graphics System

Both the installation system and Red Hat Enterprise Linux use the **Xorg** suite to provide graphical capabilities. Components of **Xorg** manage the display, keyboard and mouse for the desktop environments that users interact with. Refer to http://www.x.org/ for more information.

Remote Displays

Red Hat Enterprise Linux and **anaconda** include VNC (Virtual Network Computing) software to enable remote access to graphical displays. For more information about VNC, refer to the documentation on the RealVNC Web site: http://www.realvnc.com/support/documentation.html.

Command-line Interface

By default, Red Hat Enterprise Linux uses the GNU **bash** shell to provide a command-line interface. The GNU Core Utilities complete the command-line environment. Refer to http://www.gnu.org/software/bash/bash.html for more information on **bash**. To learn more about the GNU Core Utilities, refer to http://www.gnu.org/software/coreutils/.

Remote System Access

Red Hat Enterprise Linux incorporates the OpenSSH suite to provide remote access to the system. The SSH service enables a number of functions, which include access to the command-line from other systems, remote command execution, and network file transfers. During the installation process **anaconda** may use the **scp** feature of OpenSSH to transfer crash reports to remote systems. Refer to the OpenSSH Web site for more information: http://www.openssh.com/.

Access Control

SELinux provides Mandatory Access Control (MAC) capabilities that supplement the standard Linux security features. Refer to the SELinux Project Pages for more information: http://www.nsa.gov/research/selinux/index.shtml.

Firewall

The Linux kernel used by Red Hat Enterprise Linux incorporates the **netfilter** framework to provide firewall features. The Netfilter project website provides documentation for both **netfilter**, and the **iptables** administration facilities: http://netfilter.org/documentation/index.html.

Software Installation

Red Hat Enterprise Linux uses **yum** to manage the RPM packages that make up the system. Refer to http://yum.baseurl.org/ for more information.

Virtualization

Virtualization provides the capability to simultaneously run multiple operating systems on the same computer. Red Hat Enterprise Linux also includes tools to install and manage the secondary systems on a Red Hat Enterprise Linux host. You may select virtualization support during the installation process, or at any time thereafter. Refer to the *Red Hat Enterprise Linux Virtualization* documentation available from https://access.redhat.com/documentation/en/red-hat-enterprise-linux/ for more information.

APPENDIX I. REVISION HISTORY

Note that revision numbers relate to the edition of this manual, not to version numbers of Red Hat Enterprise Linux.

Revision 1.0-138 **Tue Mar 14 2017** **Petr Bokoč**
Asynchronous update.

Revision 1.0-137 **Tue Mar 14 2017** **Petr Bokoč**
Red Hat Enterprise Linux 6.9 General Availability release.

Revision 1.0-131 **Tue Mar 11 2016** **Clayton Spicer**
Red Hat Enterprise Linux 6.8 GA release.

Revision 1.0-127 **Fri 10 Jul 2015** **Petr Bokoč**
Red Hat Enterprise Linux 6.7 GA release

Revision 1.0-112 **Wed Oct 08 2014** **Petr Bokoč**
Red Hat Enterprise Linux 6.6 GA release

Revision 1.0-102 **Thu Nov 07 2013** **Petr Bokoč**
Red Hat Enterprise Linux 6.5 GA release

Revision 1.0-96 **Tue Feb 19 2013** **Jack Reed**
Second version for Red Hat Enterprise Linux 6.4 GA release

Revision 1.0-95 **Sun Feb 17 2013** **Jack Reed**
Red Hat Enterprise Linux 6.4 GA release

Revision 1.0-41 **Thu May 19 2011** **Rüdiger Landmann**
Red Hat Enterprise Linux 6.1 GA release

Revision 1.0-0 **Wed Aug 25 2010** **Rüdiger Landmann**
Red Hat Enterprise Linux 6.0 GA release

INDEX

Symbols

A

B

C

D

E

F

H

L

M

P

R

S

T

U

V

X

Y

Z