

Oracle Linux 9

Installing Oracle Linux



F51031-10
January 2024



Oracle Linux 9 Installing Oracle Linux,

F51031-10

Copyright © 2022, 2024, Oracle and/or its affiliates.

Contents

Preface

Documentation License	v
Conventions	v
Documentation Accessibility	v
Access to Oracle Support for Accessibility	v
Diversity and Inclusion	vi

1 Preparing to Install

System Requirements	1-1
System Configuration	1-1
Obtaining Installation Images	1-2
Preparing Installation Media	1-3

2 Customizing Boot Loader Actions

Configuring the Boot Loader	2-1
Installation Boot Options	2-1

3 Installing Oracle Linux Manually

Installing in Graphics Mode	3-1
Localization Options	3-3
Software Options	3-5
System Options	3-7
User Settings	3-9
Completing the Installation	3-10
Installing in Text Mode	3-10
Postinstallation Configuration	3-11

4 Automating the Installation by Using Kickstart

Customizing the Kickstart Configuration File	4-1
----------------------------------------------	-----

	Installation Options	4-2
	Packages to Install	4-3
	Preinstallation Options	4-5
	Postinstallation Options	4-5
	Starting the Kickstart Installation	4-6
5	Installing Using the Driver Update Disk	
6	Installing a System With a Btrfs root File System	
	Using the GUI Installer	6-1
	Using the Text Installer	6-3
	Verifying the Btrfs Installation	6-3
7	Troubleshooting Oracle Linux Installations	
	Troubleshooting Menu Options	7-1
	Debug and Log Information	7-1
A	Creating a Network Installation Setup	
	Setting Up the Network Server	A-1
	How to Configure NFS	A-1
	How to Configure dnsmasq	A-3
	Configuring PXE Boot Loading	A-5
	How to Configure for BIOS-based PXE Clients	A-5
	How to Configure for UEFI-based PXE Clients	A-7
	Supporting Different Kinds of Clients	A-9

Preface

[Oracle Linux 9: Installing Oracle Linux](#) provides information about how to install the Oracle Linux 9 release.

Documentation License

The content in this document is licensed under the [Creative Commons Attribution–Share Alike 4.0 \(CC-BY-SA\)](#) license. In accordance with CC-BY-SA, if you distribute this content or an adaptation of it, you must provide attribution to Oracle and retain the original copyright notices.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
<code>monospace</code>	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <https://www.oracle.com/corporate/accessibility/>.

For information about the accessibility of the Oracle Help Center, see the Oracle Accessibility Conformance Report at <https://www.oracle.com/corporate/accessibility/templates/t2-11535.html>.

Access to Oracle Support for Accessibility

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <https://www.oracle.com/corporate/accessibility/learning-support.html#support-tab>.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

1

Preparing to Install

Whether you are installing on a single system or on multiple systems, you should plan ahead to ensure a successful installation.

- [System Requirements](#)
- [System Configuration](#)
- [Obtaining Installation Images](#)
- [Preparing Installation Media](#)



Note:

To upgrade from the latest version of the previous major Oracle Linux release to the current major release version, use the Leapp utility. See [Oracle Linux 9: Upgrading Systems With Leapp](#).

System Requirements

Verify that your system fulfills the following minimum requirements. In general, having more resources in the system improves a system's performance.

- Minimum of 2 logical CPUs up to 2048 logical CPUs
- 1.5 GB of memory per logical CPU, up to a maximum of 64 TB
- At least 10 GB of disk space (20 GB is the recommended minimum)
- On UEFI systems, ensure that the target disk uses GPT (GUID Partition Table), as some UEFI firmwares do not support UEFI/MBR boot

Refer to the following additional resources for information related to installation issues and system requirements:

- <https://linux.oracle.com/hardware-certifications>
- The Release Notes of the Oracle Linux version you are installing at [General Oracle Linux documentation](#)
- [Unbreakable Enterprise Kernel documentation](#)

System Configuration

The configuration of the system itself also has an impact on the installation. Consider the following:

Storage

- Storage device and partition on which the operating system is installed. The installation program displays a warning if disk space is insufficient.
- Storage space for each file system (`/`, `/boot`, `/home`, `/var/tmp`, and so on), the file system type, and whether the block device underlying each file system should be encrypted.
- Layout and configuration of the storage devices such as use of logical volume management RAID configuration, encryption, and others.
- For iSCSI or FCoE connections, the WWID or the port, target, and LUN to be used.

Network

- Required network setup using DHCP or static addresses, FQDN or host name, and so on.
- Other specialized network interfaces to be configured during installation, such as VLANs and network bonding.

Software

- URLs of any additional repositories and proxy settings to be installed.
- Software packages to be installed based on system's intended purpose, such as a web server.

Obtaining Installation Images

To install Oracle Linux, download the installation images from the following locations:

- Oracle Software Delivery Cloud at <https://edelivery.oracle.com>.
- Oracle Linux yum server at <https://yum.oracle.com/oracle-linux-isos.html>.
Note, however, that the Oracle Linux yum server does not provide equivalent repositories for some channels that are available on Unbreakable Linux Network (ULN). These channels provide non-open source packages. For more information about ULN, see [Oracle Linux: Managing Software on Oracle Linux](#).

The following installation images are available for both the x86_64 platform and the aarch64 platform, unless indicated otherwise:

- Full ISO of Oracle Linux for typical installations on systems that are located locally or remotely.
- Boot ISO of Oracle Linux for network installations or if the installation media lacks space to store the full ISO.
- Boot ISO of the supported UEK release for installations in which you want to use the Btrfs file system or for installing on hardware that is supported only on UEK. For instructions to install Oracle Linux with the Btrfs files system, see [Installing a System With a Btrfs root File System](#).
- Source DVDs that contain the source code for the software packages in the release.

Aside from installation ISOs, you can also use Oracle Linux images to create compute instances on Oracle Cloud Infrastructure (OCI). For information about these images, see the release notes for the specific image that you are using on the Oracle Cloud

Infrastructure Documentation page at <https://docs.oracle.com/iaas/images/>. For information about using Oracle Linux instances, <https://docs.oracle.com/en-us/iaas/oracle-linux/home.htm>.

For information about the available ISOs for the three most recent updates to the Oracle Linux releases, refer to <https://yum.oracle.com/oracle-linux-isos.html>.

Preparing Installation Media

Before you can use an ISO image to install Oracle Linux, you must first store it in a bootable installation media, such as the following:

- [USB Flash Drive](#)
- [DVD or CD](#)
- [Network Drive](#)
- [Driver Update Disk](#)

USB Flash Drive

You can install Oracle Linux by using a boot image on portable devices such as a USB flash drive or an SD card, provided that the system's firmware supports booting from those devices.

To create a bootable drive, use the `dd` command. Or, use a separate third-party utility to write the ISO image to a drive. See, for example, [Create USB Installation Media for Oracle Linux with Fedora Media Writer](#).

Caution:

This procedure destroys any existing data on the drive. Ensure that you specify the correct device name for the USB drive on your system.

1. Insert a USB flash drive into an Oracle Linux system.
2. Check the file systems that are on the drive.

```
sudo df /media/usb
```

Filesystem	1K-blocks	Used	Available	Use%	Mounted on
/dev/sdb1	35346	35346	0	100%	/media/USB

3. Unmount the file system.

```
sudo umount /dev/sdb1
```

4. Write the contents of the ISO image file to the USB device, for example:

```
sudo dd if=./full_image.iso of=/dev/sdb bs=512k
```

The USB flash drive is now ready to be used to boot a system and start the installation.

DVD or CD

Due to storage limits, CDs or DVDs might no longer be sufficient to accommodate ISO images. Or, they can store only the boot ISO image.

To write the downloaded ISO image file to a CD or DVD, use a command such as `cdrecord`, for example:

```
sudo cdrecord -v -eject speed=16 dev=ATA:0,2,0 file_name.iso
```

To display the SCSI subsystem and device that corresponds to the CD or DVD writer, use the `cdrecord -scanbus` command.

Network Drive

For the network drive, the image that you download can either be the full ISO image or just the boot image, while the network server can be of any type, such as NFS or a web server.

To copy the ISO image to a network drive, first mount the image and then use the following command syntax:

```
sudo cp -a -T path-to-mounted-ISO-image network_dir
```

For example, if you are using a web server to host the network drive, and have mounted the ISO image on the server's `/mnt` location, you could run the following command:

```
sudo cp -a -T /mnt /var/www/html/OSimage/OL9
```

When the command completes, you can boot the system from the network drive.



Note:

Using a network drive is part of network installation, which requires you to build a network configuration that provides network installation functionality. For details, see [Creating a Network Installation Setup](#).

Driver Update Disk

A Driver Update Disk (DUD) provides a mechanism for delivering updated device drivers during system installation. On some systems, hardware might not be fully supported for an Oracle Linux release. In these cases, a DUD may be released at a later date to facilitate installation on newer hardware.

DUDs are released as modules and become available for previously unsupported hardware. The DUD is usually in the form of an ISO and is available in the Oracle Software Delivery Cloud or through MyOracle Support.

Like the other ISOs, the DUD must be stored in an appropriate media or an alternative storage device before it can be used to for installation.

NOT_SUPPORTED:

Writing the DUD ISO to the wrong device may result in data loss. Check that you reference the correct device node when you copy the image. Use the `lsblk` utility to help you to identify the different disks block devices on your system.

To use the driver update disk, issue the `dd` command line utility to transfer the DUD image to a separate storage device:

```
sudo dd if=/path/to/DUD.iso of=/dev/sdX
```

- `/path/to/DUD.iso` refers to the full path to the location of the DUD ISO file.
- `/dev/sdX` refers to the device node of the storage media that you intend to use for the DUD.

As an alternative, you can transfer the DUD image to a location that is accessible over the network such as a web server or an NFS server.

2

Customizing Boot Loader Actions

Boot options determine how the installation proceeds. In addition to default settings, you can add more options before you start the installation.

- [Configuring the Boot Loader](#)
- [Installation Boot Options](#)

Configuring the Boot Loader

Boot options are specified at the boot command line of the boot menu.

1. Boot the system.
2. When the boot menu is displayed, select any installation option, then press either **E** on UEFI-based systems or **Tab** on BIOS-based systems.

The boot options line is displayed with some default options already defined.

3. Add options at the end of the line.

Separate multiple boot options with a space. Options that require parameters must be in the `option=parameter` format.

For a list of common boot options, see [Installation Boot Options](#).

4. Save the changes by pressing either **Ctrl+X** on UEFI-based systems or **Return** on BIOS-based systems.

To discard your changes and return to the boot menu, press **Esc**.

This example shows settings for the `inst.repo` and `inst.ks` options:

```
inst.repo=nfs:nfs.example.com:/ISOs/OL9/full_image.iso \  
inst.ks=nfs:nfs.example.com:/kickstart/OL9/server-ks.cfg ip=dhcp
```

With these directives, the installation program is instructed to do the following:

- Use the full installation image stored on an NFS share directory.
- Start the installation automatically by using a file also stored on an NFS share directory.

Installation Boot Options

This list contains some of the most commonly used boot options that further determine how the installation proceeds. You specify these options at the installation menu before you initiate the installation process.

You can refer to these additional information sources for more boot options:

- <https://anaconda-installer.readthedocs.io/en/latest/boot-options.html>
- The `dracut.cmdline(7)` manual page provides Dracut kernel command line options.

Installation Type

inst.graphical

Specifies a graphical-based installation.

inst.gpt

Applies only to BIOS-based systems with less than 2 TiB sized disk. Specifies to use the GUID partition table when boot loader is installed instead of the default MBR.

Installation Source

inst.repo=cdrom[:device]

Specifies a CD or DVD drive as the location that contains everything needed to install the software.

The installation program searches all the system's CD or DVD drives, unless a *device* is specified.

If access to a network is required and no network boot options are specified, the installation program enables DHCP on all available network devices.

inst.repo=ftp://user : password@FTP_server/path

Specifies an FTP server as the location that contains everything needed to install the software.

If access to a network is required and no network boot options are specified, the installation program enables DHCP on all available network devices.

inst.repo=hd:device:path

Specifies a local disk as the location that contains everything needed to install the software.

You can specify the *device* by its device name (*sdb2*), label (*LABEL=label*), or UUID (*UUID=uuid*).

inst.repo=[http:|https:]//HTTP_server/path

Specifies a web server as the location that contains everything needed to install the software. If your system has access to the Internet, you can use the BaseOS repository on the Oracle Linux yum server for the Oracle Linux release that you are installing. For example, you could set this value to `https://yum.oracle.com/repo/OracleLinux/OL9/baseos/latest/x86_64`

inst.repo=nfs:[options:]NFS_server:path

Specifies an NFS share as the location that contains everything needed to install the software.

Use *options* to specify a comma-separated list of NFS mount options.

The NFS share can be the path to an ISO image or a directory.

inst.stage2=[installation_source]

Specifies the location to fetch the installer runtime image; packages are ignored. If this option is not specified, *inst.repo* is used instead. The path specified for *installation_source* can match any of the protocol and path options used for *inst.repo*.

The directory path specified for the *installation_source* should contain a valid *.treeinfo* file that specifies the location of the runtime image. If a *.treeinfo* file is not present at the source, the installer uses `LiveOS/squashfs.img` as the default location at the specified source.

! Important:

When specifying a runtime image in UEK network installations, use the UEK ISO for the installation repository as well. Do not combine using the standard Oracle Linux ISO image as the installation repository while specifying the Oracle Linux UEK Boot ISO for the runtime image.

Network Configuration**ip=[*interface*:]dhcp|dhcp6|auto6|ibft**

Specifies a network automatic configuration method. If *interface* is not specified, all interfaces are configured. Use *ibft* to use the MAC address of the interface specified by the iSCSI Boot Firmware Table (iBFT) in the system BIOS or firmware.

ip=*ip::gateway:netmask:hostname:interface:none*

Specifies a static IP configuration for *interface*. Enclose IPv6 addresses in square brackets, for example `[2509:f0d0:1001::0004]`.

nameserver=*IP*

Specifies the IP address of a DNS server to use during installation. Multiple *nameserver* options can be used.

bootdev=*interface*

Specifies the primary network interface. Required if you use more than one *ip* option.

inst.dhcpclass

Specifies a vendor class identifier to DHCP.

Kickstart Installations**inst.ks=*cdrom[:device]/path***

Specifies a kickstart file on a CD or DVD drive.

inst.ks=*ftp://user : password@FTP_server/path*

Specifies a kickstart file on an FTP server.

inst.ks=*hd:device:path*

Specifies a kickstart file a local disk.

You can specify the *device* by its device name (*sdb2*), label (`LABEL=label`) or UUID (`UUID=uuid`).

inst.ks=[*http:|https:|/HTTP_server/path*

Specifies a kickstart file on a web server.

inst.ks=*nfs:[options:]NFS_server:path*

Specifies a kickstart file on an NFS share.

Use *options* to specify a comma-separated list of NFS mount options.

Remote Installations**inst.vnc**

Enables remote graphical-based installation by starting a VNC server.

A VNC client can connect by using a command such as `vncviewer server:port`, where *server* is the IP address of the system being installed. After installation, the system starts in text mode even if a graphical desktop environment is selected as the base environment.

`inst.vncconnect=client[:port]`

Specifies the VNC client and optional port that is listening for connections from a VNC server (`vncviewer -listen`). The default port is 5900.

`inst.vncpassword=password`

Specifies the password for client connections using VNC.

Miscellaneous Boot Options

`inst.keymap=layout`

Specifies the keyboard layout for installation.

`inst.lang=language`

Specifies the language for installation.

`inst.loglevel=level`

Specifies the logging level for installation: `critical`, `debug`, `error`, `info`, or `warning`. The default level is `info`.

3

Installing Oracle Linux Manually

Installation begins when you boot the system from the installation media. The installation menu displays 3 options:

- Install Oracle Linux
- Test the media and then install Oracle Linux (default)
- Troubleshooting

Two modes are available to manually install Oracle Linux:

- [Graphics mode](#) where a graphical user interface guides you through the installation process. Selecting either the first or the second option in the menu starts the installation in graphic mode by default.
- [Text mode](#) which has limited options for installing the operating system. To use this mode, you must first configure the boot process further by following these steps:
 1. From the boot menu, press Tab to access the boot line options.
 2. At the end of this line, add `inst.text`.
 3. Continue booting.

At the end of the boot process, the text-based installation menu is displayed.

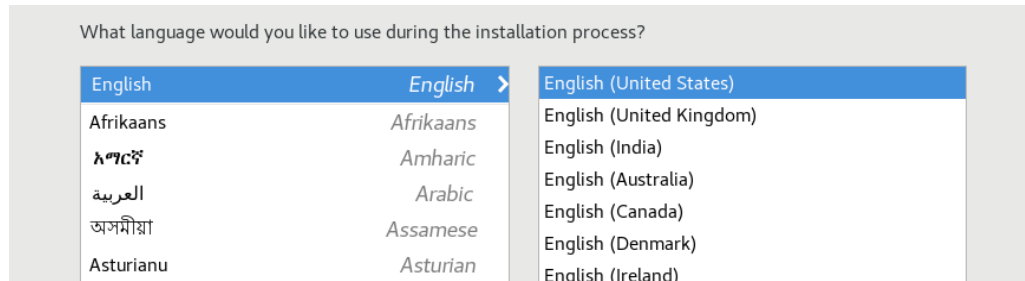
Installing in Graphics Mode

The graphics mode consists of installation directives in a series of graphical screens where you click representative icons and buttons to set the directives.

- [Welcome Screen](#)
- [Installation Summary](#)
- [Localization Options](#)
- [Software Options](#)
- [System Options](#)
- [User Settings](#)

Welcome Screen

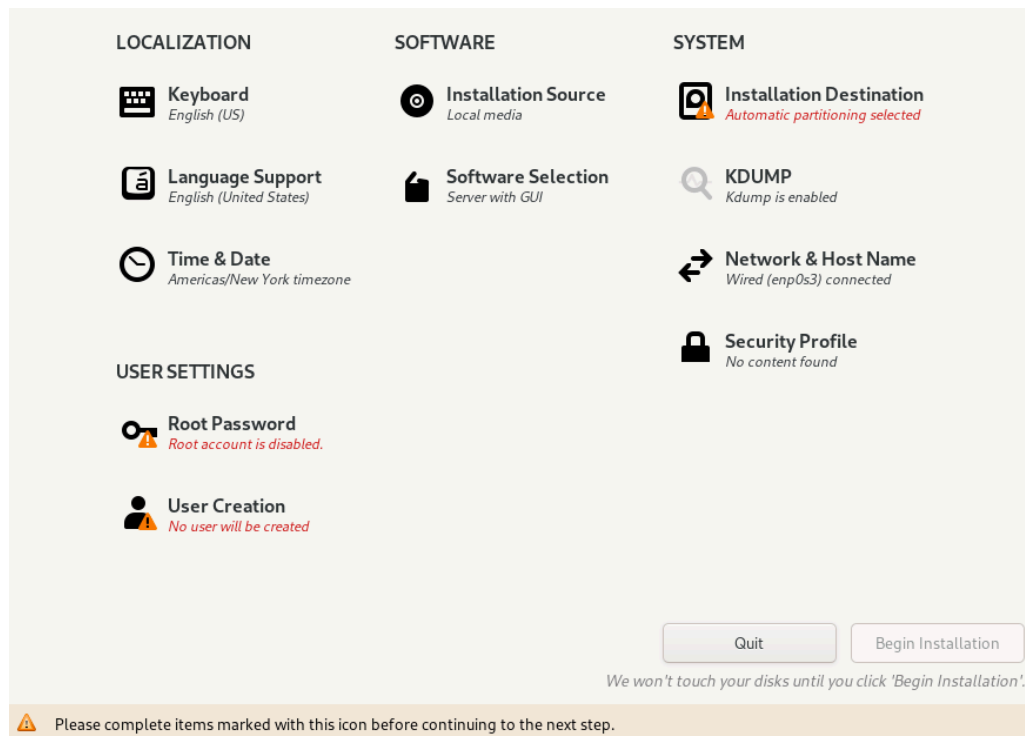
The Welcome Screen is the first screen to appear after the system completes the boot process.

Figure 3-1 Welcome Screen

On this screen, select the preferred language to be used during the installation process. You can further select a specific locale, if any, for your selected language. Then click **Continue** to proceed.

Installation Summary

The Installation Summary screen appears after you have selected the installation language. It functions as the home or main screen.

Figure 3-2 Installation Summary

The screen provides four categories of options: Localization, Software, System, and User Settings. Clicking an option under any of these categories opens additional screens to configure the selected option.

These options have default values. If you consider the values sufficient, you can skip opening those options' screens. However, as a minimum, you **must** visit the options flagged with a warning icon.

After defining directives in a specific screen, click **Done** to return to the Installation Summary screen. Then you can configure other options. As you visit each configuration screen, pay attention to any warning messages that are displayed at the bottom of the screen.

You can change the installation configuration options as much as you like. The installation does not begin until you click **Begin Installation** at the bottom of the screen. In turn, the **Begin Installation** button remains disabled until all the configuration warning flags have been cleared.

 **Note:**

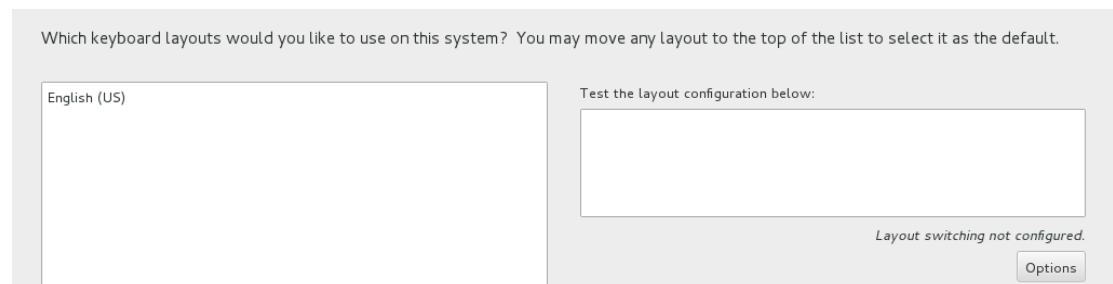
At the top right of the screen is the Keyboard switch. This switch appears in all of the option screens to enable you to change to a different available keyboard layout at any time during configuration. See [Keyboard Layout](#) for more details about keyboard layouts.

Localization Options

Under Localization, you configure the keyboard layouts, languages that would be supported, and the date, time, and time zone of the system. After configuring any of these options, click **Done** to return to the Installation Summary screen.

Keyboard Layout

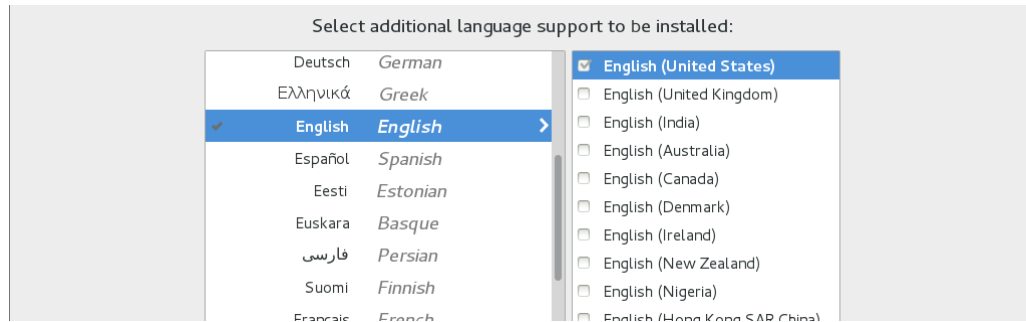
Figure 3-3 Keyboard Layout



On the left pane, you can add other keyboard layouts to the preselected default layout. Additionally, you can revise the order of the listed layouts. The layout at the top of the list becomes the default layout.

Language Support

Configuring language support consists of specifying additional locales of the selected language that you want the system to support. This option is similar to the configuration of the keyboard layout at the beginning of the installation.

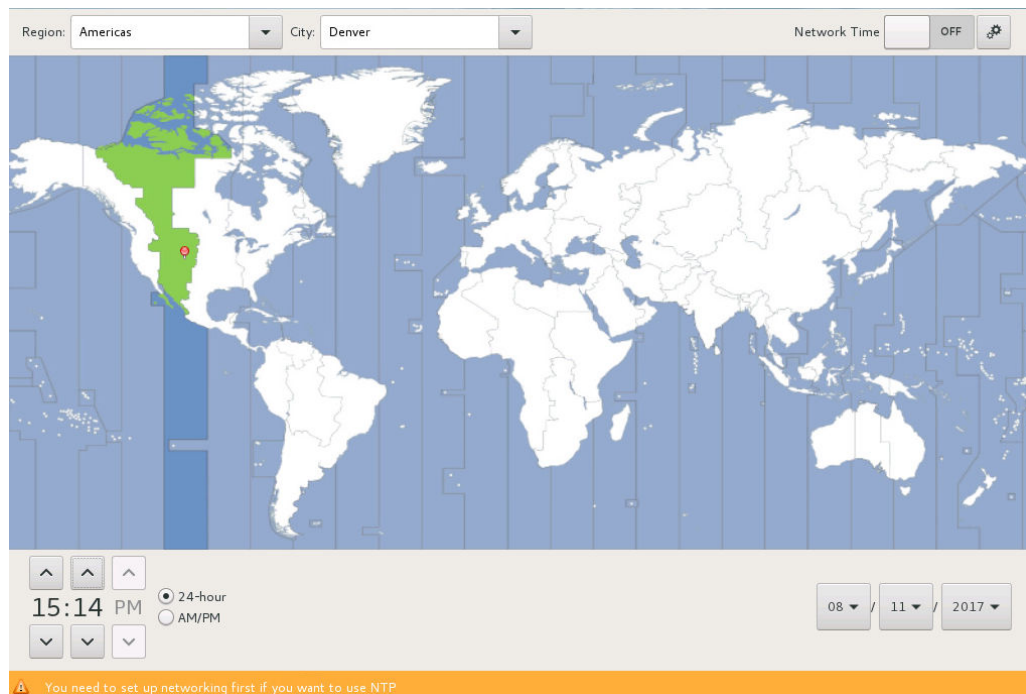
Figure 3-4 Language Support

From the list of languages on the left pane, select additional languages for the system. Then, on the right pane, select from the available locales for that language.

Date and Time

The Date & Time screen enables you to set the following options:

- Time zone for your system
- Actual time and the format for displaying time
- Current date

Figure 3-5 Date & Time

To select the system's time zone, click an area on the map that approximates the site that you want the system to be located. Alternatively, you can select from the Region and City drop down lists to specify that location.

To specify time, time format, and date, click the appropriate buttons at the bottom of the screen.

To enable NTP, toggle the **Network Time** switch at the upper right corner of the screen to ON. Click **Settings** to display a dialog box, where you can configure the NTP servers that the system would use. As an alternative, you can skip configuring NTP until later by using the Chrony suite. See [Oracle Linux 9: Setting Up Networking](#). See also [Configure Chrony on Oracle Linux](#).

Software Options

Software screens enable you to specify the installation source and the profile to be installed. After configuring any of these options, click **Done** to return to the Installation Summary screen.

Figure 3-6 Installation Source

The Installation Source window identifies the source image that you use to install Oracle Linux.

If you use the full ISO image as the source, the install program detects that image. By default, the Auto-detected installation media button is selected and Appstream is listed as an additional installation repository, as shown in the following figure:

Which installation source would you like to use?

Auto-detected installation media:

Device: sr0
Label:

On the network:

URL type:

Additional repositories

Enabled	Name
<input checked="" type="checkbox"/>	AppStream

Name:

file://

URL type:

Proxy URL:

User name:

Password:

The configuration in the image is sufficient to proceed with the installation.

To install additional packages from other sources, use the Additional repositories box to add those sources, and then provide the necessary information about those sources in the text fields next to the box.

If you use the boot ISO image as the source, the install program detects that image and selects the On the network button as the installation mode. Further, Closest mirror is selected as the repository source, as shown in the following figure:

Which installation source would you like to use?

Auto-detected installation media:

Device: sr0 Verify

Label:

On the network:

Closest mirror:

URL type: repository URL

Additional repositories

Enabled	Name
<input type="checkbox"/>	<input type="text"/>

Name:

URL type: http://

URL type: repository URL

Proxy URL:

User name:

Password:

The configuration in the image is sufficient to proceed with the installation. Optionally, you can specify a local mirror as a repository source, in which case you would need to provide the mirror's path.

In a boot installation, the installer automatically uses the required repositories from the indicated source to install the OS. However, these repositories are not listed in the window.

To install additional packages from other sources, use the Additional repositories box to add those sources, and then provide the necessary information about those sources in the text fields next to the box.

Figure 3-7 Software Selection

Software selection refers to the profile or base environment to be used during the installation.

Base Environment

- Server with GUI**
An integrated, easy-to-manage server with a graphical interface.
- Server**
An integrated, easy-to-manage server.
- Minimal Install**
Basic functionality.
- Workstation**
Workstation is a user-friendly desktop system for laptops and PCs.
- Custom Operating System**
Basic building block for a custom OL system.
- Virtualization Host**
Minimal virtualization host.

Additional software for Selected Environment

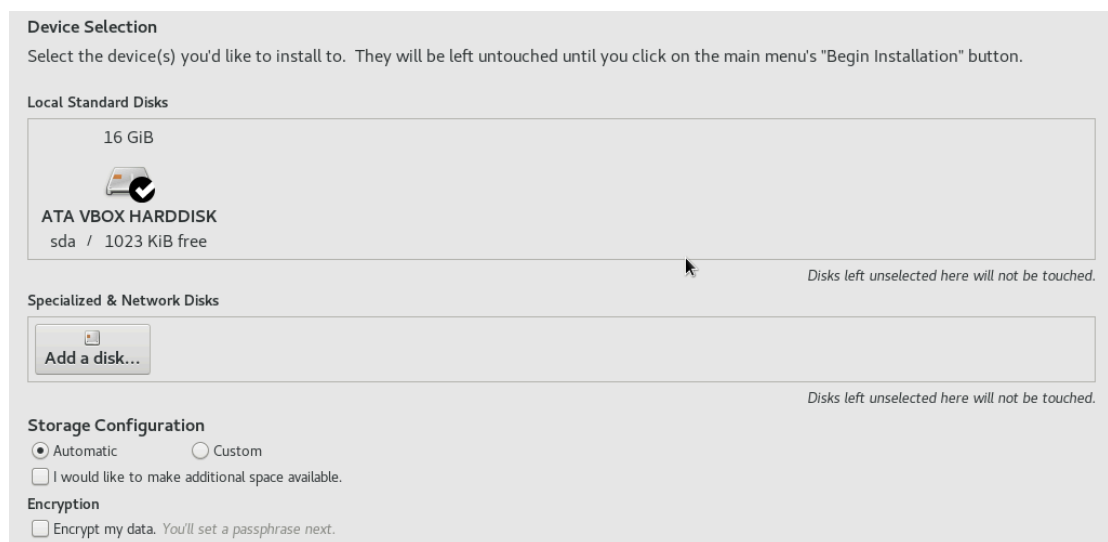
- Guest Agents**
Agents used when running under a hypervisor.
- Standard**
The standard installation of Oracle Linux.
- Legacy UNIX Compatibility**
Compatibility programs for migration from or working with legacy UNIX environments.
- Console Internet Tools**
Console internet access tools, often used by administrators.
- Container Management**
Tools for managing Linux containers
- Development Tools**
A basic development environment.
- .NET Development**
Tools to develop and/or run .NET applications
- Graphical Administration Tools**
Graphical system administration tools for managing many aspects of a system.
- Headless Management**
Tools for managing the system without an attached graphical console.

Each profile represents a set of functionalities and the required packages and software to fulfill those functionalities. By default, the Server with GUI profile is used. Choose the profile or base environment that best fits the purpose of the system on which you are installing Oracle Linux. From the upper-right pane, you can add profiles to the base profile you selected on the left pane.

System Options

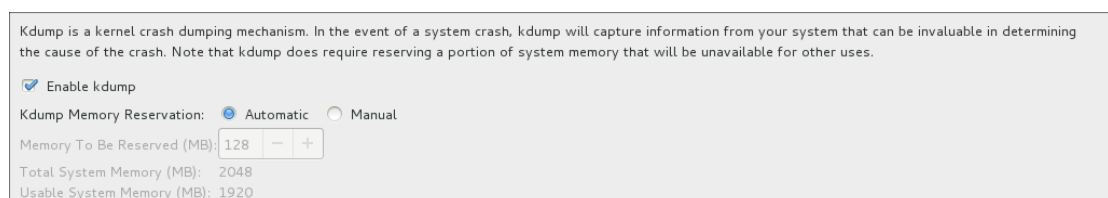
The set of options under System enables you to specify where the operation system is installed, configure Kdump, the network, and the security profile. After configuring any of these options, click **Done** to return to the Installation Summary screen.

Figure 3-8 Installation Destination



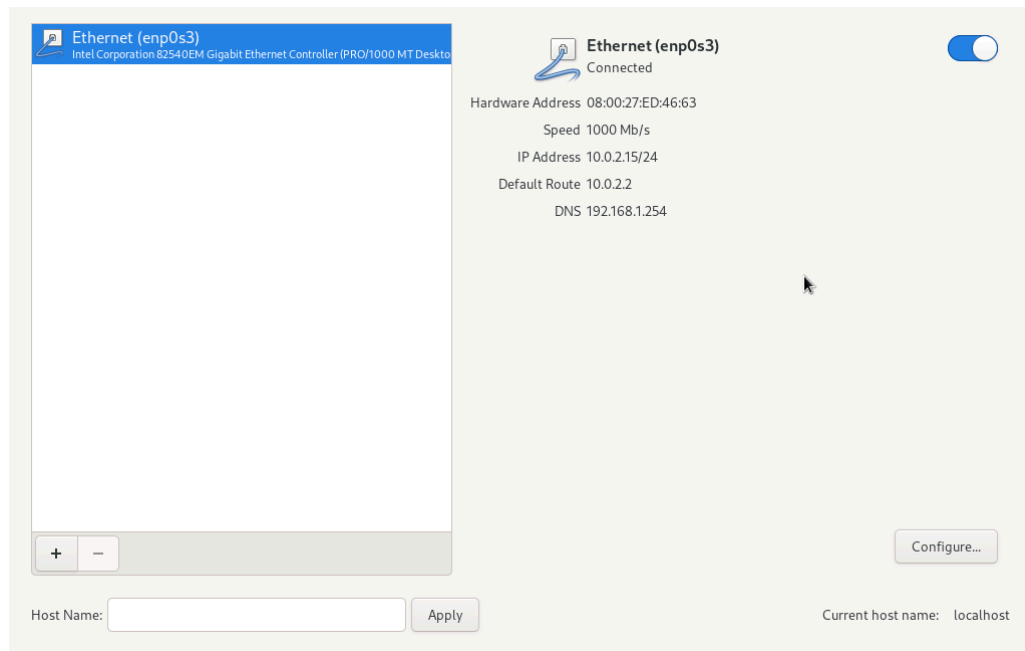
Even if you accept the default settings, you must still open the Installation Destination screen to clear the option's warning icon. Other disk configurations for the installation can be completed in this screen, such as enabling disk encryption, configuring storage, creating partitions, adding disks, and so on.

Figure 3-9 Kdump

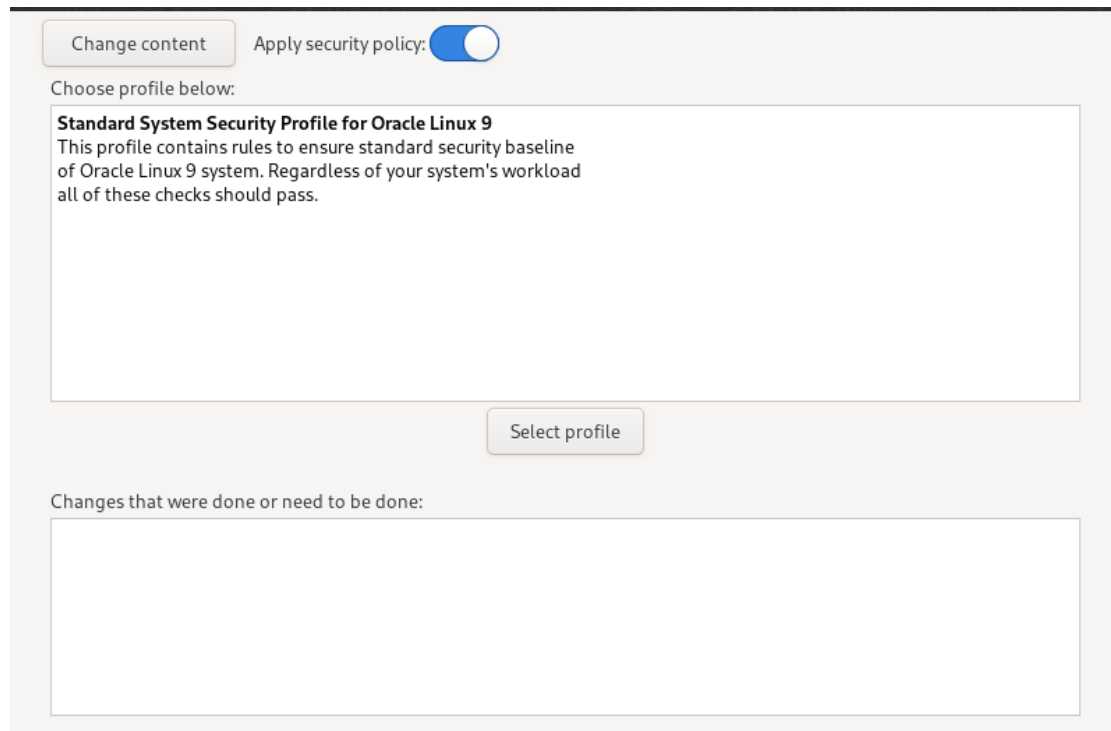


In the event of a system crash, Kdump captures information that assists in determining the cause. By default, Kdump is enabled and the amount of memory reserved for Kdump is calculated automatically. Select the **Manual** option if you want to set the amount of reserved memory yourself.

Figure 3-10 Network & Host Name



Beginning with Oracle Linux 9, network connections are automatically enabled. By default, network configuration uses DHCP for IPv4 addresses. IPv6 addresses are configured automatically. The default settings are generally sufficient for the system to provide basic network functionality. However, you can customize the network configuration, for example, by providing a custom host name, including a fully qualified domain name (FQDN). You can further opt to use static addresses instead of using DHCP, configure proxy settings, network bonds, and so on. To do these advanced types of configuration, click **Configure** and go through additional configuration screens.

Figure 3-11 Security Policy

Because security policy is not required on all systems, use the Security Policy screen only if you need to enforce a specific security policy as defined by your organization or by government regulations.

Available policies are listed in the profile window. These security policies follow the recommendations and guidelines that are defined by the Security Content Automation Protocol (SCAP) standard. Select the security profile you want to enforce and ensure that the Apply security policy is switched to On.

If you prefer not apply any security policy, toggle the switch to Off.

For more information on SCAP policies and profiles, see [Oracle Linux 9: Using OpenSCAP for Security Compliance](#).

User Settings

The User Settings section has the following options, with each option having its own configuration window.

- Root Password

In the Root Password window, aside from creating the `root` password, you also have the following additional options:

- Lock or unlock the root account. By default, the root account is locked as a security feature implementation.
- Allow `root` SSH login with password.

Keeping the root account locked is good security practice.

After creating the password and completing the optional settings as desired, click **Done**.

 **Note:**

The locked setting of the `root` account might affect automated installations by using `kickstart`. To enable SSH root login in the `kickstart`'s configuration file, see [Postinstallation Options](#).

- User Creation

This option enables you to configure a user's credentials to enable access to the system. You can optionally configure the user to have administrative privileges.

- Create a user with the required information.
- Optionally, grant the user administrator privileges.
The privileges enable the user to issue administrator commands by using `sudo`.
- To set additional user configurations, such as the user's home directory, click **Advanced**.

After completing root and user configuration, click **Done** to return to the Installation Summary screen.

Completing the Installation

From the Installation Summary screen, click **Begin Installation**. This button becomes available only when warning flags on option icons have been cleared.

The installation takes a while. After it finishes, reboot the system as prompted. Then log in to the system by using the credentials you previously set in User Settings.

For additional configurations you can set on the system, see [Postinstallation Configuration](#).

Installing in Text Mode

The Oracle Linux installation program can also run in text mode. Text mode is used automatically under certain circumstances, for example, if the system has insufficient memory or the video card is not supported. You can manually switch to the text mode by specifying `inst.text` as a boot option on the boot command line or by adding that option in a `kickstart` file.

Figure 3-12 Text-Based Installation Menu

```

Installation

1) [x] Language settings                2) [!] Time settings
   (English (United States))           (Timezone is not set.)
3) [!] Installation source             4) [!] Software selection
   (Processing...)                     (Processing...)
5) [!] Installation Destination        6) [x] Kdump
   (No disks selected)                 (Kdump is enabled)
7) [ ] Network configuration           8) [!] Root password
   (Not connected)                     (Password is not set.)
9) [!] User creation
   (No user will be created)

Please make your choice from above ['q' to quit | 'b' to begin installation |
'r' to refresh]: _

```

The numbered options on the menu are the equivalent of the screen options in graphic based installations. Each option is preceded by a flag surrounded by brackets:

- [] - Option is not configured.
- [x] - Option is configured with the default setting.
The setting is displayed between parentheses under the option.
- [!] - Option is configured but needs examination in case you want to specify your own setting.

To configure an option, type the option's number. The screen displays numbered values that are available for that option. Choose the value by typing the value's number. Then type `c` to continue. Continuing either returns you to the main menu screen, or displays additional related but non-obligatory options for you to configure. Type `c` also to skip screens.

After configuring all the necessary menu options, type `b` to begin installing. At the end of the installation, the system reboots.

**Note:**

Not all configuration options in graphical installations might be available in text-based installations.

Follow all the remaining prompts. At the end of the process, log in to the system and review the license agreement at `/usr/share/oraclelinux-release/EULA`.

For additional configurations you can set on the system, see [Postinstallation Configuration](#).

Postinstallation Configuration

Configuring Login

If you selected System with GUI as the OS profile to install and configured root passwords without creating users, then at the initial login as root, you are prompted to create an initial user. Setting up the first user can be completed in two ways:

- Create a regular user account, which is the default user setup screen. Enter the required user information as prompted, then click **Next** to complete the process.
- Create a domain account. For this option, click **Enterprise Login** on the default screen. A new window opens that prompts for the domain credentials. Use this option if your environment is configured with Active Directory or Identity Management domains for storing all user information. In this manner, the user can use domain credentials to log in to the system's GNOME desktop.

Registering the System

After you install Oracle Linux on a system, you have the option to register the system with the Unbreakable Linux Network (ULN), provided that you have an account. Registering enables the system to obtain additional packages, updates, and fixes. To register the system, choose one of the following methods:

- Visit <https://linux.oracle.com>. To obtain Oracle Linux updates from ULN, you must have an Oracle Linux support subscription.
- Use the `uln_register` shell command, which opens an interactive process.
- Use the Oracle Linux GNOME desktop menu. From the menu, select **Activities** and then search for `ULN Registration`. Click the **ULN Registration** shortcut icon to start the graphical registration wizard.

For more information about installing packages and managing software on your system, see [Oracle Linux: Managing Software on Oracle Linux](#).

If you have an Oracle Linux Premier Support account, you can opt to use Ksplice, which enables you to keep your systems secure and highly available by automatically updating them with the latest kernel security errata and other critical updates. For more information, see [Oracle Linux: Ksplice User's Guide](#).

Next Steps

After a basic Oracle Linux installation, you might want to further configure the system for optimization and customization purposes, such as setting system date and time, scheduling tasks, obtaining updates, and so on. For reference, go to the Oracle Linux tutorial page at <https://docs.oracle.com/en/operating-systems/oracle-linux/8/tutorials.html> which lists tutorials for different administrative tasks. Tutorials are available for tasks that you run at the command line or through the Cockpit web console.

For additional and more detailed information about different features of Oracle Linux, go to the Oracle Linux library at [General Oracle Linux documentation](#).

4

Automating the Installation by Using Kickstart

The kickstart feature enables you to automate the OS installation. A kickstart installation uses a configuration file that instructs the installer how it should implement a specific installation. The feature offers the following benefits:

- No user intervention is required during the installation process.
- Greater facility in installing on multiple systems.
- The configuration file is useful for troubleshooting a boot-time problem with an installed system.

You can use kickstart to install Oracle Linux locally. However, the best use of this feature is in the installation of the OS on multiple systems over the network. In network installations, a kickstart operation would include the following components:

- Kickstart configuration file
- Configured network that supports network installs such as providing connectivity so that different client systems can access the necessary installation and configuration files specific to those clients.

For an example of the configuration of a network installation, see [Creating a Network Installation Setup](#).

Customizing the Kickstart Configuration File

The kickstart configuration file consists of installation instructions in the form of options and parameter definitions.

To create a kickstart configuration file, choose one of the following methods:

- Use the `/root/anaconda-ks.cfg` file of an existing Oracle Linux system.

Every Oracle Linux installation creates a file called `/root/anaconda-ks.cfg`. The file contains configuration definitions based on the system on which Oracle Linux is installed and which can be read by kickstart. Thus, the file is ready to be used either as is for other installations, or as a template for automating various types of installations for specific target systems.

As a template, the file's contents need to be revised to be applicable to other Oracle Linux installations, such as which OS versions to install on specific clients.

If the system contains the `pykickstart` package, the following useful tools can help you with revising the file's contents:

- `ksverdiff` identifies removed or deprecated options.
- `ksvalidator` checks that the syntax in the file is correct.
- Install an Oracle Linux release manually, which generates the `/root/anaconda-ks.cfg` file. Use the file to automatically install the same Oracle Linux release on other clients. You might need to adjust some of the file's configurations depending on the specifications you want to apply to those clients.

For more information, see <https://pykickstart.readthedocs.io/en/latest/>.

The configuration file is divided into parts. Each part contains a group of options as follows:

- [Installation Options](#)
- [Packages to Install](#)
- [Preinstallation Options](#)
- [Postinstallation Options](#)

Installation Options

Installation options define parameters for system storage, keyboard definitions, language setting, network information, and so on. In the following example, the option definition in bold enables PXE to bring up the network interface and is important in network installations.

Note:

To use the Btrfs file system that's supported in UEK, ensure that you're using the UEK installation media as described in [Btrfs installation chapter](#). Then you can use the `autopart` kickstart option to automatically partition disks using Btrfs, as shown in the following example under `# Partition Information`.

If you're using local mirror repositories, ensure that these are current and that the kickstart configuration includes the UEK repository required for Btrfs to function. Either use the Oracle Linux yum server to mirror the required repositories; or if you're unable to access the Oracle Linux yum server, you can mirror the repositories included on the full installation ISO, but you must use the UEK installation ISO to boot the installer. In the following example, the system is assumed to have access to the Internet and the publicly available Oracle Linux yum server repositories are used directly.

Alternatively, you also manually partition the disks to use Btrfs as described in the upstream documentation.

```
#platform=x86, AMD64, or Intel EM64T
#version=OL9
# Firewall configuration
firewall --enabled --service=ssh

# Use Oracle Linux yum server repositories as installation source
repo --name="ol9_AppStream" --baseurl="https://yum.oracle.com/repo/
OracleLinux/OL9/appstream/x86_64/"
repo --name="ol9_UEKR7" --baseurl="https://yum.oracle.com/repo/
OracleLinux/OL9/UEKR7/x86_64/"
url --url="https://yum.oracle.com/repo/OracleLinux/OL9/baseos/latest/
x86_64"

# Root password
```

```
rootpw --iscrypted SHA512_password_hash

# Use graphical install
graphical
firstboot --disable

# Keyboard layouts
keyboard --vckeymap=us --xlayouts='us'

# System language
lang en_US.UTF-8

# SELinux configuration
selinux --enforcing

# Installation logging level
logging --level=info

# System timezone
timezone America/Los_Angeles

# Network information
network --bootproto=dhcp --device=em1 --onboot=yes
--hostname=hostname

# System bootloader configuration
bootloader --location=mbr --boot-drive=sda

# Non-administrative user
user --name=user --homedir=/home/user --password=SHA512_password_hash --
iscrypted

# Partition information

clearpart --all --initlabel --drives=sda
autopart --type=btrfs
```

Packages to Install

Packages to be installed are listed under the group heading `%packages`. The list is terminated by the `%end` line.

```
%packages
@base
@core
@desktop-debugging
@dial-up
@fonts
@gnome-desktop
@guest-agents
@guest-desktop-agents
@input-methods
@internet-browser
@multimedia
```

```
@print-client
@print-server
@x11
mtools
pax
python-dmidecode
oddjob
wodim
sgpio
genisoimage
device-mapper-persistent-data
abrt-gui
samba-winbind
certmonger
openldap-clients
pam_krb5
krb5-workstation
ldapjdk
slapi-nis
libXmu
perl-DBD-SQLite
perl-Mozilla-LDAP
%end
```

The list includes package groups as well as individual packages. Names of package groups use the @ prefix, such as @base, to distinguish them from individual packages.

To help determine which packages to include in the file, use the `dnf group list` command on an existing Oracle Linux server. The command displays both the installed package groups and the package groups that are available to install.

To specify multiple packages, you can use the wildcard character (*). To exclude a package from the installation, insert the - character as a prefix to the package name.

The `%packages` keyword can take options, such as the following useful ones.

--ignoremissing

Installs the available packages without prompting about missing packages. Without this option, kickstart would interrupt the installation and prompt you to continue or cancel the installation.

--multilib

Sets the `multilib` policy in `dnf` configuration to `all` so that 32-bit packages can be installed on the system.

Instead of listing packages directly into the configuration file, you can compile these names into a file and store it in an accessible location, such as locally in a kickstart's ramdisk file system, or on an HTTP server or an NFS share. Then on the configuration file, specify the full path to the list in an `%include` statement, for example:

```
%packages --ignoremissing
%include /tmp/package-list
%end
```

Preinstallation Options

Preinstallation options define the actions that the installer must perform before beginning the installation process. Configuring this group of options is optional. The options are under the `%pre` heading and terminated by the `%end` line.

In the following example, the installer is instructed to run the script `config-partitions` that is stored on an HTTP server. Further, the installer should also download a list of packages from the web server for use with a `%include /tmp/package-list` statement in the `%packages` section.

```
%pre
%include http://192.168.1.100/scripts/config-partitions
wget -q -O- http://192.168.1.100/scripts/package-list > /tmp/package-list
%end
```

In this example, the `wget` command saves the package list in kickstart's file system, which exists as a ramdisk in memory.

An included script or file must be accessible at the specified path or URL. If no name service is available to identify hosts, then use IP addresses.

Postinstallation Options

Postinstallation options define any actions to be completed by the installer at the end of the installation. Configuring this group of options is optional.

The options are under the `%post` heading and terminated by the `%end` line.

Beginning with Oracle Linux 9, the `root` account is locked by default. If you want to enable SSH root logins to the system, add the following line to the kickstart file.

```
%post
echo "PermitRootLogin yes" > /etc/ssh/ssh_config.d/01-permitrootlogin.conf
%end
```

By default, kickstart runs postinstallation tasks in a `chroot` environment that is based on the root file system of the newly installed system. If you need to access any files that are outside the `chroot` environment, specify the `--nochroot` option to `%post` line. You can then access files in the kickstart file system with the newly installed system's root file system being mounted at `/mnt/sysimage`.

In the following example, the script `/tmp/post-config` is run at the end of the installation.

```
%post --nochroot
%include /tmp/post-config
%end
```


If you configure the installed system's network interface to obtain its settings using DHCP, you must either use IP addresses instead of domain names or set up a temporary `resolv.conf` file, for example:

```
%post
wget -q -O- http://192.168.1.100/scripts/resolv.conf > /etc/resolv.conf
%include http://instsvr.mydom.com/scripts/post-config
.
.
.
%end
```

Starting the Kickstart Installation

To start the kickstart installation, ensure that you have prepared the following:

- Access to the full installation media, which can be on a local CD-ROM drive or USB drive. See the [instructions on media preparation](#).
- Access to the kickstart configuration file.

Installing Over the Network

1. Boot the system.
2. During the boot sequence, access the system's BIOS.
3. On the BIOS screen, change the system settings to boot from the network.
4. Save the changes and continue with the boot process.
5. When the boot menu appears, select an installation option then press either **E** (UEFI-based systems) or **Tab** (BIOS-based systems) to access the boot prompt.
6. Add the the location of the kickstart file in the network. The location must specify the network server's protocol and FQDN or IP address. For example, if you are using an NFS server, then you might enter the following line:

```
nfs:options:fgdn/path-to-file
```

For additional examples for identifying the network location of the configuration file, see [Installation Source](#).

Note:

Another way of installing from the network is by using a boot server. In this scenario, the installation automatically begins when you turn on the system. See [Creating a Network Installation Setup](#).

7. Save the boot configuration revisions and continue the boot process.
At the end of the boot process, the installation proceeds immediately.

Installing Locally

1. Boot the system from the local boot media.

2. When the boot menu appears, select an installation option then press either \mathbb{E} (UEFI-based systems) or Tab (BIOS-based systems) to access the boot prompt.
3. Add the appropriate option that identifies the location of the kickstart file, for example:

```
inst.ks=cdrom:/dev/sbd1/tmp/ks.cfg
```

For more information about boot options, see [Customizing Boot Loader Actions](#).

4. Save the boot configuration revisions and continue the boot process.
At the end of the boot process, the installation proceeds immediately.

5

Installing Using the Driver Update Disk

To use the DUD during an installation, choose from one of the following methods depending on where the DUD image is stored.

- DUD image is in an attachable media (USB)
 1. Boot the system from the standard Oracle Linux installation media.
 2. While the system is booting, but before the installer starts, attach the media that contains the DUD image.

As an example, attach the media when the system displays the GRUB boot prompt.

The installer recognizes the DUD and would use it. The installer might also prompt you to select the drivers to install.

Note:

The kernel used by the installer might not include support for USB 3.0. When using the DUD on USB media, ensure that you use a USB 2.0 compatible port when connecting the USB media to the system.

- DUD image is on the network
 1. While the system is booting, press the key that enables you to edit the boot options. On BIOS-based systems, press Tab. On UEFI-based systems, press E.
 2. Add the following line to the boot options:

```
inst.dd=network-location
```

The network location can be a URL, such as `http://www.example.com/dd.iso`, or the full path of the NFS share directory.

If you are using DUD in a kickstart installation, indicate the DUD location in the kickstart file. The specific entry to add depends on the location of the DUD image.

- DUD image is in an attached block device:

```
driverdisk /dev/sdb1
```

- DUD image is on the network location:

```
driverdisk --source=network-location
```

The network location can be a URL, such as `http://www.example.com/dd.iso`, or the full path of the NFS share directory.

6

Installing a System With a Btrfs root File System

Oracle Linux uses XFS as the default file system, but it also supports other file systems such as ext2, ext4, and so on.

The Btrfs file system is supported in Oracle Linux, but only on UEK. Thus, to install Oracle Linux with Btrfs, you must use the UEK ISO image.

For more information about file systems in Oracle Linux, see [Oracle Linux 9: Managing Local File Systems](#).

Like any installation of a standard Oracle Linux release, installing the OS with a Btrfs file system can also be completed in two modes:

- [Using the GUI Installer](#)
- [Using the Text Installer](#)

You can also install a system with a Btrfs root file system using kickstart automation as described in [Automating the Installation by Using Kickstart](#). If you choose to perform an automated installation and you create your own network installation setup, as described in [Creating a Network Installation Setup](#), make sure that you configure your network boot to use the installation kernel and ram-disk image provided on the UEK ISO image and that you have mirrored all of the required repositories, or you may get an error stating that Btrfs is not supported for your partitioning type.

Using the GUI Installer

The following steps are limited to what is relevant to a Btrfs installation. For the complete procedure, refer to [Installing in Graphics Mode](#).

1. Boot the system from the installation media using the Unbreakable Enterprise Kernel boot ISO file.
2. On the Installation Summary screen, select **Installation Destination**, then select the local disks and add any **Specialized & Network** disks that you want to use for the installation.
3. Under Storage Configuration, select **Custom** and then click **Done**.

The Manual Partitioning screen is displayed. Custom partitioning can be either automatic, where mount points are automatically created, or manual, where you individually create and configure mount points.

Automatic Partitioning

- a. From the drop-down list of partitioning schemes, select **Btrfs**.
- b. Click **Click here to create them automatically**.

The following mount points are created:

- /home

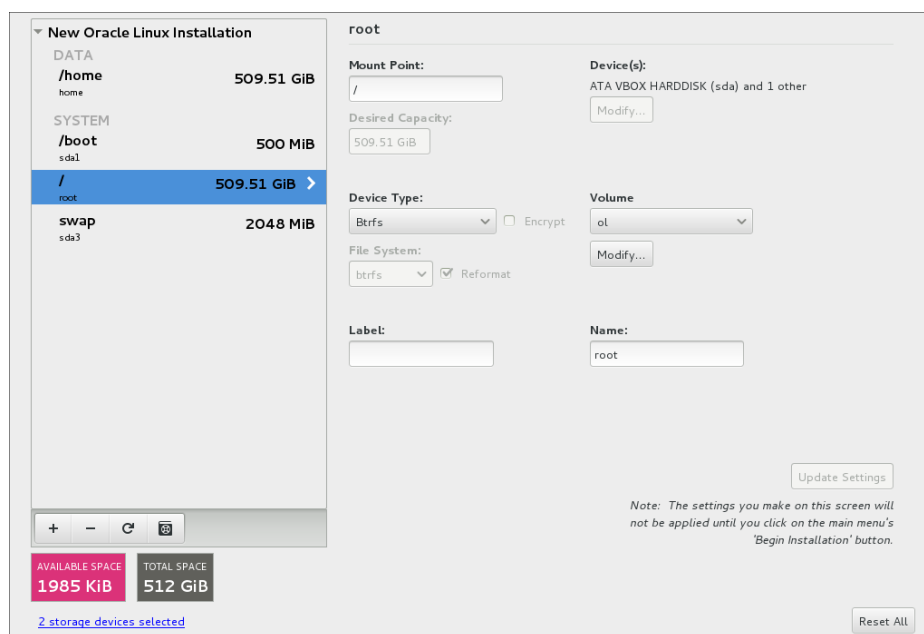
- /boot
 - / (root)
 - swap
- c. Optionally, select each mount point and verify in the corresponding information that the mount point's device type is **Btrfs**.

 **Note:**

The `swap` partition cannot be configured as a Btrfs file system.

The mount point information should resemble the following example:

Figure 6-1 Mount Points



- d. Optionally for each mount point, configure the respective volumes.
- e. Click **Done**, then on the partition summary that is displayed, click **Accept Changes**.

Manual Partitioning

- a. From the drop-down list of partitioning schemes, select **Btrfs**.
- b. Click + to create the mount point.
The **Add A New Mount Point** window opens.
- c. Select the mount point and enter its size as prompted, then click **Add mount point**.
As a minimum, create mount points for `/`, `/boot`, `/home`, and `swap`.
- d. Select each mount point you created and verify that their corresponding information is correct.

 **Note:**

The `swap` partition cannot be configured as a Btrfs file system.

- e. Optionally, modify the volume of each mount point, as needed.
- f. Click **Done** and on the partition summary that is displayed, click **Accept Changes**.
4. Complete the configuration of any additional sections in the Installation Summary screen.
5. Click **Begin Installation**.

Using the Text Installer

The following steps are limited to what is relevant to a Btrfs installation. For the complete procedure, refer to [Installing in Text Mode](#).

1. From the text based installation menu, type `5` to configure Installation Destination.
 - a. **Installation Destination:** If your preferred disk is already selected, type `c` to continue. Otherwise, type the number corresponding to the disk you want to use, press Enter, then type `c` to continue.
 - b. **Partitioning Options:** If your preferred option is already selected, type `c` to continue. Otherwise, type the number corresponding to your preferred option, press Enter, then type `c` to continue.
 - c. **Partition Scheme Options:** Type `2` to use Btrfs, press Enter, then type `c` to return to the main menu.
2. Configure all the remaining settings that are flagged with the warning symbol (!).
3. After you have completed configuring all the installation options, type `b` to begin the installation.

Verifying the Btrfs Installation

When the entire installation process has completed and you can log in to the system, you can use different tools to verify the file system configuration. The following example shows the output of the `sudo df -Thk` command:

```
sudo df -Thk

Filesystem Type 1K-blocks    Used Available Use% Mounted on
...
/dev/sda2 btrfs 15097856 4803120 10052400 33% /
/dev/sda2 btrfs 15097856 4803120 10052400 33$ /home
/dev/sda2 btrfs 15097856 4803120 10052400 33$ /boot
...
```

Optionally, you can complete other tasks as described in [Postinstallation Configuration](#).

7

Troubleshooting Oracle Linux Installations

This chapter describes some options for troubleshooting Oracle Linux installations. Also check the Oracle Linux release notes of the specific Oracle Linux release you are installing in [Oracle Linux](#) for any known issues.

For known hardware issues, see the release notes for your UEK release in the [Unbreakable Enterprise Kernel documentation](#).

Troubleshooting Menu Options

The boot menu on the Oracle Linux installation media contains a Troubleshooting option with the following options:

Install the Oracle Linux release in basic graphics mode

Use this option if the screen goes blank or appears distorted when you try to install Oracle Linux in graphics mode.

Rescue a Oracle Linux system

Use this option to boot an installed system in a mode that enables you to edit partitions or configuration files to fix a variety of boot problems.

Run a memory test

Use this option to run the `memtest86` utility, which verifies the integrity of the system's RAM.

Boot from local drive

Use this option to boot an installed system from the hard disk.

Debug and Log Information

During an installation, you can press `Ctrl+Alt+F1` to display the virtual console for the installation program. This console contains messages and debugging information output for the installation program. Additional virtual consoles are available to display log information from different sources, as described in the following section. Press `Ctrl+b`, and then press either `n` (for next) or `b` (for back) to switch between the virtual consoles. To return to the graphical installation program, press `Ctrl+Alt+F6`.

During an installation, several log files are generated, which capture messages from the following sources:

Anaconda program

The `/tmp/anaconda.log` file contains Anaconda logs relating to the installation. During the installation you can access the messages stored in this log by pressing `Ctrl+Alt+F1` to display the virtual console, then press `Ctrl+b`, and then press `3`. If the installation succeeds, the log file is copied to `/var/log/anaconda/anaconda.log`.

Hardware detection and configuration

The `/tmp/syslog` file contains messages relating to the system hardware.

If the installation succeeds, the log file is copied to `/var/log/anaconda/syslog`.

Kickstart

The `/tmp/ks-name.log` file contains logs from kickstart installations.

If the installation succeeds, the log file is copied to `/var/log/anaconda/ks-script-name.log`.

Network detection and configuration

The `/tmp/ifcfg.log` file contains logs relating to network configuration.

If the installation succeeds, the log file is copied to `/var/log/anaconda/ifcfg.log`.

Other programs

The `/tmp/program.log` file contains logs from all other programs used during the installation.

During the installation you can view the messages stored in this log by pressing `Ctrl+Alt+F1` to display the virtual console, then press `Ctrl+b`, and then press `5`.

If the installation succeeds, the log file is copied to `/var/log/anaconda/program.log`.

Package installation

The `/tmp/packaging.log` file contains package installation messages output by the `dnf` and `rpm` commands.

If the installation succeeds, the log file is copied to `/var/log/anaconda/packaging.log`.

Storage detection and configuration

The `/tmp/storage.log` file contains logs from the storage modules.

During the installation you can view the messages stored in this log by pressing `Ctrl+Alt+F1` to display the virtual console, then press `Ctrl+b`, and then press `4`.

If the installation succeeds, the log file is copied to `/var/log/anaconda/storage.log`.

If the installation fails, the messages from these log files are combined into a single log file at `/tmp/anaconda-tb-name`.

To access a shell prompt as the `root` user during the installation, press `Ctrl+Alt+F1` to display the virtual console. Then, press `Ctrl+b`, followed by pressing the number `2`. You can use the shell prompt to access the log files and to copy them to a local storage device such as a USB device. Or, you can copy the log files to a network location by using the `scp` command.

A

Creating a Network Installation Setup

Network installations are useful and advantageous especially in scenarios where you have to install the OS on multiple systems.

If you use a boot ISO or the Preboot eXecution Environment (PXE) to install the OS on systems, you can set up a network installation configuration that consists of the following components:

- A network installation server that provides the IP and network configuration, the PXE configuration files, kernel and boot images, and kernel boot directives. In this example, `dnsmasq` is used to provide these services.
- A network accessible file system server over a protocol such as NFS or HTTP, where packages, the kickstart file, and other required configuration files might be stored. In this document, NFS is used as the file system. Note that if your systems have direct access to the Internet, you can configure your kickstart installation to use the BaseOS and AppStream repositories directly available on the Oracle Linux yum server for the package installation. You can also create a yum mirror of these repositories as described in [Oracle Linux: Managing Software on Oracle Linux](#) so that systems that do not have direct access to the Internet have access to a complete set of packages. However, you would still need to provide a mechanism to serve the kickstart file up to the system when it boots.

The two components can be on separate systems. Also, they are not required to run the latest Oracle Linux version. A previous release would suffice. For convenience, the scenario in this example assumes that the two components are hosted in one system. It also assumes that kickstart installation is used.

Setting Up the Network Server

Preparing a server for a network installation consists of the following general tasks:

- [How to Configure NFS](#)
- [How to Configure dnsmasq](#)

How to Configure NFS

If you have an existing NFS server, you can use this service to host the extracted contents of an ISO and your kickstart configuration files. Ensure that the exported share directories are accessible to the IP ranges specified for your PXE boot hosts.

If you do not have an available NFS server, use the following procedure to install and configure the service to enable network installation.

 **Note:**

These NFS configuration steps are limited only to what is relevant to a network installation. For more information in configuring NFS, refer to the appropriate Oracle Linux System Administration guide.

1. Install the `nfs-utils` package:

```
sudo dnf install nfs-utils -y
```

2. If you are running a firewall service, enter the following firewall command.

```
sudo firewall-cmd --add-service nfs --permanent
```

If you are using an alternate firewall service or you have an external firewall device, ensure that you configure rules that grant PXE boot hosts access to the NFS service on this system.

3. Create a directory to store the ISO image, for example.

```
sudo mkdir /var/nfs-exports/ISOs
```

4. Export the NFS share directory.

```
sudo exportfs -i -o ro [subnet]:/var/nfs-exports/ISOs
```

This syntax grants world access to the NFS share with read only permissions. Adding *subnet*, for example `192.0.2.0/24:/var/nfs-exports/ISOs`, limits access only to the subnet's clients.

Alternatively, you can add an entry for exporting the share directory in the `/etc/exports` file, for example:

```
/var/nfs-exports/ISOs 192.0.2.0/24(ro)
```

Then, reload the `/etc/exports` to implement the entry:

```
sudo exportfs -ra
```

For more information, see the `exportfs(8)`, `exports(5)`, and `showmount(8)` manual pages.

5. Start the `nfs-server` service and configure it to automatically start after every system reboot:

```
sudo systemctl start nfs-server
sudo systemctl enable nfs-server
```

6. Extract the downloaded ISO image to a subdirectory of the NFS share directory:

```
sudo cp -a T path-to-download-image /var/nfs-exports/ISOs/ol9
```

7. If using kickstart, put the kickstart files in a subdirectory of the NFS share directory also, such as `/var/nfs-exports/ISOs/ksfiles`.
8. (Optional) From a different system, verify that the NFS share directory is accessible, for example:

```
sudo mount -t nfs NFS-server-ip:/var/nfs-exports/ISOs /mnt
```

How to Configure dnsmasq

The dnsmasq router advertisement server is designed to act as a DNS forwarder, DHCP server, and TFTP server. Dnsmasq is applicable in most network installation scenarios and is therefore a convenient alternative to configuring separate DHCP and TFTP services.

For more information about dnsmasq, see the `dnsmasq(8)` manual page, the `/usr/share/doc/dnsmasq-version` file, and <https://thekelleys.org.uk/dnsmasq/doc.html>.

1. Install the dnsmasq package.

```
sudo dnf install dnsmasq -y
```

2. Configure parameters in the `/etc/dnsmasq.conf` file.

- At a minimum, you must have the **enable-tftp** entry and a defined TFTP server directory for **tftp-root**. See the entries in bold in the following example:

```
interface=em1
dhcp-range=10.0.0.101,10.0.0.200,6h
dhcp-host=80:00:27:c6:a1:16,10.0.0.253,svr1,infinite
dhcp-boot=pxelinux/pxelinux.0
dhcp-match=set:efi-x86_64,option:client-arch,8
dhcp-boot=tag:efi-x86_64,shim.efi
enable-tftp
tftp-root=/var/lib/tftpboot
```

 **Note:**

If SELinux is enabled in enforcing mode on your system and you configured a TFTP server directory other than `/var/lib/tftpboot`, you must also install the `policycoreutils-python` and `policycoreutils` packages to enable you to run the following commands:

```
sudo /usr/sbin/semanage fcontext -a -t tftpdirt_t "/var/
tftpboot(/.*)?"
sudo /sbin/restorecon -R -v /var/tftpboot
```

These commands define the default file type of the TFTP server directory hierarchy as `tftpdirt_t` and apply the file type to the entire directory hierarchy.

The following list describes the other parameters in the `/etc/dnsmasq.conf` file:

interface

Specifies the interface to be monitored for incoming client requests.

dhcp-range

Identifies a range of available IP addresses. The `6h` setting in the example above specifies a six-hour lease of the addresses.

To configure static addresses with infinite leases, instead of a pool, specify a static network address and use the `static` and `infinite` keywords, for example:

```
dhcp-range=10.0.0.253,static,infinite
```

dhcp-host

Specifies a reserved IP address for a client system. The system is identified by its name and MAC address.

dhcp-boot

Specifies the location of the boot loader file for clients, such as `pxelinux/pxelinux.0` for BIOS-based clients. For UEFI-based clients, include the `tag:efi-x86_64` keyword in the setting before specifying the boot loader, for example:

```
dhcp-boot=tag:efi-x86_64,shim.efi
```

You must create separate entries for BIOS-based and UEFI-based clients.

- Uncomment the `tftp-no-blocksize` line in the file as shown:

```
# This option stops dnsmasq from negotiating a larger blocksize for TFTP
# transfers. It will slow things down, but may rescue some broken TFTP
# clients.
tftp-no-blocksize
```

3. (Optional) To use dnsmasq as a caching-only name server, do the following:
 - a. In the `/etc/resolv.conf` file, configure a name server entry for 127.0.0.1 that precedes other name server entries, for example:

```
nameserver 127.0.0.1
nameserver 10.0.0.8
nameserver 10.0.0.4
```

The dnsmasq server ignores the 127.0.0.1 entry and forwards DNS queries to the other listed name servers.

- b. Configure the firewall to accept DNS requests:

```
sudo firewall-cmd --add-service=dns --permanent
```

4. Start the dnsmasq service and configure it to automatically start after every system reboot.

```
sudo systemctl start dnsmasq
sudo systemctl enable dnsmasq
```

Configuring PXE Boot Loading

The steps to configure a PXE boot loader depends on the type of system that is used for the boot server.

- [How to Configure for BIOS-based PXE Clients](#)
- [How to Configure for UEFI-based PXE Clients](#)

Important:

You must use the correct kernel and ram-disk image for the installation that you intend to support. For example, if you intend to install a system with a Btrfs root partition, you must use the UEK boot ISO to obtain the correct kernel and image for the system to be able to install with Btrfs support.

How to Configure for BIOS-based PXE Clients

1. Install the `syslinux` package that contains the `pxelinux.0` boot loader.

```
sudo dnf install syslinux -y
```

2. Create the `pxelinux/pxelinux.cfg` directory under the TFTP server directory.

```
sudo mkdir -p /var/lib/tftpboot/pxelinux/pxelinux.cfg
```

3. Copy the boot loader file to the `pxelinux` subdirectory.

```
sudo cp /usr/share/syslinux/pxelinux.0 /var/lib/tftpboot/pxelinux
```

4. From the NFS share directory, copy the installation kernel (`vmlinuz`), and the ram-disk image file (`initrd.img`) to the `pxelinux` subdirectory.

```
sudo cp /var/nfs-exports/ISOs/ol9/vmlinuz /var/lib/tftpboot/  
pxelinux/vmlinuz  
sudo cp /var/nfs-exports/ISOs/ol9/initrd.img /var/lib/tftpboot/  
pxelinux/initrd.img
```

5. Copy the modules for displaying the boot loader as a menu to the boot loader directory.

```
sudo cp /usr/share/syslinux/*.c32 /var/lib/tftpboot/pxelinux
```

6. In the `pxelinux.cfg` subdirectory, create and then add entries to a boot configuration file.

You can assign any name to the file, such as `pxe.conf`. The following example shows typical entries in the file:

```
DEFAULT menu.c32  
TIMEOUT 400  
ONTIMEOUT BootLocal  
PROMPT 0  
MENU INCLUDE pxelinux.cfg/pxe.conf  
NOESCAPE 1  
LABEL BootLocal  
    localboot 0  
    TEXT HELP  
    Boot to local hard disk  
    ENDTXT  
  
LABEL OL9  
    MENU LABEL OL9  
    kernel vmlinuz  
    append initrd=initrd.img inst.repo=/var/nfs-exports/ISOs/ol9/ \  
    inst.ks.sendmac inst.ks=/var/nfs-exports/ISOs/ksfiles/kstart-  
file  
    TEXT HELP  
    Install Oracle Linux 9  
    ENDTXT
```

Based on these entries, the boot loader would automatically attempt to boot from the local drive if no user intervention occurs during the `TIMEOUT` period. If no OS is installed, then the boot loader would boot from the network and initiate the installation process.

This list explains some of the directives used in the configuration file:

- `DEFAULT` identifies the module you want to use for displaying the boot loader menu.

For a basic text display, the `menu.c32` module suffices. However, if you add directives for a graphical display, such as images and colors, then specify the `vesamenu.c32` module instead.

- `TIMEOUT` specifies the period in `timeout/10` seconds before the boot loader boots the client according to the subsequent directives. The next directive (`ontimeout`) specifies the action when the wait period expires.
- `PROMPT` determines whether or not the `boot: prompt` is displayed by default. If `PROMPT` is set to 1, the `boot: prompt` is displayed. If `PROMPT` is set to 0, the `boot: prompt` is not displayed unless the user presses the `Shift` or `Alt` key at the console.
- `MENU INCLUDE` identifies the boot configuration file you created.
- `kernel` defines the name of the kernel executable.
- `append` defines any parameters that should be appended when loading the kernel, such as the name of the ram-disk image and the location of a file. Note that the `inst.repo` variable can be set to point to the BaseOS repository on the Oracle Linux yum server if the system has access to the Internet. For example, `inst.repo` can be set to `https://yum.oracle.com/repo/OracleLinux/OL9/baseos/latest/x86_64` for an `x86_64` platform system.

! Important:

The kernel and ram-disk image file paths are assumed to be relative to the subdirectory that contains the boot loader, such as `pxelinux`. If you place the `vmlinuz` and `initrd.img` files in a subdirectory such as `pxelinux/OL9`, ensure you have the correct relative paths.

How to Configure for UEFI-based PXE Clients

1. Install the following packages:

- `grub2-efi` that contains the `grubx64.efi` boot loader
- `shim` that contains the first-stage boot loader `shim.efi`

This package is required to support Secure Boot on UEFI clients, but should be used for all UEFI clients as a passthrough boot loader.

2. Go to the package location and extract their contents.

Run the following commands:

```
sudo cd /tempdir
sudo rpm2cpio grub2-efi-version.rpm | cpio -idmv
sudo rpm2cpio shim-version.rpm | cpio -idmv
```

3. Create a subdirectory, for example, `efi`, under the TFTP server directory..

```
sudo mkdir -p /var/lib/tftpboot/efi
```

4. Copy the boot loader and other related files to the `efi` subdirectory.

Run the following commands:

```
sudo cp /tmpdir/boot/efi/EFI/redhat/grubx64.efi /var/lib/
tftpboot/efi
sudo cp /tmpdir/boot/efi/EFI/redhat/shim.efi /var/lib/tftpboot/efi
sudo cp /tmpdir/boot/efi/EFI/redhat/MokManager.efi /var/lib/
tftpboot/efi
```

 **Note:**

The `shim.efi` and `MokManager.efi` files are needed if you want to support Secure Boot on clients. The `MokManager.efi` provides utilities for managing the keys used to sign EFI binaries. As a passthrough boot loader, you would then specify `shim.efi` when setting `dhcp-boot` for UEFI-based clients in the `/etc/dnsmasq.conf` file. See [How to Configure dnsmasq](#).

5. From the NFS share directory, copy the installation kernel (`vmlinuz`) and the ram-disk image file (`initrd.img`) to the `efi` subdirectory.

```
sudo cp /var/nfs-exports/ISOs/ol9/vmlinuz -O /var/lib/tftpboot/efi/
vmlinuz
sudo cp /var/nfs-exports/ISOs/ol9/initrd.img /var/lib/tftpboot/efi/
initrd.img
```

6. In the `efi` subdirectory, create and then add entries to the `grub.cfg` boot loader configuration file.

The configuration file should have the options for booting from the network as well as for booting locally. The client boots from the network to begin the installation process. Then once the client reboots at the end of the installation, the system boots from the local drive.

The following example shows typical entries in the file:

```
set default 0
set timeout=10

menuentry 'ol9 localboot' {
  echo "Booting from local disk"
  set root=(hd0,gpt1)
  chainloader efi/shim.efi
  boot
}

menuentry 'ol9' {
  echo "Loading vmlinuz"
  linuxefi vmlinuz inst.repo=/var/nfs-exports/ISOs/ol9/
  inst.ks.sendmac \
  inst.ks=/var/nfs-exports/ISOs/ksfiles/kstart-file
  echo "Loading /initrd.img"
  initrdefi initrd.img
```



```
echo "Booting installation kernel"  
}
```

⚠ Caution:

Boot loader configuration is not uniform across UEFI-based systems. Because of variables such as differing disk layout, a specific boot loader setup will not universally apply to all systems. The previous example illustrates only one way of configuring the boot loader. You must create boot loader configurations appropriate to your own systems.

- `linuxefi` defines the name of the kernel executable and any parameter that should be appended when loading the kernel, such as the location of the installation packages and the location of a file.
- `initrdefi` defines the name of the ram-disk image.

! Important:

The kernel and ram-disk image file paths are assumed to be relative to the subdirectory that contains the boot loader, such as `efi`. If you place the `vmlinuz` and `initrd.img` files in a subdirectory, such as `efi/OL9`, ensure you have the correct relative paths.

Supporting Different Kinds of Clients

To support different types of clients, you can create a configuration file with the name based on the following:

- Client's UUID, for example, `a8943708-c6f6-51b9-611e-74e6ac80b93d`

✎ Note:

A UUID-based file name is valid only for BIOS-based clients. Do not apply it for UEFI-based clients.

- Client's MAC address prefixed by `01-`, which represents the ARP hardware type for Ethernet, and using dashes to separate each byte value instead of colons, for example, `01-80-00-27-c6-a1-16`.

Use lowercase characters when typing the MAC address.

- Client's IP address expressed in hexadecimal without any leading `0x`, for example, `0A0000FD` represents the IP address `10.0.0.253`.

To reduce the number of configuration files, you can group clients by IP address range, for example, `0A0000E` represents the IP address range `10.0.0.224` through `10.0.0.239`.

If you are serving both types of clients, ensure that the file names are distinct from each other. Where the configuration file for BIOS-based clients is `A000FC`, for example, the equivalent file for UEFI-based clients can be `grub.cfg-A000FC`, and so on.

Place the configuration files in their respective boot loader subdirectories such as `pxelinux/pxelinux.cfg` for BIOS-based clients or `efi` subdirectory for UEFI-based clients.

The boot loader searches for a configuration file in the following order, until a matching file name is found:

- `UUID` (for example, `a8943708-c6f6-51b9-611e-74e6ac80b93d`)
- `01-MAC_address` (for example, `[grub2-cfg-]01-80-00-27-c6-a1-16`)
- Full 32 bits of the IP address (for example, `[grub.cfg-]0A000FD`)
- Most significant 28 bits of the IP address (for example, `[grub.cfg-]0A0000F`)
- Most significant 24 bits of the IP address (for example, `[grub.cfg-]0A0000`)
- Most significant 20 bits of the IP address (for example, `[grub.cfg-]0A000`)
- Most significant 16 bits of the IP address (for example, `[grub.cfg-]0A00`)
- Most significant 12 bits of the IP address (for example, `[grub.cfg-]0A0`)
- Most significant 8 bits of the IP address (for example, `[grub.cfg-]0A`)
- Most significant 4 bits of the IP address (for example, `[grub.cfg-]0`)
- Default configuration file (either `default` for BIOS-based clients or `grub.cfg` for UEFI-based clients).

If several configuration files for a client type have identical content, you can use the `ln` command to link the files to a primary copy, for example:

```
sudo ln primary-ol-verson [grub.cfg-]0A0000FC
```

```
sudo ln primary-ol-verson [grub.cfg-]0A0000FD
```

```
sudo ln primary-ol-verson [grub.cfg-]0A0000FE
```

For more information about `pxelinux`, see <https://wiki.syslinux.org/wiki/index.php?title=PXELINUX>.

For more information about GRUB 2, run the `info grub` command to access the GRUB 2 manual.