NVIDIA DGX Software for Red Hat Enterprise Linux 9

NVIDIA Corporation

Feb 20, 2025

Contents

Chapter 1. Introduction

This document explains the installation and configuration of the NVIDIA DGX Software Stack on DGX systems installed with Red Hat Enterprise Linux.

NVIDIA provides the NVIDIA DGX Software Stack targeted for installation on DGX systems that have been user-installed with Red Hat Enterprise Linux. The software stack provides the same features and functionality that are provided by the original DGX OS built on the Ubuntu.

> **Note**
>
> NVIDIA acknowledges the wide use of Rocky Linux and understands that it is a community-developed derivative of the NVIDIA supported Red Hat Enterprise Linux. Support for Rocky Linux is available directly from the Rocky Linux community. NVIDIA ensures that NVIDIA provided software runs on tested Rocky Linux versions and will try to identify and correct issues related to NVIDIA provided software.
>
> While it might be possible to use other derived Linux distributions, not all have been tested and qualified by NVIDIA. Refer to the *Release Notes* for the list of tested and qualified software and Linux distributions.

1.1. Related Documentation

- NVIDIA DGX-1 User Guide
- NVIDIA DGX-2 User Guide
- NVIDIA DGX Station User Guide
- NVIDIA DGX A100 User Guide
- NVIDIA DGX Station A100 User Guide
- NVIDIA DGX H100/H200 User Guide

1.2. Before You Begin

The following are required (or recommended wherever indicated).

1.2.1. Red Hat Subscription

You need a Red Hat subscription if you plan to install and use Red Hat Enterprise Linux on the DGX system. A subscription also lets you obtain update packages and additional packages for Red Hat Enterprise Linux. You can either purchase a subscription or obtain a free evaluation subscription from the Red Hat Software & Download Center.

1.2.2. Access to Repositories

The repositories can be accessed from the internet.

If you are using a proxy server, follow the instructions in the section *Configuring a System Proxy* to make sure the system can access the necessary URIs.

- **NVIDIA Repositories**

  After installing Red Hat Enterprise Linux on the DGX system, you must enable the NVIDIA DGX software repository (https://repo.download.nvidia.com). The repository includes the NVIDIA drivers and software for supporting DGX systems.

  See the section *Enabling the DGX Software Repository* for instructions on how to enable the repository.

- **Red Hat Repositories**

  Installation of the DGX Software over Red Hat Enterprise Linux 9 requires access to several additional repositories.

  - Red Hat Enterprise BaseOS Repository: `rhel-9-for-x86_64-baseos-rpms`
  - Red Hat Enterprise AppStream Repository: `rhel-9-for-x86_64-appstream-rpms`
  - Red Hat Enterprise CodeReady Linux Builder Repository: `codeready-builder-for-rhel-9-x86_64-debug-rpms`

1.2.3. Network File System

On DGX servers, the data drives are meant to be used as a cache. DGX Station users can follow the same usage, or can alternatively opt to use these drives for storage. When using the data drives as cache, a network file system (NFS) is recommended to take advantage of the cache file system provided by the DGX software stack.

1.2.4. BMC Password

The NVIDIA DGX server includes a base management controller (BMC) for out-of-band management of the DGX system. NVIDIA recommends disabling the default username and creating a unique username and password as soon as possible.

Chapter 2. Release Notes

This section provides detailed information for releases and upgrades available for the NVIDIA DGX™ Software Stack for Red Hat Enterprise Linux 9 and Rocky Linux 9.

2.1. Current Software Versions

The following table shows the current version information of the software packages provided in the NVIDIA repositories for the NVIDIA DGX Software Stack.

Table 1: Current Software Versions (Last Updated on February 20, 2025)

Component	Version	Additional Information
GPU Driver	550.144.03	EL8: RPM installer EL9: RPM installer
GPU Driver	535.230.02	EL8: RPM installer EL9: RPM installer
CUDA Toolkit	12.4 Update 1	R550: 12.4 Update 1 download
CUDA Toolkit	12.2 Update 2	R535: 12.2 Update 2 download
DOCA OFED	2.9.1-2.0.0	2.9.1-2.0.0 download
Inbox OFED	39.0-1	For DGX OS 6 only.
NCCL	2.25.1	
cuDNN	9.7.1	
DCGM	4.1.1	
GPUDirect Storage (GDS)	▶ 1.13 for CUDA Toolkit 12.8 ▶ 1.11 for CUDA Toolkit 12.6 ▶ 1.10 for CUDA Toolkit 12.5 ▶ 1.9 for CUDA Toolkit 12.4 ▶ 1.8 for CUDA Toolkit 12.3 ▶ 1.7 for CUDA Toolkit 12.2	
NVIDIA Container Toolkit	1.17.4	NVIDIA Container Toolkit includes the following packages: ▶ nvidia-container-toolkit: 1.17.4 ▶ libnvidia-container-tools: 1.17.4 ▶ libnvidia-container1: 1.17.4
nvidia-peer-memory	1.3	

Note

- CUDA Toolkit is installed by default only for DGX stations and is optional for DGX servers. Refer to the CUDA Release Notes for driver compatibility information.
- For CUDA Toolkit minor version compatibility and the minimum required driver version, refer to CUDA Compatibility.

The following table provides information about the matching firmware versions for the NVIDIA DOCA™ Host package with the doca-ofed installation profile v2.9.1.

Table 2: Matching Firmware Versions (Last Updated on February 20, 2025)

DGX-1, DGX-2 ConnectX-4 (CX-4) or ConnectX-5 (CX-5)	DGX A100 ConnectX-6	DGX A100 ConnectX-7	DGX H100/H200 ConnectX-7
CX-5: 16.35.4030 CX-4: 12.28.2006	20.43.2026	28.43.2026	28.43.2026

For installation instructions, refer to

- NVIDIA DOCA-OFED: *Installing NVIDIA DOCA-OFED*
- NVIDIA MLNX_OFED: *Installing NVIDIA MLNX_OFED*
- ConnectX®-7 adapter cards: *Installing ConnectX-7 Firmware*
- ConnectX®-6 adapter cards: Firmware Downloads

Note

For information about LTS software versions for related networking components, refer to the Networking Long-Term Support Releases page.

2.2. Latest Release

Important

Installing or updating to the DGX Software also updates the installed Red Hat Enterprise Linux 9 distribution to the latest version.

If you use NVIDIA MLNX_OFED, before installing or updating to EL9-24.12, be sure that there is a MLNX_OFED package version available that supports the latest Red Hat Enterprise Linux 9 version.

- To check the latest Red Hat Enterprise Linux 9 version, refer to Red Hat Knowledgebase article 3078.
- To check the MLNX_OFED package OS support, visit Mellanox and click the latest NVIDIA MLNX_OFED software version. Use the side menu to navigate to **Release Notes** > **General Support** and view Supported Operating Systems.

2.2.1. Release EL9-24.12

Release Date: December 18, 2024

2.2.1.1 Release Highlights

- Adds support for Red Hat Enterprise Linux 9.5 and Rocky 9.5.
- Introduces support for the NVIDIA DGX™ H200 system.
- No support for the NVIDIA DGX™ H800 system.
- Updates GPU drivers:
  - Release 550.127.08 with CUDA Toolkit 12.4 Update 1
  - Release 535.216.03 with CUDA Toolkit 12.2 Update 2
- Adds support for the NVIDIA® BlueField®-3 DPU in NIC mode on DGX H100 and H200 systems.
  - NVIDIA BlueField-3 DPU NIC Firmware Release Notes v32.43.2026 LTS
  - NVIDIA BlueField-3 firmware download
- Includes support for the NVIDIA DOCA™ Host package with the doca-ofed installation profile v2.9.1 (providing the MLNX_OFED functionality).
  - NVIDIA DOCA 2.9.1 download
  - MLNX_OFED release transition
- Updates the DGX Software Stack.

2.2.1.2 Qualified Software Stack

The following table shows the current version information of the software packages provided in the NVIDIA repositories for the NVIDIA DGX Software Stack.

Component	Latest versions in the repositories
DGX Base OS	**EL9-24.12**
OS	Red Hat Enterprise Linux 9.5 and Rocky Linux 9.5
Kernel	5.14.0-503.15.1.el9_5.x86_64
GPU Driver	▶ 550.127.08 ▶ 535.216.03
CUDA Toolkit	▶ 12.4 Update 1 for GPU driver 550.127.08 ▶ 12.2 Update 2 for GPU driver 535.216.03
NCCL	2.23.4
cuDNN	9.5.1
DCGM	3.3.8
GPU Direct Storage	▶ 1.9 for CUDA 12.4 ▶ 1.7 for CUDA 12.2
NVIDIA System Management (NVSM)	24.06.05
Docker CE	27.3.1
NVIDIA Container Runtime	▶ nvidia-container-toolkit: 1.16.2 ▶ libnvidia-container-tools: 1.16.2 ▶ libnvidia-container1: 1.16.2
MIG Configuration Tool	0.10.0
GDRCopy	2.4.3
DLFW (Deep Learning Frameworks)	24.10

The following table provides information about the supported OS and matching firmware versions for NVIDIA DOCA™ Host package with the doca-ofed installation profile v2.9.1 and the NVIDIA® OpenFabrics Enterprise Distribution for Linux (MLNX_OFED) v24.10-1.1.4.0.

OS	DGX-1, DGX-2 ConnectX-4 (CX-4) or ConnectX-5 (CX-5)	DGX A100 ConnectX-6	DGX A100 ConnectX-7	DGX H100/H200 ConnectX-7
RHEL 9	CX-5: 16.35.4030 CX-4: 12.28.2006	20.43.2026	28.43.2026	28.43.2026

2.2.1.3 Supported DGX Systems

The EL9-24.12 release supports the following DGX systems:

- DGX H200 1,128 GB
- DGX H100 640 GB
- DGX A100 640 GB
- DGX A100 320 GB
- DGX A800 640 GB
- DGX-2
- DGX-1 32 GB
- DGX Station A100 320 GB
- DGX Station A100 160 GB
- DGX Station A800 320 GB
- DGX Station 32 GB

2.3. Previous Releases

2.3.1. Release EL9-24.06

Release Date: July 11, 2024

2.3.1.1 Release Highlights

- Added support for Red Hat Enterprise Linux 9.4 and Rocky 9.4.
- Introduced support for the NVIDIA DOCA™ Host package with the doca-ofed installation profile v2.7.0.
  - DOCA OFED download
- Included support for the NVIDIA® OpenFabrics Enterprise Distribution for Linux (MLNX_OFED) v24.04-0.6.6.0
  - MLNX_OFED download
- Continued support for single-port ConnectX-7 VPI adapter card for DGX A100 System.
- Updated the DGX Software Stack.

2.3.1.2 Qualified Software Stack

The following table shows the current version information of the software packages provided in the NVIDIA repositories for the NVIDIA DGX Software Stack.

Component	Latest versions in the repositories
DGX Base OS	**EL9-24.06**
OS	Red Hat Enterprise Linux 9.4 and Rocky Linux 9.4
Kernel	5.14.0-427.18.1.el9_4.x86_64
GPU Driver	▶ 550.90.07 ▶ 535.183.01
CUDA Toolkit	▶ 12.4 Update 1 for GPU driver 550.90.07 ▶ 12.2 Update 2 for GPU driver 535.183.01
NCCL	2.21.5
cuDNN	9.1.1
DCGM	3.3.6
GPU Direct Storage	▶ 1.9.1 for CUDA 12.4 ▶ 1.7.2 for CUDA 12.2
NVIDIA System Management (NVSM)	24.03.03
Docker CE	26.1.3
NVIDIA Container Runtime	▶ nvidia-container-toolkit: 1.15.0 ▶ libnvidia-container-tools: 1.15.0 ▶ libnvidia-container1: 1.15.0
MIG Configuration Tool	0.7.0
GDRCopy	2.4.1
DLFW (Deep Learning Frameworks)	24.05

The following table provides information about the supported OS and matching firmware versions for the NVIDIA® OpenFabrics Enterprise Distribution for Linux (MLNX_OFED) v24.04-0.6.6.0 and the NVIDIA DOCA™ Host package with the doca-ofed installation profile v2.7.0.

OS	DGX-1, DGX-2 ConnectX-4 (CX-4) or ConnectX-5 (CX-5)	DGX A100 ConnectX-6	DGX A100 ConnectX-7	DGX H100 ConnectX-7
RHEL 9	CX-5: 16.35.3502 CX-4: 12.28.2006	20.41.1000	28.41.1000	28.41.1000

2.3.1.3 Supported DGX Systems

The EL9-24.06 release supports the following DGX systems:

- DGX H100
- DGX A100 640 GB
- DGX A100 320 GB
- DGX A800 640 GB
- DGX-2
- DGX-1 32 GB
- DGX Station A100 320 GB
- DGX Station A100 160 GB
- DGX Station A800 320 GB
- DGX Station 32 GB

2.3.2. Release EL9-23.12

Release Date: December 19, 2023

2.3.2.1 Release Highlights

- Added support for Red Hat Enterprise Linux 9.3 and Rocky 9.3.
- Continued support for Red Hat Enterprise Linux 9.2 and Rocky Linux 9.2.
- Added support for single-port ConnectX-7 VPI adapter card for DGX A100 System.
- Added support for NVIDIA® OpenFabrics Enterprise Distribution for Linux (MLNX_OFED) version 23.10-1.1.9.0 - a long-term support (LTS) release.
- Continued support for DGX H100.

2.3.2.2 Qualified Software Stack

The following table shows the current version information of the software packages provided in the NVIDIA repositories for the NVIDIA DGX Software Stack.

Component	Latest versions in the repositories
DGX Base OS	**EL9-23.12**
OS	Red Hat Enterprise Linux 9.3 and Rocky Linux 9.3
Kernel	5.14.0-362.8.1.el9_3
GPU Driver and CUDA Toolkit	CUDA Toolkit 12.2 and GPU Driver 535.129.03 (Default)
NCCL	2.19.3
cuDNN	8.9.6
DCGM	3.3.0-002
GPU Direct Storage	1.7.2 or later
NVIDIA System Management (NVSM)	23.09.02
Docker-CE	24.0.7-1
NVIDIA Container Runtime	▶ nvidia-docker2: 2.13.0-1 ▶ nvidia-container-toolkit (and base): 1.14.3-1 ▶ libnvidia-container-tools: 1.14.3-1 ▶ libnvidia-container1: 1.14.3-1
MIG Configuration Tool	0.5.4-1
NGC CLI	3.17.0-1
DLFW (Deep Learning Frameworks)	23.10

The following table provides information about the supported OS and matching firmware versions for NVIDIA® OpenFabrics Enterprise Distribution for Linux (MLNX_OFED) version 23.10-1.1.9.0.

OS	DGX-1, DGX-2 ConnectX-4 (CX-4) or ConnectX-5 (CX-5)	DGX A100 ConnectX-6	DGX A100 ConnectX-7	DGX H100 ConnectX-7
RHEL 9	CX-5: 16.35.3006 CX-4: 12.28.2006	20.39.1002	28.39.1002	28.39.1002

2.3.2.3 Supported DGX Systems

NVIDIA has validated and tested EL9-23.12 with the following DGX systems:

- DGX H100
- DGX A100 640 GB
- DGX A100 320 GB
- DGX A800 640 GB
- DGX-2
- DGX-1 32 GB
- DGX Station A100 320 GB
- DGX Station A100 160 GB
- DGX Station 32 GB

2.3.2.4 Resolved Issues

The following issues have been resolved in the EL9-23.12 release:

Bug ID	Issue
4108242	Running `joc` tests resulted in an `unrecognized arguments: --local-rank` error with GPU driver R525.105.17.
4386925	GPUDirect RDMA bandwidth test failed with the `Xid (PCI:0000:0f:00): 79, pid='<unknown>', name=<unknown>, GPU has fallen off the bus` message.

2.3.3. Release EL9-23.08

2.3.3.1 Release Highlights

- Add support for NVIDIA DGX H100 System. Support is limited to the Red Hat Enterprise Linux 9.1 release.
- Add support for Red Hat Enterprise Linux 9.2 and Rocky Linux 9.2.

2.3.3.2 Qualified Software Stack

The following table provides version information for EL9-23.08 and the software it has been qualified:

Component	Latest versions in the repositories
Linux Distribution	Red Hat Enterprise Linux 9.2 and Rocky Linux 9.2 For NVIDIA DGX H100 Systems, only Red Hat Enterprise Linux 9.1 is supported.
GPU Driver	535.86.10
CUDA Toolkit	12.2.0
NCCL	2.18.3
CuDNN	8.9.2.26
DCGM	3.1.8
MLNX OFED	▶ ConnectX-7 with DGX H100: 5.9-0.5.6.0.125 ▶ ConnectX-7 with DGX A100: 5.4-3.7.5.0 ▶ ConnectX-6 with DGX A100: 5.8-3.0.7.0 ▶ ConnectX-5 and ConnectX-4: 5.8-3.0.7.0
MLNX FW	▶ ConnectX-7 and DGX H100: 28.36.2050 ▶ ConnectX-7 and DGX A100: 28.34.4000 ▶ ConnectX-6 and DGX A100: 20.35.4000 ▶ ConnectX-5: 16.35.3006 ▶ ConnectX-4: 12.28.2006
GPU Direct Storage	1.7.2
NVIDIA System Management (NVSM)	23.06.04
Docker Engine	23.0.4
NVIDIA Container Runtime	▶ nvidia-docker2: 2.13.1-1 ▶ nvidia-container-toolkit (and base): 1.13.1-1 ▶ libnvidia-container-tools: 1.13.1-1 ▶ libnvidia-container1: 1.13.1-1
MIG Configuration Tool	0.5.1
NGC CLI	3.17.0
DLFW (Deep Learning Frameworks)	23.07

The following table provides information about the supported OS and matching firmware versions for Mellanox OFED.

OS	DGX-1, DGX-2 ConnectX-4 or ConnectX-5	DGX A100 ConnectX-6 (CX-6)	DGX A100 ConnectX-7 (CX-7)	DGX H100 ConnectX-7 (CX-7)
RHEL 8	5.8-3.0.7.0 ▶ CX-5: 16.35.3006 ▶ CX-4: 12.28.2006 ▶ RHEL 8.8	5.8-3.0.7.0 ▶ CX-6: 20.35.3006 ▶ RHEL 8.8	5.4-3.7.5.0 ▶ CX-7: 28.34.4000 ▶ RHEL 8.8	5.9-0.5.6.0.127 ▶ CX-7: 28.36.2050 ▶ RHEL 8.7
RHEL 9	5.8-3.0.7.0 ▶ CX-5: 16.35.3006 ▶ CX-4: 12.28.2006 ▶ RHEL 9.2	5.8-3.0.7.0 ▶ CX-6: 20.35.3006 ▶ RHEL 9.2	5.4-3.7.5.0 ▶ CX-7: 28.34.4000 ▶ RHEL 9.2	5.9-0.5.6.0.127 ▶ CX-7: 28.36.2050 ▶ RHEL 9.1

2.3.3.3 Supported DGX Systems

NVIDIA has validated and tested EL9-23.08 with the following DGX systems:

- NVIDIA DGX H100
- NVIDIA DGX A100
- NVIDIA DGX Station A100
- NVIDIA DGX Station
- NVIDIA DGX-2
- NVIDIA DGX-1

2.3.4. Release EL9-23.01

Initial release of the DGX Software Stack for Red Hat Enterprise Linux 9.

2.3.4.1 Qualified Software Stack

The following table provides version information for EL9-23.01 and the software it has been qualified:

Component	Versions in this release
Linux Distribution	Red Hat Enterprise Linux 9.1 and Rocky Linux 9.1
GPU Driver	525.105.17
CUDA Toolkit	12.0
NCCL	2.18.1
CuDNN	8.9.1.23
DCGM	3.1.8
NVIDIA MLNX_OFED	5.8-2.0.3.0
NVIDIA ConnectX Firmware	▶ CX-4: 12.28.2006 ▶ CX-5: 16.35.2000 ▶ CX-6: 20.35.2000
NVIDIA System Management (NVSM)	22.12.04
Docker Engine	23.0.4
NVIDIA Container Runtime	▶ nvidia-docker2: 2.13.0-1 ▶ nvidia-container-toolkit (and base): 1.13.1-1 ▶ libnvidia-container-tools: 1.13.1-1 ▶ libnvidia-container1: 1.13.1-1
MIG Configuration Tool	0.5.1
NGC CLI	3.17.0
DLFW (Deep Learning Frameworks)	23.03

2.3.4.2 Supported DGX Systems

NVIDIA has validated and tested EL9-23.01 with the following DGX systems:

- DGX-1
- DGX-2
- DGX Station
- DGX A100
- DGX Station A100

Chapter 3. Installing Red Hat Enterprise Linux

Red Hat provides several methods for installing Red Hat Enterprise Linux as described in Performing a Standard RHEL 9 Installation.

The installation instructions in this topic are for the latest release of the DGX Software for Red Hat Enterprise Linux 9. The DGX Software does not include Extended Update Support (EUS), which is an optional subscription for Red Hat Enterprise Linux. For more information about EUS, refer to Red Hat Enterprise Linux (RHEL) Extended Update Support (EUS) Overview.

For convenience, this topic describes how to install Red Hat Enterprise Linux using the Quick Install method, and shows when to reclaim disk space in the process. It describes a minimal installation. If you have a preferred method for installing Red Hat Enterprise Linux, then you can skip this section but be sure to reclaim disk space occupied by the existing Ubuntu installation.

The interactive method described here installs Red Hat Enterprise Linux on DGX using a connected monitor and keyboard and USB stick with the ISO image, or remotely through the remote console of the BMC.

Important

Refer to the *Release Notes* for any critical information regarding supported releases or dependencies. You will also find the version of Red Hat Enterprise Linux that is qualified and tested for use with the DGX Software.

3.1. Obtaining Red Hat Enterprise Linux

Obtain the Red Hat Enterprise Linux 9 ISO image and store on your local disk or create a boot USB drive formatted for UEFI. See Downloading Red Hat Enterprise Linux for instructions.

3.2. Booting the Red Hat Enterprise Linux ISO Locally

1. Plug the USB flash drive containing the Red Hat Enterprise Linux 9 ISO image into the DGX.
2. Connect a monitor and keyboard directly to the DGX.
3. Boot the system and press **F11** when the NVIDIA logo appears to get to the boot menu.
4. Select the USB volume name that corresponds to the inserted USB flash drive, and boot the system from it.
5. Follow the instructions at *Installing and Configuring Red Hat Enterprise Linux*

3.3. Booting the Red Hat Enterprise Linux ISO Remotely

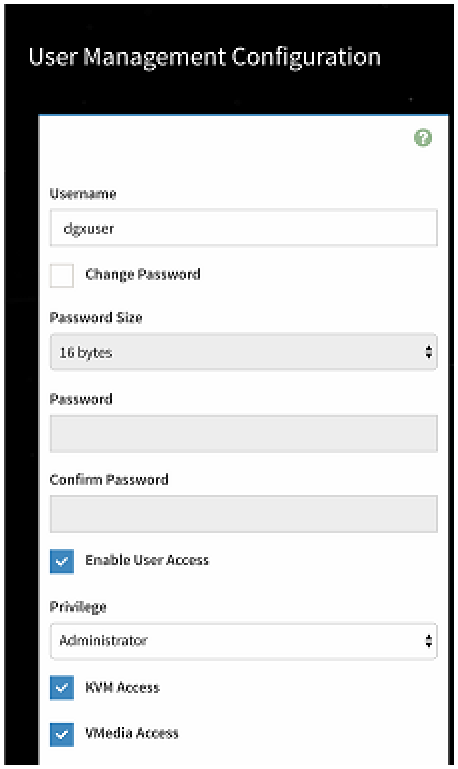
This chapter describes the steps for booting the Red Hat Enterprise Linux ISO remotely using the BMC:

- For DGX-2, DGX A100, DGX A800, or DGX H100/H200, refer to *Booting the ISO Image on the DGX-2, DGX A100, DGX A800, or DGX H100/H200 Remotely*.
- For DGX-1, refer to *Booting the ISO Image on the DGX-1 Remotely*.

Skip this chapter if you are using a monitor and keyboard for installing locally, or if you are installing on a DGX Station. The DGX Station **cannot** be booted remotely.

3.3.1. Booting the ISO Image on the DGX-2, DGX A100, DGX A800, or DGX H100/H200 Remotely

1. Connect to the BMC and ensure the required user privileges are set.
   a. Open a browser within your LAN and go to `https://<BMC-ip-address>/`, then log in.
   b. From the left-side menu, click Settings and then select User Management.
   c. Click the card with the user name that you created for the BMC.
   d. In the User Management Configuration dialog, make sure the VMedia Access checkbox is selected, then click Save.



User Management Configuration

Username: dgxuser

□ Change Password

Password Size: 16 bytes

Password:

Confirm Password:

☑ Enable User Access

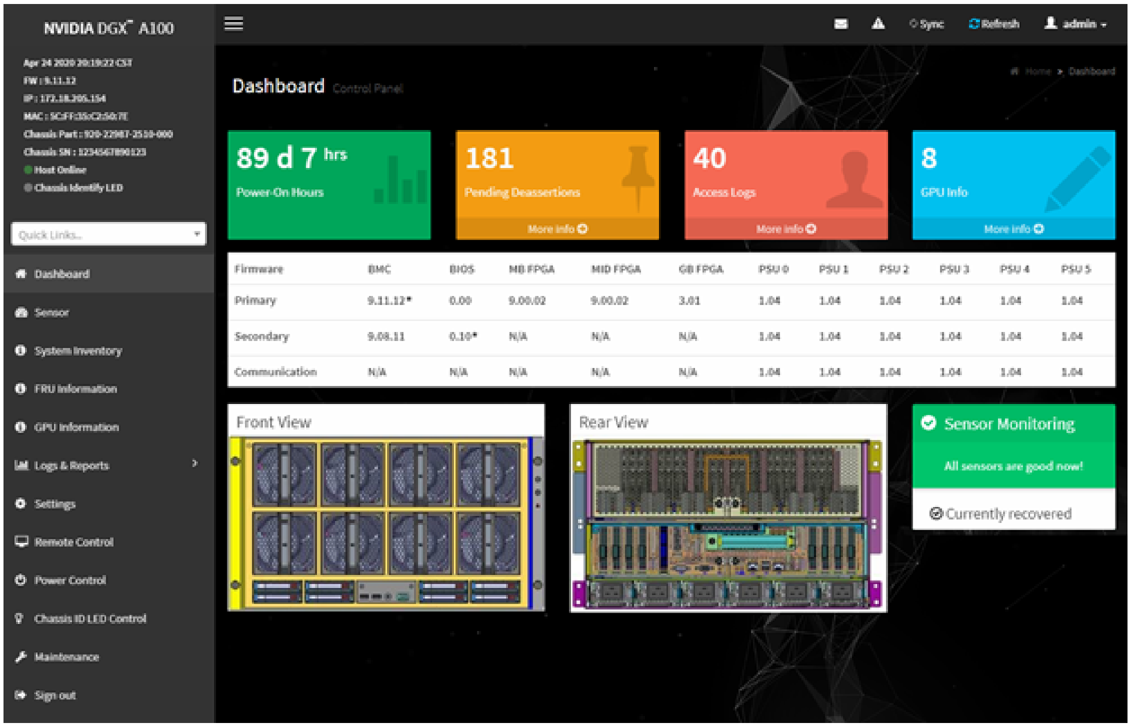
Privilege: Administrator

☑ KVM Access

☑ VMedia Access

2. Set up the ISO image as virtual media.

   a. From the left-side menu, click Remote Control.



NVIDIA DGX™ A100

Apr 24 2020 20:19:22 CST
FW : 9.11.12
IP : 172.18.205.154
MAC : 5C:FF:35:C2:50:7E
Chassis Part : 920-22987-2510-000
Chassis SN : 1234567890123
Host Online
Chassis Identify LED

Dashboard Control Panel

89 d 7 hrs Power-On Hours

181 Pending Deassertions

40 Access Logs

8 GPU Info

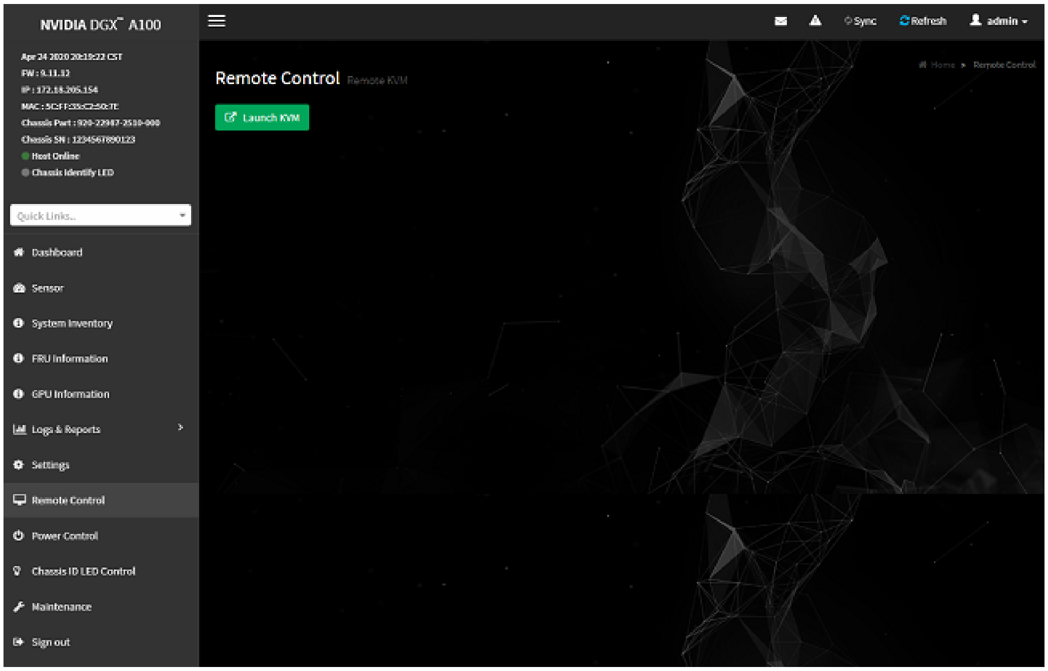
Firmware	BMC	BIOS	MB FPGA	MID FPGA	GB FPGA	PSU 0	PSU 1	PSU 2	PSU 3	PSU 4	PSU 5
Primary	9.11.12*	0.00	9.00.02	9.00.02	3.01	1.04	1.04	1.04	1.04	1.04	1.04
Secondary	9.08.11	0.10*	N/A	N/A	N/A	1.04	1.04	1.04	1.04	1.04	1.04
Communication	N/A	N/A	N/A	N/A	N/A	1.04	1.04	1.04	1.04	1.04	1.04

Sensor Monitoring

All sensors are good now!

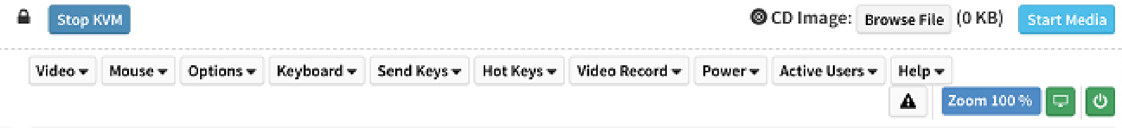
Currently recovered

   b. Select Launch KVM.



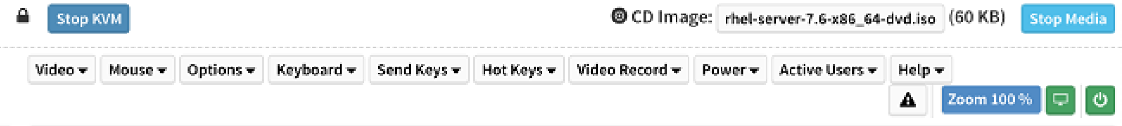
Apr 24 2020 20:19:22 CST
FW : 9.11.12
IP : 172.18.205.154
MAC : 5C:FF:35:C2:50:7E
Chassis Part : 920-22987-2510-000
Chassis SN : 1234567890123

c. From the top menu bar in the KVM window, click Browse File and select the ISO image, then click Start Media.



CD Image: Browse File (0 KB)

The CD image should now be connected.



CD Image: rhel-server-7.6-x86_64-dvd.iso (60 KB)

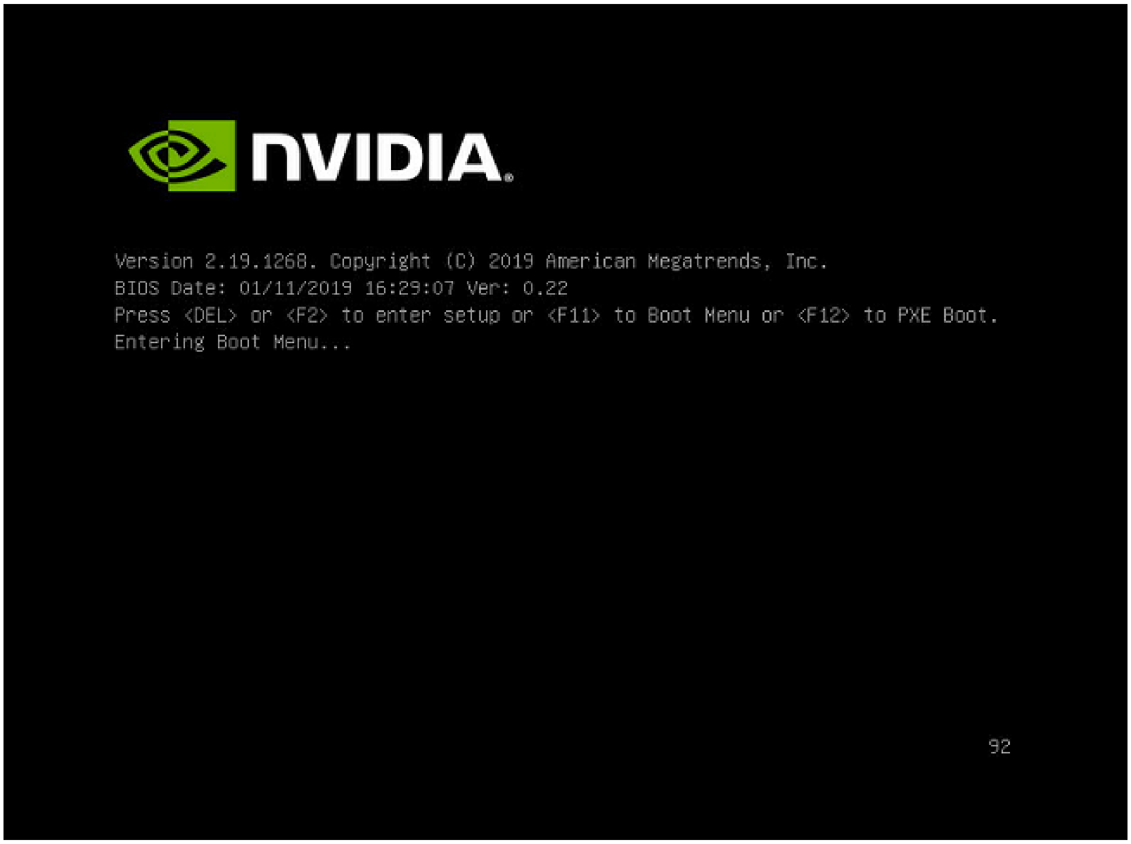
d. From the top menu bar in the KVM window, click Power and then selectReset Server.

3. Boot from the virtual media.

   Typically, the default boot order does not boot the CDROM image. You can change this in the BIOS or as a one-time option in the boot menu.

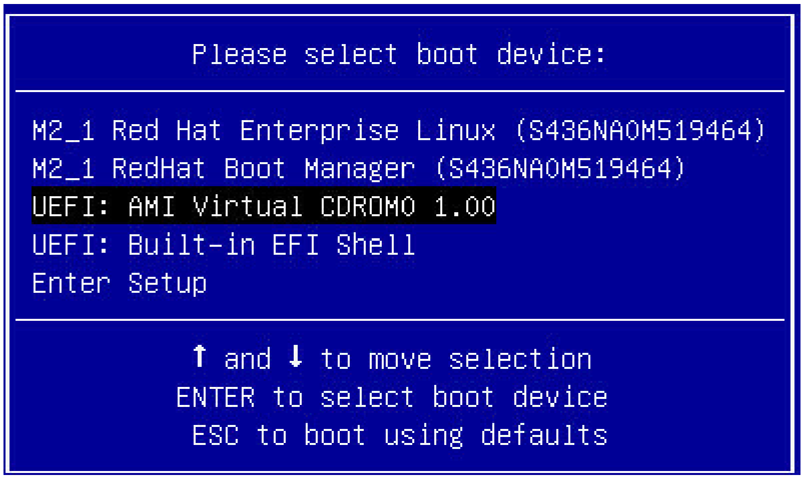
   a. To bring up the boot menu, press **F11** at the beginning of the boot process.

      Pressing **F11** will display Entering Boot Menu in the virtual display before entering the boot menu.



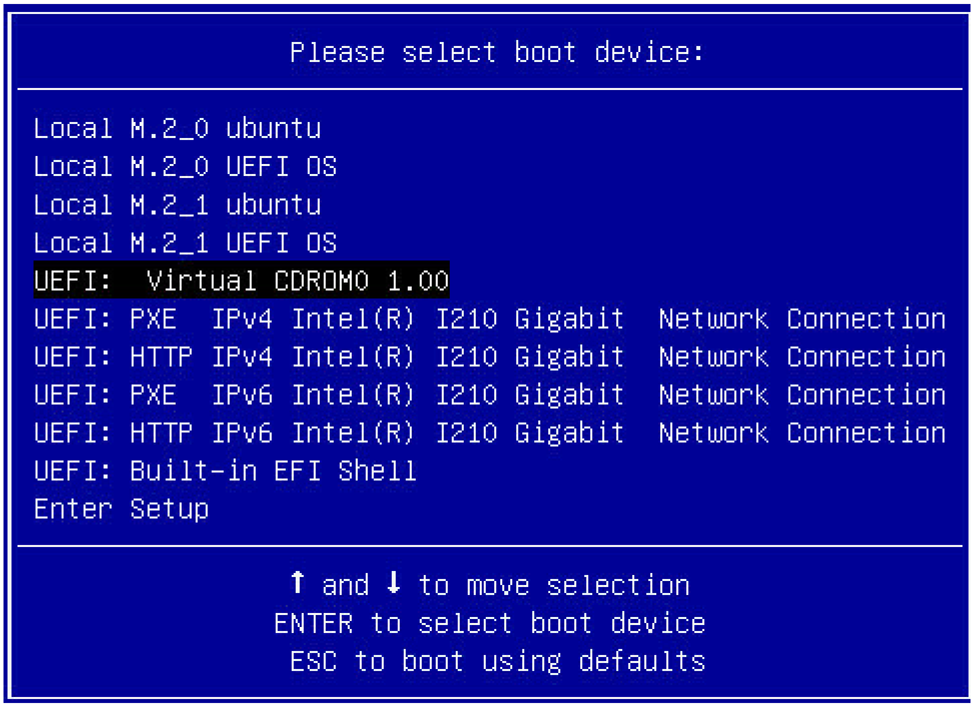
```
Version 2.19.1268. Copyright (C) 2019 American Megatrends, Inc.
BIOS Date: 01/11/2019 16:29:07 Ver: 0.22
Press <DEL> or <F2> to enter setup or <F11> to Boot Menu or <F12> to PXE Boot.
Entering Boot Menu...

                                                                    92
```

b. In the boot menu, select UEFI: AMI Virtual CDROM0 1.00 as the boot device and then press **ENTER**



```
          Please select boot device:

M2_1 Red Hat Enterprise Linux (S436NA0M519464)
M2_1 RedHat Boot Manager (S436NA0M519464)
UEFI: AMI Virtual CDROM0 1.00
UEFI: Built-in EFI Shell
Enter Setup

          ↑ and ↓ to move selection
        ENTER to select boot device
         ESC to boot using defaults
```

If you are booting from the DGX-2, the boot device is labeled UEFI: Virtual CDROM0 1.00.



```
                Please select boot device:

Local M.2_0 ubuntu
Local M.2_0 UEFI OS
Local M.2_1 ubuntu
Local M.2_1 UEFI OS
UEFI:  Virtual CDROM0 1.00
UEFI: PXE  IPv4 Intel(R) I210 Gigabit  Network Connection
UEFI: HTTP IPv4 Intel(R) I210 Gigabit  Network Connection
UEFI: PXE  IPv6 Intel(R) I210 Gigabit  Network Connection
UEFI: HTTP IPv6 Intel(R) I210 Gigabit  Network Connection
UEFI: Built-in EFI Shell
Enter Setup

                ↑ and ↓ to move selection
               ENTER to select boot device
                ESC to boot using defaults
```

c. Follow the instructions at *Installing and Configuring Red Hat Enterprise Linux*

3.3.2. Booting the ISO Image on the DGX-1 Remotely

1. Connect to the BMC and change user privileges.

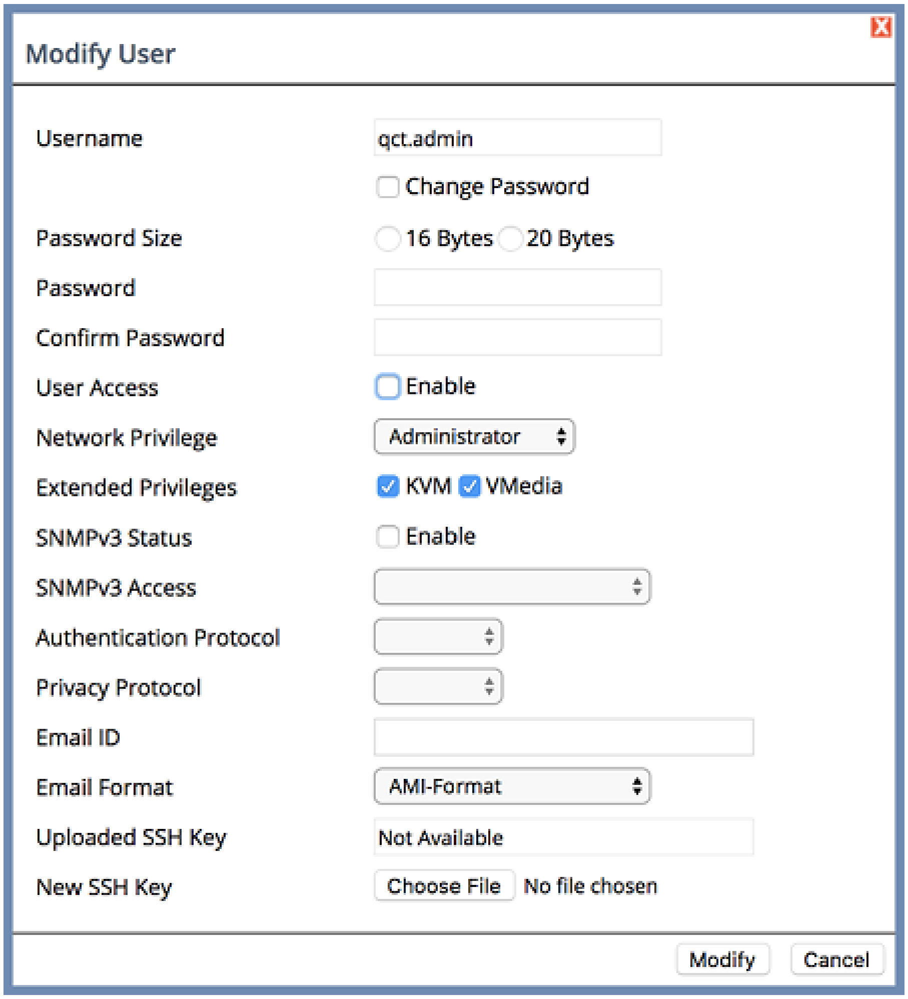
   a. Open a Java-enabled web browser within your LAN and go to `http://<BMC-ip-address>/`, then log in.

      Use Firefox or Internet Explorer. Google Chrome is not officially supported by the BMC.

   b. From the top menu, click Configuration and then select Users.

   c. Select the user name that you created for the BMC, then click Modify User.

   d. In the Modify User dialog, select the VMedia checkbox to add it to the extended privileges for the user, then click Modify.

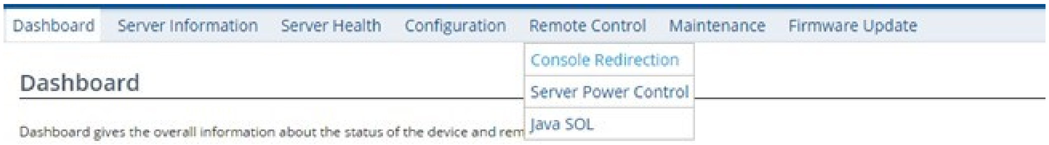


Modify User

Username	qct.admin
	□ Change Password
Password Size	○ 16 Bytes ○ 20 Bytes
Password	
Confirm Password	
User Access	□ Enable
Network Privilege	Administrator
Extended Privileges	☑ KVM ☑ VMedia
SNMPv3 Status	□ Enable
SNMPv3 Access	
Authentication Protocol	
Privacy Protocol	
Email ID	
Email Format	AMI-Format
Uploaded SSH Key	Not Available
New SSH Key	Choose File No file chosen

2. Set up the ISO image as virtual media and reboot the system.

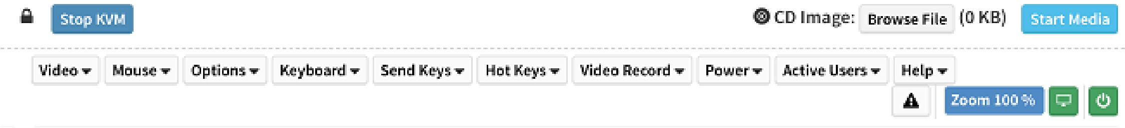
   a. From the top menu, click Remote Controland select Remote KVM.



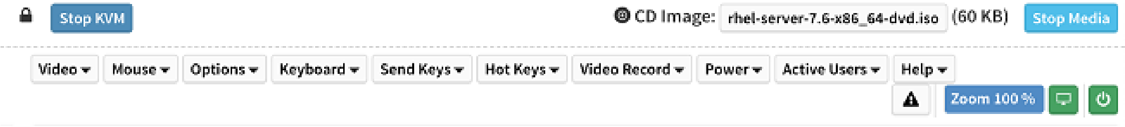
Dashboard gives the overall information about the status of the device and rem

   b. Click Launch KVM to open the remote console.

   c. From the top menu bar in the KVM window, click Browse File and select the ISO image, then click Start Media.



The CD image should now be connected.

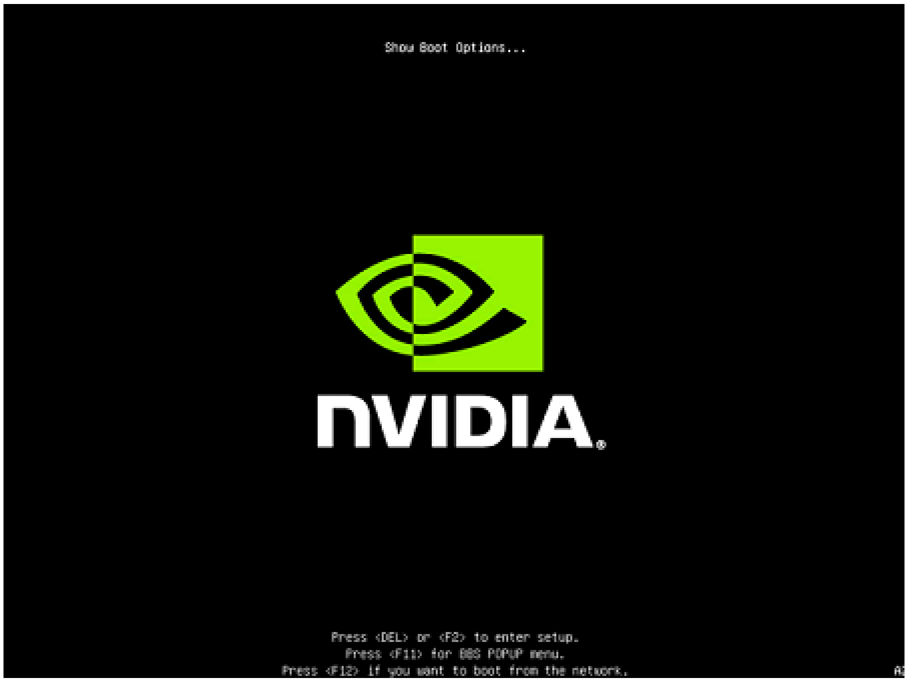


CD Image: rhel-server-7.6-x86_64-dvd.iso (60 KB)

   d. From the top menu bar in the KVM window, click Power and then selectReset Server.

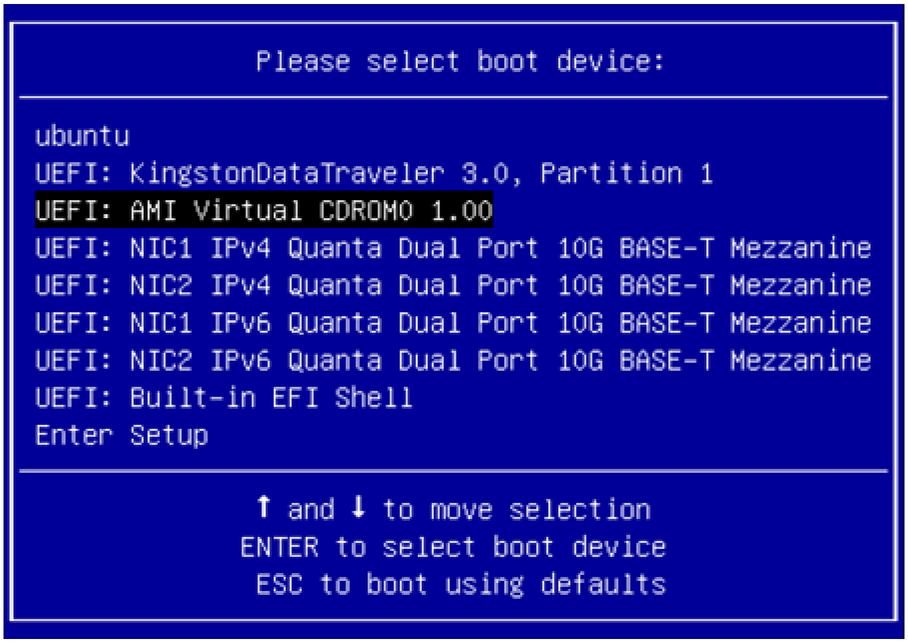
3. Boot the CD ROM image.

   The default boot order does not boot the CDROM image. This can be changed in the BIOS or as a one-time option in the boot menu. To bring up the boot menu, press F11 at the beginning of the boot process. Pressing **F11** will display Show Boot Options at the top of the virtual display before entering the boot menu. Use the “soft” keyboard (Menu Keyboard Layout SoftKeyboard <Language>) to bring up a virtual keyboard if pressing the physical key has no effect.



```
Show Boot Options...

Press <DEL> or <F2> to enter setup.
Press <F11> for BBS POPUP menu.
Press <F12> if you want to boot from the network.
```

   a. In the boot menu, select UEFI: AMI Virtual CDROM 1.00 as the boot device and then press **ENTER**



```
Please select boot device:

ubuntu
UEFI: KingstonDataTraveler 3.0, Partition 1
UEFI: AMI Virtual CDROM0 1.00
UEFI: NIC1 IPv4 Quanta Dual Port 10G BASE-T Mezzanine
UEFI: NIC2 IPv4 Quanta Dual Port 10G BASE-T Mezzanine
UEFI: NIC1 IPv6 Quanta Dual Port 10G BASE-T Mezzanine
UEFI: NIC2 IPv6 Quanta Dual Port 10G BASE-T Mezzanine
UEFI: Built-in EFI Shell
Enter Setup

↑ and ↓ to move selection
ENTER to select boot device
ESC to boot using defaults
```

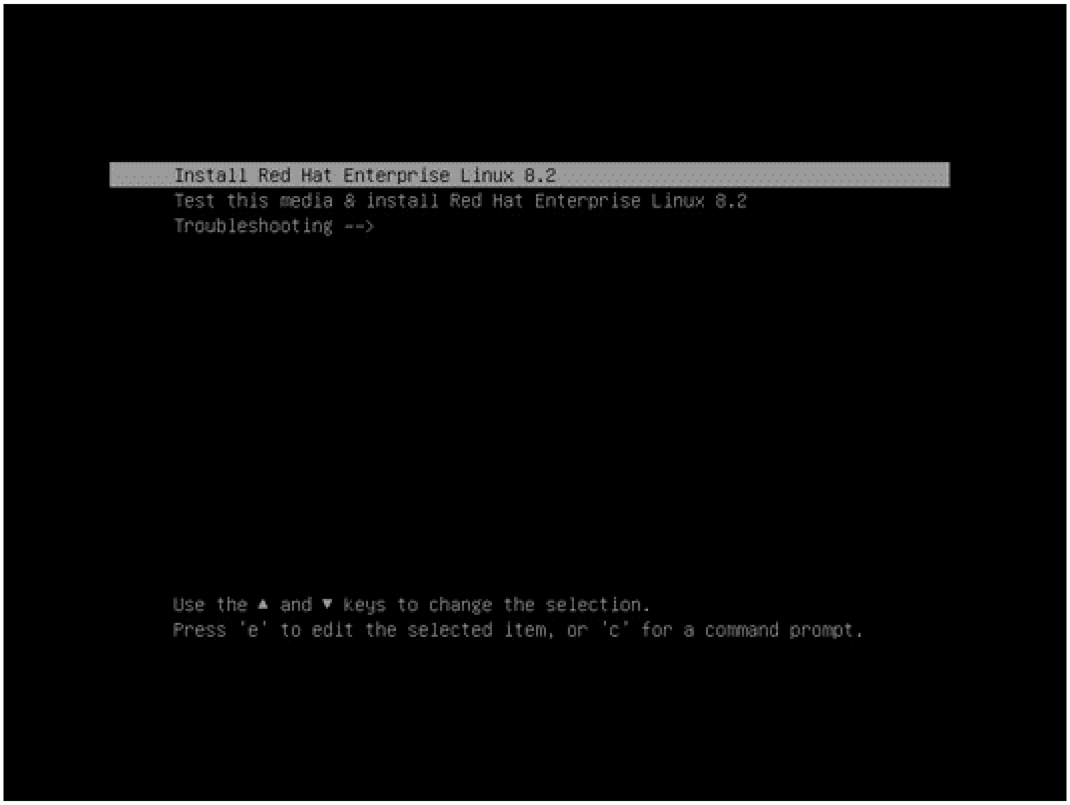
   b. Follow the instructions at *Installing and Configuring Red Hat Enterprise Linux*

3.4. Installing and Configuring Red Hat Enterprise Linux

This section assumes you have booted the Red Hat Enterprise Linux ISO image.

> **Note**
>
> **Rocky Linux**: The procedures for installing Rocky Linux are essentially the same as those described here, except where noted. The bootup installation screen will refer to Rocky Linux instead of Red Hat Enterprise Linux, and the Quick Start installer for Rocky Linux is based on the Red Hat Enterprise Linux installer, so the look is similar except, again, for the naming.

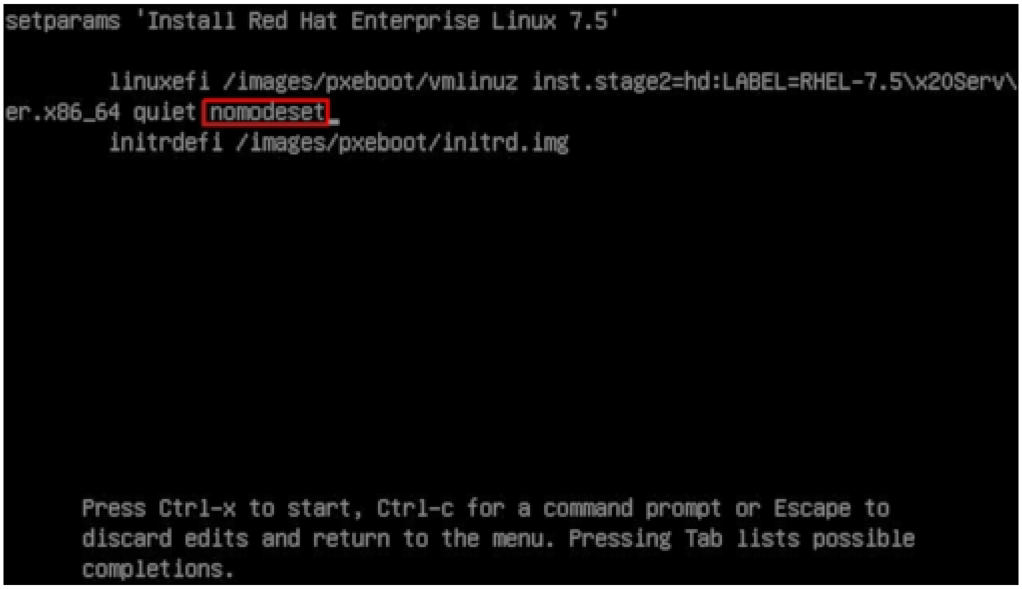
1. After booting the ISO image, the GRUB menu for the installer appears.



```
Install Red Hat Enterprise Linux 8.2
Test this media & install Red Hat Enterprise Linux 8.2
Troubleshooting -->

Use the ▲ and ▼ keys to change the selection.
Press 'e' to edit the selected item, or 'c' for a command prompt.
```

2. If you wish to automate the installation with a kickstart file, see the section Installing with Kickstart
3. Select `Install Red Hat Enterprise Linux` and press 'e' to edit the selected option.
   a. Add the "nomodeset" parameter as shown below.

      "nomodeset" prevents the in-box Nouveau driver from loading as it might not properly support the GPUs used in the DGX systems, resulting in display issues.



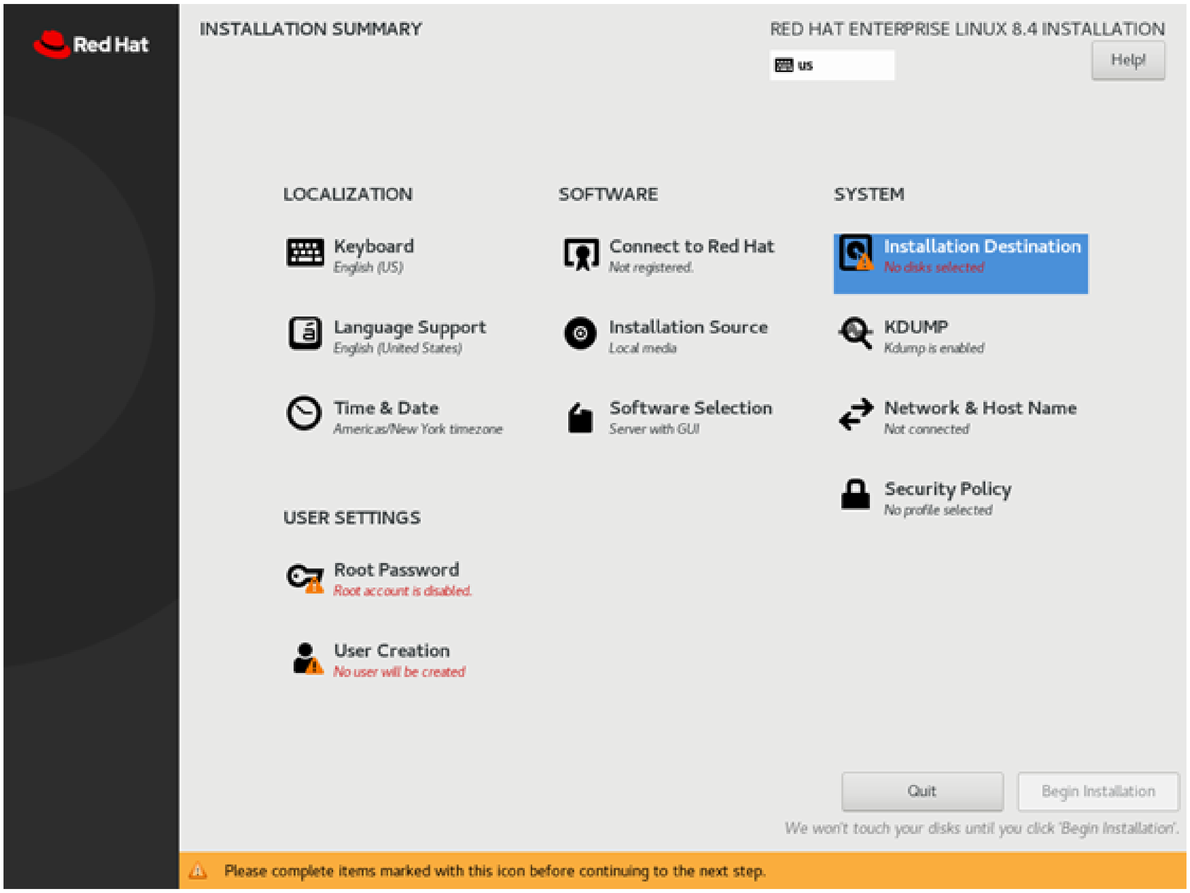
```
setparams 'Install Red Hat Enterprise Linux 7.5'

        linuxefi /images/pxeboot/vmlinuz inst.stage2=hd:LABEL=RHEL-7.5\x20Serv\
er.x86_64 quiet nomodeset_
        initrdefi /images/pxeboot/initrd.img





      Press Ctrl-x to start, Ctrl-c for a command prompt or Escape to
      discard edits and return to the menu. Pressing Tab lists possible
      completions.
```

   b. Press Ctrl-x to save your changes and start the installation.

4. Refer to Performing a Standard RHEL 9 Installation for additional guidance on using the RHEL 9 installer.
5. Configure the language, region, date, time, keyboard, and other configuration options you might need from the Installation Summary Screen.
6. Set the Software Selection to the correct value.
   - For DGX Station and DGX Station A100: Set to Server with GUI.
   - For DGX Servers (DGX 1-1, DGX-2, DGX A100, DGX A800, and DGX H100/H200): Set to Minimal Install.

Note

Setting the correct Software Selection is critical for proper operation.



INSTALLATION SUMMARY

RED HAT ENTERPRISE LINUX 8.4 INSTALLATION

LOCALIZATION

Keyboard
English (US)

Language Support
English (United States)

Time & Date
Americas/New York timezone

SOFTWARE

Connect to Red Hat
Not registered.

Installation Source
Local media

Software Selection
Server with GUI

SYSTEM

Installation Destination
No disks selected

KDUMP
Kdump is enabled

Network & Host Name
Not connected

Security Policy
No profile selected

USER SETTINGS

Root Password
Root account is disabled.

User Creation
No user will be created

We won't touch your disks until you click 'Begin Installation'.

Please complete items marked with this icon before continuing to the next step.

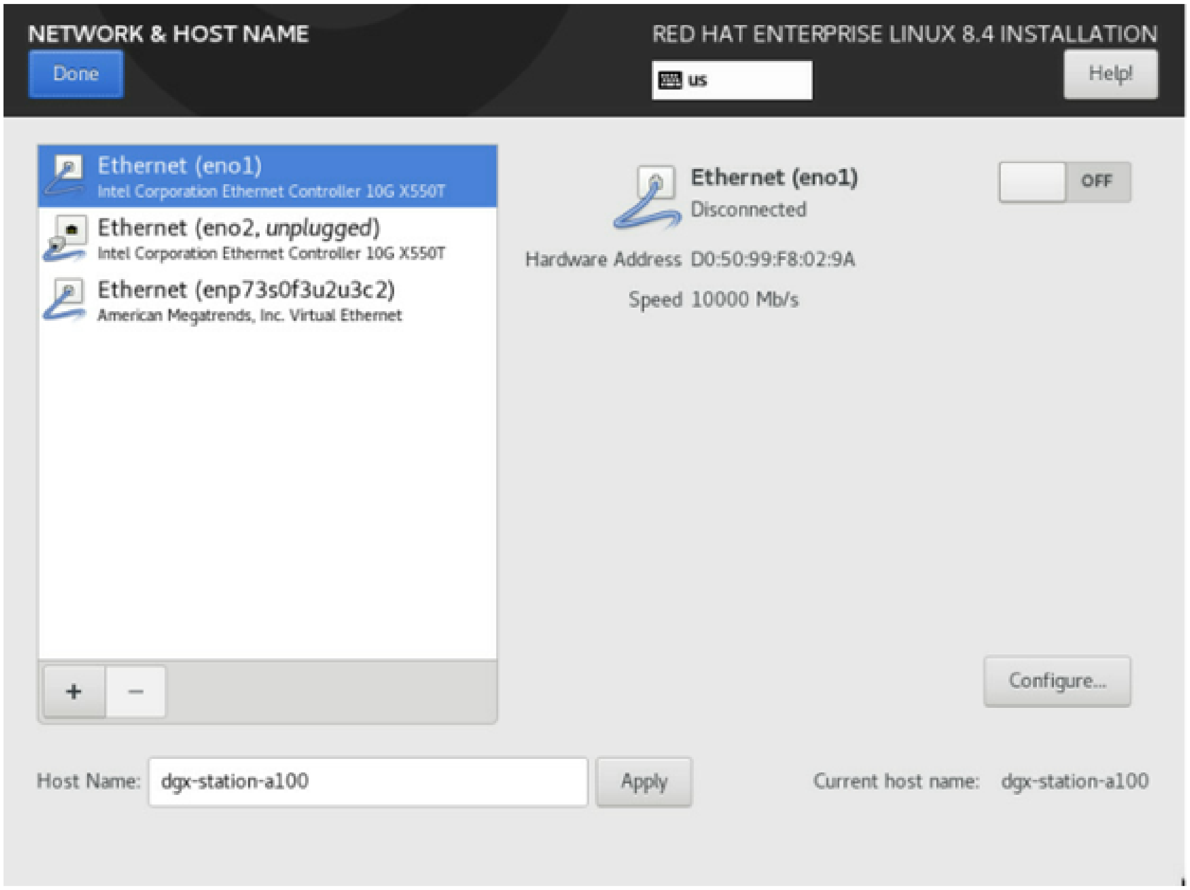
7. Partition your disks based on the platform and whether or not you would like to use encryption:

   - *Disk Partitioning for DGX-1, DGX Station, and DGX Station A100*
   - *Disk Partitioning with Encryption for DGX-1, DGX Station, and DGX Station A100*
   - *Disk Partitioning for DGX-2, DGX A100, DGX A800, and DGX H100/H200*
   - *Disk Partitioning with Encryption for DGX-2, DGX A100, DGX A800, and DGX H100/H200*

   Refer to *Red Hat Installer Fails to Delete Partitions* if you cannot delete existing partitions.

8. Configure Ethernet.

   From the Network & Host Name section, select and enable the Ethernet device. This defaults to DHCP and can be changed for static IP configurations under Configure.



Ethernet (eno1)
Intel Corporation Ethernet Controller 10G X550T

Ethernet (eno2, *unplugged*)
Intel Corporation Ethernet Controller 10G X550T

Ethernet (enp73s0f3u2u3c2)
American Megatrends, Inc. Virtual Ethernet

Ethernet (eno1)
Disconnected

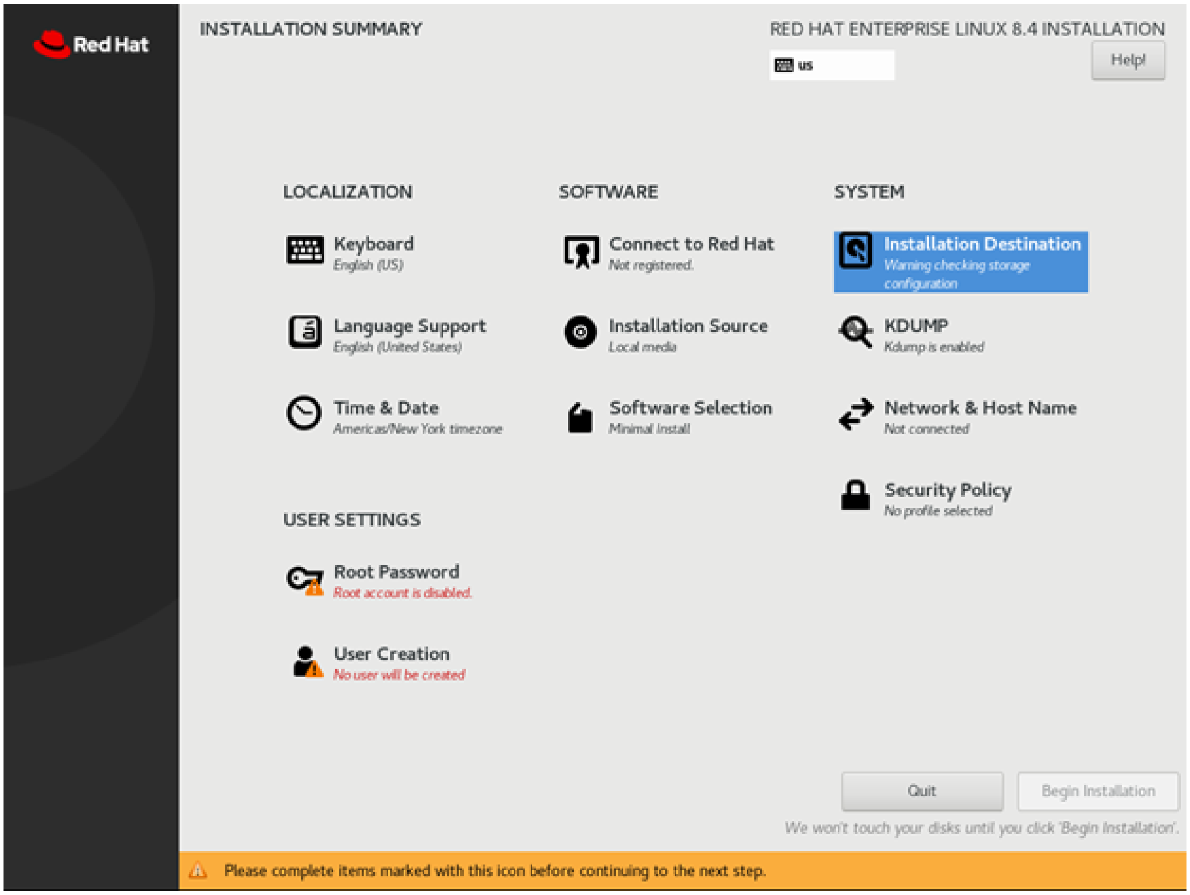
Hardware Address D0:50:99:F8:02:9A

Speed 10000 Mb/s

Host Name: dgx-station-a100

Current host name: dgx-station-a100

9. From the INSTALLATION SUMMARY screen, set your password (at **User Settings > Root Password**) and create a new user (at **User Settings > User Creation**), then click Begin installation to start the installation.



INSTALLATION SUMMARY

RED HAT ENTERPRISE LINUX 8.4 INSTALLATION

us

LOCALIZATION

Keyboard
English (US)

Language Support
English (United States)

Time & Date
Americas/New York timezone

SOFTWARE

Connect to Red Hat
Not registered.

Installation Source
Local media

Software Selection
Minimal Install

SYSTEM

Installation Destination
Warning checking storage configuration

KDUMP
Kdump is enabled

Network & Host Name
Not connected

Security Policy
No profile selected

USER SETTINGS

Root Password
Root account is disabled.

User Creation
No user will be created

We won't touch your disks until you click 'Begin Installation'.

⚠ Please complete items marked with this icon before continuing to the next step.

When the system reboots, be sure and eject any still-mounted installation discs.

10. Complete the initial Red Hat Enterprise Linux setup.

    - If you installed using the *Server with GUI* base environment, the Initial Setup starts automatically where you can accept the license agreement and register the system. See the Red Hat instructions for details.
    - If you are using CentOS you will need to accept the license agreement.
    - If you installed with any other base environment, log on to the system as root user and then register the system.

      ```
      subscription-manager register --auto-attach --username=user_name --
      ↪password=password
      ```

    - To prevent accidental upgrade of the Linux kernel, such as from RHEL 9.3 to 9.4, when precompiled GPU modules and drivers for that kernel are not available, you should pin the desired RHEL release by setting the `--set=<release>` option of the `release` command.

      For example, to stay on the RHEL 9.3 release:

      ```
      subscription-manager release --set=9.3
      ```

      You should check the *Release Notes* section for GPU driver and Linux kernel support before changing the `—set=<release>` setting and performing `sudo dnf update —nobest`.

See How to register and subscribe a system to the Red Hat Customer Portal using Red Hat Subscription-Manager for further information.

3.5. Partitioning

This chapter provides instructions for partitioning the DGX system:

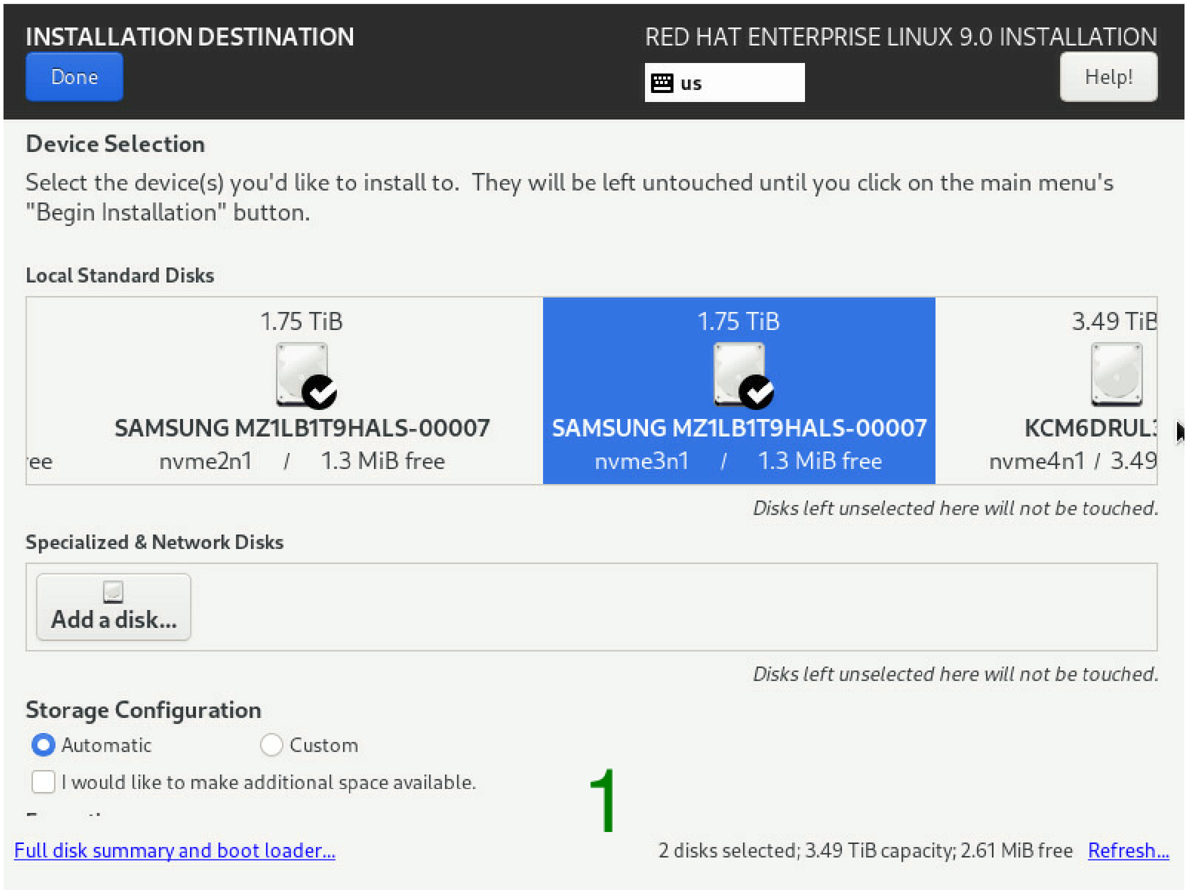
- *Disk Partitioning for DGX-1, DGX Station, and DGX Station A100*
- *Disk Partitioning for DGX-2, DGX A100, DGX A800, and DGX H100/H200*
- *Disk Partitioning with Encryption for DGX-1, DGX Station, and DGX Station A100*
- *Disk Partitioning with Encryption for DGX-2, DGX A100, DGX A800, and DGX H100/H200*

3.5.1. Disk Partitioning for DGX-1, DGX Station, and DGX Station A100

Note

The screenshots in the following section are taken from a DGX-1. The screens for the DGX Station and DGX Station A100 installations can present slightly different information for such things as disk size, disk space available, interface names, and so on.

1. At the Installation Destination screen, *select the smaller* of the available drives to install on.



INSTALLATION DESTINATION

RED HAT ENTERPRISE LINUX 9.0 INSTALLATION

Done

us

Help!

Device Selection

Select the device(s) you'd like to install to. They will be left untouched until you click on the main menu's "Begin Installation" button.

Local Standard Disks

1.75 TiB SAMSUNG MZ1LB1T9HALS-00007 nvme2n1 / 1.3 MiB free	1.75 TiB SAMSUNG MZ1LB1T9HALS-00007 nvme3n1 / 1.3 MiB free	3.49 TiB KCM6DRUL nvme4n1 / 3.49

Disks left unselected here will not be touched.

Specialized & Network Disks

Add a disk...

Disks left unselected here will not be touched.

Storage Configuration

● Automatic ○ Custom

□ I would like to make additional space available.

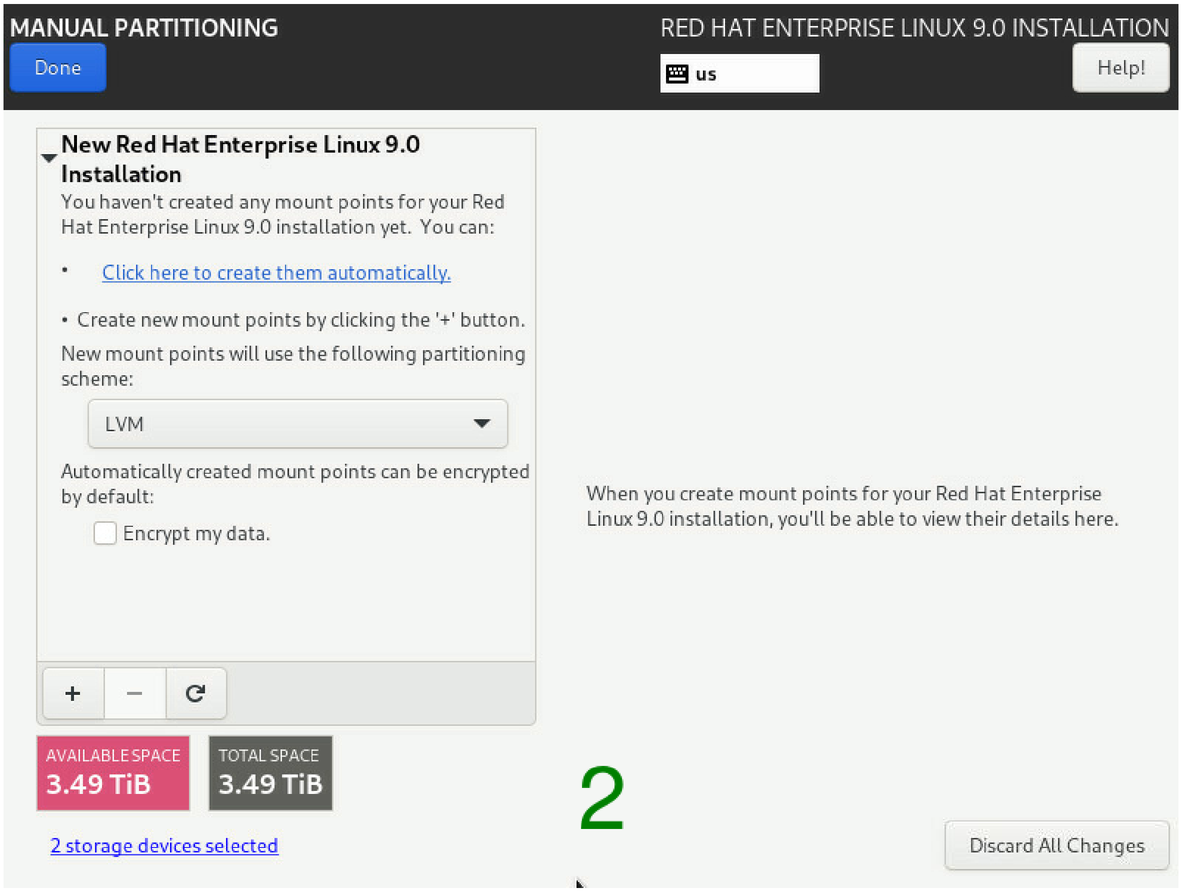
Full disk summary and boot loader...

2 disks selected; 3.49 TiB capacity; 2.61 MiB free Refresh...

Under **Storage Configuration**, click the **Custom** radio button and then click **Done**.

This brings up the **Manual Partitioning** window.

2. Expand the drop-down menu for the device you've selected and delete all existing partitions until there are none remaining.



MANUAL PARTITIONING

RED HAT ENTERPRISE LINUX 9.0 INSTALLATION

New Red Hat Enterprise Linux 9.0 Installation

You haven't created any mount points for your Red Hat Enterprise Linux 9.0 installation yet. You can:

- Click here to create them automatically.
- Create new mount points by clicking the '+' button.

New mount points will use the following partitioning scheme: LVM

Automatically created mount points can be encrypted by default:

□ Encrypt my data.

When you create mount points for your Red Hat Enterprise Linux 9.0 installation, you'll be able to view their details here.

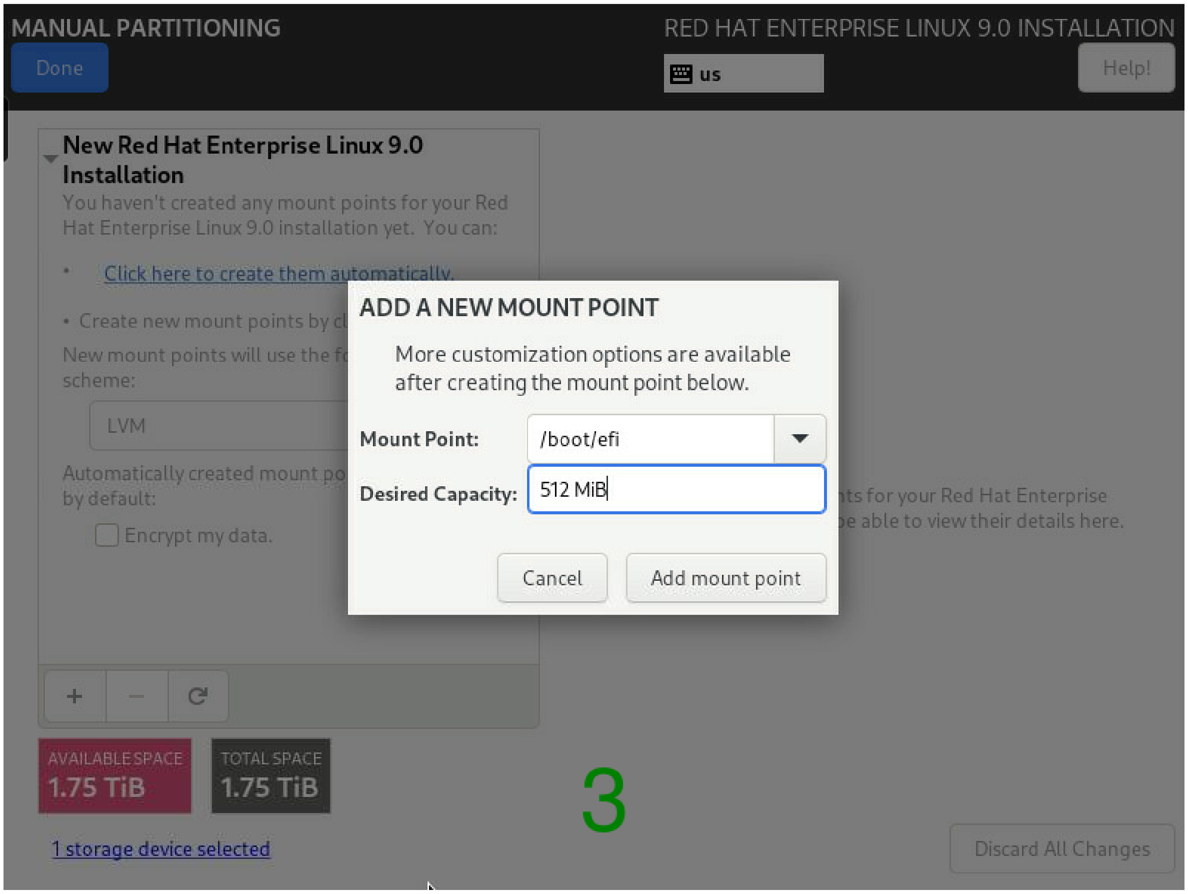
AVAILABLE SPACE 3.49 TiB

TOTAL SPACE 3.49 TiB

2 storage devices selected

3. Click the **+** button on the bottom left to create a new partition.

Create the `/boot/efi` partition with a size of 512 MiB.



New Red Hat Enterprise Linux 9.0 Installation

You haven't created any mount points for your Red Hat Enterprise Linux 9.0 installation yet. You can:

- Click here to create them automatically.
- Create new mount points by cl

New mount points will use the fo scheme:

LVM

Automatically created mount po by default:

□ Encrypt my data.

nts for your Red Hat Enterprise be able to view their details here.

ADD A NEW MOUNT POINT

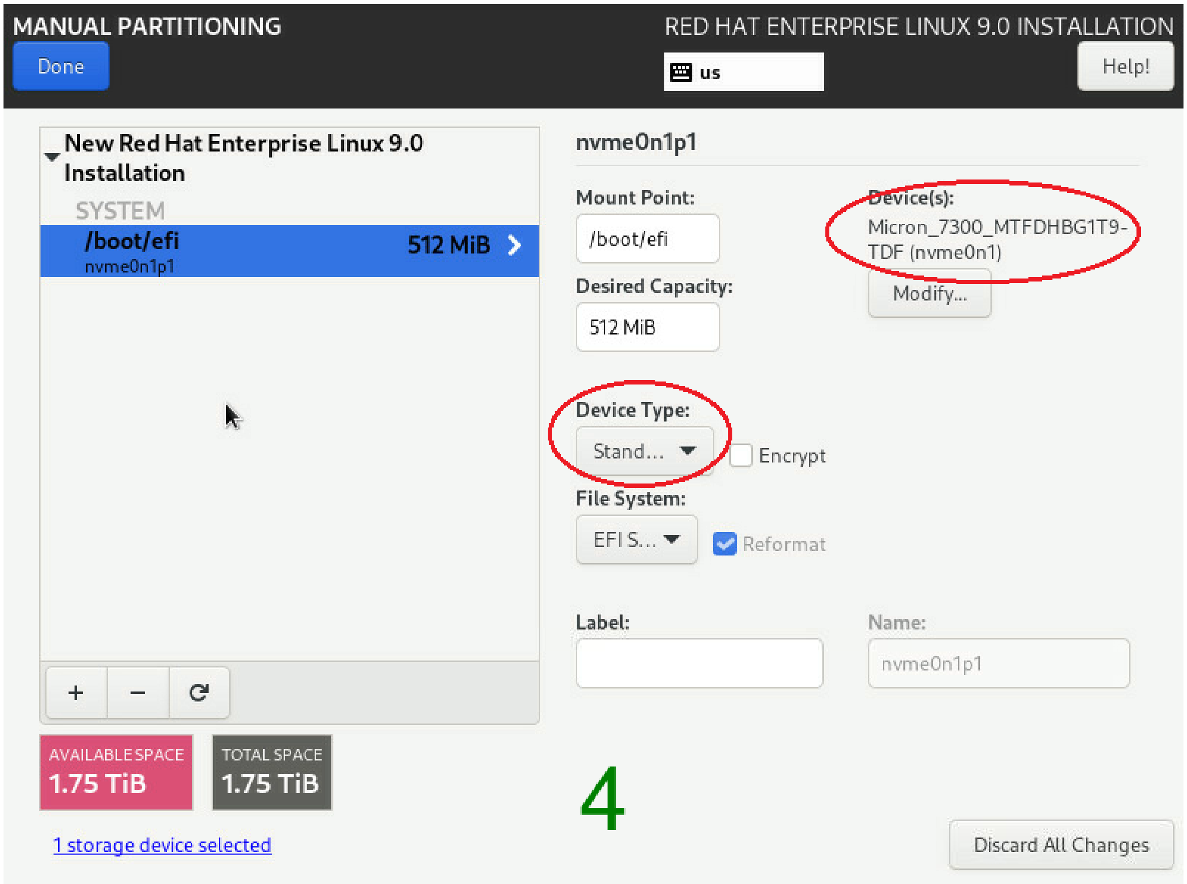
More customization options are available after creating the mount point below.

Mount Point: /boot/efi

Desired Capacity: 512 MiB

Cancel | Add mount point

4. Select **Standard Partition** for the **Device Type**.



MANUAL PARTITIONING

RED HAT ENTERPRISE LINUX 9.0 INSTALLATION

New Red Hat Enterprise Linux 9.0 Installation

SYSTEM

/boot/efi 512 MiB
nvme0n1p1

nvme0n1p1

Mount Point: /boot/efi

Device(s): Micron_7300_MTFDHBG1T9-TDF (nvme0n1)

Desired Capacity: 512 MiB

Device Type: Stand…

□ Encrypt

File System: EFI S…

☑ Reformat

Label:

Name: nvme0n1p1

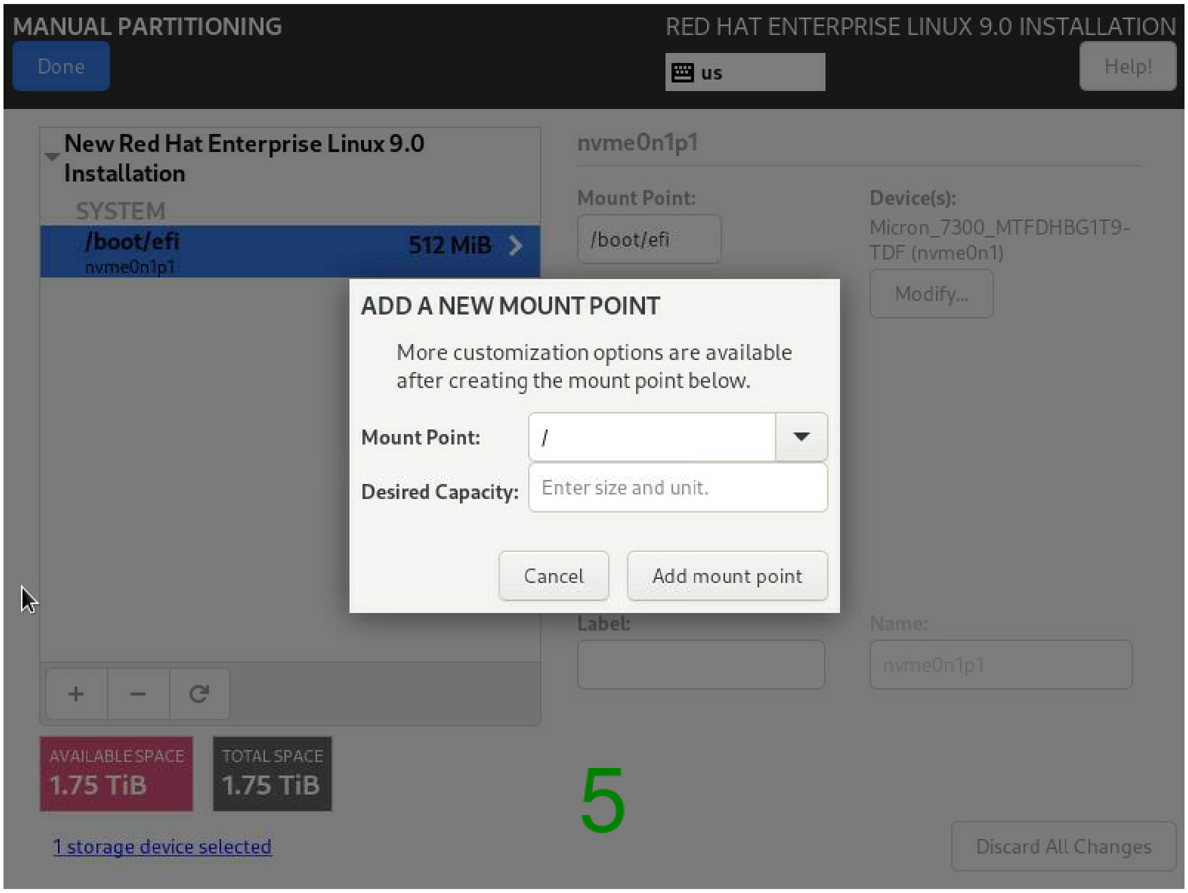
AVAILABLE SPACE 1.75 TiB

TOTAL SPACE 1.75 TiB

1 storage device selected

Under **Devices**, ensure only the drive you chose in step 1 is selected. Click **Update Settings** to confirm your changes.

5. Click the **+** button on the bottom left again to create another new partition. Create the **/** partition, this time leaving the **Desired Capacity** field empty. This lets the installer know to use the remaining capacity of the disks.



ADD A NEW MOUNT POINT

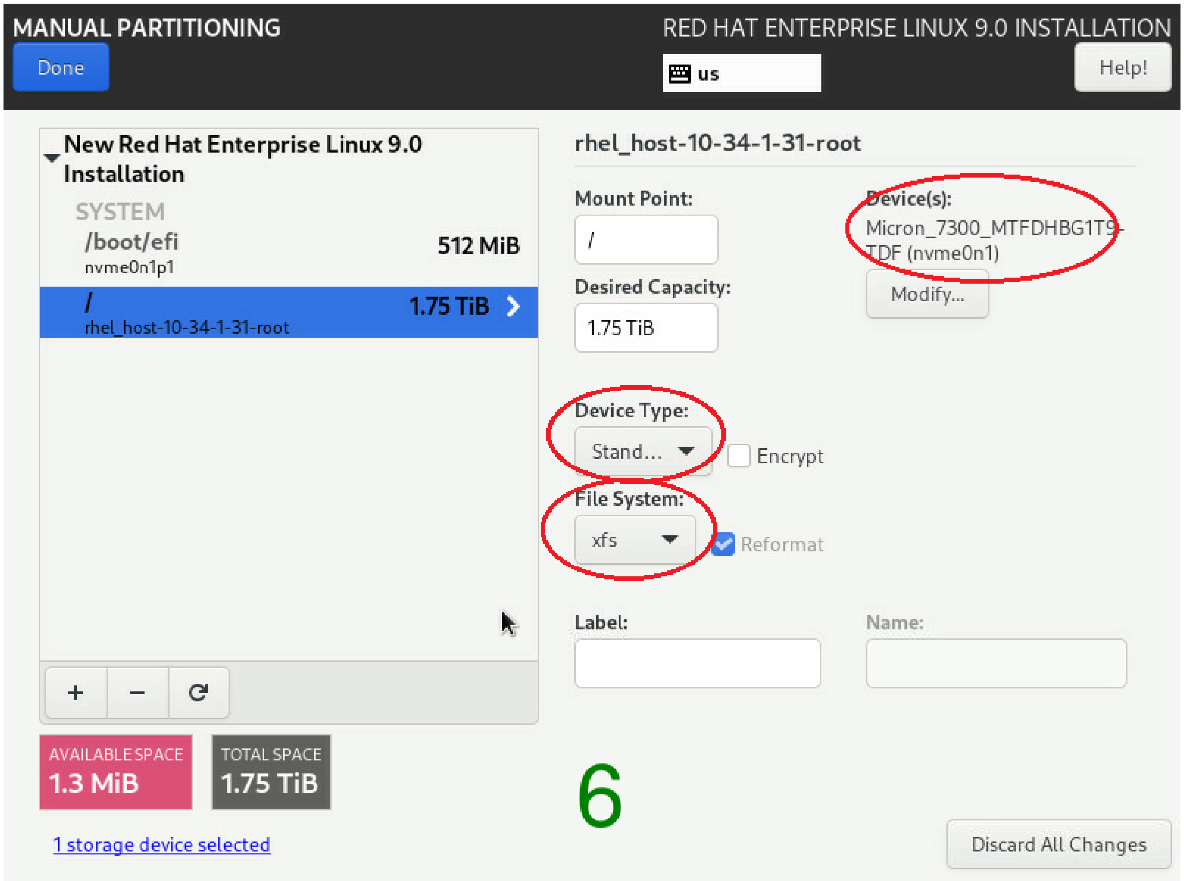
More customization options are available after creating the mount point below.

Mount Point: /

Desired Capacity: Enter size and unit.

Cancel | Add mount point

6. For the **/** partition, again select **Standard Partition** for the **Device Type**. Under **Devices**, ensure that only the drive you chose in step 1 is selected. For the **File System**, select **XFS.**



MANUAL PARTITIONING

Done

RED HAT ENTERPRISE LINUX 9.0 INSTALLATION

us

Help!

New Red Hat Enterprise Linux 9.0 Installation

SYSTEM

/boot/efi 512 MiB
nvme0n1p1

/ 1.75 TiB
rhel_host-10-34-1-31-root

AVAILABLE SPACE 1.3 MiB

TOTAL SPACE 1.75 TiB

1 storage device selected

rhel_host-10-34-1-31-root

Mount Point: /

Desired Capacity: 1.75 TiB

Device(s): Micron_7300_MTFDHBG1T9TDF (nvme0n1)

Modify...

Device Type: Stand...

□ Encrypt

File System: xfs

☑ Reformat

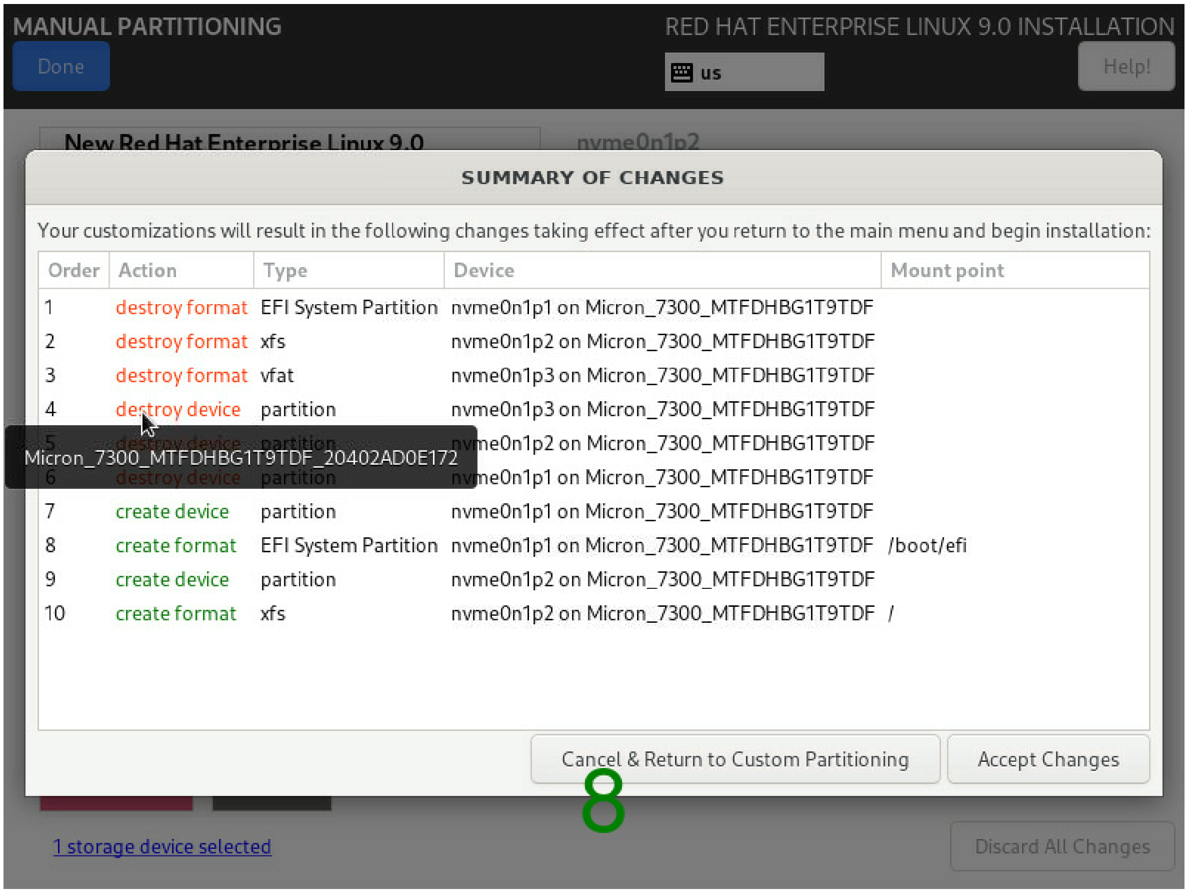
Label:

Name:

Discard All Changes

Under **Devices**, ensure that only the drive you chose in step 1 is selected. For the **File System**, select **XFS**. Click **Update Settings** to confirm your changes.

7. Click **Done**. This causes a yellow warning bar to appear on the bottom because no swap partition has been created.
8. Click **Done** again and then click **Accept Changes** to write all of our customizations to disk.


MANUAL PARTITIONING
Done
RED HAT ENTERPRISE LINUX 9.0 INSTALLATION
us
Help!
New Red Hat Enterprise Linux 9.0
nvme0n1p2
Micron_7300_MTFDHBG1T9TDF_20402AD0E172
Cancel & Return to Custom Partitioning
Accept Changes
8
1 storage device selected
Discard All Changes

SUMMARY OF CHANGES

Your customizations will result in the following changes taking effect after you return to the main menu and begin installation:

Order	Action	Type	Device	Mount point
1	destroy format	EFI System Partition	nvme0n1p1 on Micron_7300_MTFDHBG1T9TDF	
2	destroy format	xfs	nvme0n1p2 on Micron_7300_MTFDHBG1T9TDF	
3	destroy format	vfat	nvme0n1p3 on Micron_7300_MTFDHBG1T9TDF	
4	destroy device	partition	nvme0n1p3 on Micron_7300_MTFDHBG1T9TDF	
5	destroy device	partition	nvme0n1p2 on Micron_7300_MTFDHBG1T9TDF	
6	destroy device	partition	nvme0n1p1 on Micron_7300_MTFDHBG1T9TDF	
7	create device	partition	nvme0n1p1 on Micron_7300_MTFDHBG1T9TDF	
8	create format	EFI System Partition	nvme0n1p1 on Micron_7300_MTFDHBG1T9TDF	/boot/efi
9	create device	partition	nvme0n1p2 on Micron_7300_MTFDHBG1T9TDF	
10	create format	xfs	nvme0n1p2 on Micron_7300_MTFDHBG1T9TDF	/

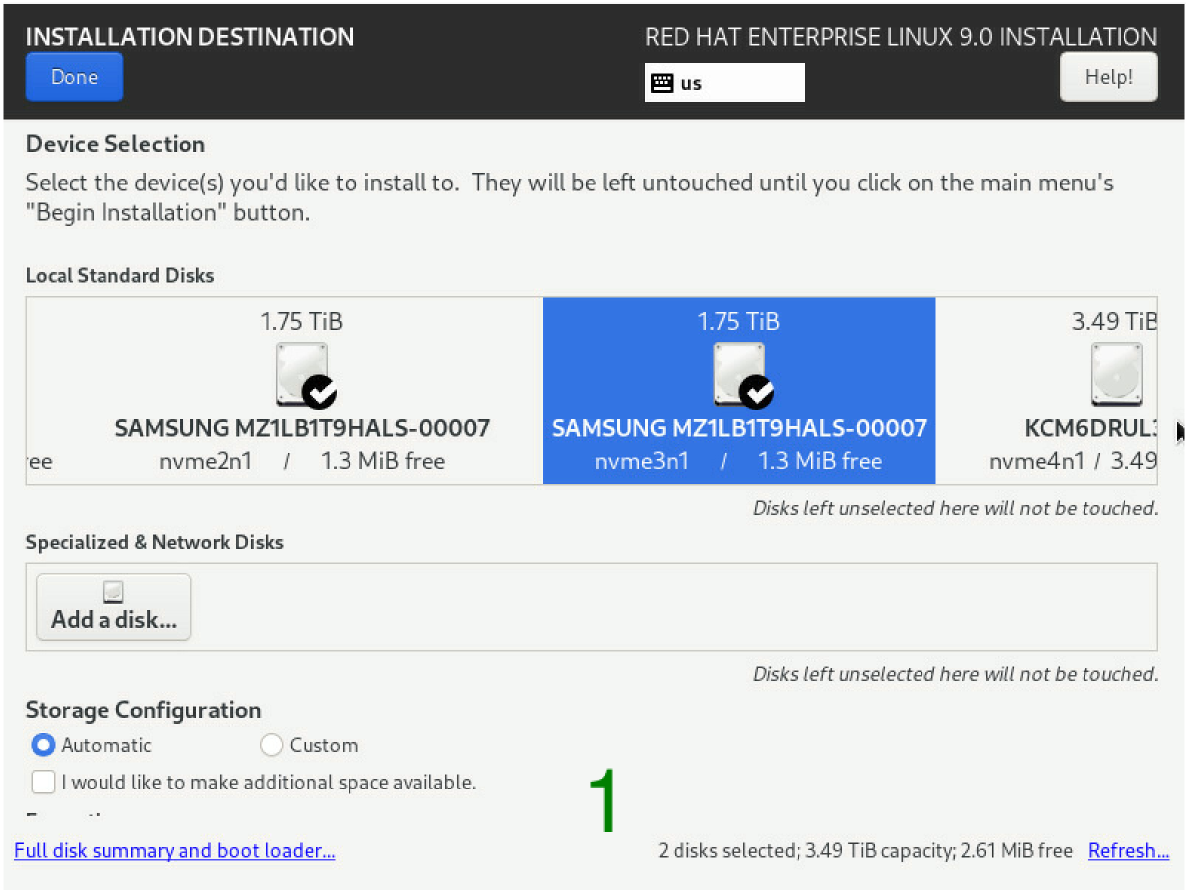
9. Return to step 8 in the *Installing and Configuring Red Hat Enterprise Linux* section.

3.5.2. Disk Partitioning with Encryption for DGX-1, DGX Station, and DGX Station A100

Note

The screenshots in the following section are taken from a DGX-1. The screens for the DGX Station and DGX Station A100 installations can present slightly different information for such things as disk size, disk space available, interface names, etc.

1. At the Installation Destination screen, *select the smaller* of the available drives to install on. Under **Storage Configuration**, click the **Custom** radio button and then click Done.



Device Selection

Select the device(s) you'd like to install to. They will be left untouched until you click on the main menu's "Begin Installation" button.

Local Standard Disks

ee

1.75 TiB
SAMSUNG MZ1LB1T9HALS-00007
nvme2n1 / 1.3 MiB free

1.75 TiB
SAMSUNG MZ1LB1T9HALS-00007
nvme3n1 / 1.3 MiB free

3.49 TiB
KCM6DRUL:
nvme4n1 / 3.49

Disks left unselected here will not be touched.

Specialized & Network Disks

Add a disk...

Disks left unselected here will not be touched.

Storage Configuration

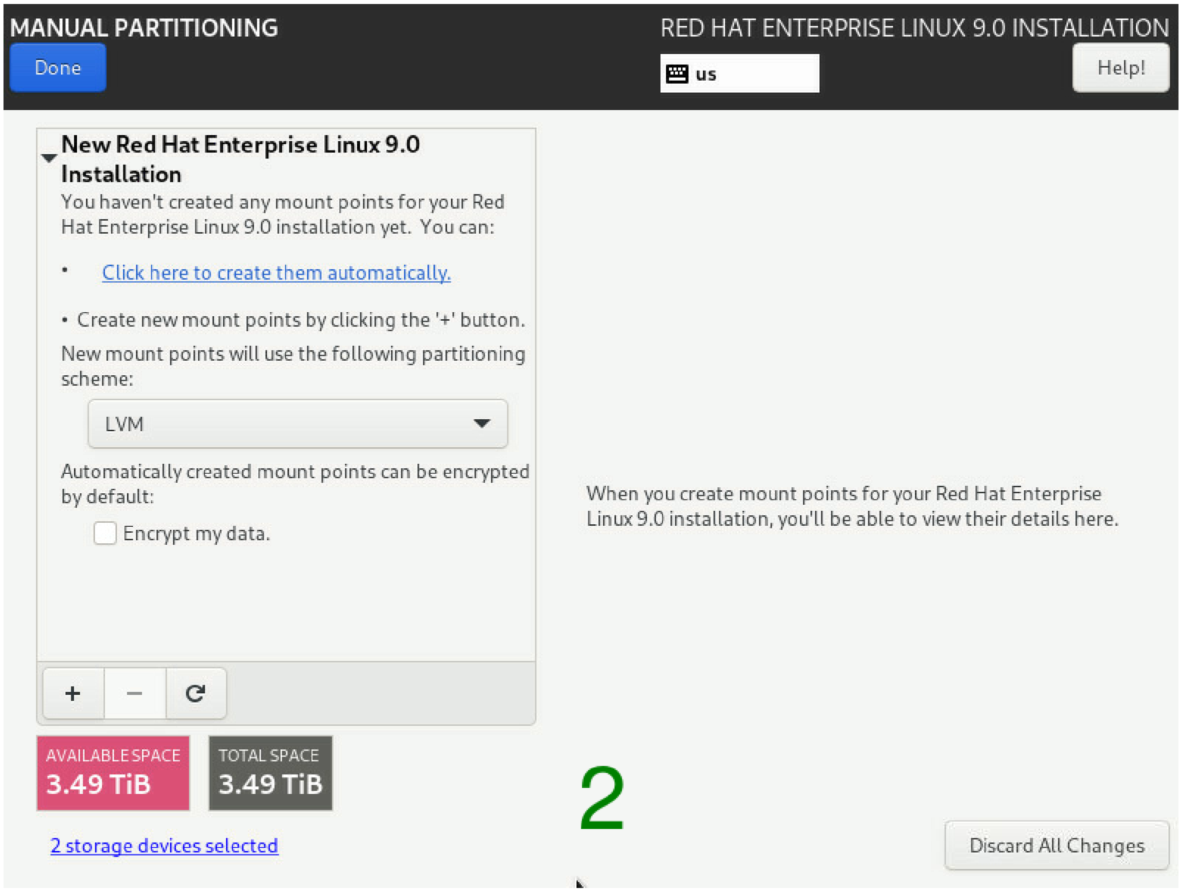
● Automatic ○ Custom

□ I would like to make additional space available.

Full disk summary and boot loader...

2 disks selected; 3.49 TiB capacity; 2.61 MiB free Refresh...

2. This brings up the **Manual Partitioning** window. Expand the drop-down menu for the device you've selected and delete all existing partitions until there are none remaining.



New Red Hat Enterprise Linux 9.0 Installation

You haven't created any mount points for your Red Hat Enterprise Linux 9.0 installation yet. You can:

- Click here to create them automatically.
- Create new mount points by clicking the '+' button.

New mount points will use the following partitioning scheme:

LVM

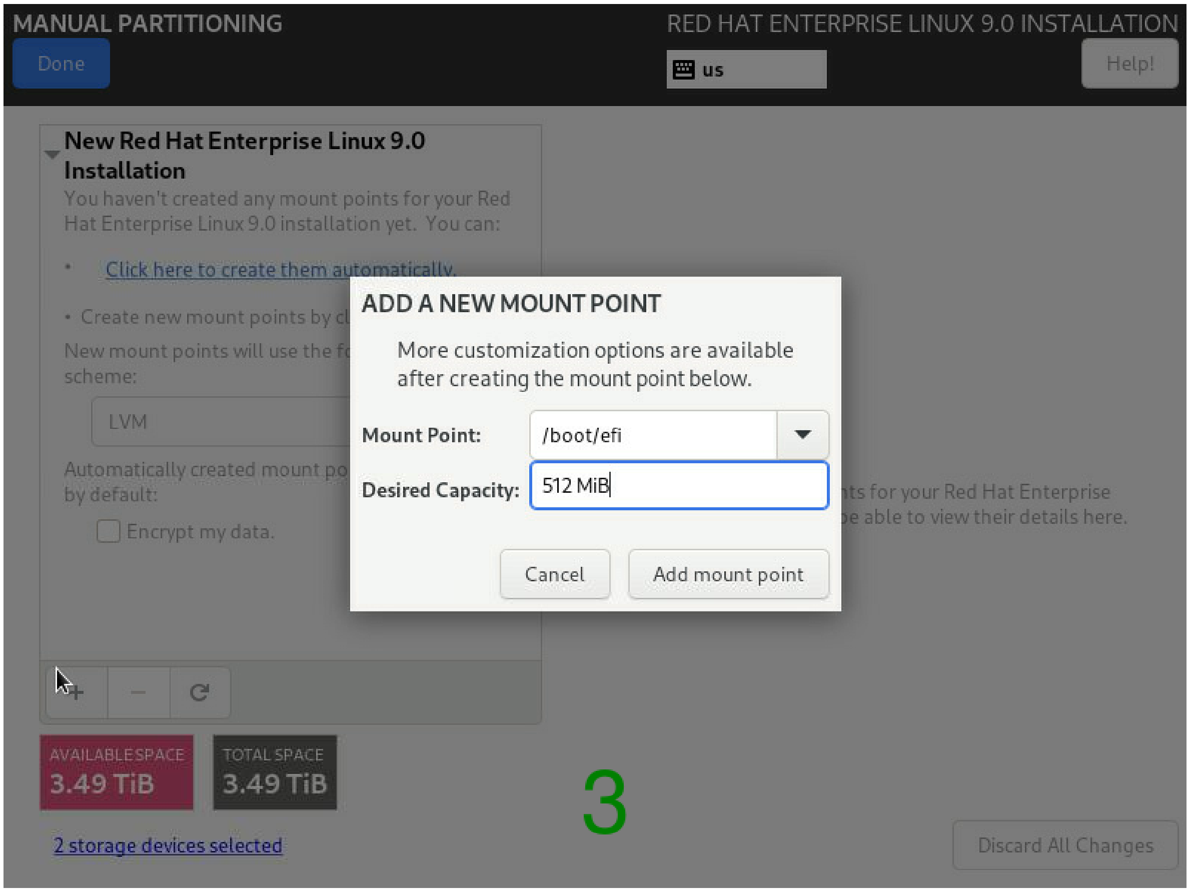
Automatically created mount points can be encrypted by default:

□ Encrypt my data.

When you create mount points for your Red Hat Enterprise Linux 9.0 installation, you'll be able to view their details here.

3. Click the **+** button on the bottom left to create a new partition.

   Create the `/boot/efi` partition with a size of 512 MiB.



New Red Hat Enterprise Linux 9.0 Installation

You haven't created any mount points for your Red Hat Enterprise Linux 9.0 installation yet. You can:

- Click here to create them automatically.
- Create new mount points by cl

New mount points will use the fo scheme:

LVM

Automatically created mount po by default:

□ Encrypt my data.

nts for your Red Hat Enterprise be able to view their details here.

ADD A NEW MOUNT POINT

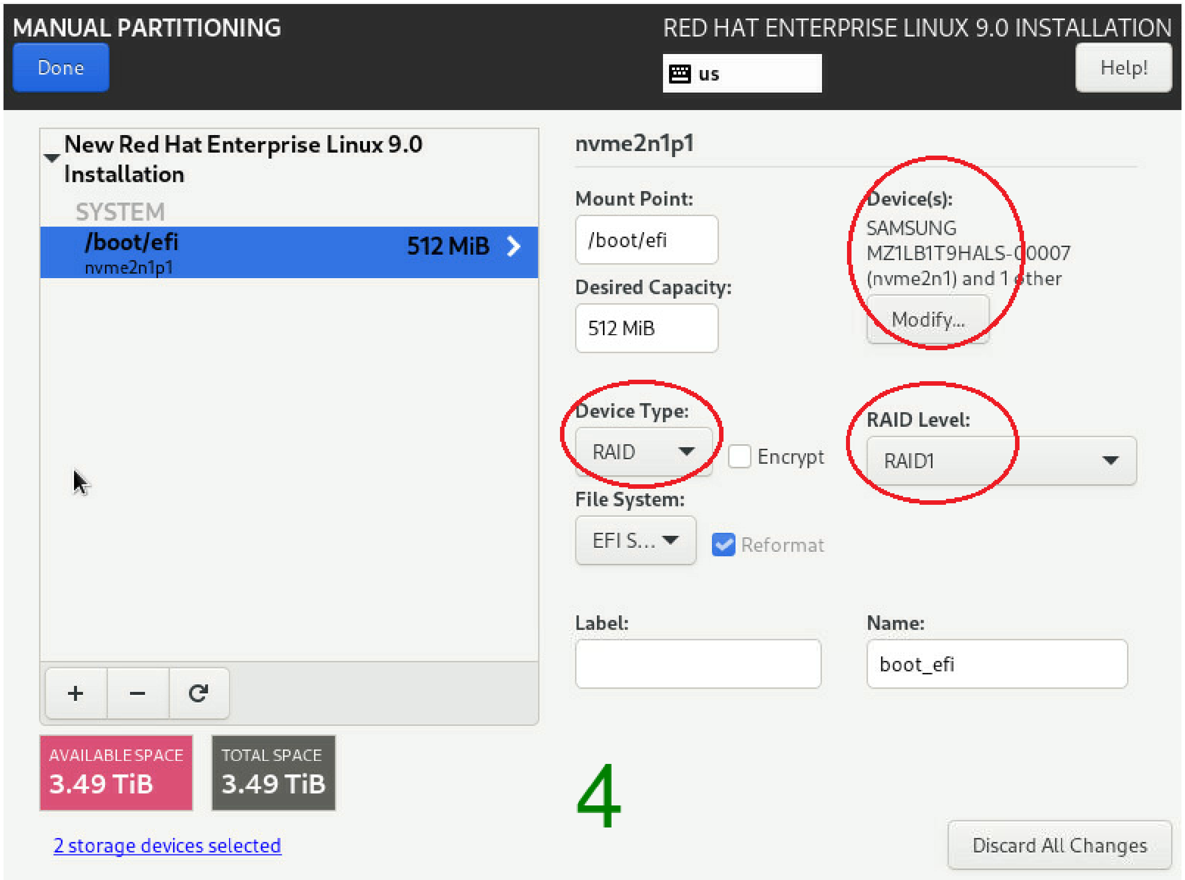
More customization options are available after creating the mount point below.

Mount Point: /boot/efi

Desired Capacity: 512 MiB

Cancel | Add mount point

4. Select Standard Partition for the Device Type.



New Red Hat Enterprise Linux 9.0 Installation

SYSTEM

/boot/efi 512 MiB
nvme2n1p1

AVAILABLE SPACE 3.49 TiB

TOTAL SPACE 3.49 TiB

2 storage devices selected

nvme2n1p1

Mount Point: /boot/efi

Desired Capacity: 512 MiB

Device(s): SAMSUNG MZ1LB1T9HALS-00007 (nvme2n1) and 1 other

Modify...

Device Type: RAID

□ Encrypt

RAID Level: RAID1

File System: EFI S...

☑ Reformat

Label:

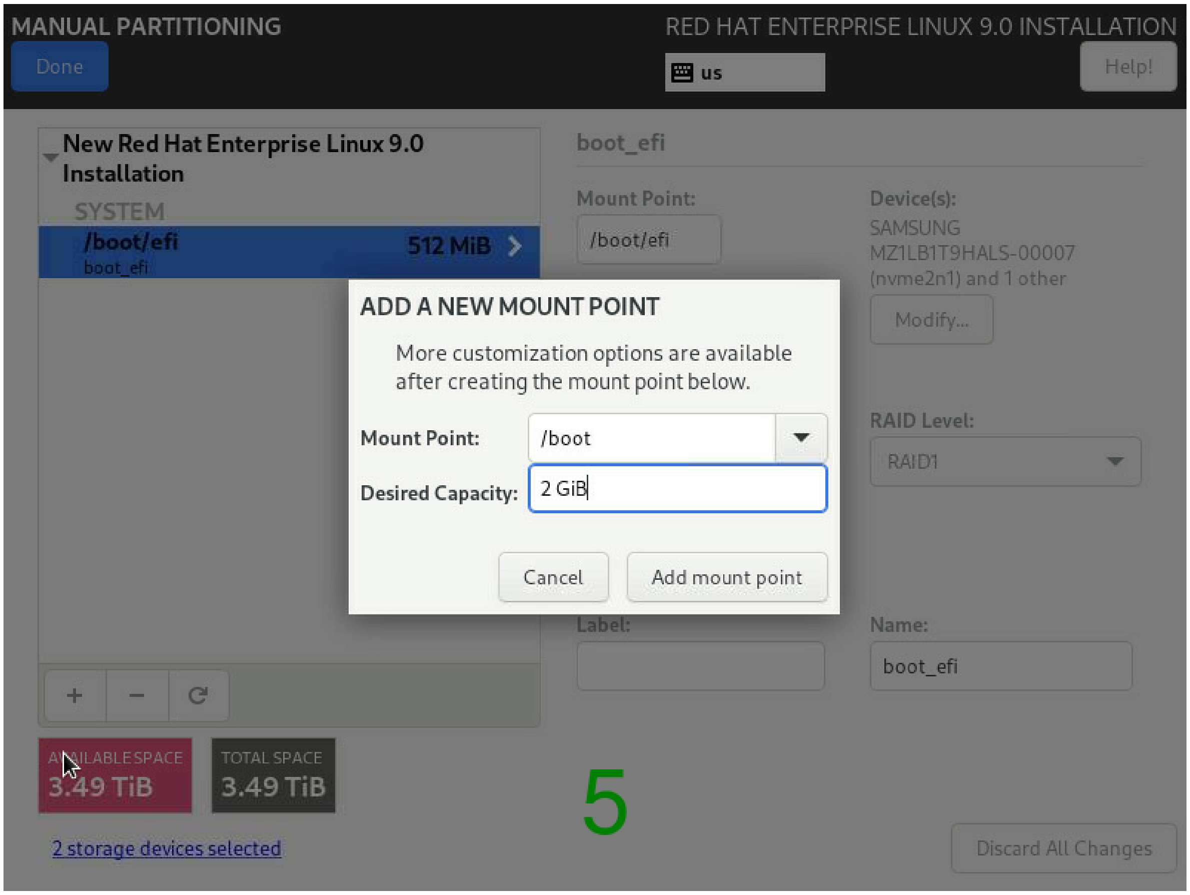
Name: boot_efi

Under **Devices**, ensure only the drive you chose in step 1 is selected.

Click **Update Settings** to confirm your changes.

5. Click the **+** button on the bottom left again to create another new partition.

   Create the /boot partition with a size of 2 GiB.



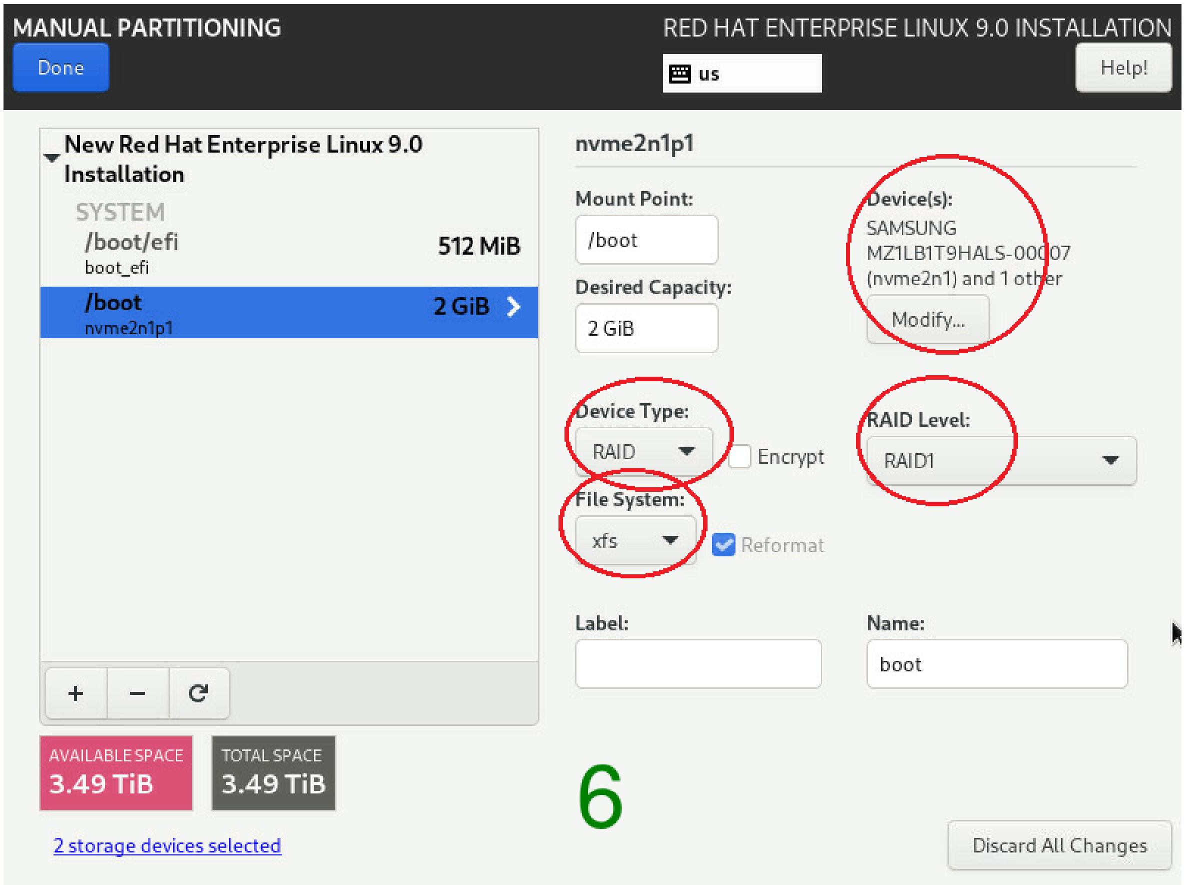
ADD A NEW MOUNT POINT

More customization options are available after creating the mount point below.

Mount Point: /boot

Desired Capacity: 2 GiB

6. For the /boot partition, again select Standard Partition for the Device Type.



New Red Hat Enterprise Linux 9.0 Installation

SYSTEM

/boot/efi boot_efi	512 MiB
/boot nvme2n1p1	2 GiB

AVAILABLE SPACE 3.49 TiB

TOTAL SPACE 3.49 TiB

2 storage devices selected

nvme2n1p1

Mount Point: /boot

Desired Capacity: 2 GiB

Device(s): SAMSUNG MZ1LB1T9HALS-00007 (nvme2n1) and 1 other

Device Type: RAID

□ Encrypt

File System: xfs

☑ Reformat

RAID Level: RAID1

Label:

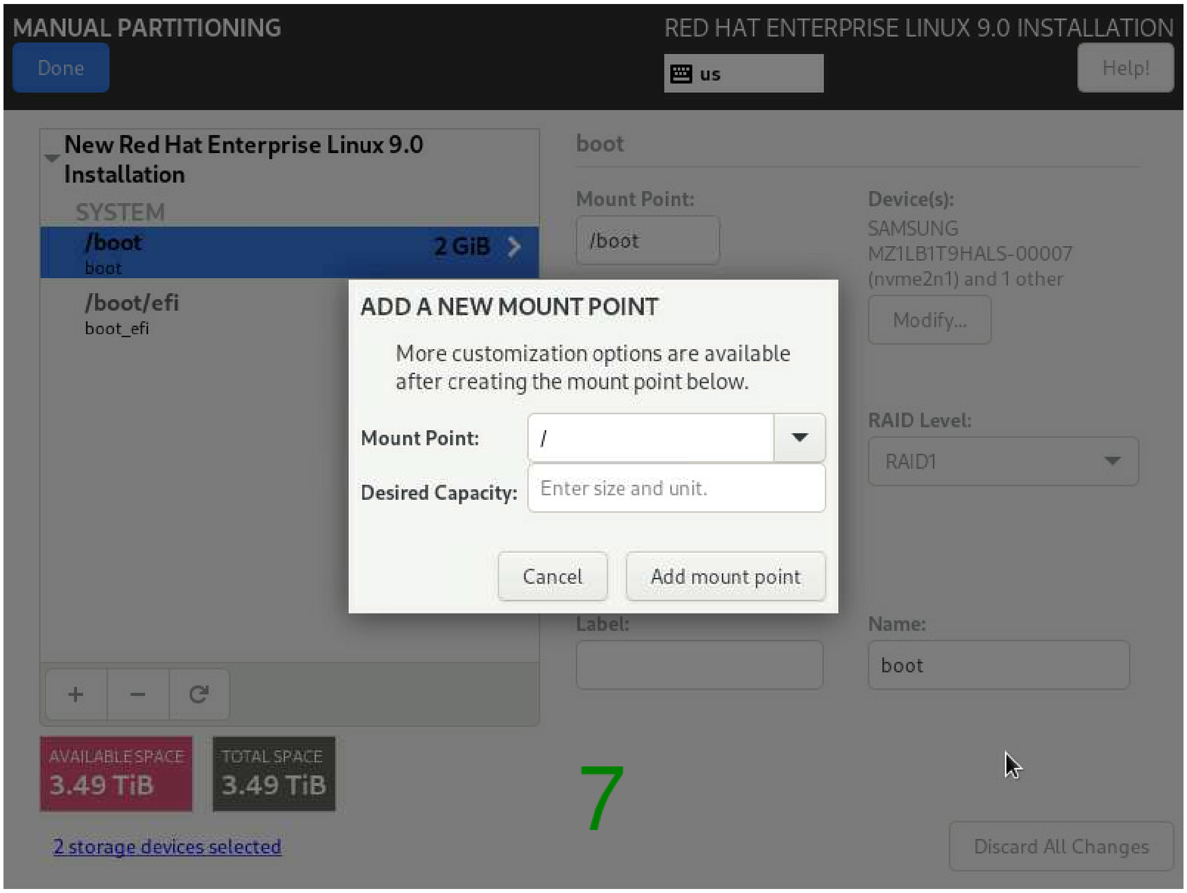
Name: boot

Under **Devices**, ensure that only the drive you chose in step 1 is selected. For the **File System**, select **XFS**.

Click Update Settings to confirm your changes.

7. Click the + button on the bottom left again to create another new partition.

   Create the **/** partition, this time leaving the **Desired Capacity** field empty. This lets the installer know to use the remaining capacity of the disks.



Mount Point: /boot

Device(s): SAMSUNG MZ1LB1T9HALS-00007 (nvme2n1) and 1 other

Modify...

RAID Level: RAID1

Label:

Name: boot

ADD A NEW MOUNT POINT

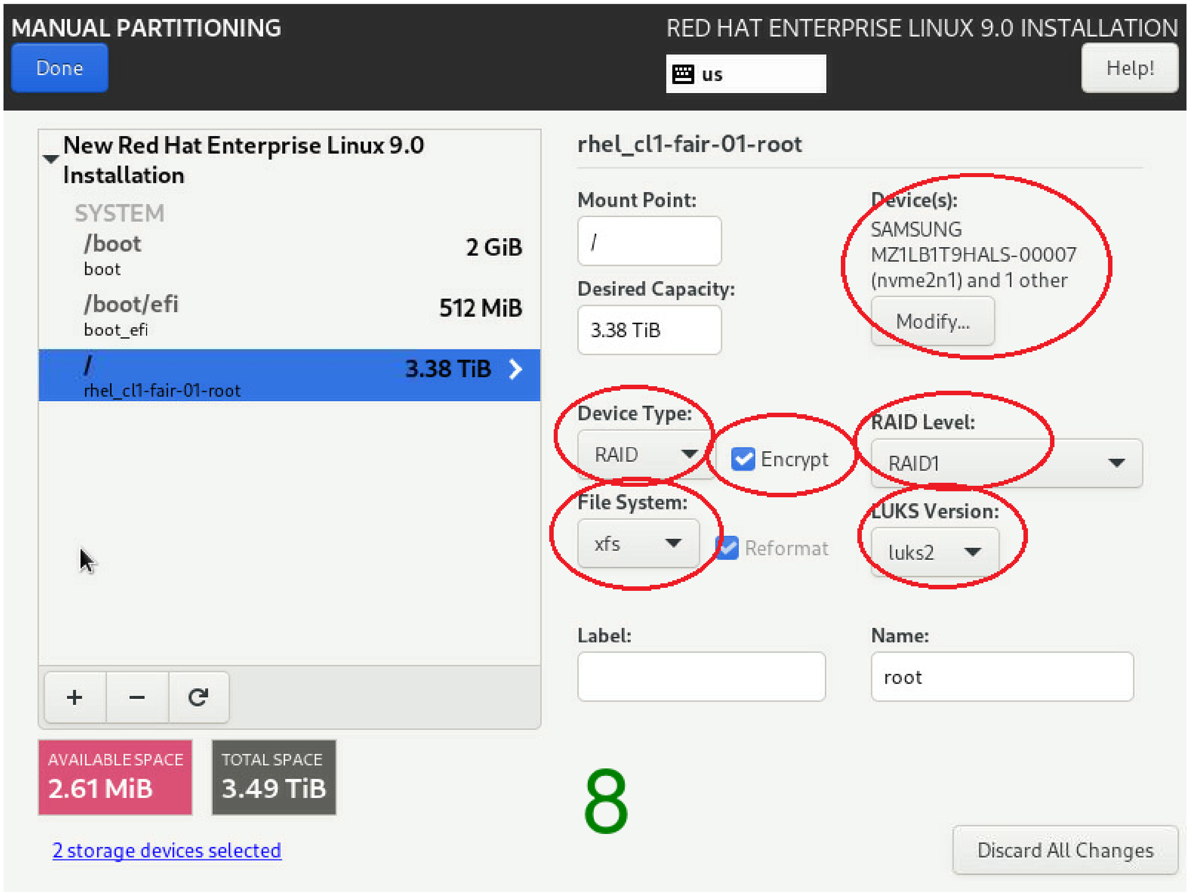
More customization options are available after creating the mount point below.

Mount Point: /

Desired Capacity: Enter size and unit.

Cancel | Add mount point

8. For the **/** partition, again select **Standard Partition** for the **Device Type**. Be sure to check the **Encrypt** checkbox, and choose **luks2** for the **LUKS Version**. For the **File System**, select **XFS.** Again confirm that this partition uses both of the devices you selected in step 2. Then click **Done**.



MANUAL PARTITIONING

RED HAT ENTERPRISE LINUX 9.0 INSTALLATION

New Red Hat Enterprise Linux 9.0 Installation

SYSTEM

/boot boot	2 GiB
/boot/efi boot_efi	512 MiB
/ rhel_cl1-fair-01-root	3.38 TiB

AVAILABLE SPACE 2.61 MiB

TOTAL SPACE 3.49 TiB

2 storage devices selected

rhel_cl1-fair-01-root

Mount Point: /

Desired Capacity: 3.38 TiB

Device(s): SAMSUNG MZ1LB1T9HALS-00007 (nvme2n1) and 1 other

Device Type: RAID

☑ Encrypt

RAID Level: RAID1

File System: xfs

☑ Reformat

LUKS Version: luks2

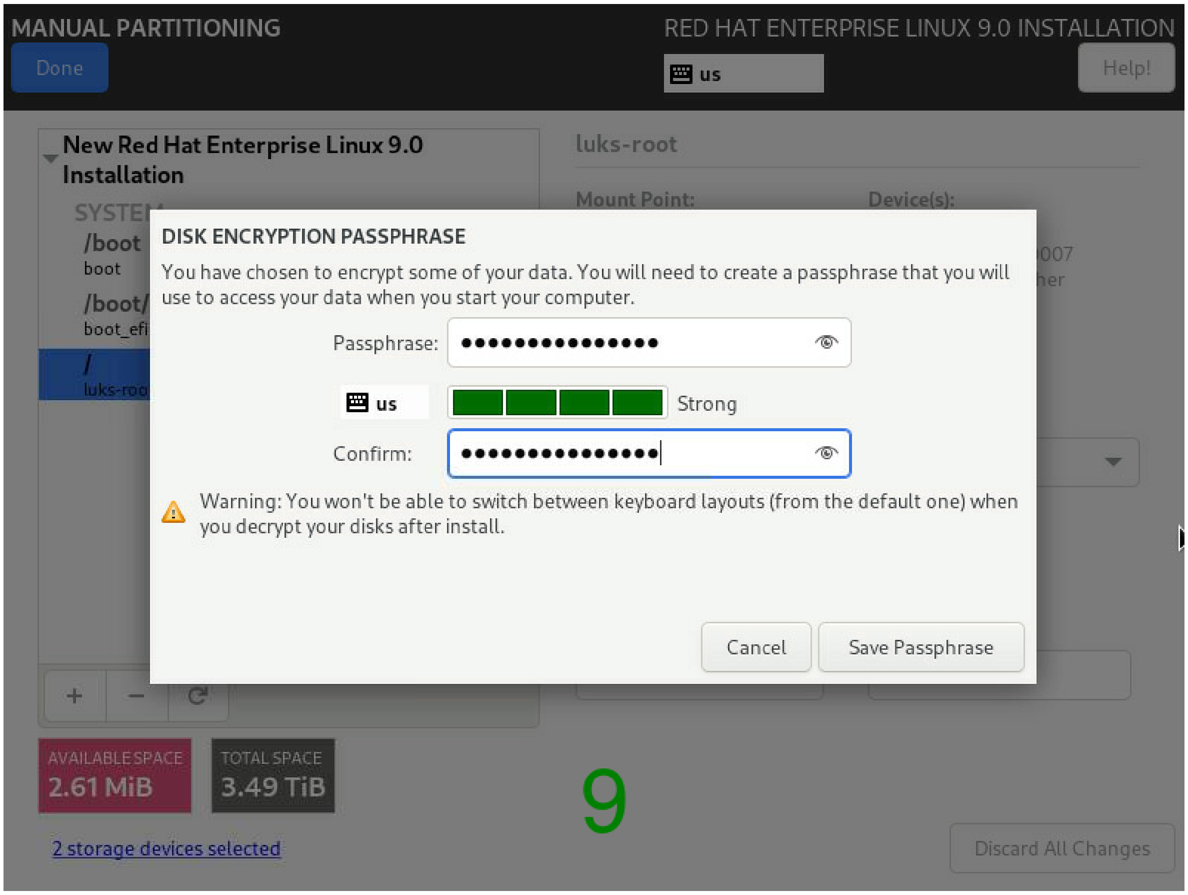
Label:

Name: root

8

9. A prompt will now appear, asking for a LUKS password. Be sure to choose a strong and secure password. After entering your password twice, click **Save Passphrase**

10. Click **Done**. This causes a yellow warning bar to appear on the bottom because no swap partition has been created.



DISK ENCRYPTION PASSPHRASE

You have chosen to encrypt some of your data. You will need to create a passphrase that you will use to access your data when you start your computer.

Passphrase: ●●●●●●●●●●●●●●●

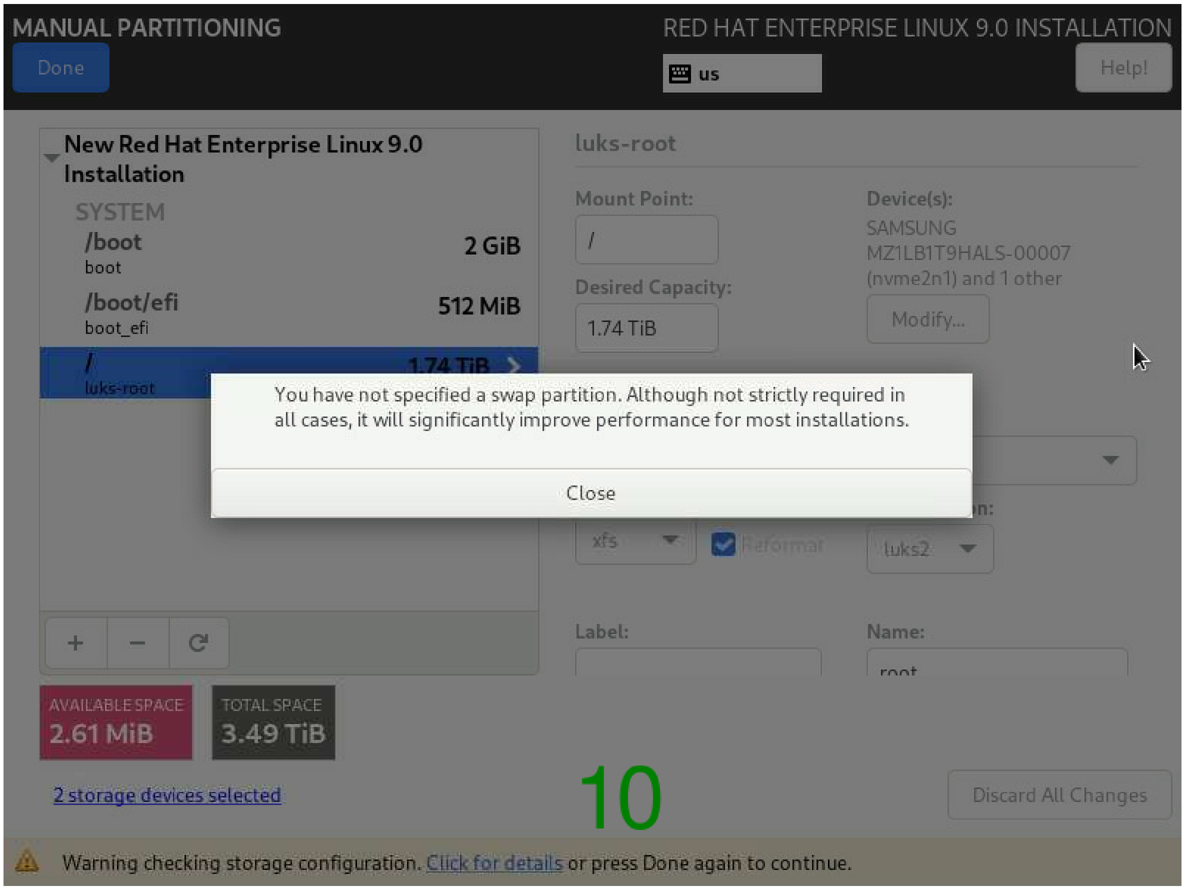
us Strong

Confirm: ●●●●●●●●●●●●●●●

Warning: You won't be able to switch between keyboard layouts (from the default one) when you decrypt your disks after install.

Cancel Save Passphrase

11. Click **Done** again and then click **Accept Changes** to write all of our customizations to disk.


MANUAL PARTITIONING
Done
RED HAT ENTERPRISE LINUX 9.0 INSTALLATION
us
Help!
New Red Hat Enterprise Linux 9.0 Installation
SYSTEM
+
−
Modify...
Close
Discard All Changes
2 storage devices selected
10

/boot 2 GiB
boot
/boot/efi 512 MiB
boot_efi
/ 1.74 TiB
luks-root

luks-root
Mount Point: /
Desired Capacity: 1.74 TiB
Device(s): SAMSUNG MZ1LBT9HALS-00007 (nvme2n1) and 1 other
xfs ☑ Reformat
luks2
Label:
Name: root

AVAILABLE SPACE 2.61 MiB
TOTAL SPACE 3.49 TiB

You have not specified a swap partition. Although not strictly required in all cases, it will significantly improve performance for most installations.

Warning checking storage configuration. Click for details or press Done again to continue.

12. Return to step 8 in the *Installing and Configuring Red Hat Enterprise Linux* section.

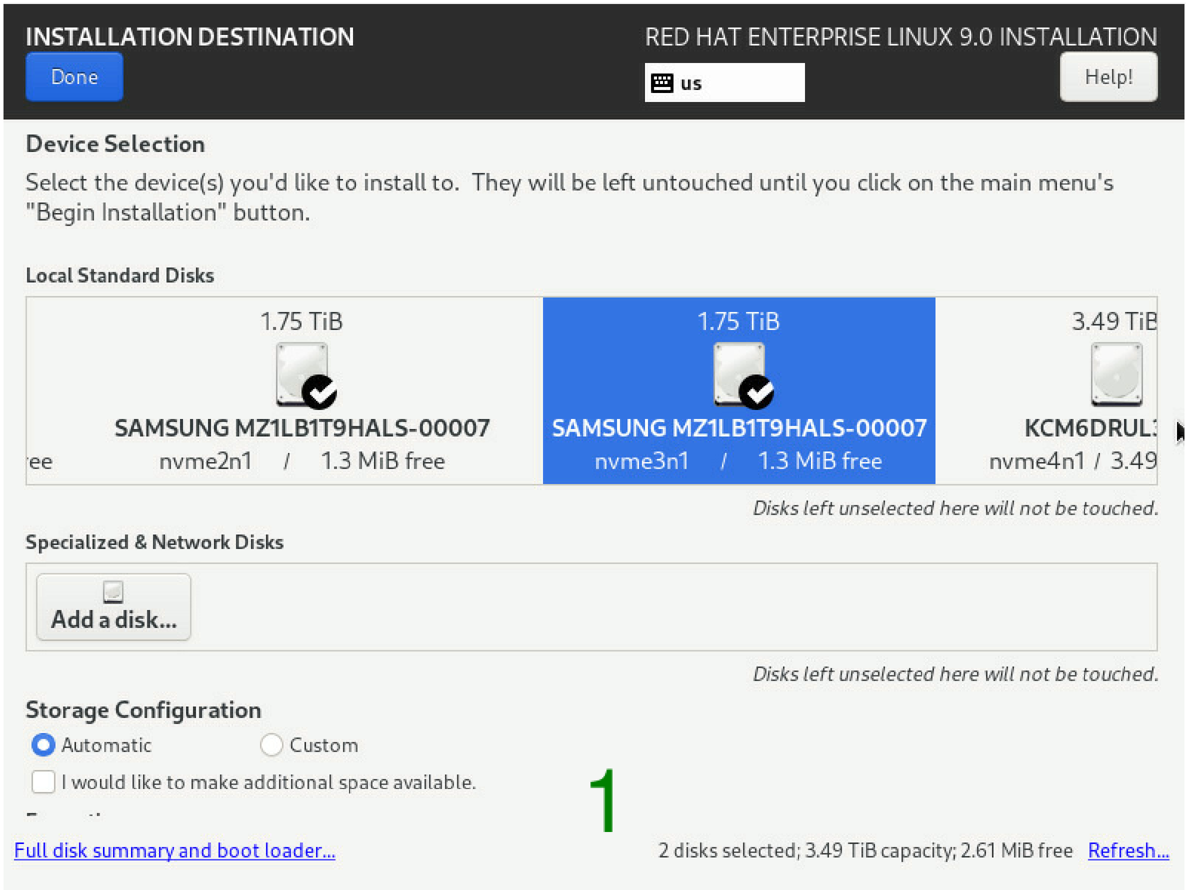
3.5.3. Disk Partitioning for DGX-2, DGX A100, DGX A800, and DGX H100/H200

Note

The screenshots in the following section are taken from a DGX A100. The instructions for the DGX-2, DGX H100/H200, and DGX A800 installation can present slightly different information, such as disk size, disk space available, interface names, and so on.

1. At the **Installation Destination** screen, select the smaller of the available drives to install on.

   Under **Storage Configuration**, click the **Custom** radio button and then click **Done**.



INSTALLATION DESTINATION

RED HAT ENTERPRISE LINUX 9.0 INSTALLATION

Device Selection

Select the device(s) you'd like to install to. They will be left untouched until you click on the main menu's "Begin Installation" button.

Local Standard Disks

1.75 TiB	1.75 TiB	3.49 TiB
SAMSUNG MZ1LB1T9HALS-00007	SAMSUNG MZ1LB1T9HALS-00007	KCM6DRUL3
nvme2n1 / 1.3 MiB free	nvme3n1 / 1.3 MiB free	nvme4n1 / 3.49

Disks left unselected here will not be touched.

Specialized & Network Disks

Disks left unselected here will not be touched.

Storage Configuration

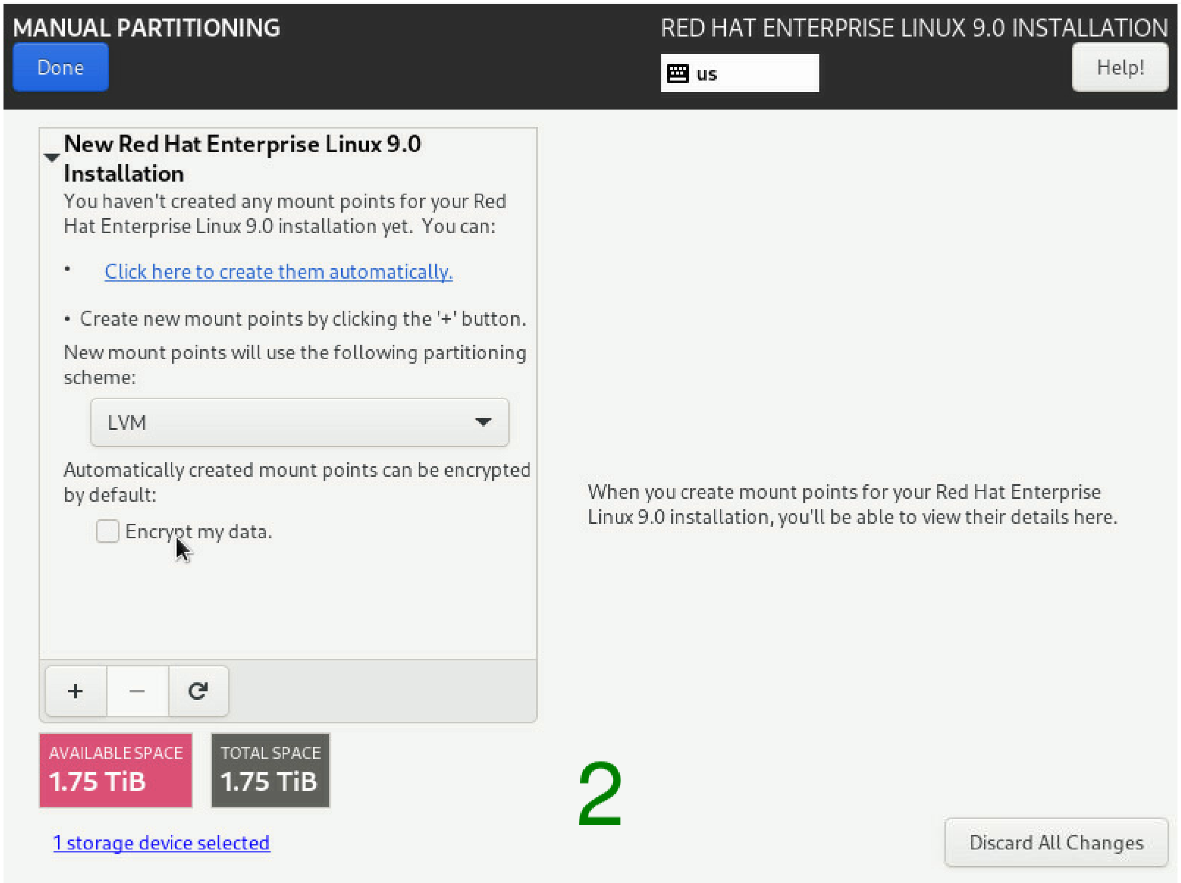
● Automatic ○ Custom

□ I would like to make additional space available.

2 disks selected; 3.49 TiB capacity; 2.61 MiB free

This brings up the **Manual Partitioning** window.

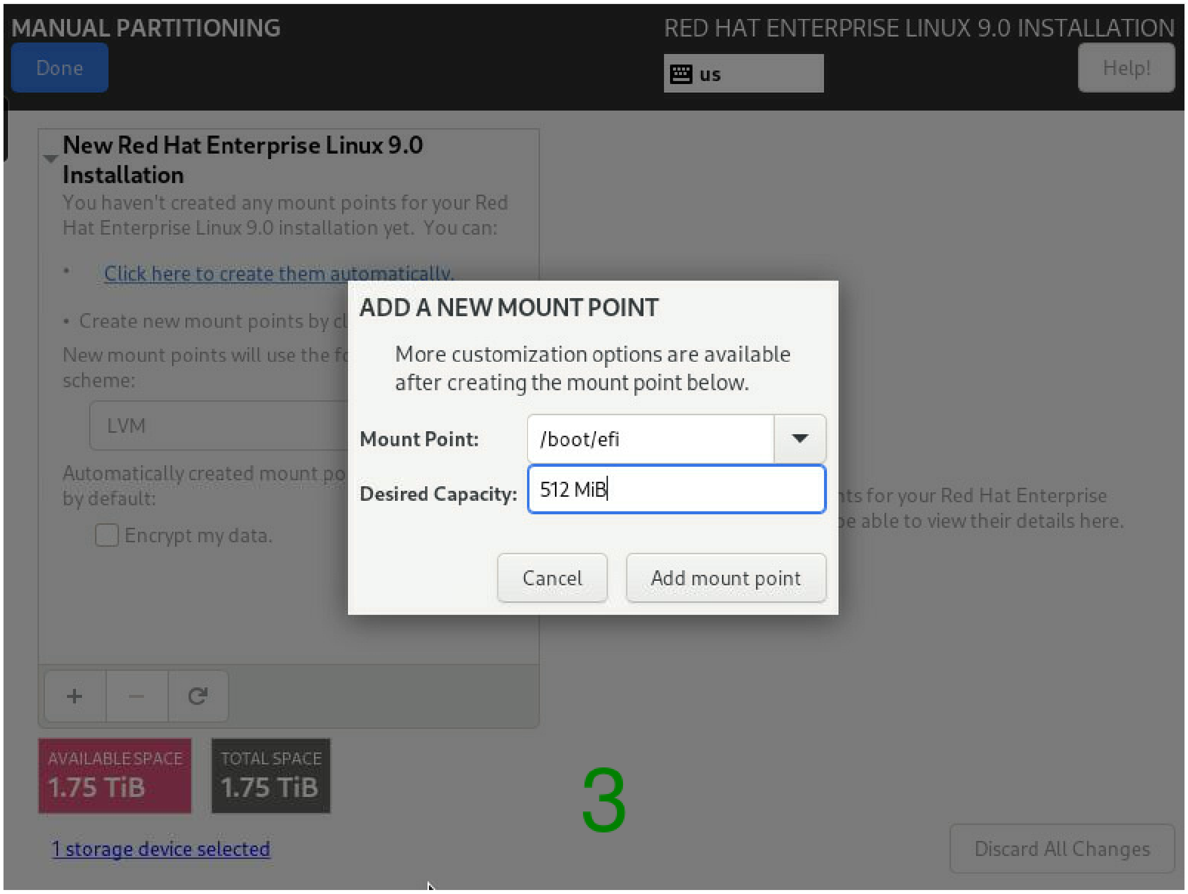
2. Expand the drop-down menu for the device you have selected and delete all existing partitions until there are none remaining.



MANUAL PARTITIONING
Done
RED HAT ENTERPRISE LINUX 9.0 INSTALLATION
us
Help!
New Red Hat Enterprise Linux 9.0 Installation
You haven't created any mount points for your Red Hat Enterprise Linux 9.0 installation yet. You can:
Click here to create them automatically.
Create new mount points by clicking the '+' button.
New mount points will use the following partitioning scheme:
LVM
Automatically created mount points can be encrypted by default:
Encrypt my data.
+
–
AVAILABLE SPACE 1.75 TiB
TOTAL SPACE 1.75 TiB
1 storage device selected
When you create mount points for your Red Hat Enterprise Linux 9.0 installation, you'll be able to view their details here.
2
Discard All Changes

3. Click the **+** button on the bottom left to create a new partition.

   Create the `/boot/efi` partition with a size of 512 MiB.



MANUAL PARTITIONING

RED HAT ENTERPRISE LINUX 9.0 INSTALLATION

New Red Hat Enterprise Linux 9.0 Installation

You haven't created any mount points for your Red Hat Enterprise Linux 9.0 installation yet. You can:

ADD A NEW MOUNT POINT

More customization options are available after creating the mount point below.

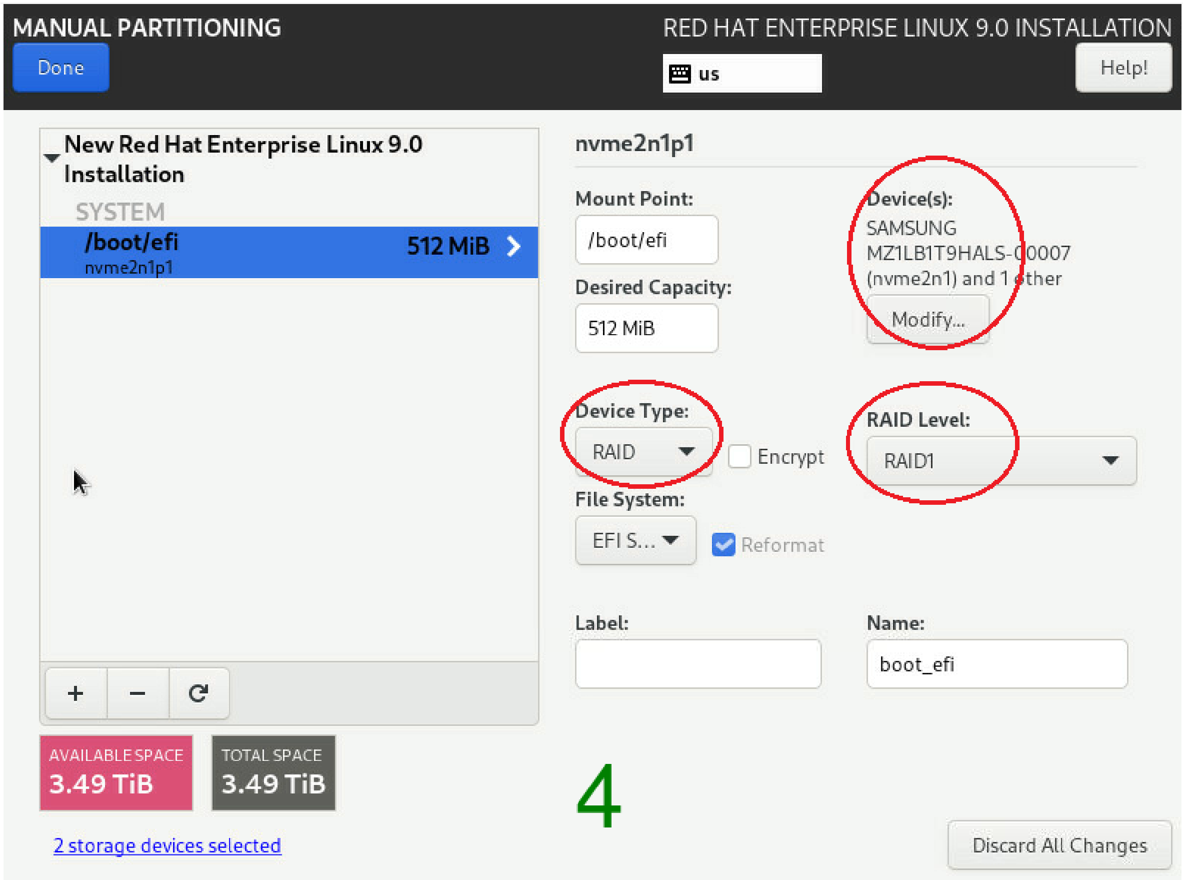
Mount Point: /boot/efi

Desired Capacity: 512 MiB

Done
us
Help!
Click here to create them automatically.
Create new mount points by cl
New mount points will use the fo
scheme:
LVM
Automatically created mount po
by default:
Encrypt my data.
Cancel
Add mount point
nts for your Red Hat Enterprise
be able to view their details here.
+
−
AVAILABLE SPACE
1.75 TiB
TOTAL SPACE
1.75 TiB
3
1 storage device selected
Discard All Changes

4. Change the **Device Type** to **RAID,** change the **RAID Level** to **RAID1**, and confirm that this partition covers both of the devices you selected in step 2 by clicking **Modify** in the **Device(s)** section and making sure both disks are still selected.

   Note that the partition information on the left might not reflect both devices yet. Click **Update Settings** to confirm your changes.


MANUAL PARTITIONING
Done
RED HAT ENTERPRISE LINUX 9.0 INSTALLATION
us
Help!
Modify...
+
−
2 storage devices selected
Discard All Changes
4

New Red Hat Enterprise Linux 9.0 Installation

SYSTEM

/boot/efi 512 MiB
nvme2n1p1

AVAILABLE SPACE 3.49 TiB

TOTAL SPACE 3.49 TiB

nvme2n1p1

Mount Point: /boot/efi

Desired Capacity: 512 MiB

Device(s): SAMSUNG MZ1LB1T9HALS-00007 (nvme2n1) and 1 other

Device Type: RAID

□ Encrypt

File System: EFI S...

☑ Reformat

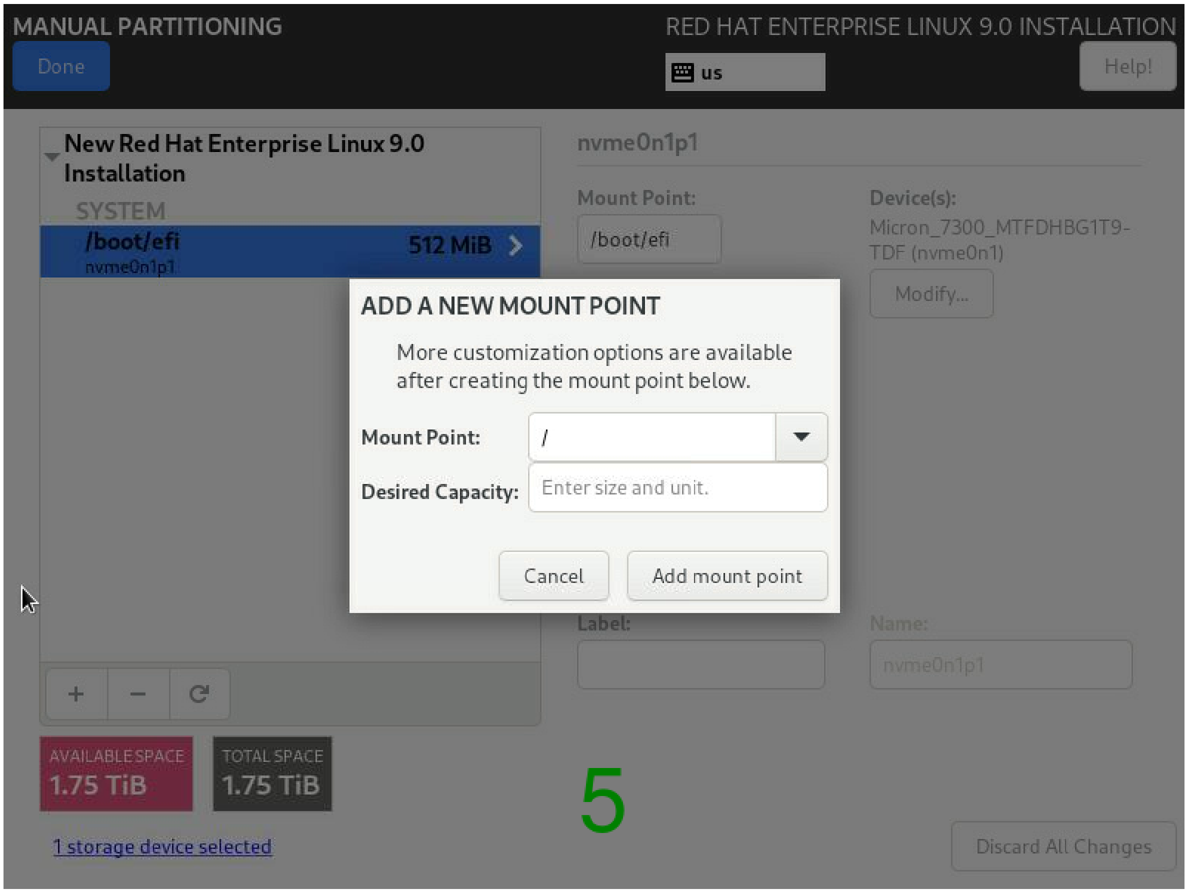
RAID Level: RAID1

Label:

Name: boot_efi

5. Click the **+** button on the bottom left again to create another new partition.

   Create the **/** partition, this time leaving the Desired Capacity field empty. This lets the installer know to use the remaining capacity of the disks.



Mount Point: /boot/efi

Name: nvme0n1p1

ADD A NEW MOUNT POINT

More customization options are available after creating the mount point below.

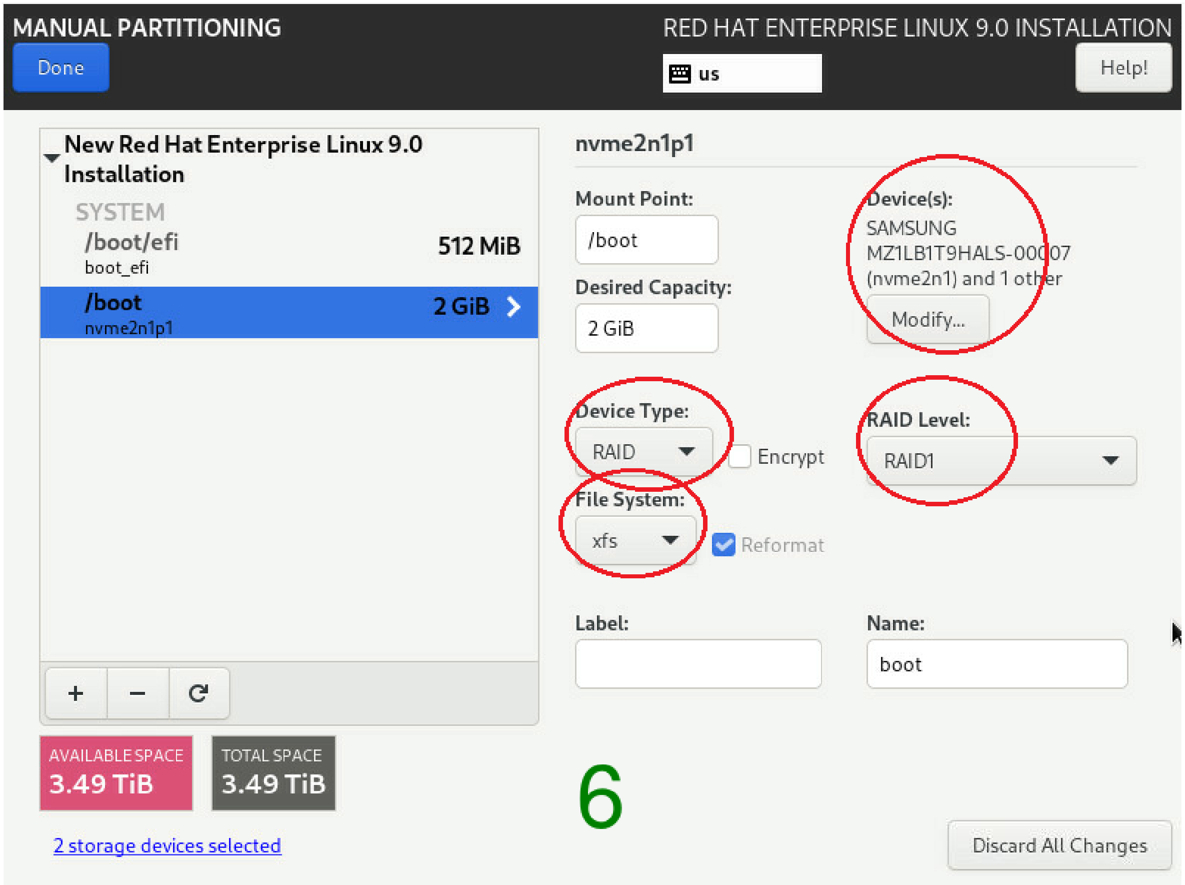
Mount Point: /

Desired Capacity: Enter size and unit.

5

6. For the **/** partition, again select **RAID** for the **Device Type** and **RAID1** for the **RAID Level**. For the **File System**, select **XFS**.

   Confirm that this partition uses both of the devices you selected in step 2 by clicking **Modify** in the **Device(s)** section and making sure both disks are still selected.



nvme2n1p1

Mount Point: /boot

Desired Capacity: 2 GiB

Device(s): SAMSUNG MZ1LB1T9HALS-00007 (nvme2n1) and 1 other

Device Type: RAID

□ Encrypt

RAID Level: RAID1

File System: xfs

☑ Reformat

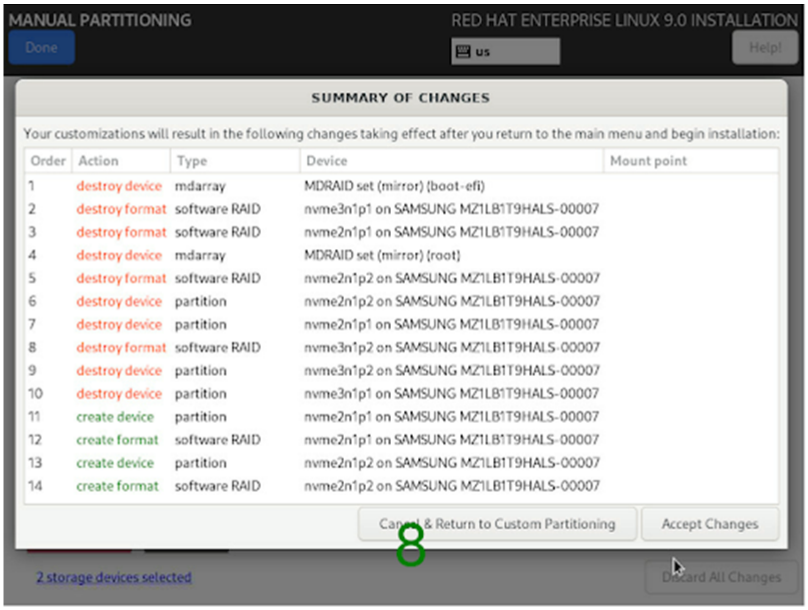
Label:

Name: boot

7. Click **Done**.

   This causes a yellow warning bar to appear on the bottom because no swap partition has been created.

8. Click **Done** again and then click **Accept Changes** to write all of our customizations to disk.


MANUAL PARTITIONING
Done
RED HAT ENTERPRISE LINUX 9.0 INSTALLATION
us
Help!
Cancel & Return to Custom Partitioning
Accept Changes
2 storage devices selected
Discard All Changes
8

SUMMARY OF CHANGES

Your customizations will result in the following changes taking effect after you return to the main menu and begin installation:

Order	Action	Type	Device	Mount point
1	destroy device	mdarray	MDRAID set (mirror) (boot-efi)	
2	destroy format	software RAID	nvme3n1p1 on SAMSUNG MZ1LB1T9HALS-00007	
3	destroy format	software RAID	nvme2n1p1 on SAMSUNG MZ1LB1T9HALS-00007	
4	destroy device	mdarray	MDRAID set (mirror) (root)	
5	destroy format	software RAID	nvme2n1p2 on SAMSUNG MZ1LB1T9HALS-00007	
6	destroy device	partition	nvme2n1p2 on SAMSUNG MZ1LB1T9HALS-00007	
7	destroy device	partition	nvme2n1p1 on SAMSUNG MZ1LB1T9HALS-00007	
8	destroy format	software RAID	nvme3n1p2 on SAMSUNG MZ1LB1T9HALS-00007	
9	destroy device	partition	nvme3n1p2 on SAMSUNG MZ1LB1T9HALS-00007	
10	destroy device	partition	nvme3n1p1 on SAMSUNG MZ1LB1T9HALS-00007	
11	create device	partition	nvme2n1p1 on SAMSUNG MZ1LB1T9HALS-00007	
12	create format	software RAID	nvme2n1p1 on SAMSUNG MZ1LB1T9HALS-00007	
13	create device	partition	nvme2n1p2 on SAMSUNG MZ1LB1T9HALS-00007	
14	create format	software RAID	nvme2n1p2 on SAMSUNG MZ1LB1T9HALS-00007	

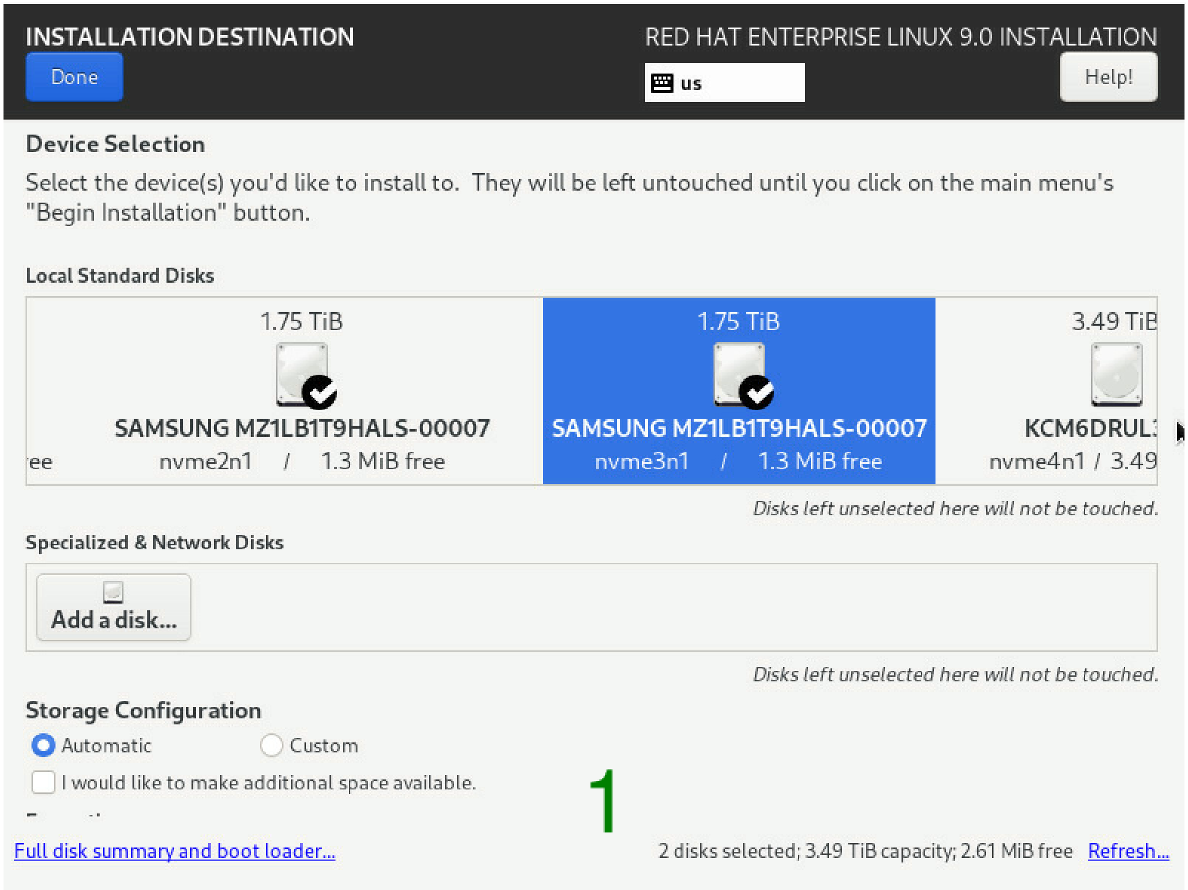
9. Return to step 8 in the *Installing and Configuring Red Hat Enterprise Linux* section.

3.5.4. Disk Partitioning with Encryption for DGX-2, DGX A100, DGX A800, and DGX H100/H200

Note

The screenshots in the following section are taken from a DGX A100. The following instructions will use LUKS to create an encrypted root filesystem on your boot devices. This is a separate process from using self-encrypting data-drives.

1. At the Installation Destination screen, *select the smaller* of the available drives to install on.



INSTALLATION DESTINATION

RED HAT ENTERPRISE LINUX 9.0 INSTALLATION

Device Selection

Select the device(s) you'd like to install to. They will be left untouched until you click on the main menu's "Begin Installation" button.

Local Standard Disks

ee	1.75 TiB SAMSUNG MZ1LB1T9HALS-00007 nvme2n1 / 1.3 MiB free	1.75 TiB SAMSUNG MZ1LB1T9HALS-00007 nvme3n1 / 1.3 MiB free	3.49 TiB KCM6DRUL: nvme4n1 / 3.49

Disks left unselected here will not be touched.

Specialized & Network Disks

Disks left unselected here will not be touched.

Storage Configuration

● Automatic ○ Custom

□ I would like to make additional space available.

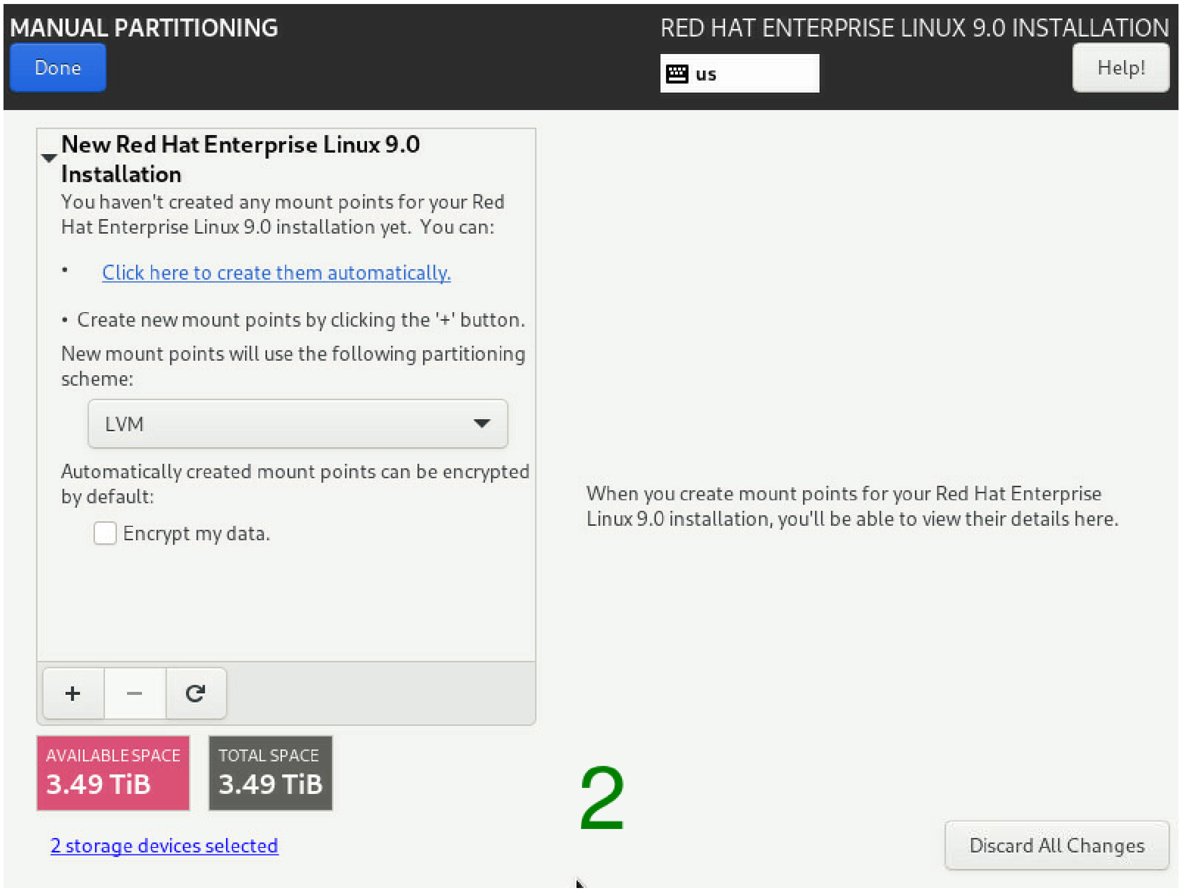
Full disk summary and boot loader...

2 disks selected; 3.49 TiB capacity; 2.61 MiB free Refresh...

Under Storage Configuration, click the Custom radio button and then click Done.

This brings up the Manual Partitioning window.

2. Expand the drop-down menu for the device you have selected and delete all existing partitions until there are none remaining.



MANUAL PARTITIONING

RED HAT ENTERPRISE LINUX 9.0 INSTALLATION

New Red Hat Enterprise Linux 9.0 Installation

You haven't created any mount points for your Red Hat Enterprise Linux 9.0 installation yet. You can:

- Click here to create them automatically.
- Create new mount points by clicking the '+' button.

New mount points will use the following partitioning scheme:

LVM

Automatically created mount points can be encrypted by default:

□ Encrypt my data.

When you create mount points for your Red Hat Enterprise Linux 9.0 installation, you'll be able to view their details here.

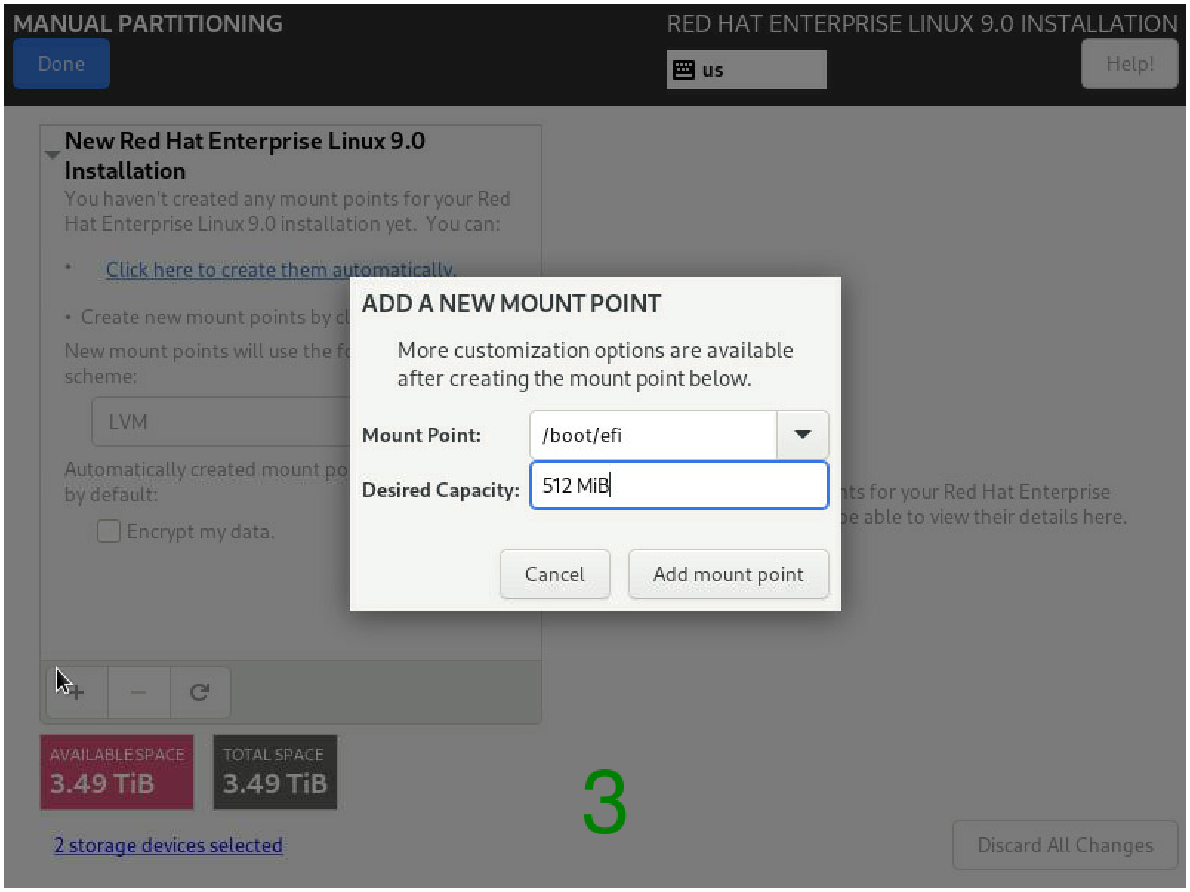
AVAILABLE SPACE 3.49 TiB

TOTAL SPACE 3.49 TiB

2 storage devices selected

3. Click the + button on the bottom left to create a new partition.

   Create the `/boot/efi`partition with a size of 512 MiB.



New Red Hat Enterprise Linux 9.0 Installation

You haven't created any mount points for your Red Hat Enterprise Linux 9.0 installation yet. You can:

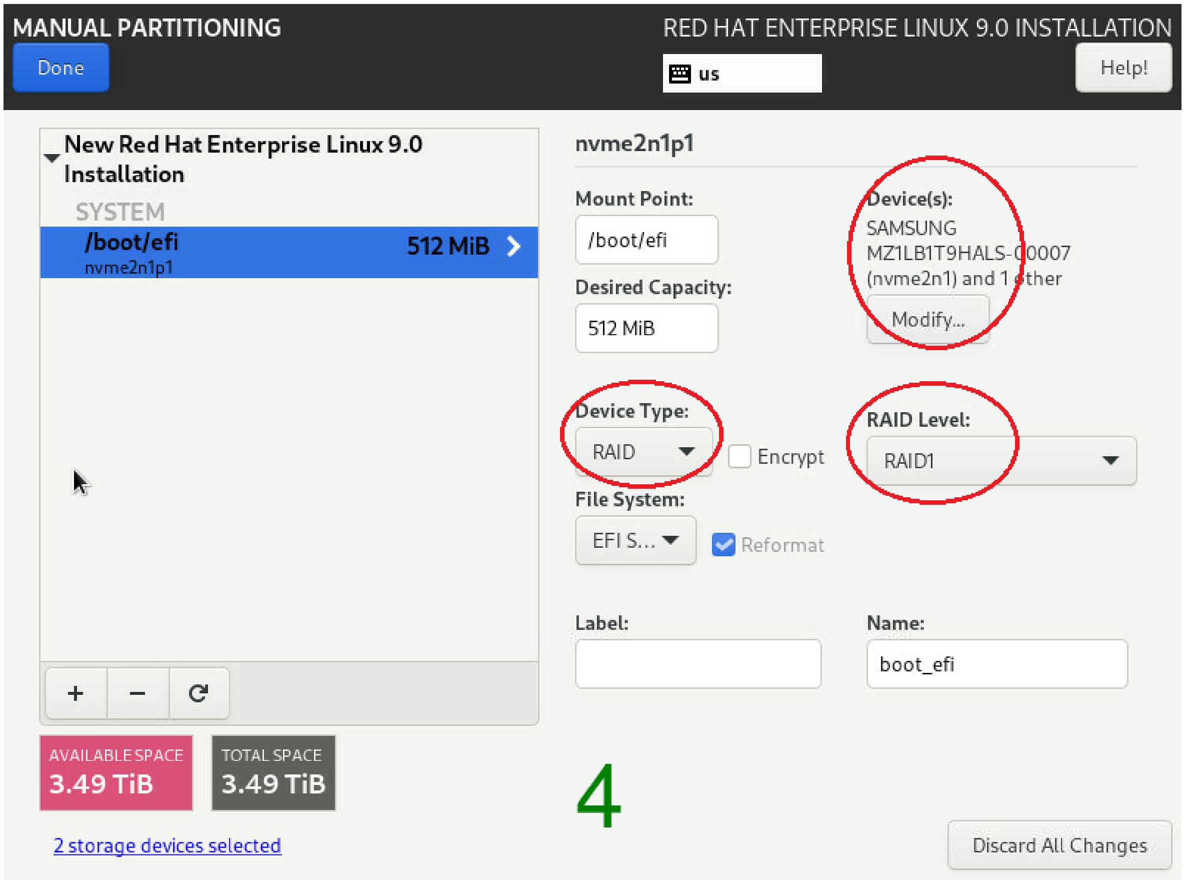
ADD A NEW MOUNT POINT

More customization options are available after creating the mount point below.

Mount Point: /boot/efi

Desired Capacity: 512 MiB

4. Change the Device Type to RAID, change the RAID Level toRAID1, and confirm that this partition covers both of the devices you selected in step 2 by clicking Modify in the Device(s) section and making sure both disks are still selected.



New Red Hat Enterprise Linux 9.0 Installation

SYSTEM

/boot/efi 512 MiB
nvme2n1p1

AVAILABLE SPACE 3.49 TiB

TOTAL SPACE 3.49 TiB

2 storage devices selected

nvme2n1p1

Mount Point: /boot/efi

Desired Capacity: 512 MiB

Device(s): SAMSUNG MZ1LB1T9HALS-00007 (nvme2n1) and 1 other

Modify...

Device Type: RAID

□ Encrypt

RAID Level: RAID1

File System: EFI S...

☑ Reformat

Label:

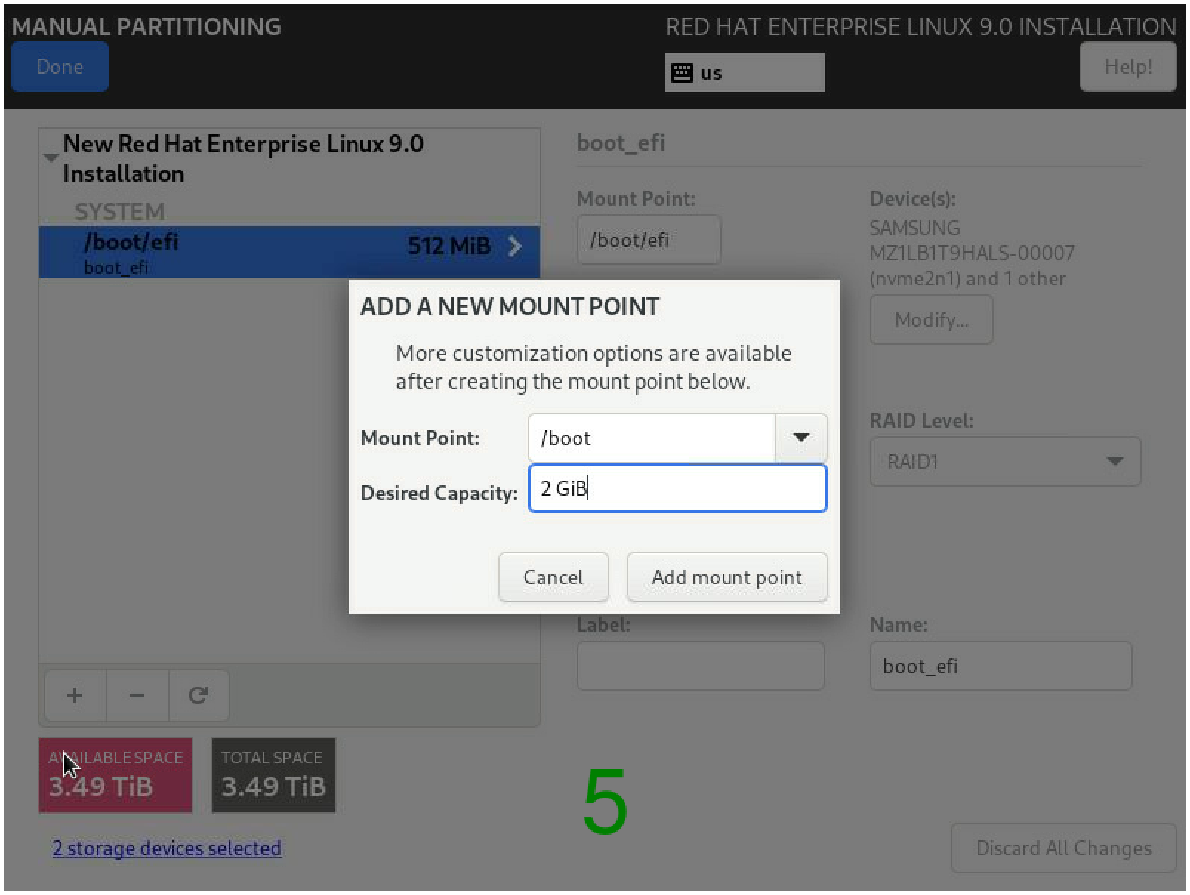
Name: boot_efi

Note that the partition information on the left might not reflect both devices yet.

Click Update Settings to confirm your changes, and verify that the `/boot/efi` partition shown lists the partition name and not the individual device.

5. Click the + button on the bottom left again to create another new partition.

   Create the /boot partition with a size of 2 GiB.



boot_efi

Mount Point: /boot/efi

Device(s): SAMSUNG MZ1LB1T9HALS-00007 (nvme2n1) and 1 other

RAID Level: RAID1

Name: boot_efi

ADD A NEW MOUNT POINT

More customization options are available after creating the mount point below.

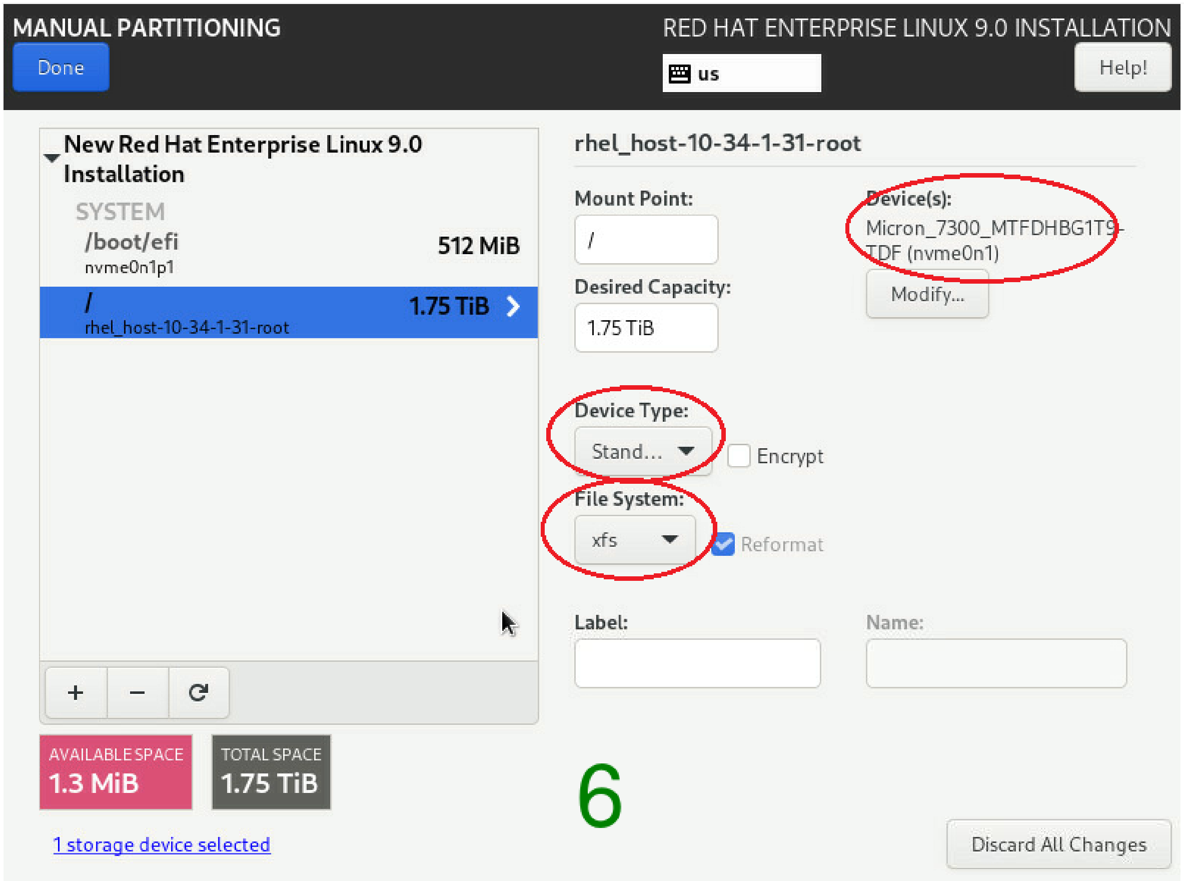
Mount Point: /boot

Desired Capacity: 2 GiB

5

6. For the /boot partition, again select RAID for the Device Type, RAID1, and the RAID Level.

   For the File System, select XFS.



MANUAL PARTITIONING

RED HAT ENTERPRISE LINUX 9.0 INSTALLATION

New Red Hat Enterprise Linux 9.0 Installation

SYSTEM

/boot/efi 512 MiB
nvme0n1p1

/ 1.75 TiB
rhel_host-10-34-1-31-root

rhel_host-10-34-1-31-root

Mount Point: /

Desired Capacity: 1.75 TiB

Device(s): Micron_7300_MTFDHBG1T9-TDF (nvme0n1)

Device Type: Stand…

□ Encrypt

File System: xfs

☑ Reformat

Label:

Name:

AVAILABLE SPACE 1.3 MiB

TOTAL SPACE 1.75 TiB

1 storage device selected

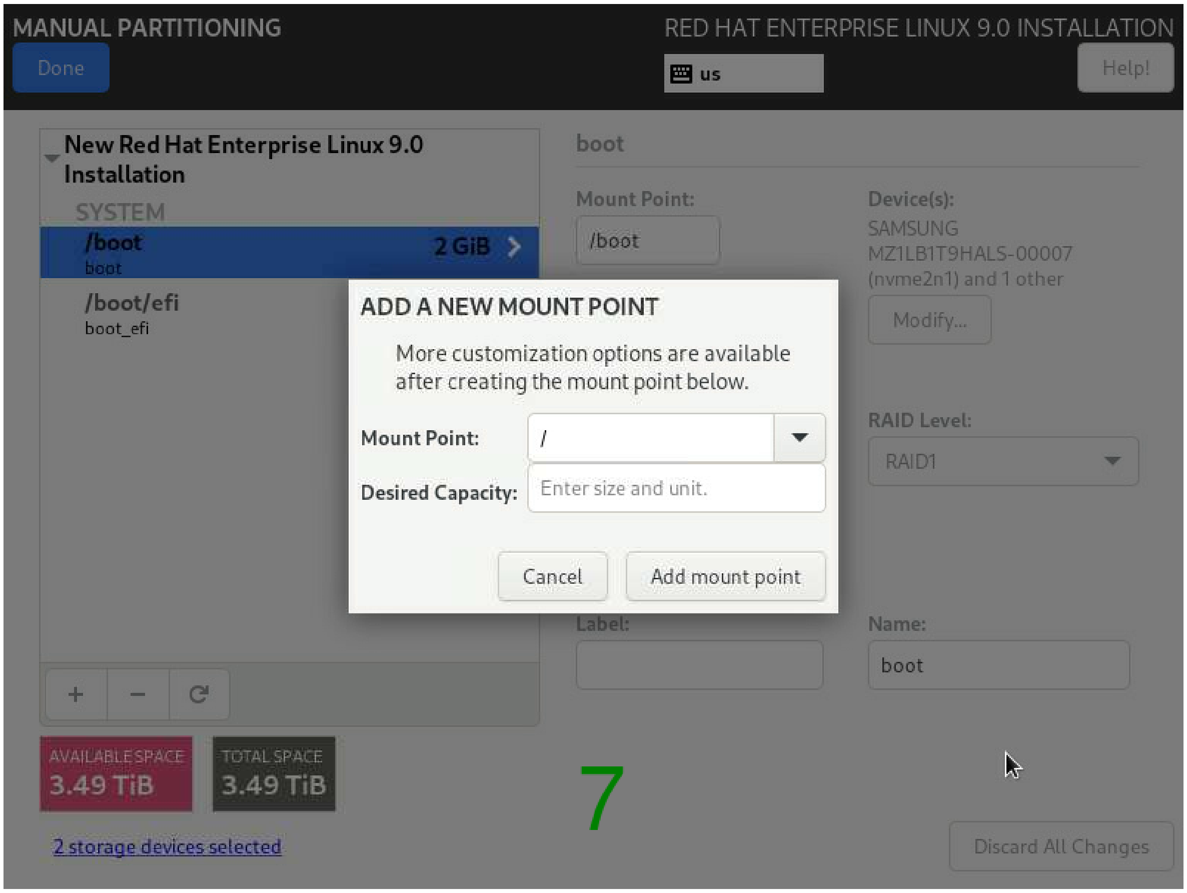
Again confirm that this partition uses both of the devices you selected in step 2 by clicking **Modify** in the **Device(s)** section and making sure both disks are still selected.

Click Update Settings to confirm your changes. Click the + button on the bottom left again to create another new partition.

Create the / partition, this time leaving the Desired Capacity field empty. This lets the installer know to use the remaining capacity of the disks.

7. For the / partition, again select RAID for the Device Type, RAID1 and the RAID Level.

   Be sure to check the Encrypt checkbox, and choose luks2 for the LUKS Version.



MANUAL PARTITIONING

RED HAT ENTERPRISE LINUX 9.0 INSTALLATION

New Red Hat Enterprise Linux 9.0 Installation

SYSTEM

/boot 2 GiB
boot

/boot/efi
boot_efi

boot

Mount Point: /boot

Device(s): SAMSUNG MZ1LB1T9HALS-00007 (nvme2n1) and 1 other

RAID Level: RAID1

Label:

Name: boot

AVAILABLE SPACE 3.49 TiB

TOTAL SPACE 3.49 TiB

2 storage devices selected

ADD A NEW MOUNT POINT

More customization options are available after creating the mount point below.

Mount Point: /

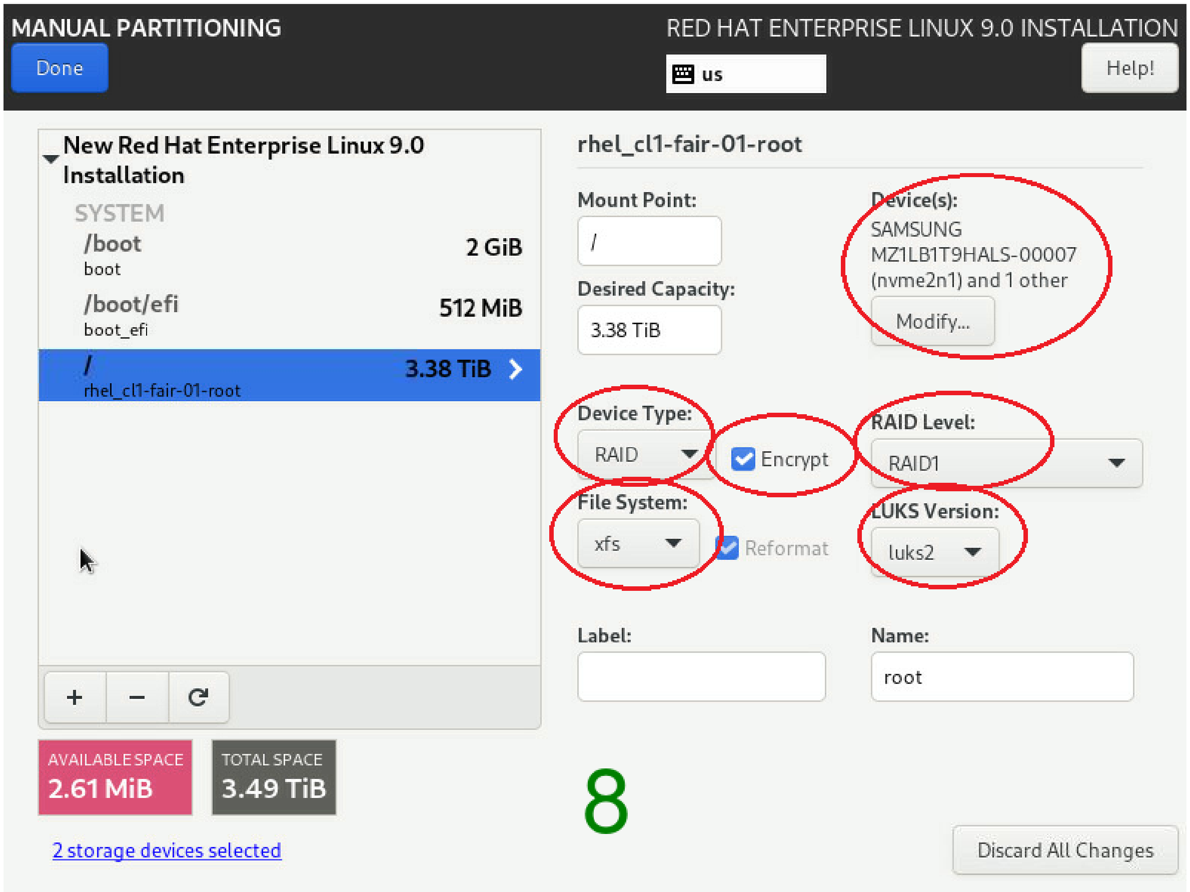
Desired Capacity: Enter size and unit.

For the File System, select XFS. Again confirm that this partition uses both of the devices you selected in step 2.

Click Update Settings to confirm your changes, and Done to begin exiting the partition menu.

A prompt will now appear asking for a LUKS password.

8. Be sure to choose a strong and secure password.



MANUAL PARTITIONING

RED HAT ENTERPRISE LINUX 9.0 INSTALLATION

New Red Hat Enterprise Linux 9.0 Installation

SYSTEM

/boot boot	2 GiB
/boot/efi boot_efi	512 MiB
/ rhel_cl1-fair-01-root	3.38 TiB

rhel_cl1-fair-01-root

Mount Point: /

Desired Capacity: 3.38 TiB

Device(s): SAMSUNG MZ1LB1T9HALS-00007 (nvme2n1) and 1 other

Device Type: RAID

☑ Encrypt

RAID Level: RAID1

File System: xfs

☑ Reformat

LUKS Version: luks2

Label:

Name: root

AVAILABLE SPACE 2.61 MiB

TOTAL SPACE 3.49 TiB

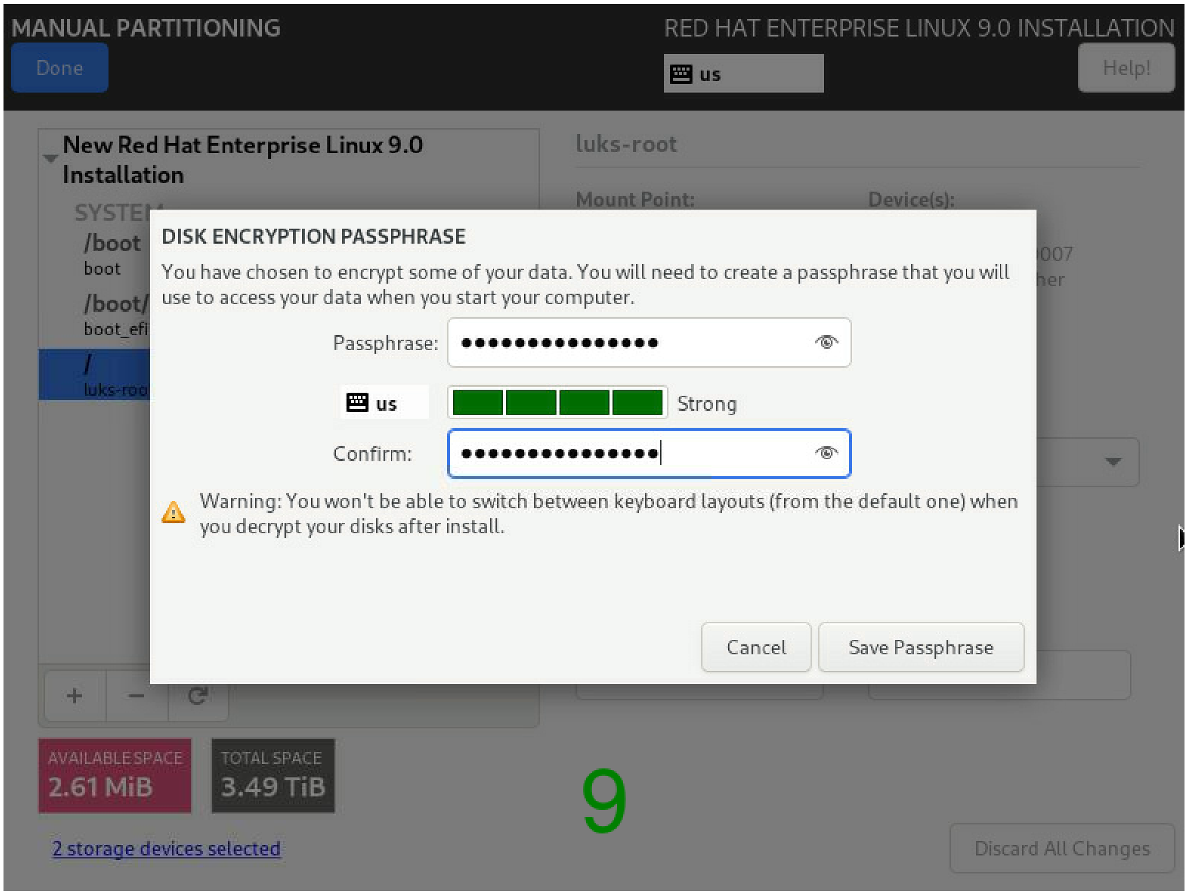
2 storage devices selected

8

After entering your password twice, click Save Passphrase.

9. Click Done.

   This causes a yellow warning bar to appear on the bottom because no swap partition has been created.



DISK ENCRYPTION PASSPHRASE

You have chosen to encrypt some of your data. You will need to create a passphrase that you will use to access your data when you start your computer.

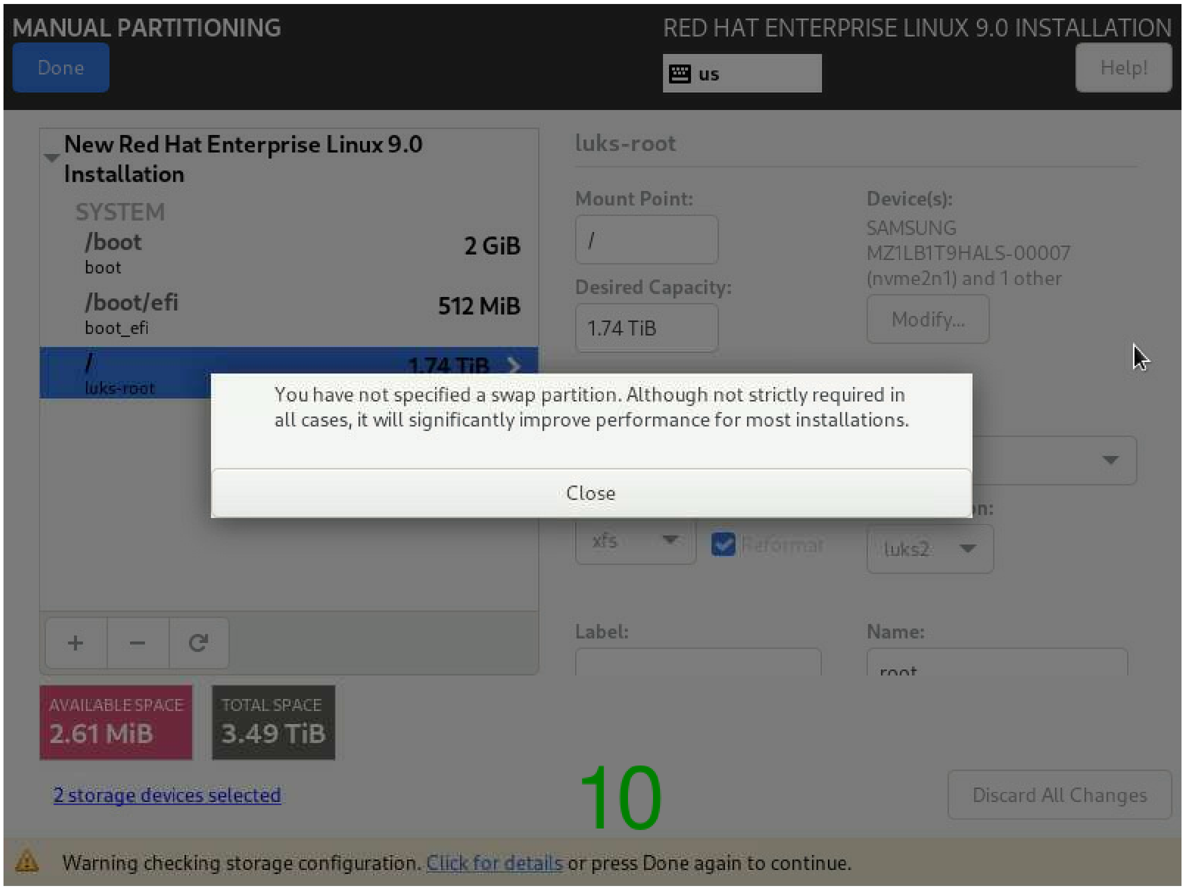
Passphrase: ●●●●●●●●●●●●●●●

us Strong

Confirm: ●●●●●●●●●●●●●●●

Warning: You won't be able to switch between keyboard layouts (from the default one) when you decrypt your disks after install.

10. Click Done again and then click Accept Changes to write all of our customizations to disk.



New Red Hat Enterprise Linux 9.0 Installation

/boot 2 GiB
boot

/boot/efi 512 MiB
boot_efi

/ 1.74 TiB
luks-root

luks-root

Mount Point: /

Desired Capacity: 1.74 TiB

Device(s): SAMSUNG MZ1LB1T9HALS-00007 (nvme2n1) and 1 other

xfs

luks2

Label:

Name: root

AVAILABLE SPACE 2.61 MiB

TOTAL SPACE 3.49 TiB

2 storage devices selected

You have not specified a swap partition. Although not strictly required in all cases, it will significantly improve performance for most installations.

10

Warning checking storage configuration. Click for details or press Done again to continue.

11. Return to step 8 in the *Installing and Configuring Red Hat Enterprise Linux* section.

Chapter 4. Installing the DGX Software

This section requires that you have already installed Red Hat Enterprise Linux or derived operating system on the DGX™ system. You can skip this section of you already installed the DGX software stack during a kickstart install.

Important

Before performing the installation, refer to the *Release Notes* for the latest information and additional instructions depending on the specific release.

4.1. Configuring a System Proxy

If your network requires you to use a proxy:

- Edit the file `/etc/dnf/dnf.conf` and make sure the following lines are present in the `[main]` section, using the parameters that apply to your network:

```
proxy=http://<Proxy-Server-IP-Address>:<Proxy-Port>
proxy_username=<Proxy-User-Name>
proxy_password=<Proxy-Password>
```

4.2. Enabling the DGX Software Repository

Attention

By running these commands you are confirming that you have read and agree to be bound by the NVIDIA Software License Agreement found on the NVIDIA Enterprise Software page. You are also confirming that you understand that any pre-release software and materials available that you elect to install in a DGX might not be fully functional, might contain errors or design flaws, and might have reduced or different security, privacy, availability, and reliability standards relative to commercial versions of NVIDIA software and materials, and that you use pre-release versions at your own risk.

Install the NVIDIA DGX Package for Red Hat Enterprise Linux.

```
sudo dnf install -y https://repo.download.nvidia.com/baseos/el/el-files/9/nvidia-repo-
↪setup-22.12-1.el9.x86_64.rpm
```

4.3. Installing Required Components

1. On Red Hat Enterprise Linux, run the following commands to enable additional repositories required by the DGX software.

   ```
   sudo subscription-manager repos --enable=rhel-9-for-x86_64-appstream-rpms
   sudo subscription-manager repos --enable=rhel-9-for-x86_64-baseos-rpms
   sudo subscription-manager repos --enable=codeready-builder-for-rhel-9-x86_64-rpms
   ```

2. Upgrade to the latest software.

   ```
   sudo dnf update -y --nobest
   ```

3. Install DGX tools and configuration files.

   - For DGX-1, install DGX-1 Configurations.

     ```
     sudo dnf group install -y 'DGX-1 Configurations'
     ```

   - For the DGX-2, install DGX-2 Configurations.

     ```
     sudo dnf group install -y 'DGX-2 Configurations'
     ```

   - For the DGX A100, install DGX A100 Configurations.

     ```
     sudo dnf group install -y 'DGX A100 Configurations'
     ```

   - For the DGX A800, install DGX A800 Configurations.

     ```
     sudo dnf group install -y 'DGX A800 Configurations'
     ```

   - For the DGX H100, install DGX H100 Configurations.

     ```
     sudo dnf group install -y 'DGX H100 Configurations'
     ```

   - For the DGX H200, install DGX H200 Configurations.

     ```
     sudo dnf group install -y 'DGX H200 Configurations'
     ```

   - For the DGX Station, install DGX Station Configurations.

     ```
     sudo dnf group install -y 'DGX Station Configurations'
     ```

   - For the DGX Station A100, install DGX Station A100 Configurations.

     ```
     sudo dnf group install -y 'DGX Station A100 Configurations'
     ```

   The configuration changes take effect only after rebooting the system. To reduce the number of reboots, you can defer rebooting until after you install the drivers.

4.4. Configuring Data Drives

The data drives in the DGX systems can be configured as RAID 0 or RAID 5. RAID 0 provides the maximum storage capacity and performance, but does not provide any redundancy.

RAID 0 is often used for data caching. You can use cachefilesd to provide a cache for NFS shares.

Important

You can change the RAID level later but this will destroy the data on those drives.

Except for the DGX-1, the RAID configuration can be configured during the operating system installation. If you have already configured the RAID array during the installation, you can skip the first step and go to step 2.

1. Configure the `/raid` partition.

   All DGX systems support RAID 0 or RAID 5 arrays.

   The following commands create a RAID array, mount it to `/raid` and create an appropriate entry in `/etc/fstab`.

   - To create a RAID 0 array:

     ```
     sudo /usr/bin/configure_raid_array.py -c -f
     ```

   - To create a RAID 5 array:

     ```
     sudo /usr/bin/configure_raid_array.py -c -f -5
     ```

   Note

   The RAID array must be configured before installing `nvidia-conf-cachefilesd`, which places the proper SELinux label on the `/raid` directory. If you ever need to recreate the RAID array — which will wipe out any labeling on `/raid` — after `nvidia-conf-cachefilesd` has already been installed, be sure to restore the label manually before restarting `cachefilesd`.

   ```
   sudo restorecon /raid
   sudo systemctl restart cachefilesd
   ```

2. (Optional) Install tools for managing the self-encrypting drives (SED) for the data drives on DGX A100, DGX A800, or DGX H100/H200 systems.

   Refer to *Managing Self-Encrypting Drives* for more information.

3. (Optional) If you wish to use your RAID array for caching, install `nvidia-conf-cachefilesd`. This will update the `cachefilesd` configuration to use the `/raid` partition.

   ```
   sudo dnf install -y nvidia-conf-cachefilesd
   ```

4.5. Installing the GPU Driver

You have the option to choose between different GPU driver branches for your DGX system. The latest driver release includes new features but might not provide the same support duration as an older release. Refer to the release notes at the NVIDIA Driver Documentation for more details and the minimum required driver release for the GPU architecture.

1. Display a list of available drivers.

```
dnf module list nvidia-driver
```

Example Output

```
Name                                    Stream
↪Profiles                                                   Summary
nvidia-driver                           latest
↪default [d], fm, ks, src                                   Nvidia driver for
↪latest branch
nvidia-driver                           latest-dkms [d]
↪default [d], fm, ks                                        Nvidia driver for
↪latest-dkms branch
nvidia-driver                           open-dkms
↪default [d], fm, ks, src                                   Nvidia driver for
↪open-dkms branch
nvidia-driver                           515
↪default [d], fm, ks, src                                   Nvidia driver for
↪515 branch
nvidia-driver                           515-dkms
↪default [d], fm, ks                                        Nvidia driver for
↪515-dkms branch
nvidia-driver                           515-open
↪default [d], fm, ks, src                                   Nvidia driver for
↪515-open branch
nvidia-driver                           520
↪default [d], fm, ks, src                                   Nvidia driver for
↪520 branch
nvidia-driver                           520-dkms
↪default [d], fm, ks                                        Nvidia driver for
↪520-dkms branch
nvidia-driver                           520-open
↪default [d], fm, ks, src                                   Nvidia driver for
↪520-open branch
nvidia-driver                           525
↪default [d], fm, ks, src                                   Nvidia driver for
↪525 branch
nvidia-driver                           525-dkms
↪default [d], fm, ks                                        Nvidia driver for
↪525-dkms branch
nvidia-driver                           525-open
↪default [d], fm, ks, src                                   Nvidia driver for
↪525-open branch
nvidia-driver                           530
↪default [d], fm, ks, src                                   Nvidia driver for
↪530 branch
nvidia-driver                           530-dkms
↪default [d], fm, ks                                        Nvidia driver for
```

(continues on next page)

(continued from previous page)

```
↪530-dkms branch
nvidia-driver                           530-open
↪default [d], fm, ks, src                                    Nvidia driver for
↪530-open branch
nvidia-driver                           535 [e]
↪default [d] [i], fm, ks, src [i]                            Nvidia driver for
↪535 branch
nvidia-driver                           535-dkms
↪default [d], fm, ks                                         Nvidia driver for
↪535-dkms branch
nvidia-driver                           535-open
↪default [d], fm, ks, src                                    Nvidia driver for
↪535-open branch
```

Before installing the NVIDIA CUDA driver and configure the system,

- Replace the release version (535) used as an example in step 2 with the release you want to install.
- If the `Stream` column in the example output does not display the compiled driver version you want (for example, 535) but only the DKMS version (`535-dkms`), replace the pertinent `sudo dnf module install --nobest -y nvidia-driver:535...` command in step 2 below with the following command regardless of the system:

  ```
  sudo dnf module install --nobest -y nvidia-driver:535-dkms/fm
  ```

- Ensure that the driver release you intend to install is supported by the GPU in the system.

2. Install the NVIDIA CUDA driver.

   1. For non-NVSwitch systems, such as DGX-1, DGX Station, and DGX Station A100, install the driver using the default and src profiles:

      ```
      sudo dnf module install --nobest -y nvidia-driver:535/{default,src}
      sudo dnf install -y nv-persistence-mode libnvidia-nscq-535
      ```

   2. For NVSwitch systems, such as DGX-2, DGX A100, DGX H100/H200, and DGX A800, install the driver using the fabric manager (fm) and source (src) profiles:

      ```
      sudo dnf module install --nobest -y nvidia-driver:535/{fm,src}
      sudo dnf install -y nv-persistence-mode nvidia-fm-enable
      ```

3. (DGX Station A100 only) Install additional packages required for DGX Station A100.

   These packages must be installed after installation of the `nvidia-driver` module.

   ```
   sudo dnf install -y nvidia-conf-xconfig nv-docker-gpus
   ```

   The configuration changes take effect only after rebooting the system. To reduce the number of reboots, you can defer rebooting until after you install the NVIDIA Container Runtime group.

4. Install and configure the NVIDIA Container Toolkit with Docker CE or Podman.

   Choose one of the following options:

   - *Installing and Running Docker CE*
   - *Installing and Running Podman*

4.5.1. Installing and Running Docker CE

To run an NVIDIA container with Docker CE:

1. Install the NVIDIA container device plugin along with Docker CE.

   Install the NVIDIA Container Runtime group:

   ```
   sudo dnf group install -y --allowerasing 'NVIDIA Container Runtime'
   ```

2. Reboot the system to load the drivers and to update system configurations.

   1. Reboot the system.

      ```
      sudo reboot
      ```

   2. After the system reboots, verify that the drivers are loaded and are handling the NVIDIA devices.

      ```
      nvidia-smi
      ```

      The output shows all available GPUs.

      Example Output

      ```
      +-----------------------------------------------------------------------------
      ↪+
      | NVIDIA-SMI 535.86.10   Driver Version: 535.86.10    CUDA Version: 12.2
      ↪|
      |-------------------------------+----------------------+----------------------
      ↪+
      | GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC
      ↪|
      | Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M.
      ↪|
      |                               |                      |               MIG M.
      ↪|
      |===============================+======================+======================|
      |   0  Tesla V100-SXM2...  On   | 00000000:06:00.0 Off |                    0
      ↪|
      | N/A   35C    P0    42W / 300W |      0MiB / 16160MiB |      0%      Default
      ↪|
      |                               |                      |                  N/A
      ↪|
      +-------------------------------+----------------------+----------------------
      ↪+
      |   1  Tesla V100-SXM2...  On   | 00000000:07:00.0 Off |                    0
      ↪|
      | N/A   35C    P0    44W / 300W |      0MiB / 16160MiB |      0%      Default
      ↪|
      |                               |                      |                  N/A
      ↪|
      +-------------------------------+----------------------+----------------------
      ↪+
      ...
      +-------------------------------+----------------------+----------------------
      ↪+
      |   7  Tesla V100-SXM2...  On   | 00000000:8A:00.0 Off |                    0
      ```

(continues on next page)

(continued from previous page)

```
↪|
| N/A   35C    P0    43W / 300W |      0MiB / 16160MiB |      0%      Default
↪|
|                               |                      |                  N/A
↪|
+-------------------------------+----------------------+----------------------
↪+

+-----------------------------------------------------------------------------
↪+
| Processes:
↪|
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory
↪|
|        ID   ID                                                   Usage
↪|
|=============================================================================|
|  No running processes found
↪|
+-----------------------------------------------------------------------------
↪+
```

3. Run the following command to verify the installation:

   ```
   sudo docker run --gpus=all --rm nvcr.io/nvidia/cuda:12.2.0-base-ubi8 nvidia-smi
   ```

   The output shows all available GPUs.

   For information about `nvcr.io`, refer to the NGC Private Registry User Guide.

4.5.2. Installing and Running Podman

To run an NVIDIA container with Podman:

1. Install Podman.

   ```
   sudo dnf install podman
   ```

2. Install the `nvidia-container-toolkit-base` package.

   ```
   sudo dnf clean expire-cache && sudo dnf install -y nvidia-container-toolkit-base
   ```

3. Check the NVIDIA Container Toolkit version.

   ```
   nvidia-ctk --version
   ```

4. Generate the Container Device Interface (CDI) specification file.

   ```
   sudo nvidia-ctk cdi generate --output=/etc/cdi/nvidia.yaml
   ```

   The sample command uses `sudo` to ensure that the file at `/etc/cdi/nvidia.yaml` is created. You can omit the `--output` argument to print the generated specification to `STDOUT`.

5. Verify that the GPU drivers are loaded and are handling the NVIDIA devices.

```
nvidia-smi -L
```

6. Run the following command to verify the installation.

   ```
   sudo podman run --rm --device nvidia.com/gpu=all ubuntu nvidia-smi -L
   ```

7. Verify your installation by running a sample container with Podman.

   ```
   podman run --security-opt=label=disable --rm --device nvidia.com/gpu=all  ubuntu
    ↪nvidia-smi
   ```

4.6. Installing Optional Components

The DGX is fully functional after installing the components as described in *Installing Required Components*. If you intend to launch NGC containers (which incorporate the CUDA toolkit, NCCL, cuDNN, and TensorRT) on the DGX system, which is the expected use case, then you can skip this section.

If you intend to use your DGX as a development system for running deep learning applications on bare metal, then install the optional components as described in this section. Install CUDA Toolkit 12.2 packages (see *Installing the NVIDIA CUDA Driver from the Local Repository*)

```
sudo dnf install -y cuda-toolkit-12-2 cuda-compat-12-2 nvidia-cuda-compat-setup
```

Note

The output of `nvidia-smi` shows the version of CUDA that is native-compatible with the installed driver (e.g. "NVIDIA-SMI 535.86.10 Driver Version: 535.86.10 CUDA Version: 12.2" in the prior steps). It is recommended to install the CUDA toolkit and compatible packages which match this version.

- To install the NVIDIA Collectives Communication Library (NCCL) Runtime, refer to the NCCL:Getting Started documentation

  ```
  sudo dnf group install -y 'NVIDIA Collectives Communication Library Runtime'
  ```

- To install the CUDA Deep Neural Networks (cuDNN) Library Runtime, refer to the NVIDIA cuDNN page.

  ```
  sudo dnf group install -y 'CUDA Deep Neural Networks Library Runtime'
  ```

- To install NVIDIA TensorRT, refer to the NVIDIA TensorRT page.

4.7. Installing NVIDIA GPUDirect Storage

4.7.1. Prerequisites

- For systems other than NVIDIA DGX-1, DGX-2, and DGX Station, to use the latest GDS version, 12.2.2-1, that is provided by nvidia-fs-dkms-2.17.5-1, you must install an NVIDIA Open GPU Kernel module driver. Refer to *Installing the GPU Driver* for more information about installing the driver.
- For NVIDIA DGX-1, DGX-2, and DGX Station running the generic Linux Kernel, the GPUs in these systems are not supported with the NVIDIA Open GPU Kernel modules. The GDS versions 12.2.2-1 and higher only support the Open GPU Kernel modules.

  For these systems, you must lock the nvidia-fs package to version 2.17.3 or lower and the nvidia-gds package to version 12.2.1-1 or lower.

  ```
  sudo dnf install python3-dnf-plugin-versionlock
  sudo dnf versionlock add nvidia-fs-0:2.17.3-1 nvidia-fs-dkms-0:2.17.3-1 nvidia-
  ↪gds-0:12.2.1-1
  ```

  Example Output

  ```
  Adding versionlock on: nvidia-fs-0:2.17.3-1.*
  Adding versionlock on: nvidia-gds-0:12.2.1-1.*
  ```

4.7.2. Procedure

To install NVIDIA GPUDirect Storage (GDS), perform the following steps.

1. Install the kernel headers and development packages for your kernel.

   ```
   sudo dnf install -y kernel-headers-$(uname -r) kernel-devel-$(uname -r)
   ```

2. Install the GDS package.

   ```
   sudo dnf install -y nvidia-gds
   ```

Refer to Verifying a Successful GDS Installation in the *NVIDIA GPUDirect Storage Installation and Troubleshooting Guide*.

4.8. Installing the Optional NVIDIA Desktop Theme

The DGX Software Repository also provides optional theme packages and desktop wallpapers to give the user-interface an NVIDIA look-and-feel for the DGX Station desktop. These packages would have been installed as part of the **DGX Station Configurations** group.

1. To apply the theme and background images, first open gnome-tweaks.

2. Under **Applications**, select one of the **NV-Yaru** themes. This comes in default, light, and dark variations.

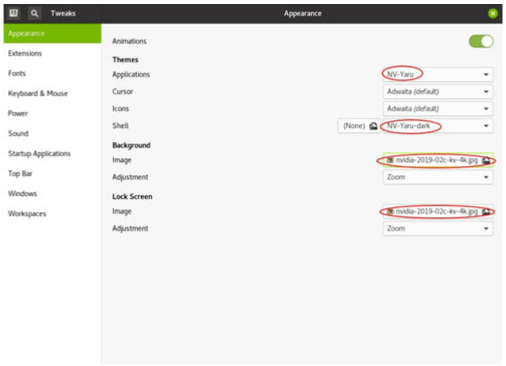
3. Under **Shell**, select the **NV-Yaru-dark** theme.

   If this field is grayed out, you might need to reboot the system or restart GDM in order to enable the user-themes extension.

4. To restart GDM, issue the following.

```
sudo systemctl restart gdm
```

5. Select one of the NVIDIA wallpapers for the background image and lock screen.



Animations

Themes

Applications: NV-Yaru

Cursor: Adwaita (default)

Icons: Adwaita (default)

Shell: (None) NV-Yaru-dark

Background

Image: nvidia-2019-02c-kv-4k.jpg

Adjustment: Zoom

Lock Screen

Image: nvidia-2019-02c-kv-4k.jpg

Adjustment: Zoom

Chapter 5. Installing on “Air-Gapped” Systems

For security purposes, some installations require that systems be isolated from the internet or outside networks. Since most DGX software updates are accomplished through an over-the-network process with NVIDIA servers, this section explains how updates can be made when using an over-the-network method is not an option. It includes a process for installing containers as well.

5.1. Registering Your System

See the Red Hat customer portal knowledge base article How to register and subscribe a system offline to the Red Hat Customer Portal

5.2. Creating a Local Mirror of the NVIDIA Repository

Instructions for setting up a private repository or mirroring the NVIDIA and the Red Hat repositories are beyond the scope of this document. It is expected that users are knowledgeable about those processes.

The Red Hat customer portal provides a knowledge base article for creating a local mirror. Pay particular attention to the instructions under *Create a local repo with Red Hat Enterprise Linux 9.* The reposync command can now download repository metadata as well, so there is no longer a need to use createrepo after.

The repositories that will need to be mirrored are:

- rhel-9-for-x86_64-appstream-rpms
- rhel-9-for-x86_64-baseos-rpms
- codeready-builder-for-rhel-9-x86_64-rpms
- nvidia-dgx-9
- CUDA

Once mirrored, be sure to configure the target system to use your local repository. This can be accomplished by creating file under `/etc/yum.repos.d/my_mirror.repo` with the following contents:

```
[nvidia-dgx-9]
name=NVIDIA DGX EL9
baseurl=file://path/to/your/nvidia-dgx-9-repo-mirror/
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-dgx-cosmos-support

[CUDA]
name=NVIDIA CUDA for EL9
baseurl=file://path/to/your/CUDA-repo-mirror/
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-cuda
```

> **Note**
>
> The instructions assume that you have the repositories enabled on the local machine. See Enabling the DGX Software Repository for instructions on enabling the NVIDIA DGX EL8 repository.

The CUDA repository makes use of modularity streams. If mirroring onto a Red Hat Enterprise Linux 9 (or Fedora) system, it's possible to synchronize the modularity metadata in a single command:

```
reposync -p /path/to/your/mirror/ --repoid=CUDA --download-metadata --downloadcomps
```

However, if you are mirroring the CUDA repository onto another distribution (for example Ubuntu, CentOS 7, Red Hat Enterprise Linux 7) then you will have to manually generate the repository metadata:

```
reposync -p /path/to/your/mirror/ --repoid=CUDA --downloadcomps
createrepo -v /path/to/your/mirror/
python3 genmodules.py /path/to/your/mirror/ /tmp/modules.yaml
modifyrepo /tmp/modules.yaml /path/to/your/mirror/repodata/
```

Note that you can find the genmodules.py script from NVIDIA's yum-packaging-precompiled-kmod repository on GitHub:

https://github.com/NVIDIA/yum-packaging-precompiled-kmod

5.3. Installing Docker Containers

This method applies to Docker containers hosted on the NGC Container Registry. Most container images are freely available, but some are locked and require that you have an NGC account to access. See the NGC Private Registry User Guide for instructions on accessing locked container images.

1. Enter the docker pull command, specifying the image registry, image repository, and tag.

   ```
   docker pull nvcr.io/nvidia/repository:tag
   ```

2. Verify the image is on your system using docker images.

   ```
   docker images
   ```

3. Save the Docker image as an archive.

```
docker save nvcr.io/nvidia/repository:tag > framework.tar
```

4. Transfer the image to the air-gapped system using removable media such as a USB flash drive.
5. Load the NVIDIA Docker image.

   ```
   docker load -i framework.tar
   ```

6. Verify the image is on your system.

   ```
   docker images
   ```

Chapter 6. Installing with Local Repositories

As an alternative to the method described in *Creating a Local Mirror of the NVIDIA Repository*, the DGX software can also be installed using local repositories. These repositories are provided by NVIDIA for installing the required packages from local repositories, allowing installation of packages without network access.

6.1. Installing the Base Red Hat Enterprise Linux System

The Red Hat components of the software must be installed either using the standard network based installation, or as described in *Installing on "Air-Gapped" Systems*. This includes registering and subscribing to the Red Hat Customer Portal.

If the system is connected to the network, follow the steps of the standard Red Hat Enterprise Linux installation, then enable the additional Red Hat repositories. Finally, execute:

```
sudo dnf update -y
sudo dnf install -y kernel-devel kernel-headers
sudo reboot
```

If the system is not connected to the network, follow the instructions in *Creating a Local Mirror of the NVIDIA Repository*

6.2. Downloading the Local Repositories

Download the packages that apply to your release.

6.2.1. Release EL9-24.12

DGX Software local repository	dgx-local-repo-24.12-8.el9.x86_64.rpm
NVIDIA CUDA driver local repository	nvidia-driver-local-repo-rhel9-550.127.08-1.0-1.x86_64.rpm nvidia-driver-local-repo-rhel9-535.216.03-1.0-1.x86_64.rpm
NVIDIA CUDA Toolkit local repository	cuda-repo-rhel9-12-4-local-12.4.1_550.54.15-1.x86_64.rpm cuda-repo-rhel9-12-2-local-12.2.2_535.104.05-1.x86_64.rpm

6.2.2. Release EL9-24.06

DGX Software local repository	dgx-local-repo-24.06-3.el9.x86_64.rpm
NVIDIA CUDA driver local repository	nvidia-driver-local-repo-rhel9-550.90.07-1.0-1.x86_64.rpm nvidia-driver-local-repo-rhel9-535.183.01-1.0-1.x86_64.rpm
NVIDIA CUDA Toolkit local repository	cuda-repo-rhel9-12-4-local-12.4.1_550.54.15-1.x86_64.rpm cuda-repo-rhel9-12-2-local-12.2.2_535.104.05-1.x86_64.rpm

6.2.3. Release EL9-23.12

DGX Software local repository	dgx-local-repo-23.12-3.el9.x86_64.rpm
NVIDIA CUDA driver local repository	nvidia-driver-local-repo-rhel9-535.129.03-1.0-1.x86_64.rpm
NVIDIA CUDA Toolkit local repository	cuda-repo-rhel9-12-2-local-12.2.2_535.104.05-1.x86_64.rpm

6.2.4. Release EL9-23.08

DGX Software local repository	dgx-local-repo-23.08-11.el9.x86_64.rpm
NVIDIA CUDA driver local repository	nvidia-driver-local-repo-rhel9-535.104.05-1.0-1.x86_64.rpm
NVIDIA CUDA Toolkit local repository	cuda-repo-rhel9-12-2-local-12.2.2_535.104.05-1.x86_64.rpm

6.2.5. Release EL9-23.01

DGX Software local repository	dgx-local-repo-23.01-14.el9.x86_64.rpm
NVIDIA CUDA driver local repository	nvidia-driver-local-repo-rhel9-525.105.17-1.0-1.x86_64.rpm
NVIDIA CUDA Toolkit local repository	cuda-repo-rhel9-12-1-local-12.1.1_530.30.02-1.x86_64.rpm

6.3. Installing the Local Repositories

Install the local repositories:

```
sudo dnf install -y ./dgx-local-repo-24.12-8.el9.x86_64.rpm
sudo dnf install -y ./nvidia-driver-local-repo-rhel9-535.216.03-1.0-1.x86_64.rpm
sudo dnf install -y ./cuda-repo-rhel9-12-2-local-12.2.2_535.104.05-1.x86_64.rpm
```

With the local repositories installed, proceed with the standard *Installing the DGX Software* instructions, with the exception of the CUDA driver installation. To install the CUDA driver from the local repository, follow the steps in the next section *Installing the NVIDIA CUDA Driver from the Local Repository*.

6.4. Installing the NVIDIA CUDA Driver from the Local Repository

- For non-NVSwitch systems, such as DGX-1, DGX Station, and DGX Station A100, install the driver using the default profile:

  ```
  sudo dnf module install -y nvidia-driver:535-dkms
  sudo dnf install -y --allowerasing nv-persistence-mode libnvidia-nscq-535
  ```

- For NVSwitch systems, such as DGX-2, DGX A100, DGX A800, and DGX H100/H200, install the driver using the fabric manager (fm) profile:

  Note

  Replace the 535 release of the GPU driver with the release family you want to install. The DGX H200 system requires the GPU driver release to be 550 or greater.

  ```
  sudo dnf module install -y nvidia-driver:535-dkms/fm
  sudo dnf install -y nv-persistence-mode nvidia-fm-enable
  ```

Chapter 7. Installing NVIDIA MLNX_OFED

The DGX software stack for Red Hat Enterprise Linux does not include the NVIDIA MLNX_OFED (OpenFabrics Enterprise Distribution) for Linux. This is to ensure that the MLNX_OFED driver is in sync with the Red Hat distribution kernel. This section describes how to download, install, and upgrade MLNX_OFED on systems that are running Red Hat Enterprise Linux.

7.1. Prerequisites

- NVIDIA validates each release of NVIDIA DGX Software for Red Hat Enterprise Linux with a specific MLNX_OFED version. Refer to the *Release Notes* for the recommended MLNX_OFED version to install.

7.2. Installing and Configuring MLNX_OFED

This section describes how to install MLNX_OFED on systems that do not yet have it installed. It is imperative that a validated MLNX_OFED version is used for the RHEL version that the DGX system is running.

Important

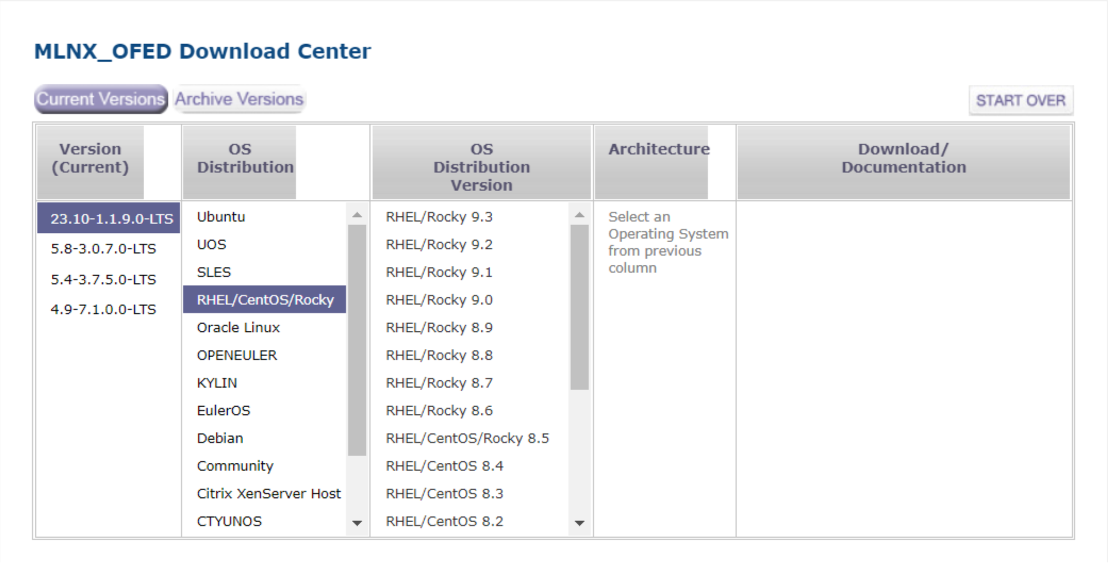
Running the `dnf update` command at any time to install the drivers can update the system to the latest Red Hat Enterprise Linux version.

1. Determine which version of Red Hat Enterprise Linux is installed on the DGX system.

   ```
   cat /etc/redhat-release
   ```

2. After referring to the release notes, download the MLNX_OFED software bundle.

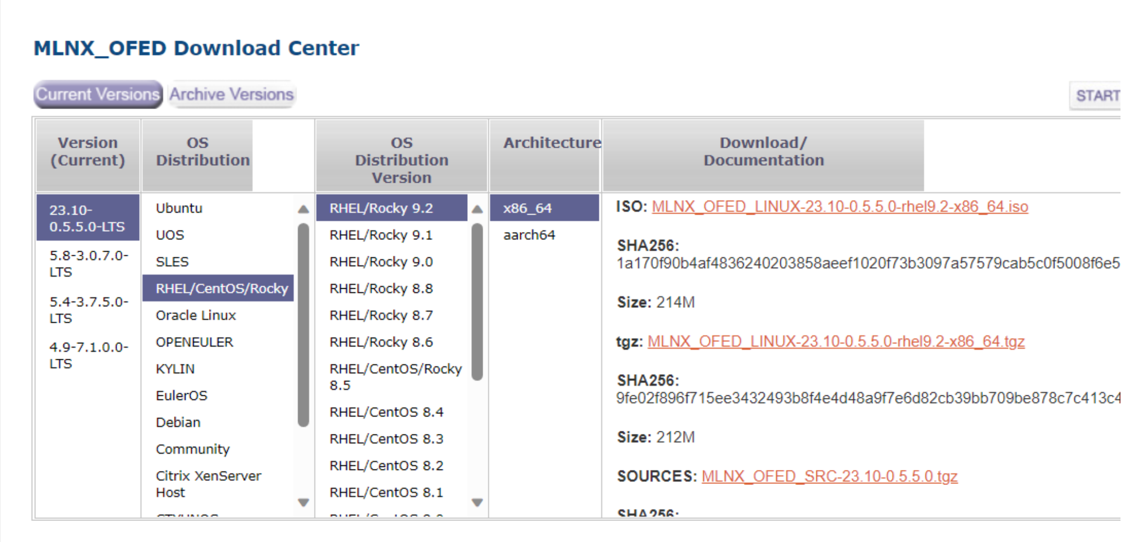
   1. Go to the Linux InfiniBand Drivers page, and scroll down to the **MLNX_OFED Download Center** matrix.



MLNX_OFED Download Center

Version (Current)	OS Distribution	OS Distribution Version	Architecture	Download/ Documentation
23.10-1.1.9.0-LTS	Ubuntu	RHEL/Rocky 9.3	Select an Operating System from previous column	
5.8-3.0.7.0-LTS	UOS	RHEL/Rocky 9.2		
5.4-3.7.5.0-LTS	SLES	RHEL/Rocky 9.1		
4.9-7.1.0.0-LTS	RHEL/CentOS/Rocky	RHEL/Rocky 9.0		
	Oracle Linux	RHEL/Rocky 8.9		
	OPENEULER	RHEL/Rocky 8.8		
	KYLIN	RHEL/Rocky 8.7		
	EulerOS	RHEL/Rocky 8.6		
	Debian	RHEL/CentOS/Rocky 8.5		
	Community	RHEL/CentOS 8.4		
	Citrix XenServer Host	RHEL/CentOS 8.3		
	CTYUNOS	RHEL/CentOS 8.2		

2. At the **MLNX_OFED Download Center** matrix, choose the MLNX_OFED version, OS distribution and distribution version, and architecture to show the software package and documentation. For example,
   - **Version**: 23.10-1.1.9.0-LTS
   - **OS Distribution**: RHEL/CentOS/Rocky
   - **OS Distribution Version**: RHEL/Rocky 9.2
   - **Architecture**: x86_64



MLNX_OFED Download Center

Version (Current)	OS Distribution	OS Distribution Version	Architecture	Download/ Documentation
23.10-0.5.5.0-LTS	Ubuntu	RHEL/Rocky 9.2	x86_64	ISO: MLNX_OFED_LINUX-23.10-0.5.5.0-rhel9.2-x86_64.iso
5.8-3.0.7.0-LTS	UOS	RHEL/Rocky 9.1	aarch64	SHA256: 1a170f90b4af4836240203858aeef1020f73b3097a57579cab5c0f5008f6e5
5.4-3.7.5.0-LTS	SLES	RHEL/Rocky 9.0		Size: 214M
4.9-7.1.0.0-LTS	RHEL/CentOS/Rocky	RHEL/Rocky 8.8		tgz: MLNX_OFED_LINUX-23.10-0.5.5.0-rhel9.2-x86_64.tgz
	Oracle Linux	RHEL/Rocky 8.7		SHA256: 9fe02f896f715ee3432493b8f4e4d48a9f7e6d82cb39bb709be878c7c413c4
	OPENEULER	RHEL/Rocky 8.6		Size: 212M
	KYLIN	RHEL/CentOS/Rocky 8.5		SOURCES: MLNX_OFED_SRC-23.10-0.5.5.0.tgz
	EulerOS	RHEL/CentOS 8.4		SHA256:
	Debian	RHEL/CentOS 8.3		
	Community	RHEL/CentOS 8.2		
	Citrix XenServer Host	RHEL/CentOS 8.1		

3. Click the supported ISO or tgz package.

   The **Mellanox OFED (MLNX_OFED) Software: End-User Agreement** page appears.

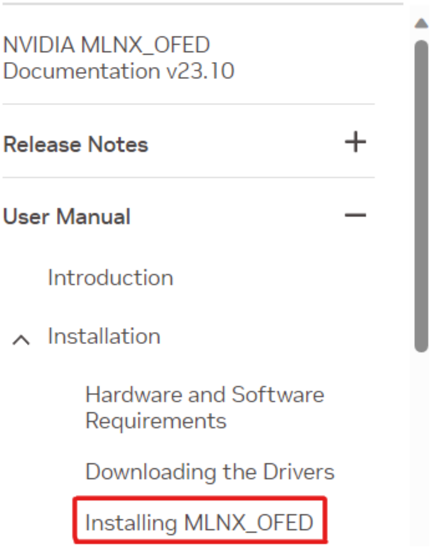
4. Accept the End User License Agreement by clicking **I Have Read the Above End User License Agreement**.

   The selected software package starts to download.

3. After downloading the correct MLNX_OFED software package, proceed with the installation steps.

   For issues during RHEL 9.2 install using MODEF mlnxofedinstall, refer to the Known Issues *MOFED mlnxofedinstall reports "Current operation system is not supported" using RHEL 9.2*.

   1. Go to the MLNX_OFED Software Releases site and select the MLNX_OFED software version that you downloaded.
   2. Click the **User Manual** link and then navigate to **Installation** > **Installing MLNX_OFED**.

      NVIDIA MLNX_OFED Documentation v23.10

      Release Notes +

      User Manual −

      Introduction

      Installation

      Hardware and Software Requirements

      Downloading the Drivers

      Installing MLNX_OFED

      Follow the installation instructions.

      > **Note**
      >
      > The system might report that additional software needs to be installed before performing the installation. If such a message appears, install the software and then retry installing the MLNX_OFED driver.

4. If you intend to use NVIDIA GPUDirect Storage (GDS), enable the driver's GDS support according to the instructions at MLNX_OFED Requirements and Installation.
5. Install `nvidia-mlnx-config`.

```
sudo dnf install -y nvidia-mlnx-config
```

6. Install kernel headers and development packages for your kernel.

   These are needed for the ensuing DKMS compilation.

```
sudo dnf install -y kernel-headers-$(uname -r) kernel-devel-$(uname -r)
```

7. After installing the MLNX_OFED drivers, install the NVIDIA peer memory module.

```
sudo dnf install -y nvidia-peer-memory-dkms
```

Note

nvidia-peer-memory-dkms version 1.2 or later, requires MOFED version 5.4-3.0.3.0 or later. Using MOFED versions that do not fit this criteria will result in a build failure of the nv_peer_mem DKMS module. For more information, see: https://github.com/Mellanox/nv_peer_memory/issues/94#

Note

While in-box drivers might be available, using the in-box drivers is not recommended as they provide lower performance than the official MLNX OFED drivers and they do not support the GPUDirect$^{\text{TM}}$ RDMA feature. For more information on configuring the in-box drivers, see the following Red Hat Enterprise Linux documentation: Configuring InfiniBand and RDMA Networks.

7.3. Updating NVIDIA MLNX_OFED

This section describes how to update MOFED on systems that already have it installed. The Mellanox InfiniBand Drivers in RPM packages are precompiled for a specific kernel version. Again – it is imperative that the correct MOFED version is used for the RHEL version that the DGX system has been updated to. There is no need to uninstall the current MOFED first, because the "mlnxofedinstall" script will automatically uninstall any previously installed versions.

1. Upgrade the Red Hat Enterprise Linux release and kernel version.

   ```
   sudo dnf update --nobest
   ```

2. Determine which version of Red Hat Enterprise Linux is installed on the DGX system.

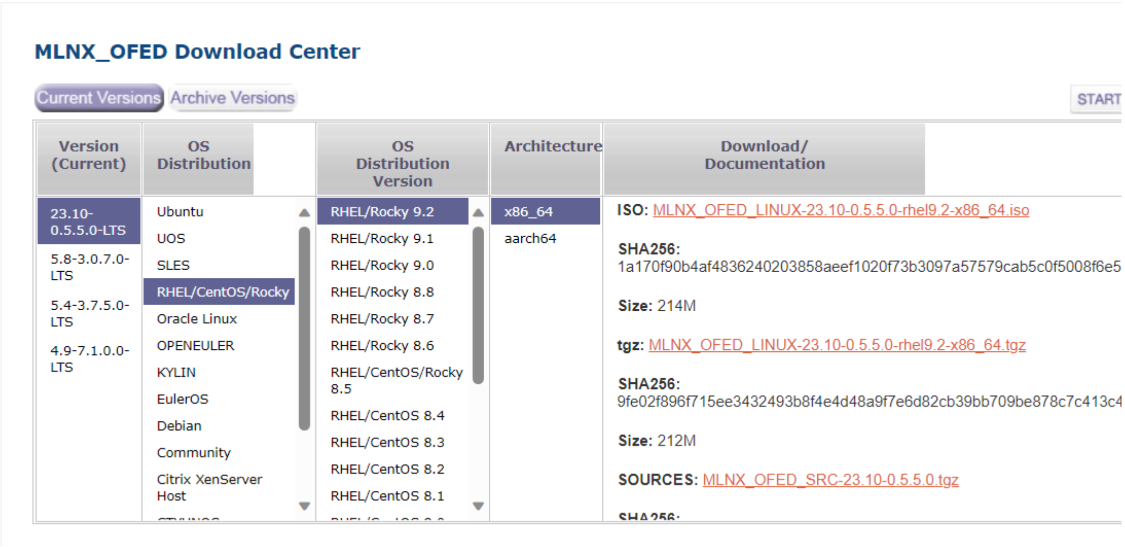
   ```
   cat /etc/redhat-release
   ```

3. After referring to the release notes, download the MLNX_OFED software bundle.

   1. Go to the Linux InfiniBand Drivers page, and scroll down to the **MLNX_OFED Download Center** matrix.

   2. At the **MLNX_OFED Download Center** matrix, choose the MLNX_OFED version, OS distribution and distribution version, and architecture to show the software package and documentation. For example,

      - **Version**: 23.10-1.1.9.0-LTS
      - **OS Distribution**: RHEL/CentOS/Rocky
      - **OS Distribution Version**: RHEL/Rocky 9.2
      - **Architecture**: X86_64



MLNX_OFED Download Center

Version (Current)	OS Distribution	OS Distribution Version	Architecture	Download/ Documentation
23.10-0.5.5.0-LTS	Ubuntu	RHEL/Rocky 9.2	x86_64	**ISO:** MLNX_OFED_LINUX-23.10-0.5.5.0-rhel9.2-x86_64.iso **SHA256:** 1a170f90b4af4836240203858aeef1020f73b3097a57579cab5c0f5008f6e5 **Size:** 214M **tgz:** MLNX_OFED_LINUX-23.10-0.5.5.0-rhel9.2-x86_64.tgz **SHA256:** 9fe02f896f715ee3432493b8f4e4d48a9f7e6d82cb39bb709be878c7c413c4 **Size:** 212M **SOURCES:** MLNX_OFED_SRC-23.10-0.5.5.0.tgz **SHA256:**
5.8-3.0.7.0-LTS	UOS	RHEL/Rocky 9.1	aarch64	
5.4-3.7.5.0-LTS	SLES	RHEL/Rocky 9.0		
4.9-7.1.0.0-LTS	RHEL/CentOS/Rocky	RHEL/Rocky 8.8		
	Oracle Linux	RHEL/Rocky 8.7		
	OPENEULER	RHEL/Rocky 8.6		
	KYLIN	RHEL/CentOS/Rocky 8.5		
	EulerOS	RHEL/CentOS 8.4		
	Debian	RHEL/CentOS 8.3		
	Community	RHEL/CentOS 8.2		
	Citrix XenServer Host	RHEL/CentOS 8.1		

   3. Click the supported ISO or tgz package.

      The **Mellanox OFED (MLNX_OFED) Software: End-User Agreement** page appears.

   4. Accept the End User License Agreement by clicking **I Have Read the Above End User License Agreement**.

      The selected software package starts to download.

4. Mount the downloaded ISO on the system.

   The following example shows the ISO being mounted on the `/mnt` directory.

   ```
   sudo mount MLNX_OFED_LINUX-<version>.iso /mnt
   ```

5. Prepare to install the driver.

   1. Remove `nvidia-mlnx-config` and `nvidia-peer-memory-dkms`.

      ```
      sudo dnf remove -y nvidia-mlnx-config nvidia-peer-memory-dkms
      ```

      The `mlnxofedinstall` step will remove packages prior to installing new ones, causing `nvidia-mlnx-config`and `nvidia-peer-memory-dkms` to fall out because they depend on some of these removed packages. Removing those components ahead of time avoids issues. These will be reinstalled as a final step.

   2. Specify the new kernel version to use when installing the driver.

      ```
      NEXTKERNEL=$(sudo grubby --default-kernel | sed 's/.*vmlinuz\-//g')
      ```

6. Install the driver with the `-k` and `-s` flags to specify the new kernel version and kernel source path.

   ```
   sudo /mnt/mlnxofedinstall -k ${NEXTKERNEL} -s /lib/modules/${NEXTKERNEL}/build --
   ↪force
   ```

Note

> Note: The system might report that additional software needs to be installed before performing the installation. If such a message appears, install the software and then repeat this step.

7. Reboot.

   ```
   sudo reboot
   ```

8. Reinstall `nvidia-mlnx-config` and `nvidia-peer-memory-dkms`.

   ```
   sudo dnf install -y nvidia-mlnx-config nvidia-peer-memory-dkms
   ```

7.4. Installing the nvidia-peermem-loader Package

The `nvidia-peermem` kernel module registers the NVIDIA GPU with the InfiniBand subsystem by using peer-to-peer APIs provided by the NVIDIA GPU driver. This module, originally maintained by Mellanox on GitHub, is now included with the NVIDIA Linux GPU driver. For more information, refer to Using nvidia-peermem in the NVIDIA GPUDirect RDMA documentation.

No service automatically loads the `nvidia-peermem` module. To load the module automatically at boot, install the NVIDIA peermem loader package (`nvidia-peermem-loader`).

```
sudo dnf install nvidia-peermem-loader
```

This package adds the `nvidia-peermem` module in `/etc/modules-load.d/nvidia-peermem.conf`.

Chapter 8. Installing NVIDIA DOCA-OFED

The NVIDIA DGX™ Software Stack for Red Hat Enterprise Linux does not include the NVIDIA DOCA™ OFED (OpenFabrics Enterprise Distribution) software for Linux. This is to ensure that the DOCA-OFED software, a subset of the full DOCA package, is in sync with the Red Hat distribution kernel. This topic describes how to download, install, and upgrade the DOCA-OFED software on systems that are running Red Hat Enterprise Linux.

8.1. DOCA-Host Installation Profiles

The DOCA software package contains several subsets called the DOCA-Host installation profiles, which are fully validated and tested installation packages. The following table lists the available DOCA-Host profiles:

DOCA-Host Profile	Description
doca-ofed	Allows you to install the same drivers and tools of MLNX_OFED using the DOCA-Host package, but without other DOCA functionality.
doca-network	Intended for users who want to use only the networking functionality of the DOCA-Host package.
doca-all	Intended for users who want to use the full extent of DOCA drivers and libraries, the full DOCA-Host installation.

For more information, refer to NVIDIA DOCA Profiles.

8.2. Prerequisites

1. Before installing a different version of DOCA-OFED software, you must remove the installed DOCA-OFED or MLNX_OFED software on your system.

   - Debian-based Linux

   ```
   # Remove the installed DOCA-OFED software.
   $ for f in $( dpkg --list | grep doca | awk '{print $2}' ); do echo $f ; sudo
   ↪apt remove --purge $f -y ; done

   # Remove the installed MLNX_OFED software.
   $ sudo /usr/sbin/ofed_uninstall.sh --force

   $ sudo apt-get autoremove
   ```

   - RPM-based Linux

   ```
   # Remove the installed DOCA-OFED software from the host.
   for f in $(rpm -qa | grep -i doca ) ; do sudo yum -y remove $f; done

   # Remove the installed MLNC_OFED software.
   sudo /usr/sbin/ofed_uninstall.sh --force

   sudo yum autoremove
   sudo yum makecache
   ```

2. Download and install the NVIDIA RPM GPG key.

   1. Download the NVIDIA RPM-GPG-KEY-Mellanox-SHA256 key.

      ```
      wget http://www.mellanox.com/downloads/ofed/RPM-GPG-KEY-Mellanox-SHA256
      ```

   2. Install the key.

      ```
      sudo rpm --import RPM-GPG-KEY-Mellanox-SHA256
      ```

   3. Verify that the key was successfully imported.

      ```
      sudo rpm -q gpg-pubkey --qf '%{NAME}-%{VERSION}-%{RELEASE}\t%{SUMMARY}\n' |
      ↪grep Mellanox
      ```

8.3. Installing DOCA-OFED on Systems with ConnectX-7 Cards

To install the DOCA-Host package with the doca-ofed profile on the host,

1. Open the Installation Files page, download the DOCA-Host installation file based on the OS and Arch options you want.

   Alternatively, you can download the installation file using the DOCA Downloads page.

2. Unpack the RPM package.

```
sudo rpm -Uvh <repo_file>.rpm
```

3. Perform an update using the yum command.

   ```
   sudo yum makecache
   ```

4. Determine if the kernel version on your host is supported as shown in Supported Host OS per DOCA-Host Installation Profile.

   If the kernel version is not supported, follow the instructions described in DOCA Extra Package.

5. Run the `yum install` command for the doca-ofed profile installation.

   ```
   sudo yum install -y doca-ofed
   ```

6. Re-create an initramfs image.

   ```
   sudo dracut -f
   ```

7. Reboot the system.

   ```
   sudo systemctl reboot
   ```

8. Register your new Red Hat Enterprise Linux system to the Customer Portal using Red Hat Subscription-Manager.

   For more information, refer to How to register and subscribe a RHEL system to the Red Hat Customer Portal using Red Hat Subscription-Manager?.

For more information about the doca-ofed profile installation on the host, refer to Installing Software on Host.

8.4. Installing DOCA-OFED on Systems with BlueField-3 in NIC Mode

If your system is equipped with the NVIDIA® BlueField®-3 DPU, ensure that the DPU is set in NIC mode (NIC Mode for BlueField-3) and then proceed with the following instructions.

1. Install the RShim driver to manage and flash the BlueField-3 DPU.

   Follow the procedure described in Installing Prerequisites on Host for Target BlueField.

   - Choose the procedure for the RPM-based Linux.

2. Determine the BlueField-3 device ID.

   Follow the instructions described in Determining BlueField Device ID.

3. Install the DOCA-Host software on the host.

   Follow the instructions for the selected DOCA-Host profile to install the DOCA drivers and tools as described in Installing Software on Host.

Additional Information

- MFT download instructions: Updating Firmware for a Single Network Interface Card (NIC)
- Changing BlueField-3 BMC default password: Changing Default Password

8.5. Installing the nvidia-peermem-loader Package

The nvidia-peermem kernel module registers the NVIDIA GPU with the InfiniBand subsystem by using peer-to-peer APIs provided by the NVIDIA GPU driver. This module, originally maintained by Mellanox on GitHub, is now included with the NVIDIA Linux GPU driver. For more information, refer to Using nvidia-peermem in the NVIDIA GPUDirect RDMA documentation.

No service automatically loads the nvidia-peermem module. To load the module automatically at boot, install the NVIDIA peermem loader package (nvidia-peermem-loader).

```
sudo dnf install nvidia-peermem-loader
```

This package adds the nvidia-peermem module in /etc/modules-load.d/nvidia-peermem.conf.

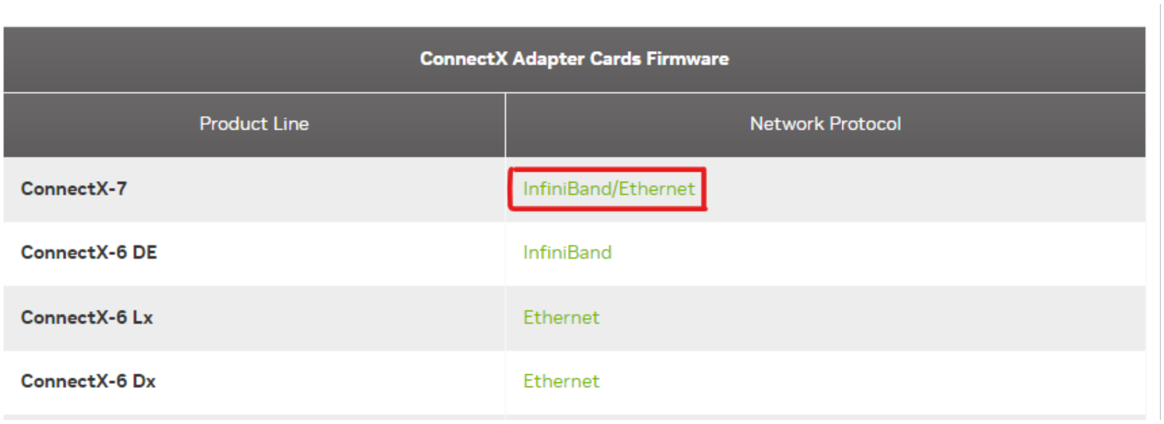
Chapter 9. Installing Firmware

This topic provides information about installing the network card firmware:

9.1. Installing ConnectX-7 Firmware

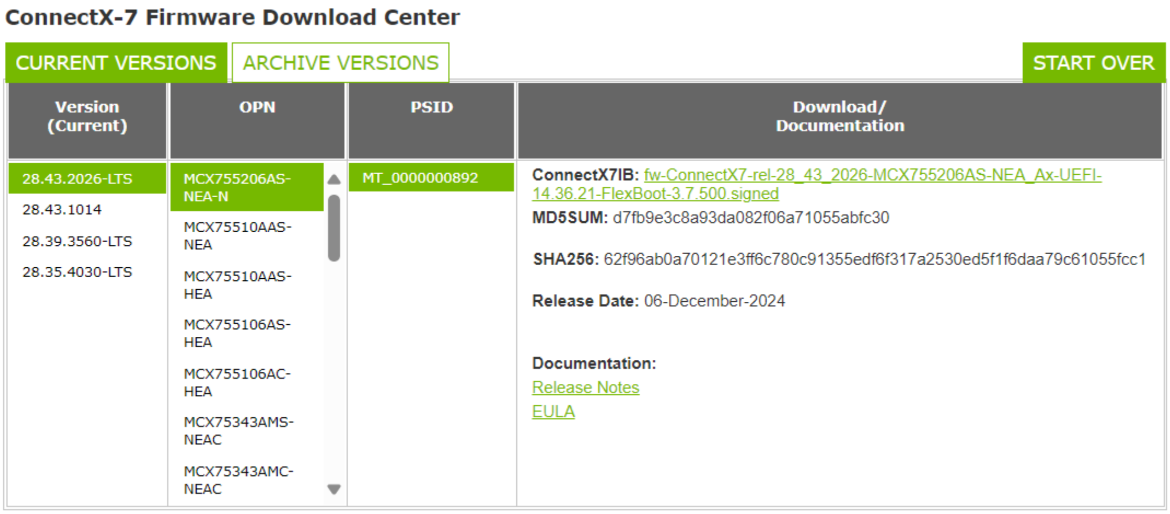
Follow these steps to update firmware for the ConnectX®-7 InfiniBand/Ethernet PCI Express Adapter Cards using the NVIDIA Networking Firmware Downloads Page.

1. Navigate to the NVIDIA Networking Firmware Downloads page.
2. From the **ConnectX Adapter Cards Firmware** table, select version **ConnectX-7** and click the **InfiniBand/Ethernet** network protocol.

ConnectX Adapter Cards Firmware	
Product Line	Network Protocol
ConnectX-7	InfiniBand/Ethernet
ConnectX-6 DE	InfiniBand
ConnectX-6 Lx	Ethernet
ConnectX-6 Dx	Ethernet

   The Firmware for ConnectX®-7 InfiniBand page opens.
3. On the **ConnectX-7 Firmware Download Center** matrix, display the firmware information and documentation.
   1. Select the **CURRENT VERSIONS** tab.
   2. Choose the **Version (Current)**, **OPN** (ordering part number), and **PSID** options you want. For example,



ConnectX-7 Firmware Download Center

Version (Current)	OPN	PSID	Download/ Documentation
28.43.2026-LTS 28.43.1014 28.39.3560-LTS 28.35.4030-LTS	MCX755206AS-NEA-N MCX75510AAS-NEA MCX75510AAS-HEA MCX755106AS-HEA MCX755106AC-HEA MCX75343AMS-NEAC MCX75343AMC-NEAC	MT_0000000892	**ConnectX7IB:** fw-ConnectX7-rel-28_43_2026-MCX755206AS-NEA_Ax-UEFI-14.36.21-FlexBoot-3.7.500.signed **MD5SUM:** d7fb9e3c8a93da082f06a71055abfc30 **SHA256:** 62f96ab0a70121e3ff6c780c91355edf6f317a2530ed5f1f6daa79c61055fcc1 **Release Date:** 06-December-2024 **Documentation:** Release Notes EULA

4. Click the **ConnectX7IB** link to download the firmware BIN file.
5. After downloading the correct ConnectX-7 firmware, proceed with the installation steps.

Alternatively, you can use the following methods to install the ConnectX-7 adapter card firmware:

- Using the `mstflint` tool

  If you have installed the MTNIC Driver on your machine, you can update the firmware using the `mstflint` tool as described in the mstflint FW Burning Tool README. You can download the `mstflint` tool from the OpenFabrics site at mstflint_SW for Linux.

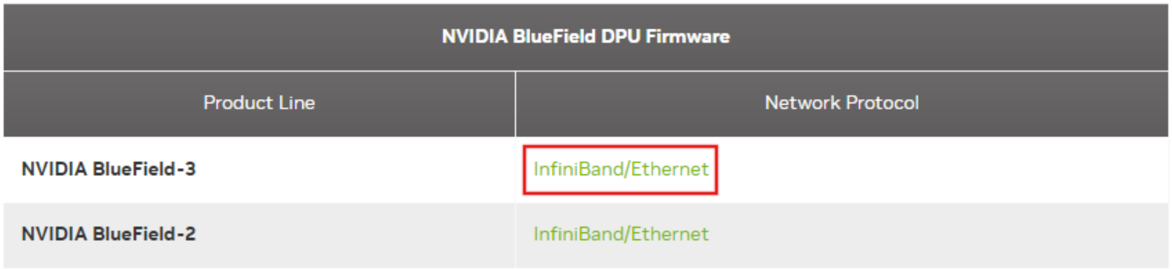
- Using the NVIDIA Firmware Tools (MFT)

  For details, refer to Updating Firmware for a Single Network Interface Card (NIC).

9.2. Installing BlueField-3 NIC Firmware

Follow these steps to update firmware for the NVIDIA® BlueField®-3 DPUs in NIC mode using the NVIDIA Networking Firmware Downloads Page.

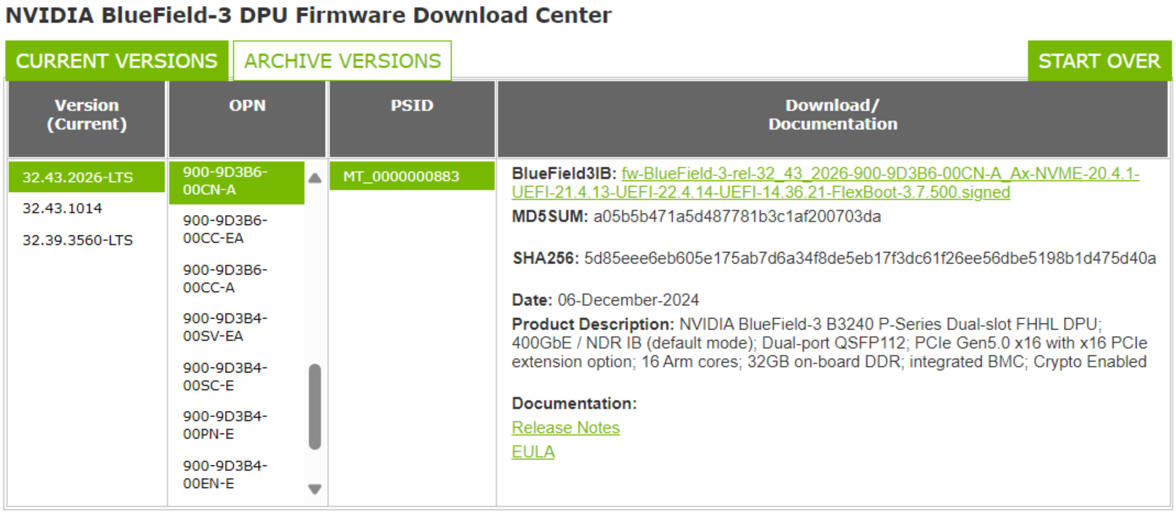
1. Navigate to the NVIDIA Networking Firmware Downloads page.
2. From the **NVIDIA BlueField DPU Firmware** table, select product line **NVIDIA BlueField-3** and click the **InfiniBand/Ethernet** network protocol.



NVIDIA BlueField DPU Firmware	
Product Line	Network Protocol
NVIDIA BlueField-3	InfiniBand/Ethernet
NVIDIA BlueField-2	InfiniBand/Ethernet

The Firmware for NVIDIA BlueField-3 page opens.

3. On the **NVIDIA BlueField-3 DPU Firmware Download Center** matrix, display the firmware information and documentation.
    1. Select the **CURRENT VERSIONS** tab.
    2. Choose the **Version (Current)**, **OPN** (ordering part number), and **PSID** options you want. For example,



NVIDIA BlueField-3 DPU Firmware Download Center

CURRENT VERSIONS | ARCHIVE VERSIONS | START OVER

Version (Current)	OPN	PSID	Download/ Documentation
32.43.2026-LTS 32.43.1014 32.39.3560-LTS	900-9D3B6-00CN-A 900-9D3B6-00CC-EA 900-9D3B6-00CC-A 900-9D3B4-00SV-EA 900-9D3B4-00SC-E 900-9D3B4-00PN-E 900-9D3B4-00EN-E	MT_0000000883	**BlueField3IB:** fw-BlueField-3-rel-32_43_2026-900-9D3B6-00CN-A_Ax-NVME-20.4.1-UEFI-21.4.13-UEFI-22.4.14-UEFI-14.36.21-FlexBoot-3.7.500.signed **MD5SUM:** a05b5b471a5d487781b3c1af200703da **SHA256:** 5d85eee6eb605e175ab7d6a34f8de5eb17f3dc61f26ee56dbe5198b1d475d40a **Date:** 06-December-2024 **Product Description:** NVIDIA BlueField-3 B3240 P-Series Dual-slot FHHL DPU; 400GbE / NDR IB (default mode); Dual-port QSFP112; PCIe Gen5.0 x16 with x16 PCIe extension option; 16 Arm cores; 32GB on-board DDR; integrated BMC; Crypto Enabled **Documentation:** Release Notes EULA

> **Note**
>
> PSID (Parameter-Set Identification) is a 16-ASCII character string embedded in the firmware image, which provides a unique identification for the firmware configuration. For more information, refer to Help in identifying the PSID of your Adapter card.

4. Click the **BlueField3IB** link to download the firmware BIN file.
5. After downloading the correct BlueField-3 NIC firmware, proceed with the installation steps.

To learn more about the BlueField-3 firmware, refer to NVIDIA BlueField-3 DPU NIC Firmware Release Notes.

Alternatively, you can use the NVIDIA firmware tools (MFT) to install the BlueField-3 NIC firmware:

- To download MFT, refer to the MFT download page.
- For installation details, refer to Updating Firmware for a Single Network Interface Card (NIC).

9.2.1. Resetting the BlueField-3 BMC Password

Note

After upgrading to EL9-24.12, reset the default password for the BlueField-3 BMC. The default password is 0penBmc.

Follow these steps to change the BMC password,

1. Download a current BFB file from NVIDIA DOCA Downloads.
2. Ensure the rshim package is installed and the rshim service is running.

   ```
   sudo dpkg -l | grep rshim
   sudo systemctl status rshim
   ```

3. Find the devices for any Blufield-3 cards present on the system.

   ```
   ls /dev/rshim*
   ```

4. Create a bf.cfg configurtion file with the following parameter values.

   ```
   NEW_BMC_PASSWORD="NewPassword"
   BMC_USER="root"
   BMC_REBOOT="yes"
   BMC_PASSWORD="0penBmc"
   UPDATE_BMC_FW="no"
   UPDATE_CEC="no"
   WITH_NIC_FW_UPDATE="no"
   UPDATE_DPU_OS="no"
   UPDATE_DPU_GOLDEN_IMAGE="no"
   UPDATE_NIC_FW_GOLDEN_IMAGE="no"
   ```

   These parameter values only update the password. If you also want to update the firmware, adjust the UPDATE_ parameter values based on your requirements.

5. Update the firmware for each rshim device found in step 3.

   ```
   sudo bfb-install --rshim rshim<N> --bfb <path_to_bfb_image> --config <path_to_bf.
   ↪cfg>
   ```

For more information, refer to

- Customizing BlueField Software Deployment Using bf.cfg
- Changing Default Password in the NVIDIA BlueField BMC Software documentation

Chapter 10. Installing with Kickstart

Kickstart provides a method of automating the installation process by providing a file which contains the answers to the questions that would be asked during installation. NVIDIA provides kickstart templates for all supported platforms in the following locations:

- https://repo.download.nvidia.com/baseos/el/el-files/9/el9-ks/dgx1-ks.cfg
- https://repo.download.nvidia.com/baseos/el/el-files/9/el9-ks/dgx2-ks.cfg
- https://repo.download.nvidia.com/baseos/el/el-files/9/el9-ks/dgx-a100-ks.cfg
- https://repo.download.nvidia.com/baseos/el/el-files/9/el9-ks/dgx-a800-ks.cfg
- https://repo.download.nvidia.com/baseos/el/el-files/9/el9-ks/dgx-h100-ks.cfg
- https://repo.download.nvidia.com/baseos/el/el-files/9/el9-ks/dgx-h200-ks.cfg
- https://repo.download.nvidia.com/baseos/el/el-files/9/el9-ks/dgxstation-ks.cfg
- https://repo.download.nvidia.com/baseos/el/el-files/9/el9-ks/dgxstation-a100-ks.cfg
- https://repo.download.nvidia.com/baseos/el/el-files/9/el9-ks/dgxstation-a800-ks.cfg

For more information on using kickstart files with Red Hat 9, refer to the 27.2. How Do You Perform a Kickstart Installation? section in Red Hat documentation.

In these files are tags that you must replace with your site specific information: language, keyboard, timezone, hostname, etc.

Each tag is of the form "CHANGE_YOUR_xxxx". These tags **must** be replaced with your specific information.

Once you have a kickstart file customized for your installation, place that file in a location that can be accessed by NFS, FTP, HTTP, or HTTPS.

After booting from the installation medium, when the grub menu appears, press e to edit the grub entry as you normally would to add the "nomodeset" option. In addition to "nomodeset", add an option in the format "inst.ks=<URL>".

For example:

```
inst.ks=https://192.168.1.2/kickstart/dgx-a100-cfg.ks
```

Lastly, after using these kickstart files – remember to reboot the system one final time to apply all settings.

Chapter 11. Configuring Storage

By default, the DGX System includes multiple SSDs in a RAID configuration (4 SSDs in the DGX-1, 8 or 16 SSDs in the DGX-2, and 3 SSDs in the DGX Station). These SSDs are intended for application caching, so NVIDIA recommends that you set up your own NFS storage for long term data storage. The following instructions describe how to mount the NFS onto the DGX System, and how to cache the NFS using the DGX SSDs for improved performance.

Make sure that you have an NFS server with one or more exports with data to be accessed by the DGX System, and that there is network access between the DGX System and the NFS server.

1. Install the NFS packages.

   ```
   $ sudo dnf install nfs-utils
   ```

2. Configure an NFS mount for the DGX.

   a. Edit the filesystem tables configuration.

      ```
      sudo vi /etc/fstab
      ```

   b. Add a new line for the NFS mount, using the local mount point of `/mnt`.

      ```
      <nfs_server>:<export_path> /mnt nfs rw,noatime,rsize=32768,wsize=32768,nolock,
      ↪tcp,intr,fsc,nofail 0 0
      ```

      `/mnt` is used here as an example mount point.

      Consult your Network Administrator for the correct values for `<nfs_server>` and `<export_path>`.

      The nfs arguments presented here are a list of recommended values based on typical use cases. However, “fsc” must always be included as that argument specifies use of FS-Cache.

   c. Save the changes.

3. Verify the NFS server is reachable.

   ```
   ping <nfs_server>
   ```

   Use the server IP address or the server name provided by your network administrator.

4. Mount the NFS export.

   ```
   sudo mount /mnt
   ```

   /mnt is the example mount point used in step 1.

5. Verify caching is enabled.

```
cat /proc/fs/nfsfs/volumes
```

Look for the text FSC=yes in the output. The NFS will be mounted and cached on the DGX System automatically upon subsequent reboot cycles.

Chapter 12. Upgrading

NVIDIA and Red Hat provide updates to the OS in the form of updated software packages between releases with security mitigations and bug fixes.

Important

Here is some important information you need to know **before** upgrading:

- An in-place upgrade from Red Hat Linux 8 to Red Hat Linux 9 with the DGX software stack installed is not supported.
- Before you install or perform the upgrade, refer to the *Release Notes* section for the latest Red Hat Linux version, known issues, and workarounds.

  To remain at the same RHEL release and prevent incompatibility between Linux kernel and GPU drivers, pin the RHEL release by using the `subscription-manager release --set=<release>` command. For example, the `subscription-manager release --set=9.3` command ties the system to RHEL 9.3.

You should evaluate the available updates in regular intervals and update the system by using the `sudo dnf update --nobest`command.

For a list of the known Common Vulnerabilities and Exposures (CVEs), including those that can be resolved by updating the OS software, refer to the Red Hat Security Updates

Note

You are responsible for upgrading the software on the DGX system to install the updates from these sources.

If updates are available, you can obtain the package upgrades by running:

```
sudo dnf update -nobest
```

Upgrades to the NVIDIA Graphics Drivers for Linux requires a restart to complete the kernel upgrade. If you upgrade the NVIDIA Graphics Drivers for Linux without restarting the DGX system, when you run the `nvidia-smi`command, an error message is displayed.

```
nvidia-smi
Failed to initialize NVML: Driver/library version mismatch
```

12.1. Upgrading the OS and DGX Software

This section provides information for upgrading your DGX system and optionally upgrading to a different GPU branch.

12.1.1. Upgrading the Software without Moving to a New Driver Branch

To upgrade your DGX system with the latest Red Hat Linux upgrades, run the following command:

```
sudo dnf update -y --nobest
```

12.1.2. Updating the Software and Moving to a New Driver Branch on non-NVSwitch Systems

This procedure applies to DGX-1, DGX-2, DGX Station. and DGX Station A100 systems.

1. Issue the following to remove the current driver package and install the new driver package.

```
sudo dnf remove -y nv-persistence-mode libnvidia-nscq-<current driver version>
sudo dnf module remove --all -y nvidia-driver
sudo dnf module reset -y nvidia-driver
sudo dnf module install -y nvidia-driver:<new driver version>/{default,src}
sudo dnf install -y nv-persistence-mode libnvidia-nscq-<new driver version>
sudo dnf update -y --nobest
```

1. For DGX Station A100 only - Install additional required DGX Station A100 packages. These packages must be installed after the nvidia-driver module. sudo dnf install nvidia-conf-xconfig nv-docker-gpus

   ```
   sudo dnf install nvidia-conf-xconfig nv-docker-gpus
   ```

2. Reboot the system.

   ```
   sudo reboot
   ```

12.1.3. Updating the Software and Moving to a New Driver Branch on NVSwitch Systems

This procedure applies to DGX-2, DGX A100, and DGX A800 systems.

1. Run the following commands to remove the current driver package and install the new driver package:

```
sudo dnf remove -y nv-persistence-mode libnvidia-nscq-<current driver version> nvidia-
↪fm-enable
sudo dnf module remove --all -y nvidia-driver
sudo dnf module reset -y nvidia-driver
sudo dnf module install -y nvidia-driver:<new driver version>/{fm,src}
sudo dnf install -y nv-persistence-mode libnvidia-nscq-<new driver version> nvidia-fm-
↪enable
sudo dnf update -y --nobest
```

1. Reboot the system.

   ```
   sudo reboot
   ```

12.2. Changing only the NVIDIA Driver Branch

To switch driver branches, you must first remove the existing branch before installing the new branch:

1. Remove and clear the existing stream:

   ```
   sudo dnf module remove --all nvidia-driver
   sudo dnf module reset nvidia-driver
   ```

2. Follow the "Install NVIDIA CUDA driver" section to install the new driver branch.
3. If the nvidia-peer-memory-dkms driver is installed it must be reinstalled to match the new driver branch:

   ```
   sudo dnf reinstall -y nvidia-peer-memory-dkms
   ```

12.3. Installing or Upgrading to a Newer CUDA Toolkit Release

> **Important**
>
> Before you install or upgrade to any CUDA Toolkit release, ensure the release is compatible with the driver that is installed on the system. Refer to CUDA Compatibility for more information and a compatibility matrix.

Only DGX Station and DGX Station A100 have a CUDA Toolkit release installed by default. DGX servers are intended to be shared resources that use containers and do not have CUDA Toolkit installed by default. However, you have the option to install a qualified CUDA Toolkit release.

Although all CUDA Toolkit releases are supported that interoperate with the installed driver, DGX releases might include a default CUDA Toolkit release that might not be the most recently released version. Unless you must use a new CUDA Toolkit version that contains the new features, we recommend that you remain on the default version that is included in the DGX RHEL9 release. Refer to the *Release Notes* for the default CUDA Toolkit release.

12.3.1. Checking the Currently Installed CUDA Toolkit Release

Here is some information about the prerequisite to determine the CUDA Toolkit release that you currently have installed.

> **Important**
>
> The CUDA Toolkit is not installed on DGX servers by default, and if you try to run the following command, no installed package will be listed

Before you install a new CUDA Toolkit release, to check the currently installed release, run the following command:

```
sudo dnf list installed "cuda-toolkit-*"
```

The following output shows that CUDA Toolkit 12.0 is installed:

```
Updating Subscription Management repositories.

Installed Packages
cuda-toolkit-12-0.x86_64 12.0.0-1 @CUDA
cuda-toolkit-12-0-config-common.noarch 12.0.107-1 @CUDA
cuda-toolkit-12-config-common.noarch 12.0.107-1 @CUDA
cuda-toolkit-config-common.noarch
```

12.3.2. Determining the New Available CUDA Toolkit Releases

These steps help you determine which new CUDA Toolkit releases are available. To see the new available CUDA Toolkit releases:

```
sudo dnf search "cuda-toolkit-*"
Updating Subscription Management repositories.
Last metadata expiration check: 1:47:39 ago on Wed 18 Jan 2023 08:10:38 AM PST.
======================================================= Name Matched: cuda-toolkit-*
↪=======================================================
cuda-toolkit-11-7.x86_64 : CUDA Toolkit 11.7 meta-package
cuda-toolkit-11-7-config-common.noarch : Common config package for CUDA Toolkit 11.7.
cuda-toolkit-11-8.x86_64 : CUDA Toolkit 11.8 meta-package
cuda-toolkit-11-8-config-common.noarch : Common config package for CUDA Toolkit 11.8.
cuda-toolkit-11-config-common.noarch : Common config package for CUDA Toolkit 11.
cuda-toolkit-12-0.x86_64 : CUDA Toolkit 12.0 meta-package
cuda-toolkit-12-0-config-common.noarch : Common config package for CUDA Toolkit 12.0.
cuda-toolkit-12-config-common.noarch : Common config package for CUDA Toolkit 12.
cuda-toolkit-config-common.noarch : Common config package for CUDA Toolkit.
```

The output shows that 11.7, 11.8, and 12.0 are the possible CUDA Toolkit versions that can be installed.

12.3.3. Installing the CUDA Toolkit or Upgrading Your CUDA Toolkit to a Newer Release

You can install or upgrade your CUDA Toolkit to a newer release.

To install or upgrade the CUDA Toolkit, run the following command:

```
sudo dnf install cuda-toolkit-12-0
```

Note

Version 12.0 is shown as an example - replace the value with the version you wish to install.

12.4. Installing GPUDirect Storage Support

NVIDIA® Magnum IO GPUDirect® Storage (GDS) enables a direct data path for direct memory access (DMA) transfers between GPU memory and storage, which avoids a bounce buffer through the CPU.

12.4.1. Installing nvidia-gds

To use GDS, perform the following steps:

1. Populate the `${NVIDIA_DRV_VERSION}`variable
2. Install `nvidia-gds`with the correct dependencies:

   ```
   sudo install nvidia-gds-{ver} nvidia-dkms-${NVIDIA_DRV_VERSION}-server
   ```

Use the CUDA Toolkit version number in place of `<ver>`; for example, `12-0`

Chapter 13. Managing Self-Encrypting Drives

The NVIDIA DGX OS software supports the ability to manage self-encrypting drives (SEDs), including setting an Authentication Key for locking and unlocking the drives on NVIDIA DGX B200, DGX H100/H200, DGX A100, DGX A800, DGX Station A100, and DGX-2 systems.

You can manage only the SED data drives. The software cannot be used to manage OS drives even if they are SED-capable.

13.1. Overview

The SED management software is in the `nv-disk-encrypt` package.

The software supports the following configurations:

- NVIDIA DGX B200, DGX H100/H200, DGX A100, DGX A800, DGX Station A100, and DGX-2 systems where all data drives are self-encrypting drives.
- Only SEDs used as data drives are supported.

  The software will not manage SEDs that are OS drives.

The software provides the following functionality:

- Identifies eligible drives on the system.
- Allows you to you assign Authentication Keys (passwords) for each SED as part of the initialization process.
  - Alternatively, the software can generate random passwords for each drive.
  - The passwords are stored in a password-protected vault on the system.
- Once initialized, SEDs are locked upon power loss, such as a system shutdown or drive removal.

  Locked drives get unlocked after power is restored and the root file system is mounted.
- Provides functionality to export the vault.
- Provides functionality for erasing the drives.
- Provides the ability to revert the initialization.

13.2. Installing the Software

Use the package manager to install the `nv-disk-encrypt` package and, optionally, the TPM2 tools package, and reboot the system. You need the TPM tools package if you plan to use the TPM2 to store security keys.

1. Update the packages.

   ```
   sudo dnf update
   ```

2. Install `nv-disk-encrypt`.

   ```
   sudo dnf install -y nv-disk-encrypt
   ```

3. (Optional) Install the TPM tools package.

   - For DGX A100, DGX A800, DGX Station A100, or DGX H100/H200, install the `tpm2-tools` package.

   ```
   sudo dnf install -y tpm2-tools
   ```

   - For DGX-2, install the `tpm-tools` package.

   ```
   sudo dnf install -y tpm-tools
   ```

4. Reboot.

   ```
   sudo reboot
   ```

If you plan to use TPM2, enable it. Refer to *Configuring Trusted Computing* for more information.

13.3. Configuring Trusted Computing

Here is some information about the controls that are required to configure Trusted Computing (TC).

The DGX system BIOS provides setup controls for configuring the following TC features:

- Trusted Platform Module

  The NVIDIA DGX B200, DGX H100/H200, DGX A100, DGX A800, and DGX Station A100 incorporate Trusted Platform Module 2.0 (TPM 2.0). The DGX-2 incorporates a TPM module. These modules can be enabled from the system BIOS and used in conjunction with the nv-disk-encrypt tool. After being enabled, the `nv-disk-encrypt` tool uses the TPM for encryption and stores the vault and SED authentication keys on the TPM instead of on the file system. Using the TPM is preferred because this allows the vault data to persist even if the system is reimaged.

- Block SID

  Certain drives shipped with the DGX systems support the Block SID authentication feature. Block SID authentication prevents malicious actors from taking ownership of drives and blocks others from using the drives. By default, the DGX BIOS will send the Block SID request. On such setups, you will need to enable the Disable Block Sid feature in the BIOS before proceeding with the initialization steps.

Note

Enabling the “Disable Block SID” option is only valid for one reboot, so if drive encryption needs to be enabled again, then the feature needs to be enabled in BIOS as well.

13.3.1. Determining Whether Drives Support SID

The drive model is a good indicator of whether the drive supports this feature. Issue the following and look for one of the following model strings:

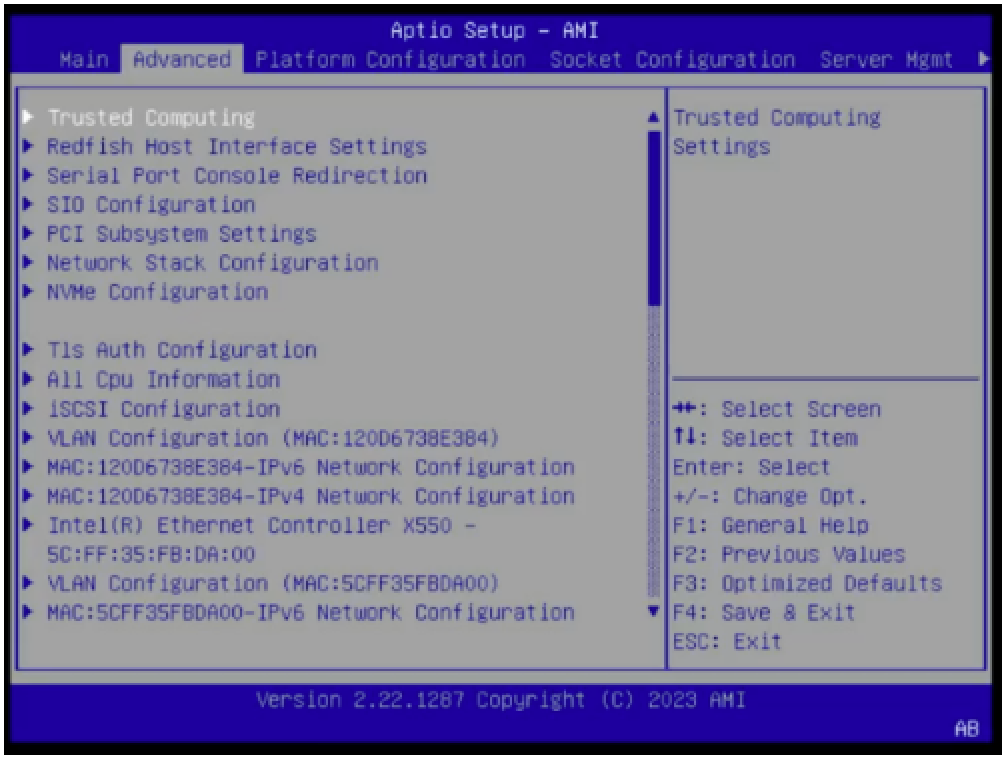
- KCM6DRUL3T84
- KCM6DRUL7T68
- MZQLB7T6HMLA-00007

```
sudo nvme list

Node          SN                            Model
------------- ----------------------------- -----
/dev/nvme0n1  70H0A0AHTTHR KCM6DRUL3T84 ...
/dev/nvme1n1  70H0A007TTHR KCM6DRUL3T84
```

13.3.2. Enabling the TPM and Preventing the BIOS from Sending Block SID Requests

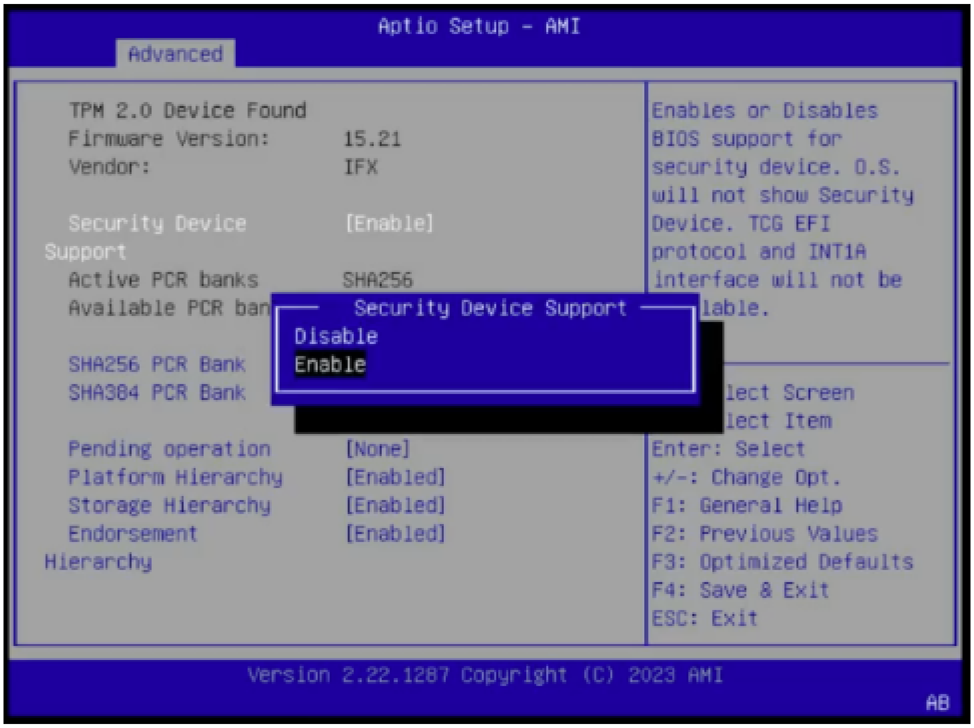
This section provides instructions to enable the TPM and prevent the SBIOS from sending Block SID requests. Each task is independent, so you can select which task to complete.

1. Reboot the system, then press [Del] or [F2] at the NVIDIA splash screen to enter the BIOS Setup.
2. Navigate to the **Advanced** tab on the top menu, then scroll to **Trusted Computing** and press [Enter].



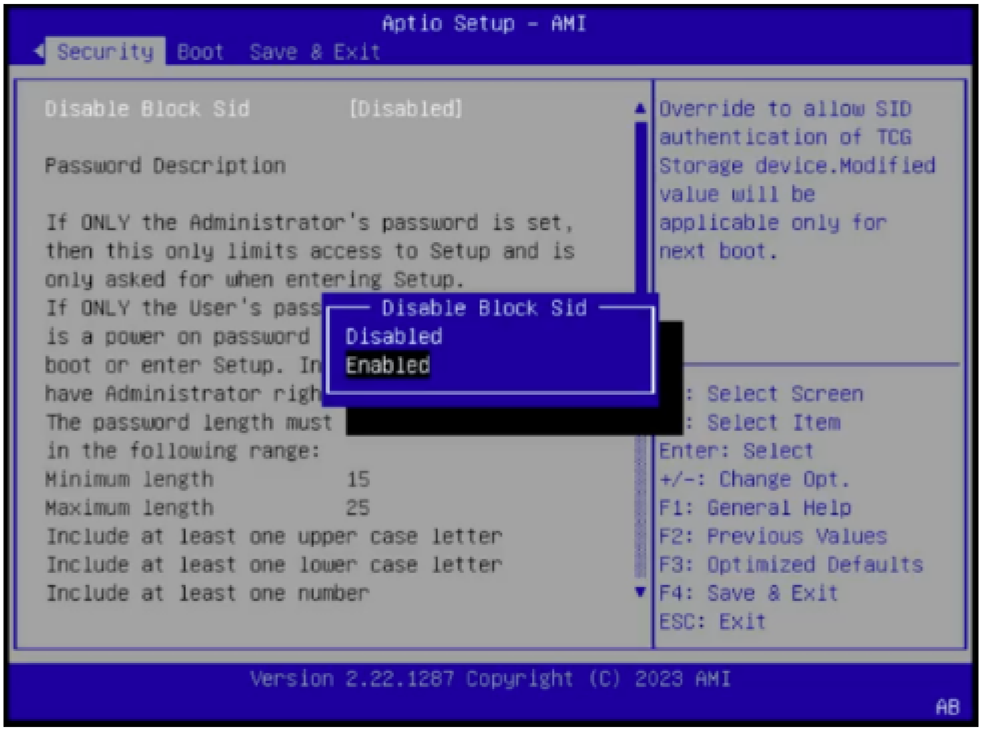
```
                          Aptio Setup - AMI
   Main  Advanced  Platform Configuration  Socket Configuration  Server Mgmt ▶

▶ Trusted Computing                                   ▲ Trusted Computing
▶ Redfish Host Interface Settings                       Settings
▶ Serial Port Console Redirection
▶ SIO Configuration
▶ PCI Subsystem Settings
▶ Network Stack Configuration
▶ NVMe Configuration

▶ Tls Auth Configuration
▶ All Cpu Information
▶ iSCSI Configuration                                   →←: Select Screen
▶ VLAN Configuration (MAC:120D6738E384)                 ↑↓: Select Item
▶ MAC:120D6738E384-IPv6 Network Configuration           Enter: Select
▶ MAC:120D6738E384-IPv4 Network Configuration           +/-: Change Opt.
▶ Intel(R) Ethernet Controller X550 -                   F1: General Help
  5C:FF:35:FB:DA:00                                     F2: Previous Values
▶ VLAN Configuration (MAC:5CFF35FBDA00)                 F3: Optimized Defaults
▶ MAC:5CFF35FBDA00-IPv6 Network Configuration         ▼ F4: Save & Exit
                                                        ESC: Exit

                Version 2.22.1287 Copyright (C) 2023 AMI
                                                                        AB
```

- ▶ To enable TPM, scroll to **Security Device Support** and switch the setting to **Enabled**.



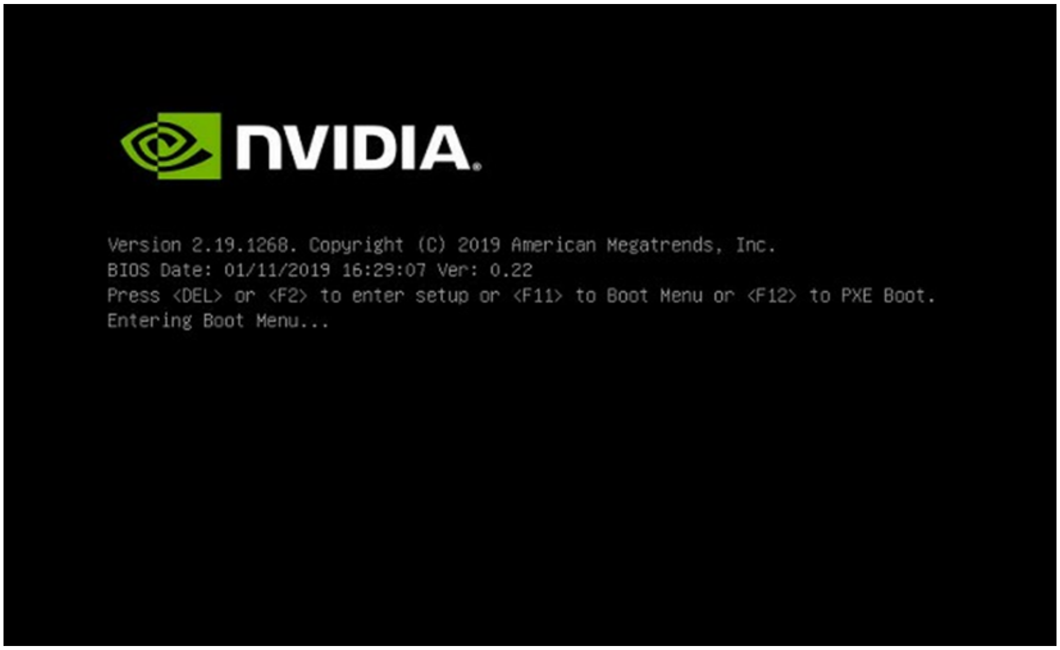
```
                          Aptio Setup - AMI
        Advanced

     TPM 2.0 Device Found                          Enables or Disables
     Firmware Version:        15.21                BIOS support for
     Vendor:                  IFX                  security device. O.S.
                                                   will not show Security
     Security Device          [Enable]             Device. TCG EFI
   Support                                         protocol and INT1A
     Active PCR banks         SHA256               interface will not be
     Available PCR ban┌── Security Device Support ──┐lable.
                      │ Disable                      │
     SHA256 PCR Bank  │ Enable                       │
     SHA384 PCR Bank  └──────────────────────────────┘lect Screen
                                                      lect Item
     Pending operation        [None]               Enter: Select
     Platform Hierarchy       [Enabled]            +/-: Change Opt.
     Storage Hierarchy        [Enabled]            F1: General Help
     Endorsement              [Enabled]            F2: Previous Values
   Hierarchy                                       F3: Optimized Defaults
                                                   F4: Save & Exit
                                                   ESC: Exit

                Version 2.22.1287 Copyright (C) 2023 AMI
                                                                        AB
```

- ▶ To disable Block SID, navigate to the **Security** tab and scroll to **Disable Block Sid**, then switch to **Enabled**.



```
                         Aptio Setup - AMI
  Security  Boot  Save & Exit

 Disable Block Sid          [Disabled]          Override to allow SID
                                                authentication of TCG
 Password Description                           Storage device.Modified
                                                value will be
 If ONLY the Administrator's password is set,   applicable only for
 then this only limits access to Setup and is   next boot.
 only asked for when entering Setup.
 If ONLY the User's pass ── Disable Block Sid ──
 is a power on password   Disabled
 boot or enter Setup. In  Enabled
 have Administrator righ                        : Select Screen
 The password length must                       : Select Item
 in the following range:                        Enter: Select
 Minimum length             15                  +/-: Change Opt.
 Maximum length             25                  F1: General Help
 Include at least one upper case letter         F2: Previous Values
 Include at least one lower case letter         F3: Optimized Defaults
 Include at least one number                    F4: Save & Exit
                                                ESC: Exit

            Version 2.22.1287 Copyright (C) 2023 AMI
                                                                AB
```

3. When changing the BIOS settings, make sure to select the **Save & Exit** option to continue the boot process.



```
Version 2.19.1268. Copyright (C) 2019 American Megatrends, Inc.
BIOS Date: 01/11/2019 16:29:07 Ver: 0.22
Press <DEL> or <F2> to enter setup or <F11> to Boot Menu or <F12> to PXE Boot.
Entering Boot Menu...
```

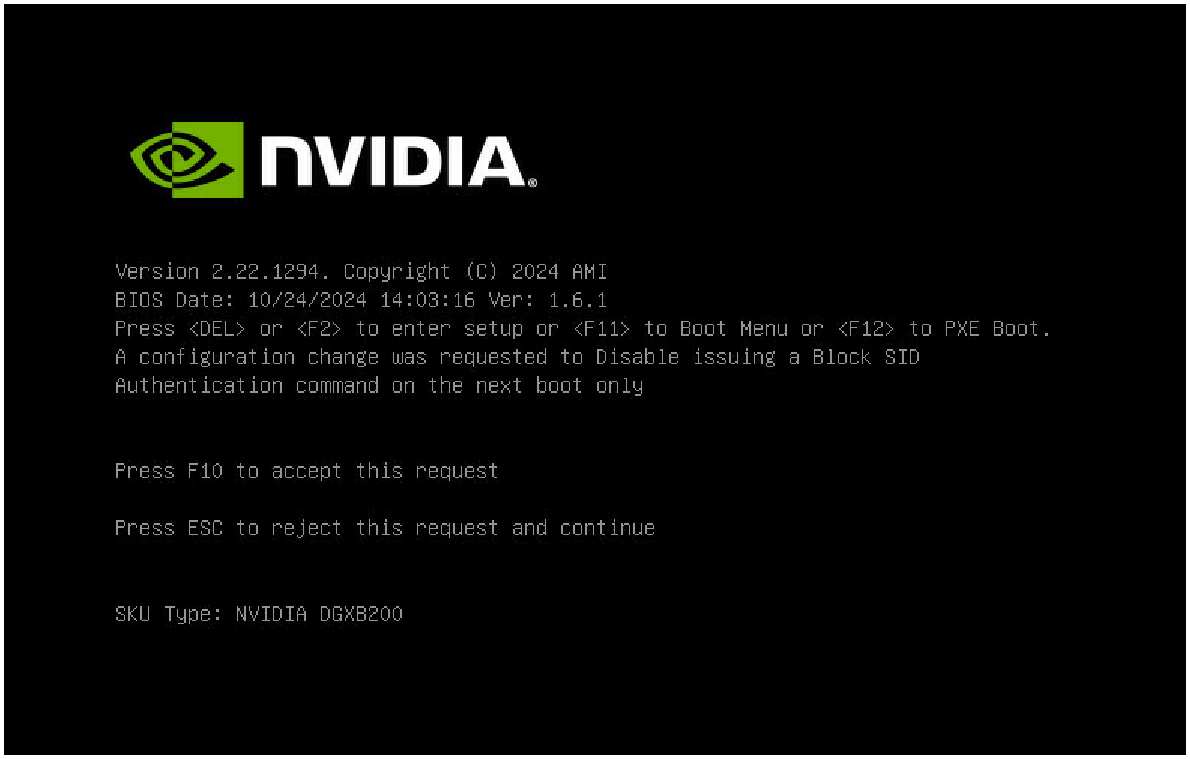
4. If the BIOS is modified to the **Enabled** setting for the **Security Device Support** or **Disable Block Sid** option and the following messages do not appear automatically after reboot:

```
Press F10 to accept this request
Press ESC to reject this request and continue
```

Press F10 at the prompt: `Press <DEL> or <F2> to enter setup or <F11> to Boot Menu`

```
or <F12> to PXE Boot.
```

The messages as mentioned above appear.



```
Version 2.22.1294. Copyright (C) 2024 AMI
BIOS Date: 10/24/2024 14:03:16 Ver: 1.6.1
Press <DEL> or <F2> to enter setup or <F11> to Boot Menu or <F12> to PXE Boot.
A configuration change was requested to Disable issuing a Block SID
Authentication command on the next boot only


Press F10 to accept this request

Press ESC to reject this request and continue


SKU Type: NVIDIA DGXB200
```

5. Press F10 to accept the BIOS settings and continue to configure SED or disk encryption.

13.4. Initializing the System for Drive Encryption

Here is some information about how to initialize the system for drive encryption.

> **Note**
>
> Before initializing drive encryption, review the information in *Configuring Trusted Computing* and follow the configuration instructions if needed.

Initialize the system for drive encryption using the nv-disk-encrypt command.

```
sudo nv-disk-encrypt init [-k <your-vault-password>] [-f <path/to/json-file>] [-g] [-
↪r]
```

Here is a list of the options:

- **-k**: Lets you create the vault password in the command.

  Otherwise, the software will prompt you to create a password before proceeding.
- **-f**: Lets you specify a JSON file that contains a mapping of passwords to drives.

  Refer to “Example 1: Passing in the JSON File” for further instructions.

- **-g**: Generates random salt values (stored in `/etc/nv-disk-encrypt/.dgxenc.salt`) for each drive password.

  Salt values are characters added to a password for enhanced password security. NVIDIA strongly recommends using this option for best security, otherwise the software will use a default salt value instead of a randomly generated one.

- **-r**: Generates random passwords for each drive.

  This avoids the need to create a JSON file or the need to enter a password one by one during the initialization.

13.5. Enabling Drive Locking

After initializing the system for SED management, issue the following command, which uses the `nv-disk-encrypt` command to enable drive locking.

```
sudo nv-disk-encrypt lock
```

After initializing the system and enabling drive locking, the drives will become locked when they lose power. The system will automatically unlock each drive when power is restored to the system and the system is rebooted.

13.6. Initialization Examples

This section provides some initialization examples.

13.6.1. Example 1: Passing in the JSON File

The following instructions in this section describe a method to specify the drive/password mapping ahead of time. This method is useful for initializing several drives at a time and avoids the need to enter the password for each drive after issuing the initialization command, or if you want control of the passwords.

Refer to the following for more information:

- *Determining Which Drives Can be Managed as Self-Encrypting*
- *Creating the Drive/Password Mapping JSON Files and Using it to Initialize the System*

13.6.1.1 Determining Which Drives Can be Managed as Self-Encrypting

Here is some information about how you can determine which drives can be managed as self-encrypting.

Review the storage layout of the DGX system to determine which drives are eligible to be managed as SEDs.

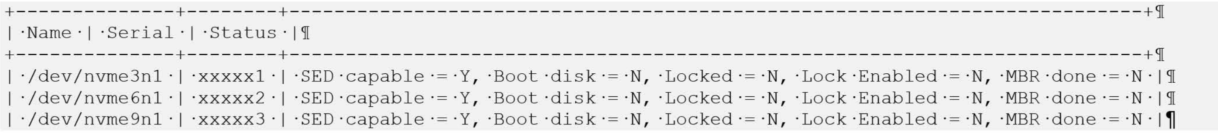
```
sudo nv-disk-encrypt info
```

The default output shows which drives can be used for encryption and which drives cannot. The following status information is provided:

- SED capable: Is this a self-encrypting drive?
- Boot disk: Is this drive currently being used as a boot drive? Does it contain the root filesystem?
- Locked: Is this drive currently in the locked state? Is it able to accept I/O?. It can only be in this state after the following conditions have been met:
  - Locking has been enabled (nv-disk-encrypt init, followed by `nv-disk-encrypt init` lock)
  - The drive is coming back from power-off.
  - The user queries this state prior to it being (automatically) unlocked.
- Lock Enabled: Are locks enabled on this drive? It will be in this state after initialization (`nv-disk-encrypt init`).
- MBR done: This setting is only relevant for drives that support MBR shadowing. On drives that support this feature, this will report 'Y' after initialization (`nv-disk-encrypt init`)

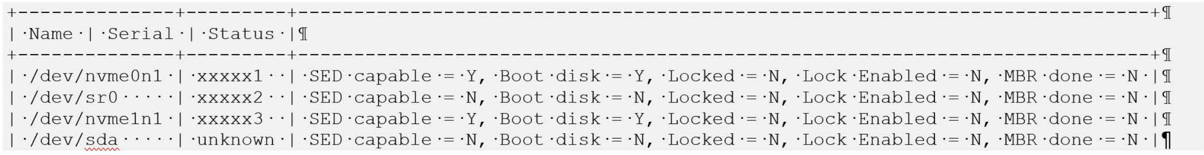
MBR done: This setting is only relevant for drives that support MBR shadowing. On drives that support this feature, this will report 'Y' after initialization (`nv-disk-encrypt init`)

The following example output snippet shows drives than can be used for encryption. Notice SED capable = Y and Boot disk = N.



```
+--------------+--------+--------------------------------------------------------------------------------+¶
|·Name·|·Serial·|·Status·|¶
+--------------+--------+--------------------------------------------------------------------------------+¶
|·/dev/nvme3n1·|·xxxxx1·|·SED·capable·=·Y,·Boot·disk·=·N,·Locked·=·N,·Lock·Enabled·=·N,·MBR·done·=·N·|¶
|·/dev/nvme6n1·|·xxxxx2·|·SED·capable·=·Y,·Boot·disk·=·N,·Locked·=·N,·Lock·Enabled·=·N,·MBR·done·=·N·|¶
|·/dev/nvme9n1·|·xxxxx3·|·SED·capable·=·Y,·Boot·disk·=·N,·Locked·=·N,·Lock·Enabled·=·N,·MBR·done·=·N·|¶
```

The following example output snippet shows drives than cannot be used for encryption. Notice SED capable = Y and Boot disk = Y, or SED capable = N.



```
+--------------+---------+--------------------------------------------------------------------------------+¶
|·Name·|·Serial·|·Status·|¶
+--------------+---------+--------------------------------------------------------------------------------+¶
|·/dev/nvme0n1·|·xxxxx1··|·SED·capable·=·Y,·Boot·disk·=·Y,·Locked·=·N,·Lock·Enabled·=·N,·MBR·done·=·N·|¶
|·/dev/sr0·····|·xxxxx2··|·SED·capable·=·N,·Boot·disk·=·N,·Locked·=·N,·Lock·Enabled·=·N,·MBR·done·=·N·|¶
|·/dev/nvme1n1·|·xxxxx3··|·SED·capable·=·Y,·Boot·disk·=·Y,·Locked·=·N,·Lock·Enabled·=·N,·MBR·done·=·N·|¶
|·/dev/sda·····|·unknown·|·SED·capable·=·N,·Boot·disk·=·N,·Locked·=·N,·Lock·Enabled·=·N,·MBR·done·=·N·|¶
```

Alternatively, you can specify the output be presented in JSON format by using the `-j` option.

```
sudo nv-disk-encrypt info -j
```

In this case, drives that can be used for encryption are indicate by the following:

```
"sed_capable": true "used_for_boot": false
```

And drives that cannot be used for encryption are indicated by one of the following:

```
"sed_capable": true "used_for_boot": true
```

Or

```
"sed_capable": false
```

13.6.1.2 Creating the Drive/Password Mapping JSON Files and Using it to Initialize the System

You can initialize the system by creating the drive and password map the JSON files.

1. Create a JSON file that lists all the eligible SED-capable drives that you want to manage.

   > **Note**
   >
   > These are the list of drives that you obtained from *Determining Which Drives Can be Managed as Self-Encrypting*

   The following example shows the format of the JSON file.

   ```
   {
   "/dev/nvme2n1": "<your-password>",
   "/dev/nvme3n1": "<your-password>",
   "/dev/nvme4n1": "<your-password>",
   "/dev/nvme5n1": "<your-password>",
   }
   ```

   - Ensure that you follow the syntax exactly.
   - Passwords must consist of only upper-case letters, lower-case letters, digits, and/or the following special characters: ~ : @ % ^ + = _ ,

2. Initialize the system and then enable locking.

   The following command assumes you have placed the JSON file in the /tmp directory.

   ```
   sudo nv-disk-encrypt init -f /tmp/<your-file>.json -g
   sudo nv-disk-encrypt lock
   ```

   When prompted, enter a password for the vault.

   Passwords must consist of only upper-case letters, lower-case letters, digits, and/or the following special characters: ~ : @ % ^ + = _ ,

13.6.2. Example 2: Generating Random Passwords

The commands in this topic use the `-k` and `-r` options so that you are not prompted to enter passwords. You pass the vault password into the command and then the command instructs the tool to generate random passwords for each drive.

The vault password must consist of only upper-case letters, lower-case letters, digits, and/or the following special-characters: ~ : @ % ^ + = _ ,

```
sudo nv-disk-encrypt init -k <your-vault-password> -g -r
sudo nv-disk-encrypt lock
```

13.6.3. Example 3: Specifying Passwords One at a Time When Prompted

If there are a small number of drives, or you do not want to create a JSON file, issue the following command.

```
sudo nv-disk-encrypt init -g
sudo nv-disk-encrypt lock
```

The software prompts you to enter a password for the vault and then a password for each eligible SED.

Passwords must consist of only upper-case letters, lower-case letters, digits, and/or the following special characters: ~ : @ % ^ + = _ ,

13.7. Disabling Drive Locking

To disable drive locking at any time after you initialize, run the following command: $ sudo nv-disk-encrypt disable

This command disables locking on all drives. You can run the initial set up again at any time after this process is complete.

13.8. Enabling Drive Locking

After initializing the system for SED management, issue the following command, which uses the nv-disk-encrypt command to enable drive locking.

```
sudo nv-disk-encrypt lock
```

After initializing the system and enabling drive locking, the drives will become locked when they lose power. The system will automatically unlock each drive when power is restored to the system and the system is rebooted.

13.9. Exporting the Vault

Here is some information about how to export the vault.

To export all drive keys out to a file, use the export function. This requires that you pass in the vault password.

```
sudo nv-disk-encrypt export -k yourvaultpassword
Writing vault data to /tmp/secrets.out
```

13.10. Erasing Your Data

Here is some information about how you can erase your data.

> **Caution**
>
> Be aware that all data will be lost when executing this. On the DGX B200 and DGX H100/H200 systems, these drives generally form a RAID 0 array, and this array will also be destroyed when you perform an erase.

After initializing the system for SED management, use the nv-disk-encrypt command to erase data on your drives after stopping cachefilesd and unmounting the RAID array as follows.

1. Completely stop the RAID.

   ```
   systemctl stop cachefilesd
   sudo umount /raid
   sudo mdadm --stop /dev/md1
   ```

2. Perform the erase.

   ```
   sudo nv-disk-encrypt erase
   ```

This command does the following:

- Sets the drives in an unlocked state.
- Disables locking on the drives.
- Removes the RAID 0 array configuration.

To rebuild the RAID array, issue the following command:

```
sudo /usr/bin/configure_raid_array.py -c -f
```

13.11. Clearing the TPM

If you've lost the password to your TPM, you will not be able to access its contents. In this case, the only way to regain access to the TPM is to clear the TPM's contents. After clearing the TPM, you will need to re-initialize the vault and SED authentication keys.

1. Reboot the system, then press [Del] or [F2] at the NVIDIA splash screen to enter the BIOS Setup.
2. Navigate to the Advanced tab on the top menu, scroll to Trusted Computing, and press [Enter].
3. Clear TPM2.
   1. Scroll to Trusted Computing and press [Enter].
   2. Scroll to Pending Operation and press [Enter].
   3. Select TPM Clear at the Pending Operation popup and press [Enter].
4. Save and exit the BIOS Setup.

13.12. Changing Disk Passwords, Adding Disks, or Replacing Disks

The same steps are needed for changing or rotating passwords, adding disks, or replacing disks.

1. Disable SED management.

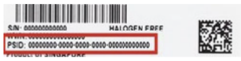
   ```
   sudo nv-disk-encrypt disable
   ```

2. Add or replace drives as needed and then rebuild the RAID array. Refer to your system's Service Manual for more information.
3. Enable SED management and assign passwords per the instructions in *Initializing the System for Drive Encryption*.

13.13. Recovering From Lost Keys

NVIDIA recommends backing up your keys and storing them in a secure location. If you lost the key used to initialize and lock your drives, you will not be able to unlock the drive again. If this happens, the only way to recover is to perform a factory-reset, which will result in a loss of data.

SED drives come with a PSID printed on the label; this value can only be obtained by physically examining the drive as exemplified in the following image.



Specify the PSID to reset the drive using the following sedutil-cli command:

```
sudo sedutil-cli ----yesIreallywanttoERASEALLmydatausingthePSID yourdrivesPSID /dev/
↪nvme3n1
```

Chapter 14. Known Issues

This section provides summaries of the issues in the DGX Software for Red Hat Enterprise Linux.

14.1. Symbolic Links Removed During nvidia-mig-manager Upgrade

Issue

During the upgrade of the `nvidia-mig-manager` package to version 0.10.1, the following two symbolic links might be inadvertently removed.

```
/etc/nvidia-mig-manager/config.yaml
/etc/nvidia-mig-manager/hooks.yaml
```

Workaround

To resolve this issue, run the following commands.

```
sudo dnf remove nvidia-mig-manager
sudo dnf install nvidia-mig-manager
```

14.2. Red Hat Installer Fails to Delete Partitions

Issue

When performing disk partitioning with the Red Hat installation utility, the utility can fail to delete partitions that were previously used by other operating systems.

Workaround

To manually wipe the devices, perform the following steps.

1. Click **Done** twice from the **Manual Partitioning** menu.

   You returned to the main menu for the installation utility.

2. Press Ctrl+Alt+F2 to use a shell in a different virtual console.

   If the installation utility does not respond to the keystrokes, add a hotkey that sends Ctrl+Alt+F2.

   1. Select **Hot Keys** > **Add Hot Keys**.

      The **User Defined Macros** window opens.

   2. Click **Add**.

      The **Add Macro** window opens.

   3. Press Ctrl+Alt+F2. Ensure that the key sequence appears in the text field and then click **Insert**.

   4. Click **Close** on the **User Defined Macros** window.

   5. Select **Hot Keys** > **Ctrl+Alt+F2** to use a different virtual console.

   Run the remaining commands in the virtual console.

3. Stop RAID devices.

   1. Run the `lsblk` command.

   2. If the command output includes any md devices, stop the devices:

      ```
      mdadm --stop /dev/md<device-id>
      ```

4. Run the *wipefs* command for all the drives:

   ```
   wipefs -a /dev/nvme0n1
   wipefs -a /dev/nvme1n1
   ...
   ```

5. Reboot the machine and restart the installation process.

14.3. Virtualization Not Supported

Issue

Virtualization technology, such as ESXi hypervisors or kernel-based virtual machines (KVM), is not an intended use case on DGX systems and has not been tested.

14.4. NVSM Service and Fabric Manager Service Reported as Inactive

Platform

DGX H100 System, A100 System, and A100 Station with EL9-24.06

Issue

After EL9-24.06 upgrade and a system reboot, the status of `nvsm.service` and `nvidia-fabricmanager.service` shows `inactive (dead)` when you run `systemctl status nvsm` and `systemctl status nvidia-fabricmanager`, respectively.

```
$ sudo systemctl status nvsm
...
nvsm.service - NVIDIA System Management service suite
  Loaded: loaded (/usr/lib/systemd/system/nvsm.service; enabled; preset: disabled)
  Active: inactive (dead)
...
```

```
$ sudo systemctl status nvidia-fabricmanager
...
nvidia-fabricmanager.service - NVIDIA fabric manager service
  Loaded: loaded (/usr/lib/systemd/system/nvidia-fabricmanager.service; enabled;
↪preset: disabled)
  Active: inactive (dead)
...
```

Workaround

The `nvsm.service` service manages the start and stop of services running under NVSM. Because the NVSM services are operating normally and NVSM is fully functioning, you can ignore the `inactive` status of `nvsm.service`. To fix the `nvsm.service` status issue, run the `systemctl start nvsm` command after a system reboot.

However, the `nvidia-fabricmanager.service` service remains inactive. To resolve this issue, manually start the service by running the `systemctl start nvidia-fabricmanager.service` command.

Explanation

After a system reboot on DGX systems running the GPU Driver Release 550 or newer, `nvsm.service` and `nvidia-fabricmanager.service` appear inactive because `systemd` finds a dependency on `nvidia-fabricmanager.service` during startup. The circular dependency between `nvsm.service` and `nvidia-fabricmanager.service` makes one service wait for the other and prevents the services from starting.

14.5. Excessive OpenSM Log Growth Causing DGX Systems to Become Inoperable

Issue

An exceptionally large `/var/log/opensm.log` file can cause DGX systems to become inoperable.

Explanation

During the installation process of the MLNX_OFED or DOCA OFED software, the `opensm` package is also installed. By default, OpenSM is disabled. On systems where OpenSM is enabled, the `/etc/logrotate.d/opensm` file should be configured to include the following options to manage the size of the `opensm.log` file:

- The maximum size of log files for log rotation, such as `maxsize 10M` or `maxsize 100M`
- The rotation duration, such as `daily`, `weekly`, or `monthly`

Not specifying the two configuration options might result in an exceptionally large `/var/log/opensm.log` file that can cause DGX systems to become inoperable. For more information about OpenSM network topology, configuration, and enablement, refer to the NVIDIA OpenSM documentation.

14.6. Reboot Hang after Configuring RAID

Platform

DGX H100 System with EL9-23.08 and RHEL 9.1 or 9.2

Issue

After installing the DGX H100 Configurations group and configuring RAID with the `sudo /usr/bin/configure_raid_array.py -c -f -5` command and rebooting, the system can hang and display console messages like the following example:

```
...
[ 1944.542085] md: md124 stopped.
[ 1944.545711] md: md124 stopped.
...
```

Workaround

Perform a power cycle to reboot the system successfully. The system boots normally on subsequent reboots.

Explanation

Before rebooting, this issue is triggered by a RAID State of `active, degraded, recovering` that can be displayed by running the `sudo mdadm --detail /dev/mdXXX` command. Replace XXX with the RAID array that you configured with the `configure_raid_array.py` command.

Refer to the following sample output:

```
$ sudo mdadm --detail /dev/md125
/dev/md125:
           Version : 1.2
     Creation Time : Wed Aug 30 11:39:08 2023
        Raid Level : raid5
        Array Size : 26254240768 (24.45 TiB 26.88 TB)
     Used Dev Size : 3750605824 (3.49 TiB 3.84 TB)
      Raid Devices : 8
     Total Devices : 8
       Persistence : Superblock is persistent

     Intent Bitmap : Internal

       Update Time : Wed Aug 30 11:55:51 2023
             State : active, degraded, recovering
    Active Devices : 7
   Working Devices : 8
    Failed Devices : 0
     Spare Devices : 1

            Layout : left-symmetric
        Chunk Size : 512K

Consistency Policy : bitmap

    Rebuild Status : 5% complete

              Name : nv-data-array
              UUID : 2dbe34c6:70decf1e:c54206a6:e78b9161
            Events : 204

    Number   Major   Minor   RaidDevice State
       0     259        1        0      active sync   /dev/nvme2n1
       1     259        3        1      active sync   /dev/nvme3n1
       2     259        6        2      active sync   /dev/nvme4n1
       3     259        7        3      active sync   /dev/nvme5n1
       4     259        9        4      active sync   /dev/nvme6n1
       5     259       13        5      active sync   /dev/nvme7n1
       6     259       14        6      active sync   /dev/nvme8n1
       8     259       15        7      spare rebuilding   /dev/nvme9n1
```

14.7. MOFED mlnxofedinstall reports "Current operation system is not supported" using RHEL 9.2

Platform

EL9-23.01 and RHEL 9.2 with MLNX_OFED_LINUX-5.8-2.0.3.0-rhel9.1-x86_64.iso

Issue

When installing MLNX MOFED driver from the downloaded ISO using `mlnxofedinstall --add-kernel-support` the system generates a warning: "Current operation system is not supported!"

Workaround

Specify the last supported version of RHEL on the commandline by adding "–distro rhel9.1" `` ` `` `mlnx-ofedinstall --distro rhel9.1 --add-kernel-support`

Explanation

The current MLNX MOFED installer script can require the most recent supported OS to be specified by name if the OS is upgraded before the installer support is added for that OS version.

14.8. Precompiled GPU Driver 525 package is not available for Rocky 9.1

Platform

Rocky 9.1 with EL9-23.01

Issue

The Pre-compiled GPU Driver might not support the installed Rocky Linux kernel.

Workaround

You can install the GPU driver by using the DKMS subsystem by running the following commands:

```
sudo dnf module reset -y nvidia-driver
sudo dnf install kernel-devel-$(uname -r) kernel-headers-$(uname -r)
sudo dnf module install nvidia-driver:525-dkms
```

14.9. Yellow screen appears during RHEL 9.1 installation

Issue

When installing the RedHat Enterprise Linux 9.1 ISO on a DGX Station V100, the first installation page shows a yellow screen. This can persist through the installation process and when complete.

Workaround

Install RedHat Enterprise Linux 9.0 on the DGX Station V100, then perform the over the air (OTA) update for the latest RHEL9 version and the DGX EL9-23.01 updates.

14.10. DGX A100: VBIOS cannot update due to running service processes

Issue

VBIOS fails to update on Red Hat Enterprise Linux 9 because service(s)/process(es) are holding onto the resource about to be upgraded.

Workaround

The following services (system processes) must be stopped manually for the firmware update to start:

- process nvidia-persistenced
- process nv-hostengine
- process cache_mgr_event
- process cache_mgr_main
- process dcgm_ipc

If *xorg* is holding the resources, try to stop it by running

```
sudo systemctl stop (display manager)
```

where the (display manager) can be acquired by

```
cat /etc/X11/default-display-manager
```

14.11. NVSM Unsupported Drive Error

Issue

When running `nvsm show storage`, the NV-DRIVE-01 alert displays an “Unsupported Drive Configuration” message.

Workaround

The following services (system processes) must be stopped manually for the firmware update to start:

1. Create a config file to disable nvme multipath:

   ```
   sudo sh -c 'echo "options nvme-core multipath=n" > /etc/modprobe.d/nvidia-nvme.
   ↪conf'
   ```

2. Recreate the initramfs.

   ```
   dracut --force /boot/initramfs-$(uname -r).img $(uname -r)
   ```

3. Reboot the system.

   ```
   sudo systemctl reboot
   ```

The message might be displayed when you log in or when you run the `nvsm show alert` and `nvsm show storage` commands and can be safely ignored. This issue will be fixed in a future release.

14.12. Tuned profiles do not take effect in graphical mode

Issue

DGX tuned profiles might not take effect due to a known Red Hat Enterprise Linux 9 issue. This affects systems that use a graphical target mode.

Workaround

This issue can be fixed by running the following commands:

- Mask the power-profiles-daemon service then tuned is able to start during boot.

  ```
  systemctl mask power-profiles-daemon
  ```

- Reboot the system

  ```
  reboot
  ```

Chapter 15. Third-Party License Notices

This NVIDIA product contains third party software that is being made available to you under their respective open source software licenses. Some of those licenses also require specific legal information to be included in the product. This section provides such information.

15.1. Micron msecli

The msecli utility is provided under the following terms:

Micron Technology, Inc. Software License Agreement PLEASE READ THIS LICENSE AGREEMENT ("AGREEMENT") FROM MICRON TECHNOLOGY, INC. ("MTI") CAREFULLY: BY INSTALLING, COPYING OR OTHERWISE USING THIS SOFTWARE AND ANY RELATED PRINTED MATERIALS ("SOFTWARE"), YOU ARE ACCEPTING AND AGREEING TO THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE WITH THE TERMS OF THIS AGREEMENT, DO NOT INSTALL THE SOFTWARE. LICENSE: MTI hereby grants to you the following rights: You may use and make one (1) backup copy the Software subject to the terms of this Agreement. You must maintain all copyright notices on all copies of the Software. You agree not to modify, adapt, decompile, reverse engineer, disassemble, or otherwise translate the Software. MTI may make changes to the Software at any time without notice to you. In addition MTI is under no obligation whatsoever to update, maintain, or provide new versions or other support for the Software. OWNERSHIP OF MATERIALS: You acknowledge and agree that the Software is proprietary property of MTI (and/or its licensors) and is protected by United States copyright law and international treaty provisions. Except as expressly provided herein, MTI does not grant any express or implied right to you under any patents, copyrights, trademarks, or trade secret information. You further acknowledge and agree that all right, title, and interest in and to the Software, including associated proprietary rights, are and shall remain with MTI (and/or its licensors). This Agreement does not convey to you an interest in or to the Software, but only a limited right to use and copy the Software in accordance with the terms of this Agreement. The Software is licensed to you and not sold.

DISCLAIMER OF WARRANTY:

THE SOFTWARE IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND. MTI EXPRESSLY DISCLAIMS ALL WARRANTIES EXPRESS ORmIMPLIED, INCLUDING BUT NOT LIMITED TO, NONINFRINGEMENT OF THIRD PARTY RIGHTS, AND ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE. MTI DOES NOT WARRANT THAT THE SOFTWARE WILL MEET YOUR REQUIREMENTS, OR THAT THE OPERATION OF THE SOFTWARE WILL BE UNINTERRUPTED OR ERROR-FREE. FURTHERMORE, MTI DOES NOT MAKE ANY REPRESENTATIONS REGARDING THE USE OR THE RESULTS OF THE USE OF THE SOFTWARE IN TERMS OF ITS CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE. THE ENTIRE RISK ARISING OUT OF USE OR PERFORMANCE OF THE SOFTWARE REMAINS WITH YOU. IN NO EVENT SHALL MTI, ITS AFFILIATED COMPANIES OR THEIR SUPPLIERS BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, INCIDENTAL, OR SPECIAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION, OR LOSS OF

INFORMATION) ARISING OUT OF YOUR USE OF OR INABILITY TO USE THE SOFTWARE, EVEN IF MTI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Because some jurisdictions prohibit the exclusion or limitation of liability for consequential or incidental damages, the above limitation may not apply to you.

TERMINATION OF THIS LICENSE: MTI may terminate this license at any time if you are in breach of any of the terms of this Agreement. Upon termination, you will immediately destroy all copies the Software.

GENERAL: This Agreement constitutes the entire agreement between MTI and you regarding the subject matter hereof and supersedes all previous oral or written communications between the parties. This Agreement shall be governed by the laws of the State of Idaho without regard to its conflict of laws rules.

CONTACT: If you have any questions about the terms of this Agreement, please contact MTI's legal department at (208) 368-4500. By proceeding with the installation of the Software, you agree to the terms of this Agreement. You must agree to the terms in order to install and use the Software.

15.2. Mellanox (OFED)

MLNX_OFED <http://www.mellanox.com/> is provided under the following terms:

Copyright (c) 2006 Mellanox Technologies. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Chapter 16. Notices

16.1. Notice

This document is provided for information purposes only and shall not be regarded as a warranty of a certain functionality, condition, or quality of a product. NVIDIA Corporation ("NVIDIA") makes no representations or warranties, expressed or implied, as to the accuracy or completeness of the information contained in this document and assumes no responsibility for any errors contained herein. NVIDIA shall have no liability for the consequences or use of such information or for any infringement of patents or other rights of third parties that may result from its use. This document is not a commitment to develop, release, or deliver any Material (defined below), code, or functionality.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and any other changes to this document, at any time without notice.

Customer should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

NVIDIA products are sold subject to the NVIDIA standard terms and conditions of sale supplied at the time of order acknowledgement, unless otherwise agreed in an individual sales agreement signed by authorized representatives of NVIDIA and customer ("Terms of Sale"). NVIDIA hereby expressly objects to applying any customer general terms and conditions with regards to the purchase of the NVIDIA product referenced in this document. No contractual obligations are formed either directly or indirectly by this document.

NVIDIA products are not designed, authorized, or warranted to be suitable for use in medical, military, aircraft, space, or life support equipment, nor in applications where failure or malfunction of the NVIDIA product can reasonably be expected to result in personal injury, death, or property or environmental damage. NVIDIA accepts no liability for inclusion and/or use of NVIDIA products in such equipment or applications and therefore such inclusion and/or use is at customer's own risk.

NVIDIA makes no representation or warranty that products based on this document will be suitable for any specified use. Testing of all parameters of each product is not necessarily performed by NVIDIA. It is customer's sole responsibility to evaluate and determine the applicability of any information contained in this document, ensure the product is suitable and fit for the application planned by customer, and perform the necessary testing for the application in order to avoid a default of the application or the product. Weaknesses in customer's product designs may affect the quality and reliability of the NVIDIA product and may result in additional or different conditions and/or requirements beyond those contained in this document. NVIDIA accepts no liability related to any default, damage, costs, or problem which may be based on or attributable to: (i) the use of the NVIDIA product in any manner that is contrary to this document or (ii) customer product designs.

No license, either expressed or implied, is granted under any NVIDIA patent right, copyright, or other NVIDIA intellectual property right under this document. Information published by NVIDIA regarding third-party products or services does not constitute a license from NVIDIA to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property rights of the third party, or a license from NVIDIA under the patents or other intellectual property rights of NVIDIA.

Reproduction of information in this document is permissible only if approved in advance by NVIDIA in writing, reproduced without alteration and in full compliance with all applicable export laws and regulations, and accompanied by all associated conditions, limitations, and notices.

THIS DOCUMENT AND ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, “MATERIALS”) ARE BEING PROVIDED “AS IS.” NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL NVIDIA BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF NVIDIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Notwithstanding any damages that customer might incur for any reason whatsoever, NVIDIA’s aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms of Sale for the product.

16.2. Trademarks

NVIDIA, the NVIDIA logo, DGX, DGX-1, DGX-2, DGX A100, DGX A800, DGX H100, DGX H200, DGX Station, and DGX Station A100 are trademarks and/or registered trademarks of NVIDIA Corporation in the Unites States and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

Chapter 17. Introduction

This document explains the installation and configuration of the NVIDIA DGX Software Stack on DGX systems installed with Red Hat Enterprise Linux.

NVIDIA provides the NVIDIA DGX Software Stack targeted for installation on DGX systems that have been user-installed with Red Hat Enterprise Linux. The software stack provides the same features and functionality that are provided by the original DGX OS built on the Ubuntu.

> **Note**
>
> NVIDIA acknowledges the wide use of Rocky Linux and understands that it is a community-developed derivative of the NVIDIA supported Red Hat Enterprise Linux. Support for Rocky Linux is available directly from the Rocky Linux community. NVIDIA ensures that NVIDIA provided software runs on tested Rocky Linux versions and will try to identify and correct issues related to NVIDIA provided software.
>
> While it might be possible to use other derived Linux distributions, not all have been tested and qualified by NVIDIA. Refer to the *Release Notes* for the list of tested and qualified software and Linux distributions.

17.1. Related Documentation

- NVIDIA DGX-1 User Guide
- NVIDIA DGX-2 User Guide
- NVIDIA DGX Station User Guide
- NVIDIA DGX A100 User Guide
- NVIDIA DGX Station A100 User Guide
- NVIDIA DGX H100/H200 User Guide

17.2. Before You Begin

The following are required (or recommended wherever indicated).

Chapter 18. Red Hat Subscription

You need a Red Hat subscription if you plan to install and use Red Hat Enterprise Linux on the DGX system. A subscription also lets you obtain update packages and additional packages for Red Hat Enterprise Linux. You can either purchase a subscription or obtain a free evaluation subscription from the Red Hat Software & Download Center.

Chapter 19. Access to Repositories

The repositories can be accessed from the internet.

If you are using a proxy server, follow the instructions in the section *Configuring a System Proxy* to make sure the system can access the necessary URIs.

- **NVIDIA Repositories**

  After installing Red Hat Enterprise Linux on the DGX system, you must enable the NVIDIA DGX software repository (https://repo.download.nvidia.com). The repository includes the NVIDIA drivers and software for supporting DGX systems.

  See the section *Enabling the DGX Software Repository* for instructions on how to enable the repository.

- **Red Hat Repositories**

  Installation of the DGX Software over Red Hat Enterprise Linux 9 requires access to several additional repositories.

  - Red Hat Enterprise BaseOS Repository: `rhel-9-for-x86_64-baseos-rpms`
  - Red Hat Enterprise AppStream Repository: `rhel-9-for-x86_64-appstream-rpms`
  - Red Hat Enterprise CodeReady Linux Builder Repository: `codeready-builder-for-rhel-9-x86_64-debug-rpms`

Chapter 20. Network File System

On DGX servers, the data drives are meant to be used as a cache. DGX Station users can follow the same usage, or can alternatively opt to use these drives for storage. When using the data drives as cache, a network file system (NFS) is recommended to take advantage of the cache file system provided by the DGX software stack.

Chapter 21. BMC Password

The NVIDIA DGX server includes a base management controller (BMC) for out-of-band management of the DGX system. NVIDIA recommends disabling the default username and creating a unique username and password as soon as possible.

Copyright