DGX Software with Red Hat Enterprise Linux 7

Installation Guide

RN-09301-001 _v08 | May 2021

Table of Contents

Chapter 1. Introduction 1
1.1. Related Documentation 1
1.2. Prerequisites 1
1.2.1. Red Hat Subscription 1
1.2.2. Access to Repositories 2
1.2.2.1. NVIDIA Repositories 2
1.2.2.2. Red Hat Repositories 2
1.2.3. Network File System 3
1.2.4. BMC Password 3
Chapter 2. Installing Red Hat Enterprise Linux 4
2.1. Obtaining Red Hat Enterprise Linux 4
2.2. Booting Red Hat Enterprise Linux ISO Locally 4
2.3. Booting the Red Hat Enterprise Linux ISO Remotely on the DGX-1, DGX-2, or DGX A100 5
2.3.1. Booting the ISO Image on the DGX-1 Remotely 5
2.3.2. Booting the ISO Image on the DGX-2 Remotely 8
2.3.3. Booting the ISO Image on the DGX A100 Remotely 12
2.4. Installing Red Hat Enterprise Linux 15
2.4.1. Installing on the DGX-1, DGX Station, or DGX Station A100 16
2.4.2. Installing on the DGX-2 27
2.4.3. Installing on the DGX A100 36
Chapter 3. Installing the DGX Software 46
3.1. Configuring a System Proxy 46
3.2. Enabling the Repositories 46
3.3. Installing Required Components 47
3.3.1. Installing DGX Tools and Updating Configuration Files 47
3.3.2. Configuring the /raid Partition 48
3.3.2.1. Configuring the /raid Partition as an NFS Cache 48
3.3.2.2. Configuring the /raid Partition for Local Persistent Storage 49
3.3.3. Installing and Loading the NVIDIA CUDA Drivers 49
3.3.4. Installing the NVIDIA Container Runtime 50
3.4. Installing Diagnostic Components 51
3.5. Replicating the EFI System Partition on DGX-2 or DGX A100 51
3.6. Installing Optional Components 52
3.7. Applying an NVIDIA Look and Feel to the Desktop User Interface 53
3.8. Managing CPU Mitigations 56

3.8.1. Determining the CPU Mitigation State of the DGX System..... 56
3.8.2. Disabling CPU Mitigations..... 57
3.8.3. Re-enabling CPU Mitigations..... 57
Chapter 4. Running Containers..... 59
Chapter 5. Configuring Storage - NFS Mount and Cache..... 60
Appendix A. Installing Software on Air-Gapped NVIDIA DGX Systems..... 62
A.1. Registering Your System..... 62
A.2. Creating the Mirrors on the Low-Side Red Hat System..... 62
A.3. Installing Red Hat Enterprise Linux on the Air-Gapped DGX-2/DGX A100..... 64
A.4. Installing DGX Software on the Air-Gapped DGX-2/DGX A100..... 67
A.5. Renaming RAID Volumes..... 70
A.6. Installing Docker Containers..... 70
Appendix B. Installing Software Using Local Repositories..... 71
B.1. Installing the Base Red Hat Enteprise Linux System..... 71
B.2. Downloading the Local Repositories..... 71
B.3. Installing the Local Repositories..... 71
Appendix C. Changing the BMC Login..... 72
C.1. Changing the BMC Login on the DGX-1..... 72
C.2. Changing the BMC Login on the DGX-2 or DGX A100..... 77
Appendix D. Installing and Mellanox InfiniBand Drivers..... 79
Appendix E. Using Custom DGX Software Utilities for the DGX Station..... 81
E.1. Rebuilding or Re-Creating the DGX Station RAID Array..... 81
E.2. Changing the RAID Level of the RAID Array..... 82
E.3. EL7-20.01 Only: Checking the Health of the DGX Station..... 83
E.4. EL7-20.01 Only: Collecting Information for Troubleshooting the DGX Station..... 84
Appendix F. Expanding the DGX Station RAID Array..... 86

Chapter 1. Introduction

The NVIDIA® DGX™ systems (DGX-1, DGX-2, and DGX A100 servers, and NVIDIA DGX Station™ and DGX Station A100 systems) are shipped with DGX™ OS which incorporates the NVIDIA DGX software stack built upon the Ubuntu Linux distribution. Instead of running the Ubuntu distribution, you can run Red Hat Enterprise Linux on the DGX system and still take advantage of the advanced DGX features.

This document explains how to install and configure the NVIDIA DGX software stack on DGX systems installed with Red Hat Enterprise Linux.

Note: While it may be possible to use other derived Linux distributions besides Red Hat Enterprise Linux, not all have been tested and qualified by NVIDIA. Refer to the DGX Software for Red Hat Enterprise Linux 7 Release Notes for the list of tested and qualified software and Linux distributions.

1.1. Related Documentation

- NVIDIA DGX Software for Red Hat Enterprise Linux - Release Notes
- NVIDIA DGX-1 User Guide
- NVIDIA DGX-2 User Guide
- NVIDIA DGX A100 User Guide
- NVIDIA DGX Station User Guide

1.2. Prerequisites

The following are required (or recommended where indicated).

1.2.1. Red Hat Subscription

You need a Red Hat subscription if you plan to install and use Red Hat Enterprise Linux 7 on the DGX. 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.

Note: Of the available Red Hat Enterprise Linux platforms, **only** Red Hat Enterprise Linux **Server** is supported on DGX systems (DGX servers and DGX Station workstation). Other Red Hat Enterprise Linux platforms are **not** supported on any DGX system.

1.2.2. Access to Repositories

The repositories can be accessed from the internet.

If your installation does not allow connection to the internet, see the section Installing Software on Air-Gapped NVIDIA DGX Systems for information about updating software on "air-gapped" systems.

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

Note:

You can use `yum-config-manager` to conveniently enable certain repositories. To use `yum-config-manager`, first install the yum utilities.

```
sudo yum -y install yum-utils
```

1.2.2.1. NVIDIA Repositories

- **NVIDIA DGX Software Repository**

  After installing Red Hat Enterprise Linux on the DGX system, you must enable the NVIDIA DGX software repository. The repository includes the NVIDIA drivers and software for supporting DGX systems.

  See the section Enabling the Repositories for instructions on how to enable the repositories.

1.2.2.2. Red Hat Repositories

Installation of the DGX Software over Red Hat Enterprise Linux 7 requires access to several additional repositories.

- **Red Hat Enterprise Server Extras Repository** (requred for container support):

  DGX Servers: `rhel-7-server-extras-rpms`

  DGX Station: `rhel-7-workstation-extras-rpms`

- **Red Hat Enterprise Server Optional Repository** (required by NVIDIA System Manager (NVSM) and the GPU driver):

  DGX Servers: `rhel-7-server-optional-rpms`

  DGX Station: `rhel-7-workstation-optional-rpms`

- **Red Hat Software Collections Repository**:

This repository is required by the NVSM tool for Python 3. If you do not have access to the Red Hat software collections repository, refer to https://access.redhat.com/solutions/472793 for instructions on requesting access for free.

Important: NVSM is not supported with the `python3` package. Be sure to only install the `rh-python36` package per the instructions in Installing Diagnostic Components.

DGX Servers: `rhel-server-rhscl-7-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 DGX BMC comes with default login credentials as specified in Appendix B: Changing the BMC Login.

Important:

NVIDIA recommends disabling the default username and creating a unique BMC username and strong password as soon as possible. Refer to Appendix B: Changing the BMC Login for instructions.

Chapter 2. Installing Red Hat Enterprise Linux

There are several methods for installing Red Hat Enterprise Linux as described in the Red Hat Enterprise Linux Installation Guide (https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/installation_guide/index).

See the DGX Software for Red Hat Enterprise Linux Release Notes for the Linux distributions that are qualified and tested for use with the DGX Software.

For convenience, this section 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.

2.1. Obtaining Red Hat Enterprise Linux

Obtain the Red Hat Enterprise Linux ISO image and store on your local disk or create a boot USB drive formatted for UEFI. See Downloading Red Hat Enterprise Linux (https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/installation_guide/chap-download-red-hat-enterprise-linux) for instructions.

2.2. Booting Red Hat Enterprise Linux ISO Locally

1. Plug the USB flash drive containing the Red Hat Enterprise Linux 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 UEFI volume name that corresponds to the inserted USB flash drive, and boot the system from it.

5. Follow the instructions at Installing Red Hat Enterprise Linux

2.3. Booting the Red Hat Enterprise Linux ISO Remotely on the DGX-1, DGX-2, or DGX A100

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.

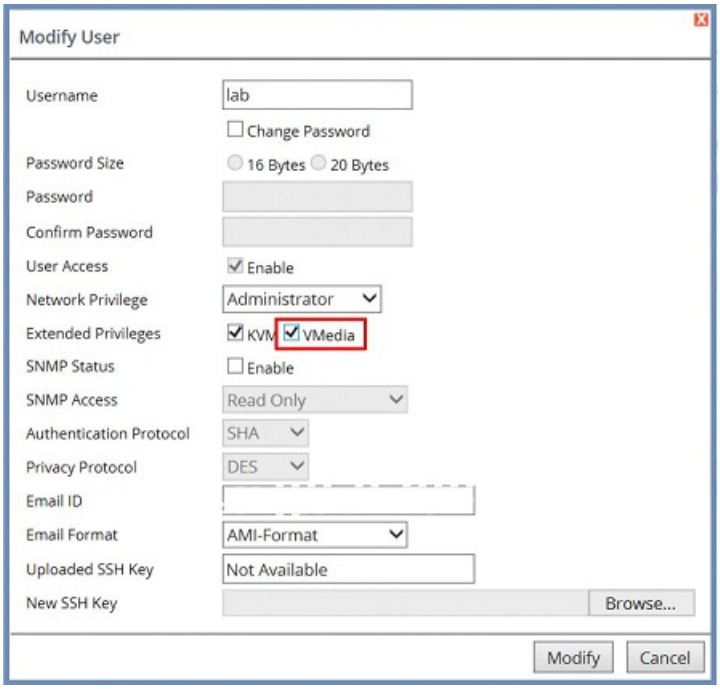
2.3.1. Booting the ISO Image on the DGX-1 Remotely

Skip this chapter if you are using a monitor and keyboard for installing locally.

- For instructions applicable to the NVIDIA DGX-2, see Booting the ISO Image on the DGX-2 Remotely
- For instructions applicable to the NVIDIA DGX A100, see Booting the ISO Image on the DGX A100 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.

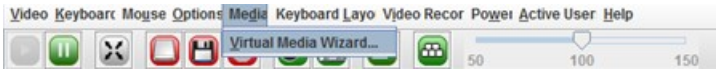
   Use Firefox or Internet Explorer. Google Chrome is not officially supported by the BMC.
   b). From the top menu, click **Configuration** and then select **User Management**.
   c). Select the user name that you created for the BMC, then click **Modify User**.
   d). In the **Modify User** dialog, select the **VMedia** checkbox to add it to the extended privileges for the user, then click **Modify**.



2. Set up the ISO image as virtual media and reboot the system.
   a). From the top menu, click **Remote Control** and select **Console Redirection**.

   b). Click **Java Console** to open the remote JViewer window. Make sure pop-up blockers are disabled for this site.

   c). From the JViewer top menu bar, click **Media** and then select **Virtual Media Wizard**.

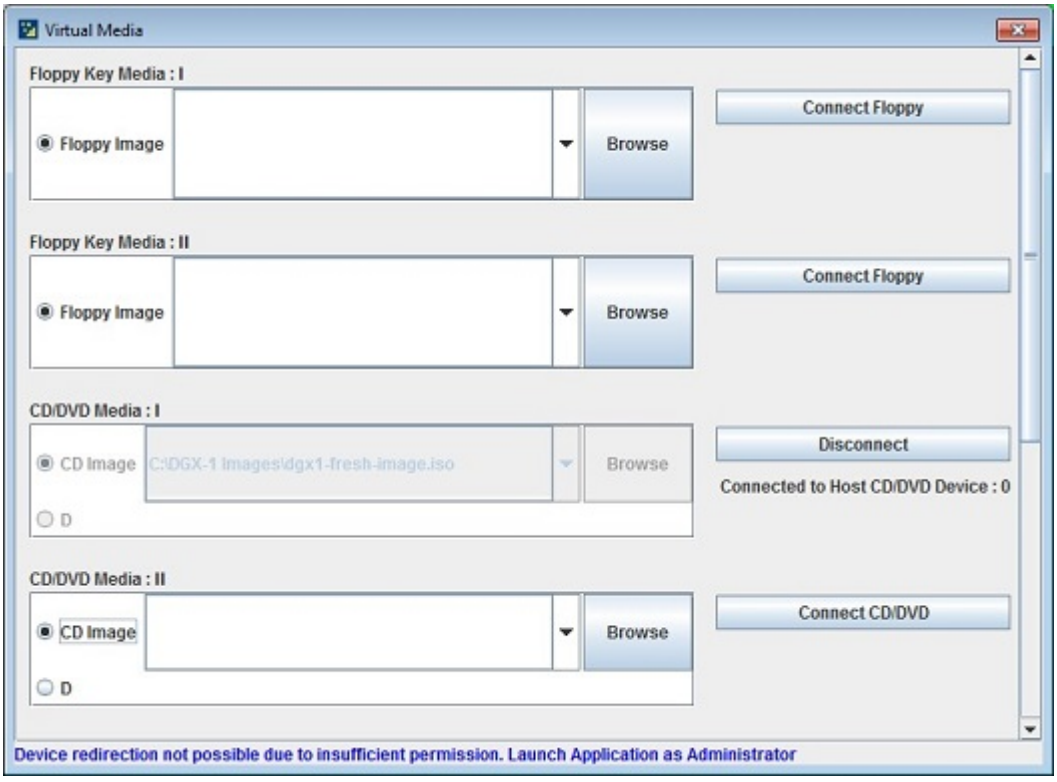


   d). From the CD/DVD Media: I section of the Virtual Media dialog, click **Browse** and then locate the Red Hat Enterprise Linux ISO file on your system and click **Open**.

   You can ignore the device redirection warning at the bottom of the Virtual Media wizard as it does not affect the ability to re-image the system.

   e). Click **Connect CD/DVD**, then click **OK** at the Information dialog.

   The Virtual Media window shows that the ISO image is connected.



f). Close the window.

The CD ROM icon in the menu bar turns green to indicate that the ISO image is attached.

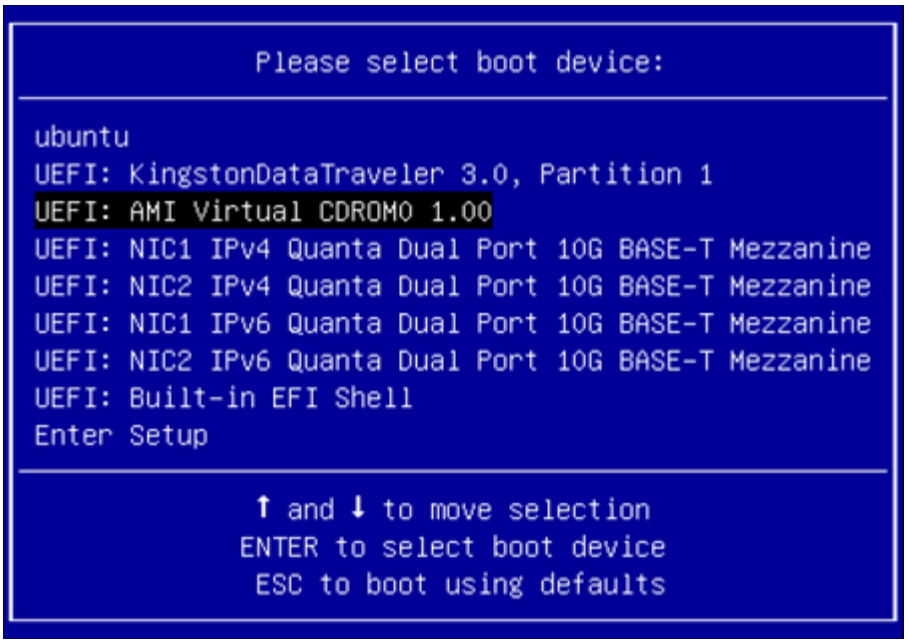
g). From the top menu, click **Power** and then select **Reset Server**.

h). Click **Yes** and then **OK** at the Power Control dialogs, then wait for the system to power down and then come back online.

3. Boot the CD ROM image.

Typically, 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.

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

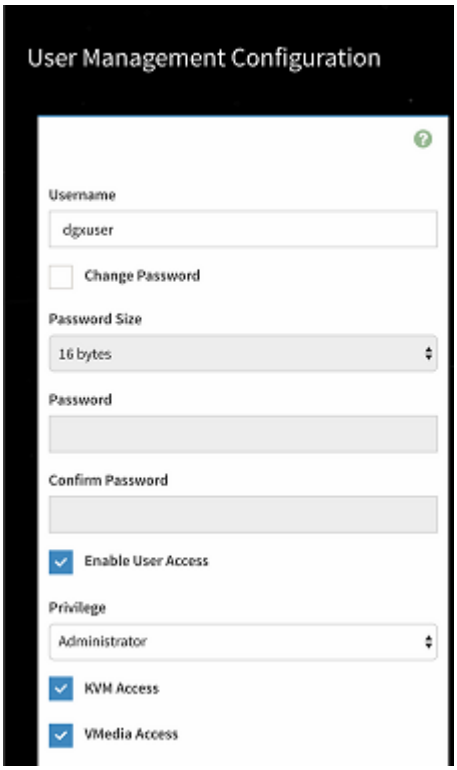


b). Follow the instructions at Installing Red Hat Enterprise Linux.

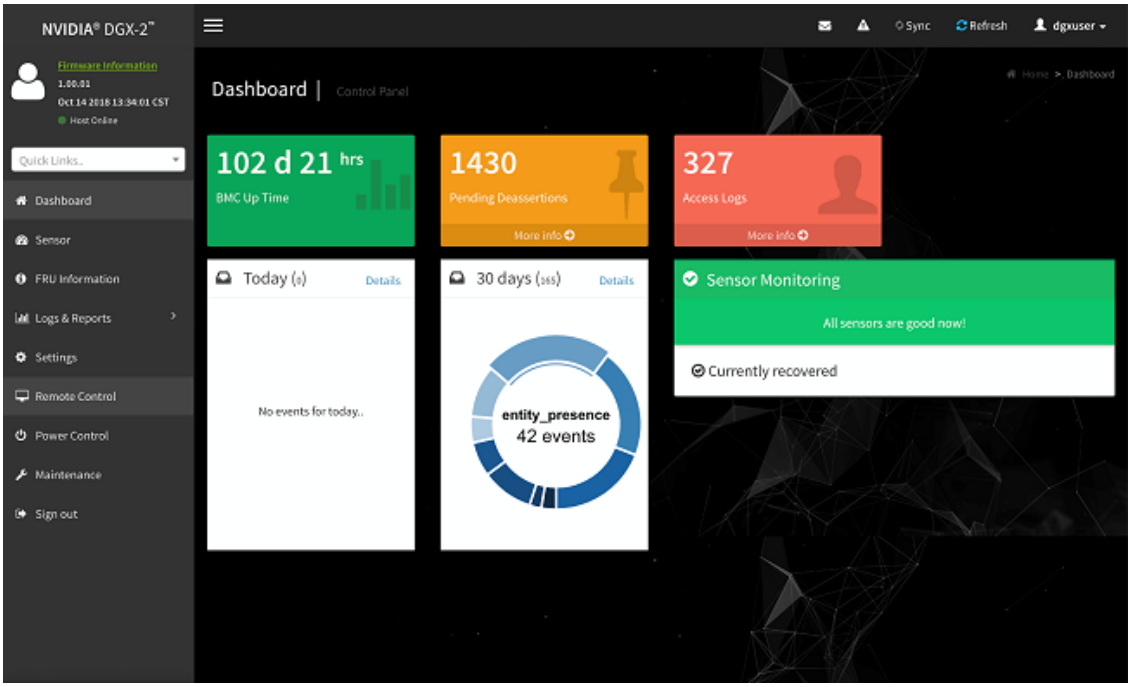
2.3.2. Booting the ISO Image on the DGX-2 Remotely

Skip this chapter if you are using a monitor and keyboard for installing locally.

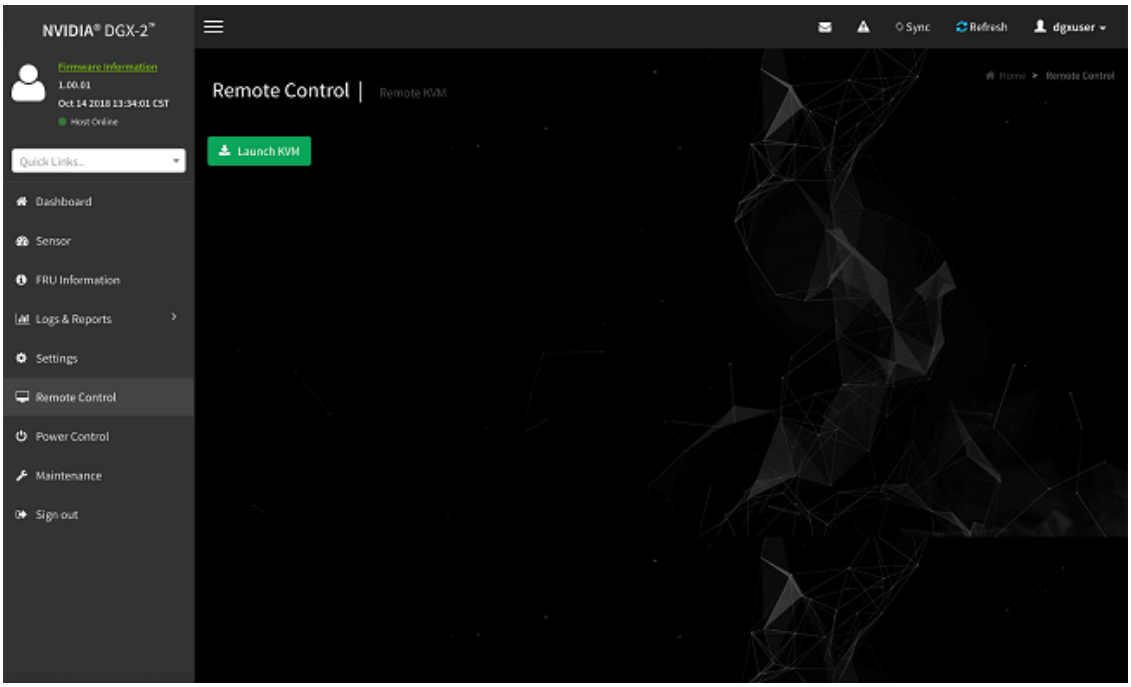
1. Connect to the BMC and ensure the required user privileges are set.
   a). Open a browser within your LAN and go to `https://<BMC-ip-address>/`, then log in.
   b). From the left-side menu, click **Settings** and then select **User Management**.
   c). Click the card with the user name that you created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



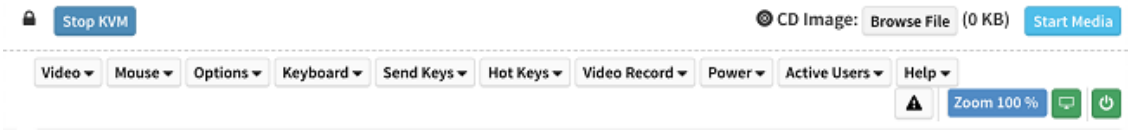
2. Set up the ISO image as virtual media.
   a). From the left-side menu, click **Remote Control**.



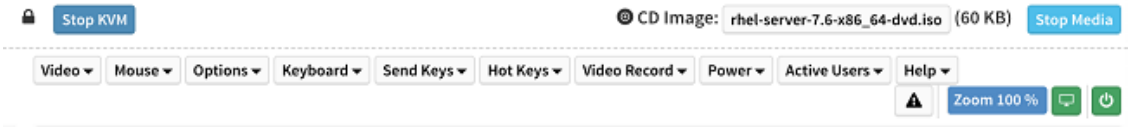
b). Select **Launch KVM**.



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.



d). From the top menu bar in the KVM window, click **Power** and then select **Reset Server**.

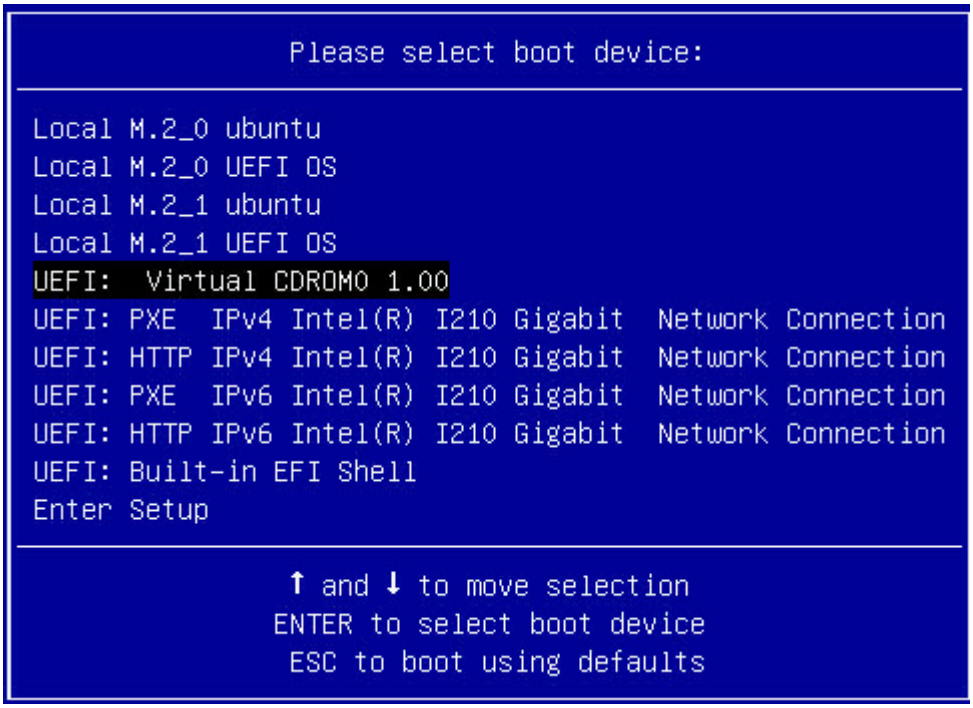
3. Boot from the virtual media.

   Typically, 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.

   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.

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

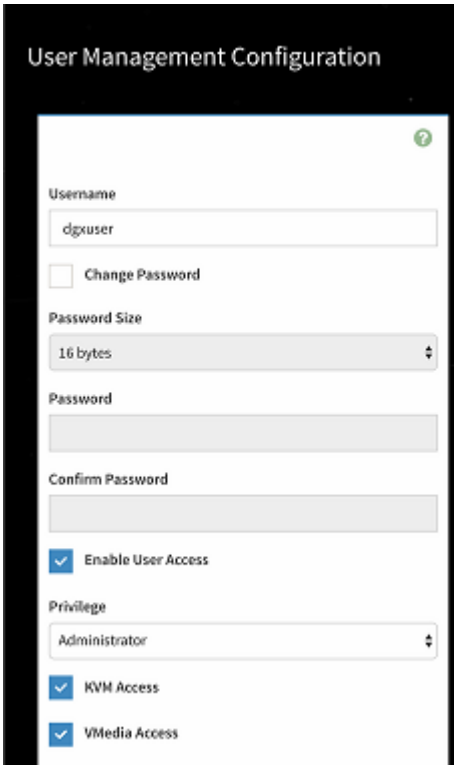


c). Follow the instructions at Installing Red Hat Enterprise Linux

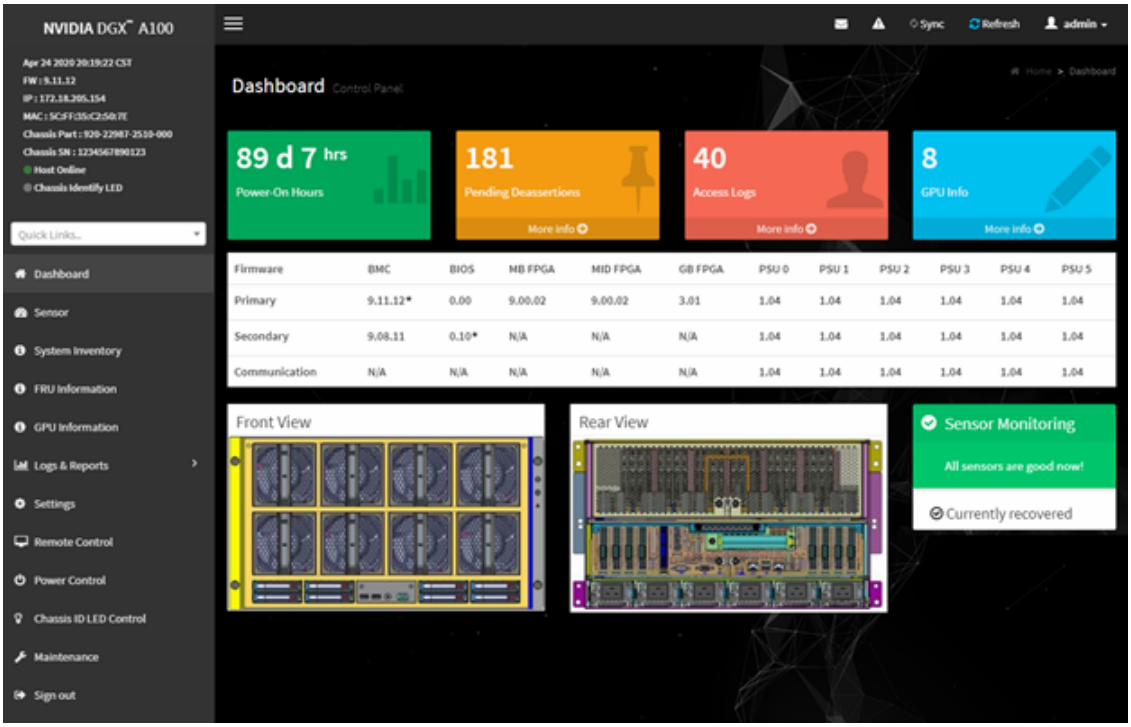
2.3.3. Booting the ISO Image on the DGX A100 Remotely

Skip this chapter if you are using a monitor and keyboard for installing locally.

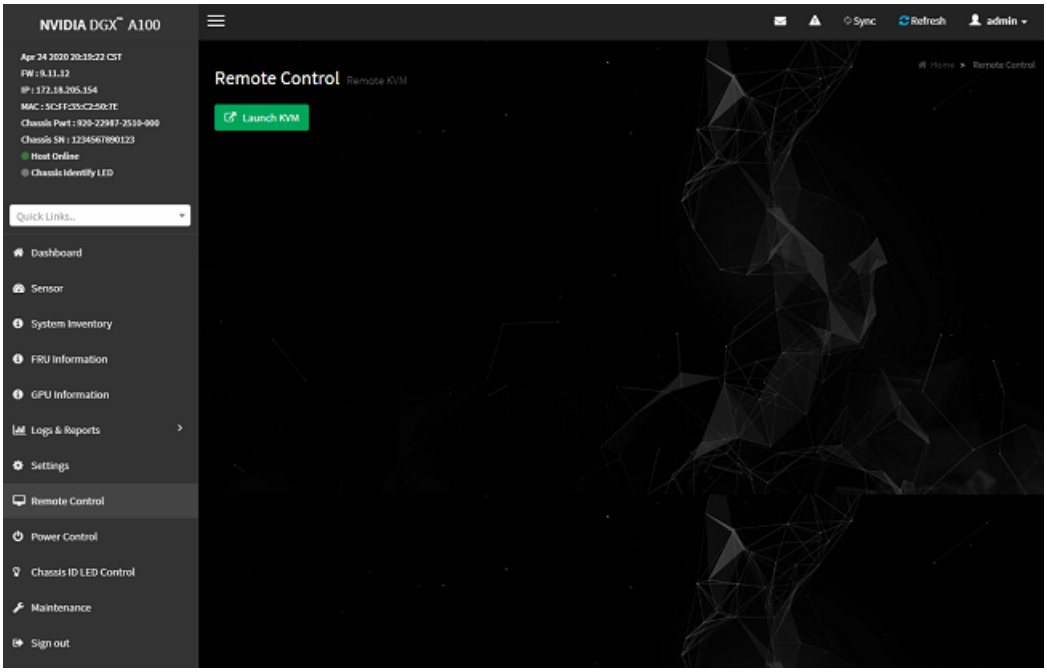
1. Connect to the BMC and ensure the required user privileges are set.
   a). Open a browser within your LAN and go to `https://<BMC-ip-address>/`, then log in.
   b). From the left-side menu, click **Settings** and then select **User Management**.
   c). Click the card with the user name that you created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



2. Set up the ISO image as virtual media.
   a). From the left-side menu, click **Remote Control**.

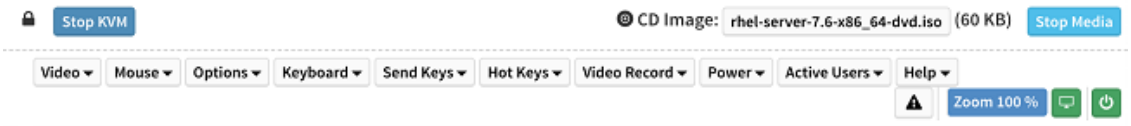


   b). Select **Launch KVM**.



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.



d). From the top menu bar in the KVM window, click **Power** and then select **Reset Server**.

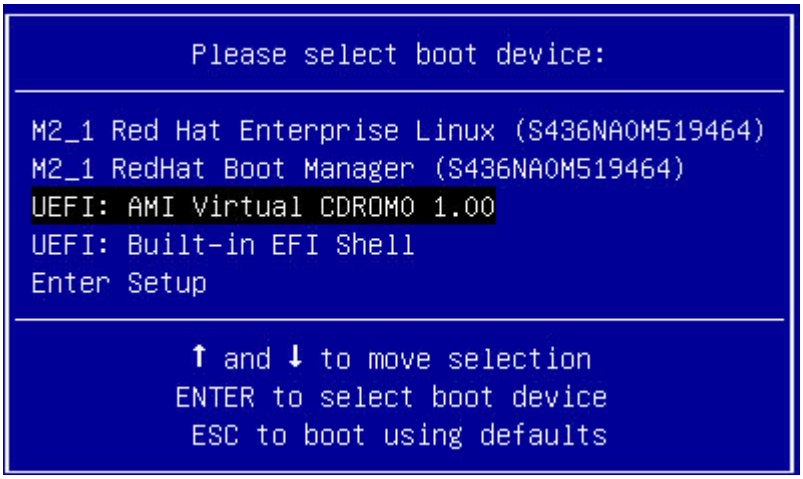
3. Boot from the virtual media.

   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.

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



c). Follow the instructions at Installing Red Hat Enterprise Linux

2.4. Installing Red Hat Enterprise Linux

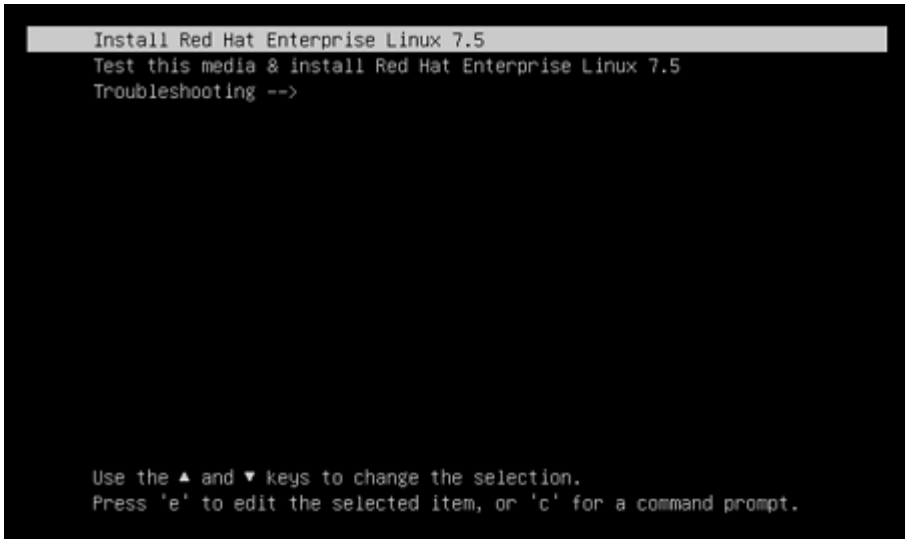
2.4.1. Installing on the DGX-1, DGX Station, or DGX Station A100

This section assumes you have booted the Red Hat Enterprise Linux ISO image, either locally or remotely for DGX-1, or locally for DGX Station. or DGX Station A100

- For instructions applicable to the DGX-2, see Installing on the DGX-2.
- For instructions applicable to the DGX A100, see Installing on the DGX A100.

The screenshots in these instructions are taken from a DGX-1 system. The DGX Station installation can have slight differences with respect to disk size, disk space available, interface names, etc.

1. After booting the ISO image through either the BMC or from the USB drive, start the installation.

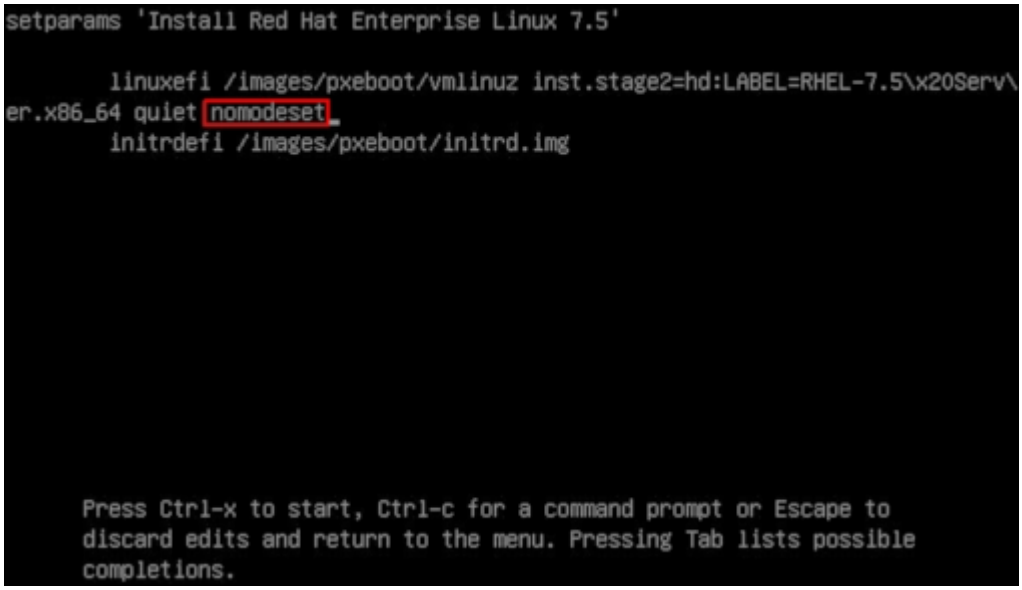


   - **On the DGX-1:** Select **Install Red Hat Enterprise Linux** and then press **Enter** to start the installation.
   - **On the DGX Station:** Update the `linuxefi` parameters to prevent the Nouveau driver in Red Hat Enterprise Linux from being loaded and then start the installation.

     The Nouveau driver does not properly support the GPUs in the DGX Station, which may cause display issues. The GPUs in the DGX Station require that drivers that are installed as explained in Installing and Loading the NVIDIA CUDA Drivers.

     a). Select **Install Red Hat Enterprise Linux** and then type **e**.

     b). Append the `nomodeset` option to the `linuxefi` parameters.



c). Press **Ctrl**+**X** to save your changes and start the installation.

The Red Hat Enterprise Linux installer starts.

For guidance on using the installer, refer to Red Hat Enterprise Linux Quick Installation Guide (https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/installation_guide/chap-simple-install).

2. Configure the language, region, date, time, keyboard, and other configuration options you may need from the **Installation Summary** screen.
3. Select the software packages that you want to install.

   For detailed instructions, refer to Software Selection in Red Hat Enterprise Linux Installation Guide (https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/installation_guide/sect-package-selection-x86).
4. Set up the system drive.

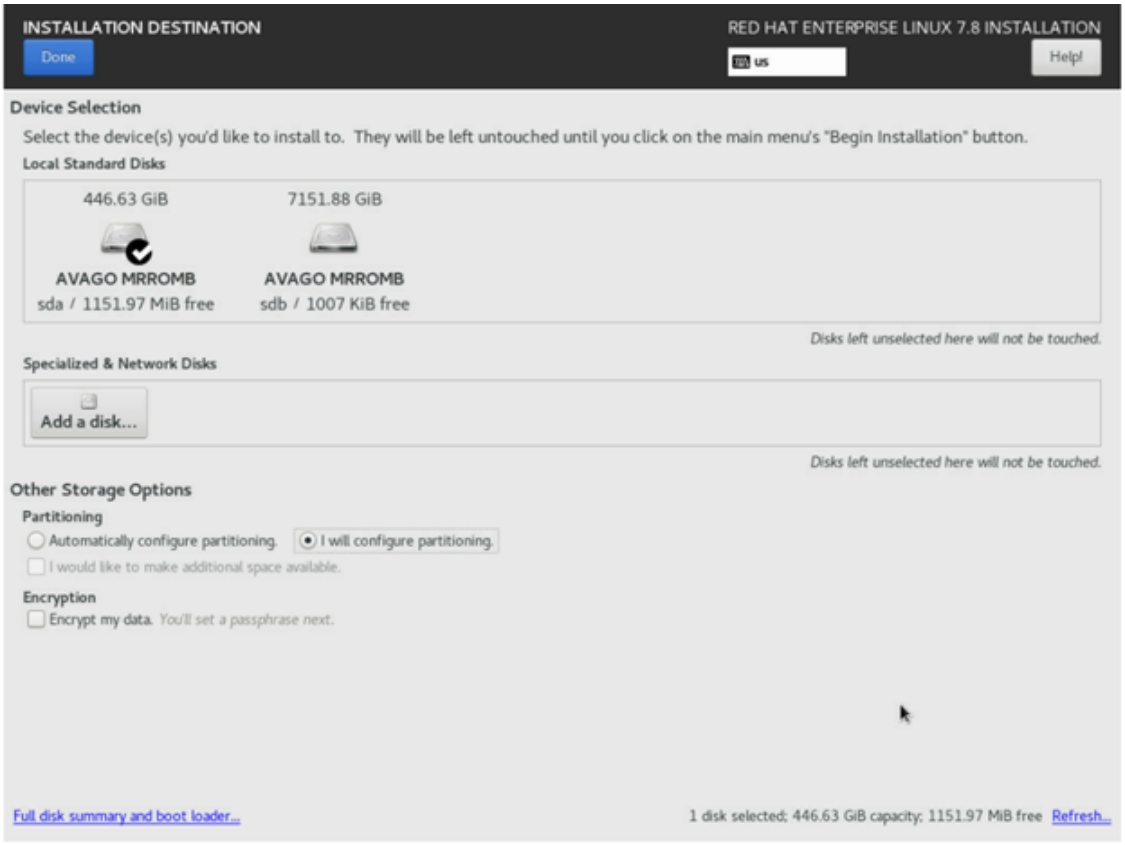
   This step removes the Ubuntu installation in order to reclaim space for the Red Hat Enterprise Linux installation.

   a). From the Installation Summary screen, click **INSTALLATION DESTINATION**.

b). At the **Installation Destination** screen, select the first drive (sda) as the installation drive.

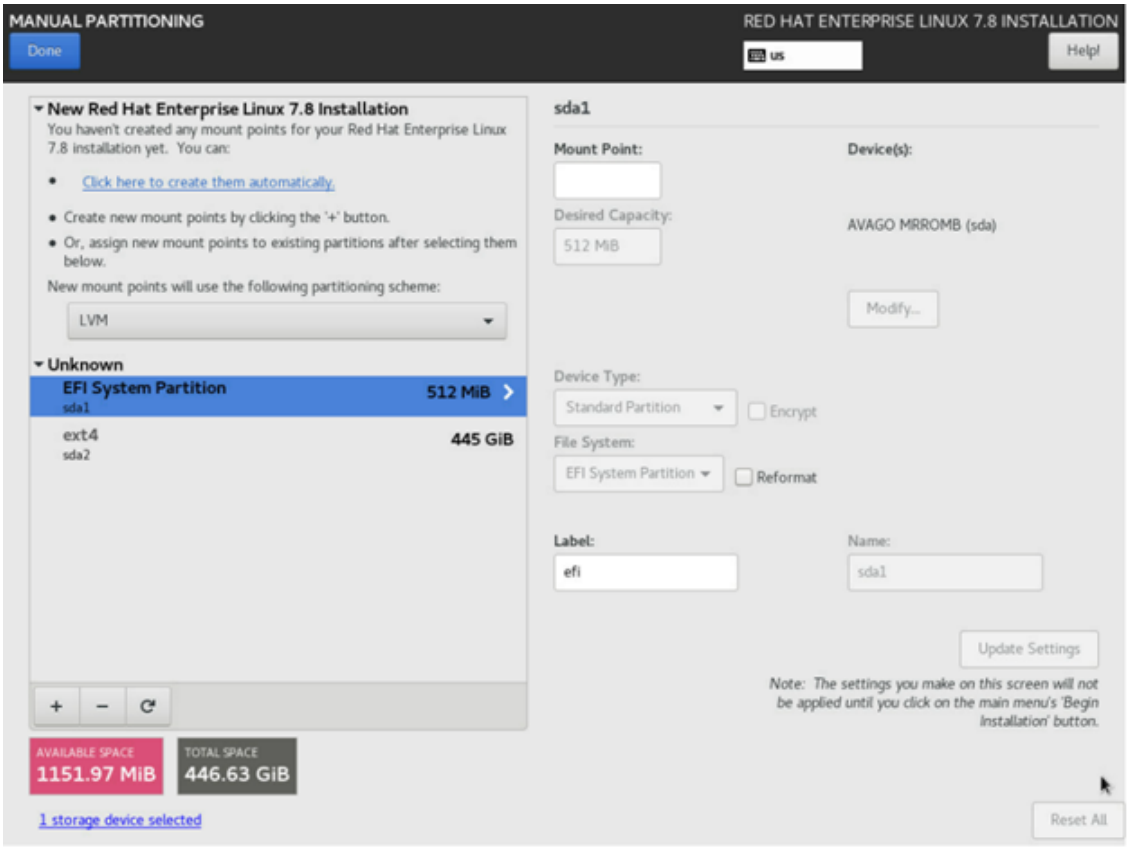
c). Under **Other Storage Options**, click the **I will configure partitioning** radio button then click **Done**.

Note: If the software is being installed on the DGX Station, the number of local standard disks, their labels, capacities, and the amount of free space shown may be different than the examples shown in the screen capture.

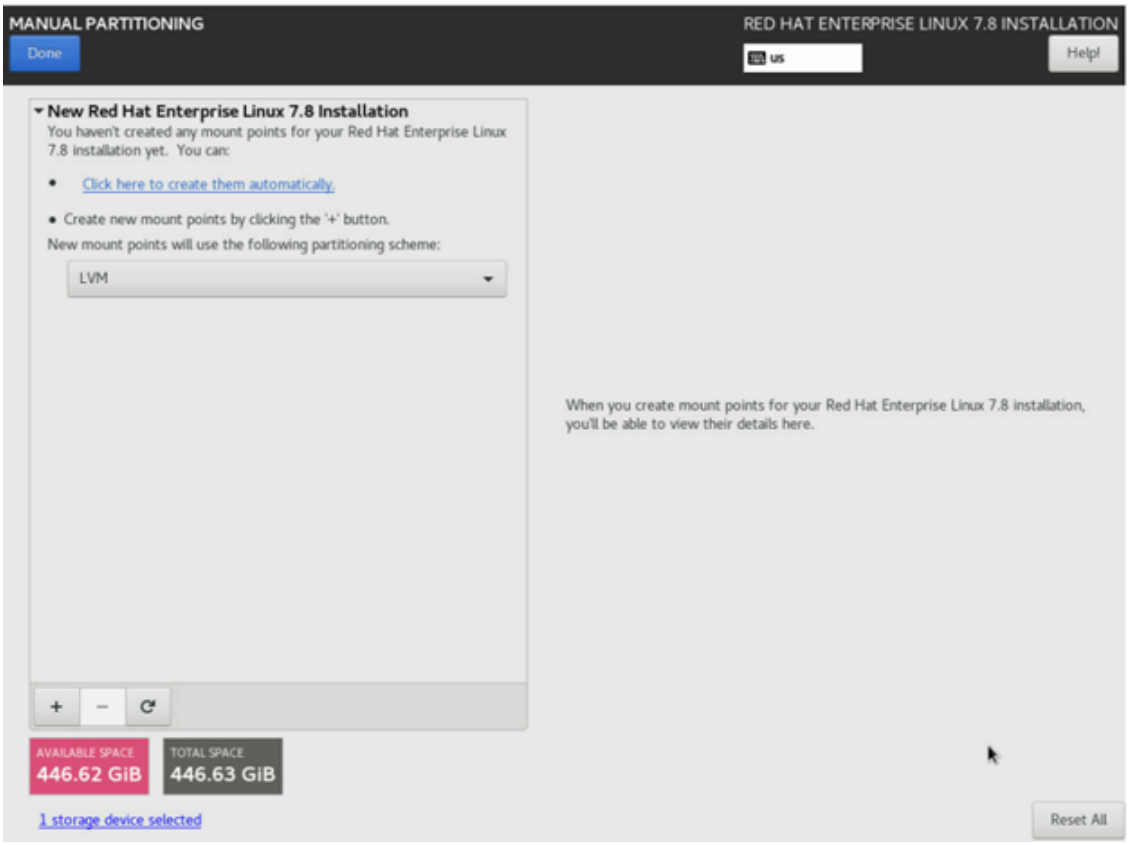


The **Manual Partitioning** dialog box appears.

d). Expand the **Unknown** drop-down menu for the device you've selected, select a partition and then click the "-" button to delete the partition.



Repeat for other existing partitions listed in the **Unknown** drop down until none remain.

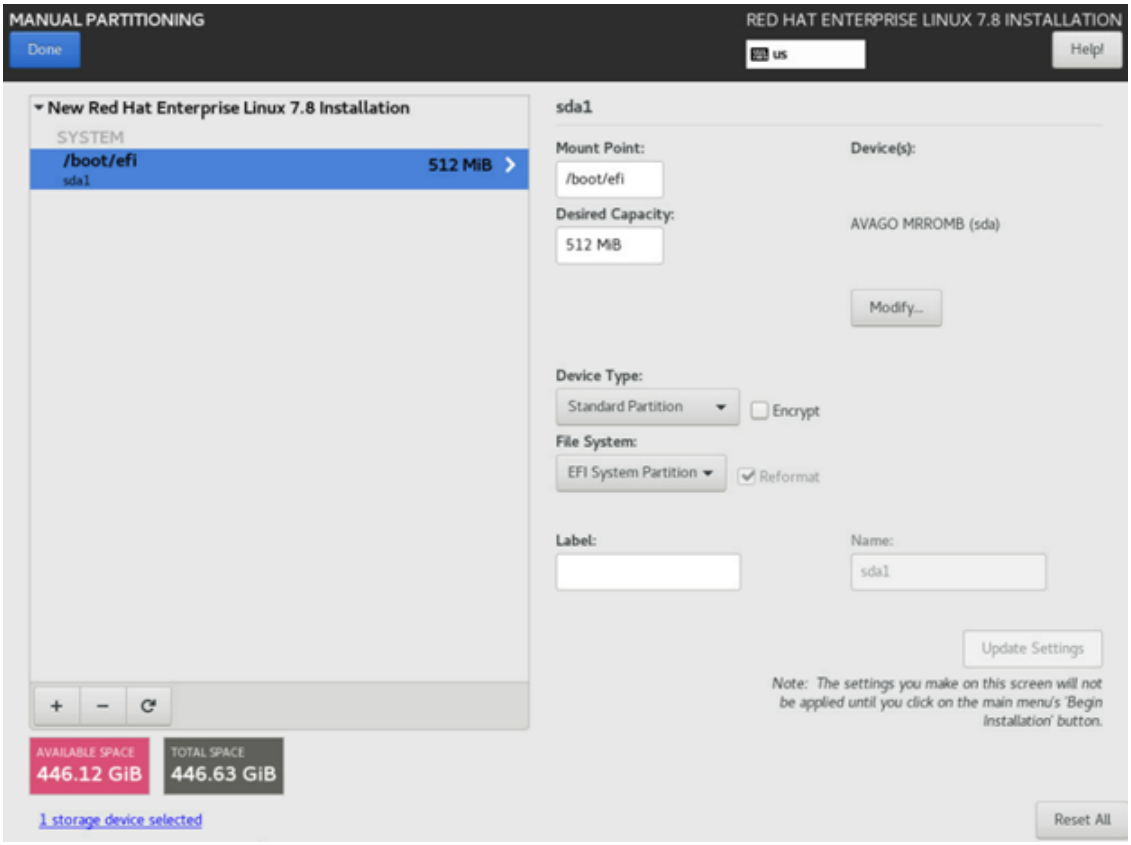


e). Create a new partition.

Click the "+" button on the bottom left and set the following fields:

- Mount point: **/boot/efi**
- Desired Capacity: **512 MiB**
- Device Type: **Standard Partition**

Under **Devices**, ensure only the sda drive you chose is selected, then click **Update Settings** to confirm your changes.

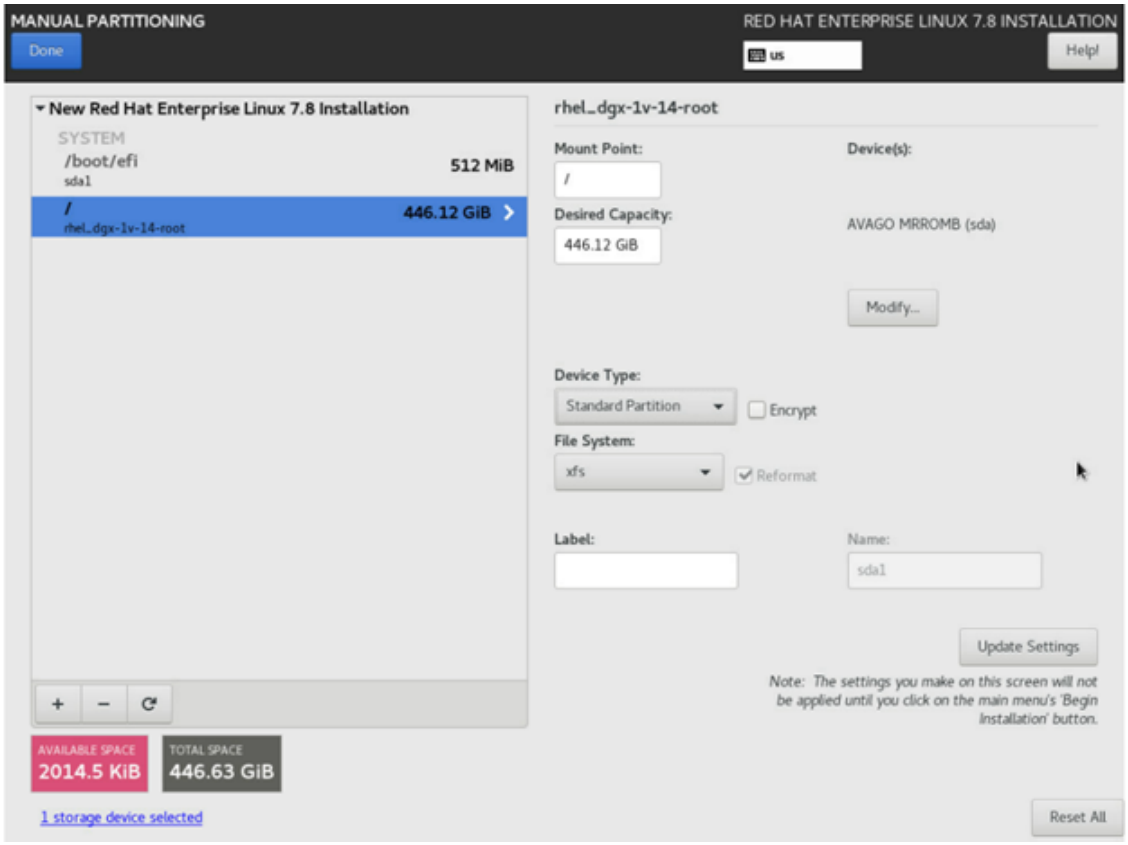


f). Create another partition.

Click the "+" button on the bottom left and set the following fields:

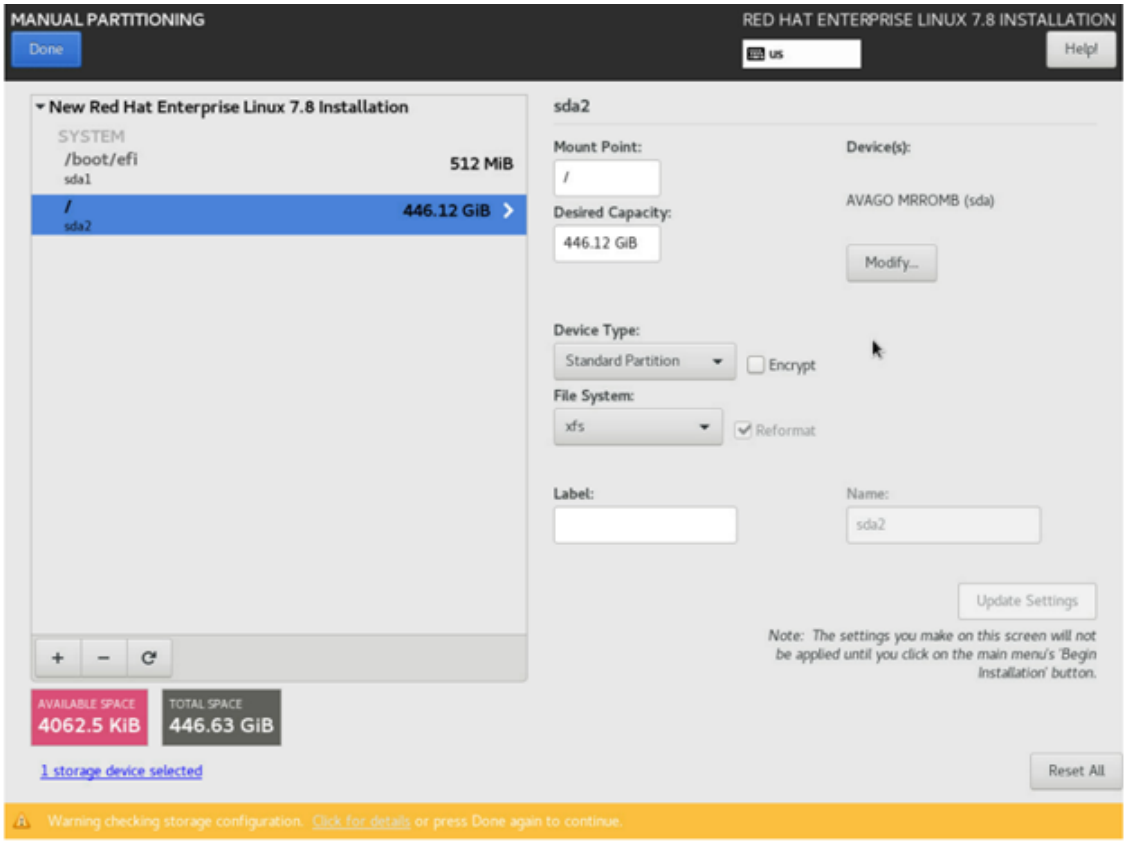
- Mount point: **/**
- Desired Capacity: [leave empty to let the installer know to use remaining capacity of the disks]
- Device Type: **Standard Partition**
- File System: **XFS**

Under **Devices**, ensure only the sda drive you chose is selected, then click **Update Settings** to confirm your changes.

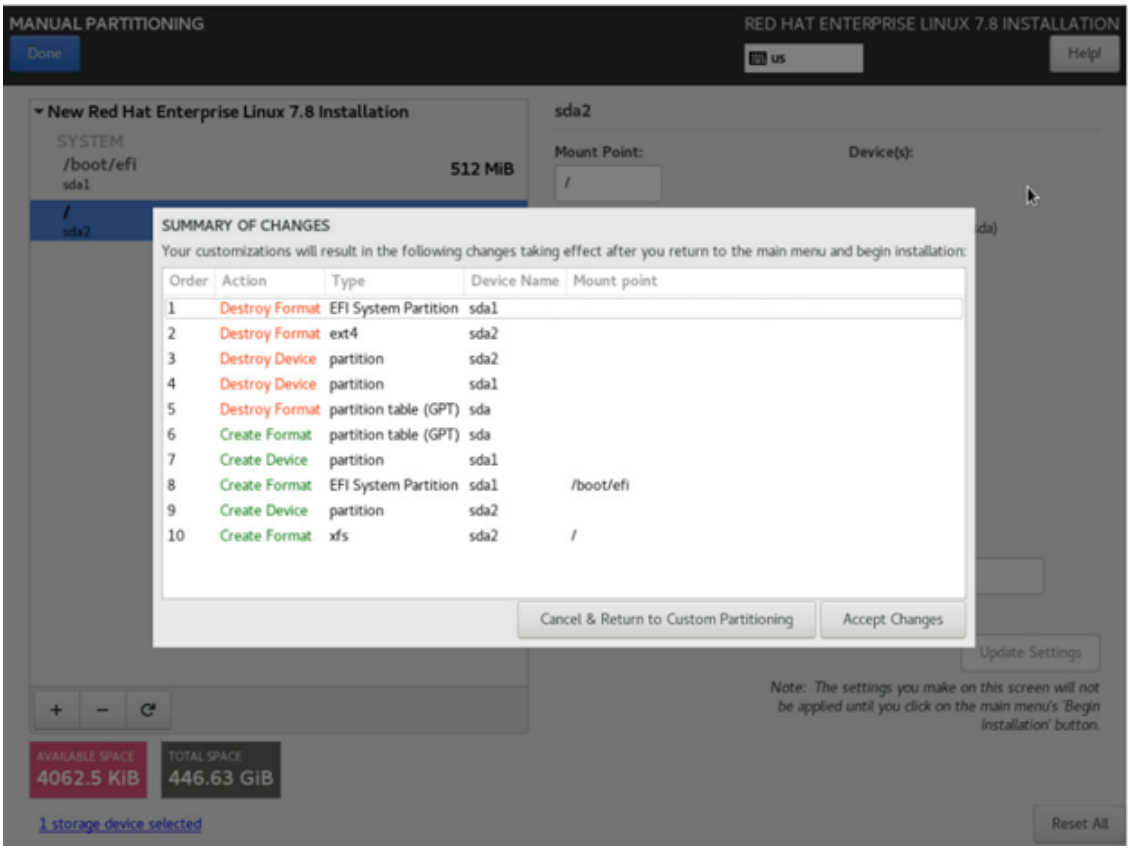


g). Click **Done** when finished.

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

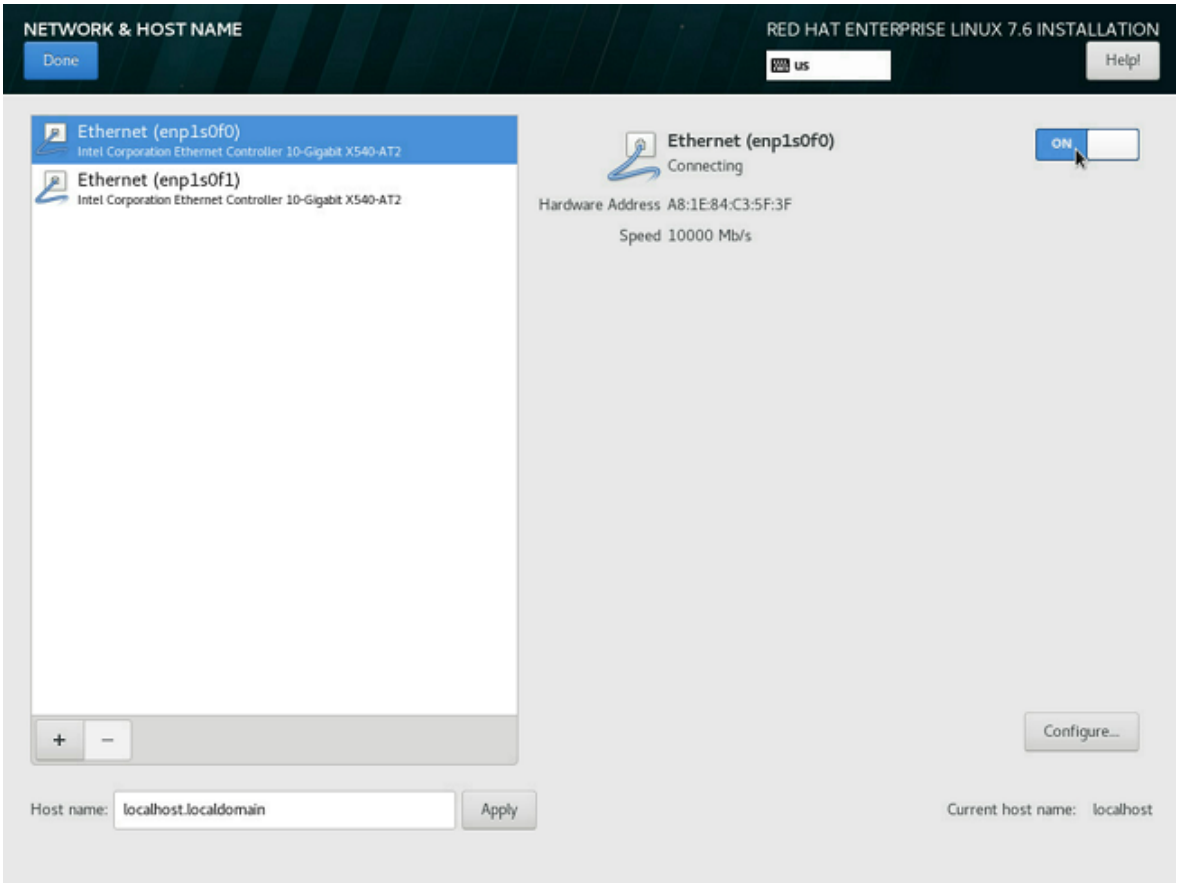


Click **Done** again and then click **Accept Changes** to write all the customizations to disk.

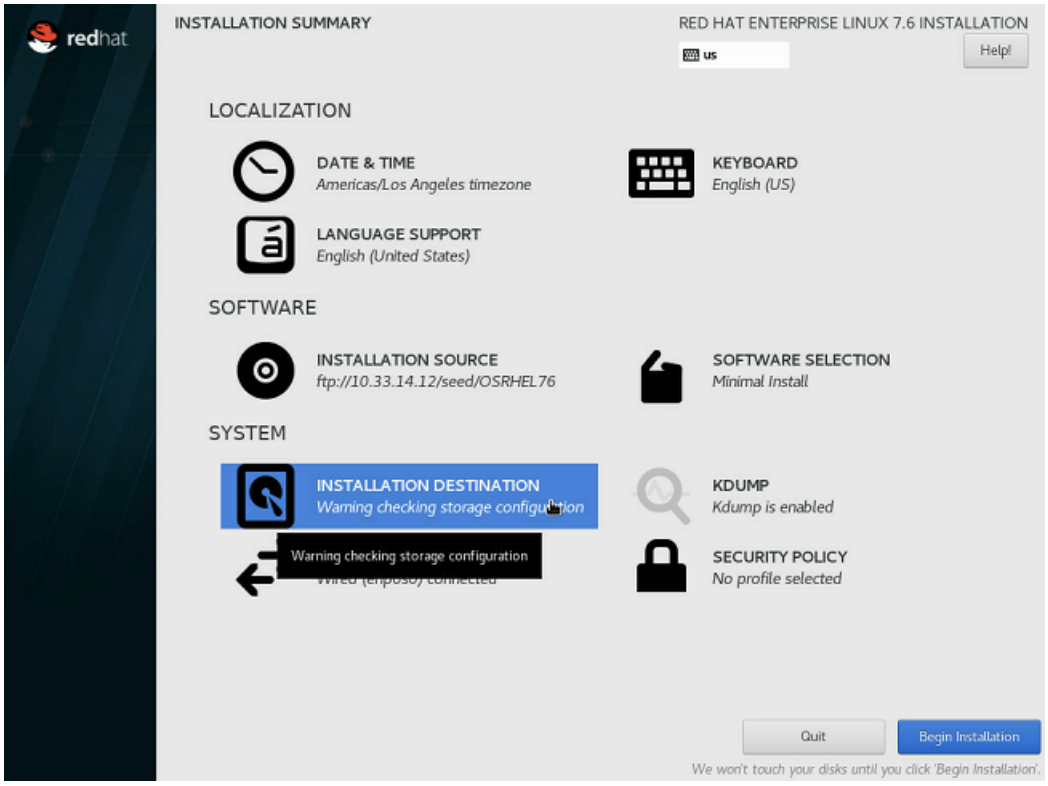


5. Configure Ethernet.

   Select and enable the Ethernet device. This defaults to DHCP and can be changed for static IP configurations under Configure.



6. From the INSTALLATION SUMMARY screen, click **Begin installation** to start the installation.



a). While the installation process is running, set your password (ROOT PASSWORD) and create a new user (USER CREATION) from the Configuration screen.

b). When the installation completes, click **Reboot** to reboot the system.

> **Note:** If you have installed Red Hat Enterprise Linux 7.5 and are using the BMC remote console, then follow the instructions provided in the release notes under Black screen on BMC Remote Console with Red Hat Enterprise Linux 7.5.

7. Register the system with the Red Hat Enterprise Customer Portal to complete the initial setup.

   - If you installed with 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 installed with any other base environment, log on to the system as root user and then register the system.

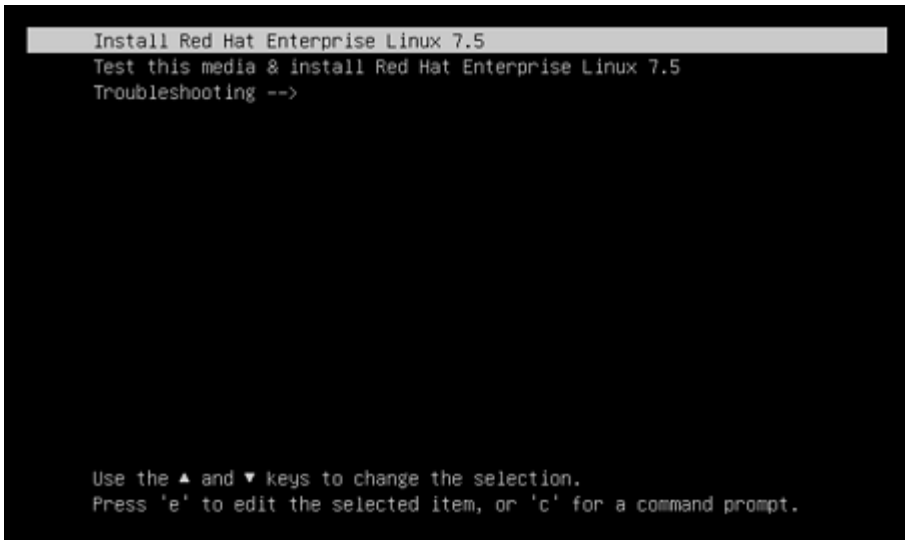
     ```
     subscription-manager register --auto-attach --username=user_name --
     password=password
     ```

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

2.4.2. Installing on the DGX-2

This section assumes you have booted the Red Hat Enterprise Linux ISO image, either locally or remotely.

1. After booting the ISO image through either the BMC or from the USB drive, select **Install Red Hat Enterprise Linux** and then press **Enter** to start the installation.



   The Red Hat Enterprise Linux installer starts.

   For guidance on using the installer, refer to Red Hat Enterprise Linux Quick Installation Guide (https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/installation_guide/chap-simple-install).

2. Configure the language, region, date, time, keyboard, and other configuration options you may need from the Installation Summary screen.
3. Select the software packages that you want to install.

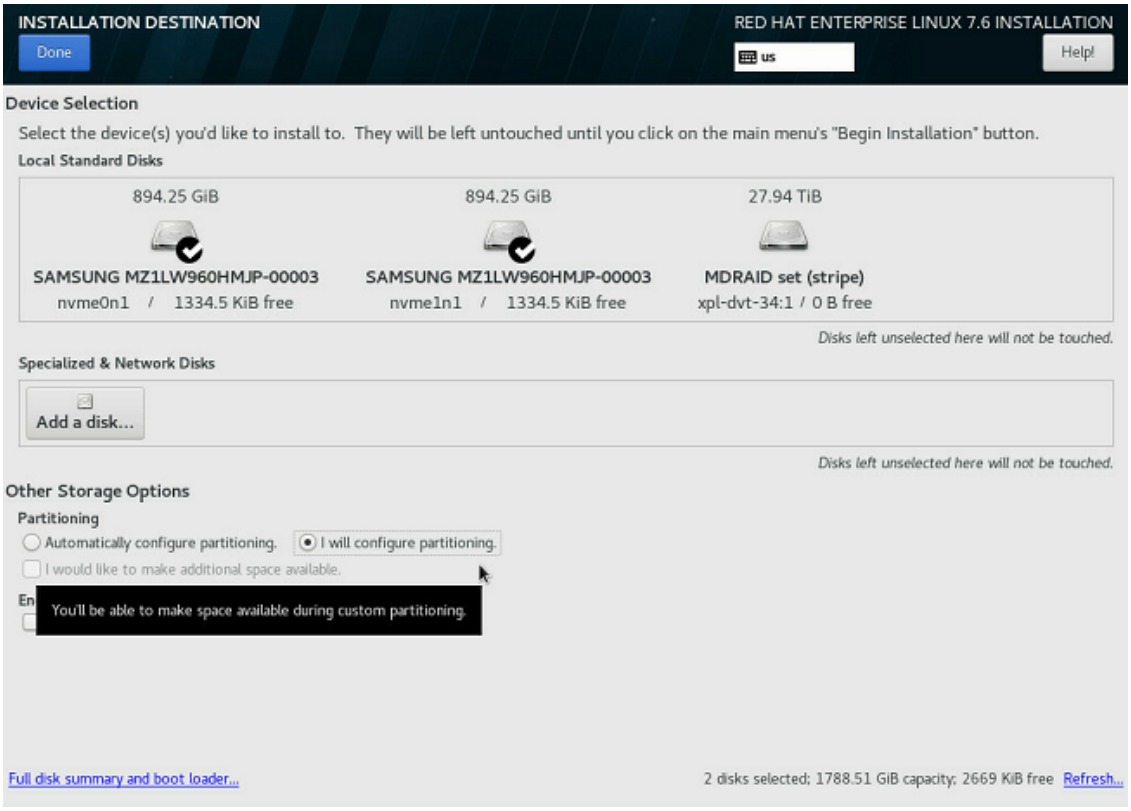
   For detailed instructions, refer to Software Selection in Red Hat Enterprise Linux Installation Guide (https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/installation_guide/sect-package-selection-x86).

4. Set up the boot disks.

   The DGX OS, which is installed in the factory before shipping out the DGX-2, creates a RAID-1 array on the two M.2 NVMe boot disks. Therefore, it is necessary to select both boot disks when installing RHEL 7 on a DGX-2.

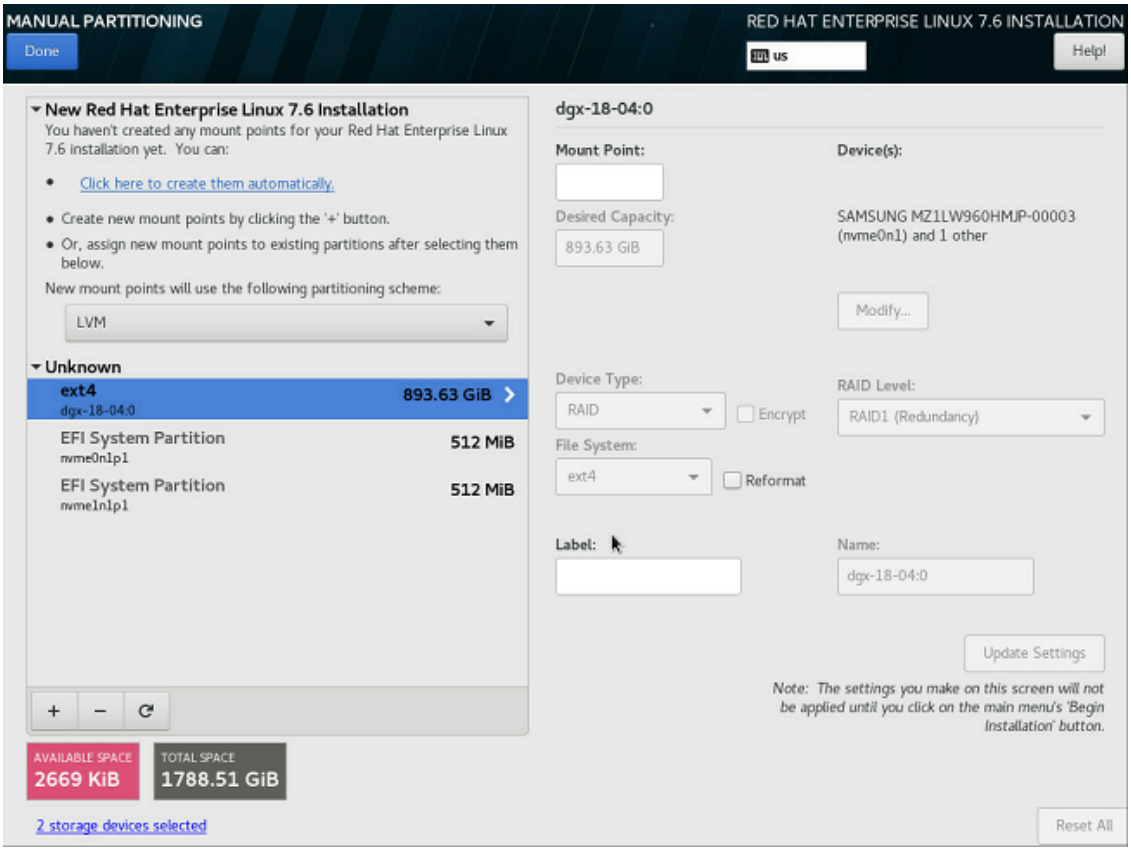
   a). From the Installation Summary screen, click **INSTALLATION DESTINATION**.

b). At the Installation Destination screen, select both M.2 disks (i.e. the 894.25 GB ones) and the "**I will configure partitioning**" radio button, then click **Done**.



The Manual Partitioning window appears.

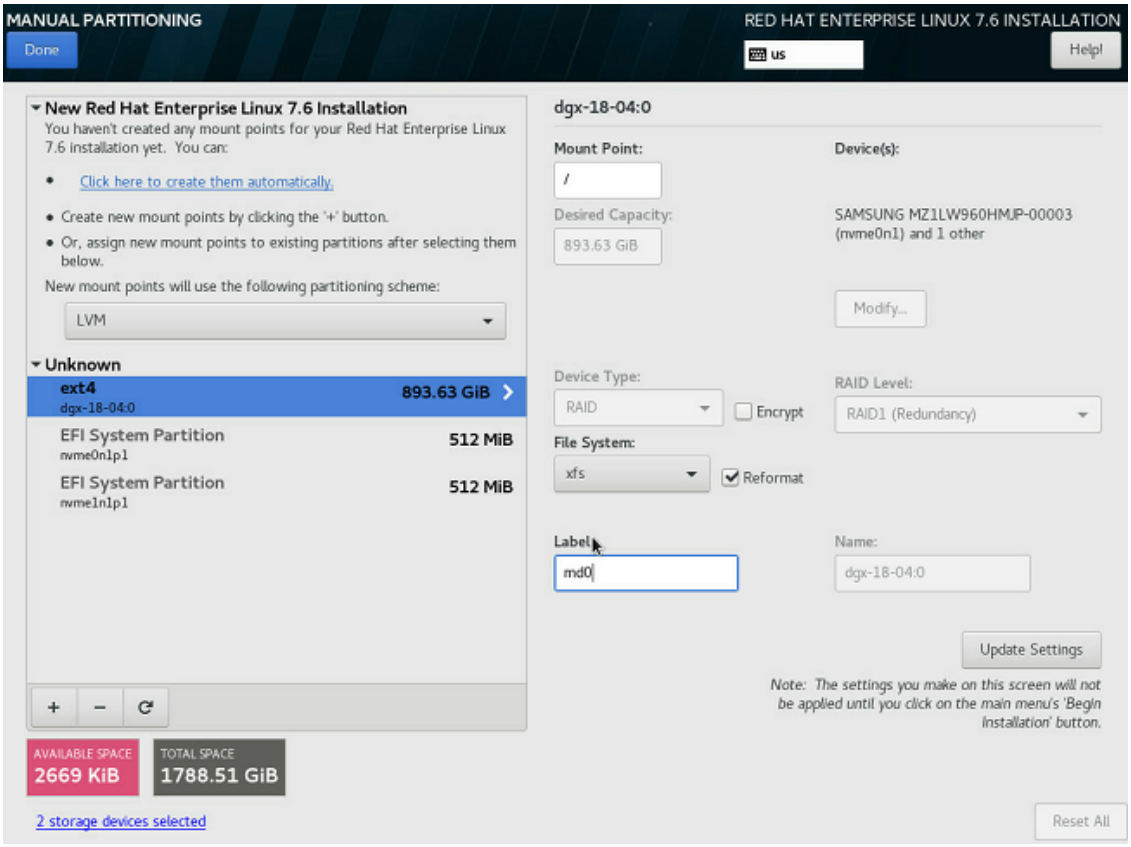
c). Expand the **Unknown** drop-down menu.



This shows the disks that Red Hat Enterprise Linux has identified. They will be

- the RAID-1 root partition (ext4, 893 GB),
- the first EFI system partition (ESP) (nvme0n1p1, 512 MB), and
- the second ESP (nvme1n1p1, 512 MB).

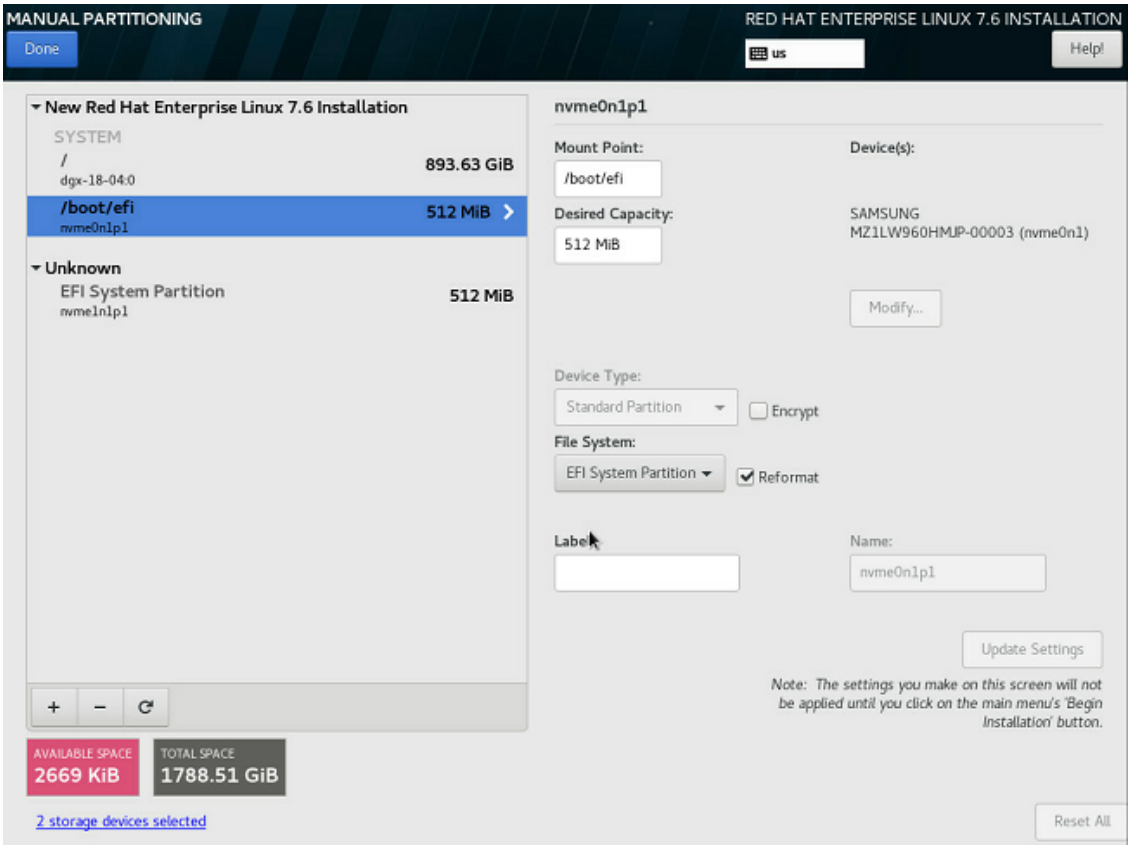
d). Select the **ext4** partition and reformat it as **xfs**, then set the mount point to "/" and label it as "**md0**"



e). Click **Update Settings**.

5. Create an EFI partition.

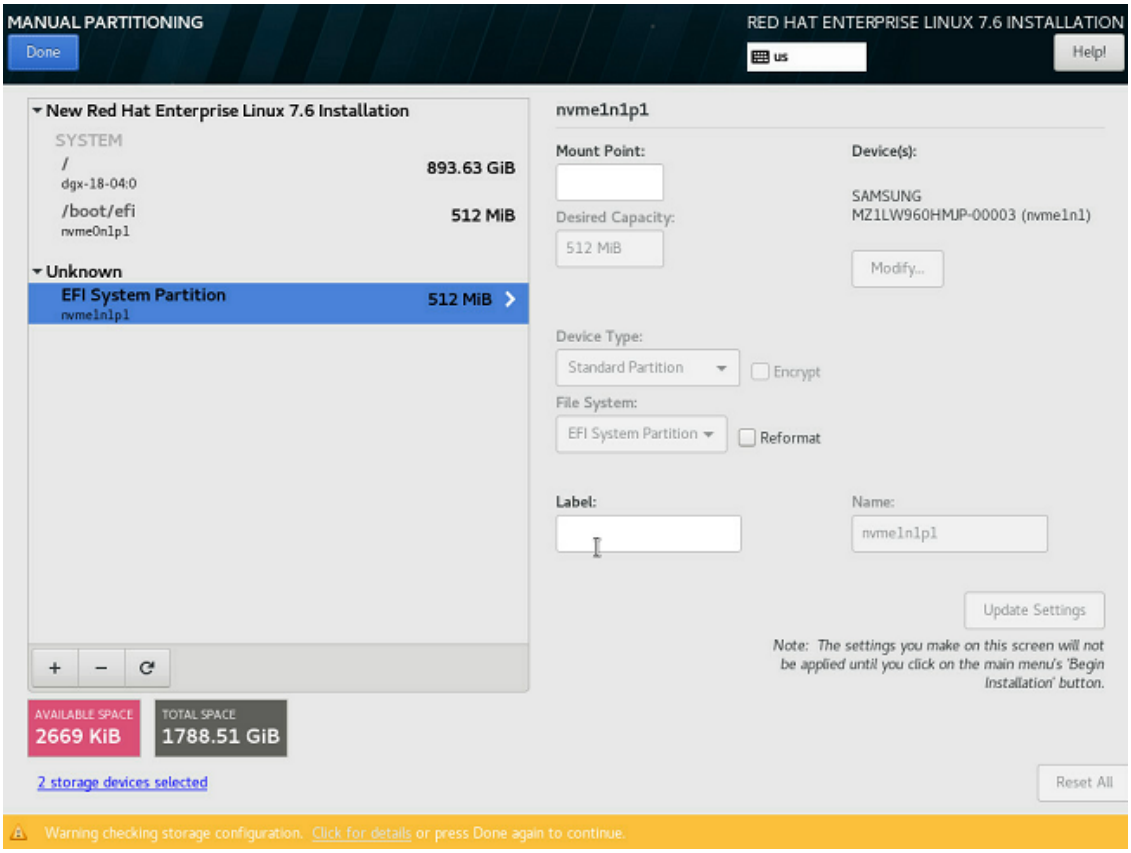
   a). Select the **nvme0n1p1** partition and reformat it as an **EFI System Partition**, then set the mount point to "`/boot/efi`" and click **Update Settings**.



At this point, the two partitions you configured will be under the **New Red Hat Enterprise Linux Installation** drop down menu, and the **nvme1n1p1** EFI partition will still be in **Unknown**. We can't create a second ESP because Red Hat Enterprise Linux won't allow you to create two partitions mounted on the same mount point, i.e. `/boot/efi`, even if they are on different disks. You can replicate the ESP after installing the "DGX System Management" yum group, which provides a tool to accomplish this.

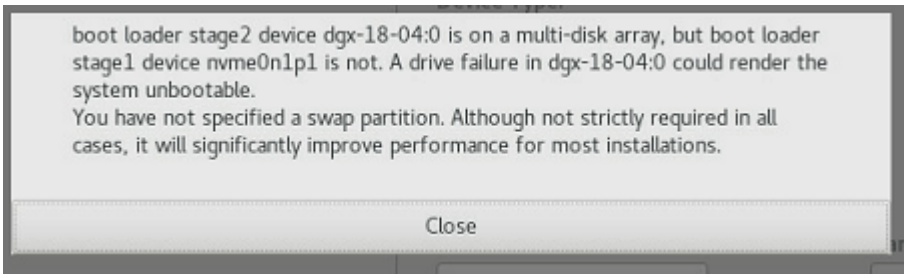
b). Click **Done**.

A yellow-highlighted warning appears on the bottom of the screen.



c). Click the yellow-highlighted warning message.

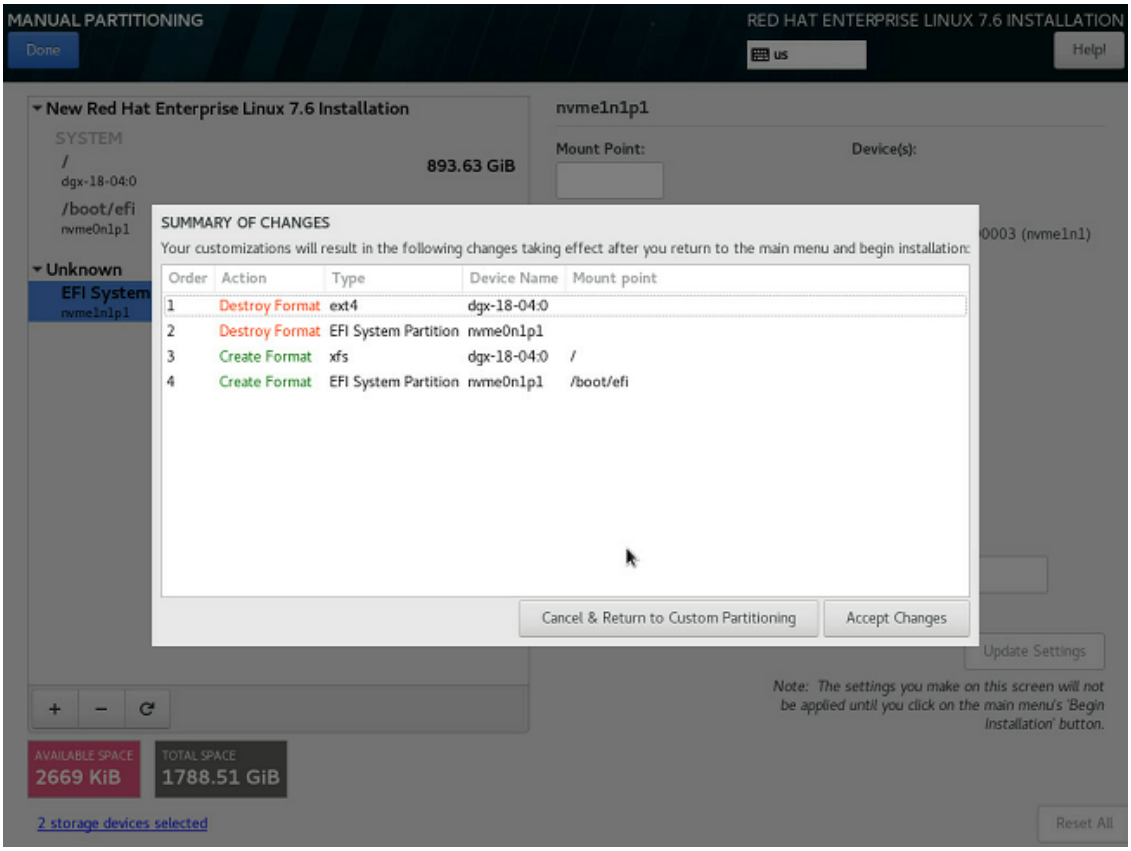
The following message appears.



The message appears because the DGX OS from the factory has created a RAID 1 array for the root partition, and has replicated the EFI system partitions.

d). Close the warning and then press **Done**.

A summary of the partitioning steps appears.



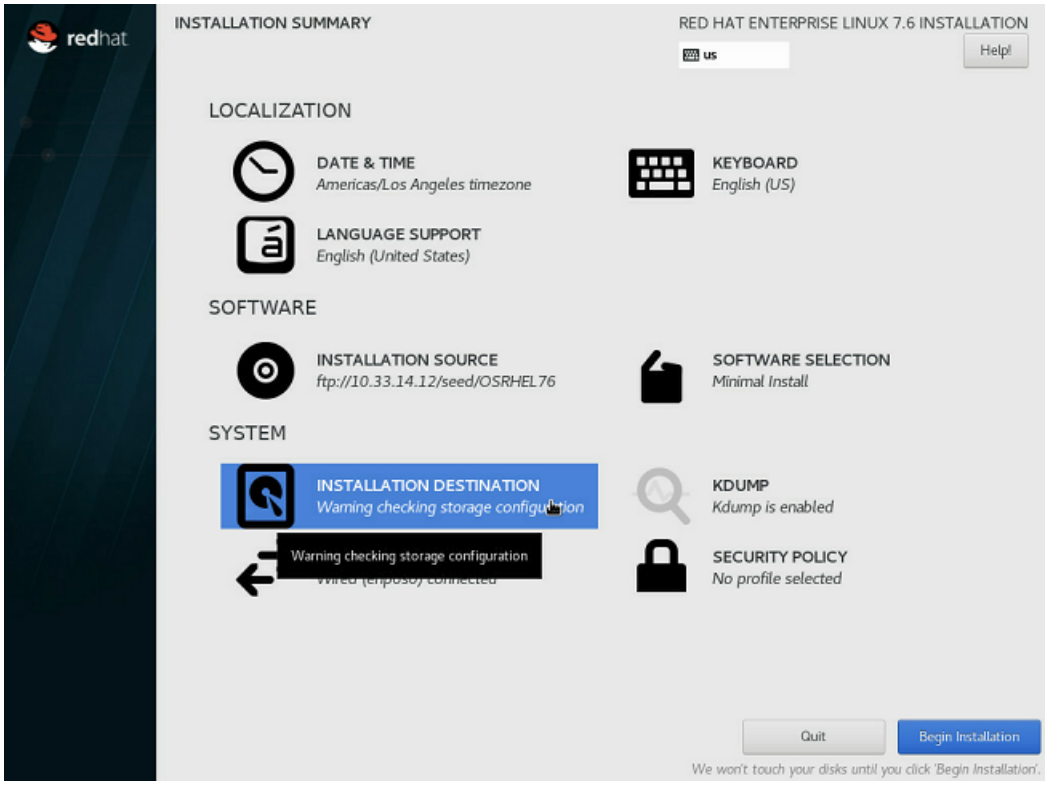
e). Click **Accept Changes** to get back to the main Installation Summary Screen.

There should no longer be a yellow warning icon on the **System: Installation Destination** option.

6. Configure Ethernet.

   Select and enable the Ethernet device. This defaults to DHCP and can be changed for static IP configurations under **Configure**.

7. From the INSTALLATION SUMMARY screen, click **Begin installation** to start the installation.



a). While the installation process is running, set your password (ROOT PASSWORD) and create a new user (USER CREATION) from the Configuration screen.

b). When the installation completes, click **Reboot** to reboot the system.

> **Note:** If you have installed Red Hat Enterprise Linux 7.5 and are using the BMC remote console, then follow the instructions provided in the release notes under **Black screen on BMC Remote Console with Red Hat Enterprise Linux 7.5**.

8. Register the system with the Red Hat Enterprise Customer Portal to complete the initial setup.

   - If you installed with 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 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
     ```

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

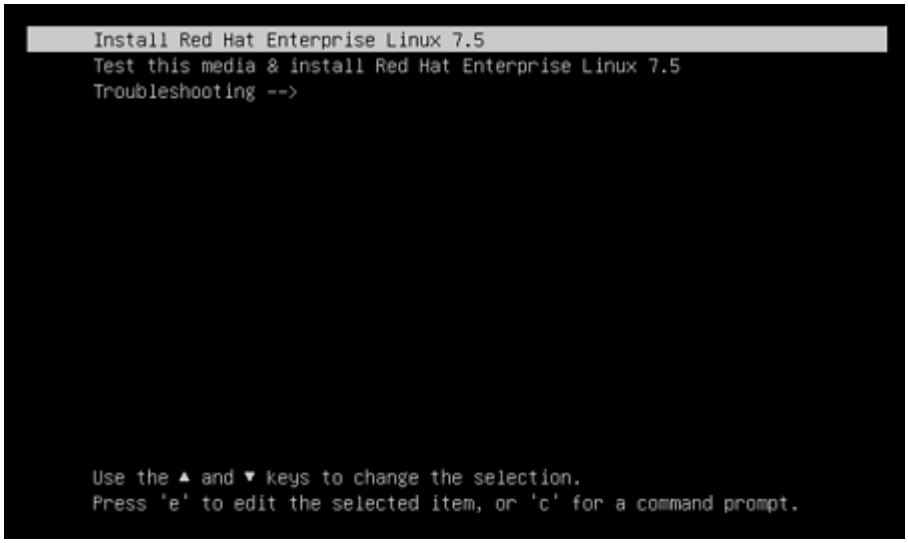
2.4.3. Installing on the DGX A100

This section assumes you have booted the Red Hat Enterprise Linux ISO image, either locally or remotely.

For instructions applicable to the DGX-1 and the DGX Station, see Installing on the DGX-1, DGX Station, or DGX Station A100.

For instructions applicable to the DGX-2, see Installing on the DGX-2.

1. After booting the ISO image through either the BMC or from the USB drive, select **Install Red Hat Enterprise Linux** and then press **Enter** to start the installation.



   The Red Hat Enterprise Linux installer starts.

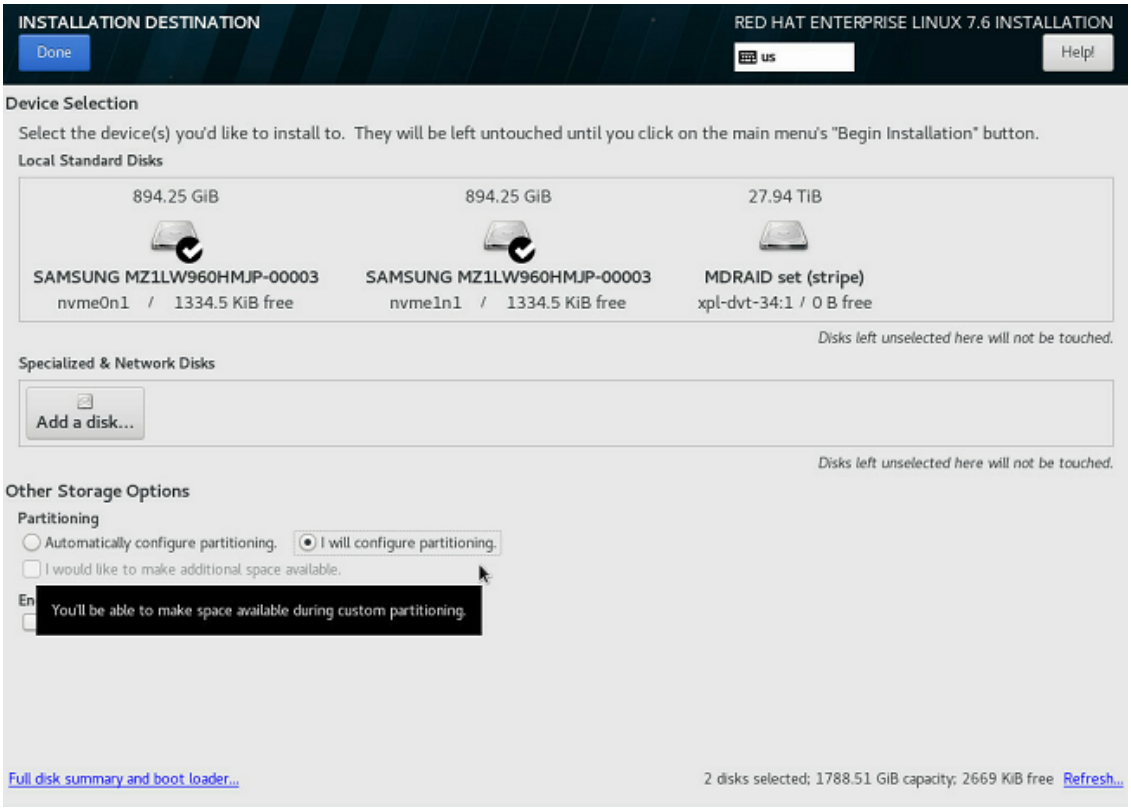
   For guidance on using the installer, refer to Red Hat Enterprise Linux Quick Installation Guide (https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/installation_guide/chap-simple-install).
2. Configure the language, region, date, time, keyboard, and other configuration options you may need from the **Installation Summary** screen.
3. Select the software packages that you want to install.

   For detailed instructions, refer to Software Selection in Red Hat Enterprise Linux Installation Guide (https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/installation_guide/sect-package-selection-x86).
4. Set up the boot disks.

   The DGX OS, which is installed in the factory before shipping out the DGX A100, creates a RAID-1 array on the two M.2 NVMe boot disks. Therefore, it is necessary to select both boot disks when installing RHEL 7 on a DGX A100.

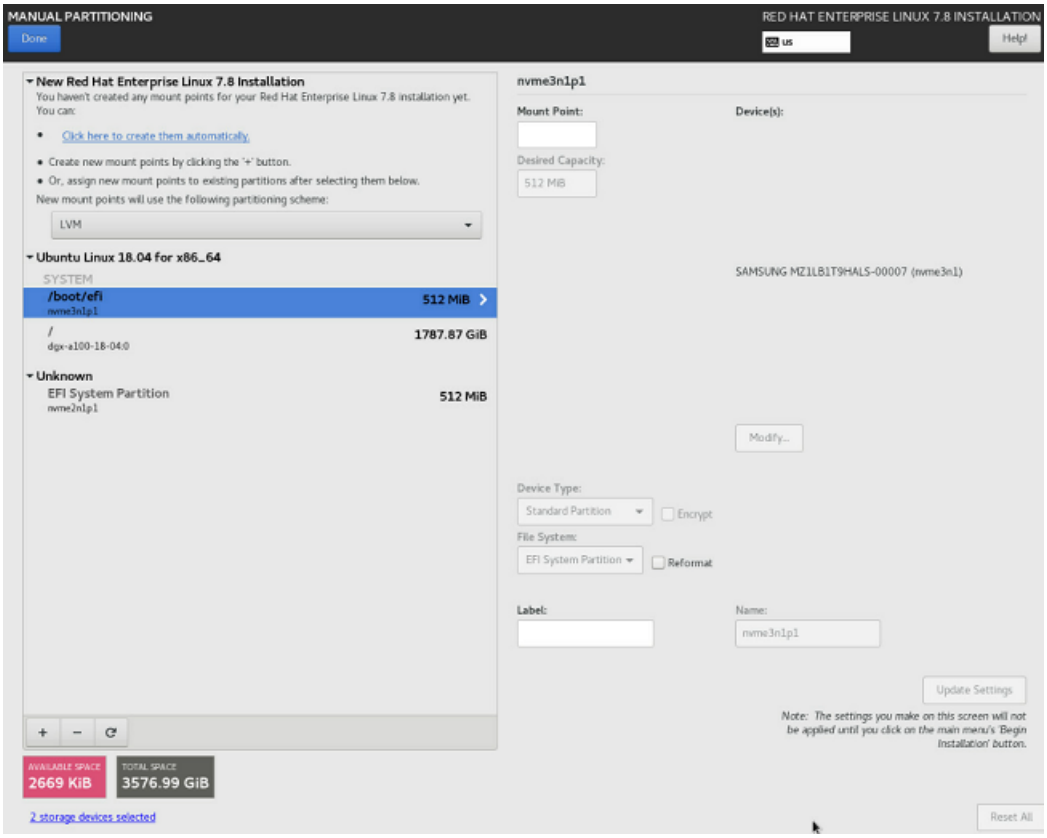
   a). From the Installation Summary screen, click **INSTALLATION DESTINATION**.

b). At the Installation Destination screen, select both M.2 disks (i.e. 1788 GiB on DGX A100) and the "**I will configure partitioning**" radio button, then click **Done**.



The Manual Partitioning window appears.

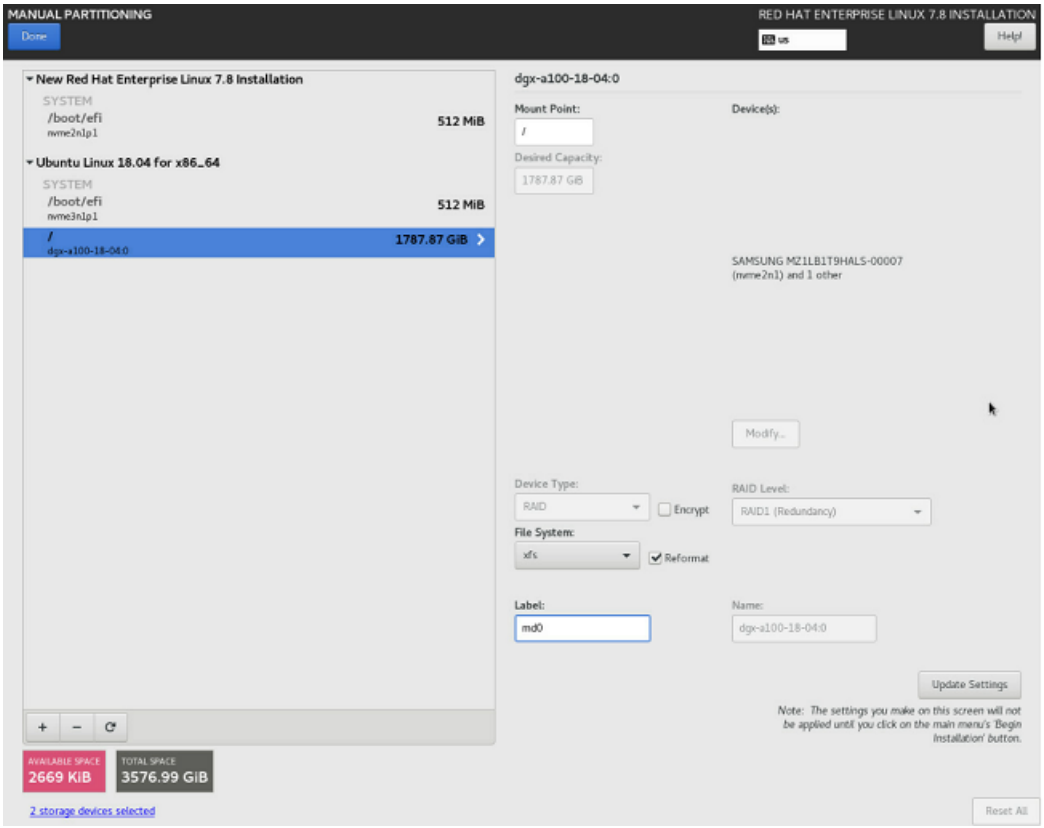
c). Expand the **Unknown** drop-down menu.



This shows the disks that Red Hat Enterprise Linux has identified. They will be

- the RAID-1 root partition (ext4, 1787.87 GB),
- the first EFI system partition (ESP) (nvme3n1p1, 512 MB), and
- the second ESP (nvme2n1p1, 512 MB).

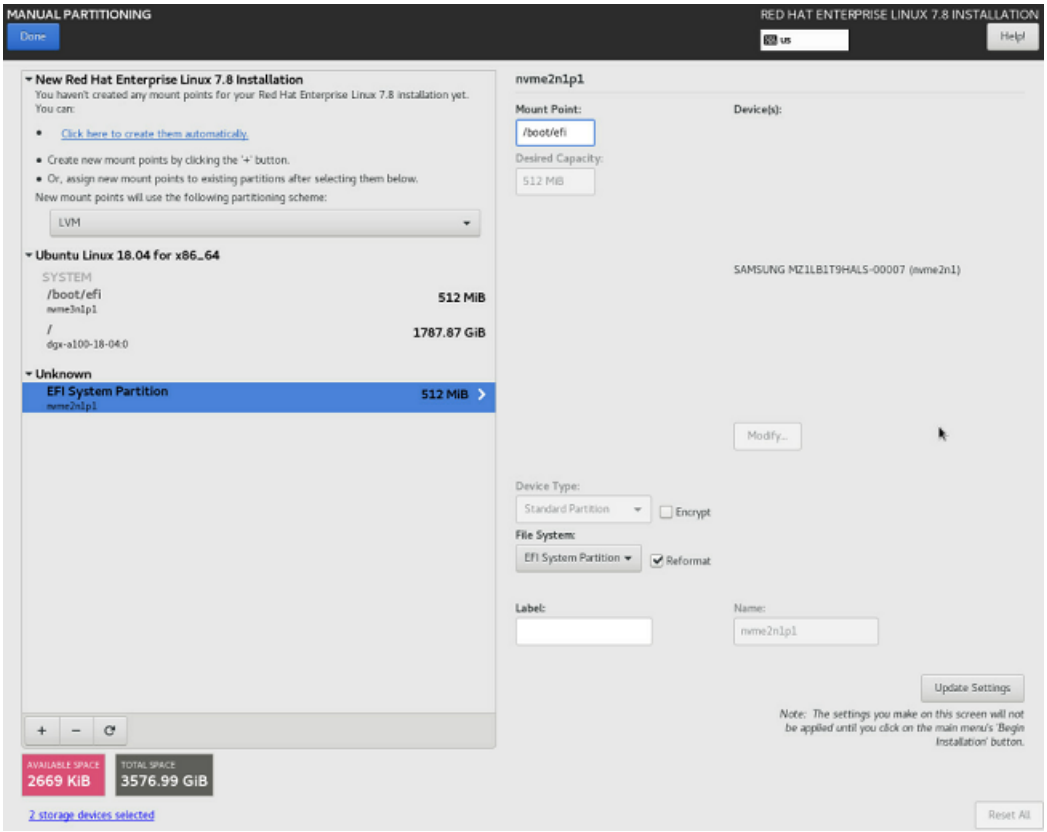
d). Select the **ext4** partition and reformat it as **xfs**, then set the mount point to "/" and label it as "**md0**"



e). Click **Update Settings**.

5. Create an EFI partition.

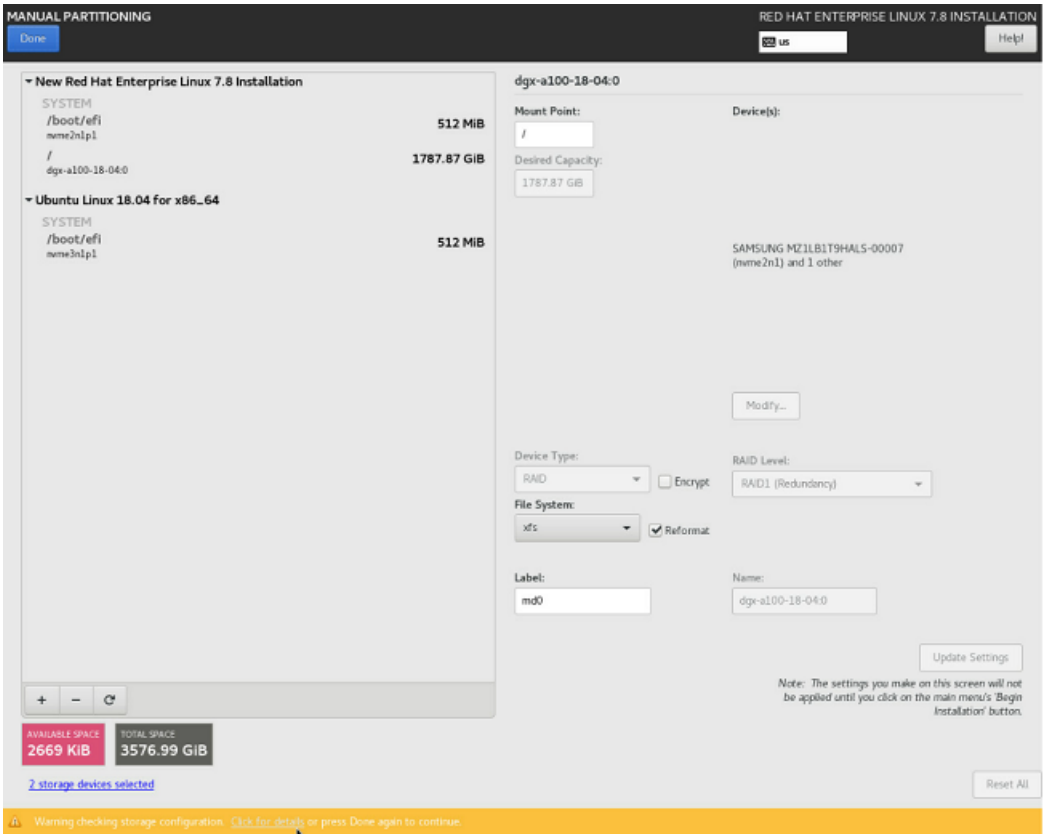
   a). Select the second EFI partition, in the Unknown section (**nvme2n1p1** on DGX A100), reformat it as an **EFI System Partition**, then set the mount point to "`/boot/efi`" and click **Update Settings**.



At this point, the two partitions you configured will be under the **New Red Hat Enterprise Linux Installation** drop down menu, and the Ubuntu EFI partition (**nvme3n1p1** on DGX A100) will be under **Ubuntu**. We can't create a second ESP because Red Hat Enterprise Linux won't allow you to create two partitions mounted on the same mount point, i.e. `/boot/efi`, even if they are on different disks. You can replicate the ESP after installing the "DGX System Management" yum group, which provides a tool to accomplish this.

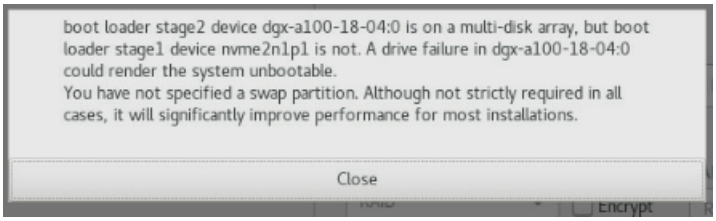
b). Click **Done**.

A yellow-highlighted warning appears on the bottom of the screen.



c). Click the yellow-highlighted warning message.

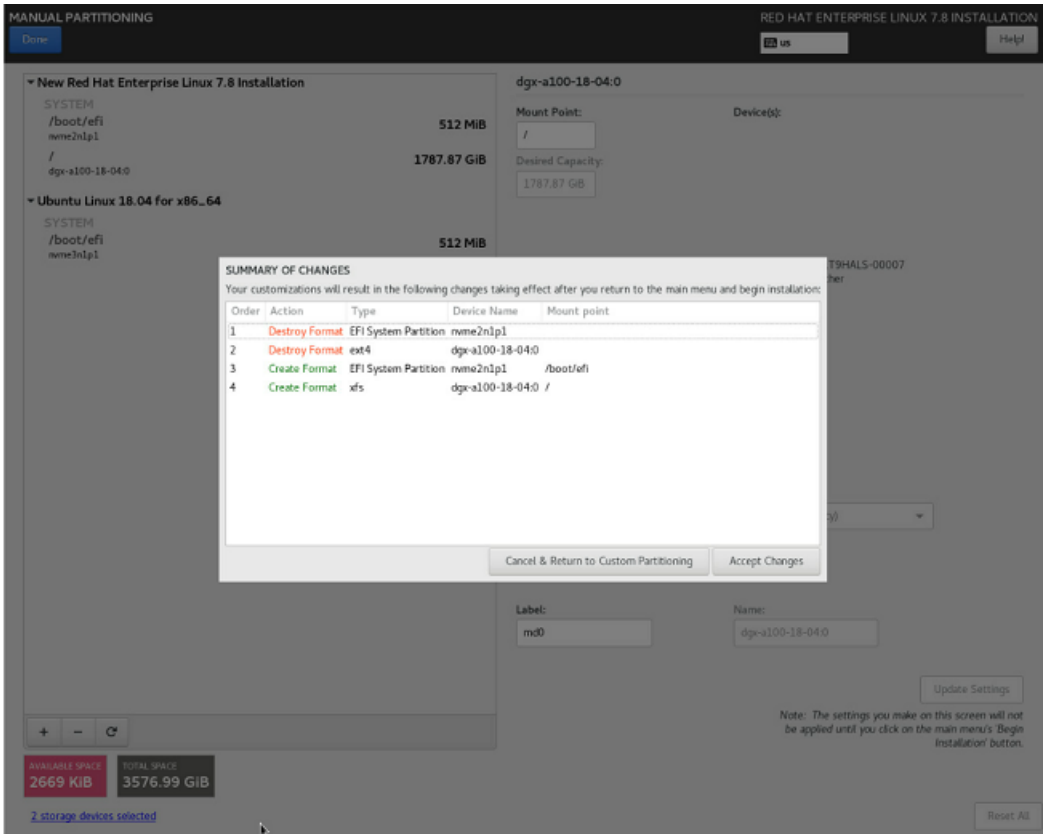
The following message appears.



The message appears because the DGX OS from the factory has created a RAID 1 array for the root partition, and has replicated the EFI system partitions.

d). Close the warning and then click **Done**.

A summary of the partitioning steps appears.



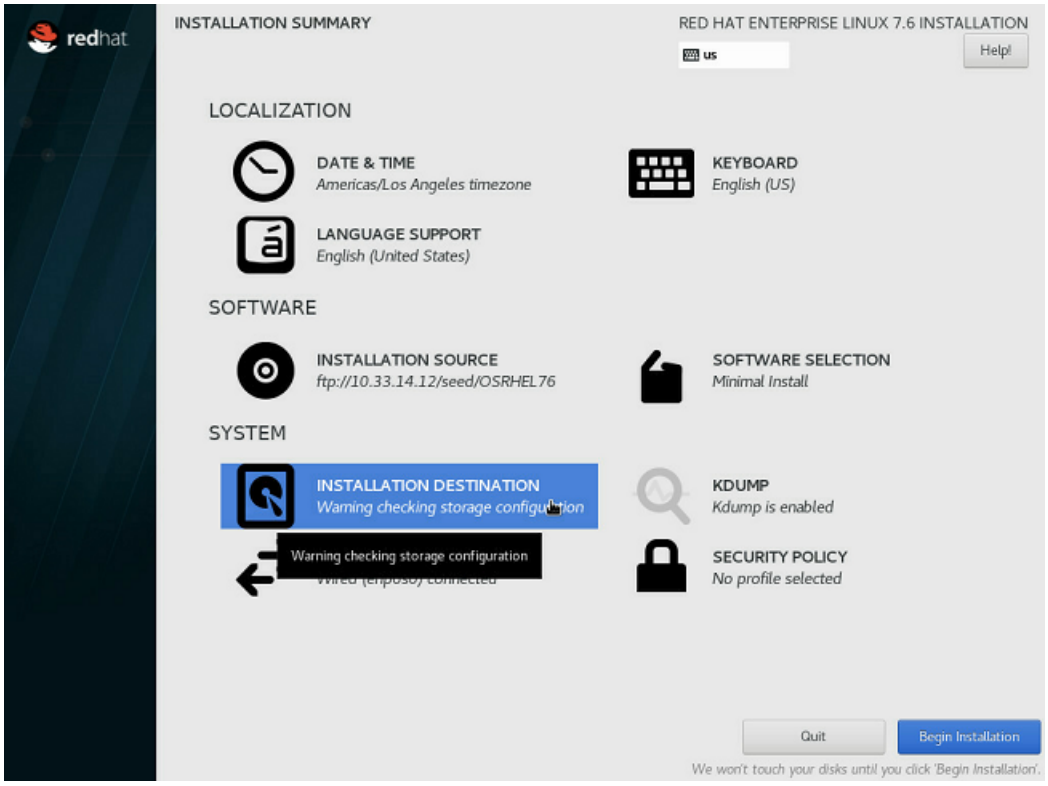
e). Click **Accept Changes** to get back to the main Installation Summary Screen.

There should no longer be a yellow warning icon on the **System: Installation Destination** option.

6. Configure Ethernet.

   Select and enable the Ethernet device. This defaults to DHCP and can be changed for static IP configurations under **Configure**. This screen also lets you set the hostname.

7. From the INSTALLATION SUMMARY screen, click **Begin installation** to start the installation.



a). While the installation process is running, set your password (ROOT PASSWORD) and create a new user (USER CREATION) from the Configuration screen.

b). When the installation completes, click **Reboot** to reboot the system.

> **Note:** If you have installed Red Hat Enterprise Linux 7.5 and are using the BMC remote console, then follow the instructions provided in the release notes under **Black screen on BMC Remote Console with Red Hat Enterprise Linux 7.5**.

8. Register the system with the Red Hat Enterprise Customer Portal to complete the initial setup.

   - If you installed with 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 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
     ```

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

Chapter 3. Installing the DGX Software

This section requires that you have already installed Red Hat Enterprise Linux" or derived operating system on the DGX server.

3.1. Configuring a System Proxy

If your network requires use of a proxy, then

- Edit the file `/etc/yum.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>
```

- Make sure that the following domains are not blocked and that the system can access them.
  - cdn.redhat.com
  - international.download.nvidia.com

3.2. Enabling the Repositories

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-7-server-extras-rpms
sudo subscription-manager repos --enable=rhel-7-server-optional-rpms
```

2. Run the following commands to install the DGX software installation package and enable the NVIDIA DGX software repository.

> **!** **ATTENTION:** By running these commands you are confirming that you have read and agree to be bound by the DGX Software License Agreement. You are also confirming that you understand that any pre-release software and materials available that you elect to install in a DGX may not be fully functional, may contain errors or design flaws, and may 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 risk.

a). Install the NVIDIA DGX Package for Red Hat Enterprise Linux.

```
yum install -y \
https://international.download.nvidia.com/dgx/repos/rhel-files/dgx-repo-
setup-20.03-1.el7.x86_64.rpm
```

b). Enable the R450 repository to move to the R450 package.

The step is required for DGX A100 and DGX Station A100, but optional for other DGX platforms.

- Either edit `/etc/yum.repos.d/nvidia-dgx-7.repo` and set `enabled=1`,

```
[nvidia-dgx-7-r450-cuda11-0]
name=NVIDIA DGX EL7 R450-CUDA11-0
baseurl=https://international.download.nvidia.com/dgx/repos/rhel7-r450-
cuda11-0/
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-dgx-cosmos-support
```

- Or (if you have the `yum-utils` package installed), issue the following.

```
sudo yum-config-manager --enable nvidia-dgx-7-r450-cuda11-0
```

3.3. Installing Required Components

3.3.1. Installing DGX Tools and Updating Configuration Files

1. Install the kernel development tools.
   a). Install the kernel-devel package.

   The kernel-devel package provides kernel headers required for the NVIDIA CUDA driver. Use the following command to install the kernel headers for the kernel version that is currently running on the system.

   ```
   sudo yum install -y "kernel-devel-uname-r == $(uname -r)"
   ```

   b). Ensure that you have installed the latest version of `gcc` installed, as older versions may not support all of the features required to build the driver.

   ```
   sudo yum install -y gcc
   ```

2. Install DGX tools and configuration files.

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

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

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

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

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

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

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

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

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

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

The configuration changes take effect only after rebooting the system, which is covered in the next step.

3. Update the kernel.
   a). Issue the following.
   ```
   $ sudo yum update
   ```
   Performing this update also updates the installed Red Hat Enterprise Linux 7 distribution to the latest version. To check the latest Red Hat Enterprise Linux 7 version, visit https://access.redhat.com/articles/3078.
   b). Reboot the server into the updated kernel.
   ```
   $ sudo reboot
   ```

3.3.2. Configuring the /raid Partition

The DGX servers and the DGX Station include multiple SSDs for data caching or data storage. Configure these SSDs as a RAID array in a partition mounted at /raid. For the DGX servers, these SSDs are intended to be used as a data cache for NFS mounted directories. For the DGX Station, these SSDs are intended to be used either for local persistent storage or as a data cache for NFS mounted directories.

3.3.2.1. Configuring the /raid Partition as an NFS Cache

If you are using the data SSDs for caching NFS reads, configure these SSDs as a RAID 0 or RAID 5 array, mounted at /raid and update the cachefilesd configuration to use the /raid partition.

1. Configure the RAID array.

   This will create the RAID group, mount it to /raid, and create an appropriate entry in /etc/fstab.

   - To configure a RAID 0 array, issue the following:
   ```
   sudo configure_raid_array.py -c -f
   ```
   - To configure a RAID 5 array, issue the following:
   ```
   sudo configure_raid_array.py -c -5 -f
   ```

   Note:

   The RAID array must be configured before installing dgx-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 dgx-conf-cachefilesd has already been installed, be sure to restore the label manually before restarting cachefilesd.

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

2. Install dgx-conf-cachefilesd to update the cachefilesd configuration to use the /raid partition.
   ```
   sudo yum install -y dgx-conf-cachefilesd
   ```

3.3.2.2. Configuring the `/raid` Partition for Local Persistent Storage

If you are using the data SSDs in the DGX Station for local persistent storage, configure these SSDs as a RAID 0 or RAID 5 array, mounted at `/raid`.

RAID 0 provides the maximum storage capacity, but does not provide any redundancy. If a single SSD in the array fails, all data stored on the array is lost. RAID 5 provides some level of protection against failure of a single SSD but with lower storage capacity than RAID 0.

- To configure a RAID 0 array, run the following command.

```
sudo configure_raid_array.py -c -f
```

- To configure a RAID 5 array, run the following command.

```
sudo configure_raid_array.py -c -f -5
```

These commands will create the RAID group, mount it to `/raid`, and create an appropriate entry in `/etc/fstab`.

3.3.3. Installing and Loading the NVIDIA CUDA Drivers

1. Install the driver package.

   This will build and install the driver kernel modules. The installation of the dkms-nvidia package can take approximately five minutes.

```
sudo yum install -y cuda-drivers dgx-persistence-mode
```

   Note: Red Hat Enterprise Linux 7.5 ships with OpenGL libraries that conflict with versions included in the CUDA drivers. Depending on the Software Selection performed in Installing Red Hat Enterprise Linux, you might encounter an error with the following libraries: mesa-libGL, mesa-libEGL, or mesa-libGLES. Simply remove these libraries and re-issue the yum install command.

```
sudo rpm -e mesa-libGL.x86_64 --nodeps
sudo rpm -e mesa-libEGL.x86_64 --nodeps
sudo rpm -e mesa-libGLES.x86_64 --nodeps
sudo yum install -y cuda-drivers dgx-persistence-mode
```

2. (**DGX Station 100 Only**): Install additional required Station A100 packages.

   These packages must be installed after installing the cuda-drivers package.

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

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

```
sudo reboot
```

4. After the system has rebooted, verify that the drivers have been loaded and are handling the NVIDIA devices.

```
nvidia-smi
```

   The output should show all available GPUs.

   Example: Output from a DGX-1 system

```
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 450.51.05    Driver Version: 450.51.05 CUDA Version: 11.0        |
|-------------------------------+----------------------+----------------------+
| GPU Name        Persistence-M | Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan Temp Perf Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|===============================+======================+======================|
|   0 Tesla V100-SXM2...  On | ...00:06:00.0 Off |                    0 |
| N/A  33C    P0    45W / 300W |   0MiB / 32480MiB |      0%      Default |
+-------------------------------+----------------------+----------------------+
|   1 Tesla V100-SXM2...  On | ...00:07:00.0 Off |                    0 |
| N/A  35C    P0    44W / 300W |   0MiB / 32480MiB |      0%      Default |
+-------------------------------+----------------------+----------------------+
:                               :                      :                      :
+-------------------------------+----------------------+----------------------+
|   7 Tesla V100-SXM2...  On | ...00:8A:00.0 Off |                    0 |
| N/A  34C    P0    44W / 300W |   0MiB / 32480MiB |      0%      Default |
+-------------------------------+----------------------+----------------------+
+-----------------------------------------------------------------------------+
| Processes:                                                       GPU Memory |
|  GPU       PID   Type   Process name                             Usage      |
|=============================================================================|
|  No running processes found                                                 |
+-----------------------------------------------------------------------------+
```

3.3.4. Installing the NVIDIA Container Runtime

1. Install Docker 1.13 from the `rhel-7-server-extras-rpms` repository.

   ```
   sudo yum install -y docker
   ```

2. Install the **NVIDIA Container Runtime** group.

   ```
   sudo yum groupinstall -y 'NVIDIA Container Runtime'
   ```

3. Run the following command to verify the installation.

   - If the R418 driver package is installed:

     ```
     sudo docker run --security-opt label=type:nvidia_container_t --rm nvcr.io/
     nvidia/cuda:10.0-base nvidia-smi
     ```

   - If the R450 driver package is installed:

     ```
     sudo docker run --security-opt label=type:nvidia_container_t --rm nvcr.io/
     nvidia/cuda:11.0-base nvidia-smi
     ```

   The output should show all available GPUs.

   Example: Output on a DGX -1 system

```
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 450.51.05   Driver Version: 450.51.05  CUDA Version: 11.0        |
|-------------------------------+----------------------+----------------------+
| GPU Name        Persistence-M | Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan Temp Perf Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|===============================+======================+======================|
|   0 Tesla V100-SXM2...  On | ...00:06:00.0 Off |                    0 |
| N/A  33C    P0    45W / 300W |   0MiB / 32480MiB |      0%      Default |
+-------------------------------+----------------------+----------------------+
|   1 Tesla V100-SXM2...  On | ...00:07:00.0 Off |                    0 |
| N/A  35C    P0    44W / 300W |   0MiB / 32480MiB |      0%      Default |
+-------------------------------+----------------------+----------------------+
:                               :                      :                      :
+-------------------------------+----------------------+----------------------+
|   7 Tesla V100-SXM2...  On | ...00:8A:00.0 Off |                    0 |
| N/A  34C    P0    44W / 300W |   0MiB / 32480MiB |      0%      Default |
+-------------------------------+----------------------+----------------------+
+-----------------------------------------------------------------------------+
| Processes:                                                       GPU Memory |
```

```
|  GPU       PID   Type   Process name                             Usage      |
|=============================================================================|
|  No running processes found                                                 |
+-----------------------------------------------------------------------------+
```

See the section Running Containers for more information about this command. For a description of nvcr.io, see the NGC Registry Spaces documentation.

To ensure that Docker can access the NGC container registry through a proxy, refer to the Red Hat customer portal knowledge base article Configure Docker to use a proxy with or without authentication.

3.4. Installing Diagnostic Components

NVIDIA System Management (NVSM) provides the diagnostic components for NVIDIA DGX systems. NVSM is a software framework for monitoring NVIDIA DGX **server** nodes in a data center. It includes active health monitoring, system alerts, and log generation. The NVSM CLI can also be used for checking the health of and obtaining diagnostic information for DGX Station workstations.

Note: The diagnostic components for NVIDIA DGX systems require Python 3. It is available from the Red Hat Enterprise Linux Software Collections (https://developers.redhat.com/products/softwarecollections/overview/). The Fedora EPEL repository also contains a version of Python 3; however, this combination has not been tested.

1. Enable the Red Hat Software Collections repository.
   ```
   sudo subscription-manager repos --enable=rhel-server-rhscl-7-rpms
   ```
   If you do not have access to the Red Hat Software Collections repository, refer to https://access.redhat.com/solutions/472793 for instructions on requesting access for free.
2. Install Python 3.6.
   ```
   sudo yum install -y rh-python36
   ```

   Important: The diagnostic components for NVIDIA DGX systems are not supported with the `python3` package. Be sure to only install the `rh-python36` package.
3. Install the DGX System Management package group.
   ```
   sudo yum groupinstall -y 'DGX System Management'
   ```

For information about using NVSM, see the NVIDIA System Management documentation.

3.5. Replicating the EFI System Partition on DGX-2 or DGX A100

This section applies only to the NVIDIA DGX-2 and DGX A100.

Once the 'DGX System Management' group is installed, the 'nvsm' tool can be used to replicate the EFI system partition (ESP) onto the second M.2 drive.

Important: Run these steps **ONLY IF**

- You are installing Red Hat Enterprise Linux on the NVIDIA DGX-2 or DGX A100, and
- You installed Red Hat Enterprise Linux on the RAID 1 array per instructions in the section Installing on DGX-2 or Installing on DGX A100 as appropriate.

1. Start the NVSM tool.

```
sudo nvsm
```

2. Navigate to `/systems/localhost/storage/volumes/md0`.

```
nvsm-> cd /systems/localhost/storage/volumes/md0
```

3. Start the rebuild process.

```
nvsm(/systems/localhost/storage/volumes/md0)-> start rebuild
```

a). At the first prompt, specify the second M.2 disk.

```
PROMPT: In order to rebuild this volume, a spare drive
        is required. Please specify the spare drive to
        use to rebuild md0.
Name of spare drive for md0 rebuild (CTRL-C to cancel): nvme1n1
```

This should be the M.2 disk on which you did NOT install the ESP.

- If you followed the instructions in the section Installing on DGX-2, this should be 'nvme1n1'
- If you followed the instructions in the section Installing on DGX A100, this should be 'nvme2n1'

b). At the second prompt, confirm that you want to proceed.

```
WARNING: Once the volume rebuild process is started, the
         process cannot be stopped.
Start RAID-1 rebuild on md0? [y/n] y
```

Upon successful completion, the following message should appear indicating that the ESP has been replicated:

```
/systems/localhost/storage/volumes/md0/rebuild started at 2019-03-07
 14:40:55.844542
RAID-1 rebuild exit status: ESP_REBUILT
```

If necessary, the RAID 1 array is rebuilt after the ESP has been replicated.

```
Finished rebuilding RAID-1 on volume md0
100.0% [========================================]
Status: Done
```

3.6. 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.

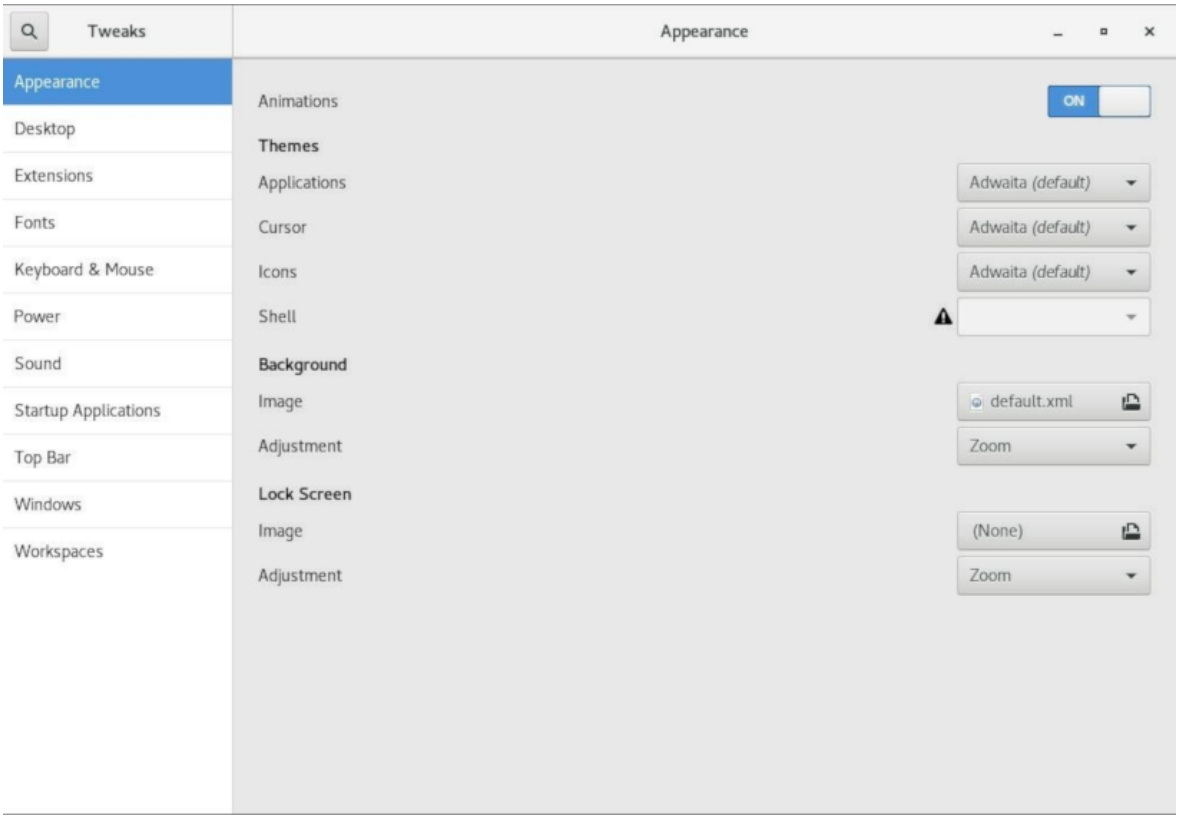
If you intend to use your DGX system as a development system for running deep learning applications on bare metal, then install the optional components as described in this section.

1. Install the CUDA toolkit.
   ```
   sudo yum install cuda
   ```
2. Install the NVIDIA Collectives Communication Library (NCCL) Runtime.
   ```
   sudo yum groupinstall 'NVIDIA Collectives Communication Library Runtime'
   ```
3. Install the CUDA Deep Neural Networks (cuDNN) Library Runtime.
   ```
   sudo yum groupinstall 'CUDA Deep Neural Networks Library Runtime'
   ```
4. Install NVIDIA TensorRT.
   ```
   sudo yum install tensorrt
   ```

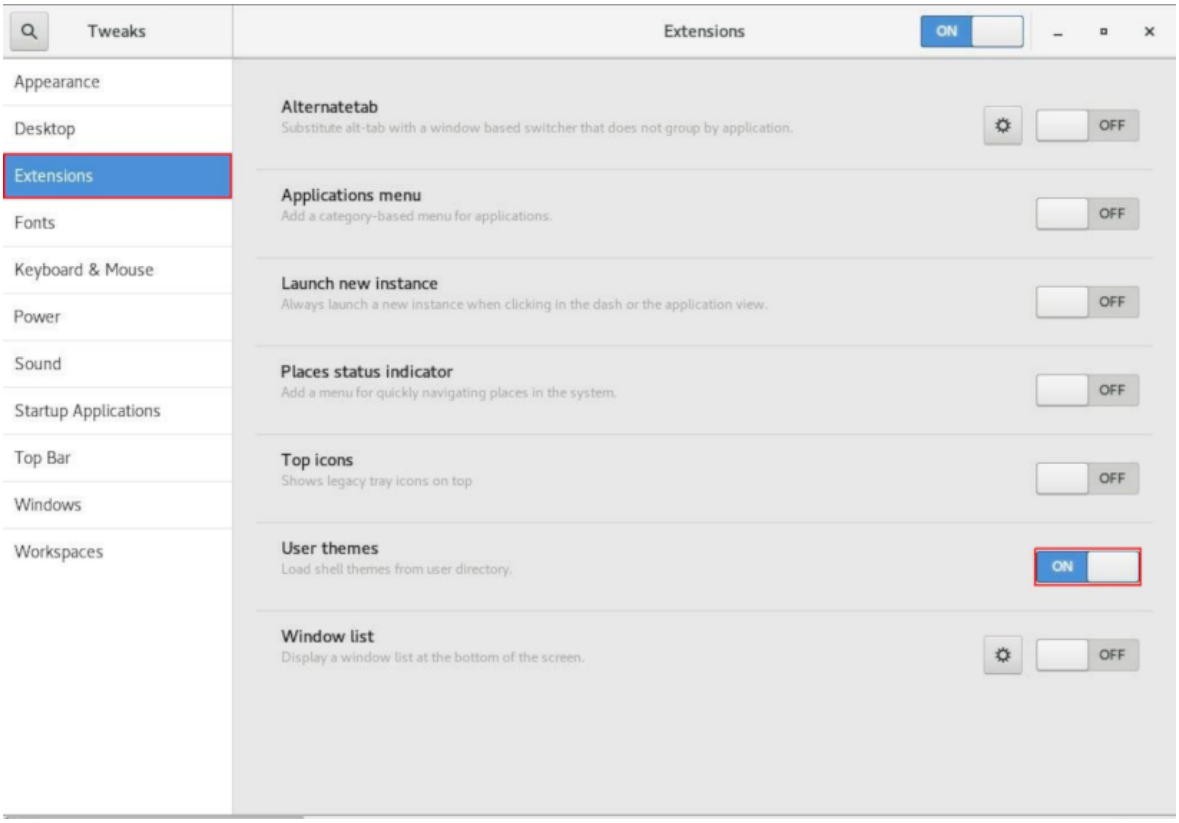
3.7. Applying an NVIDIA Look and Feel to the Desktop User Interface

If the GNOME Desktop is installed, you can optionally apply an NVIDIA look and feel to the desktop user interface by applying the NVIDIA theme to applications and the shell, and using NVIDIA images for the desktop background and lock screen.
The GNOME Desktop must already be installed and running on your system. If **SOFTWARE SELECTION** was set to **Server with GUI** when you installed Red Hat Enterprise Linux, the GNOME Desktop is already installed. If the GNOME Desktop is not installed, you must install the `X Window System` and `GNOME` package groups.

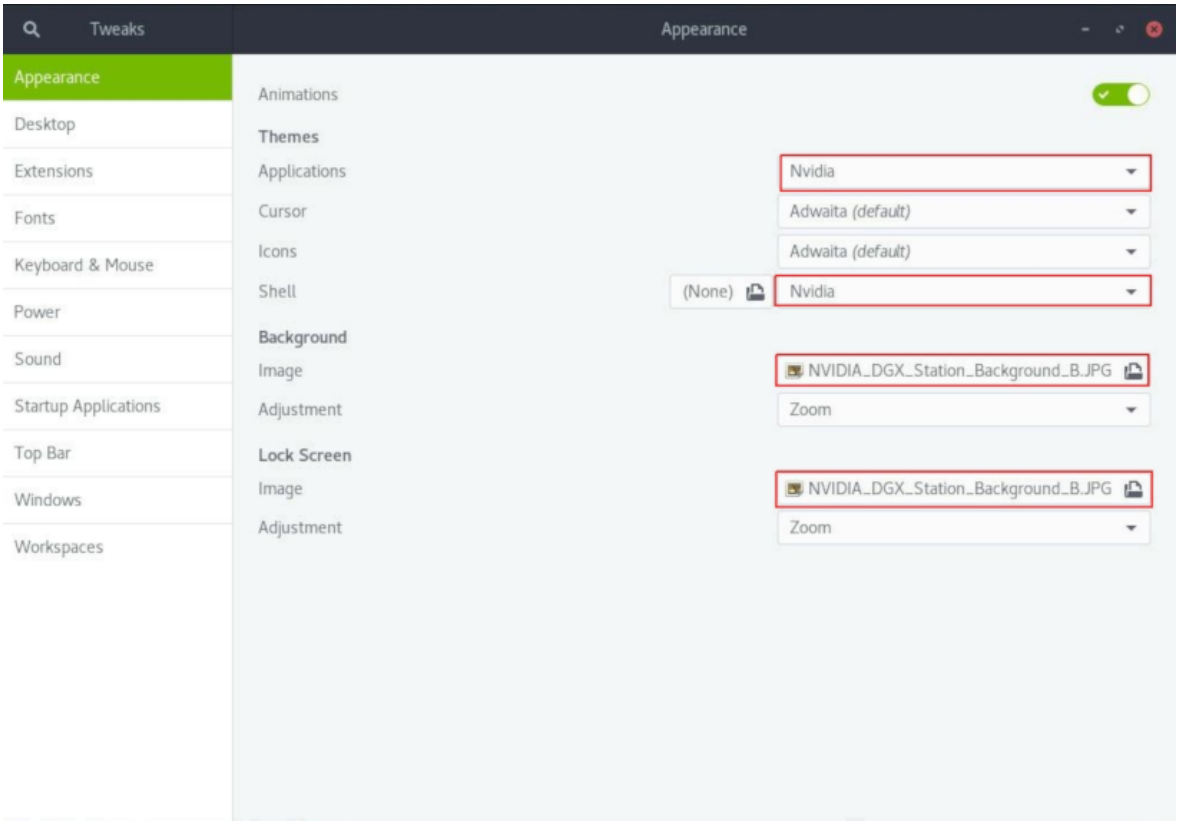
1. Install the `DGX Desktop Theme` package group.
   ```
   sudo yum groupinstall -y 'DGX Desktop Theme'
   ```
2. Start `gnome-tweaks`.
3. In the **Appearance** window that opens, under **Tweaks**, click **Extensions**.



4. In the **Extensions** window that opens, set **Extensions** in the title bar and **User themes** to **ON**.



5. Stop and restart `gnome-tweaks`.
6. In the **Appearance** window that opens, apply the NVIDIA theme to applications and the shell, and use NVIDIA images for the desktop background and lock screen.
   a). Under **Themes**, in the drop-down lists for **Applications** and **Shell**, click **Nvidia**.
   b). Under **Background** and **Lock Screen**, click the **Image** file selector.
   c). In the **Image** window that opens, select an NVIDIA DGX Station background image, for example, `NVIDIA_DGX_Station_Background_B.JPG`, and click **Open**.



3.8. Managing CPU Mitigations

DGX Software for Red Hat Enterprise Linux includes security updates to mitigate CPU speculative side-channel vulnerabilities. These mitigations can decrease the performance of deep learning and machine learning workloads.

If your installation of DGX systems incorporates other measures to mitigate these vulnerabilities, such as measures at the cluster level, you can disable the CPU mitigations for individual DGX nodes and thereby increase performance. This capability is available starting with DGX Software for Red Hat Enterprise Linuxsoftware version EL7-20.02.

3.8.1. Determining the CPU Mitigation State of the DGX System

If you do not know whether CPU mitigations are enabled or disabled, issue the following.

```
$ cat /sys/devices/system/cpu/vulnerabilities/*
```

- CPU mitigations are enabled if the output consists of multiple lines prefixed with `Mitigation:`.

  Example

```
KVM: Mitigation: Split huge pages
Mitigation: PTE Inversion; VMX: conditional cache flushes, SMT vulnerable
Mitigation: Clear CPU buffers; SMT vulnerable
Mitigation: PTI
Mitigation: Speculative Store Bypass disabled via prctl and seccomp
Mitigation: usercopy/swapgs barriers and __user pointer sanitization
Mitigation: Full generic retpoline, IBPB: conditional, IBRS_FW, STIBP:
 conditional, RSB filling
Mitigation: Clear CPU buffers; SMT vulnerable
```

- CPU mitigations are disabled if the output consists of multiple lines prefixed with `Vulnerable`.

Example

```
KVM: Vulnerable
Mitigation: PTE Inversion; VMX: vulnerable
Vulnerable; SMT vulnerable
Vulnerable
Vulnerable
Vulnerable: __user pointer sanitization and usercopy barriers only; no swapgs
 barriers
Vulnerable, IBPB: disabled, STIBP: disabled
Vulnerable
```

3.8.2. Disabling CPU Mitigations

CAUTION: Performing the following instructions will disable the CPU mitigations provided by the DGX Software for Red Hat Enterprise Linux.

1. Apply the `dgx*-no-mitigations` profile.
   - On a DGX-2 system, issue
     ```
     $ sudo tuned-adm profile dgx2-no-mitigations
     ```
   - On a DGX-1 system, issue
     ```
     $ sudo tuned-adm profile dgx-no-mitigations
     ```
   - On a DGX Station workstation, issue
     ```
     $ sudo tuned-adm profile dgxstation-no-mitigations
     ```
2. Reboot the system.
3. Verify CPU mitigations are disabled.
   ```
   $ cat /sys/devices/system/cpu/vulnerabilities/*
   ```
   The output should include several `Vulnerable` lines. See Determining the CPU Mitigation State of the DGX System for example output.

3.8.3. Re-enabling CPU Mitigations

1. Apply the `dgx*-performance` package.
   - On a DGX-2 system, issue
     ```
     $ sudo tuned-adm profile dgx2-performance
     ```
   - On a DGX-1 system, issue
     ```
     $ sudo tuned-adm profile dgx-performance
     ```
   - On a DGX Station workstation, issue

```
$ sudo tuned-adm profile dgxstation-performance
```

2. Reboot the system.
3. Verify CPU mitigations are enabled.

   ```
   $ cat /sys/devices/system/cpu/vulnerabilities/*
   ```

   The output should include several `Mitigations` lines. See Determining the CPU Mitigation State of the DGX System for example output.

Chapter 4. Running Containers

The following is an example of running the CUDA container from the NGC registry.

```
sudo docker run --security-opt label=type:nvidia_container_t --rm nvcr.io/nvidia/
cuda:10.0-base nvidia-smi
```

Note: To accommodate SELinux, the DGX software stack includes a package (`nvidia-container-selinux`) that defines a policy for allowing containers to access NVIDIA GPUs. The `--security-opt` option in the command sets the corresponding label type permitting the specified container to access NVIDIA GPUs. If SELinux is removed or disabled, then the `--security-opt` option is not needed.

Chapter 5. Configuring Storage - NFS Mount and Cache

The DGX servers and the DGX Station include multiple SSDs for data caching or data storage.

- The DGX-1 server contains four SSDs to be used in a RAID 0 configuration.
- The DGX-2 server contains 8 or 16 SSDs to be used in a RAID 0 configuration.
- The DGX A100 server contains 4 or 8 SSDs to be used in a RAID 0 configuration.
- The DGX Station contains three SSDs to be used in a RAID 0 or RAID 5 configuration.

For the DGX servers, these SSDs are intended to be used as a data cache for NFS mounted directories. For the DGX Station, these SSDs are intended to be used either for local persistent storage or as a data cache for NFS mounted directories. If you are using these SSDs as a data cache for NFS mounted directories, you must set up your own NFS mounted directories for long-term data storage.

The following instructions describe how to mount the NFS directories 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 yum 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.

Appendix A. Installing Software on Air-Gapped NVIDIA DGX Systems

When installing Red Hat Enterprise Linux on DGX systems, many of the packages that the DGX system requires are downloaded over the internet from both Red Hat and NVIDIA servers. This is not suitable for tightly secured systems which must be "Air-Gapped"; isolated from the internet or outside networks.

For air-gapped systems, Red Hat provides tools (createrepo and reposync) to make local mirrors of the internet available repositories. These repository mirrors can then be copied into the air-gapped environment and activated on the DGX system for installation.

This document assumes that you have created a mirror of the Red Hat repositories. For the NVIDIA repositories, it provides steps for mirroring NVIDIA repositories on a "low-side" system that has public internet access, and then copying and activating the repositories to an air-gapped DGX system. It also includes a process for installing Docker containers.

A.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.

A.2. Creating the Mirrors on the Low-Side Red Hat System

Perform these tasks on a system with network access (low-side Red Hat system), where

low-side-user
represents the username on the low-side system.

my-mirror
represents the folder where the mirrored files are create or stored.

1. Set up a directory for creating the mirror.

```
mkdir /home/<low-side-user>/<my-mirror>
```

2. Mirror the `nvidia-dgx-7` repo.
   a). `cd /home/<low-side-user>/<my-mirror>`
   b). `sudo reposync --newest-only --gpgcheck -l --repoid nvidia-dgx-7 --download_path=/home/<low-side-user>/<my-mirror> --downloadcomps --download-metadata`
   c). `cd nvidia-dgx-7/`
   d). `sudo createrepo -v /home/<low-side-user>/<my-mirror>/nvidia-dgx-7 -g comps.xml`
   e). `cp /etc/yum.repos.d/nvidia-dgx-7.repo /home/<low-side-user>/<my-mirror>/nvidia-dgx-7/`
   f). `cp /etc/pki/rpm-gpg/RPM-GPG-KEY-dgx-cosmos-support /home/<low-side-user>/<my-mirror>/nvidia-dgx-7/`
3. Mirror the `nvidia-dgx-7-updates` repo
   a). `cd /home/<low-side-user>/<my-mirror>`
   b). `sudo reposync --newest-only --gpgcheck -l --repoid nvidia-dgx-7-updates --download_path=path/to/mirrored/files --downloadcomps --download-metadata`
   c). `sudo createrepo -v /home/<low-side-user>/<my-mirror>/nvidia-dgx-7-updates`
4. Mirror the `rhel-7-server-extras-rpms` repo.
   a). `cd /home/<low-side-user>/<my-mirror>`
   b). `sudo reposync --newest-only --gpgcheck -l --repoid rhel-7-server-extras-rpms --download_path=/home/<low-side-user>/<my-mirror> --downloadcomps --download-metadata`
   c). `cd rhel-7-server-extras-rpms/`
   d). `sudo createrepo -v /home/<low-side-user>/<my-mirror>/rhel-7-server-extras-rpms -g comps.xml`
   e). `sudo cp /etc/yum.repos.d/redhat_repo /home/<low-side-user>/<my-mirror>/rhel-7-server-extras-rpms/`
   f). `sudo cp /etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release /home/<low-side-user>/<my-mirror>/rhel-7-server-extras-rpms`
5. Mirror the `rhel-7-server-optional-rpms` repo
   a). `cd /home/<low-side-user>/<my-mirror>`
   b). `sudo reposync --newest-only --gpgcheck -l --repoid rhel-7-server-optional-rpms --download_path=/home/<low-side-user>/<my-mirror> --downloadcomps --download-metadata`
   c). `cd rhel-7-server-optional-rpms/`
   d). `sudo createrepo -v /home/<low-side-user>/<my-mirror>/rhel-7-server-optional-rpms -g comps.xml`
6. Mirror the `rhel-7-server-rpms` repo
   a). `cd /home/<low-side-user>/<my-mirror>`

b). `sudo reposync --newest-only --gpgcheck -l --repoid rhel-7-server-rpms --download_path=/home/<low-side-user>/<my-mirror> --downloadcomps --download-metadata`

c). `cd rhel-7-server-rpms/`

d). `sudo createrepo -v /home/<low-side-user>/<my-mirror>/rhel-7-server-rpms -g comps.xml`

7. Mirror the `rhel-server-rhscl-7-rpms` repo.

a). `cd /home/<low-side-user>/<my-mirror>`

b). `sudo reposync --newest-only --gpgcheck -l --repoid rhel-server-rhscl-7-rpms --download_path=/home/<low-side-user>/<my-mirror> --downloadcomps --download-metadata`

c). `cd rhel-server-rhscl-7-rpms/`

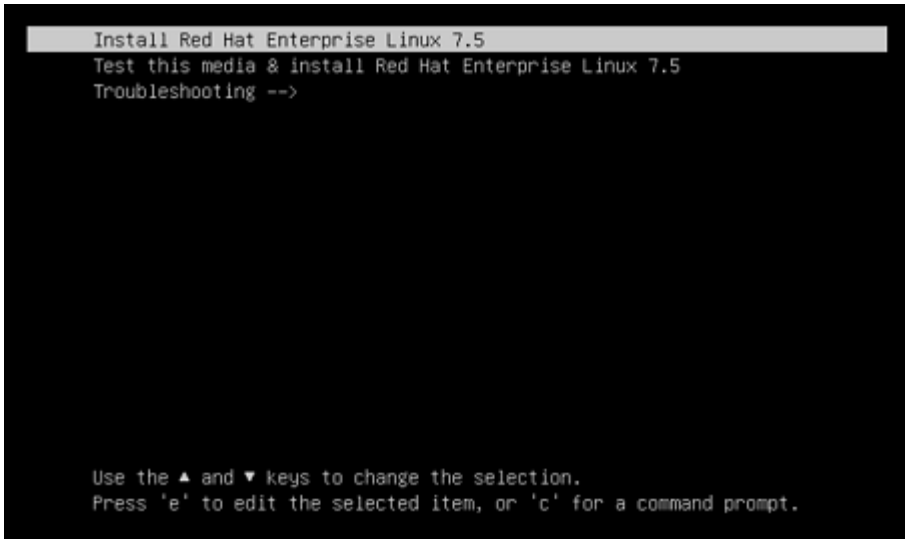
d). `sudo createrepo -v /home/<low-side-user>/<my-mirror>/rhel-server-rhscl-7-rpms -g comps.xml`

8. Create an archive of the mirrored repos.

```
tar cf <mirror-archive-name>.tar <username>
```

A.3. Installing Red Hat Enterprise Linux on the Air-Gapped DGX-2/DGX A100

1. After booting the ISO image through either the BMC or from the USB drive, select **Install Red Hat Enterprise Linux** and then press **Enter** to start the installation.



2. Refer to the Red Hat Enterprise Linux Quick Installation Guide (https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/installation_guide/chap-simple-install) for guidance on using the installer.

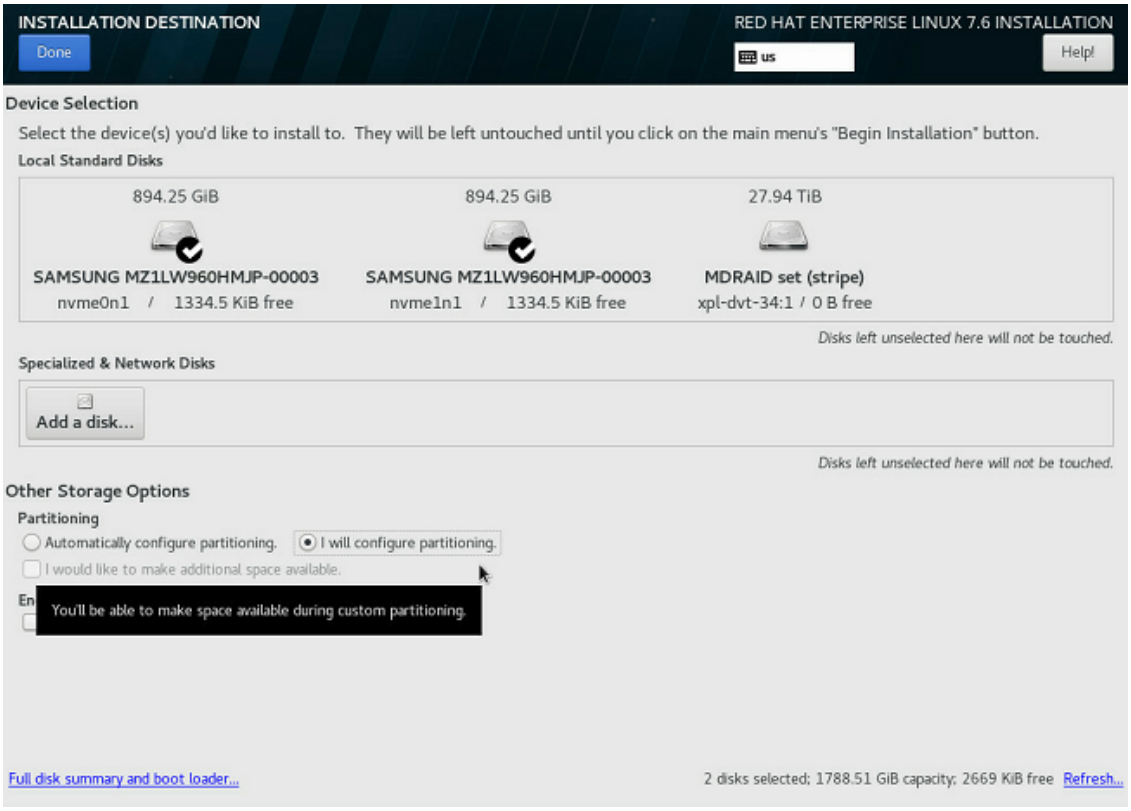
   Configure the language, region, date, time, keyboard, and other configuration options you may need from the **Installation Summary** screen.

3. Set up the system drives.
   a). Remove all partitions.

   At the Installation Destination screen, select both M.2 (894.25 GB on DGX-2 or 1788 GB on DGX A100) disks, the "**Automatically configure partitioning**" radio button and the **I would like to make additional space available** check box, then click **Done**.

   b). At the **Reclaim Disk Space** screen, click **Reclaim Space**.
4. Create EFI partitions.
   a). From the Installation Summary screen, click **INSTALLATION DESTINATION** to create new paritions.

   b). At the Installation Destination screen, select both M.2 (894.25 GB on DGX-2 or 1788 GB on DGX A100) disks and the "**I will configure partitioning**" radio button, then click **Done**.



The **Manual Partitioning** window appears.

c). Create a new partition.

At the Manual Paritioning screen, use the **Standard Partition** and then click "+" .

d). Add the mount point for the first EFI partition.

Set the Mount Point to `/boot/efi` and the Desired Capacity to 512 MB, then click **Add mount point**.

e). Verify that the installer selects drive nvme0n1p1 (DGX-2) or nvme3n1p1 (DGX A100).

If the correct drive is not designated, then click **Modify** and then select the correct drive from the **Configure Mount Point** dialog box and click **Select**.

f). Click "+" to create the alternate EFI boot partition.

g). Add the mount point for the alternate EFI partition.

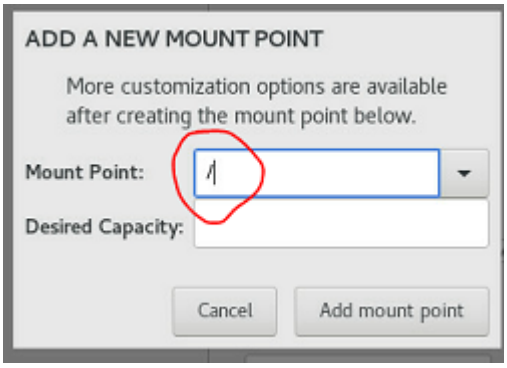
Set the Mount Point to `/boot/efi2` and the Desired Capacity to 512 MB, then click **Add mount point**

5. Create the RAID 1 partition.

a). Click "+" to add a new partition.

b). Add the mount point for the RAID partition.

Set the Mount Point to `/` and leave the Desired Capacity empty, then click **Add mount point.**



c). Select the new system partition and set the following.

Device Type
RAID
RAID-Level
RAID1
Name
`md0/`

6. Verify the following assignments:
   - DGX-2: /boot/efi is assigned drive `nvme0n1p1` and /boot/efi2 is assigned drive `nvme1n1p1`
   - DGX A100: /boot/efi is assigned drive `nvme3n1p1` and /boot/efi2 is assigned drive `nvme2n1p1`

   Occasionally, the partition names will change. If that happens, select the desired drive and click **Modify->Update settings** to correct.
7. Click **Done** to exit and commit the new partition scheme, then click **Accept Changes**.
8. Continue following the steps starting with -
   - DGX-2: Step 5 of Installing on the DGX-2.
   - DGX A100: Step 5 of Installing on the DGX A100

A.4. Installing DGX Software on the Air-Gapped DGX-2/DGX A100

Perform these tasks on the air-gapped DGX-2 or DGX A100 system, where

dgx-user
represents the DGX user

my-mirror
represents the folder where the mirrored files are create or stored

1. After the system has rebooted, log in and verify the partitioning scheme & correct EFI mount;

NVSM requires the RAID-1 volume to be named `md0`, and the EFI boot partition to be correctly ordered.

a). Verify that the partition scheme matches the output shown for the following commands:

```
$ ls -ldh /dev/md*

drwxr-xr-x. 2 root root 80 Aug 9 17:26 /dev/md
brw-rw----. 1 root disk 9, 0 Aug 9 17:26 /dev/md0
brw-rw----. 1 root disk 9, 127 Aug 9 17:26 /dev/md127
```

```
$ sudo mdadm --detail /dev/md0 | grep nvme

0 259 3 0 active sync /dev/nvme0n1p2
1 259 5 1 active sync /dev/nvme1n1p2
```

```
$ sudo lsblk -f | grep efi

├─nvme0n1p1 vfat /boot/efi
├─nvme1n1p1 vfat /boot/efi2
```

If the RAID-1 volume is not named `md0`, see the Appendix: Renaming RAID Volumes for instructions on how to rename the RAID volume.

b). Edit `/etc/fstab` and remove the line to mount `/boot/efi2`.

2. Copy over and enable the repo mirror on the DGX-2 system.

a). Switch to the directory to place the mirrored repo.

```
cd /home/<dgx-user>
```

b). Secure-copy the mirrored archive from the low-side system.

```
scp <dgx-user>@<low-side-ip-address>:home/<dgx-user>/<mirror-archive-name>.tar .
```

c). Extract the mirrored archive.

```
sudo tar xf <mirror-archive-name>.tar
```

d). Copy the `rhel-7-server-extras` and `nvidia-dgx-7` repos to the yum.repos.d.

```
sudo cp ./<my-mirror>/rhel-7-server-extras-rpms/redhat.repo /etc/yum.repos.d/redhat-mirror.repo
```

```
sudo cp ./<my-mirror>/nvidia-dgx-7/nvidia-dgx-7.repo /etc/yum.repos.d/
```

e). Edit `/etc/yum.repos.d/nvidia-dgx-7.repo`.

Change the `nvidia-dgx-7` repo

From:

```
baseurl=https://international.download.nvidia.com/dgx/repos/rhel7/
```

To:

```
baseurl=file:///home/<dgx-user>/<my-mirror>/nvidia-dgx-7
```

Change `nvidia-dgx-7-updates` repo

From:

```
baseurl=https://international.download.nvidia.com/dgx/repos/rhel7-updates/
```

To:

```
baseurl=file:///home/<dgx-user>/<my-mirror>/nvidia-dgx-7-updates
```

Change `enabled=0` to `enabled=1`

f). For each of the following repos ($reponame):

- rhel-7-server-extras-rpms

- rhel-7-server-optional-rpms
- rhel-7-server-rpms
- rhel-server-rhscl-7-rpms

Edit `/etc/yum.repos.d/redhat-mirror.repo`, find the `$reponame` section (listed in brackets), and change the text in that section:

From:

```
baseurl=https://cdn.redhat.com/content/dist/rhel/server/7/...
```

To:

```
baseurl=file:///home/<dgs-user>/<my-mirror>/$reponame
```

Set `enabled = 1`.

g). Copy the GPG key file to your system.

```
sudo cp <my-mirror>/nvidia-dgx-7/RPM-GPG-KEY-dgx-cosmos-support /etc/pki/rpm-gpg
```

h). List the repositories and verify that all 11 repositories are enabled.

```
sudo yum repolist
```

The output should look like the following:

```
Loaded plugins: product-id, search-disabled-repos, subscription-manager
This system is not registered with an entitlement server. You can use
 subscription-manager to register.
Repo rhel-7-server-extras-rpms forced skip_if_unavailable=True due to:
/etc/pki/entitlement/9124846071548788754.pem
Repo rhel-7-server-extras-rpms forced skip_if_unavailable=True due to:
/etc/pki/entitlement/9124846071548788754-key.pem
Repo rhel-7-server-rpms forced skip_if_unavailable=True due to:
/etc/pki/entitlement/9124846071548788754.pem
Repo rhel-7-server-rpms forced skip_if_unavailable=True due to:
/etc/pki/entitlement/9124846071548788754-key.pem
Repo rhel-7-server-optional-rpms forced skip_if_unavailable=True due to:
/etc/pki/entitlement/9124846071548788754.pem
Repo rhel-7-server-optional-rpms forced skip_if_unavailable=True due to:
/etc/pki/entitlement/9124846071548788754-key.pem
Repo rhel-server-rhscl-7-rpms forced skip_if_unavailable=True due to:
/etc/pki/entitlement/3657713380925279193.pem
Repo rhel-server-rhscl-7-rpms forced skip_if_unavailable=True due to:
/etc/pki/entitlement/3657713380925279193-key.pem
nvidia-dgx-7                                     | 3.6 kB 00:00
nvidia-dgx-7-updates                             | 2.9 kB 00:00
rhel-7-server-extras-rpms                        | 3.6 kB 00:00
rhel-7-server-optional-rpms                      | 3.6 kB 00:00
rhel-7-server-rpms                               | 3.6 kB 00:00
rhel-server-rhscl-7-rpms                         | 3.6 kB 00:00
(1/11): nvidia-dgx-7/group_gz                    | 644 B 00:00
(2/11): rhel-7-server-optional-rpms/group_gz     | 6.4 kB 00:00
(3/11): nvidia-dgx-7/primary_db                  | 70 kB 00:00
(4/11): rhel-7-server-optional-rpms/primary_db   | 2.1 MB 00:00
(5/11): rhel-7-server-rpms/group_gz              | 150 kB 00:00
(6/11): nvidia-dgx-7-updates/primary_db          | 56 kB 00:00
(7/11): rhel-7-server-rpms/primary_db            | 4.4 MB 00:00
(8/11): rhel-7-server-extras-rpms/group_gz       | 133 B 00:00
(9/11): rhel-7-server-extras-rpms/primary_db     | 85 kB 00:00
(10/11): rhel-server-rhscl-7-rpms/group_gz       | 133 B 00:00
(11/11): rhel-server-rhscl-7-rpms/primary_db     | 3.0 MB 00:00
repo id                    repo name
 status
nvidia-dgx-7               NVIDIA DGX EL7
 146
```

```
nvidia-dgx-7-updates         NVIDIA DGX EL7 Updates
 101
rhel-7-server-extras-rpms    Red Hat Enterprise Linux 7 Server - Extras (R
 151
rhel-7-server-optional-rpms Red Hat Enterprise Linux 7 Server - Optional
 5,190
rhel-7-server-rpms           Red Hat Enterprise Linux 7 Server (RPMs)
 5,457
rhel-server-rhscl-7-rpms     Red Hat Software Collections RPMs for Red Hat
 7,123
repolist: 18,168
```

3. Continue following the installation steps beginning with Installing Required Components.

A.5. Renaming RAID Volumes

The RAID volume should be `/dev/md0`. If, during the installation process the volume is incorrectly named `/dev/md/root`, then rename the volume as follows.

1. Convert `/dev/md/root` to `/dev/md0`.
   a). Edit `/dev/mdadm.conf`.
   Example: `sudo vi /dev/mdadm.conf`
   b). Replace `/dev/md/root` with `/dev/md0`.
2. Save the file and then reboot.

A.6. 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 Registry for DGX 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
   ```

Appendix B. Installing Software Using Local Repositories

As an alternative to the reposync method described in Appendix A, software can also be installed using local repos. These repos provided by NVIDIA will install all of the available packages locally in the filesystem, allowing installation of packages without network access

B.1. Installing the Base Red Hat Enteprise Linux System

The Red Hat components of the software must be installed either using the standard network-based installation, or as described in Appendix A. 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 installation, then enable the additional Red Hat repositories. Finally, execute:

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

If the system is not connected to the network, follow the instructions in Appendix A to mirror the Red Hat repositories using `reposync`.

B.2. Downloading the Local Repositories

Download the DGX Software local repository from http://repo.download.nvidia.com/baseos/el/el-files/7/dgx-local-repo-21.04-1.el7.x86_64.rpm

B.3. Installing the Local Repositories

Install the local repositories

```
sudo yum install -y ./dgx-local-repo-20.11-89.el8.x86_64.rpm
```

With the local repositories installed, proceed with the standard "Installing the DGX Software" instructions.

Appendix C. Changing the BMC Login

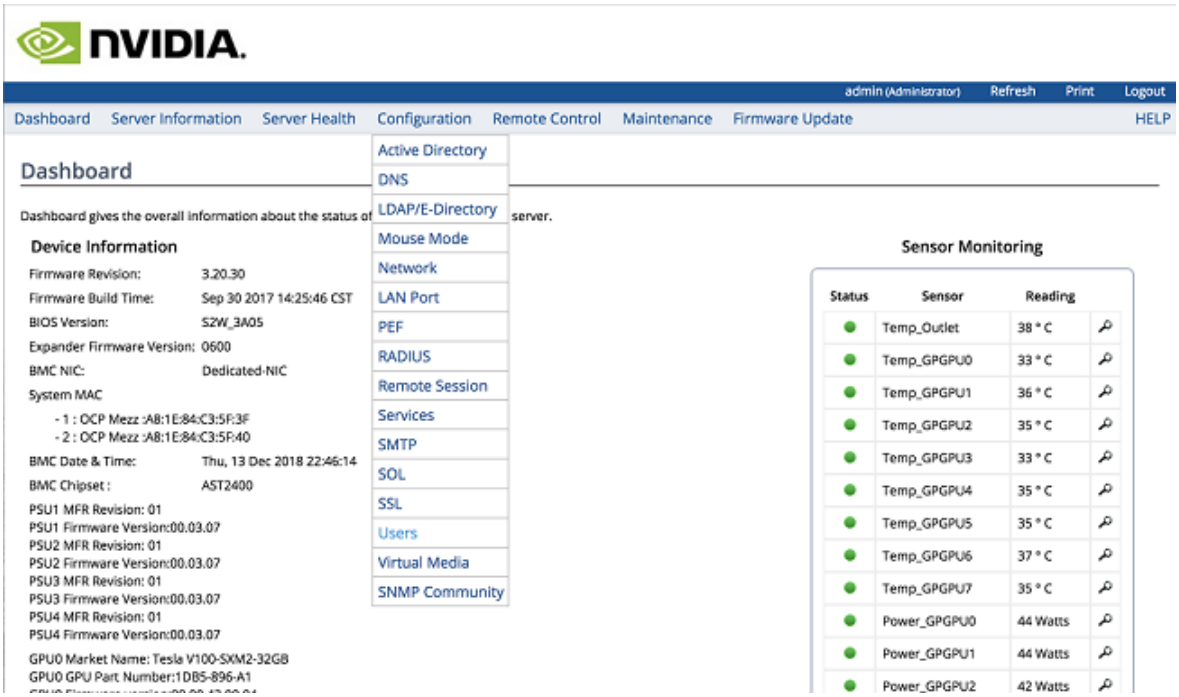
The NVIDIA DGX server includes a base management controller (BMC) for out-of-band management of the DGX system. NVIDIA recommends disabling the default username and creating a unique username and password as soon as possible.

C.1. Changing the BMC Login on the DGX-1

1. Log into the BMC.
   a). Open a browser within your LAN and go to `http://<BMC-ip-address>/`.

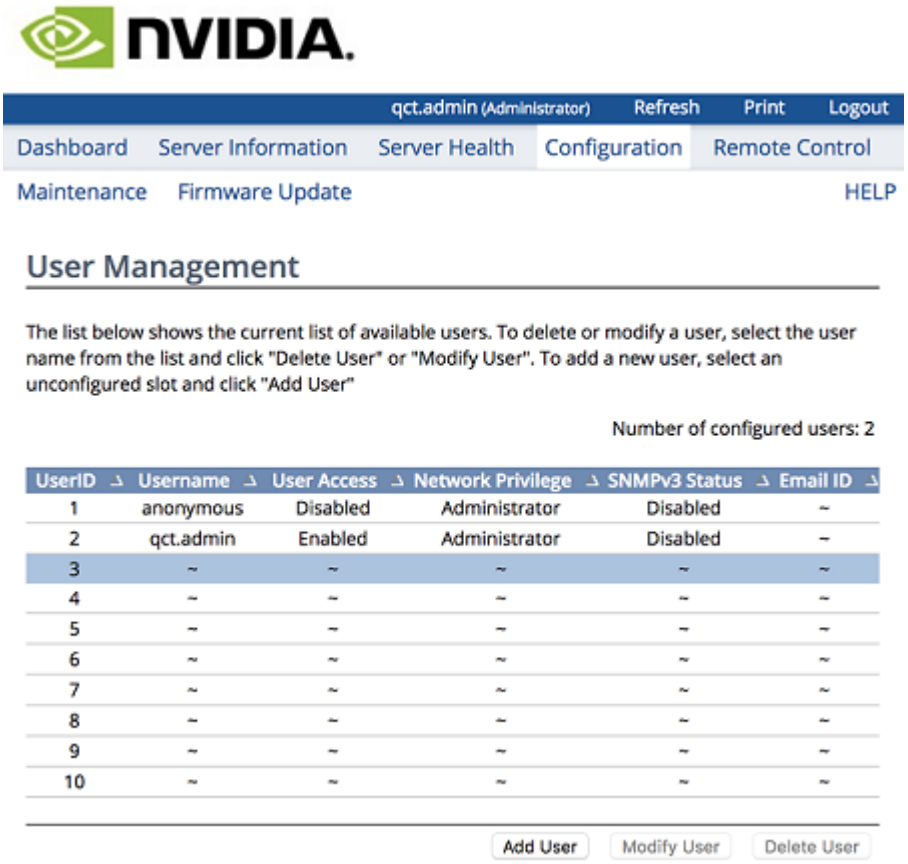
   Use Firefox or Internet Explorer. Google Chrome is not officially supported by the DGX-1 BMC.

   b). Log in, using qct.admin/qct.admin for the User ID/Password.
2. Select Configuration → Users.

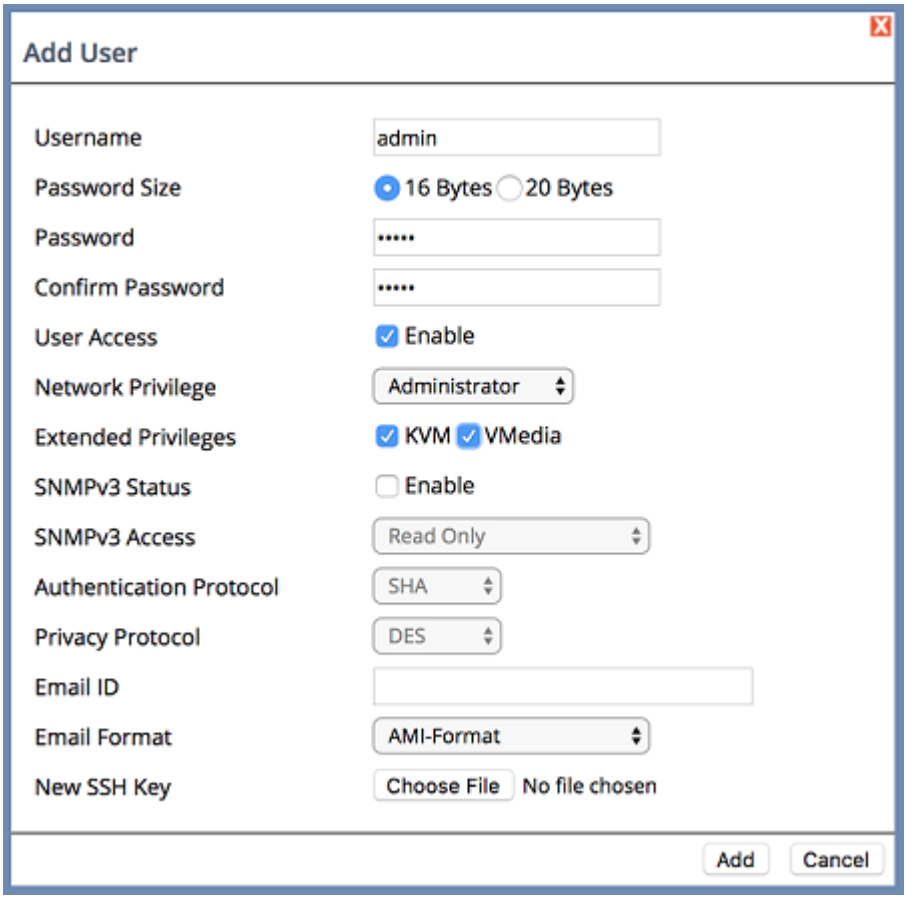


3. Add a new user.

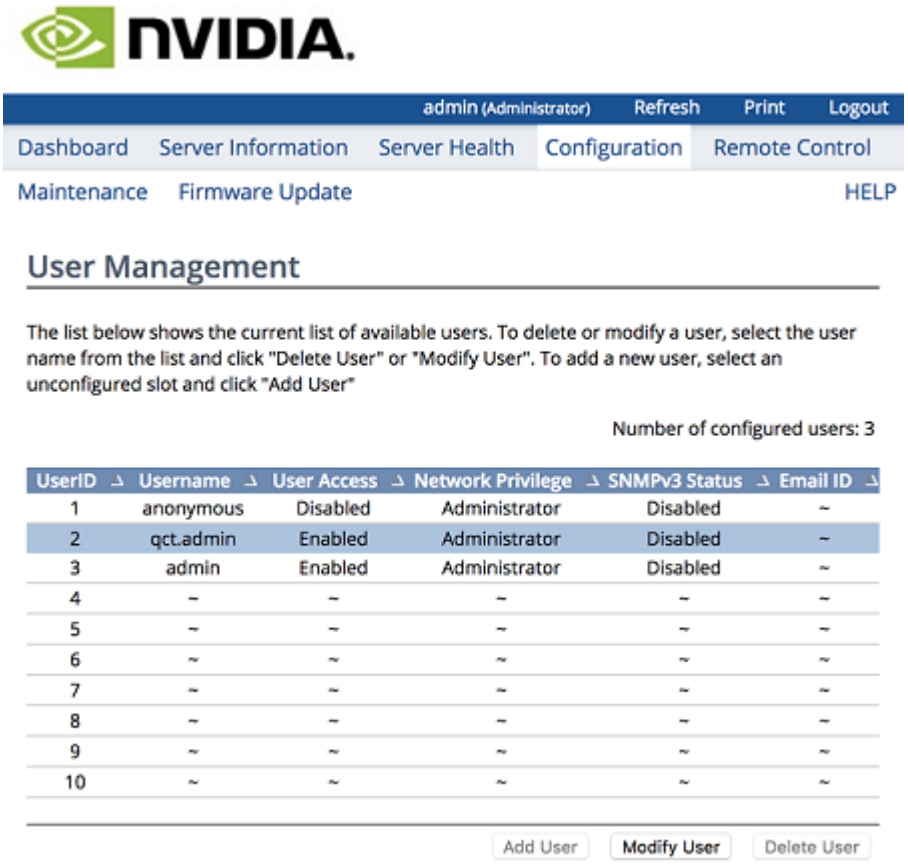
   a). Select an empty field and click Add User.



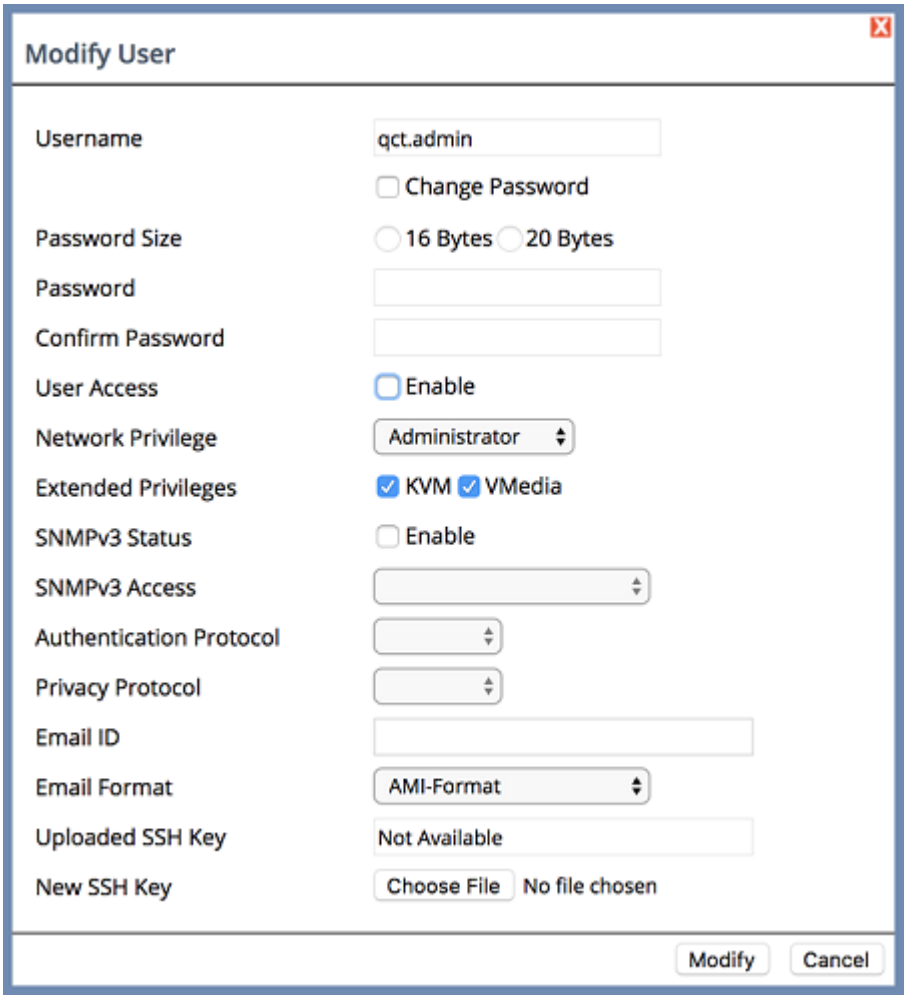
   b). Enter new user information and click Add.



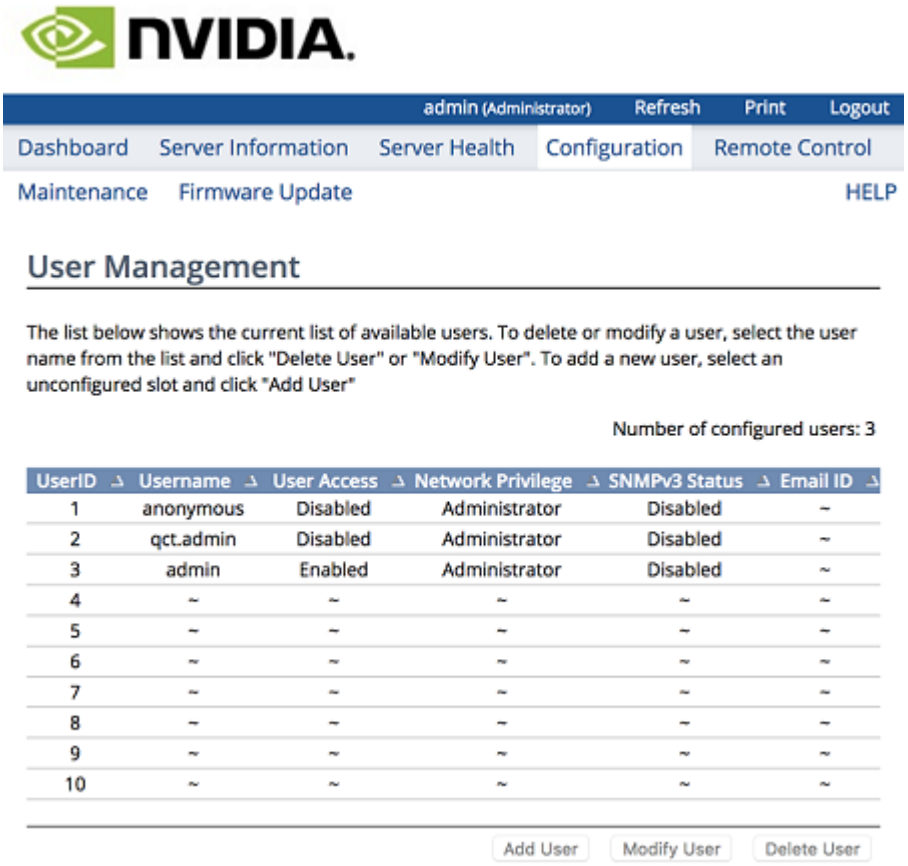
4. Log out and then log back in as the new user.
5. Select Configuration → Users.
6. Disable User Access for the user qct.admin.
   a). Select the user qct.admin user and select Modify User



b). Deselect Enable in User Access and click Modify.



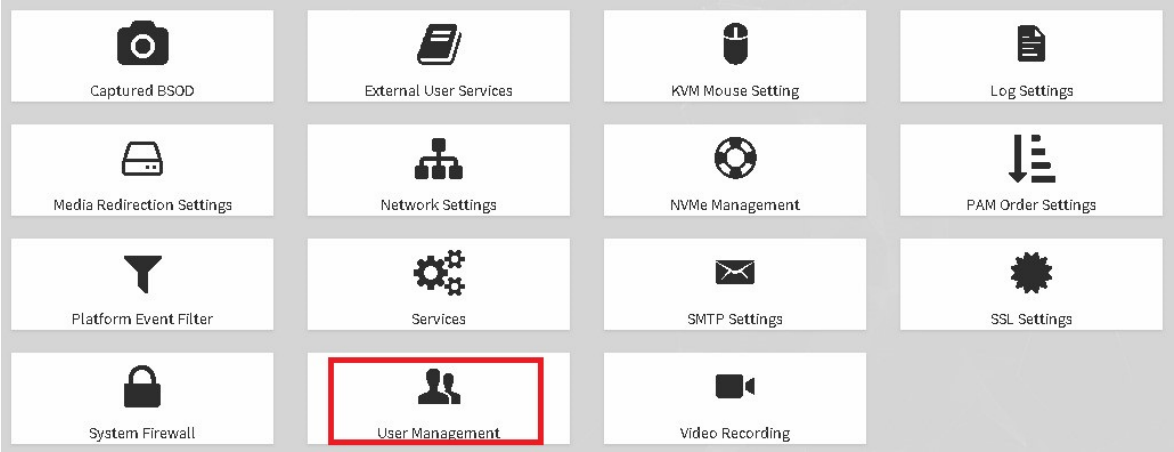
c). Ensure User Access is disabled for the user qct.admin.



7. Log out.

C.2. Changing the BMC Login on the DGX-2 or DGX A100

1. Log into the BMC.
   a). Open a browser within your LAN and go to `https://<BMC-ip-address>/`.
   b). Log in, using admin/admin for the User ID/Password.
2. Select **Settings** from the left-side navigation menu.
3. Select the **User Management** card.



4. Click the green Help icon (?) for information about configuring users, then add a new user with unique username and strong password.
5. Log out and then log back in as the new user.
6. Select **Settings** → **User Management**.
7. Disable the **admin** and **anonymous** users.

Appendix D. Installing and Mellanox InfiniBand Drivers

Unlike the DGX OS shipped with the NVIDIA DGX server, the DGX software stack for Red Hat-derived operating systems does not include the Mellanox OpenFabrics Enterprise Distribution (MLNX_OFED) for Linux. This is to avoid an installation where the MLNX_OFED kernel may be out of sync with the Red Hat distribution kernel, resulting in system instability.

To use InfiniBand on the DGX server, do the following.

1. Determine which MLNX_OFED package supports the latest version of the installed Red Hat Enterprise Linux release.
   a). Visit https://access.redhat.com/articles/3078 and determine the latest Red Hat Enterprise Linux 7 version.

   The `yum update` command that is run prior to installing the NVIDIA driver updates to the latest Red Hat Enterprise Linux 7 version.

   b). Visit https://docs.mellanox.com/category/mlnxofedib, click the latest MLNX_OFED software version and then use the side menu to navigate to Release Notes->General Support in MLNX_OFED and view Supported Operating Systems to determine the MLNX_OFED package OS support.

   Select a MLNX_OFED package only if it contains Mellanox Firmware Tools (MFT) 4.14.4-6 or higher and if it is for the Red Hat Enterprise Linux 7 version running on the DGX system.
2. Visit the Mellanox site and download and install the appropriate MLNX_OFED driver.

   Install a MLNX_OFED driver only if it contains Mellanox Firmware Tools (MFT) 4.14.4-6 or higher, and if it is for the Red Hat Enterprise Linux 7 version running on the DGX server.
3. Install `nvidia-mlnx-config`.

   ```
   sudo yum install nvidia-mlnx-config
   ```
4. After installing the MLNX_OFED drivers, install the NVIDIA peer memory module.

   ```
   sudo yum install nvidia-peer-memory-dkms
   ```

Start `nv_peer_mem`, either

- Manually, by issuing `sudo systemctl start nv_peer_mem`, or
- Set up the system to start it automatically on every system boot.

1. Create a file `/etc/modules-load.d/nv-peer-mem.conf` with contents "`nv_peer_mem`".
2. Issue `dracut --force /boot/initramfs-$(uname -r).img $(uname -r)`
3. Reboot the system.

Note: While in-box drivers may 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™ RDMA feature. For more information on configuring the in-box drivers, see the following Red Hat Enterprise Linux documentation:

- Configuring InfiniBand and RDMA Networks, and
- InfiniBand and RDMA Related Software Packages

Appendix E. Using Custom DGX Software Utilities for the DGX Station

The DGX Software includes custom utilities for maintaining the DGX Station persistent storage. Custom utilities for managing and obtaining diagnostic information for the DGX Station were included **only** in version EL7-20.01 of the DGX Software.

E.1. Rebuilding or Re-Creating the DGX Station RAID Array

Failure of a single drive in a RAID 5 array is a recoverable error but the failure causes data redundancy for the array to be lost. After replacing a single failed SSD in a RAID 5 array, you must rebuild the array to restore data redundancy for the array. Failure of any number of SSDs in a RAID 0 array and failure of more than one SSD in a RAID 5 array are both unrecoverable failures. After replacing the SSDs in response to an unrecoverable failure, you must re-create the array.
If the DGX Station RAID array is degraded because one or more SSDs failed, replace each failed SSD as explained in DGX Station User Guide.

The DGX Station software includes the custom script `configure_raid_array.py` for rebuilding or re-creating the RAID array.

- To rebuild a RAID 5 array after replacing a single failed SSD, run the following command:

```
$ sudo configure_raid_array.py -r
```

Note: The time required to rebuild a RAID 5 array depends on factors such as system load, SSD capacity, and the number of SSDs in the array. Rebuilding the array of three 1.92-terabyte SSDs in the DGX Station may require several hours.

You can monitor the progress of a long-running rebuild by examining the contents of the `/proc/mdstat` file:

```
$ cat /proc/mdstat
Personalities : [raid0] [linear] [multipath] [raid1] [raid6] [raid5] [raid4]
 [raid10]
```

```
md0 : active raid5 sdb[0] sdd[3] sdc[1]
      3750486016 blocks super 1.2 level 5, 512k chunk, algorithm 2 [3/2] [UU_]
      [>.....................]  recovery =  4.0% (75580956/1875243008)
 finish=438.3min speed=68419K/sec
      bitmap: 2/14 pages [8KB], 65536KB chunk

unused devices: <none>
```

In this example, the rebuild is 4.0% complete and the rebuild is estimated to finish in 438.3 minutes.

- To re-create a RAID 5 array after replacing more than one failed SSD, run the following command:

```
$ sudo configure_raid_array.py -c -5 -f
```

> **CAUTION:** Specify the `-c` option only if an unrecoverable failure, such as the failure of more than one SSD, has occurred. The `-c` option erases all data in the array.

- To re-create a RAID 0 array after replacing any number of failed SSDs, run the following command:

```
$ sudo configure_raid_array.py -c -f
```

The RAID array is rebuilt or re-created with the RAID level that you specified.

- If you re-created a RAID 0 or RAID 5 array, all data that was on the array is erased after array is re-created.
- If you rebuilt a RAID 5 array, the data on the array is preserved after array is rebuilt.

If you have re-created a RAID 0 or RAID 5 array and have a backup of data on the array that you want to preserve, restore the data from the backup.

E.2. Changing the RAID Level of the RAID Array

During the initial installation of the DGX software on Red Hat Enterprise Linux, the data SSDs in the DGX Station are configured as a RAID 0 or RAID 5 array. If your requirements for redundancy or storage capacity change, you can change the RAID level of the array from the level that was initially configured.

Before changing the RAID level of the DGX Station RAID array, back up all data on the array that you want to preserve. Changing the RAID level of the DGX Station RAID array erases all data stored on the array.

The DGX Station software includes the custom script `configure_raid_array.py`, which you can use to change the level of the RAID array without unmounting the RAID volume.

- To change the RAID level to RAID 5, run the following command:

```
$ sudo configure_raid_array.py -m raid5
```

Note:

After you change the RAID level to RAID 5, the RAID array is rebuilt. A RAID array that is being rebuilt is online and ready to be used, but a check on the health of the DGX Station reports the status of the RAID volume as unhealthy. Therefore, avoid checking the health of the DGX Station while the RAID array is being rebuilt. For more information, see EL7-20.01 Only: Checking the Health of the DGX Station.

The time required to rebuild the RAID array depends on the workload on the system. On an idle system, the rebuild might be complete within 30 minutes.

- To change the RAID level to RAID 0, run the following command:

```
$ sudo configure_raid_array.py -m raid0
```

To confirm that the RAID level was changed as required, run the `lsblk` command. The entry in the `TYPE` column for each SSD in the RAID array indicates the RAID level of the array.

The following example shows that the RAID level of the array is RAID 0. The name of the RAID volume is `md0` and the mount point of the volume is `/raid`.

```
~$ lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
sda      8:0    0  1.8T  0 disk
|_sda1   8:1    0  487M  0 part  /boot/efi
|_sda2   8:2    0  1.8T  0 part  /
sdb      8:16   0  1.8T  0 disk
|_md0    9:0    0  5.2T  0 raid0 /raid
sdc      8:32   0  1.8T  0 disk
|_md0    9:0    0  5.2T  0 raid0 /raid
sdd      8:48   0  1.8T  0 disk
|_md0    9:0    0  5.2T  0 raid0 /raid
```

E.3. EL7-20.01 Only: Checking the Health of the DGX Station

Note: Starting with release EL7-20.02, the NVIDIA System Health Checker (`nvhealth`) tool is replaced by NVIDIA System Management (NVSM). For information about how to use NVSM to perfrom this task, see Show Health in NVIDIA System Management User Guide.

The DGX Station provides the NVIDIA System Health Checker (`nvhealth`) tool to exercise the system and verify its health. The output of `nvhealth` is an itemized list of checks and their status, typically Healthy or Unhealthy. On a healthy system, all checks should return Healthy. You should investigate any checks that return Unhealthy to determine their root cause and resolve them.

To check the health of the DGX Station, run the following command:

```
$ sudo nvhealth [-k output-file]
```

output-file

The name and the path of the file in which the raw state of the system is written. The `nvhealth` command displays this file name at the end of the output from the command.

If you omit the output file, the information is written to the file `/tmp/nvhealth-log.`*`random-string`*`.jsonl`, for example, `/tmp/nvhealth-log.6wf3WriAC3.jsonl`.

Note:

If you run the `nvhealth` command while the RAID array is being rebuilt after a change in RAID level to RAID 5, `nvhealth` reports the status of the RAID volume as unhealthy. To avoid this potentially misleading result, wait until RAID array is rebuilt before running `nvhealth`.

To check the progress of the rebuild and show the percentage complete and an estimate of the time to completion, run this command:

```
# cat /proc/mdstat

Personalities : [raid6] [raid5] [raid4] [linear] [multipath] [raid0] [raid1]
 [raid10]
md0 : active raid5 sdb[0] sdc[1] sdd[2]
     181764096 blocks super 1.2 level 5, 512k chunk, algorithm 2 [4/3] [UUU_]
     [===>.................]  recovery = 17.2% (10426232/60588032)
 finish=45.8min speed=18238K/sec
```

E.4. EL7-20.01 Only: Collecting Information for Troubleshooting the DGX Station

Note: Starting with release EL7-20.02, the tool to collect troubleshooting information (`nvsysinfo`) tool is replaced by NVIDIA System Management (NVSM). For information about how to use NVSM to perfrom this task, see Dump Health in NVIDIA System Management User Guide.

To help diagnose and resolve issues, the DGX Station provides a tool to collect troubleshooting information for NVIDIA Support Enterprise Services.

The tool verifies basic functionality and performance of the DGX Station and collects the following information in an xz-compressed tar archive:

- Log files
- Hardware inventory
- SW inventory

To collect information for troubleshooting the DGX Station, run the following command:

```
sudo nvsysinfo [-o output-file]
```

output-file

The path of the file in which the information is written.

If you omit the output file, the name of the file to which the information is written is `/tmp/nvsysinfo-`*`host-name`*`-`*`timestamp`*`.tar.xz`.

Use any method that is convenient for you to send the file to NVIDIA Support Enterprise Services. For example, send the file as an e-mail attachment.

Appendix F. Expanding the DGX Station RAID Array

After adding SSDs to the DGX Station, you must expand the RAID array to add the new SSDs to the array. The procedure for expanding the RAID array is the same for all supported RAID levels.

Add the extra SSDs to the DGX Station as explained in DGX Station User Guide.

Because expanding a RAID array risks loss of data, ensure that you have a backup of data on the array that you want to preserve.

This task requires `sudo` privileges.

Use standard Linux operating system commands to expand the DGX Station RAID array.

1. Obtain the device IDs of the SSDs that were added by searching for 1.8 T drives that aren't mounted in the output from the lsblk (http://manpages.ubuntu.com/manpages/bionic/man8/lsblk.8.html) command.

   ```
   $ lsblk
   ```

   In the following example, the device IDs of the SSDs that were added are `sde`, `sdf`, `sdg`, and `sdh`.

   ```
   $ lsblk
   NAME   MAJ:MIN RM   SIZE RO TYPE  MOUNTPOINT
   sda      8:0    0   1.8T  0 disk
   |_sda1   8:1    0   487M  0 part  /boot/efi
   |_sda2   8:2    0   1.8T  0 part  /
   sdb      8:16   0   1.8T  0 disk
   |_md0    9:0    0   5.2T  0 raid0 /raid
   sdc      8:32   0   1.8T  0 disk
   |_md0    9:0    0   5.2T  0 raid0 /raid
   sdd      8:48   0   1.8T  0 disk
   |_md0    9:0    0   5.2T  0 raid0 /raid
   sde      8:64   0   1.8T 0 disk
   sdf      8:80   0   1.8T 0 disk
   sdg      8:96   0   1.8T 0 disk
   sdh      8:112  0   1.8T 0 disk
   ```

2. Add the SSDs that you added to the DGX Station to the RAID array.

   ```
   $ sudo mdadm --add raid-device-path ssd-device-paths
   ```

   raid-device-path

   The device path of the RAID array, for example, `/dev/md0`.

   ssd-device-paths

   A space-separated list of the device paths of the SSDs that you added, in which each path is of the form `/dev/device-id`.

This example adds the SSDs with device paths `/dev/sde`, `/dev/sdf`, `/dev/sdg`, and `/dev/sdh` to the RAID array with device path `/dev/md0`.

```
sudo mdadm --add /dev/md0 /dev/sde /dev/sdf /dev/sdg /dev/sdh
```

3. Increase the number of devices in the RAID array to 7.

```
$ sudo mdadm --grow --raid-devices=7 raid-device-path
```

raid-device-path

The device path of the RAID array, for example, `/dev/md0`.

Note: Increasing the number of devices in the RAID array to 7 may require several hours or even longer. If the system crashes, is shut down, or is rebooted while the number of devices on the array is being increased, all data that was on the array is erased.

This example increases the number of devices in the RAID array with device path `/dev/md0` to 7.

```
$ sudo mdadm --grow --raid-devices=7 /dev/md0
```

4. Resize the file system that resides on the RAID array to use the additional physical space in the array.

```
$ sudo resize2fs raid-device-path
```

raid-device-path

The device path of the RAID array, for example, `/dev/md0`.

This example resizes the file system that resides on the RAID array with device path `/dev/md0`.

```
$ sudo resize2fs /dev/md0
```

The RAID array is expanded with its existing RAID level. The data on the array is preserved, even if the array is a RAID 0 array.

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.

Trademarks

NVIDIA, the NVIDIA logo, DGX, DGX-1, DGX-2, DGX A100, 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.

Copyright

© 2021 NVIDIA Corporation. All rights reserved.

NVIDIA Corporation | 2788 San Tomas Expressway, Santa Clara, CA 95051
www.nvidia.com