

Configuring NFS Over RDMA Using ROCEv2

- Configuring NFS over RDMA on Cisco UCS Manager, on page 1
- Deleting the NFS over RDMA Interface Using Cisco UCS Manager, on page 2
- Configuring the NFS over RDMA Interface Using the UCS Manager CLI, on page 3
- Deleting the NFS over RDMA Interface Using the Cisco UCS Manager CLI, on page 4
- Configuring NFS over RDMA on the Linux Host (AMD), on page 5
- Configuring NFS over RDMA on the Linux Host (Intel), on page 6
- Installing Cisco eNIC and enic_rdma Drivers, on page 7
- Mounting and Verifying NFS over RDMA Volumes, on page 8

Configuring NFS over RDMA on Cisco UCS Manager

Use the following steps to configure the Remote Direct Memory Access (RDMA) over Converged Ethernet version 2 (RoCEv2) interface on Cisco UCS Manager to support Network File System (NFS) over RDMA.

Before you begin

The **Linux-NVMe-RoCE** adapter policy is already defined in the system.

Procedure

Step 1 In the Navigation pane, click **Servers**.

Step 2 Expand **Servers** > **Service Profiles**.

Step 3 Expand the node for the organization where you want to create the policy.

If the system does not include multitenancy, expand the root node.

Step 4 Select the desired **Service Profile**.

Step 5 Click **vNICs** and go to the **Network** tab in the work area.

Step 6 Modify the vNIC policy:

a) In the **vNICs** area, select the desired vNIC and click **Modify**

b) In the **Modify vNIC** dialog box, scroll down to the **Adapter Performance Profile** area.

c) Click the **Adapter Policy** drop-down and choose **`Linux-NVMe-RoCE`**.

d) Click **OK**.

Step 7 Click **Save Changes**.

Step 8 Select **Reboot** to apply the changes to the server.

The Cisco UCS Manager applies the new adapter policy to the server, and the server initiates a reboot to finalize the configuration.

What to do next

After the server reboots, proceed to configure the host-side NFS over RDMA settings as described in the host configuration procedure.

Deleting the NFS over RDMA Interface Using Cisco UCS Manager

Use the following steps to remove the Network File System (NFS) over Remote Direct Memory Access (RDMA) configuration from a specific vNIC.

Procedure

Step 1 In the **Navigation** pane, click **Servers**.

Step 2 Expand **Servers** > **Service Profiles**.

Step 3 Expand the node for the organization where the service profile is located. (If the system does not include multitenancy, expand the root node).

Step 4 Modify the vNIC policy, according to the steps below.

a) Click on **vNICs** and go to the **Network** tab in the work area.
b) Scroll down to the desired vNIC, select it, and click **Modify**.
c) In the popup dialog box, scroll down to the **Adapter Performance Profile** area.
d) From the **Adapter Policy** drop-down list, choose **Linux** (or the appropriate standard adapter policy for your environment) to remove the `Linux-NVMe-RoCE` configuration.
e) Click **OK**.

Step 5 Click **Save Changes** and reboot the server.

What to do next

The Cisco UCS Manager applies the standard adapter policy to the server and reboot.

Configuring the NFS over RDMA Interface Using the UCS Manager CLI

Use the following steps to configure the Network File System (NFS) over Remote Direct Memory Access (RDMA) for Linux in the Cisco UCS Manager Command Line Interface (CLI).

Before you begin

- Cisco UCS Manager is installed and accessible.
- The server is associated with a Service Profile.
- The **Linux-NVMe-RoCE** adapter policy is already defined in the system.

Procedure

Step 1 **Example:**

```
UCS-A # scope service-profile server chassis-id / blade-id or rack_server-id
```

Enter the service profile for the specified chassis, blade or UCS managed rack server ID.

Step 2 **Example:**

```
UCS-A /org/service-profile # show vnic
```

Display the vNICs available on the server.

Step 3 **Example:**

```
UCS-A /org/service-profile # scope vnic eth0
```

Enter the vnic mode for the specified vNIC (in this example, `eth0`).

Step 4 **Example:**

```
UCS-A /org/service-profile/vnic # set adapter-policy Linux-NVMe-RoCE
```

Specifies the adapter policy for the vNIC used for NFS over RDMA.

Step 5 **Example:**

```
UCS-A /org/service-profile/vnic* # commit-buffer
```

Commit the transaction to the system configuration.

The Cisco UCS Manager applies the new adapter policy to the server, and the server initiates a reboot to finalize the configuration.

Example

This example shows how to configure the *NFS over RDMA* adapter policy on the `eth01` vNIC:

```
UCS-A# scope service-profile server 1/1
UCS-A /org/service-profile # show vnic

vNIC:
    Name               Fabric ID Dynamic MAC Addr   Virtualization Preference
    ------------------ --------- ------------------ -------------------------
    eth00              A B       00:25:B5:3A:84:00  NONE
    eth01              A         00:25:B5:3A:84:01  NONE
    eth02              B         00:25:B5:3A:84:02  NONE

UCS-A /org/service-profile # scope vnic eth01
UCS-A /org/service-profile/vnic # set adapter-policy Linux-NVMe-RoCE
UCS-A /org/service-profile/vnic* # commit-buffer
UCS-A /org/service-profile/vnic #
```

What to do next

- The Cisco UCS Manager applies the new adapter policy to the server, and the server initiates a reboot to finalize the configuration.
- After the server reboots, proceed to configure the host-side NFS over RDMA settings.

Deleting the NFS over RDMA Interface Using the Cisco UCS Manager CLI

Use the following steps to remove the Network File System (NFS) over Remote Direct Memory Access (RDMA) interface configuration in the Cisco UCS Manager CLI by reverting the adapter policy to the default Linux policy.

Before you begin

You must be logged in with administrative privileges.

Procedure

Step 1 **Example:**

```
UCS-A # scope service-profile server chassis-id / blade-id or rack_server-id
```

Enter the service profile for the specified chassis, blade or UCS managed rack server ID.

Step 2 **Example:**

```
UCS-A /org/service-profile # show vnic
```

Display the vNICs available on the server.

Step 3 **Example:**

```
UCS-A /org/service-profile # scope vnic name
```

Enter the vnic mode for the specified vNIC.

Step 4 **Example:**

```
UCS-A /org/service-profile/vnic # set adapter-policy Linux
```

Removes Linux-NVMe-RoCE policy by setting the default Linux adapter policy.

Step 5 **Example:**

```
UCS-A /org/service-profile/vnic* # commit-buffer
```

Commit the transaction to the system configuration.

Step 6 Reboot the server to fully remove the RDMA-enabled interface from the host operating system.

This example shows how to remove the RoCEv2 interface on the eth01 vNIC on Linux.

Example

Removing the NFS over RDMA Interface

```
UCS-A# scope service-profile server 1/1
UCS-A /org/service-profile # show vnic

vNIC:
    Name               Fabric ID Dynamic MAC Addr   Virtualization Preference
    ------------------ --------- ------------------ -------------------------
    eth00              A B       00:25:B5:3A:84:00  NONE
    eth01              A         00:25:B5:3A:84:01  NONE
    eth02              B         00:25:B5:3A:84:02  NONE
UCS-A /org/service-profile # scope vnic eth01
UCS-A /org/service-profile/vnic # set adapter-policy Linux
UCS-A /org/service-profile/vnic* # commit-buffer
```

What to do next

Reboot the server to remove *Linux-NVMe-RoCE* adapter policy and apply the new policy.

Configuring NFS over RDMA on the Linux Host (AMD)

Before you begin

- Ensure the server is configured with RoCEv2 vNICs.
- Ensure you have administrative (root or sudo) privileges to the Linux host.
- Verify that the host system is AMD-based.

Use this procedure to configure NFS over RDMA on the Red Hat Enterprise Linux (RHEL) host for AMD-based systems.

Procedure

Step 1 Edit the GRUB configuration. Open the `/etc/default/grub` file for editing.

Step 2 Add `amd_iommu=on` to the end of the line for `GRUB_CMDLINE_LINUX` as shown in the following sample:

Sample:

```
# cat /etc/default/grub
                          GRUB_CMDLINE_LINUX="... rhgb quiet amd_iommu=on"
```

Step 3 Save the file.

Step 4 Generate the new GRUB configuration file (grub.cfg) for UEFI systems:

For UEFI boot:

Sample:

```
# grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
```

Step 5 Reboot the server to apply the changes.

```
# reboot
```

Note: The server must be rebooted for the IOMMU changes to take effect.

Step 6 Verify that the server booted with the `amd_iommu=on` option.

```
# cat /proc/cmdline | grep iommu
```

The output should confirm the inclusion of `amd_iommu=on` at the end of the line.

Sample Output:

```
BOOT_IMAGE=(hd0,gpt2)/vmlinuz-6.12.0-124.8.1.el10_1.x86_64 ... rhgb quiet amd_iommu=on
```

The IOMMU option is enabled, and the system is prepared for the installation of the necessary RDMA drivers.

What to do next

- Proceed to install the downloaded `enic` and `enic_rdma` drivers.
- Configure your NFS mount points to utilize the RDMA interface as described in the mounting procedure.

Configuring NFS over RDMA on the Linux Host (Intel)

Before you begin

- Ensure the server is configured with RoCEv2 vNICs.
- Ensure you have administrative (root or sudo) privileges to the Linux host.
- Verify that the host system is Intel-based.

Use this procedure to configure NFS over RDMA on the Red Hat Enterprise Linux (RHEL) host (Intel-based system).

Procedure

Step 1 Open the `/etc/default/grub` file for editing.

Step 2 Add `intel_iommu=on` to the end of the line for `GRUB_CMDLINE_LINUX` as shown in the sample file below.

```
sample /etc/default/grub configuration file after adding intel_iommu=on:
# cat /etc/default/grub
GRUB_TIMEOUT=5
GRUB_DISTRIBUTOR="$(sed 's, release .*$,,g' /etc/system-release)"
GRUB_DEFAULT=saved
GRUB_DISABLE_SUBMENU=true
GRUB_TERMINAL_OUTPUT="console"
GRUB_CMDLINE_LINUX="crashkernel=auto rd.lvm.lv=rhel/root rd.lvm.lv=rhel/swap biosdevname=1
 rhgb quiet intel_iommu=on
GRUB_DISABLE_RECOVERY="true"
```

Step 3 Save the file.

Step 4 After saving the file, run the following command to generate a new grub.cfg file:

- For Legacy boot:

  ```
  # grub2-mkconfig -o /boot/grub2/grub.cfg
  ```

- For UEFI boot:

  ```
  # grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
  ```

Step 5 Reboot the server.

You must reboot your server for the changes to take after enabling IOMMU.

Step 6 Verify that the server booted with the `intel_iommu=on` option: `# cat /proc/cmdline | grep iommu. The output should confirm the inclusion of intel_iommu=on at the end of the line.`

```
Sample Output: The output should confirm the inclusion of intel_iommu=on at the end of the
 line.

[root@localhost basic-setup]# cat /proc/cmdline | grep iommu
BOOT_IMAGE=/vmlinuz-3.10.0-957.27.2.el7.x86_64 root=/dev/mapper/rhel-
root ro crashkernel=auto rd.lvm.lv=rhel/root rd.lvm.lv=rhel/swap rhgb
quiet intel_iommu=on LANG=en_US.UTF-8
```

The IOMMU option is enabled, and the system is prepared for the installation of the necessary RDMA drivers.

What to do next

- Proceed to install the downloaded `enic` and `enic_rdma` drivers.
- Configure your NFS mount points to utilize the RDMA interface.

Installing Cisco eNIC and enic_rdma Drivers

Use the following steps to install Cisco eNIC and enic_rdma Drivers for NFS over RDMA.

Before you begin

- Ensure you have downloaded the matched set of eNIC and enic_rdma drivers from the Cisco support portal.

- Verify that the target server is installed with the supported RHEL version.
- Ensure you have administrative (root or sudo) privileges to perform driver installation.

Note Using the `enic_rdma` driver binary with an inbox driver is not supported. Ensure you are using the matched set of drivers provided by Cisco.

Procedure

Step 1 Install the eNIC and enic_rdma driver package on the host: `# rpm -ivh kmod-enic-<version>.x86_64.rpm kmod-enic_rdma-<version>.x86_64.rpm`

Step 2 Reboot the server to load the drivers into the running kernel: `# reboot`

Step 3 Verify that the drivers are loaded and RoCEv2 is enabled by checking the system logs: `# dmesg | grep enic_rdma`.

The system should display the following output:

```
[    1.562913] enic_rdma: Cisco VIC Ethernet NIC RDMA Driver, ver 1.10.366.0-1233.0 init
[    1.687502] enic 0000:75:00.0 enp117s0f0: enic_rdma: FW v4 RoCEv2 enabled
[    1.771968] enic 0000:75:00.1 enp117s0f1: enic_rdma: FW v4 RoCEv2 enabled
```

The eNIC and enic_rdma drivers are successfully installed and loaded. The system logs confirm that the Cisco VIC interfaces are initialized and RoCEv2 functionality is enabled on the respective ports.

What to do next

After verifying the driver installation, proceed to configure the NFS over RDMA mount points.

Mounting and Verifying NFS over RDMA Volumes

This procedure provides instructions for mounting and verifying NFS volumes using the RDMA protocol on the host operating system.

Before you begin

- Ensure the entire network infrastructure is configured to support NFS over RDMA traffic. The configuration must be consistent across the Cisco VIC adapter, upstream network switches, and the storage target.
- Confirm Cisco VIC firmware and drivers (**enic** and **enic_rdma**) are up to date and compatible with RDMA and RoCEv2.
- Ensure MTU is set appropriately on the VIC, host OS, and network switches to optimize RDMA performance.

- Confirm Cisco UCS Manager or IMC is configured correctly for RoCEv2, including enabling RoCE properties on the vNIC and disabling incompatible failover features.
- Validate that RDMA interfaces are up and configured with correct IP addressing matching the storage network.

Procedure

Step 1 Create a local directory to serve as the mount point:

```
# mkdir /<mount_point>
```

Sample

```
# mkdir /mnt/nfs_volume1
```

Step 2 Mount the NFS volume using the RDMA protocol:

- **IPv4:**

  ```
  mount -o proto=rdma <storage IPV4>:/<nfs volume>/<mount_point>
  ```

  Sample

  ```
  mount -o proto=rdma 192.168.1.100:/mnt/nfs_volume1
  ```

- **IPv6:**

  ```
  mount -o proto=rdma6 '[IPv6_address]:/<volume path>'/<mount_point>
  ```

  Sample

  ```
  mount -o proto=rdma6 '[20:34:d0:27:ea:19:70:50]:/<volume path>'/<mount point>
  ```

  Replace <IPv4_address>/<IPv6_address>, <volume_path>, and <mount_point> with your actual IP addresses, NFS export path, and desired local mount directory.

Step 3 Verify that the NFS over RDMA mount is created and active. Run the following command to filter the mount list for RDMA-specific connections:

```
# mount | grep rdma
```

Sample Output

```
50.19.40.5:/mnt/nfs_volume1 type nfs4 (rw, relatime, vers=4.1, rsize=65536, wsize=65536,
namlen=255, hard, fatal_neterrors=none, proto=rdma, nconnect=4, port=20049, timeo=600,
retrans=2, sec=sys, clientaddr=, local_lock=none, addr=)
```

Step 4 Verify the volume capacity and mount status using the df command. Use the `df` command to ensure the volume is correctly mapped and reporting the expected storage capacity:

```
# df -h
```

Sample Output

```
<target IP>:/nfs_vol    501G  3.6G  497G   1% /mnt/nfs_volume1
```

Step 5 Check the RDMA statistics to verify traffic flow:

This command provides NFS statistics on the mount, confirming RDMA usage and showing incrementing counts during active traffic.

```
# mountstats --xprt <mount_point>
```

Step 6 To unmount the volume, use the following command:

```
# umount <mount_point>
```

What to do next

After successfully mounting and verifying the NFS over RDMA volume, perform the following tasks:

- Configure Persistent Mounting: To ensure the volume mounts automatically after a system reboot, add an entry to the `/etc/fstab` file. *Example entry:*

  ```
  <storage_ip>:/<volume_path> <mount_point> nfs rdma,nconnect=4,x-systemd.automount 0
          0
  ```

- Configure Application Workloads: Update your application or database configurations to point to the new `<mount_point>` to begin utilizing the high-throughput, low-latency RDMA storage.