Technical White Paper

PVRDMA Deployment and Configuration of QLogic CNA devices in VMware ESXi

Abstract

In server connectivity, transferring large amounts data can be a major overhead on the processor. In a conventional networking stack, packets received are stored in the memory of the operating system and later transferred to the application memory. This transfer causes a latency. Network adapters that implement Remote Direct Memory Access (RDMA) write data directly to the application memory. Paravirtualized RDMA (PVRDMA) introduces an RDMA-capable network interface for virtual machines explicitly to communicate over an RDMA-capable physical device. This white paper provides step-by-step instructions to set up PVRDMA for QLogic Converged Network Adapter (CNA) devices.

June 2020

Revisions

Date	Description
June 2020	Initial release

Acknowledgements

Author: Syed Hussaini, Software Engineer

Support: Krishnaprasad K, Senior Principal Engineering Technologist

Gurupreet Kaushik, Technical Writer, IDD.

The information in this publication is provided “as is.” Dell Inc. makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose.

Table of contents

Executive summary

The speed at which data can be transferred is critical for efficiently using information. RDMA offers an ideal option for helping with better data center efficiency by reducing overall complexity and increasing the performance of data delivery.

RDMA is designed to transfer data from storage device to server without passing the data through the CPU and main memory path of TCP/IP Ethernet. Better processor and overall system efficiencies are achieved as the computation power is harnessed and not used in processing network traffic.

RDMA enables sub-microsecond latencies and up to 56 Gb/s bandwidth, translating into swift application performance, better storage and data center utilization, and simplified network management.

Until recently, RDMA was only available in InfiniBand fabrics. With the advent of RDMA over Converged Ethernet (RoCE), the benefits of RDMA are now available for data centers that are based on an Ethernet or mixed-protocol fabric as well.

For more information about RDMA and the protocols that are used, see Dell Networking – RDMA over Converged.

1 Introduction

This document is intended to help the user understand Remote Direct Memory Access and provides step-by-step instructions to configure the RDMA or RoCE feature on Dell EMC PowerEdge server with QLogic network card on VMware ESXi.

1.1 Audience and Scope

This white paper is intended for IT administrators and channel partners planning to configure Paravirtual RDMA on Dell EMC PowerEdge servers. PVRDMA is a feature that is aimed at customers who are concerned with faster data transfer between hosts that are configured with PVRDMA, and to ensure less CPU utilization with reduced cost.

1.2 Remote Direct Memory Access

Note: The experiments described here are conducted using the Dell EMC PowerEdge R7525 server.

RDMA allows us to perform direct data transfer in and out of a server by implementing a transport protocol in the network interface card (NIC) hardware. The technology supports zero-copy networking, a feature that enables data to be read directly from the main memory of one system and then write the data directly to the main memory of another system.

If the sending device and the receiving device both support RDMA, then the communication between the two is quicker when compared to non-RDMA network systems.

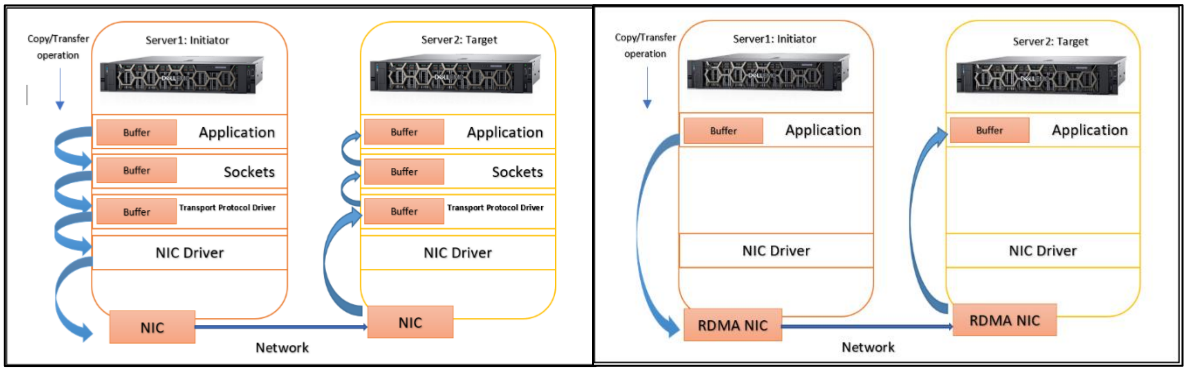


Figure 1 RDMA workflow

The RDMA workflow figure shows a standard network connection on the left and an RDMA connection on the right. The initiator and the target must use the same type of RDMA technology such as RDMA over Converged Ethernet or InfiniBand.

Studies have shown that RDMA is useful in applications that require fast and massive parallel high-performance computing (HPC) clusters and data center networks. It is also useful when analyzing big data, in supercomputing environments that process applications, and for machine learning that requires lower latencies with higher data transfer rates. RDMA is implemented in connections between nodes in compute clusters and in latency-sensitive database workloads. For more information about the studies and discussions carried out on RDMA, see www.vmworld.com.

1.3 Paravirtual RDMA

Paravirtual RDMA (PVRDMA) is a new PCIe virtual network interface card (NIC) which supports standard RDMA APIs and is offered to a virtual machine on VMware vSphere 6.5.

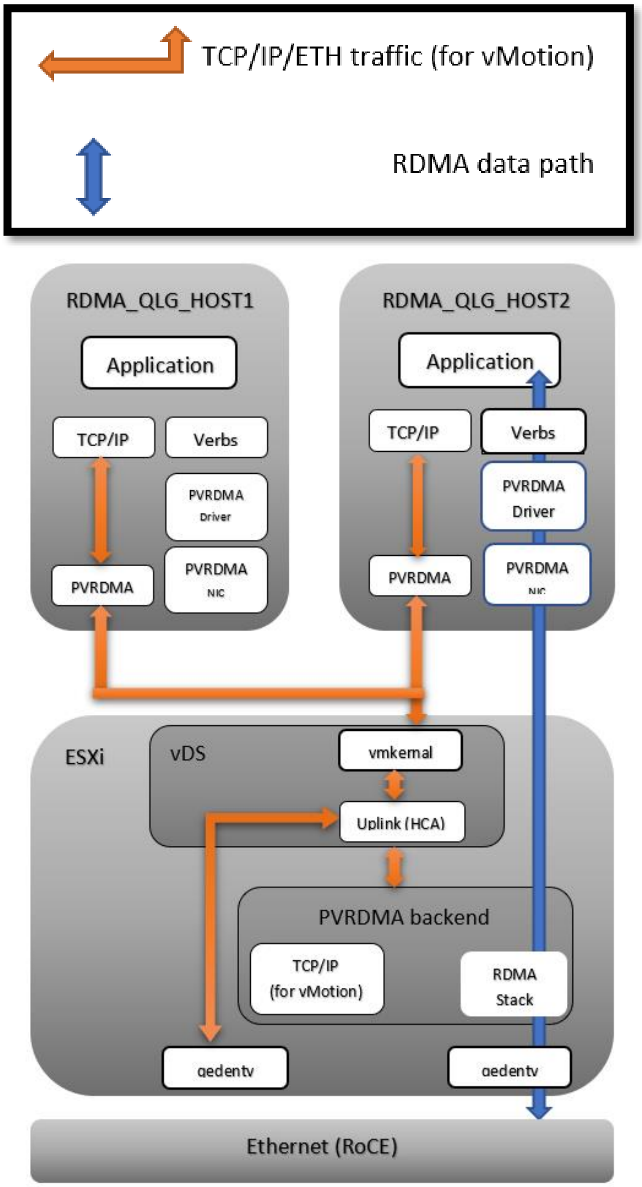


Figure 2 PVRDMA architecture

In Figure 2, notice that PVRDMA is deployed on two virtual machines: RDMA_QLG_HOST1 and RDMA_QLG_HOST2. The following table describes the components of the architecture.

Table 1 Components of PVRDMA architecture

Component	Description
PVRDMA NIC	The virtual PCIe device providing Ethernet Interface through PVRDMA, the adapter type and RDMA.
Verbs	RDMA API calls that are proxied to the PVRDMA back-end. The user library provides direct access to the hardware with a path for data.
PVRDMA driver	Enables the virtual NIC (vNIC) with the IP stack in the kernel space. It also provides full support for Verbs RDMA API in the user space.
ESXi PVRDMA backend	Creates virtual RDMA resources for the virtual machine where guests can make use of the resources. It supports features such as Live vMotion, snapshots and high availability (HA).
ESXi Server	Provides physical Host Channel Adapter (HCA) services on all virtual machines. Leverages native RDMA and core drivers and creates corresponding resources in HCA.

1.4 Hardware and software requirements

Note: Configuration of the servers chosen to perform this task should be verified for compatibility on VMware Compatibility Guide.

To enable RoCE feature on Dell EMC PowerEdge server with QLogic Network Card on VMware ESXi, the following components are used:

- Server**:** Dell EMC PowerEdge R7525 server
- Network card**:** QLogic 2x10GE QL41132HQCU NIC
- Cable: Dell Networking, Cable, SFP+ to SFP+, 10GbE, Copper twinax direct attach cable
- Host OS: VMware ESXi 6.7 or later
- Guest OS: Red Hat Enterprise Linux Version 7.6 or later

The installation process of the network drivers for the attached NIC is contingent on the virtual machine tools that are used and the version of the operating system that is installed on the host and guest environments.

1.5 Supported configuration

For information about the network cards that are supported to set up this configuration, see the VMware Compatibility Guide page. Select **RoCE v1** and **RoCE v2** options from the **Features** tab and then select the **DELL** option from the **Brand Name** tab.

For information about the network protocols used, see Dell Networking – RDMA over Converged or the Intel® Ethernet Network Adapters Support page.

vSphere 6.5 and later versions support PVRDMA only in environments with specific a configuration. For more information, see PVRDMA Support.

2 Configuring PVRDMA on VMware vSphere

This section describes how PRDMA is configured as a virtual NIC when assigned to virtual machines and steps to enable it on host and guest operating systems. This section also includes the test results when using PVRDMA on virtual machines.

Note: Ensure that the host is configured and meets the prerequisites to enable PVRDMA.

2.1 VMware ESXi host configuration

Dell EMC PowerEdge servers with VMware ESXi host installed are used to validate and test the functionality of RDMA. Note the IP addresses and vmnic names as they are used to set nodes and parameters.

2.1.1 Checking host configuration

To check host configuration, do the following:

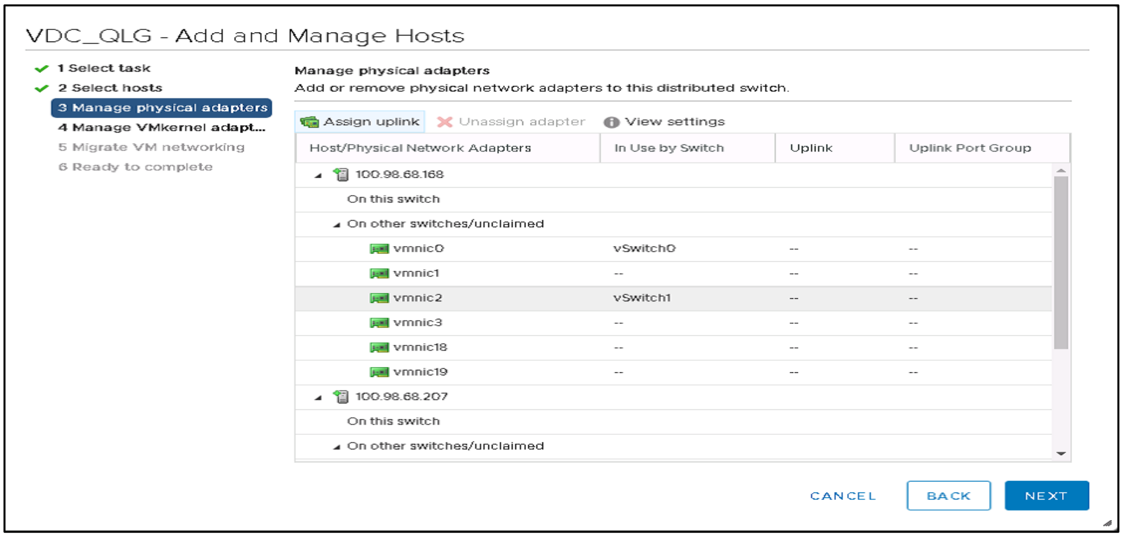
1. Enable SSH access to the VMware ESXi server.
2. Log in to the VMware ESXi vSphere command-line interface with root permissions.
3. Verify that the host is equipped with an adapter card which supports RDMA or RoCE.

2.1.2 Deploying PVRDMA on VMware vSphere

A vSphere Distributed Switch (vDS) must be created before deploying PVRDMA. It provides for a centralized management and monitoring of the networking configurations for all the hosts that are associated with the switch. A distributed switch must be set up on a vCenter Server system, and the settings are propagated to all the hosts associated with this switch.

To start the deployment process:

1. Create a vSphere Distributed Switch (vDS). For more information about how to create a vDS, see Setting Up Networking with vSphere Distributed Switches.
2. Configure vmnic in each VMware ESXi host as an Uplink 1 for vDS.



VDC_QLG - Add and Manage Hosts

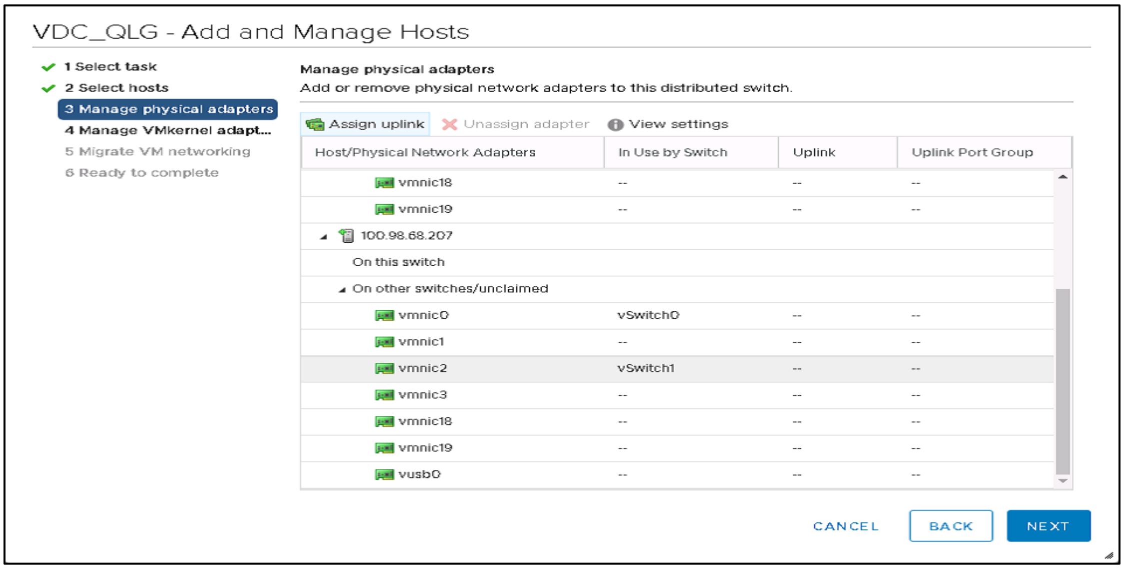
Manage physical adapters

Add or remove physical network adapters to this distributed switch.

Host/Physical Network Adapters	In Use by Switch	Uplink	Uplink Port Group
100.98.68.168			
On this switch			
On other switches/unclaimed			
vmnic0	vSwitch0	--	--
vmnic1	--	--	--
vmnic2	vSwitch1	--	--
vmnic3	--	--	--
vmnic18	--	--	--
vmnic19	--	--	--
100.98.68.207			
On this switch			
On other switches/unclaimed			

Figure 3 Configure vmnic in each ESXI host as an Uplink for vDS.

3. Assign the uplinks.

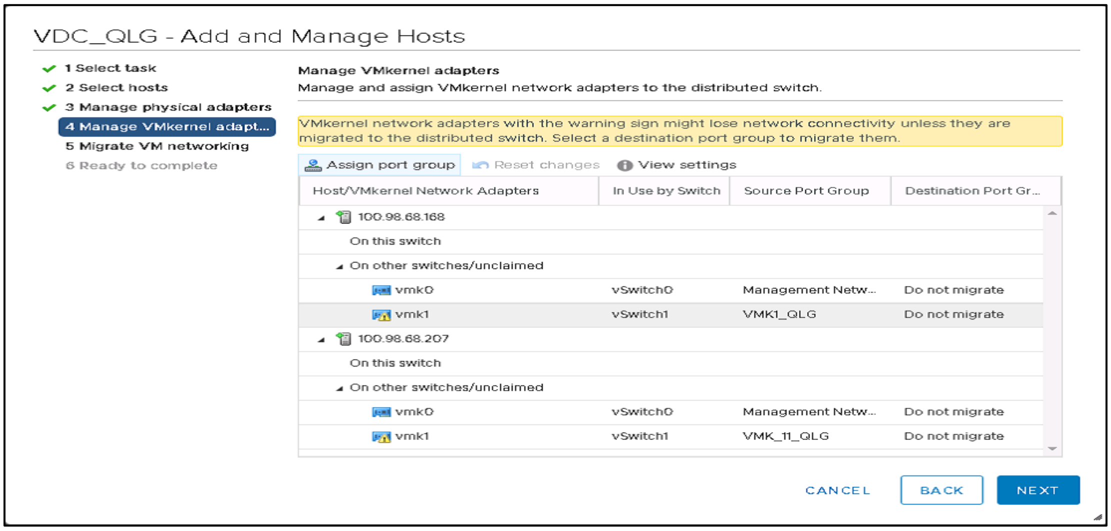


Manage physical adapters
Add or remove physical network adapters to this distributed switch.

Host/Physical Network Adapters	In Use by Switch	Uplink	Uplink Port Group
vmnic18	--	--	--
vmnic19	--	--	--
100.98.68.207			
On this switch			
On other switches/unclaimed			
vmnic0	vSwitch0	--	--
vmnic1	--	--	--
vmnic2	vSwitch1	--	--
vmnic3	--	--	--
vmnic18	--	--	--
vmnic19	--	--	--
vusb0	--	--	--

Figure 4 Assigning the uplinks

4. Attach the vmkernel adapter vmk1 to vDS port group.



Manage VMkernel adapters
Manage and assign VMkernel network adapters to the distributed switch.

VMkernel network adapters with the warning sign might lose network connectivity unless they are migrated to the distributed switch. Select a destination port group to migrate them.

Host/VMkernel Network Adapters	In Use by Switch	Source Port Group	Destination Port Gr...
100.98.68.168			
On this switch			
On other switches/unclaimed			
vmk0	vSwitch0	Management Netw...	Do not migrate
vmk1	vSwitch1	VMK1_QLG	Do not migrate
100.98.68.207			
On this switch			
On other switches/unclaimed			
vmk0	vSwitch0	Management Netw...	Do not migrate
vmk1	vSwitch1	VMK_11_QLG	Do not migrate

Figure 5 Attach the vmkernel adapter vmk1 to vDS port group.

5. Click **Next** and then, click **Finish**.

2.1.3 Updating vSphere settings

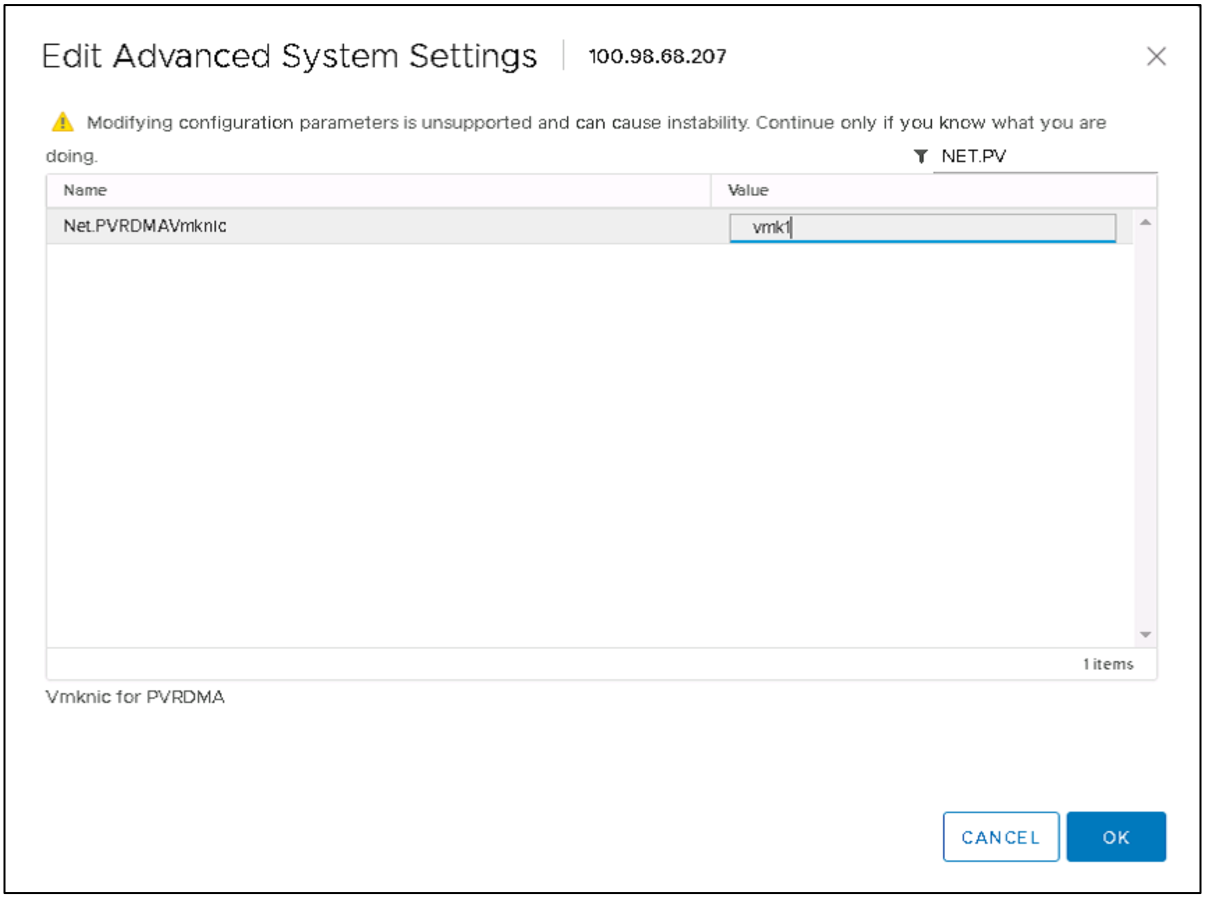
With the vSphere Distributed Switch active, the following features can be enabled on vSphere to configure a VMware ESXi host for PVRDMA:

- Tag a VMKernel adapter that was created earlier for PVRDMA
- Enable the firewall rule for PVRDMA
- Assign the PVRDMA adapter to one or more virtual machines

To enable the features listed:

1. Tag a VMkernel Adapter.

    a. Go to the host on the vSphere Web Client.
    b. Under the **Configure** tab, expand the **System** section and click **Advanced System Settings**.



Edit Advanced System Settings | 100.98.68.207

Modifying configuration parameters is unsupported and can cause instability. Continue only if you know what you are doing.

NET.PV

Name	Value
Net.PVRDMAVmknic	vmk1

1 items

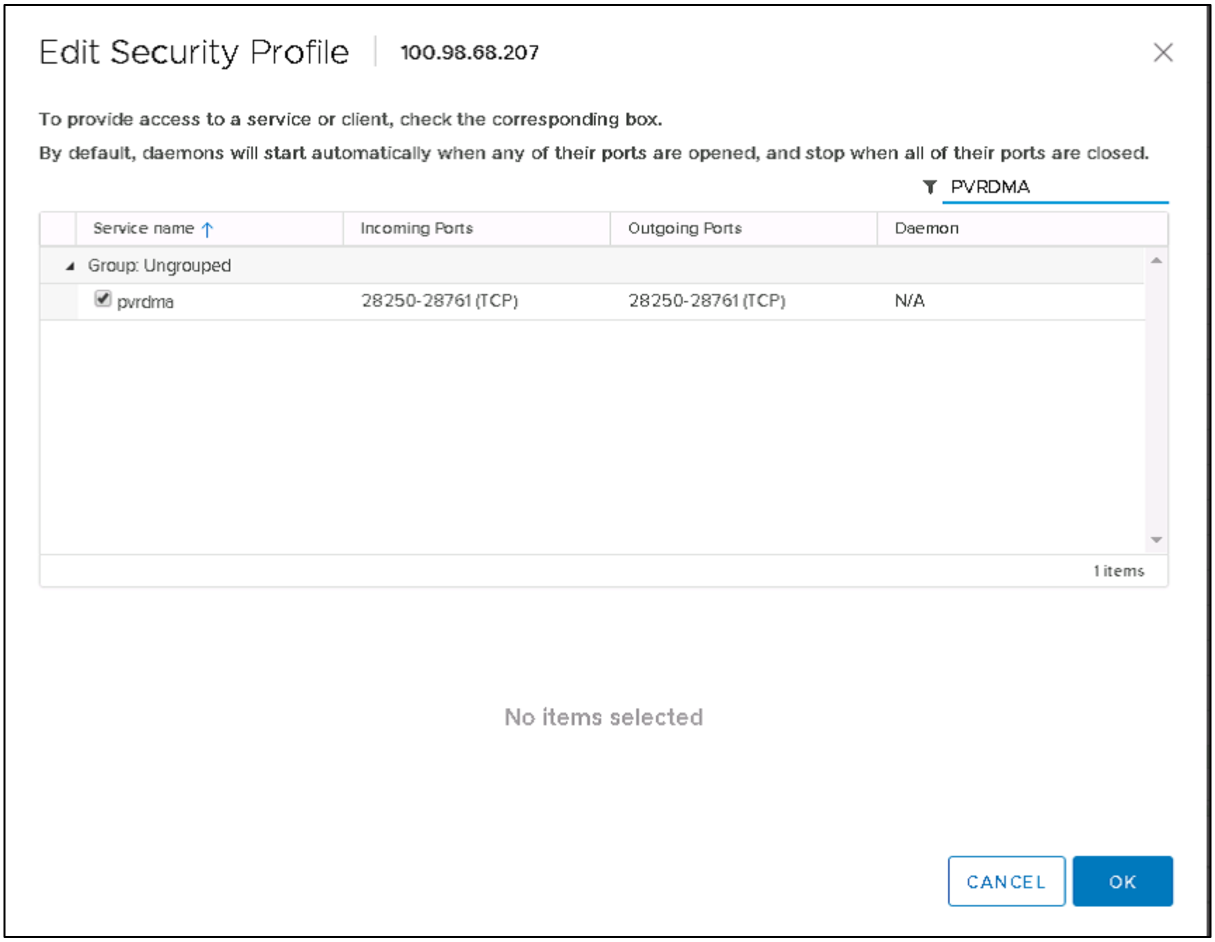
Vmknic for PVRDMA

Figure 6 Tag a vmkernal Adapter

    c. Locate Net.PVRDMAvmknic and click **Edit**.
    d. Enter the value of the VMkernel adapter that will be used and click **OK** to finish.

2. Enable the firewall rule for PVRDMA.

    a. Go to the host on the vSphere Web Client.
    b. Under the **Configure** tab, expand the **Firewall** tab and click **Edit.**
    c. Find the PVRDMA rule by scrolling down and **select the checkbox** next to it.



Edit Security Profile | 100.98.68.207

To provide access to a service or client, check the corresponding box.

By default, daemons will start automatically when any of their ports are opened, and stop when all of their ports are closed.

PVRDMA

Service name ↑	Incoming Ports	Outgoing Ports	Daemon
Group: Ungrouped			
☑ pvrdma	28250-28761 (TCP)	28250-28761 (TCP)	N/A

1 items

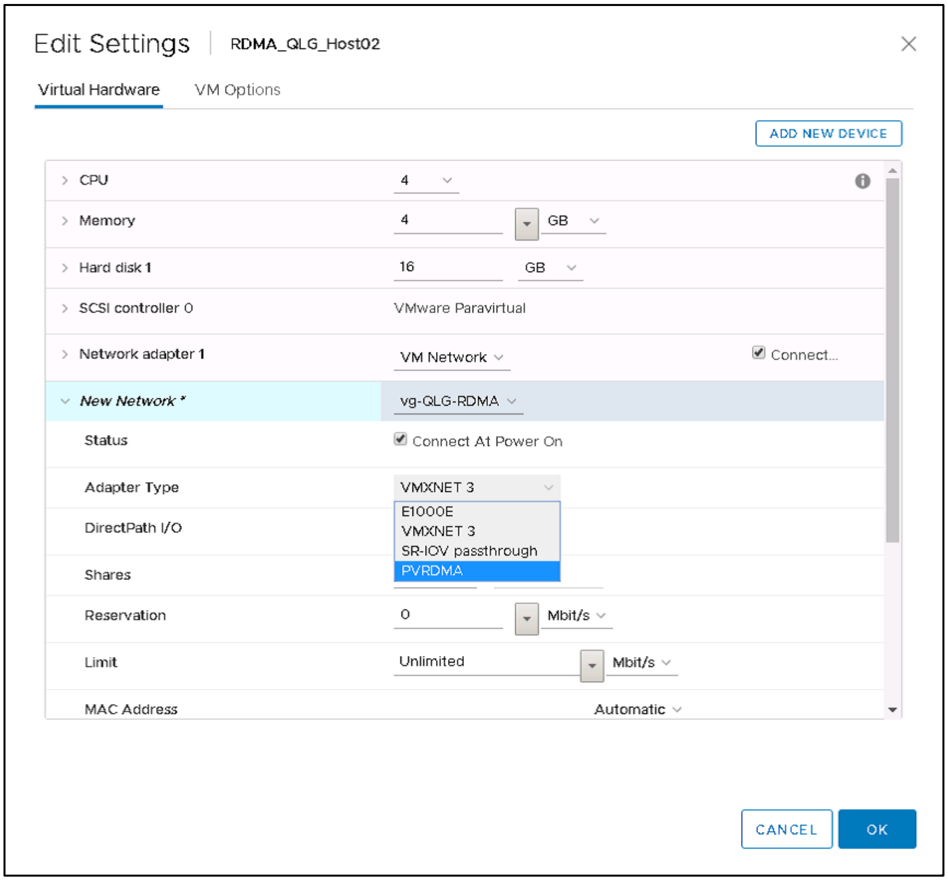
No items selected

Figure 7 Enabling the firewall rule for PVRDMA

d. Click **OK** to complete enabling the firewall rule.

3. Assign the PVRDMA adapter to a virtual machine.

   a. Locate the virtual machine on the vSphere web client.
   b. **Right-click** on the VM and choose to **Edit**.
   c. VM Hardware is selected by default.
   d. Click the **Add new device** and select **Network Adapter**.
   e. Select the distributed switch created earlier from the Deploying PVRDMA on VMware vSphere section and click **OK**.



Edit Settings | RDMA_QLG_Host02

CPU 4

Memory 4 GB

Hard disk 1 16 GB

SCSI controller 0 VMware Paravirtual

Network adapter 1 VM Network ☑ Connect...

*New Network ** vg-QLG-RDMA

Status ☑ Connect At Power On

Adapter Type VMXNET 3

DirectPath I/O

Shares

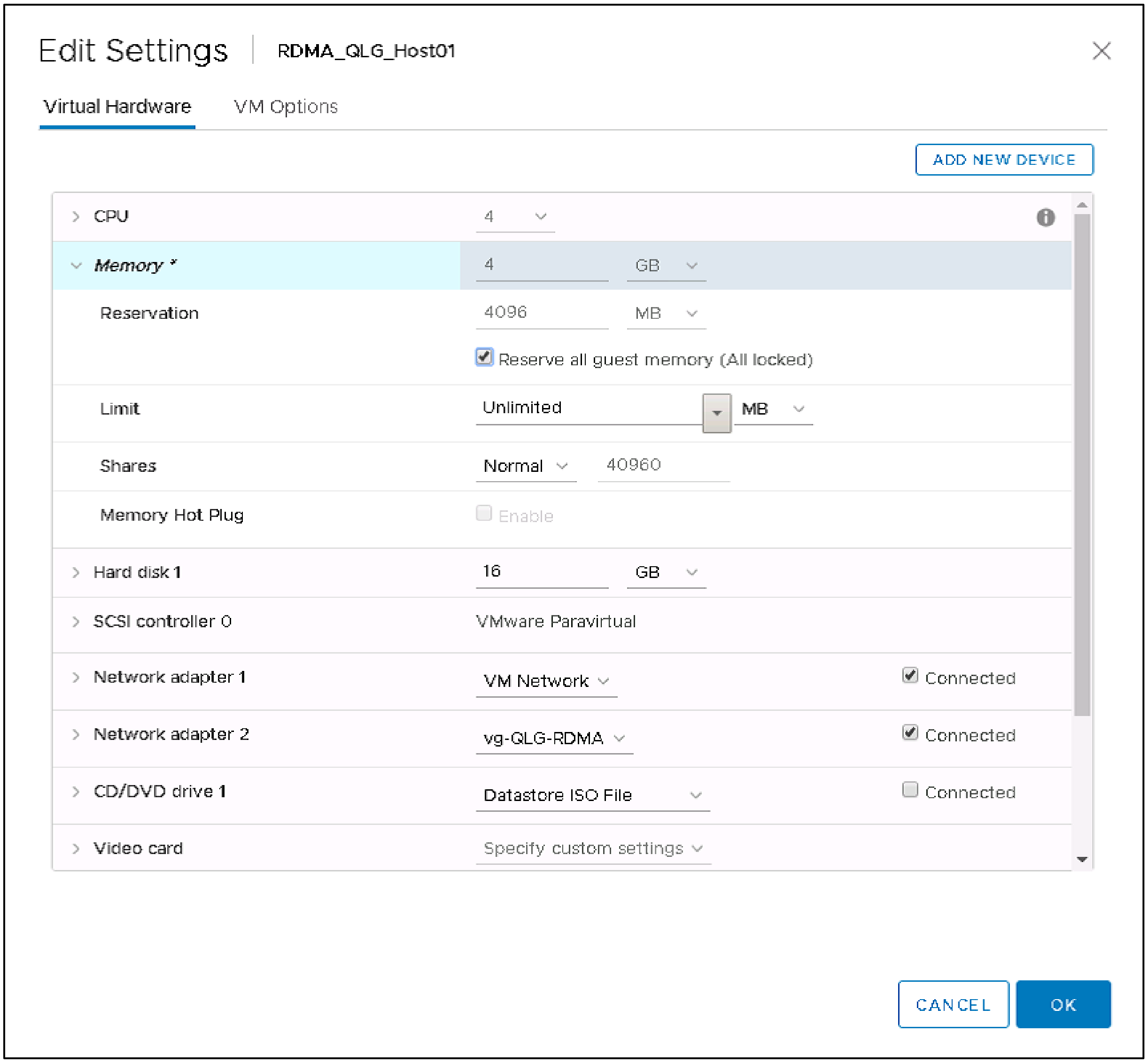
Reservation 0 Mbit/s

Limit Unlimited Mbit/s

MAC Address Automatic

Figure 8 Change the Adapter Type to PVRDMA

f. Expand the **New Network** * section and select the option **PVRDMA** as the **Adapter Type**.



CPU	4		
*Memory **	4	GB	
Reservation	4096	MB	
	☑ Reserve all guest memory (All locked)		
Limit	Unlimited	MB	
Shares	Normal	40960	
Memory Hot Plug	□ Enable		
Hard disk 1	16	GB	
SCSI controller 0	VMware Paravirtual		
Network adapter 1	VM Network		☑ Connected
Network adapter 2	vg-QLG-RDMA		☑ Connected
CD/DVD drive 1	Datastore ISO File		□ Connected
Video card	Specify custom settings		

Figure 9 Select the checkbox for Reserve all guest memory

g. Expand the **Memory** section and **select the checkbox** next to **Reserve all guest memory (All locked)**.

h. Click **OK** to close the window.

i. Power on the virtual machine.

2.2 Configuring PVRDMA on a guest operating system

Two virtual machines are created describing the configurations for both, the server (VM1) and the client (VM2).

2.2.1 Configuring a server virtual machine (VM1)

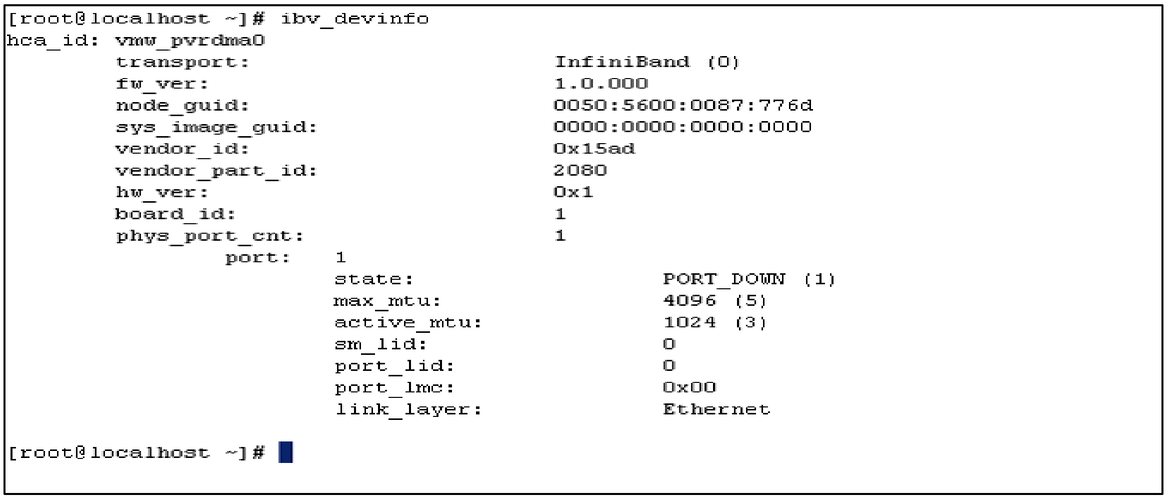
To configure PVRDMA on a guest operating system, PVRDMA driver must be installed. The installation process is contingent on the virtual machine tools, VMware ESXi version and guest operating system version.

To configure PVRDMA on a guest operating system, follow the steps:

1. Create a virtual machine and add a PVRDMA adapter over a vDS port-group from the vCenter. See Deploying PVRDMA on VMware vSphere for instructions.
2. Install the following packages:

    a. rdma-core (yum install rdma-core)
    b. infiniband-diags (yum install inifiniband-diags)
    c. perftest (yum install perftest)
    d. libibverbs-utils

3. Use the `ibv_devinfo` command to get information about InfiniBand devices available on the user-space.

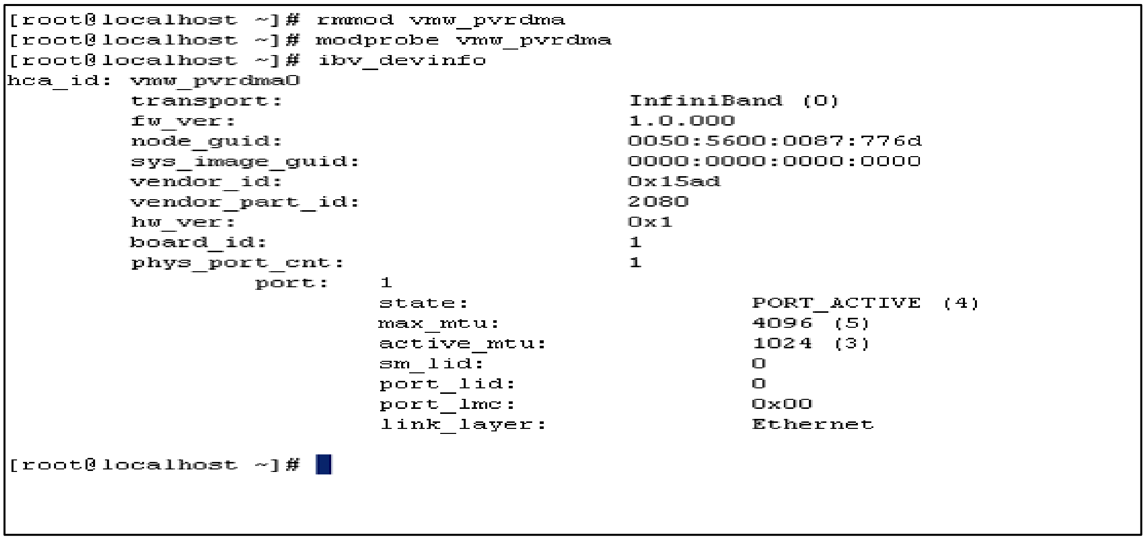


```
[root@localhost ~]# ibv_devinfo
hca_id: vmw_pvrdma0
        transport:                      InfiniBand (0)
        fw_ver:                         1.0.000
        node_guid:                      0050:5600:0087:776d
        sys_image_guid:                 0000:0000:0000:0000
        vendor_id:                      0x15ad
        vendor_part_id:                 2080
        hw_ver:                         0x1
        board_id:                       1
        phys_port_cnt:                  1
                port:   1
                        state:                  PORT_DOWN (1)
                        max_mtu:                4096 (5)
                        active_mtu:             1024 (3)
                        sm_lid:                 0
                        port_lid:               0
                        port_lmc:               0x00
                        link_layer:             Ethernet

[root@localhost ~]#
```

Figure 10 Query for the availble devices on the user-space

The query for the available devices on the user-space shows the device HCA_ID vmw_pvrdma0 listed with the transport type as InfiniBand (0). The port details show the port state is PORT_DOWN. The message indicates that the PVRDMA module must be removed from the kernel. Use the command `rmmod pvrdma` or `modprobe pvrdma` to remove the module from the kernel. The port state is now PORT_ACTIVE.



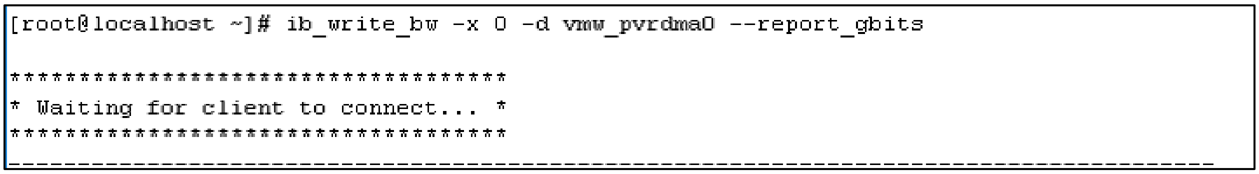
```
[root@localhost ~]# rmmod vmw_pvrdma
[root@localhost ~]# modprobe vmw_pvrdma
[root@localhost ~]# ibv_devinfo
hca_id: vmw_pvrdma0
        transport:                      InfiniBand (0)
        fw_ver:                         1.0.000
        node_guid:                      0050:5600:0087:776d
        sys_image_guid:                 0000:0000:0000:0000
        vendor_id:                      0x15ad
        vendor_part_id:                 2080
        hw_ver:                         0x1
        board_id:                       1
        phys_port_cnt:                  1
                port:   1
                        state:                  PORT_ACTIVE (4)
                        max_mtu:                4096 (5)
                        active_mtu:             1024 (3)
                        sm_lid:                 0
                        port_lid:               0
                        port_lmc:               0x00
                        link_layer:             Ethernet

[root@localhost ~]#
```

Figure 11 Bring up the Ports by removing PVRDMA from the kernel.

4. Use the query `ib_write_bw -x 0 -d vmw_pvrdma0 –report_gbits` to open the connection and wait for the client to connect.

Note: The query `ib_write_bw` is used to start a server and wait for connection. `-x` uses GID with GID index (Default: `IB - no gid . ETH - 0`). `-d` uses IB device (insert the HCA_id). `–report_gbits` Report Max/Average BW of test in Gbit/sec instead of MB/sec.



```
[root@localhost ~]# ib_write_bw -x 0 -d vmw_pvrdma0 --report_gbits

************************************
* Waiting for client to connect... *
************************************
---------------------------------------------------------------------------------------
```

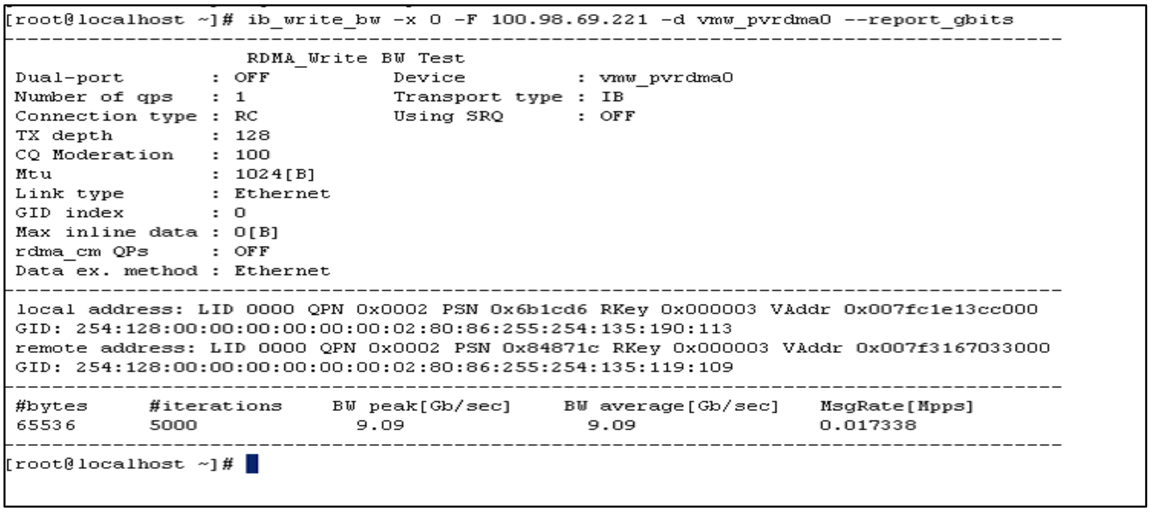
Figure 12 Open the connection from VM1

2.2.2 Configuring a client virtual machine (VM2)

Now that the connection is open from the server virtual machine and the VM is in a wait state, do the following to configure the client virtual machine:

1. Follow steps 1-3 for configuring the server virtual machine (VM1).
2. Connect the server VM and to test the connection:

```
ib_write_bw -x 0 -F <ip of VM1> -d vmw_pvrdma0 --report_gbits
```



```
[root@localhost ~]# ib_write_bw -x 0 -F 100.98.69.221 -d vmw_pvrdma0 --report_gbits
---------------------------------------------------------------------------------------
                    RDMA_Write BW Test
 Dual-port        : OFF          Device         : vmw_pvrdma0
 Number of qps    : 1            Transport type : IB
 Connection type  : RC           Using SRQ      : OFF
 TX depth         : 128
 CQ Moderation    : 100
 Mtu              : 1024[B]
 Link type        : Ethernet
 GID index        : 0
 Max inline data  : 0[B]
 rdma_cm QPs      : OFF
 Data ex. method  : Ethernet
---------------------------------------------------------------------------------------
 local address: LID 0000 QPN 0x0002 PSN 0x6b1cd6 RKey 0x000003 VAddr 0x007fc1e13cc000
 GID: 254:128:00:00:00:00:00:00:02:80:86:255:254:135:190:113
 remote address: LID 0000 QPN 0x0002 PSN 0x84871c RKey 0x000003 VAddr 0x007f3167033000
 GID: 254:128:00:00:00:00:00:00:02:80:86:255:254:135:119:109
---------------------------------------------------------------------------------------
 #bytes     #iterations    BW peak[Gb/sec]    BW average[Gb/sec]   MsgRate[Mpps]
 65536      5000             9.09               9.09               0.017338
---------------------------------------------------------------------------------------
[root@localhost ~]# 
```

Figure 13 Open the connection from VM2 and begin testing the connection between the two VMs

3 Summary

This white paper describes how to configure PVRDMA for QLogic CNA devices on VMware ESXi and how PVRDMA can be enabled on two virtual machines with Red Hat Enterprise Linux 7.6. A test was performed using perftest which helped gather reports upon data transmission over a PVRDMA configuration. For using features such as vMotion, HA, Snapshots, and DRS together with VMware vSphere, configuring PVRDMA is an optimal choice.

4 References

- Configure an ESXi Host for PVRDMA
- vSphere Networking