The Basics of Remote Direct Memory Access (RDMA) in vSphere

VMware Application Acceleration

Table of contents

The Basics of Remote Direct Memory Access (RDMA) in vSphere 3

Overview 3

Traditional Data Path vs RDMA 4

RDMA Support in vSphere 5

RMDA Support for Virtual Machines 6

How to Configure PVRDMA? 7

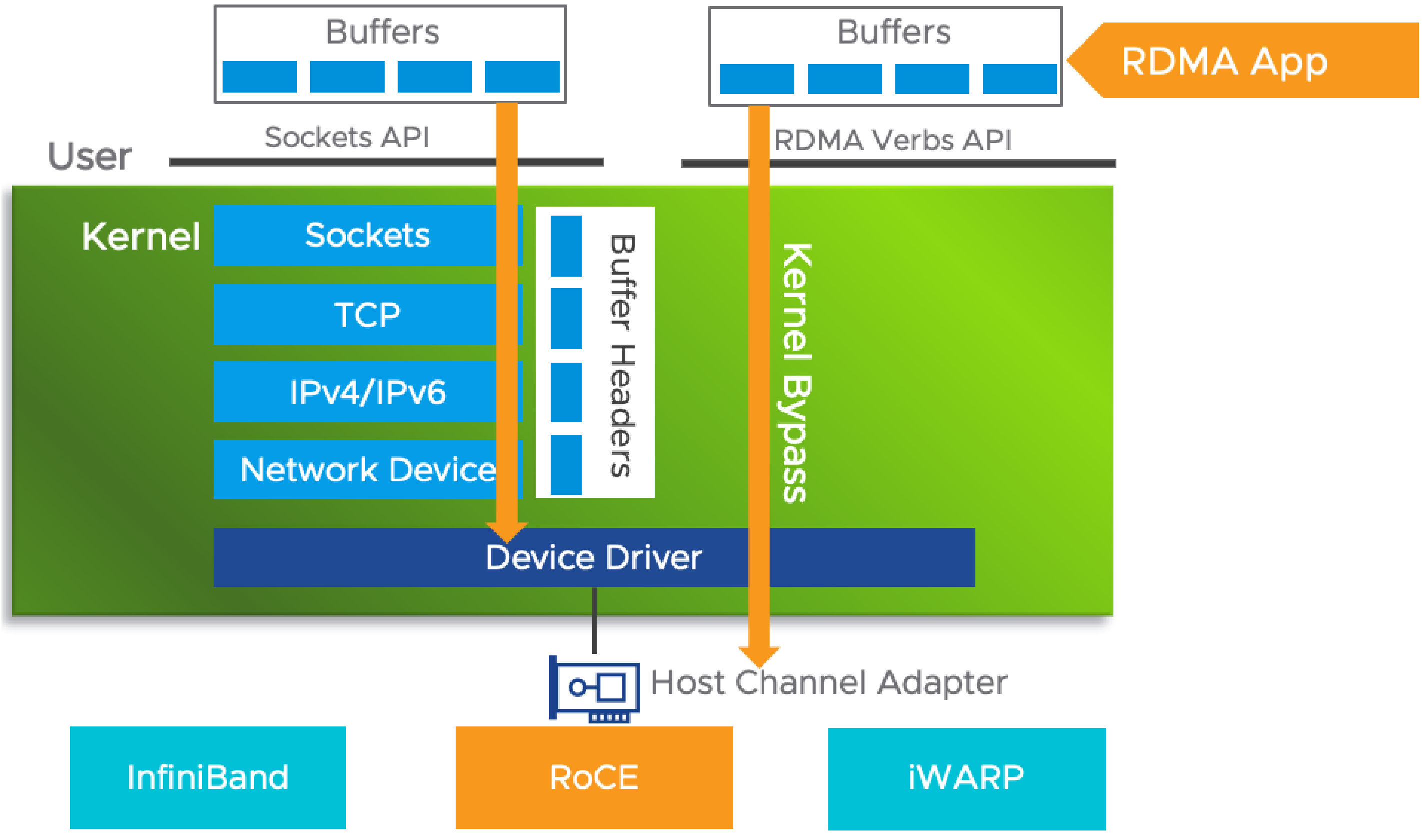
The Basics of Remote Direct Memory Access (RDMA) in vSphere

Overview

Remote Direct Memory Access (RDMA) is an extension of the Direct Memory Access (DMA) technology, which is the ability to access host memory directly without CPU intervention. RDMA allows for accessing memory data from one host to another. A key characteristic of RDMA is that it greatly improves throughput and performance while lowering latency because less CPU cycles are needed to process the network packets.

Traditional Data Path vs RDMA

In a traditional network data path, an application needs to go through the buffers using the sockets API in the user space. In the kernel, the data path includes the TCP, IPv4/6 stack all the way down to the device driver and eventually the network fabric. All these steps require CPU cycles for processing. With high bandwidth networks today (25,40,50, and 100GbE) this can pose a challenge because of the amount of CPU time required to process data and put that data on the wire. That is where RDMA comes into play.

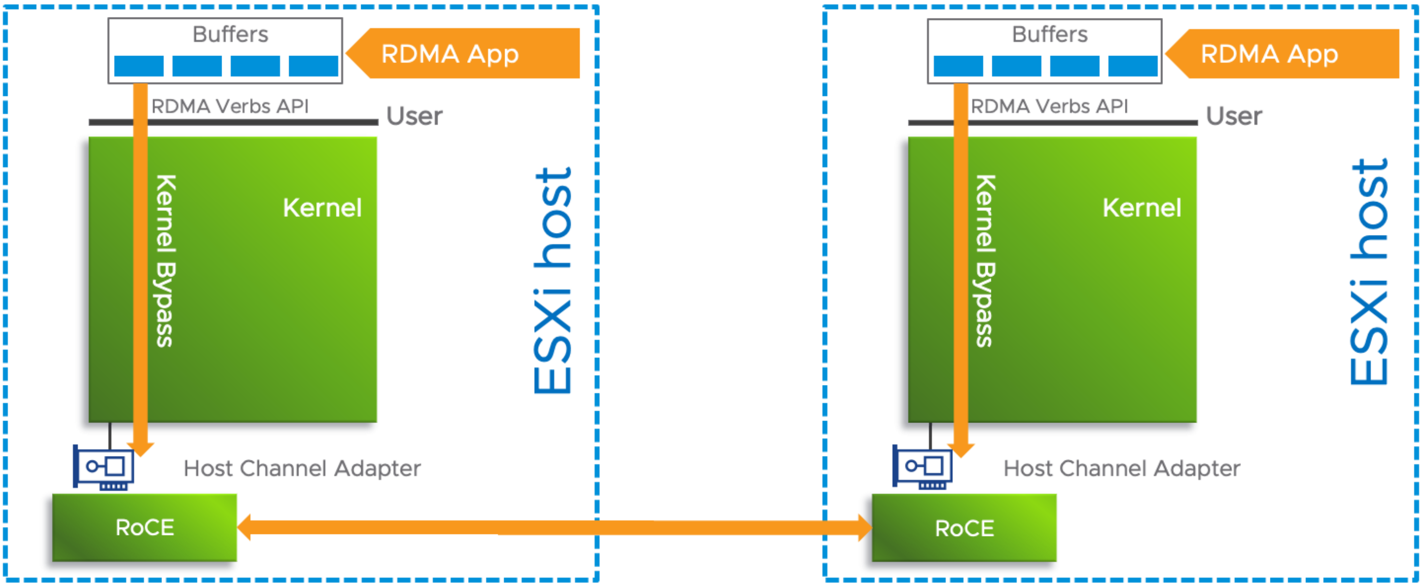


When the so-called communication path between the source and destination is set up, the kernel is by-passed. By doing so, the network latency is lowered while data throughput is increased because there's a lot less CPU cycles involved. The RDMA application speaks to the Host Channel Adapter (HCA) direclty using the RDMA Verbs API. You can see a HCA as a RDMA capable Network Interface Card (NIC). To transport RDMA over a network fabric, InfiniBand, RDMA over Converged Ethernet (RoCE), and iWARP are supported.

RDMA Support in vSphere

RDMA is supported with vSphere. The same logic applies, that when two ESXi hosts communicate with eachother over a fabric, using HCAs, RDMA is possible. Good examples of RDMA usage with vSphere are various storage featues like iSCSI extentions over RDMA introduced in vSphere 6.7, or NVMeoF using RDMA in vSphere 7. vSphere Bitfusion is another solution greatly benefits from using RDMA. With the release of vSphere 7 Update 1, native endpoints (non-PVRDMA endpoints) like with storage arrays are supported with RDMA, too.

Using RDMA for virtual workloads in a vSphere environment is supported. High-Performance Computing (HPC) applications, database backends, big data platforms are just some examples of the wide variety of applications that can profit from higher I/O rates against lower latency with RDMA.



RMDA Support for Virtual Machines

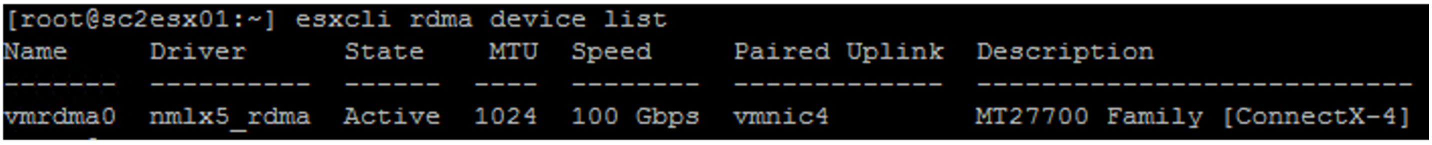
There's multiple options to expose RDMA to virtual machines (VM). The first option is to use (Dynamic) DirectPath I/O to passthrough a HCA, or a RDMA capable NIC, to the VM. Using passthrough creates a 1-to-1 relationship between the VM and the RDMA network device. The downside here is that this vSphere features like vMotion are not supported with passthrough.

There's a second option that supports vMotion to keep workload portability, by using Paravirtual RDMA (PVRDMA), also referred to as vRDMA. PVRDMA is available since VM hardware version 13, introduced with vSphere 6.5, for a x64 Linux guest OS.

RDMA over Converged Ethernet (RoCE) is supported with PVRDMA. The beauty of RoCE is that a ethernet fabric can be used. There's not explicit requirement for a seperate fabric like with Infiniband. The ethernet fabric needs to support RDMA, primarily by supporting Priority Flow Control (PFC). PVRDMA supports both RoCE v1 and v2. The difference here is that RoCE v1 supports switched networks only, where RoCE v2 supports routed networks.

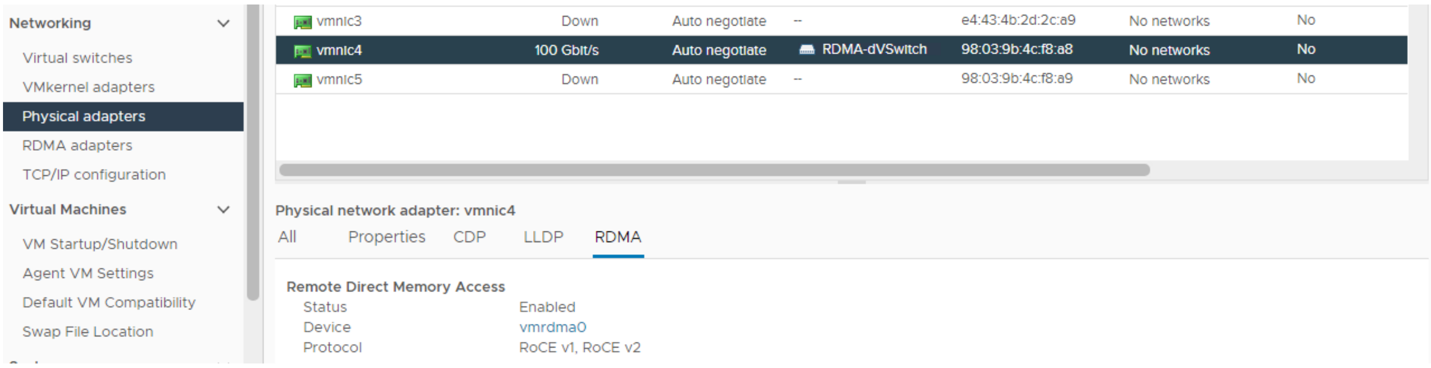
How to Configure PVRDMA?

To configure PVRDMA for a VM, the ethernet fabric should support RDMA, and a HCA is required per host. The following esxcli command can be used to check for RDMA capable NICs.



```
[root@sc2esx01:~] esxcli rdma device list
Name      Driver      State    MTU   Speed     Paired Uplink  Description
-------   ----------  ------   ----  --------  -------------  -------------------------
vmrdma0   nmlx5_rdma  Active   1024  100 Gbps  vmnic4         MT27700 Family [ConnectX-4]
```

The equivelant is also found in the vSphere Client. Go to an ESXi host > Configure > Physical Adapters. All the host's network interfaces are shown with RDMA section, showing if RDMA is supported.



vmnic3	Down	Auto negotiate	--	e4:43:4b:2d:2c:a9	No networks	No
vmnic4	100 Gbit/s	Auto negotiate	RDMA-dVSwitch	98:03:9b:4c:f8:a8	No networks	No
vmnic5	Down	Auto negotiate	--	98:03:9b:4c:f8:a9	No networks	No

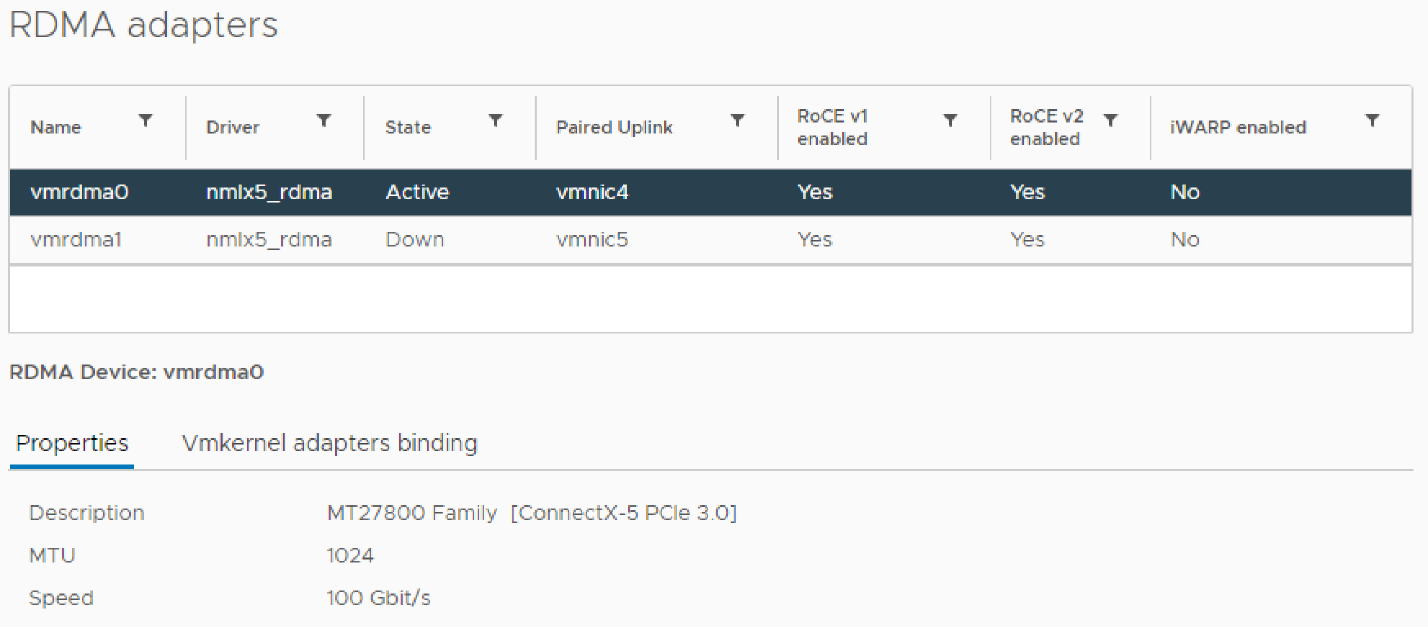
Physical network adapter: vmnic4

Remote Direct Memory Access

Status	Enabled
Device	vmrdma0
Protocol	RoCE v1, RoCE v2

> **Note**: VMs that reside on different ESXi hosts require HCA to use RDMA . You must assign the HCA as an uplink for the (Distributed) vSwitch. PVRDMA does not support NIC teaming. The HCA must be the only uplink on the vSphere (Distributed) Switch. For virtual machines on the same ESXi hosts or virtual machines using the TCP-based fallback, the HCA is not required.

If RDMA is enabled, the RDMA adapter is paired to a physical uplink and bound to a VMkernel interface.



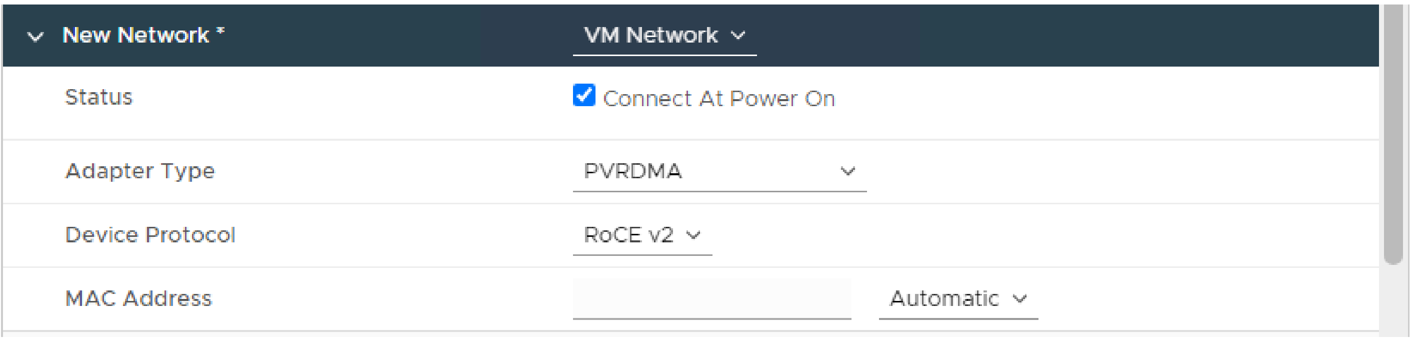
RDMA adapters

Name	Driver	State	Paired Uplink	RoCE v1 enabled	RoCE v2 enabled	iWARP enabled
vmrdma0	nmlx5_rdma	Active	vmnic4	Yes	Yes	No
vmrdma1	nmlx5_rdma	Down	vmnic5	Yes	Yes	No

RDMA Device: vmrdma0

Description	MT27800 Family [ConnectX-5 PCIe 3.0]
MTU	1024
Speed	100 Gbit/s

On the VM side, VM hardware version 13 is required. When configuring a Linux x64 guest OS, you'll be presented with the option to choose PVRDMA as the adapter type. Also notice the device protocol option for both RoCE v1 and v2.



New Network *	VM Network	
Status	☑ Connect At Power On	
Adapter Type	PVRDMA	
Device Protocol	RoCE v2	
MAC Address		Automatic

When another VM an another ESXi host is set up for PVRDMA, the applications in the PVRDMA-enabled VMs can utilize RDMA for greatly increased I/O against lower latency! More details on how to configure PVRDMA for virtual machines is found here.