Lenovo

Implementing RDMA on Linux

Describes how to implement Remote Direct Memory Access on RHEL 7

Lists what Linux RPM packages are needed

Provides the commands needed for a RoCE test and an iWARP test

Suitable for IT Specialists looking to implement RDMA

Guangzhe Fu

LENOVO PRESS

Abstract

This paper explains the steps required to set up a connection between applications using InfiniBand, Remote Direct Memory Access (RoCE) and iWARP and how to operation required to use the remote direct memory access read and write data. This paper is intended for IT administrators. Readers are expected to have network deployment knowledge.

At Lenovo® Press, we bring together experts to produce technical publications around topics of importance to you, providing information and best practices for using Lenovo products and solutions to solve IT challenges.

See a list of our most recent publications at the Lenovo Press web site:

http://lenovopress.com

Do you have the latest version? We update our papers from time to time, so check whether you have the latest version of this document by clicking the **Check for Updates** button on the front page of the PDF. Pressing this button will take you to a web page that will tell you if you are reading the latest version of the document and give you a link to the latest if needed. While you're there, you can also sign up to get notified via email whenever we make an update.

Contents

Introduction 3
Configuring RDMA on RHEL 7.x 3
RoCE test 5
iWARP evaluation 9
Omni-Path Architecture 11
For more information 12
Author 12
Notices 13
Trademarks 14

Introduction

Remote Direct Memory Access (RDMA) is a direct memory access method that allows a user to access the memory of one server from the memory of another server, without involving the operating system of either server.

There are 3 kinds of RDMA technology:

- InfiniBand
- RDMA over Converged Ethernet (RoCE)
- iWARP

InfiniBand refers to two distinctly different concepts:

- A physical link-layer protocol for InfiniBand networks
- A higher-level programming API called the InfiniBand Verbs API

The InfiniBand Verbs API is an implementation of a remote direct memory access (RDMA) technology. RDMA over Converged Ethernet (RoCE) is a network protocol that allows remote direct memory access (RDMA) over an Ethernet network.

iWARP is a computer networking protocol that implements RDMA for efficient data transfer over Internet Protocol networks. Because iWARP is layered on IETF-standard congestion-aware protocols such as TCP and SCTP, it makes few demands on the network, and can be successfully deployed in a broad range of environments.

Configuring RDMA on RHEL 7.x

In this section, we describe how to enable the RDMA feature with Linux Inbox driver. We are using RHEL 7.x for the examples.

Package installation

Installation is as follows:

1. Install the rdma package and enable rdma service using the following commands:

   ```
   yum install rdma
   dracut -f
   systemctl enable rdma
   ```

   The output of these commands is shown in Figure 1 on page 4.

```
[root@localhost ~]# yum install rdma
Loaded plugins: langpacks, product-id, search-disabled-repos, subscription-manager
rhel-7-server-optional-fastrack-rpms                      | 2.1 kB  00:00:00
rhel-7-server-optional-rpms                             | 2.0 kB  00:00:00
rhel-7-server-rpms                                    | 2.0 kB  00:00:00
Resolving Dependencies
--> Running transaction check
---> Package rdma.noarch 0:7.3_4.7_rc2-5.el7 will be installed
Removing rdma.noarch 0:7.3_4.7_rc2-5.el7 - u due to obsoletes from installed
rdma-core-13-7.el7.x86_64
--> Restarting Dependency Resolution with new changes.
--> Running transaction check
---> Package rdma.noarch 0:7.3_4.7_rc2-5.el7 will be installed
--> Finished Dependency Resolution
* dracut –f
[root@localhost ~]# dracut -f
[root@localhost ~]#
* systemctl enable rdma
[root@localhost ~]# systemctl enable rdma
[root@localhost ~]#
```

Figure 1 Installing and enabling RDMA

Editing the configuration files

The following configurations need to be modified after the installation is complete:

- /etc/rdma/rdma.conf
- /etc/udev.d/rules.d/70-persistent-ipoib.rules
- /etc/security/limits.d/rdma.conf

The rdma service reads /etc/rdma/rdma.conf to find out which kernel-level and user-level RDMA protocols the administrator wants to be loaded by default. You should edit this file to turn various drivers on or off.

The rdma package provides the file /etc/udev.d/rules.d/70-persistent-ipoib.rules. This udev rules file is used to rename IPoIB devices from their default names (such as ib0 and ib1) to more descriptive names. You should edit this file to change how your devices are named.

RDMA communications require that physical memory in the computer be pinned (meaning that the kernel is not allowed to swap that memory out to a paging file in the event that the overall computer starts running short on available memory). Pinning memory is normally a very privileged operation. In order to allow users other than root to run large RDMA applications, it will likely be necessary to increase the amount of memory that non-root users are allowed to pin in the system. This is done by adding the file rdma.conf file in the /etc/security/limits.d/directory with contents such as shown in Figure 2.

Figure 2 Contents of rdma.conf

```
[root@localhost ~]# more /etc/security/limits.d/rdma.conf
# configuration for rdma tuning
***     soft    memlock         unlimited
***     hard    memlock         unlimited
# rdma tuning end
```

RoCE test

This section, we describe how to use RoCE. We will be using the Emulex OCe14102 adapter as an example.

We used the following test environment:

- ThinkServer® RD650
- Emulex OneConnect OCe14102-UM and Emulex OneConnect OCe14102-NM adapters
- Red Hat Enterprise Linux RHEL7.2
- Emulex drivers, Version 3.10.0-327.36.3.el7.x86 64 and 3.10.0-327.el7.x86 64

Procedure

Because RDMA applications are so different from Berkeley Sockets-based applications and from normal IP networking, most applications that are used on an IP network cannot be used directly on an RDMA network. RHEL 7 comes with a number of different software packages for RDMA network administration, testing and debugging, high level software development APIs, and performance analysis.

In order to utilize these networks, some or all of the following packages need to be installed (this list is not exhaustive, but does cover the most important packages related to RDMA):

- rdma
- libocrdma
- libibverbs-utils
- perftest

1. Issue the following commands:

   ```
   yum install rdma libocrdma libibverbs-utils perftest
   systemctl start rdma
   systemctl enable rdma
   ```

   Output of these commands is shown in Figure 3.

   Figure 3 Output from commands

   ```
   [root@localhost ~]# yum install rdma libocrdma libibverbs-utils perftest
   Loaded plugins: langpacks, product-id, search-disabled-repos, subscription-manager
   rhel-7-server-optional-fastrack-rpms                      | 2.1 kB  00:00:00
   rhel-7-server-optional-rpms                              | 2.0 kB  00:00:00
   rhel-7-server-rpms                                      | 2.0 kB  00:00:00
   Package matching libibverbs-utils-1.2.1-1.el7.x86_64 already installed. Checking for
   update.
   Package matching perftest-3.0-7.el7.x86_64 already installed. Checking for update.
   Resolving Dependencies
   --> Running transaction check
   ---> Package libocrdma.x86_64 0:1.0.8-1.el7 will be installed
   ---> Package rdma.noarch 0:7.3_4.7_rc2-5.el7 will be installed
   Removing libocrdma.x86_64 0:1.0.8-1.el7 - u due to obsoletes from installed
   libibverbs-13-7.el7.x86_64
   Removing rdma.noarch 0:7.3_4.7_rc2-5.el7 - u due to obsoletes from installed
   rdma-core-13-7.el7.x86_64
   --> Restarting Dependency Resolution with new changes.
   --> Running transaction check
   ---> Package libocrdma.x86_64 0:1.0.8-1.el7 will be installed
   ---> Package rdma.noarch 0:7.3_4.7_rc2-5.el7 will be installed
   ```

```
--> Finished Dependency Resolution
* systemctl start rdma
[root@localhost ~]# systemctl start rdma
[root@localhost ~]#
* systemctl enable rdma
[root@localhost ~]# systemctl enable rdma
[root@localhost ~]#
```

2. Add network setting in /etc/sysconfig/network-scripts/ifcfg-suffix files as shown in Figure 4.

Figure 4   Additional settings in /etc/sysconfig/network-scripts/ifcfg-suffix

```
* Config OCe14102 port
DEVICE = ens1f0
TYPE = Ethernet
BOOTPROTO = none
ONBOOT = yes
* Config VLAN
DEVICE = ens1f0 .8
BOOTPROTO = none
ONBOOT = yes
IPADDR =192.168.8.50
PREFIX =24
VLAN = yes
```

3. Restart the network service to enable the settings using the command:

```
systemctl restart network
```

Test results

To confirm the changes, review port information using tools such as **ibv_devinfo** and **ib_write_bw**.

Port information

We use the **ibv_devinfo** command to display InfiniBand device information, which is configured according to our requirement. The output is shown in Figure 5.

Figure 5   Output from ibv_devinfo command

```
[root@xxxxx network-scripts]# ibv_devinfo
hca_id: ocrdma1
        transport:                      InfiniBand (0)
        fw_ver:                         10.6.228.36
        node_guid:                      0290:faff:fe30:9ade
        sys_image_guid:                 0290:faff:fe30:9ade
        vendor_id:                      0x10df
        vendor_part_id:                 1824
        hw_ver:                         0x410
        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
```

```
hca_id: ocrdma0
        transport:                      InfiniBand (0)
        fw_ver:                         10.6.228.36
        node_guid:                      0290:faff:fe30:9ad6
        sys_image_guid:                 0290:faff:fe30:9ad6
        vendor_id:                      0x10df
        vendor_part_id:                 1824
        hw_ver:                         0x410
        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
```

Performance

The following example shows how to run a diagnostic between a local node client and a remote node server with the **ib_write_bw** command.

First configure the remote node server with the command:

```
ib_write_bw -d ocrdma0 -b -F -D 30 --cpu_util
```

The important parameters have the following meaning:

-d ocrdma0 Uses the InfiniBand device ocrdma0.

-b Measure bidirectional bandwidth (default unidirectional).

-F Do not show a warning even if cpufreq_ondemand module is loaded, and cpu-freq is not on max.

-D 30 Run test period is 30 seconds.

--cpu_util Show CPU Utilization in report, valid only in Duration mode.

The command and output are shown in Figure 6.

Figure 6 Output from ib_write_bw command

```
[root@xxxxx network-scripts]# ib_write_bw -d ocrdma0 -b -F -D 30 --cpu_util

************************************
* Waiting for client to connect... *
************************************
----------------------------------------------------------------------------------------
                    RDMA_Write Bidirectional BW Test
 Dual-port       : OFF          Device         : ocrdma0
 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 0x000c PSN 0xd7785c RKey 0x81fff28 VAddr 0x007f14cd279000
 GID: 00:00:00:00:00:00:00:00:00:00:255:255:192:168:08:50
 remote address: LID 0000 QPN 0x000c PSN 0x28560b RKey 0x81fff28 VAddr 0x007f924052a000
 GID: 00:00:00:00:00:00:00:00:00:00:255:255:192:168:08:60
---------------------------------------------------------------------------------------
 #bytes     #iterations    BW peak[MB/sec]    BW average[MB/sec]   MsgRate[Mpps]
CPU_Util[%]
Conflicting CPU frequency values detected: 3698.925000 != 2938.359000
Test integrity may be harmed !
Warning: measured timestamp frequency 3491.78 differs from nominal 3698.93 MHz
 65536      278000           0.00               2171.83            0.034749         12.50
---------------------------------------------------------------------------------------
```

Then run the command on the local node client the following command as shown in Figure 7:

```
ib_write_bw -d ocrdma0 -b -F -D 30 --cpu_util server-IP-address
```

In the command substitute the IP address with the IP address of your server.

Figure 7 Output from ib_write_bw command

```
[root@localhost ~]# ib_write_bw -d ocrdma0 -b -F -D 30 --cpu_util 192.168.8.50
---------------------------------------------------------------------------------------
                    RDMA_Write Bidirectional BW Test
 Dual-port       : OFF          Device         : ocrdma0
 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 0x000c PSN 0x28560b RKey 0x81fff28 VAddr 0x007f924052a000
 GID: 00:00:00:00:00:00:00:00:00:00:255:255:192:168:08:60
 remote address: LID 0000 QPN 0x000c PSN 0xd7785c RKey 0x81fff28 VAddr 0x007f14cd279000
 GID: 00:00:00:00:00:00:00:00:00:00:255:255:192:168:08:50
---------------------------------------------------------------------------------------
 #bytes     #iterations    BW peak[MB/sec]    BW average[MB/sec]   MsgRate[Mpps]
CPU_Util[%]
Conflicting CPU frequency values detected: 3600.578000 != 3524.300000
Test integrity may be harmed !
Warning: measured timestamp frequency 2294.63 differs from nominal 3600.58 MHz
 65536      278000           0.00               2171.83            0.034749         2.78
---------------------------------------------------------------------------------------
```

The above output confirms that communication between the client and server has been successfully established. Confirm that the output contains these entries:

```
Device         : ocrdma0
Transport type : IB
```

The parameter `-d ocrdma0` means that you are using the InfiniBand device ocrdma0 to establish network session between client and server.

Tip: If communication is not working correctly, you will get error messages as the output.

iWARP evaluation

In this section, we describe how to enable iWARP. We use the Intel X722 Ethernet Controller as an example.

We used the following environment for this test:

- ThinkSystem™ SR630 and ThinkSystem SR650
- 10GBASE-T LAN on Motherboard (LOM) adapter (rev 03)
- Red Hat Enterprise Linux 7.3
- Intel driver, Version 3.10.0-514.el7.x86_64

Procedure

The steps to enable iWARP are as follows:

1. Issue the three commands as described in step 1 on page 5 in the RoCE test. That is, configure the basic space for RMDA, using the following commands:

   ```
   yum install rdma libocrdma libibverbs-utils perftest
   systemctl start rdma
   systemctl enable rdma
   ```

2. Load the i40iw driver for Intel X722 network controller using the following command:

   ```
   modprobe i40iw
   ```

3. Add the following network settings to the /etc/sysconfig/network-scripts/ifcfg-suffix file:

   ```
   * Config OCe14102 port
   DEVICE = ens1f0
   TYPE = Ethernet
   BOOTPROTO = none
   ONBOOT = yes
   * Config VLAN
   DEVICE = ens1f0 .8
   BOOTPROTO = none
   ONBOOT = yes
   IPADDR =192.168.8.1
   PREFIX =24
   VLAN = yes
   ```

   Figure 8 Additions to /etc/sysconfig/network-scripts/ifcfg-suffix

4. Restart the network service to enable setting using the following command:

   ```
   systemctl restart network
   ```

Test Results

We will use **`ib_write_bw`** for performance testing. As you will see, iWARP has a greater CPU usage than RoCE.

To configure the remote node server with the command:

```
ib_write_bw -d i40iw3 -b -F -D 30 --cpu_util --rdma_cm
```

The parameters have the following meaning:

`-d i40iw3` Uses the IWARP device i40iw3.

`-b` Measure bidirectional bandwidth (default unidirectional).

`-F` Do not show a warning even if cpufreq_ondemand module is loaded, and cpu-freq is not on max.

`-D 30` Run test period is 30 seconds.

`--cpu_util` Show CPU Utilization in report, valid only in Duration mode.

`--rdma_cm` Connect QPs with rdma_cm and run test on those QPs.

The command and output are shown in Figure 9.

Figure 9 Output from ib_write_bw command

```
# ib_write_bw -d i40iw3 -b -F -D 30 --cpu_util --rdma_cm

************************************
* Waiting for client to connect... *
************************************
---------------------------------------------------------------------------------------
                    RDMA_Write Bidirectional BW Test
 Dual-port       : OFF          Device         : i40iw3
 Number of qps   : 1            Transport type : IW
 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     : ON
 Data ex. method : rdma_cm
---------------------------------------------------------------------------------------
 Waiting for client rdma_cm QP to connect
 Please run the same command with the IB/RoCE interface IP
---------------------------------------------------------------------------------------
 local address: LID 0x01 QPN 0x0004 PSN 0x3659e1
 GID: 124:211:10:178:10:88:00:00:00:00:00:00:00:00:00:00
 remote address: LID 0x01 QPN 0x0004 PSN 0xb7e62b
 GID: 124:211:10:198:162:104:00:00:00:00:00:00:00:00:00:00
---------------------------------------------------------------------------------------
 #bytes     #iterations    BW peak[MB/sec]    BW average[MB/sec]   MsgRate[Mpps]
CPU_Util[%]
 65536      285600           0.00               2236.49            0.035784        12.50
---------------------------------------------------------------------------------------
```

Next, we run the command on the local node client as shown in Figure 10:

```
ib_write_bw -d i40iw3 -b -F -D 30 --cpu_util 192.168.8.1 --rdma_cm
```

The only difference between this and the previous command is that here the command contains the server IP address of the server.

Figure 10 Output from ib_write_bw command

```
# ib_write_bw -d i40iw3 -b -F -D 30 --cpu_util 192.168.8.1 --rdma_cm
---------------------------------------------------------------------------------------
                    RDMA_Write Bidirectional BW Test
 Dual-port       : OFF          Device         : i40iw3
 Number of qps   : 1            Transport type : IW
 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     : ON
 Data ex. method : rdma_cm
---------------------------------------------------------------------------------------
 local address: LID 0x01 QPN 0x0004 PSN 0xb7e62b
 GID: 124:211:10:198:162:104:00:00:00:00:00:00:00:00:00:00
 remote address: LID 0x01 QPN 0x0004 PSN 0x3659e1
 GID: 124:211:10:178:10:88:00:00:00:00:00:00:00:00:00:00
---------------------------------------------------------------------------------------
 #bytes     #iterations    BW peak[MB/sec]    BW average[MB/sec]   MsgRate[Mpps]
CPU_Util[%]
 65536      285600           0.00               2236.49            0.035784         6.25
---------------------------------------------------------------------------------------
```

To verify successful communication, confirm the output contains the following entries:

```
Device         : i40iw3
Transport type : IW
```

The parameter `-d i40iw3` means that you are using the IWARP device i40iw3 to establish network session between client and server.

If the communication has failed, you will instead get an error on the output.

Omni-Path Architecture

Red Hat's official documentation suggests the use of the out-of-box Intel OPA driver, as described in the following web page:

https://access.redhat.com/articles/2039623

Please refer to Intel's documentation to install Omni-Path drivers and tools.

For more information

For more information on RDMA, see the following Red Hat documentation page:

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/networking_guide/sec-configuring_the_base_rdma_subsystem

Author

Guangzhe Fu is a Linux Engineer in the Lenovo Data Center Group in Beijing, China. He joined the OS team in 2017. His major focus is the Network and Virtualization feature of Linux kernel development in Lenovo. He has two years experience as a Software Architecture engineer, six years experience as a Linux Kernel Development engineer.

Thanks to the following people for their contributions to this project:

- David Watts, Lenovo Press
- Siyuan Wang Lenovo Linux Engineer

Notices

Lenovo may not offer the products, services, or features discussed in this document in all countries. Consult your local Lenovo representative for information on the products and services currently available in your area. Any reference to a Lenovo product, program, or service is not intended to state or imply that only that Lenovo product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any Lenovo intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any other product, program, or service.

Lenovo may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

Lenovo (United States), Inc.
1009 Think Place - Building One
Morrisville, NC 27560
U.S.A.
Attention: Lenovo Director of Licensing

LENOVO PROVIDES THIS PUBLICATION “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. Lenovo may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

The products described in this document are not intended for use in implantation or other life support applications where malfunction may result in injury or death to persons. The information contained in this document does not affect or change Lenovo product specifications or warranties. Nothing in this document shall operate as an express or implied license or indemnity under the intellectual property rights of Lenovo or third parties. All information contained in this document was obtained in specific environments and is presented as an illustration. The result obtained in other operating environments may vary.

Lenovo may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any references in this publication to non-Lenovo Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this Lenovo product, and use of those Web sites is at your own risk.

Any performance data contained herein was determined in a controlled environment. Therefore, the result obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

This document was created or updated on January 11, 2018.

Send us your comments via the **Rate & Provide Feedback** form found at
http://lenovopress.com/lp0823

Trademarks