Infiniband RDMA native setup for Linux

20.10.23

Pegasi Knowledge
https://ghost.pegasi.fi/wiki/

Table of Contents

Infiniband RDMA native setup for Linux

Update: Rocky Linux / Almalinux / RHEL 9 added

Overview

Native Infiniband RDMA enables lossless storage traffic, low CPU loads, high speeds and a very low latency. We decided to with 40Gbps native Infiniband with our new NVMe based storage backend. For software solution we use Linstor that supports native Infiniband RDMA and gives us flexibility.

Here is a quick sheet on how to get native Infiniband up and running with Rocky Linux 9 / Almalinux 9 / RedHat Enterprise Linux 9. Works as well with RHEL version 8 derivatives.

Interfaces

We have ConnectX-3 cards and two Mellanox 56Gbps switches where one will be a stand-by and other will be in production. We have connected cables our two storage backend nodes and one of our front end nodes. The rest are still operating in the legacy storage and will be upgraded once the virtual guests have been migrated to the new storage.

Setup

Here are the tasks required to set up the native Infiniband environment. Do this in all storage servers. One server needs to be primary and it needs to hold the highest PRIORITY flag (look below).

- Rocky Linux 9 minimal install
- dnf in vim rdma-core libibverbs-utils librdmacm librdmacm-utils ibacm infiniband-diags opensm
- dnf up
- reboot
- rdma link show
- ip link show ibs6
- ip link show ibs6d1
- vim /etc/security/limits.d/rdma.conf. Add lines:

```
@rdma    soft    memlock     unlimited
@rdma    hard    memlock     unlimited
```

- ibstat
    - write down the mlx* port GUIDs
- do not touch /etc/rdma/opensm.conf

- vim /etc/sysconfig/opensm. Modify following and replace the GUIDS with the ones you wrote down (to master server PRIORITY 15, others below that):

```
GUIDS="0xXXXXXXXXXXXXXX 0xXXXXXXXXXXXXXX"
PRIORITY=15
```

- vim /etc/rdma/partitions.conf, add your native partiton definition as follows

```
DataVault_A=0x0002,rate=7,mtu=5,scope=2,defmember=full:ALL=full;
```

- systemctl enable opensm
- reboot

IP setup for Infiniband

Originally I did not want to do this but since iSER seems to outperform SRP then why not try it.

Let's use nm-cli and setup our interfaces with commands. First check that you have the infiniband connections there.

```
nmcli con show
```

If you for some strange reason do not have ib0/ib1 devices set up automatically you can add one with this command

```
nmcli connection add type infiniband con-name ibs6 ifname ibs6 transport-
mode Connected mtu 65520
```

Otherwise you can do

```
nmcli connection modify ibs6 transport-mode Connected
nmcli connection modify ibs6 mtu 65520
nmcli connection modify ibs6 ipv4.addresses '10.0.0.1/24'
nmcli connection modify ibs6 ipv4.method manual
nmcli connection modify ibs6 connection.autoconnect yes
nmcli connection up ibs6
```

I skipped gateway / dns setups since I do not need them in a storage network.

iSER Target setup

This is a very compact list of commands and example terminal output

```
dnf in targetcli
```

```
systemctl enable target --now
firewall-cmd --permanent --add-port=3260/tcp
firewall-cmd --reload
targetcli
/> ls
o- /
.....................................................................
..................................... [...]
  o- backstores
.....................................................................
............................... [...]
  | o- block
.....................................................................
.................... [Storage Objects: 0]
  | o- fileio
.....................................................................
.................... [Storage Objects: 0]
  | o- pscsi
.....................................................................
.................... [Storage Objects: 0]
  | o- ramdisk
.....................................................................
.................. [Storage Objects: 0]
  o- iscsi
.....................................................................
............................. [Targets: 0]
  o- loopback
.....................................................................
........................... [Targets: 0]
  o- srpt
.....................................................................
............................. [Targets: 0]
/> iscsi/
/iscsi> create
/iscsi> create iqn.2021-01-01.com.domain:abc01
Created target iqn.2021-01-01.com.domain:abc01.
Created TPG 1.
Global pref auto_add_default_portal=true
Created default portal listening on all IPs (0.0.0.0), port 3260.
/iscsi> ls
o- iscsi
.....................................................................
............................... [Targets: 1]
  o- iqn.2021-01-01.com.domain:abc1
.....................................................................
........ [TPGs: 1]
    o- tpg1
```

```
.....................................................................
..................... [no-gen-acls, no-auth]
      o- acls
.....................................................................
.............................. [ACLs: 0]
      o- luns
.....................................................................
.............................. [LUNs: 0]
      o- portals
.....................................................................
........................... [Portals: 1]
        o- 0.0.0.0:3260
.....................................................................
............................ [OK]
/iscsi> /backstores/block
/backstores/block> create name=dv1_nvme0n1 dev=/dev/nvme0n1
Created block storage object dv1_nvme0n1 using /dev/nvme0n1.
/backstores/block> create name=dv1_nvme1n1 dev=/dev/nvme1n1
Created block storage object dv1_nvme1n1 using /dev/nvme1n1.
/backstores/block> create name=dv1_nvme2n1 dev=/dev/nvme2n1
Created block storage object dv1_nvme2n1 using /dev/nvme2n1.
/backstores/block> ls
o- block
.....................................................................
........................... [Storage Objects: 3]
  o- dv1_nvme0n1
.....................................................................
[/dev/nvme0n1 (1.5TiB) write-thru deactivated]
  | o- alua
.....................................................................
........................... [ALUA Groups: 1]
  |   o- default_tg_pt_gp
.....................................................................
[ALUA state: Active/optimized]
  o- dv1_nvme1n1
.....................................................................
[/dev/nvme1n1 (1.5TiB) write-thru deactivated]
  | o- alua
.....................................................................
........................... [ALUA Groups: 1]
  |   o- default_tg_pt_gp
.....................................................................
[ALUA state: Active/optimized]
  o- dv1_nvme2n1
.....................................................................
[/dev/nvme2n1 (1.5TiB) write-thru deactivated]
    o- alua
```

```
.....................................................................
.......................... [ALUA Groups: 1]
      o- default_tg_pt_gp
.....................................................................
[ALUA state: Active/optimized]
/backstores/block> /iscsi/iqn.2021-01-01.com.domain:abc1/tpg1/portals delete
0.0.0.0 3260
Deleted network portal 0.0.0.0:3260
/backstores/block> /iscsi/iqn.2021-01-01.com.domain:abc1/tpg1/portals create
192.168.222.1 3260
Using default IP port 3260
Created network portal 192.168.222.1:3260.
/backstores/block>
/iscsi/iqn.2021-01-01.com.domain:abc1/tpg1/portals/192.168.222.1:3260
enable_iser boolean=true
iSER enable now: True
/backstores/block> /
/> ls
o- /
.....................................................................
.......................................... [...]
  o- backstores
.....................................................................
................................... [...]
  | o- block
.....................................................................
...................... [Storage Objects: 3]
  | | o- dv1_nvme0n1
.....................................................................
[/dev/nvme0n1 (1.5TiB) write-thru deactivated]
  | | | o- alua
.....................................................................
........................ [ALUA Groups: 1]
  | | |   o- default_tg_pt_gp
.....................................................................
[ALUA state: Active/optimized]
  | | o- dv1_nvme1n1
.....................................................................
[/dev/nvme1n1 (1.5TiB) write-thru deactivated]
  | | | o- alua
.....................................................................
........................ [ALUA Groups: 1]
  | | |   o- default_tg_pt_gp
.....................................................................
[ALUA state: Active/optimized]
  | | o- dv1_nvme2n1
.....................................................................
```

```
[/dev/nvme2n1 (1.5TiB) write-thru deactivated]
  | |   o- alua
.....................................................................
..................... [ALUA Groups: 1]
  | |     o- default_tg_pt_gp
.....................................................................
[ALUA state: Active/optimized]
  | o- fileio
.....................................................................
.................... [Storage Objects: 0]
  | o- pscsi
.....................................................................
.................... [Storage Objects: 0]
  | o- ramdisk
.....................................................................
.................... [Storage Objects: 0]
  o- iscsi
.....................................................................
.............................. [Targets: 1]
  | o- iqn.2021-01-01.com.domain:abc1
.....................................................................
...... [TPGs: 1]
  |   o- tpg1
.....................................................................
................... [no-gen-acls, no-auth]
  |     o- acls
.....................................................................
............................ [ACLs: 0]
  |     o- luns
.....................................................................
............................ [LUNs: 0]
  |     o- portals
.....................................................................
....................... [Portals: 1]
  |       o- 192.168.222.1:3260
.....................................................................
..................... [iser]
  o- loopback
.....................................................................
............................ [Targets: 0]
  o- srpt
.....................................................................
.............................. [Targets: 0]
/> iscsi/iqn.2021-01-01.com.domain/tpg1/luns
/> create /backstores/block/dv1_nvme0n1
/> create /backstores/block/dv1_nvme1n1
/> create /backstores/block/dv1_nvme2n1
```

```
/> iscsi/iqn.2021-01-01.com.domain:abc1/tpg1/acls create iqn.2021-07.host-
z.domain.com
/> iscsi/iqn.2021-01-01.com.domain:abc1/tpg1/acls create iqn.2021-07.host-
x.domain.com
/> iscsi/iqn.2021-01-01.com.domain:abc1/tpg1/acls create iqn.2021-07.host-
y.domain.com
/> saveconfig
```

Look the initiator names from the frontends

Add new initiator to target ACLs

When adding new initiator you must give it permission to use iSer.

```
targetcli
/iscsi/iqn.2021-01-01.com.domain:abc1/tpg1/acls create <initiator id>
```

iSER initiator setup

Set initiator name first so that the target can allow this with acl using this name.

```
vim /etc/iscsi/initiatorname.iscsi
```

Then discover and log in.

```
iscsiadm -m discovery -t st -p 192.168.222.1:3260
192.168.222.1:3260,1 iqn.2021-01-01.com.domain:abc1
iscsiadm -m node -T iqn.2021-01-01.com.domain:abc1 -o update -n
iface.transport_name -v iser
iscsiadm -m node -l
systemctl enable iscsid --now
```

NVME-of RDMA Target Setup

Firewalld

```
firewall-cmd --new-zone=nvmeof --permanent
firewall-cmd --reload
firewall-cmd --zone=nvmeof --add-source=1.2.3.4/24 --permanent
firewall-cmd --zone=nvmeof --add-source=1.2.3.5/24 --permanent
firewall-cmd --zone=nvmeof --add-port=4420/tcp --permanent
```

```
firewall-cmd --reload
```

NVME-of setup with config filesystem

```
/bin/mount -t configfs none /sys/kernel/config/ #if configfs not mounted
modprobe nvmet-rdma
echo nvmet-rdma > /etc/modules-load.d/nvme.conf
mkdir /sys/kernel/config/nvmet/subsystems/datavault01
cd /sys/kernel/config/nvmet/subsystems/datavault01
echo 1 > attr_allow_any_host
mkdir namespaces/10
cd namespaces/10
echo -n /dev/nvme0n1 > device_path
echo 1 > enable
mkdir /sys/kernel/config/nvmet/ports/1
cd /sys/kernel/config/nvmet/ports/1
echo -n <ip address> > addr_traddr #use the ib0/ib1 address that is
connected to the client
echo rdma > addr_trtype
echo 4420 > addr_trsvcid
echo ipv4 > addr_adrfam
ln -s /sys/kernel/config/nvmet/subsystems/datavault01
/sys/kernel/config/nvmet/ports/1/subsystems/datavault01
dmesg | grep "enabling port"
[1034711.759527] nvmet_rdma: enabling port 1 (<ip address>:4420)
```

Save config and enable boot time

```
nvmetcli save
systemctl enable nvmet
```

nvmet service starts too early so we must make it start at a more appropriate time. Look for systemd targets with command

```
systemctl list-units --type target
```

And look for network-online.target or similar which must be active before nvmet can kick in. Modify /usr/lib/systemd/system/nvmet.service and set “After” line to this:

```
After=sys-kernel-config.mount network.target local-fs.target NetworkManager-
wait-online.service
```

Also in my case the IB interfaces initialize very slow and I had to set a delay under [Service]

```
ExecStartPre=/bin/sleep 40
```

Then run “systemctl daemon reload” and try reboot

NVME-of RDMA target add device

```
cd /sys/kernel/config/nvmet/subsystems/datavault01
mkdir namespaces/11
cd namespaces/11
echo -n /dev/nvme1n1 > device_path
echo 1 > enable
```

NVME-of RDMA client setup

```
dnf in nvme-cli
modprobe nvme-rdma
echo "nvme-rdma" > /etc/modules-load.d/nvme.conf
nvme discover -t rdma -a <server ip address> -s 4420
nvme connect -t rdma -n testnqn -a <server ip address> -s 4420
systemctl enable nvmf-autoconnect
echo "-t rdma -a <ip address> -s 4420" > /etc/nvme/discovery.conf
echo "-t rdma -a <ip address> -s 4420" >> /etc/nvme/discovery.conf
```

Autoconnect wants to start connecting before IB interface gets IP addresses. Modify /usr/lib/systemd/system/nvmf-autoconnect.service and set a delay under [Service]:

```
ExecStartPre=/bin/sleep 40
```

Then run “systemctl daemon reload” and try reboot

Test

Test RDMA connectivity with ibping. First write down each server Port GUIDs by issuing command

```
ibstat
```

Then start ibping in server mode on each server

```
ibping -S
```

From each client run ping with the Port GUID you wrote down earlier

```
ibping -G 0xXXXXX
```

You should see very low latency pongs as a response.

Other notes

If you want to change your Mellanox ConnectX-3 or above card to Infiniband, Ethernet or autodetect mode install package mstflint and use commands:

```
lspci | grep Mellanox                          # find out the PCI address
mstconfig -d NN:NN.n q                         # query the card
mastconfig -d NN:NN.n set LINK_TYPE_P1=2       # set mode 1=ib, 2=ethernet,
3=autodetect
```

Comments

All comments and corrections are welcome.