100G NVMe over Fabrics JBOF

T6 iWARP RDMA Bandwidth, IOPS and Latency Performance

Executive Summary

NVMe over Fabrics specification extends the benefits of NVMe to large fabrics, beyond the reach and scalability of PCIe. NVMe enables deployments with hundreds or thousands of SSDs using a network interconnect, such as RDMA over Ethernet. T6 iWARP RDMA provides a low latency, high throughput, plug-and-play Ethernet solution for connecting high performance NVMe SSDs over a scalable, congestion controlled and traffic managed fabric, with no special configuration needed.

This paper presents the significant performance benefits of Chelsio T6 NVMe-oF over 100GbE iWARP fabric JBOF (just a bunch of flash) solution. Chelsio's T6 adapter delivers line-rate throughput and more than 2.5 Million IOPS at 4K I/O size. In addition, with only 9 µs delta latency between remote and local storage, Chelsio's solution proves to be the best-in-breed in providing the next generation, scalable storage network over standard and cost effective Ethernet infrastructure with an efficient processing path.

Test Results

The following graph presents NVMe-oF READ, WRITE IOPS and throughput results of Chelsio iWARP solution using Null Block devices and SSDs. The results are collected using the **fio** tool with I/O size varying from 4 to 256 KBytes with an access pattern of random READs and WRITEs.

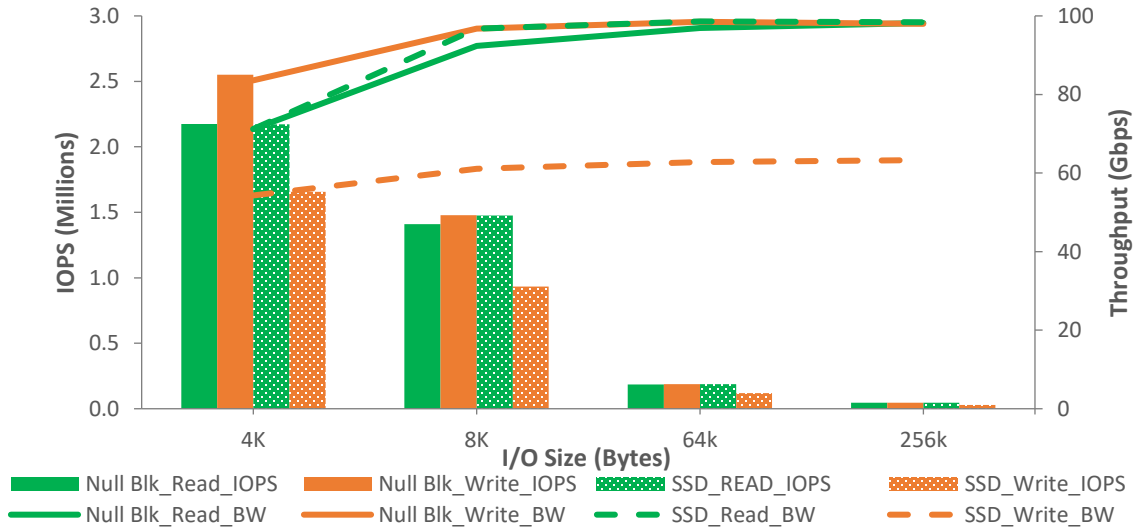


Figure 1 – NVMe-oF IOPS and Throughput vs. I/O size

As evident from the graphs above, T6 solution delivers line-rate throughput for READ (both SSD and Null Block) and WRITE (Null Block). WRITE IOPS (Null Block) exceeds 2.5 Million and READ IOPS (both SSD and Null Block) exceed 2.1 Million at 4K I/O size. *Please note that SSD WRITE throughput and IOPS numbers are limited by SSD performance.*

The following graph presents the latency numbers at 4K I/O size varying the number of connections.

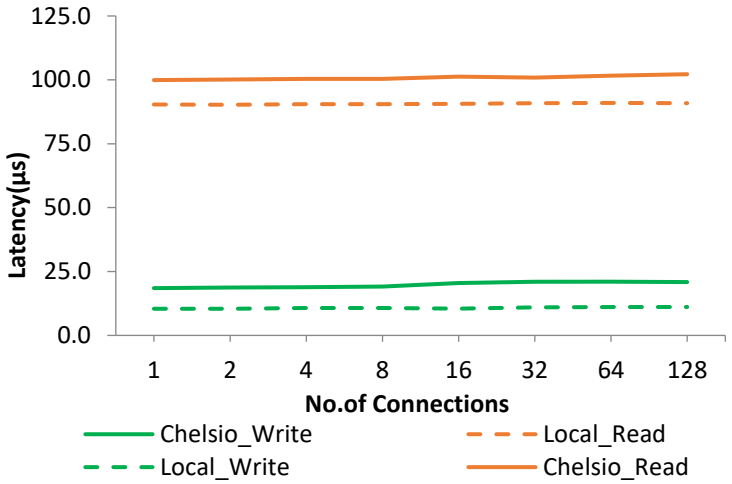
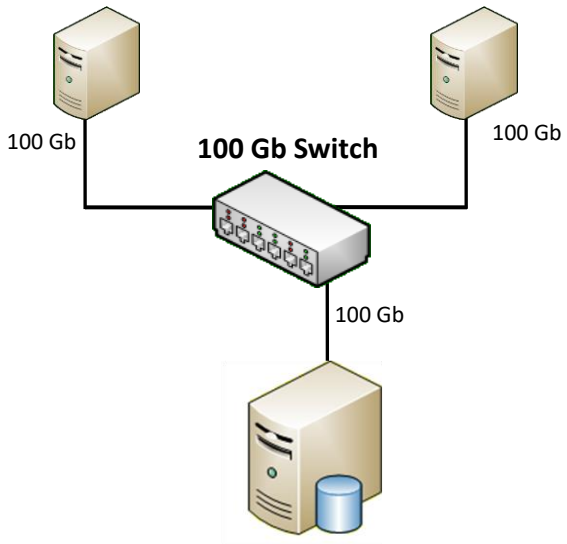


Figure 2 - Latency vs. #. connections

From the graph above, observe that remote versus local NVMe device adds only 9 µs latency at 4K I/O for both READ and WRITE operations. The latency delta does not exceed 12 µs even with 128 connections.

The Demonstration



- Supermicro X10SRA-F Initiators with T62100-CR adapter
- 1 Intel Xeon CPU E5-1620 v4 4-core @ 3.50GHz (HT enabled)
- 16GB RAM
- RHEL 7.3 (4.14.51 kernel)

- Supermicro X10DRG-Q Target with T62100-CR adapter
- 2 Intel Xeon CPUs E5-2687W v4 12-core @ 3.00GHz (HT disabled)
- 128GB RAM
- RHEL 7.3 (4.14.51 kernel)

Figure 3 – Test setup

The setup consists of an NVMe target machine connected to 2 initiator machines through a 100GbE switch using single port on each system. MTU of 9000B is used. Latest Chelsio Unified Wire driver is installed on each machine.

Storage configuration

For the Null Block scenario, the target is configured with 4 LUNs, each of 1GB size. In the SSD scenario, the target is configured with 4 LUNS from 4 Intel 1.6TB NVMe SSDs. In both cases, each initiator connects to 2 targets using 2 connections each.

For latency test, the target is configured with 1 - 128 LUNs from 1 Intel 1.6TB NVMe SSD. 1 Initiator connects to target using 1 connection.

Setup Configuration

Target/Initiator Configuration

i. Disable virtualization, c-state technology, VT-d, Intel I/O AT, Hyperthreading and SR-IOV in BIOS.
ii. Compile and install 4.14.51 kernel from https://github.com/larrystevenwise/linux, linux-4.14-nvme branch.
iii. Install Chelsio Unified Wire v3.8.0.2 and reboot the machine.

```
[root@host~]# make CONF=NVME_PERFORMANCE install
```

iv. For latency test, add *idle=poll* to the kernel command line.
v. Set the below tuned-adm profile for BW/IOPS test:

```
[root@host~]# tuned-adm profile network-throughput
```

Set the below tuned-adm profile for Latency test:

```
[root@host~]# tuned-adm profile network-latency
```

vi. Load the Chelsio iWARP RDMA driver.

```
[root@host~]# modprobe iw_cxgb4
```

vii. Chelsio interface was assigned with IPv4 address and brought-up.
viii. Load the NVMe drivers.

```
[root@host~]# modprobe nvmet
[root@host~]# modprobe nvme
[root@host~]# modprobe nvmet-rdma
```

ix. CPU affinity was set for BW/IOPs test.

```
[root@host~]# t4_perftune.sh -n -Q rdma
```

Target Configuration

BW/IOPs test:

i. Create 4 Null Block devices, each of 1GB size.

```
[root@host~]# modprobe null_blk nr_devices=4 gb=1 use_per_node_hctx=Y
```

Create 4 Logical volumes, each of size 1GB using SSDs.

```
[root@host~]# pvcreate /dev/nvme0n1 /dev/nvme1n1 /dev/nvme2n1 /dev/nvme3n1
/dev/nvme4n1 /dev/nvme5n1 /dev/nvme6n1 /dev/nvme7n1
[root@host~]# vgcreate VG_1 /dev/nvme0n1 /dev/nvme1n1 /dev/nvme2n1
/dev/nvme3n1 /dev/nvme4n1 /dev/nvme5n1 /dev/nvme6n1 /dev/nvme7n1
[root@host~]# for i in `seq 0 3`; do lvcreate -n lun$((i)) -L 1G VG_1 -i 8 -y;
done
```

Latency test:

Create 128 Logical volumes, each of size 1GB using single SSD.

```
[root@host~]# pvcreate /dev/nvme0n1
[root@host~]# vgcreate VG_1 /dev/nvme0n1
[root@host~]# for i in `seq 0 127`; do lvcreate -n lun$((i)) -L 1G VG_1 -i 8 -
y; done
```

ii. Configure the target using the below script:

```
IPPORT="4420"        # 4420 is the reserved NVME/Fabrics RDMA port
IPADDR="10.1.1.149"  # the ipaddress of your target rdma interface
NAME="nvme-nullb"    # Use "nvme-ssd" while configuring SSDs
DEV="/dev/nullb"     # Use "/dev/VG_1/lun" while configuring SSDs

for i in `seq 0 3`; do      # For latency test, use `seq 0 127`
mkdir /sys/kernel/config/nvmet/subsystems/${NAME}${i}
mkdir -p /sys/kernel/config/nvmet/subsystems/${NAME}${i}/namespaces/1
echo -n ${DEV}${i}
>/sys/kernel/config/nvmet/subsystems/${NAME}${i}/namespaces/1/device_path
echo 1 > /sys/kernel/config/nvmet/subsystems/${NAME}${i}/attr_allow_any_host
echo 1 > /sys/kernel/config/nvmet/subsystems/${NAME}${i}/namespaces/1/enable
done

mkdir /sys/kernel/config/nvmet/ports/1
echo 8192 > /sys/kernel/config/nvmet/ports/1/param_inline_data_size
echo "ipv4" > /sys/kernel/config/nvmet/ports/1/addr_adrfam
echo "rdma" > /sys/kernel/config/nvmet/ports/1/addr_trtype
echo $IPPORT > /sys/kernel/config/nvmet/ports/1/addr_trsvcid
echo $IPADDR > /sys/kernel/config/nvmet/ports/1/addr_traddr

for i in `seq 0 3`; do
 ln -s /sys/kernel/config/nvmet/subsystems/${NAME}${i}
/sys/kernel/config/nvmet/ports/1/subsystems/${NAME}${i}
done
```

Initiator Configuration

Target was discovered:

```
[root@host~]# nvme discover -t rdma -a <target_ip> -s 4420
```

BW/IOPs test:

i. Initiator1 connected to Target.

```
[root@host1~]# nvme connect -i 2 -t rdma -a <target_ip> -s 4420 -n nvme_lun0
[root@host1~]# nvme connect -i 2 -t rdma -a <target_ip> -s 4420 -n nvme_lun1
```

ii. Initiator2 connected to Target.

```
[root@host2~]# nvme connect -i 2 -t rdma -a <target_ip> -s 4420 -n nvme_lun2
[root@host2~]# nvme connect -i 2 -t rdma -a <target_ip> -s 4420 -n nvme_lun3
```

iii. *fio* tool was run on both initiators.

```
[root@host~]# fio --rw=randwrite/randread --ioengine=libaio --name=random  --
size=400m --invalidate=1 --direct=1 --runtime=30 --time_based --
fsync_on_close=1 --group_reporting --filename= <device list> --iodepth=64 --
numjobs=16 --bs=<value>
```

Latency test:

i. Single Initiator connects to the target:

```
for i in `seq 0 $((N))`; do /root/nvme-cli/nvme connect -t rdma -a 10.1.1.149
-n nvme-ssd${i} -i 1; done
```

where N specifies the number of target devices.

ii. *fio* tool was run on the initiator.

```
[root@host~]# fio --rw=randwrite/randread --ioengine=libaio --name=random  --
size=400m --invalidate=1 --direct=1 --runtime=30 --time_based --
fsync_on_close=1 --group_reporting --filename= <device list> --iodepth=1 --
numjobs=1 --bs=4K
```

iii. After collecting latency for single target device, initiator was disconnected and logs into multiple target devices for latency collection. This procedure is repeated from 1 to 128 target devices.

Conclusion

This paper showcases the remote storage access performance capabilities of Chelsio T6 NVMe-oF over 100GbE iWARP fabric solution. Using iWARP RDMA enables the NVMe storage devices to be shared, pooled and managed more effectively across a low latency, high performance network. The results show that Chelsio's iWARP RDMA:

- delivers line-rate throughput for READ (with both SSD and Null Block) and WRITE (with Null Block). WRITE performance with SSDs is limited by the number of SSDs used.
- reaches more than 2.5 Million WRITE IOPS at 4K I/O size.
- adds less than 9 µs latency compared to local NVMe device access.

Related Links

NVMe over Fabrics Performance for AMD EPYC
High Performance NVMe-oF with T6 100G iWARP RDMA
NVMe Over Fabrics Performance for Qualcomm ARM
NVMe over Fabrics iWARP Performance