iSCSI JBOF with T7

Using Chelsio T7, Celestica Nebula G2 & Micron SSD

Overview

The Terminator 7 (T7) ASIC from Chelsio Communications, Inc. is a seventh generation, high-performance 1/10/25/40/50/100/200/400 Gbps, Smart NIC Data Processing Unit (DPU) that offers offload support for a wide range of Storage (NVMe/TCP, NVMe-oF, iSCSI, iSER, S2D, SMB), Network (TCP/IP, UDP/IP, RDMA - iWARP and RoCEv2), Virtualization (SR-IOV, VMMQ, VXLAN, NVGRE, Geneve), Crypto (IPsec, TLS/SSL), classification and filtering, and Traffic Management protocols. T7 has an embedded general-purpose ARM processor, fully capable of supporting all the functions of the host server processor. Chelsio Smart NICs unburden communication responsibilities and processing overhead from servers and storage systems resulting in a significant saving of server CPU cycles. This can permit radically reduced system cost by enabling the use of a less expensive CPU or the freed-up CPU can be used for running other applications or workloads.

T7 is a high-performance server or embedded ASIC, fully capable of supporting all the functions of a typical system on a single chip. T7 can be configured as a regular End Point (EP), which can be managed by the server as a traditional NIC or offload adapter. T7 has dual quad-core 1.8GHz, A72 ARM processors which can be used for any additional assistance or higher level processing of the data that is tunneled or offloaded. The ARM system on T7 comes pre-loaded with a Linux kernel. It can be used to run full-offload operations like iSCSI, NVMe/TCP, and NVMe-oF etc. The users can also deploy their custom software stack on the ARM system.

T7 can be configured as a Root Complex (RC), making it the host for other end point devices like PCIe SSDs. The ARM system, embedded in T7 can configure and manage these SSDs. T7 will now act as a JBOF (Just a Bunch Of Flash) storage controller, moving the data directly to the SSDs, without involving the server CPU.

This paper describes the T7's capability to be used as a JBOF and demonstrates it with the T7 Emulation platform, Celestica Nebula G2 storage expansion system and Micron SSD. An iSCSI target is configured on the T7 ARM system, bypassing the server. The Initiator connects to the target over the T7 ethernet ports and accesses the SSD.

Celestica Nebula G2 is an all-flash PCIe 4.0 storage array, supporting up to 24 dual-port NVMe SSDs. It provides high performance, low latency, resource sharing, maximum storage density per enclosure, and high availability, ideal for enterprise workloads.

The Demonstration

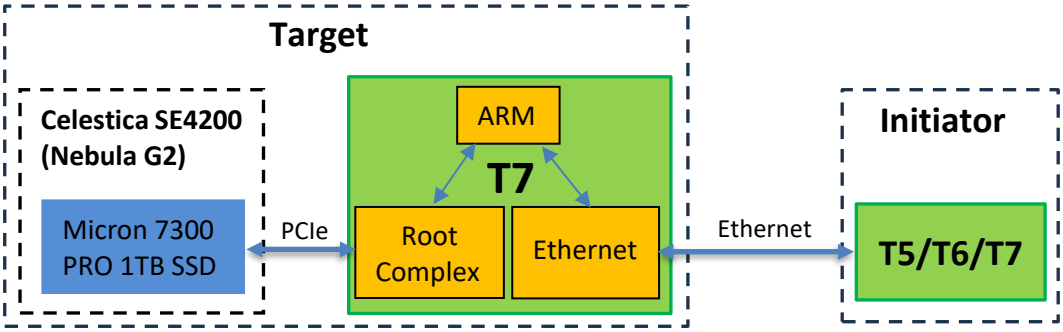

Figure 1 – iSCSI JBOF with T7 ARM, Celestica Nebula G2, and Micron SSD

The test setup consists of a server with the T7 emulation platform configured in RC mode. The Celestica SE4200 (Nebula G2) storage expansion system with a Micron 7300 PRO 1 TB NVMe SSD connects to the T7 Root Complex. The T7 ARM system detects the Micron SSD and configures an LIO iSCSI Target with the SSD.

The Initiator, with a T6 adapter connects to the LIO iSCSI target running on the T7 ARM system. The initiator connects directly (no switch) to the T7 Ethernet port. A standard MTU of 1500 is used on the ports under test.

Test Configuration

iSCSI Target

i. Configure the T7 emulation platform with the required RC images and firmware.
ii. Connect to the T7 ARM system.
iii. Verify that the NVMe SSD is detected.

```
# nvme list

Node             SN                   Model
Namespace Usage                      Format           FW Rev
---------------- -------------------- ---------------------------------------
-- --------- ------------------------- ---------------- --------
/dev/nvme0n1     203129AFD9FD         Micron_7300_MTFDHBE960TDF
1         960.20  GB / 960.20  GB    512   B +  0 B   95420100

# lspci
00:00.0 PCI bridge: Chelsio Communications Inc Device d000
01:00.0 PCI bridge: Broadcom / LSI PEX880xx PCIe Gen 4 Switch (rev b0)
02:00.0 PCI bridge: Broadcom / LSI PEX880xx PCIe Gen 4 Switch (rev b0)
...
03:00.0 System peripheral: Broadcom / LSI Virtual PCIe Placeholder Endpoint
(rev b0)
0d:00.0 System peripheral: Broadcom / LSI Virtual PCIe Placeholder Endpoint
(rev b0)
...
df:00.0 Non-Volatile memory controller: Micron Technology Inc 7300 PRO NVMe
SSD (rev 01)
```

```
e9:00.0 System peripheral: Broadcom / LSI Virtual PCIe Placeholder Endpoint
(rev b0)
f3:00.0 Serial Attached SCSI controller: Broadcom / LSI PCIe Switch
management endpoint (rev b0)
```

iv. Load the Chelsio NIC driver (*cxgb4*) and bring up the T7 Ethernet port with an IPv4 address.

```
# insmod cxgb4.ko
# ifconfig ethX 102.55.55.70/24 up
```

v. Load the LIO Target driver.

```
# insmod cxgbit.ko
```

vi. Configure the target with Micron SSD using the below script.

```
# lsblk
NAME        MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
nvme0n1     259:0    0 894.3G  0 disk
|-nvme0n1p1 259:1    0    95M  0 part
|-nvme0n1p2 259:2    0    10G  0 part
`-nvme0n1p3 259:3    0     1G  0 part

# sh iscsi_target.sh /dev/nvme0n1p3 102.55.55.70
Target iqn.2003-01.org.linux-iscsi.buildroot:nvme0n1p3, portal 0.0.0.0:3260
has been created

# cat iscsi_target.sh

print_usage() {
    cat <<EOF
Usage: $(basename $0) [-p PORTAL] DEVICE|FILE
Export a block device or a file as an iSCSI target with a single LUN
EOF
}

die() {
    echo $1
    exit 1
}

while getopts "hp:" arg; do
    case $arg in
        h) print_usage; exit 0;;
        p) PORTAL=${OPTARG};;
    esac
done
shift $(($OPTIND - 1))

DEVICE=$1
[ -n "$DEVICE" ] || die "Missing device or file argument"
[ -b $DEVICE -o -f $DEVICE ] || die "Invalid device or file: ${DEVICE}"
IQN="iqn.2003-01.org.linux-iscsi.$(hostname):$(basename $DEVICE)"
[ -n "$PORTAL" ] || PORTAL="0.0.0.0:3260"

CONFIGFS=/sys/kernel/config
CORE_DIR=$CONFIGFS/target/core
ISCSI_DIR=$CONFIGFS/target/iscsi
# Load the target modules and mount the config file system
```

```
lsmod | grep -q configfs || modprobe configfs
lsmod | grep -q target_core_mod || modprobe target_core_mod
mount | grep -q ^configfs || mount -t configfs none $CONFIGFS
mkdir -p $ISCSI_DIR

# Create a backstore
if [ -b $DEVICE ]; then
    BACKSTORE_DIR=$CORE_DIR/iblock_0/data
    mkdir -p $BACKSTORE_DIR
    echo "udev_path=${DEVICE}" > $BACKSTORE_DIR/control
else
    BACKSTORE_DIR=$CORE_DIR/fileio_0/data
    mkdir -p $BACKSTORE_DIR
    DEVICE_SIZE=$(du -b $DEVICE | cut -f1)
    echo "fd_dev_name=${DEVICE}" > $BACKSTORE_DIR/control
    echo "fd_dev_size=${DEVICE_SIZE}" > $BACKSTORE_DIR/control
    echo 1 > $BACKSTORE_DIR/attrib/emulate_write_cache
fi
echo 1 > $BACKSTORE_DIR/enable

# Create an iSCSI target and a target portal group (TPG)
mkdir $ISCSI_DIR/$IQN
mkdir $ISCSI_DIR/$IQN/tpgt_1/

# Create a LUN
mkdir $ISCSI_DIR/$IQN/tpgt_1/lun/lun_0
ln -s $BACKSTORE_DIR $ISCSI_DIR/$IQN/tpgt_1/lun/lun_0/data
echo 1 > $ISCSI_DIR/$IQN/tpgt_1/enable

# Create a network portal
mkdir $ISCSI_DIR/$IQN/tpgt_1/np/$PORTAL

# Disable authentication
echo 0 > $ISCSI_DIR/$IQN/tpgt_1/attrib/authentication
echo 1 > $ISCSI_DIR/$IQN/tpgt_1/attrib/generate_node_acls

# Allow write access for non authenticated initiators
echo 0 > $ISCSI_DIR/$IQN/tpgt_1/attrib/demo_mode_write_protect

echo "Target ${IQN}, portal ${PORTAL} has been created"
```

iSCSI Initiator

i. Load the Chelsio NIC driver (*cxgb4*) and bring up the interface with an IPv4 address.

```
[root@initiator~]# modprobe cxgb4
[root@initiator~]# ifconfig ethX <IPv4 address> up
```

ii. Login to the target.

```
[root@initiator~]# iscsiadm -m discovery -t st -p 102.55.55.70:3260 -I
default -l

102.55.55.70:3260,1 iqn.2003-01.org.linux-iscsi.buildroot:nvme0n1p3
Logging in to [iface: default, target: iqn.2003-01.org.linux-
iscsi.buildroot:nvme0n1p3, portal: 102.55.55.70,3260]
Login to [iface: default, target: iqn.2003-01.org.linux-
iscsi.buildroot:nvme0n1p3, portal: 102.55.55.70,3260] successful.
```

iii. Format the disk and mount it.

```
[root@initiator~]# lsscsi
[0:0:0:0]    disk    ATA      SanDisk SDSSDA12 80RL  /dev/sda
[10:0:0:0]   disk    LIO-ORG  IBLOCK           4.0   /dev/sdb

[root@initiator~]# mkfs.ext4 /dev/sdb
mke2fs 1.45.6 (20-Mar-2020)
/dev/sdb contains a ext4 file system
        created on Tue Dec  5 23:57:56 2023
Proceed anyway? (y,N) y
Creating filesystem with 262144 4k blocks and 65536 inodes
Filesystem UUID: 2de92f0e-806b-449d-8921-25629b2c88cd
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376

Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done

[root@initiator~]# mount /dev/sdb /mnt/iscsi0/
```

iv. *iozone* tool was run on the mounted disk.

```
[root@initiator~]# cd /mnt/iscsi0/
[root@initiator~]# iozone -a -d -+I
       Iozone: Performance Test of File I/O
                Version $Revision: 3.398 $
                Compiled for 64 bit mode.
                Build: linux-AMD64
 …

        Run began: Wed Dec  6 00:03:54 2023

        Auto Mode
        Command line used: iozone -a -d -+I
        Output is in Kbytes/sec
        Time Resolution = 0.000001 seconds.
        Processor cache size set to 1024 Kbytes.
        Processor cache line size set to 32 bytes.
        File stride size set to 17 * record size.
                                                                    random  random
bkwd   record   stride
              KB  reclen   write rewrite    read    reread    read   write
read  rewrite     read   fwrite frewrite   fread  freread
              64       4  621657 2006158  4897948  4564786 3791156 1768283
1679761  1947927  2133730  1484662  2772930 2892445  5735102 …
```

Conclusion

The Chelsio T7 solution enables NVMe SSDs to be shared, pooled, and managed more effectively across a low-latency, high-performance network. The T7 ARM can do the complete processing, freeing up significant server CPU resources for application processing. As data centers become saturated with information, the need to offload tasks from server's CPU is more important now than ever.

With concurrent support for offloading multiple networking, storage, virtualization, and crypto protocols, and delivering industry-leading performance and efficiency, Chelsio has taken the Unified Wire solution to the next level. All the traffic runs over a single network, rather than building and maintaining multiple networks, resulting in significant acquisition and operational cost savings.

Related Links

T7 Product Brief
NVMe/TCP & iSCSI JBOF Demonstration with T7 and ASMedia PCIe Switch
NVMe/TCP PDU Offload Demonstration with T7
Chelsio T7 DPU Line Launched for 400G Generation