

How to setup NVMe over Fabric on Azure cloud with MayaScale

In this post we will find out how simple it is to setup NVMe Fabric over TCP on Azure cloud with the recently announced general availability of the Lsv2-series Azure Virtual Machines (VMs) that come with high throughput, low latency, and directly mapped local NVMe storage.

The Lsv2-series VMs are available in the following sizes in the following regions: West US 2, East US, East US 2, West Europe, and SE Asia.

Size	vCPU's	Memory (GiB)	NVMe Disk	NVMe Disk Throughput (Read IOPS/MBps)
L8s_v2	8	64	1 x 1.92 TB	340,000 / 2,000
L16s_v2	16	128	2 x 1.92 TB	680,000 / 4,500
L32s_v2	32	256	4 x 1.92 TB	1,400,000 / 9,000
L64s_v2	64	512	8 x 1.92 TB	2,700,000 / 18,000
L80s_v2	80	640	10 x 1.92 TB	3,400,000 / 22,000

MayaScale Cloud Data Platform provides high-Performance shared storage using NVMe Fabric over TCP, that can readily deployed from [Azure Marketplace](#).

1. Deploy [MayaScale Cloud Data Platform](#) from Azure Marketplace and configure it to use L8s_v2, which comes with 1.92 TB of NVMe storage.
2. For clients we can deploy compute optimized VM of type Standard F8s (8 vcpus, 16 GB memory) running Centos 7.5
3. Connect to the Mayascale server using SSH and make sure the NVMe disk is available.

```
[root@mayatest1 ~]# /opt/zettalane/bin/nvme list
Node SN Model Namespace Usage Format FW Rev
-----
-----
-----
/dev/nvme0n1 e6734e2f833300000001 Microsoft NVMe Direct Disk 1 0.00 B
/ 1.92 TB 512 B + 0 B NVMDV001
```

4. Using our mayacli we can share the entire NVMe namespace over nvme-tcp. First define the volume to be shared as mynvme1 and then create mapping over nvmet-tcp. In this example the default NVME target controller name will be used.

```
[root@mayatest1 ~]# mayacli create vol mynvme1 disk=/dev/nvme0n1
[root@mayatest1 ~]# mayacli create mapping mynvme1 controller=nvmet-
tcp
[root@mayatest1 ~]# mayacli show mapp
Configured Volume Mappings:
Stat: A = Active, I = Inactive, C = Cluster E = Error

Volume Device Controller TID LUN Stat Options
-----
-----
mynvme1 /dev/nvme0n1 nvmet-tcp0 0 1 I nqn.2018-07.com.zettalane:
mayatest1.10a0600
[root@mayatest1 ~]#
```

5. To make this NVMe volume discoverable we have to activate it.

```
[root@mayatest1 ~]# mayacli bind mapping mynvme1
[root@mayatest1 ~]# mayacli show mapping
Configured Volume Mappings:
Stat: A = Active, I = Inactive, C = Cluster E = Error

Volume Device Controller TID LUN Stat Options
-----
-----
mynvme1 /dev/nvme0n1 nvmet-tcp0 0 1 A nqn.2018-07.com.zettalane:
mayatest1.10a0600
```

From the output `nqn.2018-07.com.zettalane:mayatest1.10a0600` is the default namespace discoverable at default TCP port 4420

6. On the client VM install the host driver for NVMeoF over TCP using this `kmod-nvme-tcp-1.0-1.x86_64.rpm` package, available in your account Downloads section. Once installation is successful load the driver `nvme-tcp`.

```
[root@client1 ~]# yum install kmod-nvme-tcp-1.0-1.x86_64.rpm
[root@client1 ~]# modprobe nvme-tcp
```

7. Now the fun part begins. We will try to discover the NVMe storage from client

```
[root@client1 ~]# /opt/zettalane/bin/nvme discover -a 10.1.0.6 -t tcp
-s 4420

Discovery Log Number of Records 1, Generation counter 1
====Discovery Log Entry 0====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 0
trsvcid: 4420
subnqn: nqn.2018-07.com.zettalane:mayatest1.10a0600
traddr:
```

8. To start using the discovered NVMe device we have to connect to it.

```
[root@client1 ~]# /opt/zettalane/bin/nvme connect -a 10.1.0.6 -t tcp -
s 4420 -n nqn.2018-07.com.zettalane:mayatest1.10a0600
[root@client1 ~]# /opt/zettalane/bin/nvme list
Node SN Model Namespace Usage Format FW Rev
-----
-----
-----
/dev/nvme0n1 d15fed2f1b9f83e0 Linux 1 1.92 TB / 1.92 TB 512 B + 0 B
3.10.0-6
```

9. Now we perform IO to the block device /dev/nvme0n1 and make use of it. Once finished we can disconnect as follows.

```
[root@client1 ~]# /opt/zettalane/bin/nvme disconnect -n nqn.2018-07.
com.zettalane:mayatest1.10a0600
NQN:nqn.2018-07.com.zettalane:mayatest1.10a0600 disconnected 1
controller(s)
```

10. Now time for you to experience NVME Fabric on cloud by deploying [MayaScale Cloud](#) from marketplace.