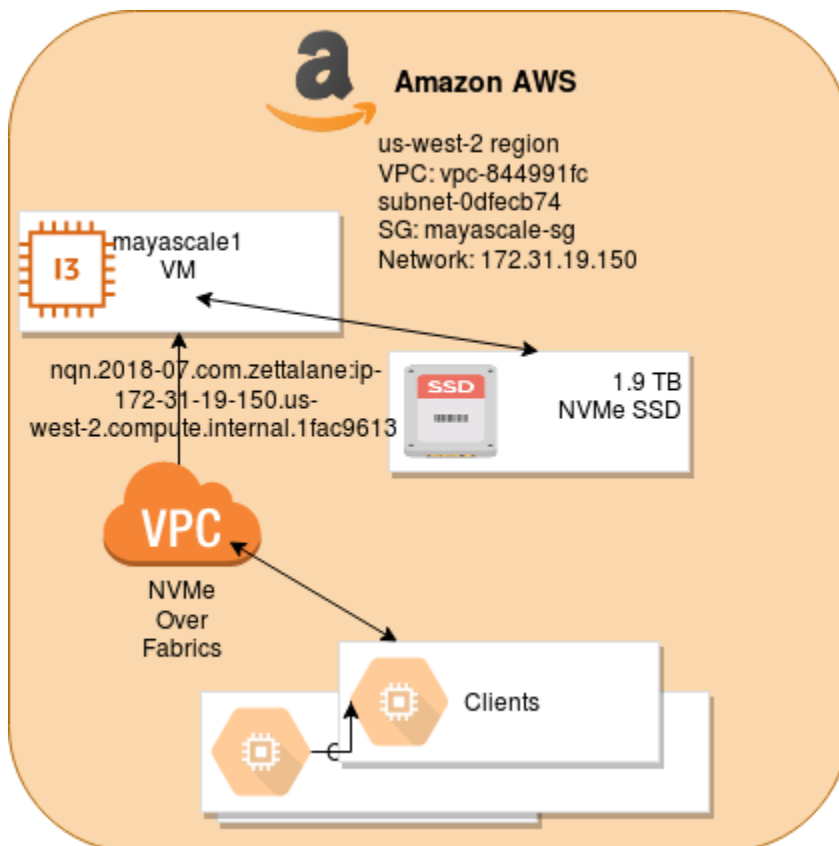# MayaScale Data Platform on AWS

Amazon AWS Cloud Platform provides rich set of high-performance NVMe resources for building true enterprise-class NVMe Over Fabrics readily. Note that the network bandwidth is tied to the number of cpu cores of the compute instance. The storage IOPS is based on capacity of provisioned storage. Please refer to AWS cloud documentation for more information on storage optimized instance types.

Purpose	Machine	Cores	Memory	Network	Storage
Shared block storage for IP-SAN or NVMeoF	Storage Optimized i3.xlarge Other i3 series	4	30.5GB	~10Gbps	Ephemeral direct NVMe or SSD 1 x 950
High Performance NFS Server	Storage Optimized i3.2xlarge i2.2xlarge	8	61GB	~10Gbps	zpool 1 x 1900 (NVMe) 2 x 800 (SSD)
High-Availability	High Availability Set			Secondary Private IP address	Synchronous mirroring by using zpool mirror dm raid-1 over NVMe target DRBD replication

Here is the sequence of steps involved in experiencing the industry's very first elastic NVMe after you've deployed a MayaScale instance from AWS marketplace. In this tutorial we will assume you have launched EC2 i3-xlarge instance type that comes with 1 x 1.9T NVMe SSD.

- 1 EC2 instance `mayascale1`
- 1.9 TB NVMe SSD (ephemeral)
- Default network with private IP 172.31.19.150 (ip-172-31-19-150.us-west-2.compute.internal)

1. Connect to the MayaScale Cloud Data Platform instance using SSH to secure the Web console GUI access by changing the default password to something random by running

```
# /opt/mayastor/web/genrandpass.sh
```

Or to set your own password

```
# /opt/mayastor/web/changepass.sh
Login name (default admin):
Login password:
Password again:
```

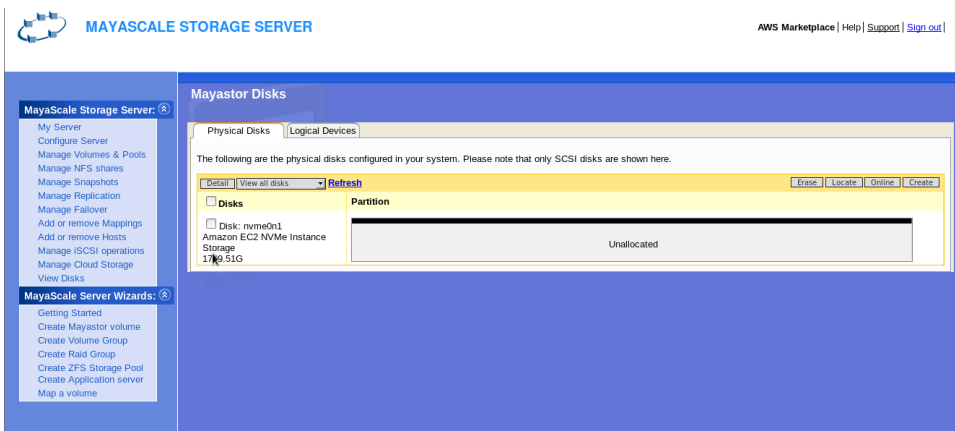And then restart the web server for password changes to take effect

```
# /opt/mayastor/web/stop

# /opt/mayastor/web/start
```

2. Now you can proceed with provisioning NVMe storage using the Administration Web console available on http://<mayascale1-ip>:2020

3. Based on the number of NVMe SSD that is available on the instance you will be able to view them on the GUI from **View Disks**

For provisioning NVMe storage to clients you can create volume group from the physical NVMe and then slice them as logical volumes for flexible provisioning. But for this guide we will be provisioning the whole NVMe resource to client.

Using Mayascale Server Wizards Create Mayastor Volume. Then **Select Simple Volume** and click **Next**
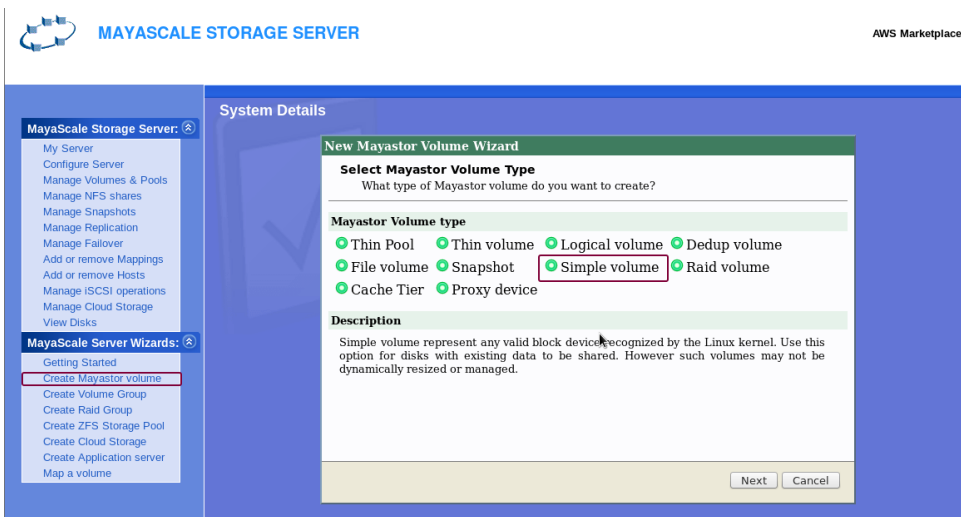
4. Select the device for provisioning and click **Next**. In this example nvme0n1 is chosen.

New Mayastor Volume Wizard

Select block device for Mayastor volume

Select the block device that will be shared as a SCSI compliant disk to the application server.

All available block devices:

☐ Show All

	Name	Capacity	1K-blocks	Major	Minor
☐	xvda	8.000G	8388608	202	0
☐	xvda1	7.999G	8387584	202	1
☐	nvme0n1	1769.513G	1855468750	259	0

Back   Next   Cancel

5. The next screen is to specify a name for the provisioned volume with brief description.

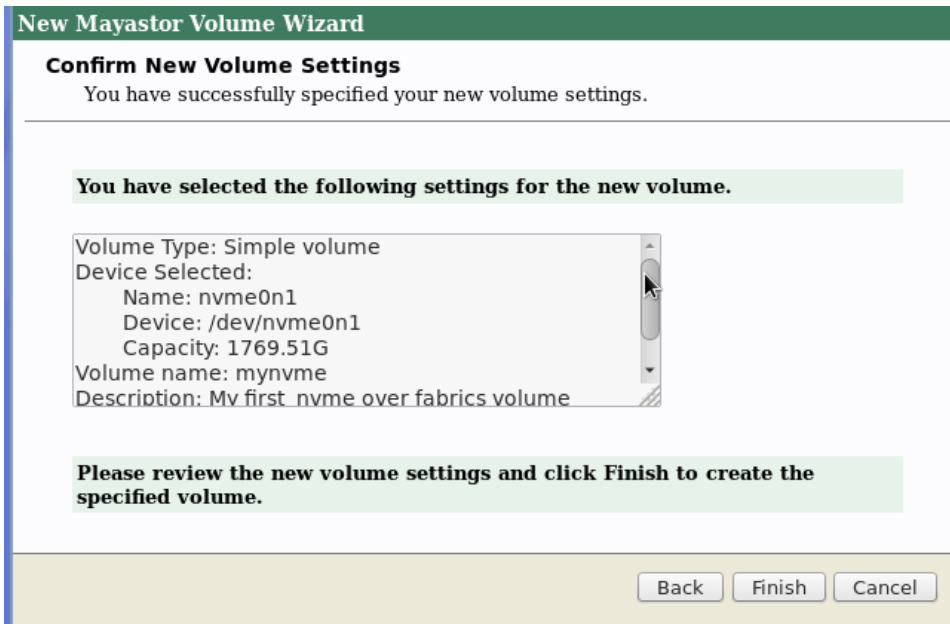### New Mayastor Volume Wizard

Create Volume

Please specify a name for this Volume. You may also provide a brief description for your reference.

Volume Label:

mynvme

Description:

My first nvme over fabrics volume

Back   Next   Cancel

6. Confirm the new volume settings.

7. Once the volume mynvme was created successfully the next step is to make them available to clients using the NVMe TCP protocol. For this use the wizard **Map a volume** . Here select the newly create volume *mynvme* and choose nvmet-tcp as the controller. For iSCSI provisioning the controller will be iscsi. Then click **Next**

8. Here the LUN refers the NVMe namespace id and it starts from 1. For iSCSI the LUN starts from 0.

9. Confirm mapping settings.

10. Once mapping was created successfully it can be viewed by clicking **Add or remove Mappings** from the sidebar. Here you would notice the default NQN for the new volume.

11. The next step is to activate the newly created mapping so that it is discoverable from NVMe hosts in the network. To activate click on the checkbox associated with *mynvme*
and click **Bind**

Volume | LUN

Bind Unbind Delete New   **Refresh** View all ▾

	Volume	Status	Device	Controller	TID	LUN	Host	ClusterID	Options
☐	mynvme	online	/dev/nvme0n1	nvmet-tcp	0	1			nqn.2018-07.com.zettalane:ip-172-31-19-150.us-west-2.compute.internal.1fac9613
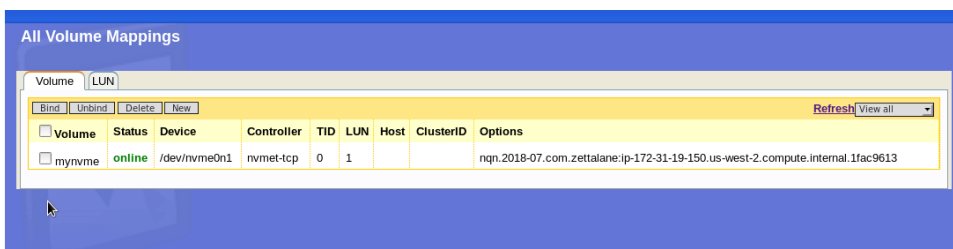
12. To discover this NVMe target from another client with NVMe over fabrics support.

```
[root@ip-172-31-20-45 ~]# modprobe nvme-tcp
[root@ip-172-31-20-45 ~]# /opt/zettalane/bin/nvme  discover -a 172.31.19.150 -s 4420 -t tcp

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:ip-172-31-19-150.us-west-2.compute.internal.1fac9613
traddr:
[root@ip-172-31-20-45 ~]#
```

13. For the client to perform IO on the discovered NVMe target, it has to be connected and configured with the host as follows.

```
[root@ip-172-31-20-45 ~]# /opt/zettalane/bin/nvme  connect -a 172.31.19.150   -s 4420  -t tcp -n  nqn.
2018-07.com.zettalane:ip-172-31-19-150.us-west-2.compute.internal.1fac9613
[root@ip-172-31-20-45 ~]# /opt/zettalane/bin/nvme list
[root@ip-172-31-20-45 ~]# /opt/zettalane/bin/nvme list
Node            SN                   Model                                    Namespace
Usage                    Format          FW Rev
--------------- -------------------- ---------------------------------------- ---------
------------------------- --------------- --------
/dev/nvme0n1    vol047c5ca2e5ca231b8 Amazon Elastic Block Store               1          0.00  B /
8.59  GB    512  B +  0 B   1.0
/dev/nvme1n1    7e2888517b895b8      Linux                                    1          1.90  TB /
1.90  TB    512  B +  0 B   3.10.0-6
```

14. Congratulations you have created NVMe target with MayasScale and mapped over NVMeof-TCP successfully to be discovered by clients.