

# FIO Performance characterization on GCP

## Background

NVMe has already established as storage choice for modern workloads requiring low latency and high performance. On cloud it is available as NVMe ephemeral resources that can be easily attached to computing instances. Though this option is within reach it comes with other challenges

- Storage is ephemeral in nature and cannot be used when persistence is required as the contents do not survive stop
- Compute instances cannot be stopped and it requires removal and recreating of instance if one does. Now your applications need to consider this unnecessarily.
- The minimum capacity is 475Gib even if the application will use only 1/10 of that capacity.
- Hinder any solution that require separation of storage and compute.
- The Elastic Block Storage for a single compute instance will peak at 40K IOPS.

MayaScale provides NVMe Fabric over TCP on Google Cloud for high-performance shared storage and its performance is compared against running locally using the ephemeral local NVMe SSDs.

## Benchmark Setup

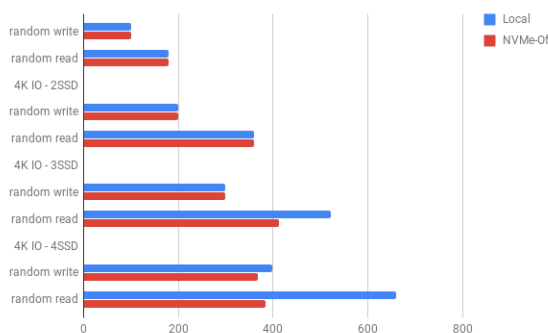
MayaScale works better when configured with multiple cores to provide same parallelism seen with NVMe hardware queues and also the compute instance network and storage performance is also tied to the number of provisioned CPU cores. The software has very low memory requirement only.

- server node of machine type n1-highcpu-16 running MayaScale software
- 16 CPU cores (Broadwell) and ~14 GB memory
- 4 Local NVMe SSDs of 375GB
- client node of machine type n1-highcpu-16

## FIO Performance Report

4K IO	Local IOPS (K)	NVMe-Of IOPS (K)
random write	100	100
random read	180	180

4K IO	Local IOPS (K)	NVMe-Of IOPS (K)
random write	200	199
random read	360	360
4K IO	Local IOPS (K)	NVMe-Of IOPS (K)
random write	299	299



rando m read	522	413
--------------------	-----	-----

4K IO	Local IOPS (K)	NVMe- Of IOPS (K)
rando m write	398	368
rando m read	659	384

## Conclusion

Mayascale is able to deliver same IOPS and Bandwidth but with added latency even over NVME fabric using TCP.