MinIO S3 Throughput Benchmark on NVMe SSD Using mc support perf

DECEMBER 2022

MinIO S3 Throughput Benchmark on NVMe SSD

MinIO is a high-performance, Kubernetes-native object store. Optimized for cloud-native workloads, MinIO is completely compatible with S3, inside AWS or on other public and private clouds and Kubernetes distributions. Enterprises use MinIO to deliver against ML/AI, analytics, backup and archival workloads - all from a single platform. Remarkably simple to install and manage, MinIO offers a rich suite of enterprise features targeting security, resiliency, data protection, scalability and identity management.

Applications access data over the network using atomic, immutable object APIs where the data is often in a Binary Large Object (BLOB) format. The relevant performance metrics for object storage are measured in terms of I/O throughput, rather than IOPS.

This document describes the benchmarks that MinIO engineering ran to determine the performance of the MinIO Object Storage Server when run on NVMe. Specifically, this document shows how to setup the benchmarking environment, how to run the benchmarking tools and reviews the performance results in detail.

Our results running on a 32 node MinIO cluster can be summarized as follows:

InstanceType	PUT	GET	Parity	mc CLI version	MinIO version
i3en.24xlarge	165 GiB/sec	325 GiB/sec	EC:4	RELEASE.2021-12-29T06-52-55Z	RELEASE.2021-12-29T06-49-06Z

1. Benchmark Environment

For the purpose of this benchmark, MinIO utilized AWS bare-metal, storage optimized instances with local NVMe drives and 100 GbE networking. These are the same instances that MinIO recommends to its production clients for use in the AWS cloud.

1.1 Hardware

Instance	# Nodes	AWS Instance type	CPU	MEM	Storage	Network
Server	32	i3en.24xlarge	96	768 GB	8x7500GB	100 Gbps

1.2 Software

Property	Value
Server OS	RELEASE.2021-12-29T06-52-55Z
MinIO Version	RELEASE.2021-12-29T06-49-06Z
Benchmark Tool	mc support perf

1.3 MC Support Perf

MinIO used **mc support perf** as the benchmarking tool for this work. This tool is built into the MinIO Server and is accessed through the **Console UI** or **mc support perf** command. It requires no special skills or additional software. This built-in tool measures the purest expression of the speed of MinIO.

1.4 Performance Tuning

No additional tuning is needed. MinIO used the default Ubuntu 20.04 install on AWS and the latest MinIO binary.

2. Understanding Hardware Performance

2.1 Measuring Single Drive Performance

The performance of each drive was measured using the command dd. DD is a unix tool used to perform bit-by-bit copy of data from one file to another. It provides options to control the block size of each read and write.

Here is a sample of a single NVMe drive's Write Performance with 16MB block-size, O_DIRECT option for a total of 64 copies. Note that we achieve greater than 1.1 GB/sec of write performance for each drive.

```
ubuntu@ip-172-31-62-201:~$ dd if=/dev/zero of=/disk1/test bs=16M count=64 oflag=direct
64+0 records in
64+0 records out
1073741824 bytes (1.1 GB, 1.0 GiB) copied, 1.03775 s, 1.0 GB/s
```

Here is the output of a single HDD drive's Read Performance with 16MB block-size using the O_DIRECT option and a total count of 64. Note that we achieved greater than 2.3 GB/sec of read performance for each drive.

```
ubuntu@ip-172-31-62-201:~$ dd if=/disk1/test of=/dev/null bs=16M count=64 iflag=direct
64+0 records in
64+0 records out
1073741824 bytes (1.1 GB, 1.0 GiB) copied, 0.466429 s, 2.3 GB/s
```

2.2 Measuring JBOD Performance

JJBOD performance with O_DIRECT was measured using https://github.com/minio/dperf. dperf is a filesystem benchmark tool that generates and measures filesystem performance for both read and write. dperf command operating with 64 parallel threads, 4MB block-size and O_DIRECT by default.

```
ubuntu@ip-172-31-62-201:~$ ./dperf -b 4MiB -f 512MiB /disk{1..8}/tmp{1..8}
Aggregate READs: 16 GiB/s
Aggregate WRITEs: 8.0 GiB/s
```

2.3 Network Performance

The network hardware on these nodes allows a maximum of 100 Gbit/sec. 100 Gbit/sec equates to 12.5 Gbyte/sec (1 Gbyte = 8 Gbit).

Therefore, the maximum throughput that can be expected from each of these nodes would be 12.5 Gbyte/sec.

3. Running the 32-node Distributed MinIO benchmark

1 Running MC Support Perf

MinIO ran mc support perf in autotune mode. This mode autotunes concurrency to obtain maximum throughput and IOPS (Input/Output Per Second).

```
$ mc support perf minio/
```

The test will run and present results on screen. The test may take anywhere from a few seconds to several minutes to execute depending on your MinIO cluster. The flag -v indicates verbose mode.

1.1 MINIO_STORAGE_CLASS_STANDARD=EC:2

```
ubuntu@ip-172-31-62-70:~$ ./mc support perf minio/
...
ObjectPerf: ✓

     THROUGHPUT  IOPS
PUT: 197 GiB/s   3,145 objs/s
GET: 323 GiB/s   5,167 objs/s

MinIO 2021-12-29T06:49:06Z, 32 servers, 256 drives, 64 MiB objects, 72 threads
```

1.2 MINIO_STORAGE_CLASS_STANDARD=EC:3

```
ubuntu@ip-172-31-62-70:~$ ./mc support perf minio/
...
ObjectPerf: ✓

     THROUGHPUT  IOPS
PUT: 181 GiB/s   2,900 objs/s
GET: 323 GiB/s   5,171 objs/s

MinIO 2021-12-29T06:49:06Z, 32 servers, 256 drives, 64 MiB objects, 72 threads
```

1.3 MINIO_STORAGE_CLASS_STANDARD=EC:4 (default)

```
ubuntu@ip-172-31-62-70:~$ ./mc support perf minio/
...
ObjectPerf: ✓

     THROUGHPUT  IOPS
PUT: 165 GiB/s   2,639 objs/s
GET: 325 GiB/s   5,193 objs/s

MinIO 2021-12-29T06:49:06Z, 32 servers, 256 drives, 64 MiB objects, 72 threads
```

3.2 Interpretation of Results

The average network bandwidth utilization during the write phase was 77 Gbit/sec and during the read phase was 84.6 Gbit/sec. This represents client traffic as well as internode traffic. The portion of this bandwidth available to clients is about half for both reads and writes.

The network was almost entirely choked during these tests. Higher throughput can be expected if a dedicated network was available for inter-node traffic.

Note that the write benchmark is slower than read because benchmark tools do not account for write amplification (traffic from parity data generated during writes). In this case, the 100 Gbps network is the bottleneck as MinIO gets close to hardware performance for both reads and writes.

4. Conclusion

Based on the results above, we found that MinIO takes complete advantage of the available hardware. Its performance is only constrained by the underlying hardware available to it. This benchmark has been tested with our recommended configuration for performance workloads and can be easily replicated in an hour for less than $350.