



2019 级

《大数据数据存储与管理》课程

## 实 验 报 告

姓 名 杨超平

学 号 U201915015

班 号 物联网 1901 班

日 期 2022.04.15

## 目 录

一、实验目的 .....	1
二、实验背景 .....	1
三、实验环境 .....	1
四、实验内容 .....	2
4.1 对象存储技术实践 .....	2
4.2 对象存储性能分析 .....	2
五、实验过程 .....	2
六、实验总结 .....	6
参考文献 .....	8

## 一、实验目的

1. 熟悉对象存储技术，代表性系统及其特性；
2. 实践对象存储系统，部署实验环境，进行初步测试；
3. 基于对象存储系统，架设实际应用，示范主要功能。

## 二、实验背景

本次实验为对象存储入门实验，其中主要的部分有：基础环境搭建；对象存储服务器端准备；对象存储客户端准备；对象存储测评工具的使用。

使用的对象存储服务端是 Minio，是一个基于 Apache License V2.0 开源协议的对象存储服务。专为大型私有云基础架构而设计。它与 Amazon S3 云存储服务兼容，最适合存储非结构化数据，如照片，视频，日志文件，备份和容器/虚拟机映像。

使用的对象存储客户端是 Minio 的客户端部分。

对象存储测评工具是 COSBench，COSBench（cloud object storage benchmark）是 Intel 开发的，对云存储的测试工具。

## 三、实验环境

实验平台为 Windows11 操作系统：

设备名称	USER-20211031IS
处理器	Intel(R) Core(TM) i7-10510U CPU @ 1.80GHz 2.30 GHz
机带 RAM	12.0 GB (11.8 GB 可用)
设备 ID	DCA299F1-CC80-448B-A23C-970A950F183C
产品 ID	00331-20350-21990-AA214
系统类型	64 位操作系统, 基于 x64 的处理器
笔和触控	没有可用于此显示器的笔或触控输入

服务器端	Minio
客户端	mc

## 四、实验内容

### 4.1 对象存储技术实践

1. 在 Windows 环境下下载配置 minio 服务器；
2. 下载 mc 客户端，创建 bucket，上传文件；
3. 安装使用 cosbench 提交 xml 文件测试。

### 4.2 对象存储性能分析

1. 测试读写性能的优劣。
2. 测试 work 值对各项指标的影响，包含吞吐率，带宽等。
3. 测试不同块大小对各项指标的影响。

## 五、实验过程

### 1. 配置 minio server 端。

```
C:\Users\dell>E:\minio\minio.exe server E:\minio\minioserver

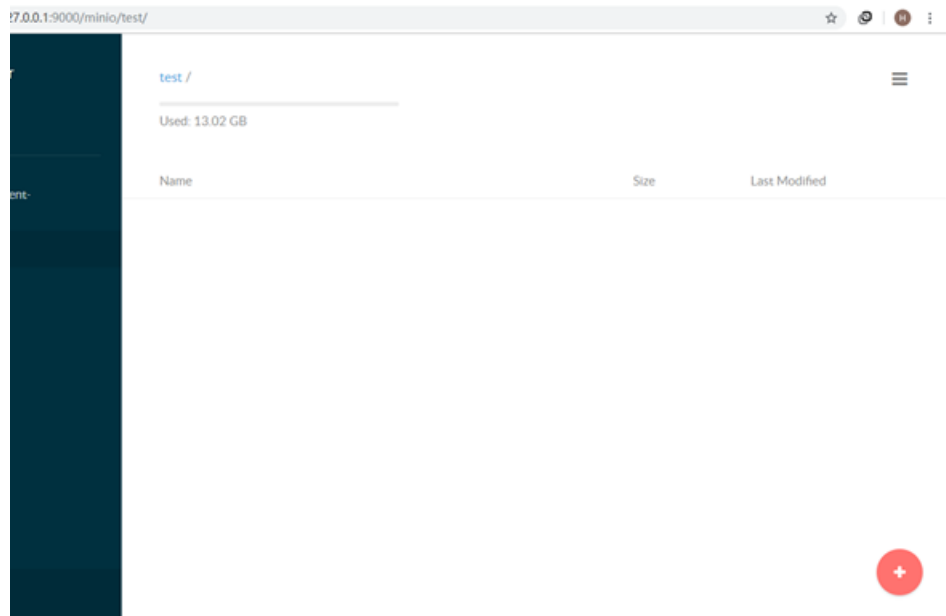
Endpoint:  http://169.254.34.249:9000  http://169.254.138.248:9000  http://10.11.55.235:9000  http://169.254.235.88:9000
           http://169.254.38.21:9000  http://192.168.1.2:9000  http://127.0.0.1:9000
AccessKey: INP1BWF9UASLA6ZW9E7W
SecretKey: EuKUeXSUvc6EXDFf+THWj+4aD+BQSZIlocNTcG++

Browser Access:
  http://169.254.34.249:9000  http://169.254.138.248:9000  http://10.11.55.235:9000  http://169.254.235.88:9000  http://
/169.254.38.21:9000  http://192.168.1.2:9000  http://127.0.0.1:9000

Command-line Access: https://docs.min.io/docs/minio-client-quickstart-guide
$ mc.exe config host add myminio http://169.254.34.249:9000 INP1BWF9UASLA6ZW9E7W EuKUeXSUvc6EXDFf+THWj+4aD+BQSZIlocNT
cG++

Object API (Amazon S3 compatible):
Go:      https://docs.min.io/docs/golang-client-quickstart-guide
Java:    https://docs.min.io/docs/java-client-quickstart-guide
Python:  https://docs.min.io/docs/python-client-quickstart-guide
JavaScript: https://docs.min.io/docs/javascript-client-quickstart-guide
.NET:    https://docs.min.io/docs/dotnet-client-quickstart-guide
```

在浏览器中打开 <http://127.0.0.1:9000>，为服务器的可视化管理界面，输入 AccessKey 和 SerectKey 登陆。

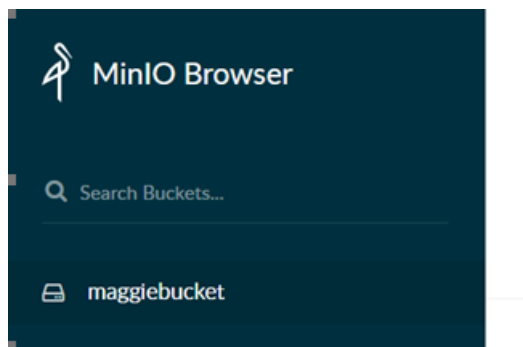


## 2. 配置 minio client 端

下载客户端 `mc.exe`，通过命令行添加服务器 `minioserver`

```
Windows PowerShell
PS G:\> G:\mc.exe config host add minio http://127.0.0.1:9000 mumuzyl 12345654321 S3v4
Added minio successfully.
PS G:\> G:\mc.exe mb minio/mumubucket
PS G:\> G:\mc.exe mb minio/mumubucket
Bucket created successfully minio/mumubucket.
PS G:\>
```

创建 bucket



然后是测试部分，采用 `COSBench` 进行测试

`COSBench` 测试：

打开 `cosbench` 监听窗口

直接双击 `start-all.bat` 程序可以打开监听窗口。

```
COSBench Driver - start-driver.bat
G:\0.4.2.c4>java -Dcosbench.tomcat.config=conf/driver-tomcat-server.xml -server -cp main/* org.eclipse.equinox.launcher.Main -configuration conf/.driver -console 18089
Listening on port 0.0.0.0/0.0.0.0:18089 ...
Persistence bundle starting...
Persistence bundle started.
-----
!!! Service will listen on web port: 18088 !!!

COSBench Controller - start-controller.bat
G:\0.4.2.c4>java -Dcosbench.tomcat.config=conf/controller-tomcat-server.xml -server -cp main/* org.eclipse.equinox.launcher.Main -configuration conf/.controller -console 19089
Listening on port 0.0.0.0/0.0.0.0:19089 ...
Persistence bundle starting...
Persistence bundle started.
-----
!!! Service will listen on web port: 19088 !!!
```

用浏览器访问 127.0.0.1:19088/controller/index.html 进入 cosbench 的测试界面。

将 workload-example.xml 提交给 cosbench 进行测试（更新测试样例中的 endpoint、accesskey、secretkey 值。）



接着点击“submit new workloads”即可上传测试文件进行测试，测试部分结果如下：

General Report							
Op-Type	Op-Count	Byte-Count	Avg-ResTime	Avg-ProcTime	Throughput	Bandwidth	Succ-Ratio
op1: init -write	0 ops	0 B	N/A	N/A	0 op/s	0 B/S	N/A
op1: prepare -write	8 ops	64 KB	2877 ms	2876.75 ms	2.79 op/s	22.3 KB/S	100%
op2: prepare -write	8 ops	128 KB	2933.88 ms	2933.62 ms	2.73 op/s	43.67 KB/S	100%
op3: prepare -write	8 ops	256 KB	2921.12 ms	2921 ms	2.75 op/s	88.06 KB/S	100%
op4: prepare -write	8 ops	512 KB	1764.75 ms	1759.62 ms	5.36 op/s	342.88 KB/S	100%
op5: prepare -write	8 ops	1.02 MB	2959.75 ms	2958.38 ms	2.69 op/s	344.51 KB/S	100%
op6: prepare -write	8 ops	2.05 MB	2449.12 ms	2443.88 ms	3.37 op/s	863.44 KB/S	100%
op7: prepare -write	8 ops	4.1 MB	2966.62 ms	2961.62 ms	2.7 op/s	1.38 MB/S	100%
op8: prepare -write	8 ops	8 MB	2857.38 ms	2728.25 ms	2.81 op/s	2.81 MB/S	100%
op1: read	1.98 kops	15.83 MB	6.92 ms	6.79 ms	66.3 op/s	530.39 KB/S	97.83%
op2: write	473 ops	3.78 MB	474.64 ms	474.63 ms	15.85 op/s	126.79 KB/S	100%
op1: read	2.62 kops	41.94 MB	5.65 ms	5.47 ms	88.4 op/s	1.41 MB/S	97.54%
op2: write	637 ops	10.19 MB	348.37 ms	348.25 ms	21.5 op/s	343.99 KB/S	100%
op1: read	2.23 kops	71.33 MB	5.5 ms	5.17 ms	74.56 op/s	2.39 MB/S	95.62%
op2: write	608 ops	19.46 MB	175.37 ms	175.01 ms	20.34 op/s	650.83 KB/S	100%
op1: read	2.3 kops	147.07 MB	5.98 ms	5.27 ms	77 op/s	4.93 MB/S	92.66%
op2: write	610 ops	39.04 MB	170.78 ms	169.67 ms	20.44 op/s	1.31 MB/S	100%
op1: read	940 ops	120.32 MB	9.46 ms	7.75 ms	31.33 op/s	4.01 MB/S	100%
op2: write	229 ops	29.31 MB	91.86 ms	89.16 ms	7.63 op/s	977.07 KB/S	100%

可以看到，COSBench 的测试结果十分齐全，包含了诸如平均处理时间、读写速率和吞吐率等信息。

### 3. 测试 work 值对各项指标的影响

op1: read	1.28 kops	163.2 MB	9.67 ms	7.94 ms	42.56 op/s	5.45 MB/S	100%
op2: write	307 ops	39.3 MB	57.07 ms	53.16 ms	10.25 op/s	1.31 MB/S	100%
op1: read	2.14 kops	273.41 MB	9 ms	7.41 ms	71.26 op/s	9.12 MB/S	100%
op2: write	505 ops	64.64 MB	80.31 ms	76.32 ms	16.85 op/s	2.16 MB/S	100%
op1: read	2.89 kops	370.18 MB	9.31 ms	7.73 ms	96.67 op/s	12.37 MB/S	100%
op2: write	705 ops	90.24 MB	131.25 ms	127.66 ms	23.56 op/s	3.02 MB/S	100%
op1: read	3.08 kops	394.75 MB	10.13 ms	8.66 ms	103.66 op/s	13.27 MB/S	100%
op2: write	761 ops	97.41 MB	271.4 ms	267.78 ms	25.58 op/s	3.27 MB/S	100%
op1: read	3.11 kops	397.57 MB	12.84 ms	11.24 ms	104.64 op/s	13.39 MB/S	100%
op2: write	735 ops	94.08 MB	591.15 ms	587.53 ms	24.79 op/s	3.17 MB/S	100%
op1: read	3.48 kops	445.31 MB	23.1 ms	21.54 ms	121.36 op/s	15.53 MB/S	100%
op2: write	808 ops	103.42 MB	1038 ms	1034.39 ms	28.21 op/s	3.61 MB/S	100%
op1: read	3.72 kops	476.29 MB	16.24 ms	14.77 ms	141.52 op/s	18.11 MB/S	100%
op2: write	828 ops	105.98 MB	1972.86 ms	1969.66 ms	31.5 op/s	4.03 MB/S	100%
op1: read	4.25 kops	543.62 MB	37.11 ms	35.56 ms	165.42 op/s	21.17 MB/S	100%
op2: write	890 ops	113.92 MB	3619.2 ms	3616.02 ms	34.19 op/s	4.38 MB/S	100%

将 workload-example 中的块大小保持 128K 不变，workers 的数量从 1 依次翻倍到 128，保存并重新提交，运行报告如图 10 所示。

从报告中可以看出随着 worker 的增加，其吞吐量，时延和带宽均呈现增加趋势，而增加速度之间减慢，说明 worker 数量尚未到服务器能力上限，性能随 worker 数量增加而提升，若到达饱和状态后，服务器出现拥挤，造成堵塞，就会出现性能降低的情况。

### 4. 测试块大小对各项指标的影响

将 workload-example 中的 workers 数量保持 4 不变，块大小从 8kb 依次翻倍到 1mb，保存并重新提交，运行报告如图 11 所示。

op1: read	2.96 kops	23.7 MB	7.81 ms	7.68 ms	98.91 op/s	791.3 KB/S	100%
op2: write	743 ops	5.94 MB	129.75 ms	129.71 ms	24.8 op/s	198.43 KB/S	100%
op1: read	2.83 kops	45.22 MB	8.27 ms	8.1 ms	94.41 op/s	1.51 MB/S	100%
op2: write	723 ops	11.57 MB	132.7 ms	132.32 ms	24.15 op/s	386.45 KB/S	100%
op1: read	2.92 kops	93.44 MB	7.68 ms	7.48 ms	97.64 op/s	3.12 MB/S	100%
op2: write	732 ops	23.42 MB	132.35 ms	131.3 ms	24.48 op/s	783.33 KB/S	100%
op1: read	2.99 kops	191.42 MB	8.42 ms	7.78 ms	99.92 op/s	6.39 MB/S	100%
op2: write	729 ops	46.66 MB	129.32 ms	127.12 ms	24.35 op/s	1.56 MB/S	100%
op1: read	2.67 kops	341.63 MB	9.99 ms	8.26 ms	89.23 op/s	11.42 MB/S	100%
op2: write	655 ops	83.84 MB	141.61 ms	137.88 ms	21.9 op/s	2.8 MB/S	100%
op1: read	2.22 kops	567.04 MB	10.68 ms	7.47 ms	74.48 op/s	19.07 MB/S	100%
op2: write	544 ops	139.26 MB	174.91 ms	167.47 ms	18.29 op/s	4.68 MB/S	100%
op1: read	1.87 kops	958.98 MB	13.97 ms	7.67 ms	62.5 op/s	32 MB/S	100%
op2: write	449 ops	229.89 MB	208.35 ms	194.16 ms	14.98 op/s	7.67 MB/S	100%
op1: read	1.13 kops	1.13 GB	21.59 ms	9.09 ms	37.96 op/s	37.96 MB/S	100%
op2: write	298 ops	298 MB	317.79 ms	285.89 ms	10 op/s	10 MB/S	100%

从报告中可以看出随着块增大，其读写的处理时延成增大趋势，且增加速度越来越快，说明一开始块较小，处理时延增加不大，而之后超出其处理能力后时延会迅速增加。带宽一直成增加趋势，而吞吐量则先稳定在一个值左右，随和当超出处理能力后快速下降，吞吐量与时延的变化相对应，说明当块较小时，服务器处理能力和带宽相对充足，因此吞吐率基本不变，时延增加不多，而带宽值几乎也是翻倍地增加，但当块增加到一定程度，服务器能力开始饱和甚至不足，吞吐率严重下降。

## 六、实验总结

在本次实验中，主要是搭建了一个简单的分布式对象存储系统 minio，并通过工具对其性能进行了测试分析。Minio 感觉使用起来非常的方便，只要服务器运行着，就可以随时向其中添加文件，由此实现文件的汇总，操作也很简单。而通过测试工具，我也更加直观的了解到时延，吞吐量，带宽这些指标的意义，修改不同的数据，对这些



指标的影响也是很明显的。

这个实验过程中遇到最多的问题还是概念和工具的不熟悉,尤其是命令行操作时,如果不清楚某个参数的含义就容易出现错误。

而通过自己动手从头开始搭建了一次对象存储系统,使我对其工作原理和概念有了更加深刻和具体的了解,总体而言收获还是很大的。

## 参考文献

- [1] ARNOLD J. OpenStack Swift[M]. O'Reilly Media, 2014.
- [2] ZHENG Q, CHEN H, WANG Y 等. COSBench: A Benchmark Tool for Cloud Object Storage Services[C]//2012 IEEE Fifth International Conference on Cloud Computing. 2012: 998–999.
- [3] WEIL S A, BRANDT S A, MILLER E L 等. Ceph: A Scalable, High-performance Distributed File System[C]//Proceedings of the 7th Symposium on Operating Systems Design and Implementation. Berkeley, CA, USA: USENIX Association, 2006: 307–320.