#### Minio安装配置

|  |  |  |  |
| --- | --- | --- | --- |
| 日期 | 版本 | 说明 | 作者 |
| 2020/3/10 | V1.0 | 配置指导书 | 张凯敏 |
|  |  |  |  |

目录

**[1. 源码安装 1](#_Toc1339)**

[1) 安装go环境 1](#_Toc26655)

[2) 安装minio 1](#_Toc28808)

[3) cloud.google.com 无法下载go环境无法下载依赖 1](#_Toc10496)

**[2. Bin安装 1](#_Toc7088)**

**[3. 启动 2](#_Toc31687)**

[1) 设置user key 2](#_Toc19902)

[2) 启动服务 2](#_Toc28949)

[3) 后台数据 2](#_Toc11141)

**[4. 网页验证（mc客户端同步） 2](#_Toc32326)**

**[5. mc 客户端验证(网页端的数据也会同步) 2](#_Toc29059)**

[1) 添加对象存储 2](#_Toc16451)

[2) 创建bucket 3](#_Toc22992)

[3) 查看存储中的bucket 3](#_Toc15917)

[4) 删除bucket 3](#_Toc30387)

[5) 上传文件 3](#_Toc29243)

[6) 删除文件 3](#_Toc20102)

[7) 其他命令通过mc -h查看 3](#_Toc21101)

##### 



##### 简介

1.1概述

Minio对象存储是为海量数据存储、人工智能、大数据分析而设计，完全兼容amazon S3接口，单个对象最大可达到5TB，适合海量存储图片、视频、日志文件、备份数据、虚拟机镜像等，主要采用golang语言实现，整个系统运行在操作系统的用户态空间，访问可通过http/https协议。

1.2特性

部署简单，只有一个可执行文件，启动只需一个命令：minio server /data；本身可扩展支持分布式；无元数据数据库；支持多种成熟的服务的加密模式，支持worm功能，支持用户身份管理。

##### 源码安装

###### 2.1安装go环境

下载go包

https://dl.google.com/go/go1.14.linux-amd64.tar.gz

tar -C /usr/local -xzf go1.14.linux-amd64.tar.gz

将bin添加到环境变量

export PATH=$PATH:/usr/local/go/bin

###### 2.2cloud.google.com 无法下载go环境无法下载依赖

export GOPROXY=https://goproxy.io

###### 2.3安装minio

下载minio源码

https://github.com/minio/minio.git

Make

##### Bin安装

<https://dl.min.io/server/minio/release/linux-amd64/minio>

##### 配置启动

###### 4.1设置user key

export MINIO\_ACCESS\_KEY=zhangkaimin  
 export MINIO\_SECRET\_KEY=12345678

###### 4.3启动服务

4.3.1本地一个磁盘：

minio server /data &

默认配置 [--address localhost:9000 --config-dir ~/.minio --certs-dir ~/.minio/certs]

端口默认是9000；配置默认是在~/.minio；

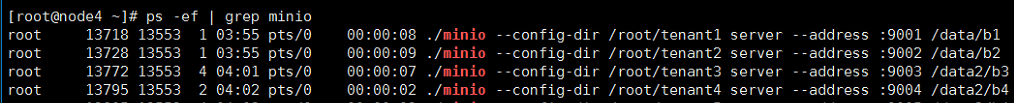
4.3.2本地多个磁盘：

minio --config-dir ~/tenant1 server --address :9001 /data/b1 &

minio --config-dir ~/tenant2 server --address :9002 /data/b2 &

minio --config-dir ~/tenant3 server --address :9003 /data2/b1 &

minio --config-dir ~/tenant4 server --address :9004 /data2/b1 &



###### 4.4多用户策略

4.4.1安装mc服务

4.4.2配置客户端

mc config host add myminio http://66.66.67.111:9000 zhangkaimin 12345678

4.4.3创建策略文件/tmp/readwrite.json

{

"Version": "2012-10-17",

"Statement": [

{

"Action": [

"s3:\*"

],

"Effect": "Allow",

"Resource": [

"arn:aws:s3:::\*"

],

"Sid": ""

}

]

}

4.4.3新增策略

mc admin policy add myminio readwrite /tmp/readwrite.json

4.4.4新增用户

mc admin user add myminio user1 user1234

4.4.5为新用户应用策略readwrite

mc admin policy set myminio readwrite user=user1

4.4.6禁用用户

mc admin user disable myminio user1

4.4.7删除用户

mc admin user remove myminio user1

4.4.7删除策略

mc admin policy remove myminio readwrite

###### 4.5临时用户

###### 4.6修改user key

export MINIO\_ACCESS\_KEY=newzhang

export MINIO\_SECRET\_KEY=87654321

export MINIO\_ACCESS\_KEY\_OLD=zhangkaimin  
 export MINIO\_SECRET\_KEY\_OLD=12345678

重启minio服务

Kill -9 pid; minio server /data &

###### 4.7后台数据

Server启动前存在与根目录下目录会被识别为bucket，根目录下的文件不被识别。Object（或者后台）的操作都应基于bucket（目录）。



##### 网页验证（mc客户端同步）

[http://66.66.67.114:9000/](http://66.66.67.114:9000/minio/)

登录后可创建bucket，上传下载文件

##### mc 客户端验证(网页端的数据也会同步)

###### 6.1安装

Git clone <https://github.com/minio/mc.git>

Make

###### 6.2添加对象存储

mc config host add $minio http://66.66.67.111:9000 zhangkaimin 12345678

mc config host add s3 https://s3.amazonaws.com user key

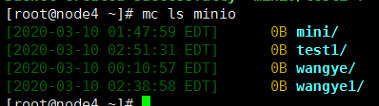
###### 6.3创建bucket

mc mb minio/test1



###### 6.4查看存储中的bucket

mc ls minio



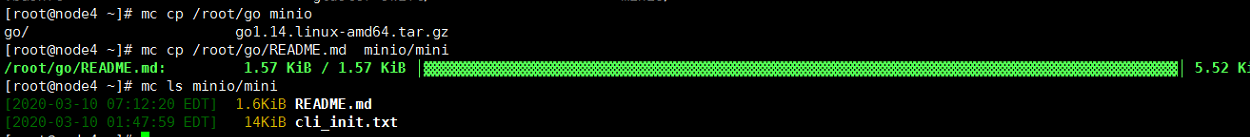
###### 6.5删除bucket

mc rb minio/test1



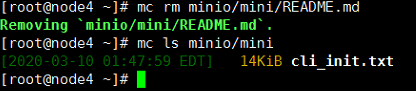
###### 6.6上传文件

mc cp file minio/bucket



###### 6.7删除文件

mc rm minio/mini/file



###### 6.8其他命令通过mc -h查看