

**2016** 级

《物联网数据存储与管理》课程

**实 验 报 告**

**姓 名 潘翔**

**学 号 U201614898**

**班 号 物联网1601班**

**日 期 2019.06.05**

目 录

[1 实验目的 1](#_Toc1759032993)

[2 实验背景 1](#_Toc1362187969)

[3 实验环境 1](#_Toc1883698347)

[3.1 系统环境. 1](#_Toc907500535)

[3.2 Minio-Server 1](#_Toc1639469475)

[3.3 Minio-Client 3](#_Toc127372092)

[3.4 Minio-python API 3](#_Toc1962149094)

[3.5 AWS-CLI 5](#_Toc581167524)

[3.6 s3bench 6](#_Toc71303760)

[3.7 CosBench测试 7](#_Toc2103975916)

[3.8 OpenStack Swift CLI 7](#_Toc104170806)

[3.9 OpenStack Swift Python-API 9](#_Toc365248026)

[4 实验内容 10](#_Toc793160724)

[4.1 对象存储技术实践 10](#_Toc105980445)

[4.2 对象存储性能分析 10](#_Toc7729906)

[5 实验过程 10](#_Toc1844674439)

[6 实验总结 10](#_Toc1832533977)

[参考文献 10](#_Toc2106213184)

# 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。

# 实验背景

# 实验环境

## 系统环境.

## Minio-Server

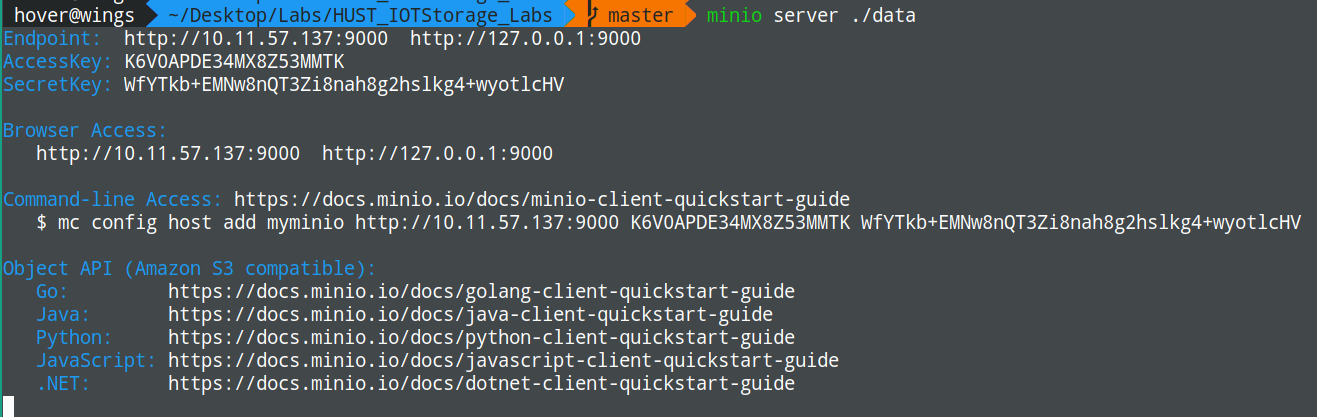


图3.1 Minio-Server启动验证



图3.2 Minio Host添加

此时在浏览器打开127.0.0.1:9000(endpoint)，在打开的minio browser，输入自己的AccessKey和SerectKey登陆。

此时可以选择右下角红色标记随意添加bucket和上传文件，方便地实现类似网盘的功能。

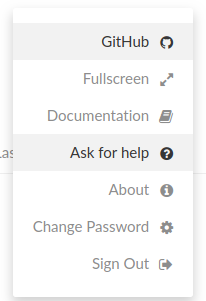


图3.3 账户管理相关选项

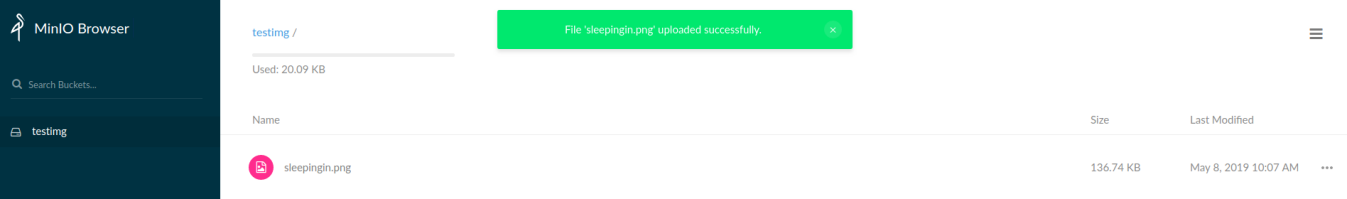


图3.4 Bucket/file测试图

注意，bucket有相应的命名限制，只能采用小写字母，而不允许下划线和相应大写字母的出现。

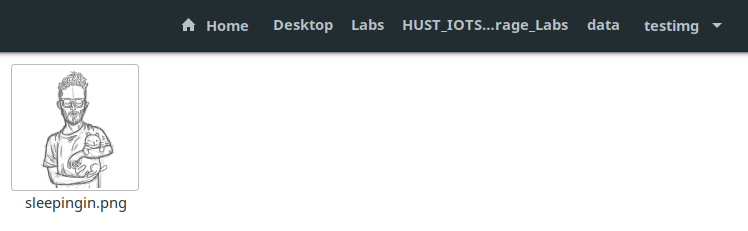


图3.5 本地文件验证

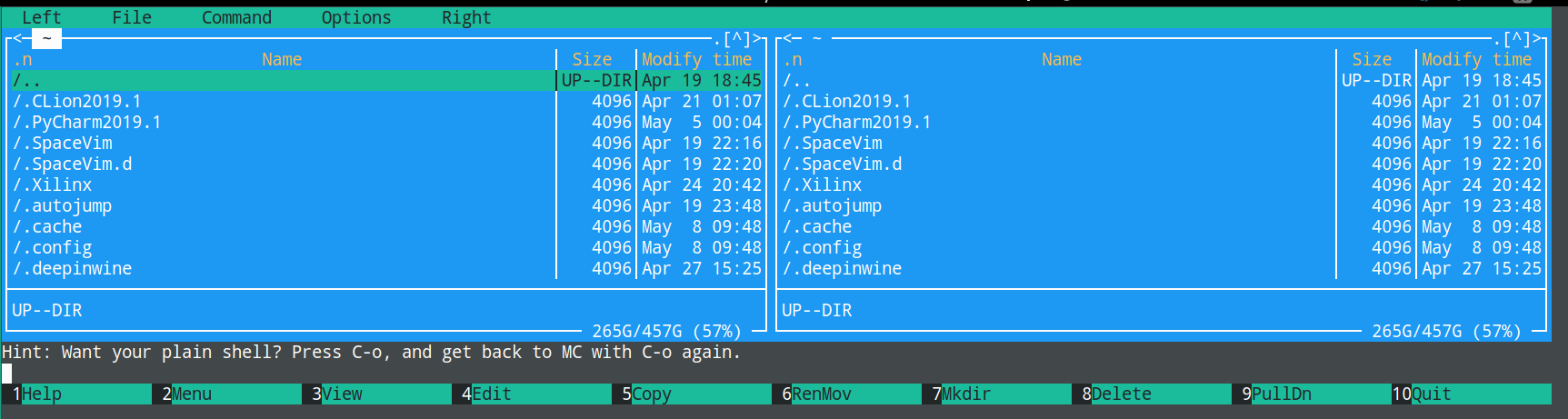


图3.6 MC界面

## Minio-Client

## Minio-python API

使用python-tutorial 进行API编程测试，使用官方测试仓库，代码如下

代码3.1 Minio-python API

from minio import Minio

from minio.error import ResponseError

from minio.error import (ResponseError, BucketAlreadyOwnedByYou,

                         BucketAlreadyExists)

minioClient = Minio('play.min.io:9000',

                  access\_key='Q3AM3UQ867SPQQA43P2F',

                  secret\_key='zuf+tfteSlswRu7BJ86wekitnifILbZam1KYY3TG',

                  secure=True)

# Make a bucket with the make\_bucket API call.

**try**:

       minioClient.make\_bucket("maylogs", location="us-east-1")

except BucketAlreadyOwnedByYou as err:

       pass

except BucketAlreadyExists as err:

       pass

except ResponseError as err:

**raise**

**else**:

        # Put an object 'pumaserver\_debug.log' with contents from 'pumaserver\_debug.log'.

**try**:

               minioClient.fput\_object('maylogs', 'pumaserver\_debug.log', '/tmp/pumaserver\_debug.log')

        except ResponseError as err:

               print(err)

运行结果如下

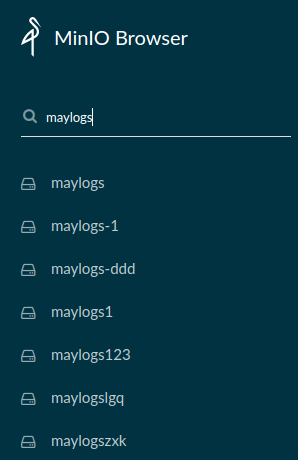


图3.7 Minio-python 测试结果

## AWS-CLI

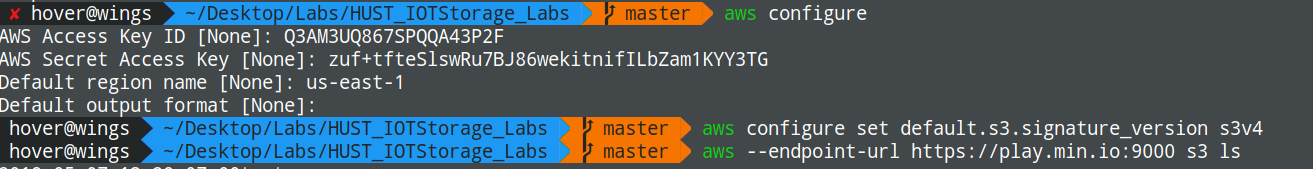


图3.8 使用官方测试AWS-CLI测试

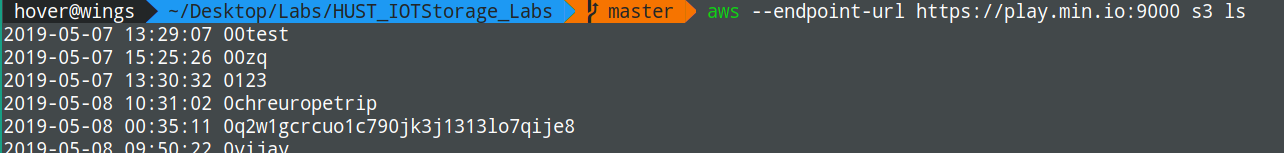


图3.9 此存储下的bucket



图3.10 创建bucket

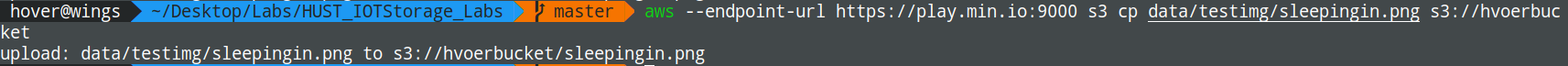


图3.11 上载IMG测试



图3.12 删除IMG测试

## s3bench

使用如下命令进行测试，其中参数可自行调节

go run ./s3bench.go -accessKey=KEY -accessSecret=SECRET -bucket=loadgen -endpoint=http://endpoint1:80,http://endpoint2:80 -numClients=2 -numSamples=10 -objectNamePrefix=loadgen -objectSize=1024

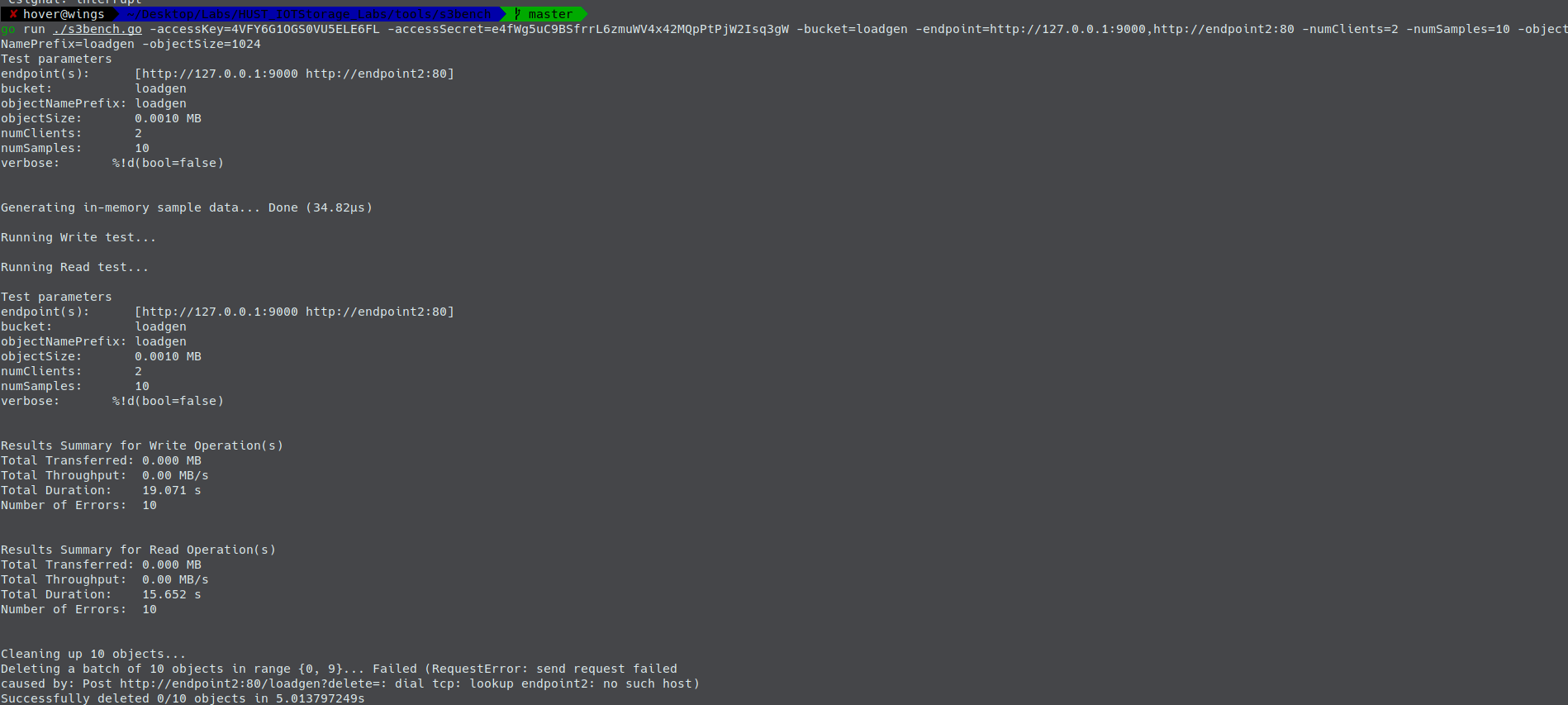
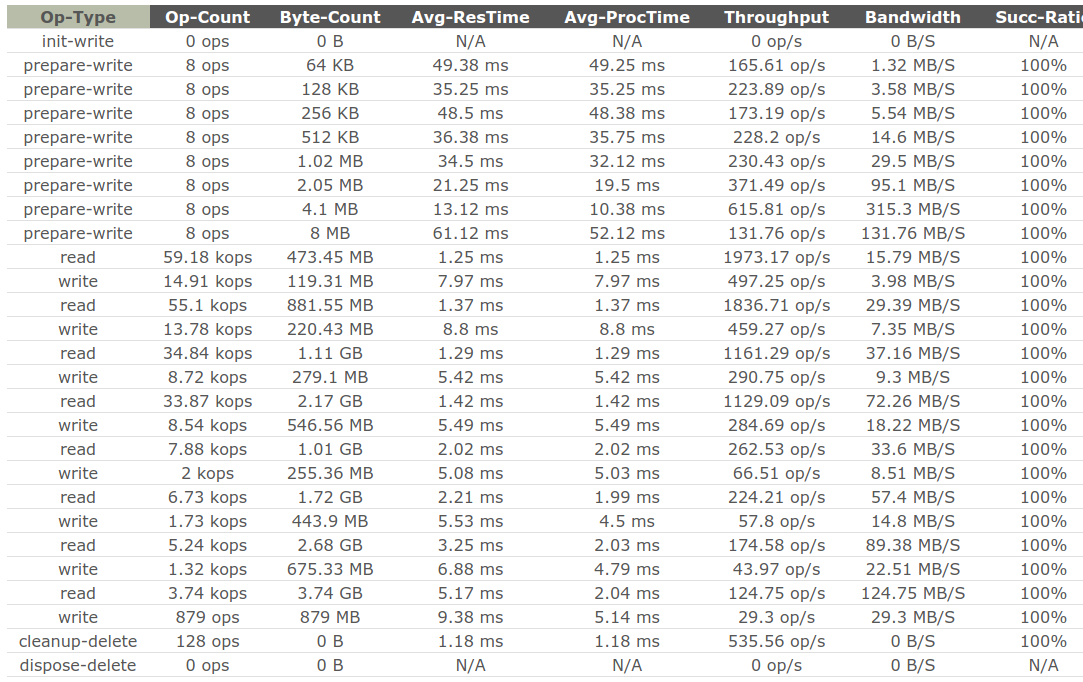


图3.13 s3bench测试

## CosBench测试



## OpenStack Swift CLI

使用OpenStack Swift进行相关测试

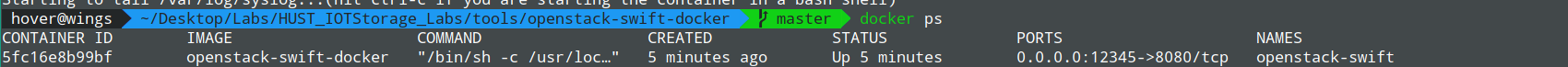


图3.14 Swift Docker启动

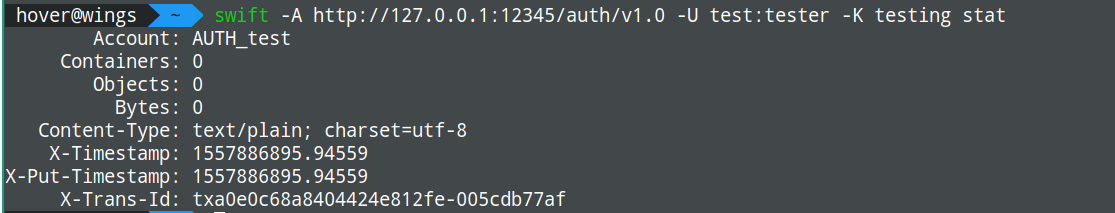


图3.15 Swift Docker启动检测

设置系统环境变量，避免每次命令行需要输入，由于采用V1.0验证，具有和V2.0不同的环境变量配置，需要注意。

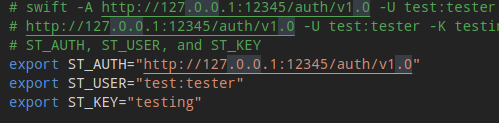


图3.16 SwiftAUTH环境变量设置

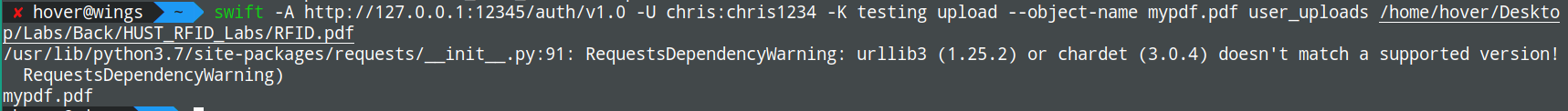


图3.17 Swift上传测试

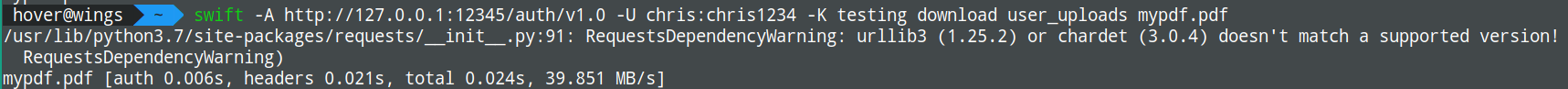


图3.18 Swift下载测试

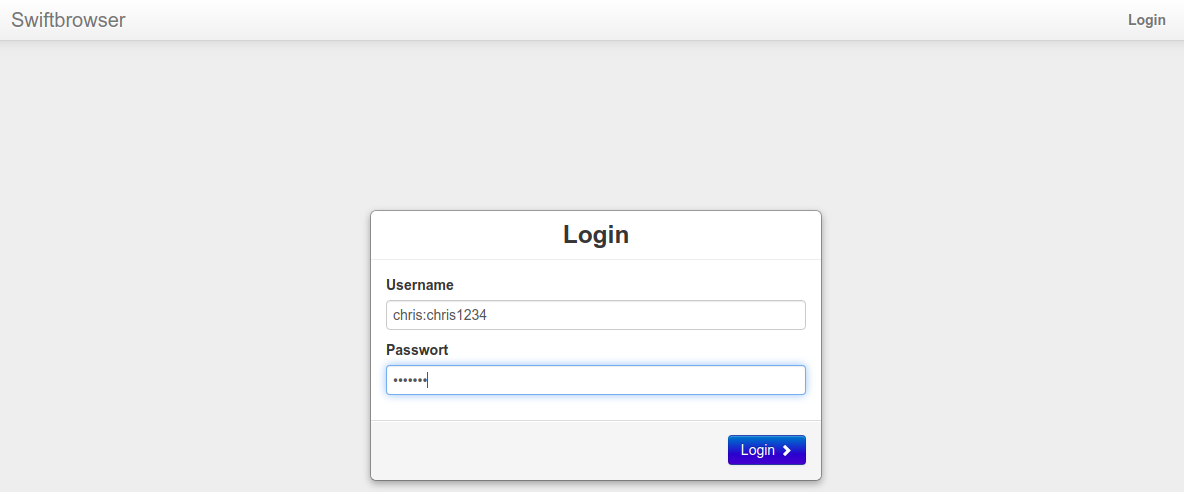


图3.19 Swift GUI-Client 测试

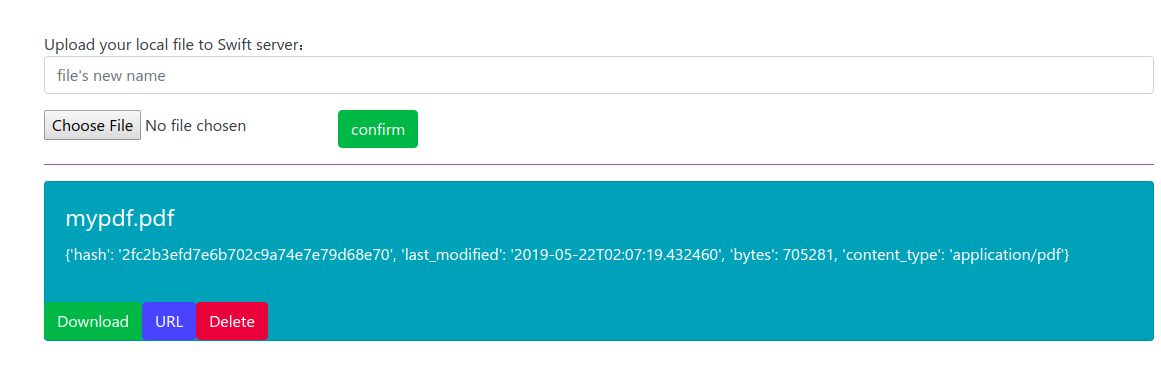


图3.20 Swift GUI测试

## OpenStack Swift Python-API

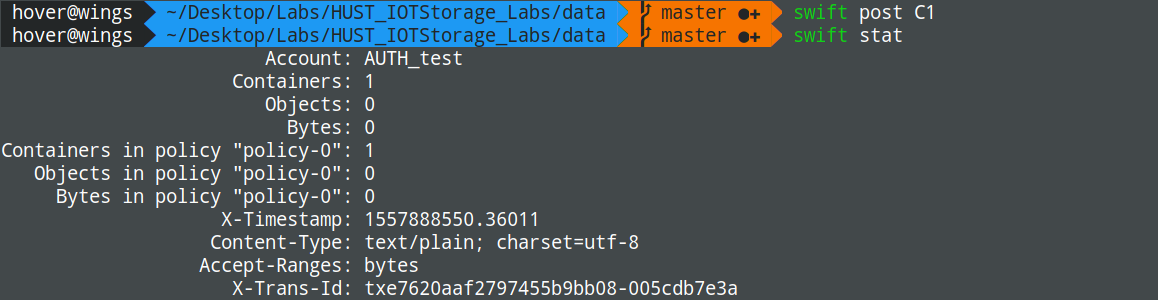


图3.21 Swift创建Container

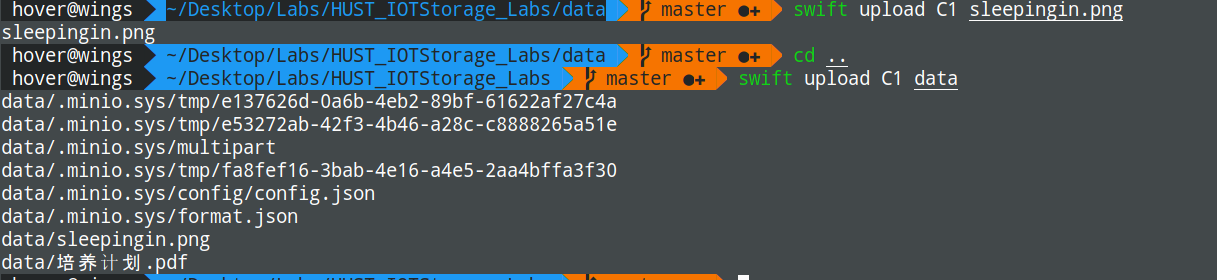


图3.22 Swift上传文件和文件夹测试

# 实验内容

## [对象存储技术实践](#_Toc509412099)

## 对[象存储性能分析](#_Toc509412100)

# 实验过程

# 实验总结

# 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.