

## 1. 实训目的

本次实训的主要目的是锻炼自己的动手能力,通过实践操作培养自己理论结合实际的能力,增加项目经验,提高自己独立解决问题的能力 and 科研能力。实训的目标是在 swift 开源云存储系统上,设计并实现一个 PC 端客户端,实现数据的上传下载,访问策略的制定以及加解密上传下载等过程。

## 2. 实训时间和地点

实训时间: 2014 年 3 月-6 月

实训地点: 武汉光电国家实验室

## 3. 实训单位简介

华中科技大学“数据存储与信息服务实验室”,依托华中科技大学计算机科学与技术学院“计算机外存储系统国家专业实验室”暨“信息存储系统教育部重点实验室”和武汉光电国家实验室“信息存储与光显示功能实验室”,拥有自由开放的学术氛围和国际前沿的研究方向,现主要从事云存储及应用、存储安全、大数据分析、数据保护、磁记录、半导体存储等方面的研究。

该实验室近年来先后主持和参与了国家发改委、科技部、教育部、工信部、国家自然科学基金委等多项科研项目,包括国家 973 项目课题、973 前期研究专项、国家自然科学基金重点项目和面上项目、863 重大项目、六五和七五科技攻关项目、国防预研基金课题、教育部博士点基金项目、国家发改委信息安全专项、物联网发展专项等,取得了一批具有国际先进或国际领先水平的重大科研成果,获得国家或部委的多项奖励。

## 4. 实训内容

本次实训所涉及的内容是设计并实现 swift 云存储系统的 PC 客户端,主要包含以下工作:设计客户端界面,支持文件的加解密上传下载,明文长传下载,支持文件的访问策略制定,并预留后续审计模块的接口。

### 4.1 Swift 简介及环境搭建

#### 4.1.1 swift 简介

Swift 最初是由 Rackspace 公司开发的高可用分布式对象存储服务，并于 2010 年贡献给 OpenStack 开源社区作为其最初的核心子项目之一，为其 Nova 子项目提供虚拟机镜像存储服务。Swift 构筑在比较便宜的标准硬件存储基础设施之上，无需采用 RAID（磁盘冗余阵列），通过在软件层面引入一致性散列技术和数据冗余性，牺牲一定程度的数据一致性来达到高可用性和可伸缩性，支持多租户模式、容器和对象读写操作，适合解决互联网的应用场景下非结构化数据存储问题。此项目是基于 Python 开发的。

Swift 采用完全对称、面向资源的分布式系统架构设计，所有组件都可扩展，避免因单点失效而扩散并影响整个系统运转；通信方式采用非阻塞式 I/O 模式，提高了系统吞吐和响应能力。如图 1 显示 swift 系统架构。

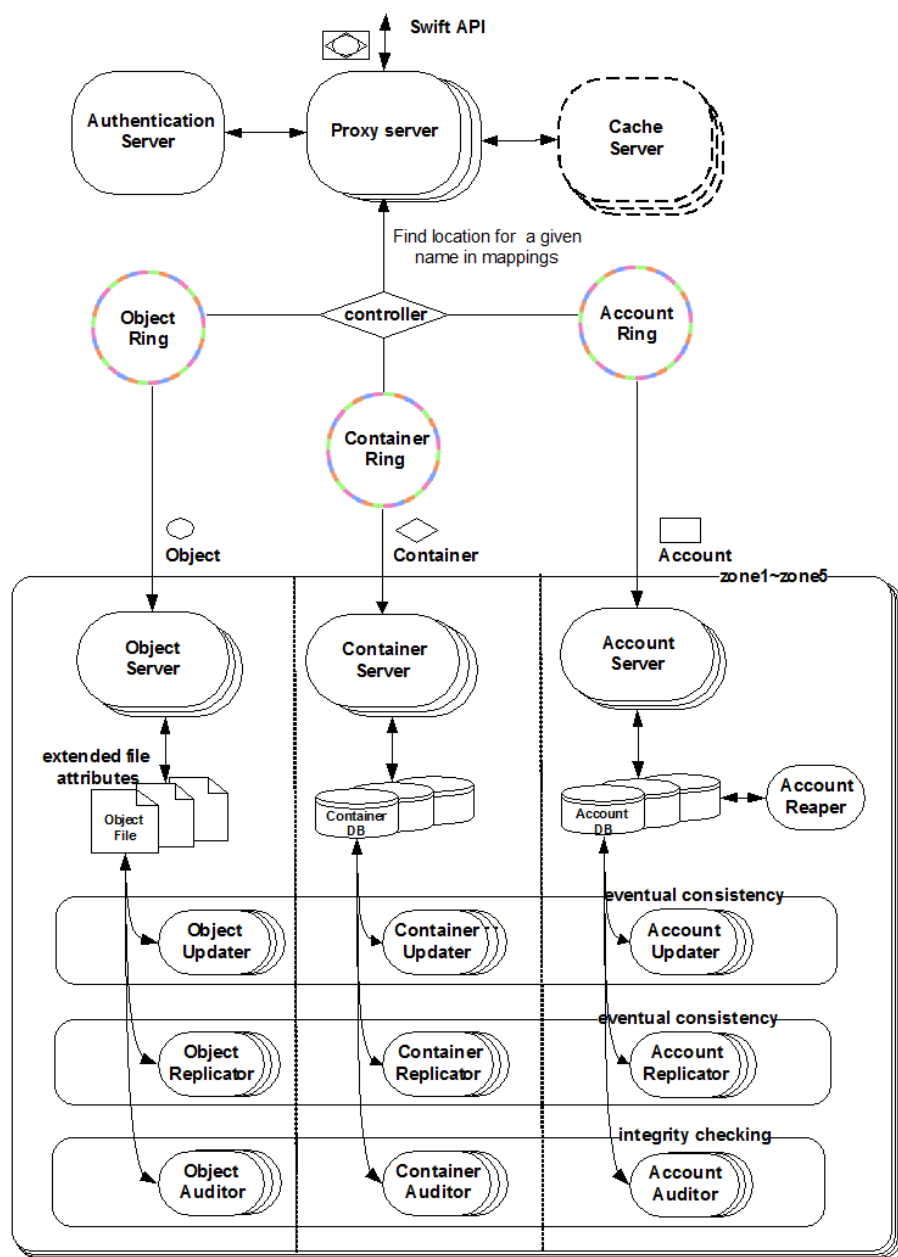


图 1 swift 系统架构图

### 4.1.2 swift 平台搭建

本项目基于 swift 云存储平台进行开发，因此首先需要搭建 swift 开发平台。其中主要需要搭建的模块分别为 swift 存储服务器，元数据服务器，认证授权服务器，其中我们需要对认证授权服务器的相关内容进行扩展和修改，以满足本次实训目的之一，实现基于属性的访问控制。

实验平台为 ubuntu12.10 64bit，利用 python 作为开发语言，界面端采用 PyQt 进行可视化开发，eclipse 作为编程 IDE，使用 python2.7 版本进行编译运行。

- 1) 安装 python2.7 编程开发工具
- 2) 根据 swift 官方文档搭建 swift 平台
- 3) 配置 eclipse IDE 编程环境
- 4) 安装 pyqt dev 开发环境

## 4.2 登录注册模块

登录注册模块主要用于用户的判断和授权，客户端将用户名和口令通过安全信道发送给 swift 存储系统的授权认证模块，若认证通过则接受认证模块返回的令牌，该令牌将作为用户后续操作的身份标识。登录端界面如图 1

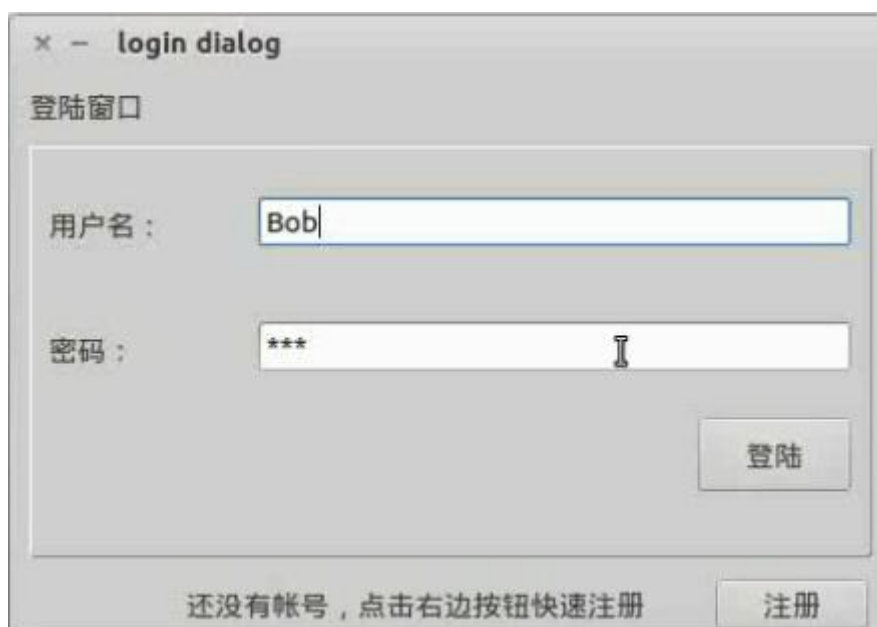


图 2 登录界面

## 4.3 主界面

主界面主要向用户提供一个显示用户数据的可视化视图，本客户端利用两个

窗口分别显示文件目录和文件，文件目录以及文件都支持增删改等基本功能，图 2 显示了用户主界面，从中可以看出，主界面主要分为三个不同的工作空间，个人空间，群组空间和公共空间。

其中个人空间主要存放用户私人数据，可以作为用户的云端备份使用；群组空间相比较个人空间，其主要区别在与群空间内数据可以实现群组成员之间的数据共享，系统支持基于属性的访问权限判断方法，实现对用户的授权处理；公共空间则是面向所有用户的免费数据共享以及采用扩展的基于动态属性的访问权限的数据共享

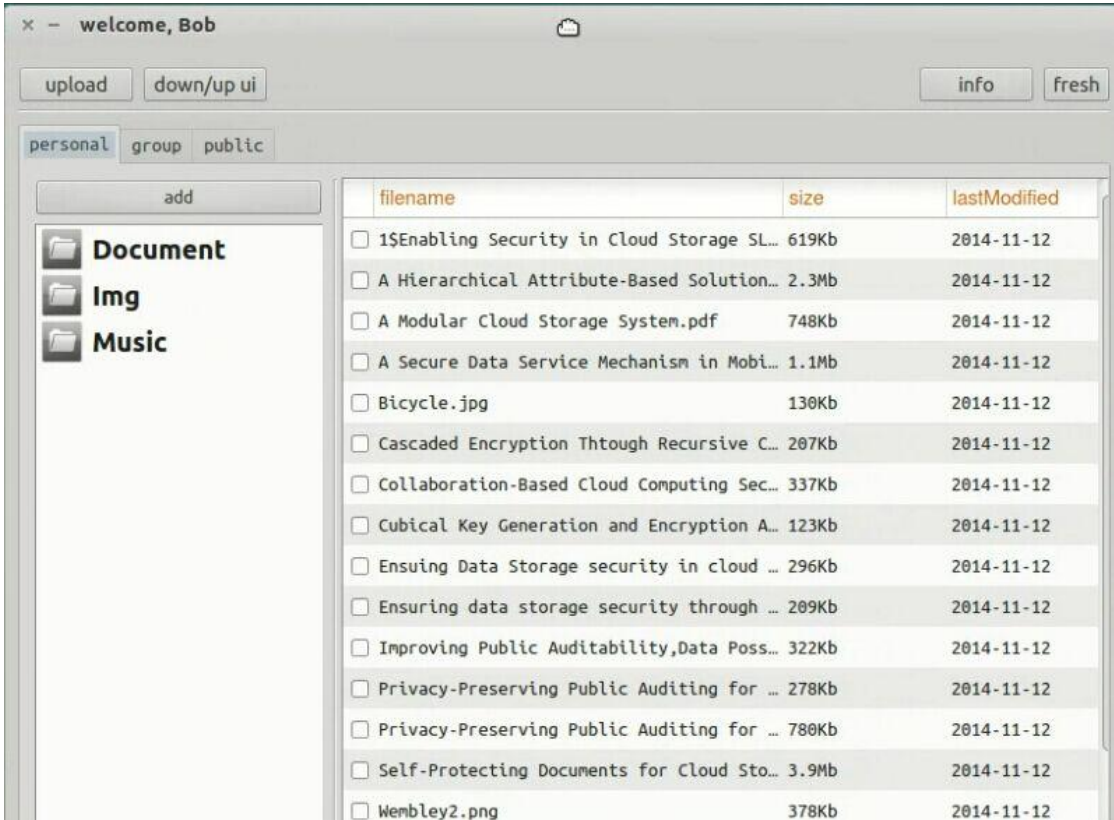


图 3 主界面

#### 4.4 访问策略的制定

本次实训的客户端主要是在基于属性的访问控制模型基础上实现用户权限判断，访问策略是由属性与属性之间通过逻辑词进行组合形成，是一种形如公式 1 的表达式，访问策略制定界面如图 3 所示

公式一：  $(attr1 \text{ and } attr2) \text{ or } (attr3 \text{ and not}(attr4 \text{ or } attr5))$

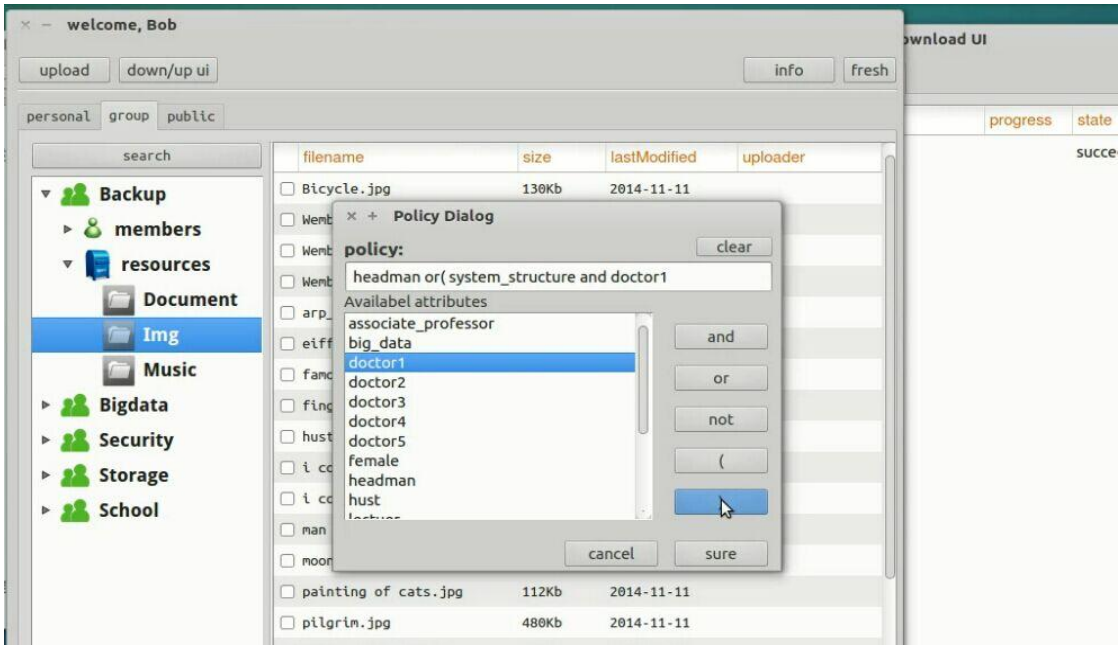


图 4 访问策略制定

4.5 上传下载界面

上传下载界面则实现文件的上传下载，其界面如图 4 所示，当确定上传或下载文件时，客户端将封装命令格式，并与 swift 云存储系统建立连接，若权限判断通过则建立数据传输通道，实现数据的传输。

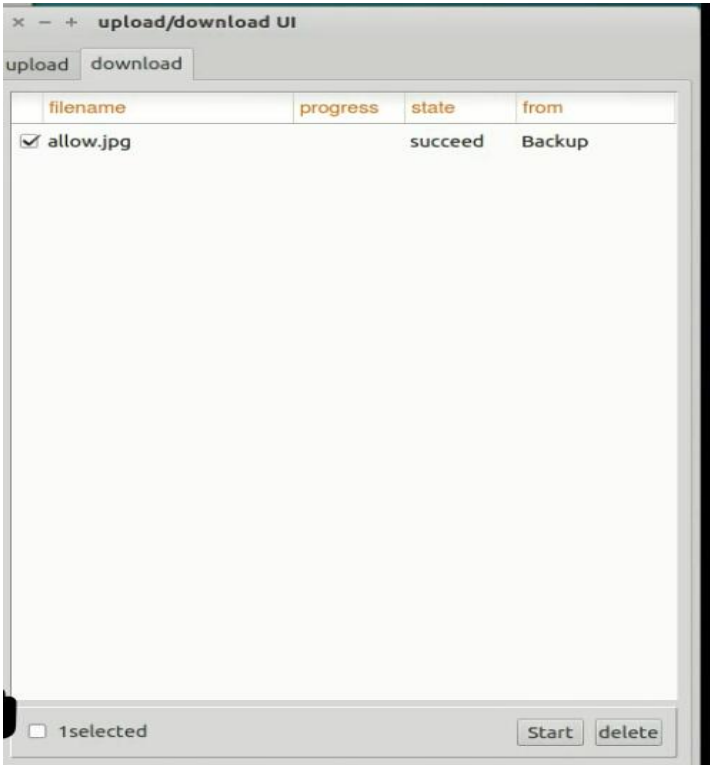


图 5 上传下载界面

## 5. 实训收获

通过此次专业实训，一方面让我对 swfit 云存储系统的有了较深的了解，并在客户端的开发过程中，清楚了应用程序的开发流程，从需求出发，设计客户端所需满足的功能要求，并学习软件设计模式，提供代码的质量。

此外，在本次实训中，也遇到了很多的问题，如由于边界效应引起的程序偶然性崩溃问题，代码的扩展性问题等，在实际的编程规范中，应该充分的考虑的代码的健壮性，而且需要考虑高内聚低耦合，实现模块的封装，以此为后续的扩展减少工作量。

研 究 生 签 字 \_\_\_\_\_

指 导 教 师 签 字 \_\_\_\_\_

院(系、所)领导签字 \_\_\_\_\_

年      月      日