

对象存储 US3

产品文档

目录

目录	2
概览	28
产品概述	32
主要概念	32
相关服务	35
使用 US3	35
US3 定价	35
功能概览	37
存储功能	37
管理功能	38
增值功能	39
产品优势	40
稳定可靠的存储能力	40
低成本的弹性存储空间	40
灵活便捷的业务接入	40
简单易用的管理工具	40
安全可靠的分布式系统	41
多样化的云应用场景	41
使用限制	42

支持地域	42
HTTPS 访问限制	43
域名访问限制	43
上传文件大小限制	43
图片处理限制	44
其他限制项	44
存储类型	46
标准存储	46
低频存储	46
归档存储	47
存储类型转换	47
存储类型支持的 API	48
地域和域名	50
计量计费	52
US3收费模式	52
产品价格	54
中国大陆	54
香港、台北及海外	55
补充说明	57
计费规则	61

计费项目	61
计费方式	61
补充说明	62
欠费处理	63
欠费对 US3 服务造成的影响	63
欠费处理时间轴	64
计费案例	65
案例一	65
案例二	65
案例三	66
资源包	67
计费说明	67
购买	68
升级	71
续费	73
使用明细	74
退款规则	75
开通 US3 服务	76
使用 US3	76
开始使用 US3	78

使用 UCloud 管理控制台	78
使用命令行管理工具 US3 Tools	79
使用 API/SDK	79
存储空间	80
创建存储空间	80
删除存储空间	83
查看存储空间	86
修改存储空间类型	88
存储空间命名必须符合名称规范,规范如下:	90
日志管理	91
设置日志存储	91
日志命名规则	94
日志格式和示例	94
日志示例	96
文件管理	98
上传文件	100
修改 MIME-Type	101
修改存储类型	103
删除文件	105
下载文件	108
重命名文件	111
获取地址	112

设置HTTP头	116
生命周期	120
使用场景	120
注意事项	121
管理生命周期	121
计费说明	125
备注	126
静态网站托管	127
概念	127
使用说明	127
举例说明	128
域名管理	130
绑定自定义域名	130
解绑自定义域名	135
镜像回源	137
使用场景	137
设置回源规则	137
备注	144
跨区域复制	145
使用场景	145

注意事项	145
设置跨区域复制规则	146
备注	148
跨域访问	149
设置跨域访问策略	149
备注	152
防盗链	153
设置防盗链	153
统计报表	156
令牌管理	159
创建/编辑令牌	159
示例	161
对象标签	163
使用说明	163
设置对象标签	163
计费说明	166
简介	168
图片处理参数	168
使用限制	168

计费说明	168
图片处理参数	170
功能支持	170
基本图片信息获取	171
EXIF信息获取	171
图片缩放	173
图片裁剪	175
索引切割	176
内切圆	176
圆角矩形	177
图片旋转	177
自适应方向	178
亮度	178
对比度	178
锐化	179
模糊	179
获取图片主色调	180
水印	180
图片格式转换	182
渐进显示	183
管道顺序调用多种图片处理功能	183
图片样式	184
创建样式	184
使用样式	190

编辑样式	191
删除样式	191
注意事项	192
原图保护	193
访问方式	193
操作步骤	193
注意事项	197
解压缩服务	199
注意事项	199
配置解压缩	199
修改规则	204
删除规则	207
文档预览	209
概念	209
注意事项	209
调用参数	209
控制台预览	210
开放地域	213
内容审核服务	214
概览	214
操作步骤	214

参数说明表	218
开放地域	219
计费规则	220
常用工具	221
管理工具	221
工具简介	222
概述	222
版本和运行环境	222
下载与环境准备	223
下载地址	223
环境准备	224
快速上手	226
视频教学	226
配置管理	226
创建存储空间	231
上传文件或文件夹	233
下载文件或文件夹	236
拷贝文件或文件夹	238
删除文件或文件夹	238
查询文件基本信息	240
获取文件列表	240

文件基本信息修改	241
常用命令	243
config	244
mb	251
rb	253
stat	254
du	256
cp	258
sync	263
rcat	268
mkdir	270
cat	271
mv	273
rm	274
ls	277
sign	280
modify	281
restore	283
etag	284
create-token	286
delete-token	287
update-token	288
describe-token	290
update	292
version	293

常见问题	295
报错 Open leveledb err: resource temporarily unavailable	295
Bucket操作如stat、ls、du报错 bucket not found	295
并发数parallel和qps有什么区别?	296
上传文件不指定文件key, 上传成功但找不到文件	296
将本工具放入bin目录下全局使用时, 使用update命令提示找不到文件的问题	297
报错 Signature VerifyAC Error	297
1.7.3	299
新增	299
1.7.2	300
修复	300
1.7.0	301
新增	301
1.6.2	302
修复	302
1.6.1	303
修复	303
1.6.0	304

新增	304
1.5.2	305
修复	305
1.5.1	306
新增	306
1.5.0	307
修改	307
新增	307
1.4.0	308
新增	308
1.3.0	309
修改	309
新增	309
1.2.2	310
修改	310
新增	310
1.2.0	311
新增	311

修改	311
删除	311
工具简介	312
概述	312
版本和运行环境	312
主要功能	313
使用限制	314
下载与环境准备	315
运行环境	315
下载链接	315
配置账号访问信息	316
快速上手	319
视频教学	319
使用方式	320
使用帮助	323
配置文件对应终端参数	328
选项列表	331
使用示例	336
场景化参数设置	339
自动挂载	341
常见问题	343

适用于所有挂载失败的情况	343
挂载失败，日志出现“internal error”	343
挂载时出现403	343
异常退出后再次挂载报错	343
读写发生错误	344
当前用户挂载后其他用户无法访问	344
挂载后发现跟控制台文件信息不一致	344
降低读写时延的方法	344
内存使用量高，导致OOM	345
控制台有对应的目录，但是us3fs挂载路径没有	345
系统日志出现‘too many open files’	346
历史版本	347
US3FS v2.1.0	347
US3FS v2.0.9	347
US3FS v2.0.8	347
US3FS v2.0.7	348
US3FS v2.0.6	348
US3FS v2.0.5	348
US3FS v2.0.3	348
US3FS v2.0.2	349
US3FS v2.0.1	349
US3FS v2.0.0	349
US3FS V1.7.12	350
US3FS V1.7.11	350
US3FS V1.7.10	350

US3FS V1.7.9	350
US3FS V1.7.8	351
US3FS V1.7.7	351
US3FS V1.7.6	351
US3FS V1.7.5	352
US3FS V1.7.4	352
US3FS V1.7.3	353
US3FS V1.7.2	353
US3FS V1.7.1	353
US3FS V1.7.0	353
US3FS V1.6.8	354
US3FS V1.6.7	355
US3FS V1.6.6	355
US3FS V1.6.5	356
US3FS V1.6.4	356
US3FS V1.6.3	356
US3FS V1.6.2	356
US3FS V1.6.0	357
US3FS v1.5.5	357
US3FS v1.5.4	358
US3FS v1.5.3	359
US3FS v1.5.2	360
US3FS v1.5.1	360
US3FS v1.5.0	360
US3FS v1.4.0	361

工具简介	362
概述	362
主要功能	369
文件结构	369
与原迁移工具对比	370
版本和运行环境	370
下载与环境准备	372
运行环境	372
下载并解压	372
启动master服务	373
添加worker节点	373
视频参考	374
快速上手	375
视频教学	377
界面说明	377
使用说明	385
缓存说明	396
迁移类型说明	396
常见问题	398
master部署后，浏览器无法看到界面	398
worker部署后，启动任务后正在迁移的任务一直显示为0	398
任务有报错，在导出的任务日志中，报错为"Client.Timeout or context cancellation"	399

在导出的任务日志中, 报错状态码为403	399
v1.10.4	401
v1.10.1	402
v1.10.0	403
v1.9.1	404
v1.9.0	405
v1.8.0	406
v1.7.0	407
v1.6.1	408
v1.6.0	409
v1.5.0	410
v1.4.0	411
v1.3.1	412

v1.3.0	413
v1.2.0	414
v1.1.2	415
v1.1.1	416
v1.1.0	417
v1.0.0	418
工具简介	419
概述	419
原理说明	419
版本和运行环境	420
下载与环境准备	421
下载与安装	421
安装适配器	421
快速上手	423
视频教学	423
使用说明	423
场景示例	430

常见问题	434
对US3进行HDFS命令操作时报错	434
对US3进行操作时报错	434
修复	434
新增	435
新增	435
重构	435
功能	435
修复	436
功能	436
修复	436
优化	436
功能	437
工具简介	438
概述	438
原理说明	438
运行环境	440
下载与环境准备	441
下载与安装	441
高可用集群部署	442
快速上手	444

使用说明	444
性能数据与分析	449
测试环境	449
测试数据	451
缺陷	467
常见问题	468
内存消耗情况如何?	468
us3vmds启动后,发现通过hadoop命令看到的目录下文件与us3控制台上看到不一致,该怎么办?	468
工具简介	469
概述	469
原理说明	469
运行环境	470
下载与安装	471
快速上手	472
参数说明	472
使用同步	474
使用校验	482
增量迁移	487
缺陷	487

常见问题	488
工具简介	489
概述	489
ClickHouse介绍	489
下载与配置使用	490
源码下载	490
编译	490
配置与使用	491
性能分析	494
US3Browser 下线说明	500
概述	500
版本和运行环境	500
主要功能	501
迁移工具（历史版本）	502
介绍	502
安装步骤	503
创建存储空间	504
操作方式	504
设置空间读写权限	505

获取存储空间信息	506
操作方式	506
删除存储空间	507
操作方式	507
上传文件	508
简单上传	508
表单上传	509
分片上传	509
上传后续操作	511
上传回调	511
下载文件	512
简单下载	512
断点续传下载	512
查看文件列表	512
删除文件	513
API 签名算法	514
空间管理签名算法	514
文件管理签名算法	515
文件 ETag 生成方法	520
签名工具	521

在URL中包含签名	522
注意事项	522
签名实现	522
上传策略说明	524
US3上传回调	524
用法示例	525
AWS S3 协议支持说明	527
概述	527
支持的 API	527
访问域名 (Endpoint)	532
回调扩展功能支持	533
图片操作支持	534
存储类型转换规则	534
AWS S3 协议应用场景	535
文件浏览器工具	535
网络文件系统 S3FS	541
goofys	548
基于 US3 的 FTP 服务	550
s3cmd	552
rclone	558
Cyberduck	560

API 列表	564
管理类 API	564
数据类 API	564
SDK 列表	566
在微信小程序端上传文件到US3	567
简介	567
预先准备	567
上传步骤	567
如何将US3上的文件打包下载	575
简介	575
预先准备	575
基本原理	575
操作步骤	576
性能测试	579
数据迁移方案	580
背景	580
US3 镜像回源功能	580
US3 数据迁移工具	581
数据库备份方案	582
背景	582

应用场景	582
方案优势	583
方案实施	584
数据归档方案	590
背景	590
US3 归档类型	590
使用 US3 归档类型存储冷数据	590
归档存储适用场景	593
使用STS临时访问凭证访问US3	595
前提条件	595
适用场景	595
步骤一：创建子用户	596
步骤二：创建用于获取临时访问凭证的角色	597
步骤三：为角色授予上传文件的权限	598
步骤四：获取临时访问凭证	600
步骤五：使用临时访问凭证上传文件至US3	602
归档转标准/低频存储类型	603
背景	603
方案	603
参考代码	603
测试用例	608

常见问题	610
对象存储空间和 Key 是什么?	610
公共空间和私有空间有何区别?	610
如何查看及管理上传的文件?	610
是否支持目录、文件列表?	610
如何使用对象存储空间提供的两个域名?	611
我可以存储多少数据? 对象存储空间是否有容量上限?	611
文件大小有什么限制?	611
对象存储空间如何通过内网访问?	611
如何删除大量对象?	612
US3 域名被第三方平台告知有安全风险怎么处理?	612
有跨域需求时如何申请?	612
CDN 回源 US3 的流量如何计费?	613
镜像回源支持的空间类型	613
我的账号为何无法进行存储空间或文件操作?	613
如何存储额外的文件元数据信息?	614
子帐号授权后进入文件管理页面提示: 非法授权	614
为什么使用Firefox浏览器时, 无法批量解冻以及下载已解冻的归档文件等解冻相关操作?	617

概览

- 产品简介
 - 产品概述
 - 功能概览
 - 产品优势
 - 使用限制
 - 存储类型
 - 地域和域名
- 计量计费
 - 产品价格
 - 计费规则
 - 欠费处理
 - 计费案例
 - 资源包
- 快速上手
 - 开通 US3 服务
 - 开始使用 US3
- 控制台指南
 - 存储空间
 - 日志管理
 - 文件管理
 - 生命周期
 - 静态网站托管
 - 域名管理
 - 镜像回源
 - 跨区域复制
 - 跨域设置
 - 防盗链
 - 统计报表

- 令牌管理
- 对象标签
- 增值服务
 - 图片处理服务
 - 解压缩服务
 - 文档预览
- 常用工具
 - US3CLI 命令行工具
 - 工具简介
 - 下载与环境准备
 - 快速上手
 - 常用命令
 - 常见问题
 - 历史版本
 - US3FS 文件挂载工具
 - 工具简介
 - 下载与环境准备
 - 快速上手
 - 常见问题
 - 历史版本
 - US3SYNC 迁移工具
 - 工具简介
 - 下载与环境准备
 - 快速上手
 - 常见问题
 - 历史版本
 - US3Hadoop 大数据适配工具
 - 工具简介
 - 下载与环境准备
 - 快速上手
 - 常见问题
 - 历史版本
 - US3VMDS 索引加速工具(公测)

- 工具简介
- 下载与环境准备
- 快速上手
- 性能数据与分析
- 常见问题
- US3Distcp 大数据备份工具(公测)
 - 工具简介
 - 下载与安装
 - 快速上手
 - 常见问题
- US3ClickHouse 插件(公测)
 - 工具简介
 - 下载与配置使用
 - 性能分析
- US3Browser 可视化界面工具(已下线)
- 其他工具
 - 原迁移工具
- 开发者指南
 - 存储空间(Bucket)
 - 创建存储空间
 - 获取存储空间信息
 - 删除存储空间
 - 对象/文件(Object)
 - 上传文件
 - 下载文件
 - 鉴权方法说明
 - API 签名算法
 - 签名工具
 - 上传策略说明
 - AWS S3 协议支持
 - AWS S3 协议支持说明
 - AWS S3 协议应用场景
- API 列表

- SDK 列表
- 解决方案与最佳实践
 - 微信小程序端上传方案
 - US3文件打包下载方案
 - 数据迁移方案
 - 数据库备份方案
 - 数据归档方案
 - 使用STS临时访问凭证访问US3
- 常见问题

产品概述

对象存储 (US3) 是为互联网应用提供非结构化数据存储的服务。相对于传统硬盘存储, 对象存储具有存储无上限、支持高并发访问、成本更低等优势。其数据持久性不低于 99.999999999%, 标准型存储服务可用性不低于 99.95%。

您可以使用 US3 提供的 API、SDK 接口或者 US3 迁移工具轻松地将海量数据移入或移出 US3。数据存储到 US3 以后, 您可以选择标准存储类型的 US3 服务作为移动应用、大型网站、图片分享或热点音视频的主要存储方式; 也可以选择成本更低、存储期限更长的低频存储类型和归档存储类型的 US3 服务作为不经常访问数据的备份和归档。

该内容为动态文件, 无法展示

主要概念

对象存储空间 (Bucket)

对象存储空间 (简称存储空间) 是文件的组织管理单位, 一个文件必然隶属于某个空间中。空间名称全局唯一, 且无法进行修改。每个账号最多可以创建 20 个存储空间, 存储空间内文件数量无限制。用户可以将存储空间设置为公开或私有, 以控制存储空间内文件的访问权限。

私有空间

所有文件所有操作须获得拥有者的 API 密钥授权才能访问。

公开空间

所有文件下载可通过 URL 直接访问。上传、删除、列表还是需要API密钥授权才能访问。

对象/文件 (Object)

文件是存储空间的逻辑存储单元。对于每个账号,该账号里存放的每个文件都有唯一的一对存储空间(Bucket)与键(Key)作为标识。

文件名 (Key)

文件名是对应文件的名称,在存储空间中全局唯一,每个文件名在存储空间均标识了一个文件,写入文件时,用户可以自定义文件名。上传同样文件名的文件,会导致原文件名文件被覆盖。使用者在下载文件时只需要知道下载出口的域名,而无需了解文件具体会被存放到哪个机房的哪个设备,也无需知道具体存放形式。只需在浏览器输入对应的URL访问即可。

文件名命名规范

- 1.使用 UTF-8 编码
- 2.长度必须在 1-1023 字节之间
- 3.可以“/”字符开头,但不允许出现“{}^[]<>#~%”。

访问域名 (Endpoint)

Endpoint 表示US3对外服务的访问域名。US3以HTTP RESTful API的形式对外提供服务,当访问不同地域的时候,需要不同的域名。通过内网和外网访问同一个地域所需要的域名也是不同的。具体的内容请参见地域和域名。

Token密钥 (API Access)

Token 密钥,是一对公私钥;用户可以创建 Token 密钥,针对 bucket 授予不同 Token 权限,不同 Token 可以分发给不同的用户,实现对Bucket 的细分权限管理。另外,Token 密钥可以

设置有效期,也可以随时删除,确保访问 Bucket 的安全性;

API密钥 (API Access)

用户在注册 UCloud 账号后,系统会为用户生成 API 密钥,用以标识用户。API 密钥用于调用 API 时进行身份验证,防止他人恶意篡改您的请求数据,如密钥泄露,请立即重置,重置成功后需要退出网站再登录。API 密钥包含公钥与私钥两部分。在 API 请求前,需要先使用公钥和私钥生成签名。为保障账户安全,请妥善保管私钥,避免外传。尽量不直接使用 API 密钥访问 US3,泄露后风险较大,推荐使用 Token 密钥。

地域 (Region)

地域表示 US3 的数据中心所在物理位置。您可以根据费用、请求来源等综合选择数据存储的地域。详情请参见 US3 已开通的 Region。

单地域空间管理

支持创建、查询、删除存储bucket,并支持各种bucket属性设置。

多地域跨区复制

用户可以针对指定的 2 个或 2 个以上的 bucket,设置跨区复制,实现多地域的数据上传同步功能,实现数据多地备份容灾;通过配合自定义域名和 cname 到多个 bucket 的循环跨区复制,还能实现就近上传的功能;

存储类型

US3 提供标准、低频、归档三种存储类型,全面覆盖从热到冷的各种数据存储场景。其中标准存储类型提供高可靠、高可用、高性能的对象存储服务,能够支持频繁的数据访问;低频存储类型适合长期保存不经常访问的数据,存储单价低于标准类型;归档存储类型适合需要长期保存(建议半年以上)的归档数据,在三种存储类型中单价最低。详情请参见存储类型介绍。

生命周期

用户可以配置生命周期删除,可以对指定前缀的文件进行定期删除,节省用户存储空间;另外生命周期的自动降冷功能,可以对指定前缀的文件进行降冷,自动转换为低频存储、或者转换成归档存储,节省用户的存储成本。

相关服务

您把数据存储到 US3 以后,就可以使用 UCloud 提供的其他产品和服务对其进行相关操作。以下是您会经常使用到的 UCloud 产品和服务:

产品	服务
云主机(UHost)	提供简单高效、处理能力可弹性伸缩的云端计算服务。请参见UHost产品详情页面。
内容分发网络(UCDN)	将源站资源缓存到各区域的边缘节点,供您就近快速获取内容。请参见UCDN产品详情页面。

使用 US3

US3 提供了 Web 服务页面,方便您管理 US3。您可以登录 US3 管理控制台,操作存储空间和对象。关于管理控制台的操作,请参见控制台用户指南。

US3 也提供了丰富的 API 接口和各种语言的 SDK 包,方便您灵活地管理 US3。请参见 API列表和SDK列表。

US3 定价

传统的存储服务供应商会要求您购买预定量的存储和网络传输容量,如果超出此容量,就会关闭对应的服务或者收取高昂的超容量费用;如果没有超过此容量,又需要您按照全部容量支付费用。US3 仅按照您的实际使用容量收费,您无需预先购买存储和流量容量,随着您业务的发展,您将享受到更多的基础设施成本优势。

关于 US3 的价格,请参见 计量计费。

功能概览

存储功能

US3 产品提供了以下对象存储的常用上传、下载功能支持。

普通上传 (Put File)

普通上传是指用户使用对象存储 API 中的 Put File 方法上传单个 File,可以适用在任何一次 HTTP 请求交互即可完成上传的场景,比如小文件的上传。

表单上传 (Post File)

表单上传是一种适用于小文件上传的上传方式,是指用户使用对象存储 API 中的 Post File 请求来完成 File 的上传。

分片上传 (Multipart Upload)

分片上传是指将待上传文件分成多个数据块来分别上传,上传完成之后再调用对象存储 API 将这些 Part 组合成一个 File。

普通下载 (Get File)

普通下载是指用户下载已经上传的文件,文件下载是使用 HTTP 的 GET 请求来完成的。

分片下载 (Range Get)

分片下载是指从 File 指定的位置开始下载的功能,对于大文件可以分为多次下载。如果中途下载中断,重启的时候也可以从上次最后完成的地点开始继续下载。

管理功能

US3 产品除了支持标准的对象存储上传、下载功能外,提供了丰富的管理功能。

域名管理

对象存储空间支持提供测试域名进行访问,您同时可以选择绑定自定义域名创建 CDN 加速。

静态网站托管

可以通过控制台对已经绑定的自定义域名的US3存储桶配置静态网站托管策略。

生命周期

开通存储空间生命周期功能,您可以实现存储空间内所有文件或特定前缀文件的生命周期管理,设置生命周期规则进行归档存储或删除文件操作。

镜像回源

当您向 US3 存储空间请求的数据不存在时,您可以通过镜像回源规则设置,对于获取数据的请求以多种方式进行回源读取,满足您对于数据热迁移、特定请求重定向的需求。

跨域访问

跨域访问(CORS)是指当用户从一个域名的网页去请求另一个域名的资源的操作。您可以通过跨域访问策略管理功能,对跨域访问的权限进行限制。

跨区域复制

您可以选择两个不同地域的存储空间进行绑定,进行文件的同步。提供存储空间跨地域容灾能力的同时能够满足您数据复制的需求。

日志管理

在设置日志管理存储空间后,所有用户对于存储空间的访问日志,会以小时为单位,按照固定的命名,存储到您指定的存储空间中。

令牌管理

令牌功能可以根据用户的需求灵活的开放存储空间和文件管理权限。

增值功能

图片处理

支持对上传保存在对象存储空间中的图片进行一系列图片处理操作。

解压缩服务

解压缩服务是UCloud对外提供的低成本高可靠的解压服务,可设置规则对上传的压缩包自动解压。

文档预览

文档预览功能支持存储在US3的文字文件、演示文件、表格文件、以及pdf文件的在线预览。

产品优势

稳定可靠的存储能力

基于高可用架构设计,将用户所存文件的多个副本分布于不同存储集群,即使出现集群规模的故障也不会影响存储文件的可用性,提供 11个9 的数据持久性。存储服务支持高并发访问,突破传统磁盘I/O限制,满足用户高访问量和高速下载业务的在线存储需求。

低成本的弹性存储空间

云存储空间无上限,无需考虑存储空间扩容问题,支持小于等于 5TB 的单文件。适用于音视频、图片分享等 UGC 类应用海量文件存储。按实际使用量计费,不设最低费用,无存储和带宽资源闲置浪费。

灵活便捷的业务接入

提供 API/SDK、命令行工具和控制台等多种访问方式,同时兼容 AWS S3 API,适合多种语言。帮助用户无缝接入原有业务,能够极大缩短开发周期,帮助业务快速上线。与 UCloud 各类产品紧密协同,UHost 用户可以通过云主机进行内网访问,提供稳定高速的内网上传下载速度。

简单易用的管理工具

提供命令行US3CLI等多种管理工具对存储空间和文件进行管理,操作简单易上手。提供US3FS工具能把存储空间挂载到本地,可以像使用本地文件系统一样操作对象存储。提供大数据适配工具,解决Hadoop访问对象存储的问题,支持Hive、Spark、Flink等大数据计算框架可以像访问HDFS文件系统一样,来读写对象存储上的数据。提供迁移工具US3SYNC,可以便捷地从本地或者其他云环境将数据迁移至US3存储空间中。

安全可靠的分布式系统

提供防盗链、客户端加密、服务端加密、token白名单等功能;支持SSL加密传输;支持以UCloud攻击防御系统来防御DDos攻击。

多样化的云应用场景

适配各类 UCloud 公有云服务,为各类服务提供稳定可靠的后端存储支持。结合 CDN 解决海量数据的存储和分发问题,有效降低访问延迟,提升全网终端用户的访问体验。支持对接图片处理、视频流处理服务,为用户提供各类多媒体在线处理的能力。还可作为数据源,为大数据分析、AI 训练推断等场景提供海量的数据集支持。

使用限制

支持地域

地域	单地域空间管理	跨区域复制
华北一	支持	支持
华北二	支持	支持
上海二	支持	支持
广州	支持	支持
香港	支持	不支持
洛杉矶	支持	不支持
新加坡	支持	不支持
雅加达	支持	不支持
台北	支持	不支持
拉各斯	支持	不支持
圣保罗	支持	不支持
迪拜	支持	不支持

法兰克福	支持	不支持
胡志明市	支持	不支持
华盛顿	支持	不支持
孟买	支持	不支持
首尔	支持	不支持
东京	支持	不支持
曼谷	支持	不支持

HTTPS 访问限制

US3内网域名不支持https访问

域名访问限制

1. US3外网域名不支持任意类型文件的预览,如需外网预览文件,请使用自定义域名
2. US3外网域名不支持apk及ipa类型文件下载,如需外网下载apk及ipa类型文件,请使用自定义域名

上传文件大小限制

通过控制台上传、上传文件 PutFile、表单上传 PostFile的文件大小不能超过 512MB,要上传大小超过 512MB 的文件必须使用分片上传方式。

图片处理限制

地域	大小限制	并发限制
北京	20M	50
上海二	20M	10
广州	20M	10
香港	20M	10
台北	20M	10
其他地域	20M	5

备注:如有更大需求,请联系客户经理或者技术支持。

其他限制项

限制项	说明
归档存储类型	访问前需要解冻,已经存储的数据从冷冻状态恢复到可读取状态根据文件大小以及解冻类型需要不同的等待时间,请参见产品价格补充说明; 目前US3归档存储类型的香港、台北及海外可用区包括:香港、洛杉矶、新加坡、台北、胡志明市、华盛顿、东京、首尔。
存储空间(Bucket)	同一账号在同一地域内创建的存储空间总数不能超过 20 个。
...	存储空间一旦创建成功,其名称、所处地域、存储类型不能修改。

∴	单个存储空间的容量不限制。
上传/下载文件	同一账号在同一地域内的上传或下载的带宽缺省阈值为： 内网上传下载速度：中国大陆地域 10Gbit/s、其他地域 1Gbit/s； 外网上传下载速度：1Gbit/s。 US3默认外网域名部分地域可能存在单个连接下载速度限制，建议使用自定义域名下载 如您的业务（如大数据离线处理等）有更大的带宽需求，请联系技术支持。
QPS(写)	普通用户每秒 1000 个请求（非上限），海外 100 个，如您的业务有更大的需求，请联系技术支持。
QPS(读)	普通用户每秒 2000 个请求（非上限），海外 100 个，如您的业务有更大的需求，请联系技术支持。
删除文件	文件删除后无法恢复。控制台批量删除文件的上限为 1000 个，海外 100 个，如您有更大批量的删除需求，请联系技术支持。
域名绑定	中国大陆地域绑定的域名必须在工信部备案，其他地域的域名绑定不需要在工信部备案。每个存储空间最多可以绑定 100 个域名。

存储类型

US3 提供标准、低频、归档三种存储类型,分别用于频繁访问的热点数据、低频访问的备份数据和适用于长期保存的归档数据,全面覆盖从热到冷的各种数据存储场景。

该内容为动态文件,无法展示

标准存储

标准存储提供高可靠、高可用、高性能的对象存储服务,具有高吞吐和低延时的服务响应能,能够支持频繁的热点数据访问。

关键特性:

数据持久性:99.999999999%

可用性:99.95%

访问:实时访问

适用场景:各种社交、分享类的图片、音视频应用、大型网站、大数据分析、移动应用、游戏程序

低频存储

低频存储类型提供高可靠性,较低存储成本和较低访问时延的对象存储服务,适合长期保存不经常访问的数据。存储单价低于标准类型,低频存储有最短存储时间和最小对象大小,存储时间

短于 30 天提前被删除会产生一定费用。单个文件大小低于 64KB, 会按照 64KB 计算存储空间, 数据获取会产生费用。

关键特性:

数据持久性:99.999999999%

可用性:99.9%

访问:实时访问

适用场景:各类移动应用、智能设备、政企业务数据、企业数据的长期备份, 支持实时数据访问

归档存储

归档存储类型提供了高可靠性, 极低存储成本和长期保存的离线冷数据存储, 适合需要长期保存 (建议半年以上) 的归档数据, 在存储周期内极少被访问。在三种存储类型中单价最低, 归档存储有最短存储时间和最小对象大小, 存储时间短于 60 天的文件提前删除会产生一定费用。文件大小低于 64KB, 会按照 64KB 计算存储空间, 数据获取会产生费用。当前归档类型可用区仅限中国大陆 (华北一、华北二、上海、广州)。

关键特性:

数据持久性:99.999999999%

可用性:99.9%

访问:访问前需要解冻, 从冷冻状态恢复到可读取状态根据文件大小以及解冻类型需要不同的等待时间, 请参见产品价格补充说明

适用场景:长期保存的档案数据等合规性文件归档、医疗影像、科学资料、影视素材

存储类型转换

支持以下存储类型的自动转换:

1. 标准存储转换为低频存储
2. 标准存储转换为归档存储
3. 低频存储转换为归档存储

备注:

1. 对于需要将归档类型的文件重新转换为标准类型或低频类型的文件,低频类型文件转换为标准类型的文件,可以通过重新上传文件,指定对应存储类型来实现存储类型转换。
例如,用户需要将已经存储成低频类型的文件重新转换成标准类型,可以重新上传该文件,上传的时候指定标准存储类型,新写入的文件是标准存储类型。
2. 对于已经转储成归档类型的文件,需要先执行 RESTORE 操作,解冻成可读取状态后,才能被读取。

存储类型支持的 API

API	标准存储类型	低频访问存储类型	归档存储类型
空间管理
CreateBucket	支持	支持	支持
DescribeBucket	支持	支持	支持
UpdateBucket	支持	支持	支持
DeleteBucket	支持	支持	支持
文件管理
PutFile	支持	支持	支持
PostFile	支持	支持	支持

UploadHit	支持	支持	支持
GetFile	支持	支持	支持,需要先解冻
HEADFile	支持	支持	支持,需要先解冻
DeleteFile	支持	支持	支持
Multipart操作
InitiateMultipartUpload	支持	支持	支持
UploadPart	支持	支持	支持
FinishMultipartUpload	支持	支持	支持
AbortMultipartUpload	支持	支持	支持
PrefixFileList	支持	支持	支持
RestoreObject	不支持	不支持	支持
其他功能的API
日志管理	支持	支持	支持
图片处理	支持	支持	支持

地域和域名

US3 对象存储提供外网、内网域名两种访问方式：外网域名可响应所有来自于互联网的访问请求，内网域名可响应来自于同一地域下的 UCloud 公有云服务的访问请求（如：UHost 云主机、UK8S 容器云等）。

若使用外网域名对数据进行访问，会产生外网流出流量费用。

在您创建存储空间后，您存储空间的访问域名地址格式为：<存储空间名称>.<域名地址>。

公有云下**US3**各地域访问域名（Endpoint）如下：

地域	外网域名	内网域名
华北一	cn-bj.ufileos.com	ufile.cn-north-02.ucloud.cn
华北二	cn-wlcb.ufileos.com	internal-cn-wlcb.ufileos.com
上海二	cn-sh2.ufileos.com	internal-cn-sh2-01.ufileos.com
广州	cn-gd.ufileos.com	internal-cn-gd-02.ufileos.com
香港	hk.ufileos.com	internal-hk-01.ufileos.com
洛杉矶	us-ca.ufileos.com	internal-us-ca-01.ufileos.com
新加坡	sg.ufileos.com	internal-sg-01.ufileos.com
雅加达	idn-jakarta.ufileos.com	internal-idn-jakarta-01.ufileos.com
台北	tw-tp.ufileos.com	internal-tw-tp.ufileos.com

拉各斯	afr-nigeria.ufileos.com	internal-afr-nigeria.ufileos.com
圣保罗	bra-saopaulo.ufileos.com	internal-bra-saopaulo.ufileos.com
迪拜	uae-dubai.ufileos.com	internal-uae-dubai.ufileos.com
法兰克福	ge-fra.ufileos.com	internal-ge-fra.ufileos.com
胡志明市	vn-sng.ufileos.com	internal-vn-sng.ufileos.com
华盛顿	us-ws.ufileos.com	internal-us-ws.ufileos.com
孟买	ind-mumbai.ufileos.com	internal-ind-mumbai.ufileos.com
首尔	kr-seoul.ufileos.com	internal-kr-seoul.ufileos.com
东京	jpn-tyo.ufileos.com	internal-jpn-tyo.ufileos.com
曼谷	th-bkk.ufileos.com	internal-th-bkk.ufileos.com
英国	uk-london.ufileos.com	internal-uk-london.ufileos.com
莫斯科	rus-mosc.ufileos.com	internal-rus-mosc.ufileos.com

注意：

1. 由于存储集群所在地域不同，跨国访问可能会存在响应延迟或失败的情况，海外加速服务功能处于内测阶段，如需使用请联系技术支持。
2. 兼容支持AWS S3协议的访问域名请参见AWS S3协议支持说明。

计量计费

计量计费章节介绍了 US3 对象存储服务的费用组成及计费方式,您可前往 [费用中心](#) 查看每日的账单明细。

对象存储US3支持以下计费方式:

- **后付费:**所有计费项默认采用后付费模式,US3 每小时都会统计用户使用量,按天出日账单,账单粒度为具体计费项。先使用,后付费,适用于业务用量经常有变化的场景。
- **资源包:**针对部分常用计费项支持预付费模式的优惠抵扣包。在费用结算时,优先从资源包抵扣用量,先购买,后抵扣。资源包到期/用尽后,会自动转为按使用后付费。适用于业务用量相对稳定的场景。

US3收费模式

该内容为动态文件,无法展示

如您对您在使用 US3 过程中产生的费用明细存在疑问,请参考以下文档:

1. US3 的计费项及价格介绍,请参考 [产品价格](#)。
2. US3 的计费规则详情,请参考 [计费规则](#)。
3. 关于 US3 的欠费处理策略:数据的保留和销毁时间、以及相关计费说明,请参考 [欠费处理](#)。
4. 您如果对 US3 计费存在疑问,建议您参考 [计费案例](#),了解实际使用场景中 US3 对象存储服务的计费方式。

5. US3 的预付费模式的详情, 请参考资源包。

产品价格

本文介绍 US3 对象存储服务后付费的计费项和定价标准,US3 定价会因存储类型 和 地域 有所不同,当前地域定价分为“中国大陆”及“香港、台北及海外”两部分。

现US3新推出预付费的资源包,针对部分计费项目,比后付费更优惠,可以根据实际需求购买。详情请参见资源包。

中国大陆

项目	计费项	标准类型单价	低频类型单价	归档类型单价
存储费用	数据存储	0.004 元/GB/天	0.002 元/GB/天	0.0008 元/GB/天
流量费用	内/外网流入流量	免费	免费	免费
:::	内网流出流量	免费	免费	免费
:::	外网流出流量	00:00-08:00 (闲时):0.25 元/GB	00:00-08:00 (闲时):0.25 元/GB	00:00-08:00 (闲时):0.25 元/GB
:::	:::	08:00-24:00 (忙时):0.45 元/GB	08:00-24:00 (忙时):0.45 元/GB	08:00-24:00 (忙时):0.45 元/GB
:::	CDN 回源流出流量	0.15 元/GB	0.15 元/GB	0.15 元/GB
请求费用	PUT类型请求次数	0.01 元/万次	0.1元/万次	0.1元/万次

:::	GET类型请求次数	0.01 元/万次	0.1元/万次	0.1元/万次
数据取回	数据取回	/	0.03 元/GB	文件处于已解冻状态时免费
:::	数据解冻	无	无	* 高优先级:0.2元/GB * 标准:0.06 元/GB
图片处理	基础图片处理	0.025元/GB	0.025元/GB	0.025元/GB
:::	图片高级压缩	0.1元/千次	0.1元/千次	0.1元/千次
最短存储期限	最短存储期限	无	30 天	60 天
最小文件大小	最小文件大小	无	64KB	64KB

目前 **US3** 中国大陆的可用区包括：华北一、华北二、上海二、广州。

注意：**【图片处理】**从**2022年7月1号**开始计费。

香港、台北及海外

项目	计费项	标准类型单价	低频类型单价	归档类型单价
存储费用	数据存储	0.004 元/GB/天	0.002 元/GB/天	0.0008 元/GB/天
流量费用	内/外网流入流量	免费	免费	免费
:::	内网流出流量	免费	免费	免费
:::	外网流出流量	0.4 元/GB	0.4 元/GB	0.4 元/GB

:::	CDN 回源流出流量	0.4 元/GB	0.4 元/GB	0.4 元/GB
请求费用	PUT类型请求次数	0.01 元/万次	0.1元/万次	0.1元/万次
:::	GET类型请求次数	0.01 元/万次	0.1元/万次	0.1元/万次
数据取回	数据取回	免费	0.03 元/GB	文件处于已解冻状态时免费
:::	数据解冻	无	无	0.06 元/GB
图片处理	基础图片处理	0.025元/GB	0.025元/GB	0.025元/GB
:::	图片高级压缩	0.1元/千次	0.1元/千次	0.1元/千次
最短存储期限	最短存储期限	无	30 天	60 天
最小文件大小	最小文件大小	无	64KB	64KB
对象标签费用	对象标签	0.00189 元/每万个标签	0.00189 元/每万个标签	0.00189 元/每万个标签

目前 **US3** 香港、台北及海外的可用区包括：香港、洛杉矶、新加坡、雅加达、台北、拉各斯、圣保罗、迪拜、法兰克福、胡志明市、华盛顿、孟买、东京、首尔、曼谷、英国、莫斯科

注意：

1. 【图片处理】从2022年7月1号开始计费；
2. 当前归档类型可用区仅限中国大陆(华北一、华北二、上海、广州)；
3. 目前对象标签支持越南、巴西,如其他地域有需要,请联系技术支持。

补充说明

1. **最短存储期限**:低频存储和归档存储类型都有最短存储期限,早于最短存储期限删除、修改、覆盖文件,需要补足未满最短存储期限的剩余天数的存储费用。
2. **最小文件大小**:低频存储和归档存储类型,单文件小于 64KB 文件按照 64KB 文件大小计算存储空间,超过 64KB 的文件按照实际大小计算存储空间。
3. **解冻归档存储类型文件**:
 - 使用 RESTORE 进行解冻操作,需要收取数据解冻费用。解冻完成后,数据取回不再收取额外费用,仅收取正常流出流量费用。
 - 解冻操作完成后数据保留 3 天,已解冻的数据取回不收取取回费用,超过 3 天则无法下载,需要重新进行解冻操作。
 - 解冻类型:访问归档文件请提前解冻,不同解冻模式的首字节取回时间如下:
 - 高优先级 (Expedited):表示15分钟内完成;
 - 标准 (Standard):表示1~5小时内完成。如果不传入解冻类型,则默认为Standard。
4. **跨区域复制**:当使用跨区域复制功能时,会产生外网流出流量费用,同时各个区域的存储空间按各自的存储容量和计费项价格进行收费。全球化存储空间功能已下线(建议使用跨区域复制功能),原全球化存储空间计费项目和计费规则暂时不变,保持旧版价格和旧计费方式。
5. **数据取回费用的说明**:对于存储类型为低频类型的数据,取回费用 = 下载文件大小 * 取回费用单价 + 流出流量费用,不论内网还是外网下载都会收取取回费用;
对于存储类型为归档类型的数据,解冻费用 = 解冻文件大小 * 解冻费用单价,下载已解冻的数据时,收取流出流量费用。数据取回费用独立于流量费用,如数据取回过程产生外网流量费用则正常收取,内网下载不收流量费用;在解冻期间,不再收取额外解冻费用。
6. **请求费用说明**:

Put类请求

Put类请求包括通过API、SDK或者US3管理控制台等方式进行数据上传、拷贝、删除等操作。

API请求	对应操作
-------	------

PutFile	上传文件
PostFile	表单上传
RestoreFile	解冻文件
ClassSwitch	文件存储类型转换
Copy	拷贝文件
UploadHit	秒传文件
OpMeta	操作文件的Meta信息
InitiateMultipartUpload	初始化分片
UploadPart	上传分片
FinishMultipartUpload	完成分片
AbortMultipartUpload	放弃分片
ListObject	获取目录文件列表
PrefixFileList	前缀列表查询
DeleteFile	删除文件

Get类请求

Get类请求包括通过API、SDK或者OSS管理控制台等方式执行除ListObject以外的所有对于文件以及其他信息的读取操作。

API请求	对应操作
HeadFile	查询文件基本信息

GetFile	下载文件
GetMultiUploadId	获取正在执行的分片上传id

7. 计费项Code说明：

us3订单详情中,资源ID的形式:"计费项Code-项目id",其中计费项以及Code如下

项目	计费项	计费项Code
存储费用	标准存储费用	ufile-storage
:::	低频存储费用	ufile-iastorage
:::	归档存储费用	ufile-acstorage
:::	低频最短存储期限费用	ufile-iasstorage
:::	归档最短存储期限费用	ufile-acstorage
流量费用	忙时下载流量	ufile-busyflow
:::	闲时下载流量	ufile-idleflow
:::	(海外)下载流量	ufile-flow
:::	CDN 回源流出流量	ufile-cdnflow
请求费用	PUT类型请求次数(标准类型)	ufile-standputcount
:::	PUT类型请求次数(低频类型)	ufile-iaputcount
:::	PUT类型请求次数(归档类型)	ufile-acspcount
:::	GET类型请求次数(标准类型)	ufile-standgetcount

:::	GET类型请求次数(低频类型)	ufile-iagetcount
:::	GET类型请求次数(归档类型)	ufile-acsggetcount
数据取回	归档文件高优先级解冻	ufile-archiveretrieval
:::	低频文件取回费用	ufile-iagetsize
:::	归档文件标准解冻	ufile-acrestore
图片处理	基础图片处理	ufile-imageHandle
:::	图片高级压缩	ufile-imagecompresscount

计费规则

计费项目

US3 产品目前的计费项目包括数据存储、外网流出流量、CDN 回源流量、请求费用、数据取回。具体计费项目的价格请参考 产品价格。

计费方式

US3 对象存储服务按使用量后付费,每小时统计用户使用量,以日结形式产生费用账单,您可前往 [费用中心](#) 查看每日的账单明细,账单粒度为项目。各个计费项目的计费规则详情如下。

现US3新推出预付费的资源包,针对部分计费项目,比后付费更优惠,可以根据实际需求购买。详情请参见[资源包](#)

存储容量

每天存储量费用 = 每天平均存储量 * 日存储量单价

每天平均存储量 = 当天每小时平均存储量之和 ÷ 24 小时

外网流出流量

每天闲时流量费用 = 每天闲时下载流量 * 闲时流量单价

每天忙时流量费用 = 每天忙时下载流量 * 忙时流量单价

CDN 回源流量

每天 CDN 回源流量费用 = 每天 CDN 回源流量 * CDN 回源流量单价

请求费用

请求费用 = 每天请求次数 × 每万次请求的单价 ÷ 10000

补充说明

1. 欠费 30 天内补足欠费, 服务会自动开启。超过 30 天后, 将视您主动放弃 US3 存储服务, 空间和数据删除都无法恢复。详情请参照 [欠费处理](#)。
2. US3 已取消预付费-购买配额包方式, 不影响原购买配额包未使用完的用户, 可以继续使用。
3. 调用 US3 API 的 Put/Get 类请求产生的请求次数费用, 请求无论成功还是失败, 都会统计请求次数。

欠费处理

当您的 UCloud 处于账户欠费状态时,在控制台进入 US3 产品页面时可看到以下提示:

US3服务受限

您的US3因欠费使用受限,可用性会逐步降低

欠费30天内完成支付,服务会自动恢复

超过30天,将视为您主动放弃US3存储服务,存储空间将被删除。空间内的数据会被清理且不可恢复。



关闭

欠费对 US3 服务造成的影响

1. 欠费 3 天后,服务(上传、下载、删除等)会被限制,无法创建US3存储空间,部分请求会失败,直至所有服务都不可用。限制服务期间存储数据仍被保留。

计费案例

案例一

用户A 使用国内 US3,同时开启了 UCDN 加速服务。在 18年12月1号 的日消费量:文件存储(标准类型)容量为 2.5TB,产生 CDN 回源流量200GB。该用户的加速场景产生了 CDN 下载流量 900GB。

当天费用计算:

US3 产品侧总费用如下:

存储空间费用: $2.5 * 1024GB * 0.004 \text{ 元/GB/天} = 10.24 \text{ 元}$

CDN 回源流量费用: $200GB * 0.15 \text{ 元/GB} = 30 \text{ 元}$

总费用: $10.24 + 30 = 40.24 \text{ 元}$

UCDN 产品侧总费用如下:

CDN 下载流量费用= $900GB * 0.34 \text{ 元/GB} = 306 \text{ 元}$

US3+UCDN 总费用: $40.24 + 306 = 346.24 \text{ 元}$

备注:用户因为开启了 UCDN 加速服务,还会生成一项 CDN 下载流量的费用,在 UCDN 产品侧统计和计费。计费价格详见:UCDN 产品价格。

案例二

用户B 使用 US3,没有开启 UCDN 加速服务。在 18年12月1号 的日消费量:文件存储(标准类型)容量为 2.5TB,在闲时段产生外网流出流量 150GB,在忙时段产生外网流出流量 800GB。

当天费用计算:

存储空间费用: $2.5 * 1024 \text{ GB} * 0.004 \text{ 元/GB/天} = 10.24 \text{ 元}$

外网流出流量费用: $150\text{GB} * 0.25 \text{ 元/GB} + 800\text{GB} * 0.45 \text{ 元/GB} = 397.5 \text{ 元}$

总费用: $10.24 + 397.5 = 407.74 \text{ 元}$

案例三

用户C 使用 US3, 没有开启 UCDN 加速服务。在 18年12月1号 的日消费量: 文件存储(标准类型)容量为 1TB, 文件存储(低频类型)容量为 1TB, 文件存储(归档类型)容量为 0.5TB。在忙时段产生标准类型外网流出流量 500GB, 低频类型外网流出流量 10GB, 归档类型解冻数据量 15GB, 归档类型外网流出流量 10GB。

当天费用计算:

存储空间费用: $1 * 1024\text{GB} * 0.004 \text{ 元/GB/天} + 1 * 1024\text{GB} * 0.002 \text{ 元/GB/天} + 0.5 * 1024\text{GB} * 0.0008 \text{ 元/GB/天} = 10.24 \text{ 元}$

外网流出流量费用: $(500\text{GB} + 10\text{GB} + 10\text{GB}) * 0.45 \text{ 元/GB} = 234 \text{ 元}$

低频类型取回费用: $10\text{GB} * 0.03 \text{ 元/GB} = 0.3 \text{ 元}$

归档类型标准解冻费用: $15\text{GB} * 0.06 \text{ 元/GB} = 0.9 \text{ 元}$ (注: 如执行解冻操作, 即使数据未被取回也会收取数据解冻费用) 总费用: $10.24 + 234 + 0.3 + 0.9 = 245.44 \text{ 元}$

资源包

US3对象存储服务默认按使用量后付费模式。资源包是US3推出的一种预付费优惠抵扣包,针对部分计费项目,比后付费更优惠,可以根据实际需求购买。当您购买指定资源包后,US3会按您的实际使用情况扣除资源包的额度,资源包到期/用尽后,会自动转为按使用量后付费。目前只有国内地域(华北二、华北一、广州、上海二)支持资源包。

计费说明

针对不同类型的资源包的计费说明及抵扣规则。

资源包类型	说明	抵扣规则
标准存储包	<ul style="list-style-type: none"> - 抵扣标准类型的存储费用。 - 支持升级、续费、退费、查看明细 - 不支持叠加购买,若当前资源包规格不足,可直接操作升级 	<p>每日进行抵扣,抵扣当天的平均存储量,每日会重置抵扣量,剩余抵扣量不可转移到次日。</p> <p>如购买1个月10GB的标准存储包,则每日都能享用10GB标准存储,超出的存储量部分将自动转为后付费。</p>
外网流出流量包: - 闲时 - 忙时	<ul style="list-style-type: none"> - 抵扣闲时/忙时外网流出所产生的流量费用 - 支持续费、退费、查看明细 - 不支持升级操作,若当前资源包规格不足,可叠加购买外网流出流量包 	<p>每日进行抵扣,抵扣当天的外网流出闲时/忙时总流量,每月会重置抵扣量,剩余抵扣量不可转移到下月。</p> <p>如购买1年100GB规格的外网流出闲时流量,则1年内每月都能享用100GB的外网流出闲时流量的抵扣额度,超出部分将自动转为后付费。</p>

1. US3资源包仅支持抵扣当前UCloud账号下产生的费用,不支持跨账号抵扣。
2. US3资源包为US3服务专用资源包,不支持跨产品抵扣。
3. US3资源包仅支持抵扣资源包购买后产生的费用,不支持抵扣资源包购买前产生的费用。
4. 一种类型的资源包只能抵扣该资源包对应计费项产生的费用,不能抵扣其余计费项的费用。
5. US3资源包是地域专属的,不支持共享给其他地域使用,请按需选择。
6. US3资源包每个周期的剩余抵扣量,不支持转移到下个周期。
7. US3资源包到期或用完后,存储空间的数据无需迁移,会自动转为按使用后付费。
8. US3资源包不能与其它折扣及活动叠加使用,购买资源包后,会优先使用资源包。

购买

1. 登陆US3对象存储控制台。
2. 在顶部导航栏选择**资源包管理**,进入资源包管理列表页。
3. 单击**购买资源包**。

对象存储 US3 切换旧版

· 单地域空间管理 · 统计报表 · **资源包管理** · 管理工具及SDK · 令牌管理

当您购买指定资源包后，US3会按您的实际使用情况扣除资源包的额度，资源包到期/用尽后，会自动转为按量后付费。详情请[查看文档](#)

有效资源包 **购买资源包**

已过期资源包

资源包名称	资源ID	资源包类型	生效地域	总量	余量	生效时间	失效时间	操作
标准存储包 (华北二)		标准存储包	华北二	100GB	- (按日重置)	2023-12-19 16:40:10	2024-06-21 23:59:59	<input type="button" value="使用明细"/> <input type="button" value="升级"/> <input type="button" value="更多"/>

< 1 > 10条/页 /1

4. 根据您的实际业务选择资源包类型、所属地域、资源包规格、资源包时长等参数，可以自定义资源包名称后，单击立即购买。

[资源包管理 / 购买资源包](#)

资源包配置

资源包类型

标准存储包 闲时-外网流出流量包 忙时-外网流出流量包

【抵扣说明】若您购买100GB规格的外网流出流量包一年，则您从购买之日起一年内，每个月都可以抵扣100GB的外网流量流量，超出部分自动按量计费。

地域

地域专属资源包，不支持共享给其他地域使用，请根据您的资源所在地谨慎选择。

华北二

规格

100GB 300GB 500GB 1TB 2TB 5TB 10TB 30TB 50TB 100TB 300TB 500TB

资源包配置

资源包名称 *

闲时-外网流出流量包 (

付费信息

按量 元

1个月 折合: 元/月

合计费用 **9.5折** 元

原价 ;

立即购买

我已阅读并同意《对象存储US3计费说明》和《退款规则》

5. 按照页面指引,完成购买流程后,资源包管理页会展示对应资源包信息。

对象存储 US3 切换旧版

· 单地域空间管理 · 统计报表 · **资源包管理** · 管理工具及SDK · 令牌管理

① 当您购买指定资源包后，US3会按您的实际使用情况扣除资源包的额度，资源包到期/用尽后，会自动转为按量后付费。详情请[查看文档](#)

有效资源包 | 已过期资源包

[购买资源包](#)

资源包名称	资源ID	资源包类型	生效地域	总量	余量	生效时间	失效时间	操作
闲时-外网流出流量包		闲时-外网流出流量包		100GB	100GB (按月重置)	2023-11-22 11:12:06	2024-07-26 23:59:59	使用明细 续费 ...
忙时-外网流出流量包		忙时-外网流出流量包		100GB	100GB (按月重置)	2023-11-22 15:14:27	2024-06-25 23:59:59	使用明细 续费 ...
标准存储包		标准存储包		1TB	- (按日重置)	2023-11-22 15:14:47	2025-02-28 23:59:59	使用明细 升级 ...

< 1 > 10条/页 /1

- 购买折扣视当前活动折扣为准。购买时长越久，活动力度越大。

资源包列表部分参数说明：

1. 总量：资源包一个周期的总量，即资源包规格；如购买100GB标准存储包，总量为单日的标准存储包总量。
2. 余量：标准存储包为按日重置抵扣量，不支持查看余量；仅支持查看外网流出流量当前周期内的剩余抵扣量，余量统计数据非实时数据，有一天的延迟。
3. 生效时间：购买资源包成功后立即生效。
4. 失效时间：每月固定31天，根据生效时间和资源包购买时长得出。如1月10日购买1个月标准存储包，则失效时间为2月9日23:59:59到期。

升级

当前仅标准存储包支持对其升级(升配)，但不支持降级(降配)。外网流出流量包可以通过叠加购买的方式达到升级效果。

1. 登陆US3对象存储控制台。
2. 在顶部导航栏选择资源包管理, 进入资源包管理列表页。
3. 单击资源包右侧的升级。

对象存储 US3 切换旧版

· 单地域空间管理 · 统计报表 · 资源包管理 · 管理工具及SDK · 令牌管理

当您购买指定资源包后, US3会按您的实际使用情况扣除资源包的额度, 资源包到期/用尽后, 会自动转为按量后付费。详情请[查看文档](#)

有效资源包 购买资源包

已过期资源包

资源包名称	资源ID	资源包类型	生效地域	总量	余量	生效时间	失效时间	操作
标准存储包 (华北二)		标准存储包	华北二	100GB	- (按日重置)	2023-12-19 16:40:10	2024-06-21 23:59:59	使用明细 升级 ...

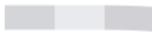
< 1 > 10条/页 /1

4. 在资源包升级页面, 选择要升级的资源包规格, 单击立即购买, 按照页面指引完成支付流程。

资源包升级

① 升级将增加资源包当前有效期内的规格，资源包有效期不会变化，若您的资源包即将到期，请选择续费。

当前配置

资源包名称	标准存储包（华北二）	资源ID		资源包类型	标准存储包
生效地域	华北二	当前规格	100GB	失效时间	2024-06-21 23:59:59

配置调整

升级后规格 *

仅可对规格进行升级操作

500GB	1TB	2TB	5TB	10TB	20TB	30TB	50TB	70TB	100TB	150TB
200TB	300TB	400TB	500TB	800TB	1PB	2PB				

付费信息

付费方式 按量

到期时间 2024-06-21 23:59:59

合计费用  元

差价为购买周期内需退/补的预估金额

更改配置

我已阅读并同意《对象存储US3计费说明》和《退款规则》

- 升级是在该资源包剩余有效期内，补规格配置前后的价差。升级折扣视当前活动折扣为准。

续费

有效资源包在到期前支持续费操作，续费时不支持修改生效地域、规格、类型。当前仅支持在资源包管理页手动续费。

1. 登陆US3对象存储控制台。
2. 在顶部导航栏选择**资源包管理**，进入资源包管理列表页。
3. 单击资源包右侧的**续费**。

续费 ⊗

资源包名称	标准存储包（华北二）	资源包类型	标准存储包
生效地域	华北二	当前规格	100GB

续费时长 *

1个月 9.0折 ▾

我已阅读并同意 [《对象存储US3计费说明》](#) 和 [《退款规则》](#) 取消 确定

4. 在资源包续费页面，选择要续费的资源包规格，单击**立即购买**，按照页面指引完成支付流程。

使用明细

支持查看资源包按日的使用明细图。

1. 登陆US3对象存储控制台。
2. 在顶部导航栏选择**资源包管理**，进入资源包管理列表页。
3. 单击资源包右侧的**使用明细**，即可查看且支持.csv格式下载明细。

使用明细

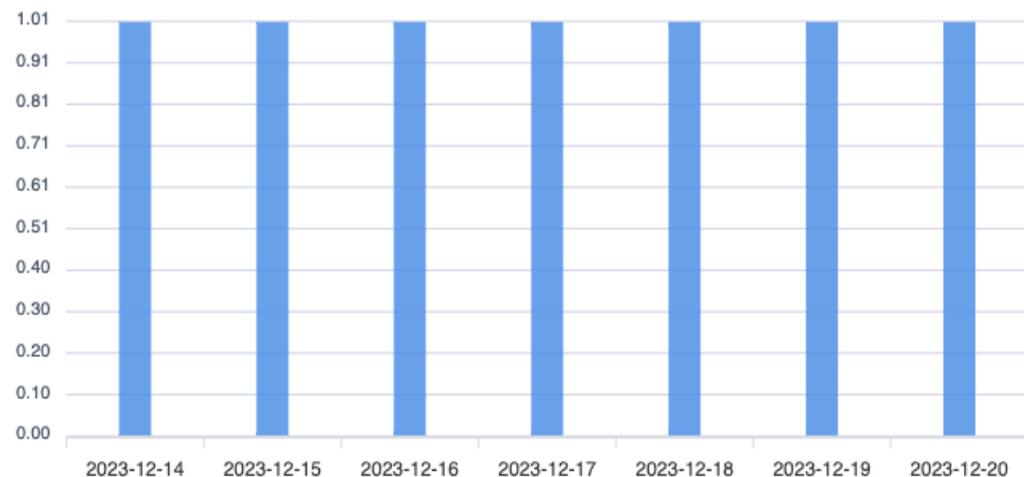


最近7天 ▾

● 2023-12-14 - 2023-12-21



● 资源包使用量 (GB)



关闭

退款规则

资源包退款不会影响存储的数据。资源包退款后,如果当前账户下无抵扣的资源包,则按使用后付费。

退款条件:

1. 用量:资源包没有产生用量、即未使用。
2. 有效期:资源包在有效期内。

开通 US3 服务

当您注册 UCloud 账号并完成实名认证后,即可使用 US3 对象存储服务。

UCloud 对象存储服务 US3 为您提供基于网络的数据存取服务。使用 US3 您可以通过网络随时存储和调用包括文本、图片、音频和视频等在内的各种非结构化数据文件。

初次使用 US3,请您先了解 US3 使用限制。

使用 US3

1. 登录 UCloud 官网。
2. 点击官网右上角 登录控制台 按钮。
3. 在控制台界面左上角点击全部产品按钮,在产品列表中点击 对象存储 US3。

U
全部产品
消息 | 告警 | 帮助与反馈

收藏

- ☰
- ⊕
- ☰
- ⊖
- ⊕

🔍 请输入产品名称

产品动态 NEW

计算

- ☰ 云主机 UHost
- 🏠 私有专区 UDHost
- ☁️ 物理云主机 UPHost
- 📄 混合云 UHybrid
- ⚡ 弹性伸缩 UAS
- 📦 容器服务 CaaS

网络

- 🌐 基础网络 UNet
- 🔗 私有网络 VPC
- ⚖️ 负载均衡 ULB
- 🔍 云解析 UDNS
- 🌍 全球动态加速 PathX

私有网络

- 🔒 VPN网关 IPsecVPN
- 🔗 NAT网关 NAT
- 📄 路由表 RouteTable
- 🔒 网络ACL NetworkACL
- 💻 虚拟网卡 UNI
- 📍 内网VIP VIP

存储

- 📁 文件存储 UFS
- 📄 对象存储 US3
- 🚢 数据方舟 UDataArk
- 📁 归档存储 UArchive

云分发

- 🌐 云分发 UCDN
- 📡 开放式分发节点 UODN

数据库

- ☰ 云数据库 UDB
- 📄 云内存存储 UMem
- 👥 分布式数据库 UDDB
- 📄 分布式NewSQL数据库 TIDB
- 📄 时序数据库 UTSDB

数据分析

- 👥 托管Hadoop集群 UHadoop
- 📄 云数据仓库 UDW
- 🔍 数据湖分析 USQL
- 📄 ES服务 ElasticSearch

中间件

- 📄 工作流服务 StepFlow
- 📄 Kafka消息队列 UKafka
- 🔗 API网关 UAPIGateway
- 📄 消息队列 URocketMQ
- 📄 分布式应用服务 UEDAS
- 🌐 SSH加速通道 GlobalSSH

容器服务

- 📄 容器云 UK8S
- 📄 通用计算 UGC
- 👥 容器镜像库 UHub
- 📄 容器集群 UDocke
- 📄 容器实例 Cube

物联网

- 📄 物联网通信云平台 UIoT Core
- 📄 物联网边缘网关 UIoT Edge
- 📄 智能接入盒子 AccessBox

边缘计算

- 📄 边缘计算节点 UEDN
- 🎮 云游戏 UCGS

人工智能

- 📄 AI在线服务 UAI Inference
- 📄 AI训练服务 UAI Train
- 📄 AI内容审核 UAI Censor
- 📄 AI算法平台 UAI Algorithm

视频服务

- 📄 云直播 ULive
- 📄 媒体工厂 UMedia
- 📄 视频工具 UVideoSDK
- 📄 实时音视频 URTC

账号与安全访问

- 📄 操作日志 ULog
- 📄 密钥管理服务 UKMS

安全防护

- 📄 WEB应用防火墙 UWAF
- 📄 DDoS攻击防护 UDDoS
- 📄 主机入侵检测 UHIDS
- 📄 WEB漏洞扫描 UWS

产品动态

- 即将上线**

NVMe版MySQL支持数据库归档

提供更完备的数据库归档能力，支持归档到原实例，且支持对具体的库/表进行数据库归档。发布范围：全量
- 即将上线**

UHadoop特性更新

1、UHadoop2.8.5版本支...

[查看全部预告](#)
- 9月30

USMC支持专线迁移

[新特性](#) [详情](#)
- 9月18

USMC支持Ubuntu 14 和 CentOS 8 系统迁移

[新特性](#) [详情](#)
- 9月14

UIoT Core支持物模型命令、事件调用参数子集

[新特性](#) [详情](#)
- 9月14

UIoT Core规则引擎支持设备变量

[新特性](#) [详情](#)

[查看更多动态](#)

开始使用 US3

US3 对象存储服务将数据文件以对象 (object) 的形式上传到存储空间 (bucket) 中。您可以进行以下操作：

创建一个或者多个存储空间, 向每个存储空间中添加一个或多个文件。

通过获取已上传文件的地址进行文件的分享和下载。

通过设置和修改 Token 令牌的读写权限 (ACL) 设置来控制用户的对存储空间或文件的访问权限。

通过 UCloud 管理控制台、各种便捷工具、以及丰富的 SDK 包执行基本和高级US3 操作。

使用 UCloud 管理控制台

使用 UCloud 管理控制台来完成 US3 基本操作的流程如下：

1. 开通 US3 服务
2. 创建存储空间
3. 上传文件
4. 下载文件
5. 删除文件
6. 删除存储空间

使用命令行管理工具 **US3 Tools**

US3 Tools 是以命令行方式管理 US3 数据的工具,支持 Windows、Linux、Mac 平台。US3 Tools 提供方便、简洁、丰富的 Bucket 和 Object 管理命令,操作简单易用。支持文件和文件夹(目录)上传下载、分片断点续传等。详情请参见US3CLI 命令行工具。

使用 **API/SDK**

US3 提供多种语言的 API/SDK 包,方便您快速进行二次开发。各 SDK 详细信息请参见US3 SDK 文档。各接口的详细信息请参见US3 API 文档。

存储空间

US3 对象存储服务将数据文件以对象 (object) 的形式上传到存储空间 (bucket) 中, 如需使用 US3 服务存储文件, 您需要先创建存储空间。

创建存储空间

在空间管理页点击左上角操作栏的“创建存储空间”。

对象存储 US3

切换旧版

单地域空间管理

全球化空间管理

统计报表

管理工具及SDK

令牌管理

创建存储空间

更改业务组

删除存储空间

<input type="checkbox"/>	存储空间域名①	地域▼	业务组	空间类型▼	数据分析②	创建时间	操作
<input type="checkbox"/>	ucos-ucos-gd.ufileos.com	广州	HadoopTest	私有空间		2019-06-05	详情 文件管理 域名管理 ...
<input type="checkbox"/>	ucos-ucos-sh2.ufileos.com	上海二	未分组	公开空间		2019-06-27	详情 文件管理 域名管理 ...
<input type="checkbox"/>	ucos-ucos-bj.ufileos.com	北京	未分组	公开空间		2019-07-05	详情 文件管理 域名管理 ...
<input type="checkbox"/>	ucos-ucos-bj.ufileos.com	北京	未分组	公开空间		2019-08-08	详情 文件管理 域名管理 ...
<input type="checkbox"/>	ucos-ucos-bj.ufileos.com	北京	未分组	私有空间		2019-09-02	详情 文件管理 域名管理 ...
<input type="checkbox"/>	ucos-ucos-tp.ufileos.com	台北	未分组	公开空间		2019-09-11	详情 文件管理 域名管理 ...
<input type="checkbox"/>	ucos-ucos-bj.ufileos.com	北京	S3FS	私有空间		2019-09-25	详情 文件管理 域名管理 ...
<input type="checkbox"/>	ucos-ucos-bj.ufileos.com	北京	未分组	私有空间		2019-09-25	详情 文件管理 域名管理 ...
<input type="checkbox"/>	ucos-ucos-sh2.ufileos.com	上海二	未分组	私有空间		2019-09-30	详情 文件管理 域名管理 ...
<input type="checkbox"/>	ucos-ucos-test.ufileos.com	胡志明市	小set	公开空间		2019-10-13	详情 文件管理 域名管理 ...

在“创建存储空间”弹窗页中,可以自定义存储地域、空间类型、业务组、存储空间域名,选择是否授权开启数据分析。

创建存储空间 ✕

选择地域

北京

香港

广州

上海二

洛杉矶

新加坡

雅加达

拉各斯

圣保罗

迪拜

胡志明市

台北

孟买

华盛顿

法兰克福

首尔

空间类型 ?

私有空间

公开空间

业务组

未分组 v

存储空间域名* ?

.cn-bj.ufileos.com

版本控制

未开启

! 1.开启了多版本建议同时开启生命周期，清理过期脏数据。2.多版本开启以后，不支持关闭。3.多版本状态切换，5分钟生效。 ✕

数据分析 ?

数据湖分析USQL

未授权

取消

确定

字段	说明
选择地域	您可就近选择一个地域,作为存储空间中文件对象的物理存储位置,存储空间创建后无法更换地域。如需要通过 UHost 云主机内网访问 US3 对象存储,需要选择与您 UHost 节点相同的地域。

空间类型	公开空间:所有文件可通过URL直接访问。 私有空间:所有文件须获得拥有者的 API 密钥授权才能访问,支持通过密钥构建临时URL提供用户访问。
业务组	您可选择或创建一个业务组,将存储空间按业务组进行归类,便于查询。
存储空间域名	填写存储空间对应的域名地址作为对外的访问地址,由于存储空间域名全局唯一,遇到冲突请更换名称。

确定后即创建存储空间成功。存储空间创建成功后,用户在控制台 SDK 及工具页下载客户端管理工具进行空间和文件管理操作,同时也可以使用 API 或 SDK 进行空间和文件管理操作。

删除存储空间

选择指定存储空间,在右侧操作栏中点击“删除存储空间”。

对象存储 US3

切换旧版

单地域空间管理

全球化空间管理

统计报表

管理工具及SDK

令牌管理

创建存储空间

更改业务组

删除存储空间

已选择 1 取消选择

<input type="checkbox"/>	存储空间域名	地域	业务组	空间类型	数据分析	创建时间	操作
<input type="checkbox"/>	gd.ufileos.com	广州	HadoopTest	私有空间		2019-06-05	详情 文件管理 域名管理 ...
<input checked="" type="checkbox"/>	cn-s1.ufileos.com	上海二	未分组	公开空间		2019-06-27	详情 文件管理 域名管理 ...
<input type="checkbox"/>	bj.ufileos.com	北京	未分组	公开空间		2019-07-05	详情 文件管理 域名管理 ...
<input type="checkbox"/>	cn-bj.ufileos.com	北京	未分组	公开空间		2019-08-08	详情 文件管理 域名管理 ...
<input type="checkbox"/>	ufileos.com	北京	未分组	私有空间		2019-09-02	详情 文件管理 域名管理 ...
<input type="checkbox"/>	tp.ufileos.com	台北	未分组	公开空间		2019-09-11	详情 文件管理 域名管理 ...
<input type="checkbox"/>	bj.ufileos.com	北京	S3FS	私有空间		2019-09-25	详情 文件管理 域名管理 ...
<input type="checkbox"/>	cn-bj.ufileos.com	北京	未分组	私有空间		2019-09-25	详情 文件管理 域名管理 ...
<input type="checkbox"/>	ufileos.com	上海二	未分组	私有空间		2019-09-30	详情 文件管理 域名管理 ...
<input type="checkbox"/>	ufileos.com	胡志明市	小set	公开空间		2019-10-13	详情 文件管理 域名管理 ...

如需批量删除,选择多个存储空间后,点击左上角操作栏的“删除存储空间”。

对象存储 US3

切换旧版

单地域空间管理

全球化空间管理

统计报表

管理工具及SDK

令牌管理

创建存储空间

更改业务组

删除存储空间

已选择 4 取消选择

<input type="checkbox"/>	存储空间域名	地域	业务组	空间类型	数据分析	创建时间	操作
<input checked="" type="checkbox"/>	store-xxxxx-gd.ufileos.com	广州	HadoopTest	私有空间		2019-06-05	详情 文件管理 域名管理 ...
<input checked="" type="checkbox"/>	xxxxx-sh2.ufileos.com	上海二	未分组	公开空间		2019-06-27	详情 文件管理 域名管理 ...
<input checked="" type="checkbox"/>	xxxxx-cn-bj.ufileos.com	北京	未分组	公开空间		2019-07-05	详情 文件管理 域名管理 ...
<input checked="" type="checkbox"/>	xxxxx-cn-bj.ufileos.com	北京	未分组	公开空间		2019-08-08	详情 文件管理 域名管理 ...
<input type="checkbox"/>	xxxxx-cn-bj.ufileos.com	北京	未分组	私有空间		2019-09-02	详情 文件管理 域名管理 ...
<input type="checkbox"/>	xxxxx-tp.ufileos.com	台北	未分组	公开空间		2019-09-11	详情 文件管理 域名管理 ...
<input type="checkbox"/>	xxxxx-cn-bj.ufileos.com	北京	S3FS	私有空间		2019-09-25	详情 文件管理 域名管理 ...
<input type="checkbox"/>	xxxxx-cn-bj-1-cn-bj.ufileos.com	北京	未分组	私有空间		2019-09-25	详情 文件管理 域名管理 ...
<input type="checkbox"/>	xxxxx-cn-bj.ufileos.com	上海二	未分组	私有空间		2019-09-30	详情 文件管理 域名管理 ...
<input type="checkbox"/>	xxxxx-hk.ufileos.com	胡志明市	小set	公开空间		2019-10-13	详情 文件管理 域名管理 ...

查看存储空间

在存储空间列表页,可查看存储空间的基本信息,如存储空间域名、地域、空间类型、创建时间等。

对象存储 US3

切换旧版

单地域空间管理

全球化空间管理

统计报表

管理工具及SDK

令牌管理

创建存储空间

更改业务组

删除存储空间

<input type="checkbox"/>	存储空间域名	地域	业务组	空间类型	数据分析	创建时间	操作
<input type="checkbox"/>	ufileaks.com	广州	HadoopTest	私有空间		2019-06-05	详情 文件管理 域名管理 ...
<input type="checkbox"/>	ufileaks.com	上海二	未分组	公开空间		2019-06-27	详情 文件管理 域名管理 ...
<input type="checkbox"/>	ufileaks.com	北京	未分组	公开空间		2019-07-05	详情 文件管理 域名管理 ...
<input type="checkbox"/>	-bj.ufileaks.com	北京	未分组	公开空间		2019-08-08	详情 文件管理 域名管理 ...
<input type="checkbox"/>	-bj.ufileaks.com	北京	未分组	私有空间		2019-09-02	详情 文件管理 域名管理 ...
<input type="checkbox"/>	tp.ufileaks.com	台北	未分组	公开空间		2019-09-11	详情 文件管理 域名管理 ...
<input type="checkbox"/>	-bj.ufileaks.com	北京	S3FS	私有空间		2019-09-25	详情 文件管理 域名管理 ...
<input type="checkbox"/>	-bj.ufileaks.com	北京	未分组	私有空间		2019-09-25	详情 文件管理 域名管理 ...
<input type="checkbox"/>	ufileaks.com	上海二	未分组	私有空间		2019-09-30	详情 文件管理 域名管理 ...
<input type="checkbox"/>	ufileaks.com	胡志明市	小set	公开空间		2019-10-13	详情 文件管理 域名管理 ...

字段

说明

存储空间域名	存储空间对应的域名地址作为对外的访问地址。
地域	存储空间中文件对象的物理存储位置,存储空间创建后无法更换地域。如需要通过 UHost 云主机内网访问 US3 对象存储,需要选择与您 UHost 节点相同的地域。
业务组	存储空间所在的业务组名称。
空间类型	公开空间:所有文件可通过URL直接访问。 私有空间:所有文件须获得拥有者的 API 密钥授权才能访问,支持通过密钥构建临时URL提供用户访问。
创建时间	存储空间的创建时间。

点击存储空间名称或侧操作栏中点击“详情”,可进入存储空间详情页可查看基本信息和监控信息。

修改存储空间类型

选择指定存储空间,在右侧操作中点击修改空间类型。

对象存储 US3

切换旧版

单地域空间管理

全球化空间管理

统计报表

管理工具及SDK

令牌管理

创建存储空间

更改业务组

删除存储空间

已选择 1 取消选择

存储空间域名	地域	业务组	空间类型	数据分析	创建时间	操作
...	广州	HadoopTest	私有空间		2019-06-05	详情 文件管理 域名管理 ...
...	上海二	未分组	公开空间		2019-06-27	详情 文件管理 域名管理 ...
...	北京	未分组	公开空间		2019-07-05	详情 文件管理 域名管理 ...
...	北京	未分组	公开空间		2019-08-08	详情 文件管理 域名管理 ...
...	北京	未分组	私有空间		2019-09-02	详情 文件管理 域名管理 ...
...	台北	未分组	公开空间		2019-09-11	详情 文件管理 域名管理 ...
...	北京	S3FS	私有空间		2019-09-25	详情 文件管理 域名管理 ...
...	北京	未分组	私有空间		2019-09-25	详情 文件管理 域名管理 ...
...	上海二	未分组	私有空间		2019-09-30	详情 文件管理 域名管理 ...
...	胡志明市	小set	公开空间		2019-10-13	详情 文件管理 域名管理 ...

- 生命周期
- 镜像回源
- 跨域设置
- 修改空间类型**
- 更改业务组
- 删除存储空间

空间类型	说明
------	----

公开空间	公共读私有写,即所有文件可通过未携带签名信息的URL直接访问(包括图片处理),其他的操作如拉取列表、删除、覆盖操作需要携带签名信息。
私有空间	私有读写,即所有文件、列表操作必须携带签名信息。

注: 由于政策限制, 个人用户无法在海外创建公开空间, 如需使用请升级认证企业用户。

存储空间命名必须符合名称规范,规范如下:

- 存储空间命名长度为3~63字节
- 只能包含小写字母, 数字, 连接符(短横线)
- 首尾字符不能是短横线
- 空间名称不能以www或者地域名称为开头

日志管理

用户在访问 US3 文件时,会产生大量的访问日志,可以开启日志管理功能。开启后该空间的访问日志,会以约2分钟为单位,按照固定的命名,存储到用户指定的空间中。请求日志会尽最大可能传输到您的桶中,不过可能会出现少量的日志延迟或丢失。

设置日志存储

选择对应空间,在右侧操作中点击日志管理按钮

对象存储 US3

切换旧版

单地域空间管理

全球化空间管理

统计报表

管理工具及SDK

令牌管理

创建存储空间

更改业务组

删除存储空间

已选择 1 取消选择

存储空间域名	地域	业务组	空间类型	数据分析	创建时间	操作
[redacted].ufileos.com	北京	未分组	私有空间	[icon]	2019-03-19	详情 文件管理 域名管理 ...
[redacted].ufileos.com	北京	byh-test	公开空间	[icon]	2019-06-14	详情 文件管理 域名管理 ...
[redacted].ufileos.com	香港	未分组	公开空间	[icon]	2019-06-25	详情 文件管理 域名管理 ...
[redacted].ufileos.com	法兰克福	hongshi-test	私有空间	[icon]	2019-07-31	详情 文件管理 域名管理 ...
[redacted].com	新加坡	未分组	公开空间	[icon]	2019-10-22	详情 文件管理 域名管理 ...
[redacted].ufileos.com	首尔	未分组	公开空间	[icon]	2019-12-09	详情 文件管理 域名管理 ...
[redacted].ufileos.com	拉各斯	hongshi-test	私有空间	[icon]	2019-12-31	详情 文件管理 域名管理 ...
[redacted].ufileos.com	首尔	未分组	私有空间	[icon]	2020-06-25	详情 文件管理 域名管理 ...
[redacted].ufileos.com	北京	import	公开空间	[icon]	2020-10-19	详情 文件管理 域名管理 ...
[redacted].ufileos.com	北京	未分组	私有空间	[icon]	2020-10-22	详情 文件管理 域名管理 ...

- 生命周期
- 镜像回源
- 跨域设置
- 日志管理
- 修改空间类型
- 更改业务组
- 删除存储空间

点击开启日志存储功能按钮

对象存储 US3

切换旧版

日志管理

存储直接访问该空间的日志, 更多请参考日志管理 使用指南

日志存储功能 ON

日志存储空间 listtest

日志前缀 ufile-log/

取消 确定

存储空间	地域	分组	空间类型	状态	创建时间	操作
存储空间域名 1						详情 文件管理 域名管理 ...
存储空间域名 2	香港	未分组	公开空间	正常	2019-06-25	详情 文件管理 域名管理 ...
存储空间域名 3	法兰克福	hongshi-test	私有空间	正常	2019-07-31	详情 文件管理 域名管理 ...
存储空间域名 4	新加坡	未分组	公开空间	正常	2019-10-22	详情 文件管理 域名管理 ...
存储空间域名 5	首尔	未分组	公开空间	正常	2019-12-09	详情 文件管理 域名管理 ...
存储空间域名 6	拉各斯	hongshi-test	私有空间	正常	2019-12-31	详情 文件管理 域名管理 ...
存储空间域名 7	首尔	未分组	私有空间	正常	2020-06-25	详情 文件管理 域名管理 ...
存储空间域名 8	北京	import	公开空间	正常	2020-10-19	详情 文件管理 域名管理 ...

选择日志存储空间和填写日志前缀, 日志存储空间默认为当前空间, 日志前缀默认ufile-log/, 前缀可以为空。例: 填写前缀为ufile-log/2019/, 日志生成后将被存储在该填写的目录下。

日志命名规则

\<TargetPrefix\>\<SourceBucket\>-YYYY-MM-DD-HH-UniqueString.csv

命名规则字段说明:

字段	说明
TargetPrefix	由用户指定,表示存储访问日志记录的名称前缀,可以为空
YYYY-MM-DD-HH	分别是该日志被创建时的阿拉伯数字的年、月、日、小时
UniqueString	为系统生成的字符串,用于唯一标识该 Log 文件

存储空间访问日志的名称例子如下: ufile-log/SourceBucket-2018-10-22-08-537928a6-8be3-462c-a93c-b531028fe3fb.csv

日志格式和示例

日志以csv格式提供,日志首行为日志头,对应下面日志正文每一列的含义。

日志头如下:

Action,Bucket,Key,Host,UserAgent,RemoteIP,Referer,HttpStatus,TimeStamp,CostTime,ErrCode,ErrMsg,IsUCDN,FileSize,StorageClass,CreatedTime,RequestSize,To
kenPublicKey,SessionId

字段描述

字段	示例	说明
----	----	----

Action	LIST_OBJECTS	请求的接口
Bucket	my-bucket	请求的存储桶
Key	example-object	请求的对象名称
Host	example-bucket.cn-bj.ufileos.com	请求访问的目标域名
UserAgent	curl/7.77.0	HTTP的User-Agent头
RemoteIP	192.168.0.1	请求者的IP
Referer	https://console.ucloud.cn/	请求的HTTP Referer标头
HttpStatus	200	HTTP状态码
TimeStamp	1654653684	请求的时间戳
CostTime	13	请求耗时(ms)
ErrCode	0	请求错误码(0为没有报错)
ErrMsg	success	请求错误信息(success为请求成功)
IsUCDN	false	是否使用了UCDN
FileSize	1024	文件大小(B),非GET请求时此值为0
StorageClass	STANDARD	文件存储类型:STANDARD表示标准存储;IA表示低频访问存储;ARCHIVE表示归档存储。非GET请求时此值为UNKNOWN
CreatedTime	1654653681	文件创建时间,非GET请求时此值为0
RequestSize	4096	本次上传/下载请求时的大小, 仅在执行PutFile, PostFile, UploadPart, GetFile四种上传下载请求时有意义

TokenPublicKey	Token_xxxx_xxx_xxx	请求使用的token公钥
SessionId	d0e27b98-6d78-4a8d-ac3c-9dd3e3e7b405	请求的session id

日志示例

```

Action,Bucket,Key,Host,UserAgent,RemoteIP,Referer,HttpStatus,TimeStamp,CostTime,ErrMsg,IsUCDN,FileSize,StorageClass,CreateTime,RequestSize,TokenPublicKey,SessionId
LIST_OBJECTS,example-bucket,,example-bucket.cn-bj.ufileos.com,Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML like Gecko) Chrome/102.0.5005.61 Safari/537.36,10.75.220.2,https://console.ucloud.cn/,200,1654653684,13,0,success,false,0,UNKNOWN,0, 0, Token_ABCDEFG, d0e27b98-6d78-4a8d-ac3c-9dd3e3e7b405
OPTIONS,example-bucket,,example-bucket.cn-bj.ufileos.com,Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML like Gecko) Chrome/102.0.5005.61 Safari/537.36,10.75.220.2,https://console.ucloud.cn/,200,1654653684,0,0,success,false,0,UNKNOWN,0, 0, Token_ABCDEFG, d0e27b98-6d78-4a8d-ac3c-9dd3e3e7b405
OPTIONS,example-bucket,,example-bucket.cn-bj.ufileos.com,Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML like Gecko) Chrome/102.0.5005.61 Safari/537.36,10.75.220.2,https://console.ucloud.cn/,200,1654653681,0,0,success,false,0,UNKNOWN,0, 0, Token_ABCDEFG, d0e27b98-6d78-4a8d-ac3c-9dd3e3e7b405
MPUT_DATA,example-bucket,2022-05-26-17-00000000.csv,example-bucket.ufileos.com,Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML like Gecko) Chrome/110.0.0.0 Safari/537.36,106.75.220.2,https://console.ucloud.cn/,200,1676257320,303,0,success,false,0,UNKNOWN,0,325,Token_xxxxxxxxxxxx,9bbff734-3555-4d29-a75b-e4221f1a3e9c

```


文件管理

选择指定存储空间,在右侧操作中单击“文件管理”按钮进入文件管理页。



对象存储 US3

单地域空间管理 | 统计报表 | 管理工具及SDK | 令牌管理

创建存储空间 | 更改业务组 | 删除存储空间

存储空间域名	资源ID	地域	默认存储类型	空间类型	数据分析	创建时间	操作
[redacted]	ufile-4gzmzhle	广州	标准	公开空间	📊	2019-05-17	详情 文件管理 域名管理 ...
[redacted].m	ufile-z4ul3q2x	广州	标准	公开空间	📊	2019-07-02	详情 文件管理 域名管理 ...
[redacted].s.com	ufile-rbzizzsv	胡志明市	标准	私有空间	📊	2019-09-11	详情 文件管理 域名管理 ...
[redacted]....	ufile-synxed0i	伦敦	标准	私有空间	📊	2019-10-17	详情 文件管理 域名管理 ...
a.[redacted]	ufile-mycrpw0l	华北一	标准	私有空间	📊	2020-01-15	详情 文件管理 域名管理 ...
[redacted]	ufile-druv3yvi	华北一	标准	公开空间	📊	2020-02-13	详情 文件管理 域名管理 ...
[redacted]	ufile-upwaga2u	华北一	标准	公开空间	📊	2020-02-14	详情 文件管理 域名管理 ...
[redacted].com	ufile-p3amd1lg	上海二	标准	私有空间	📊	2020-02-17	详情 文件管理 域名管理 ...
te.[redacted]	ufile-ebbnpufw	华北一	标准	公开空间	📊	2020-03-17	详情 文件管理 域名管理 ...
t.[redacted]	ufile-3zgqxbv2	新加坡	标准	公开空间	📊	2020-03-18	详情 文件管理 域名管理 ...

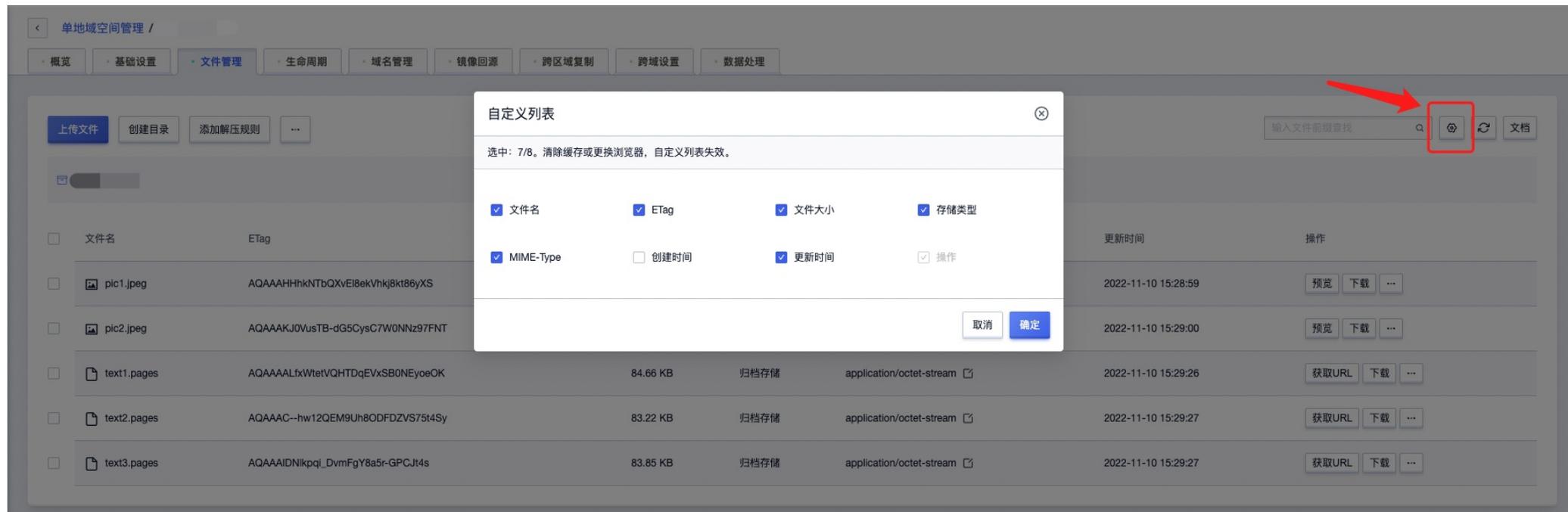
< 1 2 3 4 5 ... 22 > 10条/页 /22

在文件管理列表页,可查看文件的基本信息,如文件名、ETag、文件大小、存储类型、MIME-TYPE、更新时间等。

The screenshot shows the 'File Management' (文件管理) tab in the UCloud console. The interface includes a top navigation bar with '全部产品', '研发短期测试 (小于7天)', and '全球服务'. Below the navigation bar, there are tabs for '概览', '基础设置', '文件管理', '生命周期', '域名管理', '镜像回源', '跨区域复制', '跨域设置', and '数据处理'. The '文件管理' tab is active, displaying a list of files. The table has the following columns: '文件名' (Filename), 'ETag', '文件大小' (File Size), '存储类型' (Storage Type), 'MIME-Type', '更新时间' (Update Time), and '操作' (Actions). A red box highlights the header row. The table contains five rows of files, including 'pic1.jpeg', 'pic2.jpeg', 'text1.pages', 'text2.pages', and 'text3.pages'. Each row has corresponding action buttons like '预览', '下载', '获取URL', and '更多'.

文件名	ETag	文件大小	存储类型	MIME-Type	更新时间	操作
<input type="checkbox"/> pic1.jpeg	AQAAAHhkNTbQXvEi8ekVhkj8kt86yXS	83.28 KB	标准存储	image/jpeg	2022-11-10 15:28:59	<input type="button" value="预览"/> <input type="button" value="下载"/> <input type="button" value="更多"/>
<input type="checkbox"/> pic2.jpeg	AQAAAKJ0VusTB-dG5CysC7W0NNz97FNT	89 KB	标准存储	image/jpeg	2022-11-10 15:29:00	<input type="button" value="预览"/> <input type="button" value="下载"/> <input type="button" value="更多"/>
<input type="checkbox"/> text1.pages	AQAAAAALfxWtetVQHTDqEVxSB0NEyoeOK	84.66 KB	归档存储	application/octet-stream	2022-11-10 15:29:26	<input type="button" value="获取URL"/> <input type="button" value="下载"/> <input type="button" value="更多"/>
<input type="checkbox"/> text2.pages	AQAAAC--hw12QEM9Uh8ODFDZVS75t4Sy	83.22 KB	归档存储	application/octet-stream	2022-11-10 15:29:27	<input type="button" value="获取URL"/> <input type="button" value="下载"/> <input type="button" value="更多"/>
<input type="checkbox"/> text3.pages	AQAAAIIDNikpqI_DvmFgY8a5r-GPCJt4s	83.85 KB	归档存储	application/octet-stream	2022-11-10 15:29:27	<input type="button" value="获取URL"/> <input type="button" value="下载"/> <input type="button" value="更多"/>

单击文件管理右上角的设置按钮,可以自定义文件列表。



上传文件

在文件管理页单击“上传文件”，弹窗中可以设置路径前缀、对象标签以及选择存储类型。

其中“继承Bucket”是指以Bucket存储类型为准，如需了解存储类型相关说明，请参见存储类型

上传文件

警告：严禁储存、处理、传输、发布任何涉密、色情、暴力、侵权等违法违规信息

基础设置

上传位置

选择存储类型

US3提供标准、低频、归档三种存储类型。如何选择适合的存储类型？

对象标签

针对Object设置标签，详情可参考 [对象标签文档](#)

键	值
+	

文件上传

拖拽目录或者文件到这里

超过500M的文件请用[工具上传](#)

修改 MIME-Type

直接单击“MIME-Type”下对应文件的修改按钮，或者勾选文件后单击“修改MIME-Type”即可完成修改。

单地域空间管理 / ...

概览 基础设置 文件管理 生命周期 域名管理 镜像回源 跨区域复制 跨域设置 数据处理

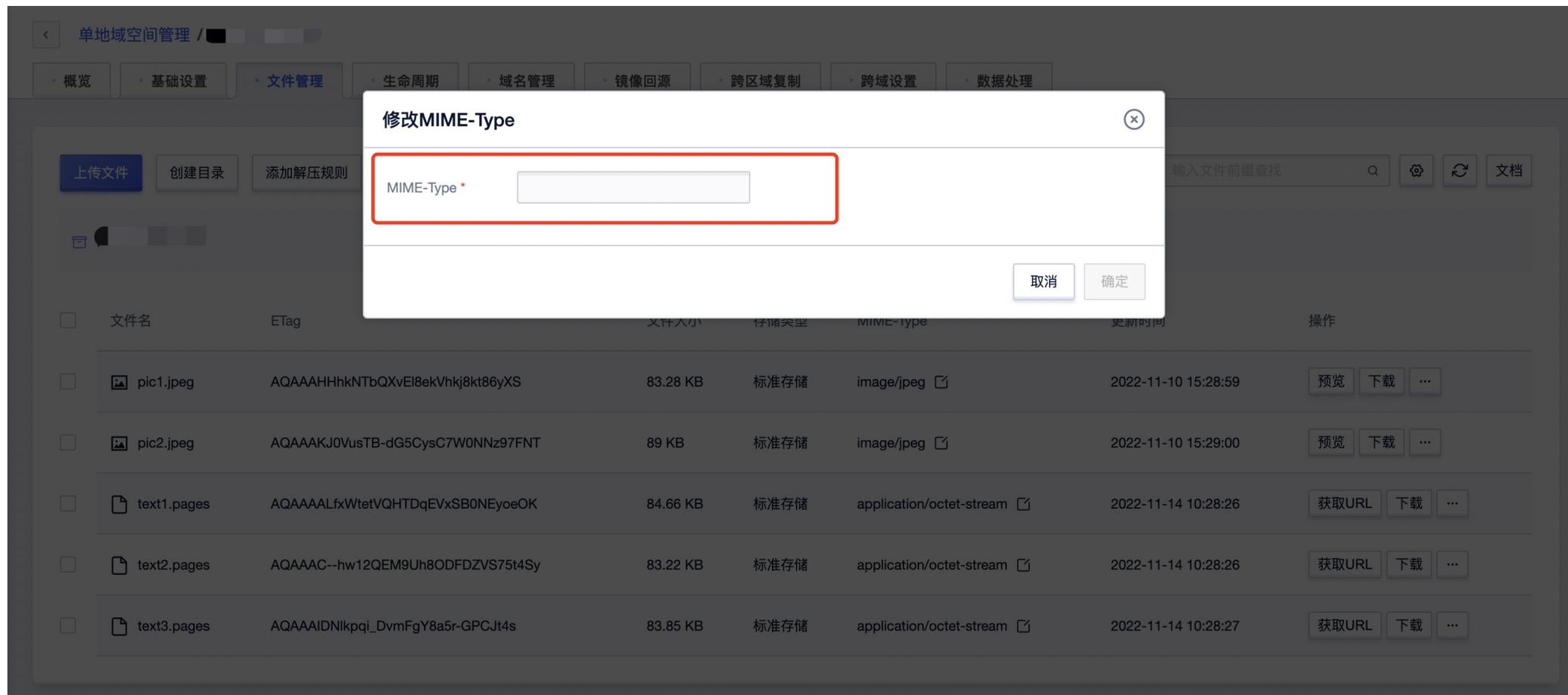
上传文件 创建目录 添加解压规则 ...

输入文件前缀查找

已选择 1 取消选择

- 下载
- 导出URL列表
- 修改MIME-Type
- 删除

文件名	ETag	文件大小	存储类型	MIME-Type	更新时间	操作
<input checked="" type="checkbox"/>  pic1.jpeg	AQAAAHHkNTbQXvEi8ekVhkJ8kt86yXS	83.28 KB	标准存储	image/jpeg 	2022-11-10 15:28:59	预览 下载 ...
<input type="checkbox"/>  pic2.jpeg	AQAAAKJ0VusTB-dG5CysC7W0NNz97FNT	89 KB	标准存储	image/jpeg 	2022-11-10 15:29:00	预览 下载 ...
<input type="checkbox"/>  text1.pages	AQAAAALfxWtetVQHTDqEVxSB0NEyoeOK	84.66 KB	归档存储	application/octet-stream 	2022-11-10 15:29:26	获取URL 下载 ...
<input type="checkbox"/>  text2.pages	AQAAAC--hw12QEM9Uh8ODFDZVS75t4Sy	83.22 KB	归档存储	application/octet-stream 	2022-11-10 15:29:27	获取URL 下载 ...
<input type="checkbox"/>  text3.pages	AQAAAIIDNikpqj_DvmFgY8a5r-GPCJt4s	83.85 KB	归档存储	application/octet-stream 	2022-11-10 15:29:27	获取URL 下载 ...



修改存储类型

直接单击“操作”下的“...”选择“修改存储类型”即可完成修改。

单地域空间管理 /

概览 基础设置 文件管理 生命周期 域名管理 镜像回源 跨区域复制 跨域设置 数据处理

上传文件 创建目录 添加解压规则 ...

输入文件前缀查找

已选择 1 取消选择

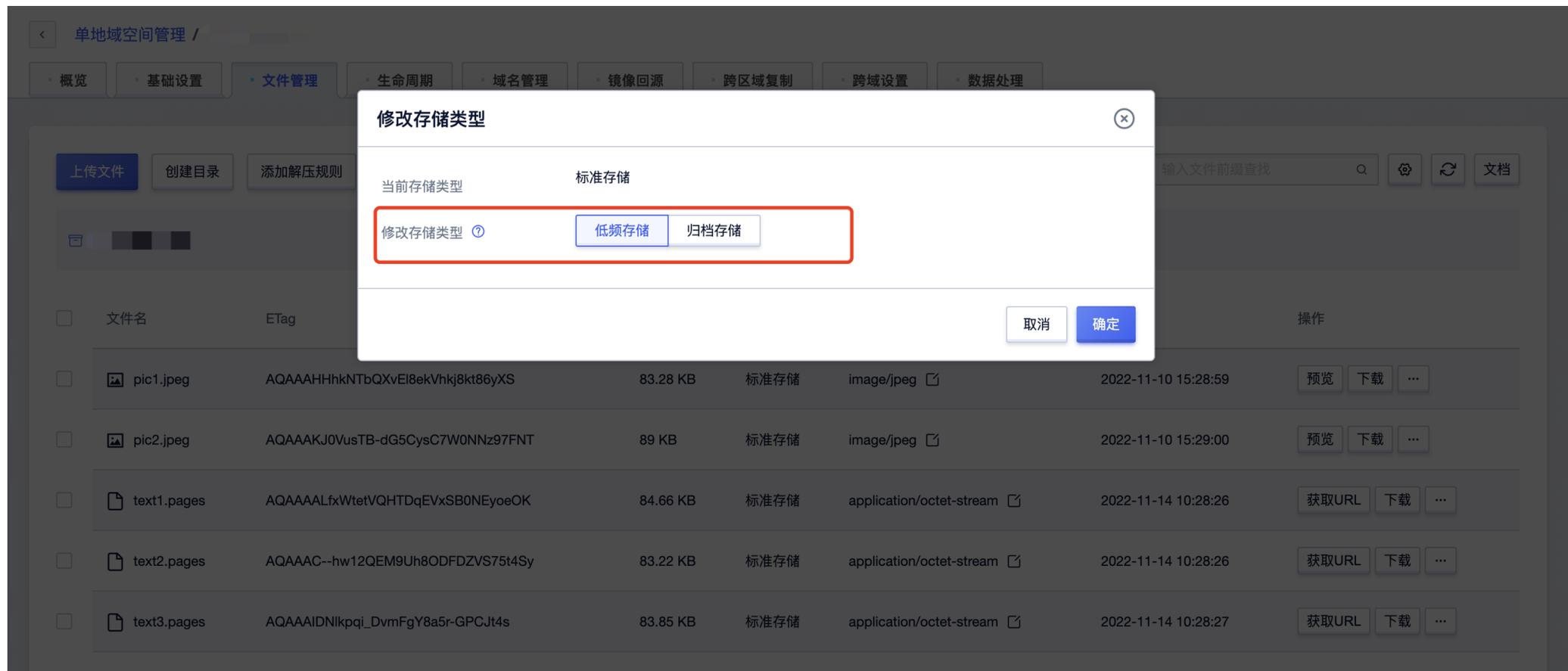
文件名	ETag	文件大小	存储类型	MIME-Type	更新时间	操作
<input checked="" type="checkbox"/>  pic1.jpeg	AQAAAHhkNTbQXvEI8ekVhkj8kt86yXS	83.28 KB	标准存储	image/jpeg 	2022-11-10 15:28:59	预览 下载 ... 
<input type="checkbox"/>  pic2.jpeg	AQAAAKJ0VusTB-dG5CysC7W0NNz97FNT	89 KB	标准存储	image/jpeg 	2022-11-10 15:29:00	预览 下载
<input type="checkbox"/>  text1.pages	AQAAAALfxWtetVQHTDqEVxSB0NEyoeOK	84.66 KB	归档存储	application/octet-stream 	2022-11-10 15:29:26	获取URL 下 获取URL
<input type="checkbox"/>  text2.pages	AQAAAC--hw12QEM9Uh8ODFDZVS75t4Sy	83.22 KB	归档存储	application/octet-stream 	2022-11-10 15:29:27	获取URL 下载 ...
<input type="checkbox"/>  text3.pages	AQAAAIDNikpqj_DvmFgY8a5r-GPCJt4s	83.85 KB	归档存储	application/octet-stream 	2022-11-10 15:29:27	获取URL 下载 ...

修改存储类型

删除

获取URL





删除文件

单个文件删除:直接单击“操作”下的“...”选择“删除”即可。

< 单地域空间管理 /

概览 基础设置 文件管理 生命周期 域名管理 镜像回源 跨区域复制 跨域设置 数据处理

上传文件 创建目录 添加解压规则 ...

输入文件前缀查找

<input type="checkbox"/>	文件名	ETag	文件大小	存储类型	MIME-Type	更新时间	操作
<input type="checkbox"/>	 pic1.jpeg	AQAAAHhkNTbQXvEI8ekVhkj8kt86yXS	83.28 KB	标准存储	image/jpeg 	2022-11-10 15:28:59	预览 下载 ...
<input type="checkbox"/>	 pic2.jpeg	AQAAAKJ0VusTB-dG5CysC7W0NNz97FNT	89 KB	标准存储	image/jpeg 	2022-11-10 15:29:00	预览 下载 ...
<input type="checkbox"/>	 text1.pages	AQAAAALfxWtetVQHTDqEVxSB0NEyoeOK	84.66 KB	标准存储	application/octet-stream 	2022-11-14 10:28:26	获取URL 下载 获取URL
<input type="checkbox"/>	 text2.pages	AQAAAC--hw12QEM9Uh8ODFDZVS75t4Sy	83.22 KB	标准存储	application/octet-stream 	2022-11-14 10:28:26	获取URL 下载 ...
<input type="checkbox"/>	 text3.pages	AQAAAIDNlkpqj_DvmFgY8a5r-GPCJt4s	83.85 KB	标准存储	application/octet-stream 	2022-11-14 10:28:27	获取URL 下载 ...

修改存储类型 删除

批量文件删除：勾选待删除文件后单击“删除”即可。但需注意目前仅支持空目录删除，删除后将不可恢复。

< 单地域空间管理

- 概览
- 基础设置
- 文件管理
- 生命周期
- 域名管理
- 镜像回源
- 跨区域复制
- 跨域设置
- 数据处理

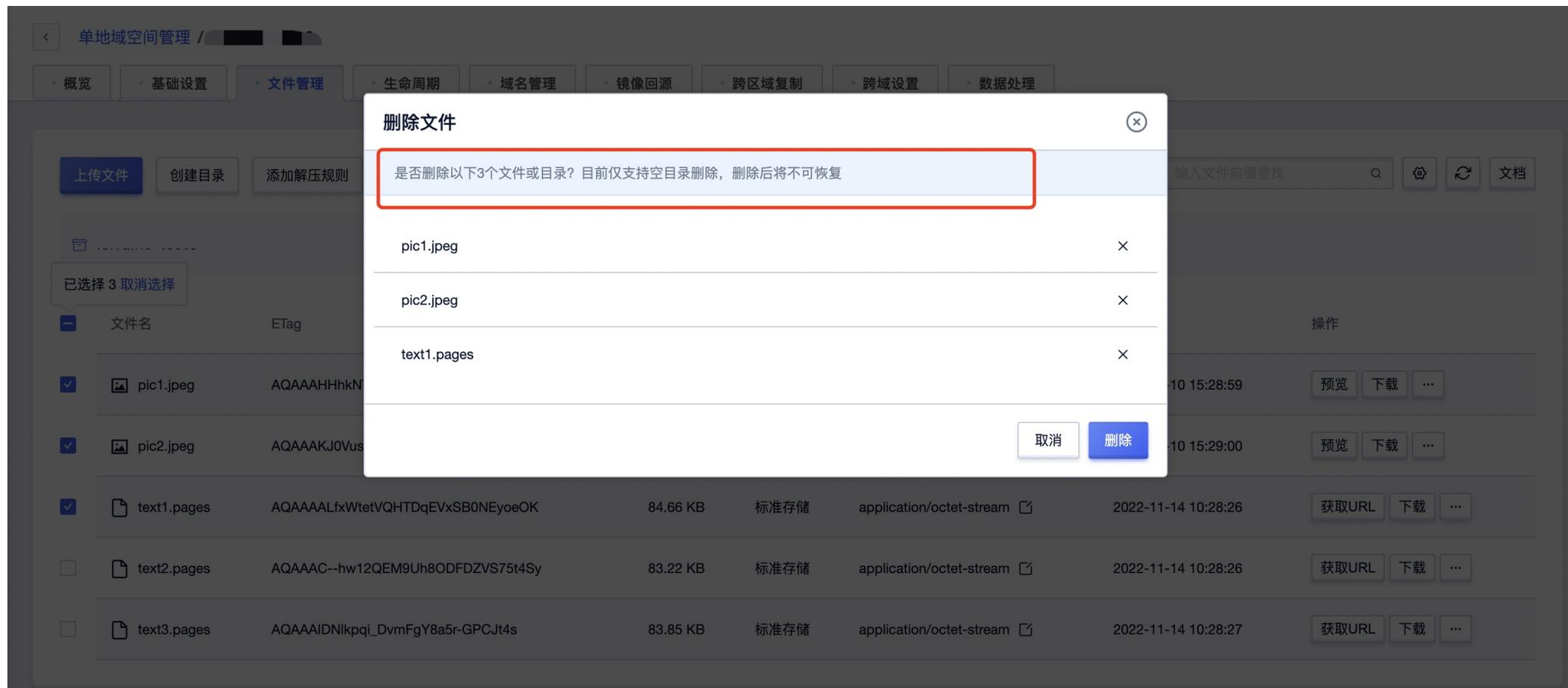
上传文件 创建目录 添加解压规则 ...

输入文件前缀查找

已选择 3 取消选择

- 下载
- 导出URL列表
- 修改MIME-Type
- 删除

<input type="checkbox"/>	文件名	ETag	文件大小	存储类型	MIME-Type	更新时间	操作
<input checked="" type="checkbox"/>	pic1.jpeg	AQAAAHHkNTbQXvEi8ekVhkj8kt86yXS	83.28 KB	标准存储	image/jpeg	2022-11-10 15:28:59	<input type="button" value="预览"/> <input type="button" value="下载"/> ...
<input checked="" type="checkbox"/>	pic2.jpeg	AQAAAKJ0VusTB-dG5CysC7W0NNz97FNT	89 KB	标准存储	image/jpeg	2022-11-10 15:29:00	<input type="button" value="预览"/> <input type="button" value="下载"/> ...
<input checked="" type="checkbox"/>	text1.pages	AQAAAALfxWtetVQHTDqEVxSB0NEyoeOK	84.66 KB	标准存储	application/octet-stream	2022-11-14 10:28:26	<input type="button" value="获取URL"/> <input type="button" value="下载"/> ...
<input type="checkbox"/>	text2.pages	AQAAAC--hw12QEM9Uh8ODFDZVS75t4Sy	83.22 KB	标准存储	application/octet-stream	2022-11-14 10:28:26	<input type="button" value="获取URL"/> <input type="button" value="下载"/> ...
<input type="checkbox"/>	text3.pages	AQAAAIDNkppqj_DvmFgY8a5r-GPCJt4s	83.85 KB	标准存储	application/octet-stream	2022-11-14 10:28:27	<input type="button" value="获取URL"/> <input type="button" value="下载"/> ...



下载文件

单个文件下载:直接单击“操作”下的“下载”即可。

< 单地域空间管理 /

· 概览 · 基础设置 · 文件管理 · 生命周期 · 域名管理 · 镜像回源 · 跨区域复制 · 跨域设置 · 数据处理

上传文件 创建目录 添加解压规则 ...

输入文件前缀查找 🔍 🔄 📄

<input type="checkbox"/>	文件名	ETag	文件大小	存储类型	MIME-Type	更新时间	操作
<input type="checkbox"/>	 pic1.jpeg	AQAAAHhkNTbQXvEi8ekVhkj8kt86yXS	83.28 KB	标准存储	image/jpeg 🔗	2022-11-10 15:28:59	<input type="button" value="预览"/> <input type="button" value="下载"/> <input type="button" value="..."/>
<input type="checkbox"/>	 pic2.jpeg	AQAAAKJ0VusTB-dG5CysC7W0NNz97FNT	89 KB	标准存储	image/jpeg 🔗	2022-11-10 15:29:00	<input type="button" value="预览"/> <input type="button" value="下载"/> <input type="button" value="..."/>
<input type="checkbox"/>	 text1.pages	AQAAAALfxWtetVQHTDqEvxSB0NEyoeOK	84.66 KB	归档存储	application/octet-stream 🔗	2022-11-10 15:29:26	<input type="button" value="获取URL"/> <input type="button" value="下载"/> <input type="button" value="..."/>
<input type="checkbox"/>	 text2.pages	AQAAAC--hw12QEM9Uh8ODFDZVS75t4Sy	83.22 KB	归档存储	application/octet-stream 🔗	2022-11-10 15:29:27	<input type="button" value="获取URL"/> <input type="button" value="下载"/> <input type="button" value="..."/>
<input type="checkbox"/>	 text3.pages	AQAAAIIDNlkpqj_DvmFgY8a5r-GPCJt4s	83.85 KB	归档存储	application/octet-stream 🔗	2022-11-10 15:29:27	<input type="button" value="获取URL"/> <input type="button" value="下载"/> <input type="button" value="..."/>

批量文件下载:勾选待下载文件后单击“下载”即可,下载可能存在延迟敬请谅解。但需注意选中的文件中如果包含归档/解冻的文件,请先完成解冻后再进行下载。

单地域空间管理 / 1

· 概览 · 基础设置 · 文件管理 · 生命周期 · 域名管理 · 镜像回源 · 跨区域复制 · 跨域设置 · 数据处理

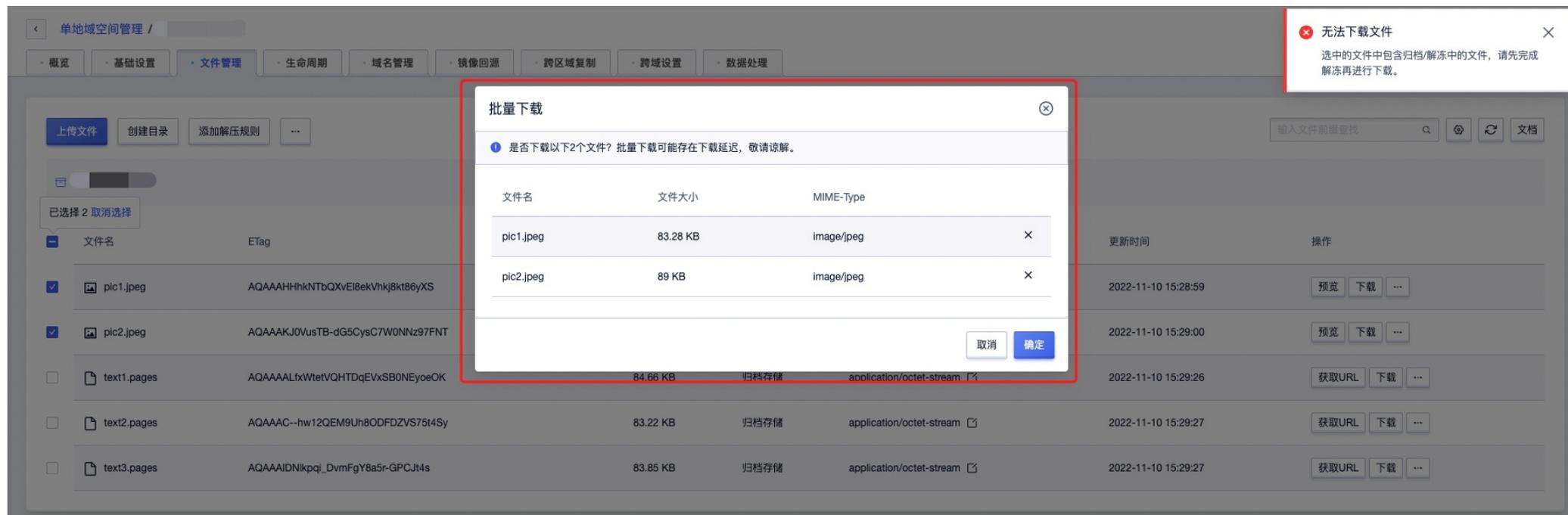
上传文件 创建目录 添加解压规则 ...

输入文件前缀查找 🔍 🔄 📄

已选择 3 取消选择

下载
导出URL列表
修改MIME-Type
删除

文件名	ETag	文件大小	存储类型	MIME-Type	更新时间	操作
<input checked="" type="checkbox"/>  pic1.jpeg	AQAAAHHkNTbQXvEI8ekVhkj8kt86yXS	83.28 KB	标准存储	image/jpeg 📄	2022-11-10 15:28:59	预览 下载 ...
<input checked="" type="checkbox"/>  pic2.jpeg	AQAAAKJ0VusTB-dG5CysC7W0NNz97FNT	89 KB	标准存储	image/jpeg 📄	2022-11-10 15:29:00	预览 下载 ...
<input checked="" type="checkbox"/>  text1.pages	AQAAAALfxWtetVQHTDqEVxSB0NEyoeOK	84.66 KB	归档存储	application/octet-stream 📄	2022-11-10 15:29:26	获取URL 下载 ...
<input type="checkbox"/>  text2.pages	AQAAAC--hw12QEM9Uh8ODFDZVS75t4Sy	83.22 KB	归档存储	application/octet-stream 📄	2022-11-10 15:29:27	获取URL 下载 ...
<input type="checkbox"/>  text3.pages	AQAAAIIDNkqqi_DvmFgY8a5r-GPCJt4s	83.85 KB	归档存储	application/octet-stream 📄	2022-11-10 15:29:27	获取URL 下载 ...



重命名文件

在同一个存储空间内对文件进行重命名操作, 实际上是通过Copy接口将源文件拷贝至目标文件, 然后通过DeleteFile接口删除源文件来实现。

重命名注意事项:

1. 文件存储类型为低频且存储未满足规定时长, 则会产生存储不足规定时长容量费用。价格详情请参见产品价格;
2. 文件存储类型为归档时, 不支持重命名操作;
3. 重命文件名长度必须在1-1024字节之间, 且不能包含"/", 不允许重名文件名;
4. 文件大小超过100MB, 请使用命令行工具cli、US3 SDK或US3 API等方式, 通过UploadPartCopy接口进行重命名操作。

将鼠标悬停在目标文件上,然后单击重命名图标,对文件进行重命名。重命名时,文件名称需包含后缀。



The screenshot shows a file management interface with a table of files. The table has columns for '文件名' (Filename), 'ETag', '文件大小' (File Size), '存储类型' (Storage Type), 'MIME-Type', '更新时间' (Update Time), and '操作' (Actions). A red arrow points to the '重命名' (Rename) icon in the '操作' column of the first file row.

文件名	ETag	文件大小	存储类型	MIME-Type	更新时间	操作
副本5.pages	AQ...	83.85 KB	标准存储	application/x-iwork-pages-sfp...	2023-10-27 15:07:46	获取URL 下载 ...
副本6.pages	AQA...	83.85 KB	标准存储	application/x-iwork-pages-sfp...	2023-10-27 15:07:47	获取URL 下载 ...
副本7.pages	AQAAAID...	83.85 KB	标准存储	application/x-iwork-pages-sfp...	2023-10-27 15:07:49	获取URL 下载 ...
副本8.pages	AQA...	83.85 KB	标准存储	application/x-iwork-pages-sfp...	2023-10-27 15:07:50	获取URL 下载 ...

获取地址

单个文件获取URL:直接单击“操作”下的“...”选择“获取URL”,弹窗内可以设置传输协议(HTTPS或HTTP)、过期时间(5分钟、30分钟、12小时或自定义)、自有域名,单击“Copy”可复制该URL。

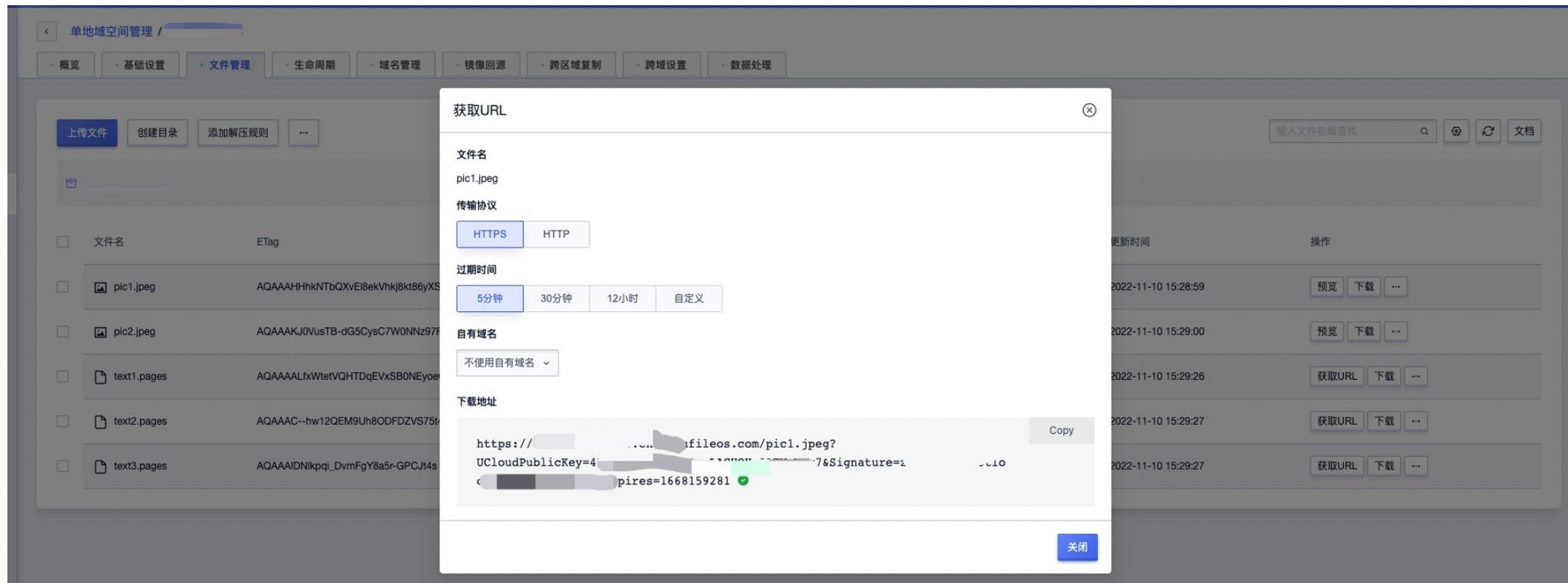
单地域空间管理 / [Progress Bar]

· 概览 · 基础设置 · 文件管理 · 生命周期 · 域名管理 · 镜像回源 · 跨区域复制 · 跨域设置 · 数据处理

上传文件 创建目录 添加解压规则 ...

输入文件前缀查找 🔍 ⚙️ ↻ 文档

<input type="checkbox"/>	文件名	ETag	文件大小	存储类型	MIME-Type	更新时间	操作
<input type="checkbox"/>	pic1.jpeg	AQAAAHhkNTbQXvEI8ekVhkj8kt86yXS	83.28 KB	标准存储	image/jpeg	2022-11-10 15:28:59	预览 下载 ...
<input type="checkbox"/>	pic2.jpeg	AQAAAKJ0VusTB-dG5CysC7W0NNz97FNT	89 KB	标准存储	image/jpeg	2022-11-10 15:29:00	预览 下载
<input type="checkbox"/>	text1.pages	AQAAAALfxWtetVQHTDqEVxSB0NEyoeOK	84.66 KB	归档存储	application/octet-stream	2022-11-10 15:29:26	获取URL 下 获取URL
<input type="checkbox"/>	text2.pages	AQAAAC--hw12QEM9Uh8ODFDZVS75t4Sy	83.22 KB	归档存储	application/octet-stream	2022-11-10 15:29:27	获取URL 下载 ...
<input type="checkbox"/>	text3.pages	AQAAAIIDNlkpqi_DvmFgY8a5r-GPCJ4s	83.85 KB	归档存储	application/octet-stream	2022-11-10 15:29:27	获取URL 下载 ...



批量文件获取URL:勾选待导出URL文件后单击“导出URL列表”,弹窗内可以设置传输协议(HTTPS或HTTP)、过期时间(5分钟、30分钟、12小时或自定义)、自有域名,确认资源后即可导出URL列表。

单地域空间管理 /

· 概览 · 基础设置 · 文件管理 · 生命周期 · 域名管理 · 镜像回源 · 跨区域复制 · 跨域设置 · 数据处理

上传文件 创建目录 添加解压规则 ...

输入文件前缀查找

已选择 5 取消选择

- 下载
- 导出URL列表
- 修改MIME-Type
- 删除

文件名	ETag	文件大小	存储类型	MIME-Type	更新时间	操作
<input checked="" type="checkbox"/>  pic1.jpeg	AQAAAHhkNTbQXvEi8ekVhkj8kt86yXS	83.28 KB	标准存储	image/jpeg	2022-11-10 15:28:59	预览 下载 ...
<input checked="" type="checkbox"/>  pic2.jpeg	AQAAAKJ0VusTB-dG5CysC7W0NNz97FNT	89 KB	标准存储	image/jpeg	2022-11-10 15:29:00	预览 下载 ...
<input checked="" type="checkbox"/>  text1.pages	AQAAAAfxWtetVQHTDqEVxSB0NEyoeOK	84.66 KB	归档存储	application/octet-stream	2022-11-10 15:29:26	获取URL 下载 ...
<input checked="" type="checkbox"/>  text2.pages	AQAAAC--hw12QEM9Uh8ODFDZVS75t4Sy	83.22 KB	归档存储	application/octet-stream	2022-11-10 15:29:27	获取URL 下载 ...
<input checked="" type="checkbox"/>  text3.pages	AQAAAIIDNkqqi_DvmFgY8a5r-GPCJt4s	83.85 KB	归档存储	application/octet-stream	2022-11-10 15:29:27	获取URL 下载 ...

单地域空间管理 /

概览 基础设置 **文件管理** 生命周期 域名管理 镜像回源 跨区域复制

上传文件 创建目录 添加解压规则 ...

已选择 5 取消选择

文件名	ETag
<input checked="" type="checkbox"/> pic1.jpeg	AQAAAHhkNTbQXvEiBekVhkj8kt86yXS
<input checked="" type="checkbox"/> pic2.jpeg	AQAAAKJ0VusTB-dG5CysC7W0NNz97FNT
<input checked="" type="checkbox"/> text1.pages	AQAAAALfxWtetVQHTDqEVxSB0NEyoeOK
<input checked="" type="checkbox"/> text2.pages	AQAAAC--hw12QEM9Uh8ODFDZVS75t4Sy
<input checked="" type="checkbox"/> text3.pages	AQAAAIIDNikpqI_DvmFgY8a5r-GPCJt4s

导出URL列表

是否导出以下5个文件的URL列表?

导出设置

传输协议

HTTPS HTTP

过期时间

300 秒

自有域名

不使用自有域名

确认资源

文件名	文件大小	MIME-Type
pic1.jpeg	83.28 KB	image/jpeg ×
pic2.jpeg	89 KB	image/jpeg ×
text1.pages	84.66 KB	application/octet-stream ×
text2.pages	83.22 KB	application/octet-stream ×
text3.pages	83.85 KB	application/octet-stream ×

取消 确定

设置HTTP头

文件(Object)的 HTTP 头部(元数据头部)是服务器以 HTTP 协议传送 HTML 资料到浏览器前所送出的字符串。通过修改 HTTP 头部(元数据头部),可以改变页面的响应形式,或者传达

配置信息,例如修改缓存时间。修改对象的 HTTP 头部不会修改对象本身。对象元数据包括有两种元数据:HTTP标准属性(HTTP Header)和用户定义元数据(User Meta)。

HTTP标准属性 (HTTP Header) 为上传至存储空间中的每个文件保留如下HTTP标准属性。

名称	描述	示例
MIME-Type	文件的 MIME 信息	image/jpeg
Content-Encoding	文件的编码格式	gzip
Content-Language	声明Object内容使用的语言。	zh-CN
Content-Disposition	MIME 协议的扩展	attachment;filename="fname.ext"
Cache-Control	文件的缓存机制	no-cache;max-age=200
Expires	用来控制缓存的失效日期	Wed, 21 Oct 2015 07:28:00 GMT

用户自定义元数据 (User Meta) 用户可以在上传文件(Object)时,为文件添加自定义元数据(User Meta),用于标识文件的用途或属性等。

- 一个Object可以有多个自定义元数据,但所有的自定义元数据总大小不能超过8 KB。
- 自定义元数据是一组键值对,元数据名称必须以X-Ufile-Meta-开头。
- 调用GetFile或者HeadFile接口时,将在HTTP头部返回自定义元数据。

直接单击“操作”下的“...”选择“设置HTTP头”即可完成修改。

< 单地域空间管理, -8

- 概览
- 基础设置
- 文件管理
- 生命周期
- 域名管理
- 镜像回源
- 跨区域复制
- 跨域设置
- 数据处理

上传文件 创建目录 添加解压规则 ...

输入文件前缀查找

文件名	ETag	文件大小	存储类型	MIME-Type	更新时间	操作
6.pages	AQAAAG	84.72 KB	低频存储	application/x-iwork-pages-sff...	2023-12-12 17:58:37	获取URL 下载 ...
text3.pages	AQAAADt	83.85 KB	归档存储	application/x-iwork-pages-sff...	2023-12-12 17:58:48	获取URL 下载 修改存储类型 设置HTTP头
text4.pages	AQAAA	84.7 KB	标准存储	application/x-iwork-pages-sff...	2023-10-27 15:07:52	获取URL 下载 删除

修改存储类型
设置HTTP头

单地域空间管理 / [模糊]

概览 · 基础设置 · 文件管理 · 生命周期 · 域名管理 · 镜像回源 · 跨区域复制

上传文件 创建目录 添加解压规则 ...

文件名	Etag	文件大小
6.pages	AQAAA[模糊]	84.72 KB
8.pages	AQAAAC[模糊]	84.72 KB
9.pages	AQAAA[模糊]	84.7 KB

设置HTTP头

HTTP标准属性

参数	值
MIME-Type	application/x-work-pages-sffpages
Content-Encoding	请输入文本
Content-Language	请输入文本
Content-Disposition	请输入文本
Cache-Control	请输入文本
Expires	请输入文本

自定义元数据

参数	值
X-File-Meta- 请输入	请输入文本
+	

取消 确定

生命周期

用户可以通过创建存储空间(Bucket)生命周期规则,实现存储空间内所有文件或特定前缀文件的生命周期管理(可以配置一个或者多个规则),设置生命周期规则转换存储类型或删除文件操作。

- 删除文件:支持设置指定的文件在 N 天后被自动删除。
- 转低频存储:可以设置指定的文件在 N 天后,自动转换成低频存储类型。
- 转归档存储:可以设置指定的文件在 N 天后,自动转换成归档存储类型。

使用说明:存储类型转换和删除文件操作的开始时间是如何计算的呢?

US3将对象上传时间与设置天数相加,得到的时间入到后一天的午夜(CST: 中国标准时间),从而得到对象操作开始时间。例如对象在 2015年1月1日 上午 10:00 CST 上传,设置在 7 天后被删除,则会在 2015年1月9日 00:00 CST 之后当天内删除文件。

注: 设置生命周期规则后,当天晚上会执行生命周期操作,删除操作不可恢复,请谨慎设置。如一个文件为用户在 **31** 天前上传,用户配置了生命周期规则为删除 **30** 天前的文件,则 **US3** 服务在生命周期规则被设置的当晚会自动执行删除该文件的操作。

使用场景

通过生命周期规则,用户可以更高效的管理存储的数据,节省大量人力及存储成本。用户可以通过设置匹配特定前缀的规则,定期将非热门数据转换为低频访问或归档存储,将不再需要访问的数据删除。例如:

1. 监控视频上传,根据相关条例,监控保存时间一般规定如下:一般小区的是7天;一般娱乐场所是 15 天;金融行业是 26 天;银行 3 个月到半年。可以通过设置生命周期规则,上传的文件,会在匹配规则命中后,根据上传时间,到达指定天数后自动过期删除,从而减少存储量,降低成本。

2. 电子公文类数据, 根据相关条例, 需要长期保存达 10 年, 超过 1 年的电子公文调阅次数减少, 超过 3 年的电子公文通常不再被调阅。可以通过设置生命周期规则, 上传的文件, 会在匹配规则命中后, 根据上传时间, 到达指定天数后自动转低频、归档存储, 从而降低存储的单位成本。

注意事项

1. 存储空间的生命周期功能一旦开启, 对存储空间内所有符合规则范围的文件生效 (包括开启生命周期功能前上传的文件), 并于后一天的 00:00 触发生命周期规则, 请确认无误后再保存生命周期规则。
2. 删除文件的操作是不可逆的, 请根据您的需求合理设置生命周期规则, 避免重要数据丢失。

管理生命周期

1. 选择对应存储空间, 在右侧操作中单击生命周期按钮。

对象存储 US3

切换旧版

单地域空间管理

全球化空间管理

统计报表

管理工具及SDK

令牌管理

创建存储空间

更改业务组

删除存储空间

已选择 1 取消选择

<input type="checkbox"/>	存储空间域名	地域	业务组	空间类型	数据分析	创建时间	操作
<input type="checkbox"/>	...	北京	未分组	私有空间		2019-03-19	详情 文件管理 域名管理 ...
<input checked="" type="checkbox"/>	...	北京	byh-test	公开空间		2019-06-14	详情 文件管理 域名管理 ...
<input type="checkbox"/>	...	香港	未分组	公开空间		2019-06-25	详情 文件管理 域名管理 ...
<input type="checkbox"/>	...	法兰克福	hongshi-test	私有空间		2019-07-31	详情 文件管理 域名管理 ...
<input type="checkbox"/>	...	新加坡	未分组	公开空间		2019-10-22	详情 文件管理 域名管理 ...
<input type="checkbox"/>	...	首尔	未分组	公开空间		2019-12-09	详情 文件管理 域名管理 ...
<input type="checkbox"/>	...	拉各斯	hongshi-test	私有空间		2019-12-31	详情 文件管理 域名管理 ...
<input type="checkbox"/>	...	首尔	未分组	私有空间		2020-06-25	详情 文件管理 域名管理 ...
<input type="checkbox"/>	...	北京	import	公开空间		2020-10-19	详情 文件管理 域名管理 ...
<input type="checkbox"/>	...	北京	未分组	私有空间		2020-10-22	详情 文件管理 域名管理 ...

生命周期

镜像回源

跨域设置

日志管理

修改空间类型

更改业务组

删除存储空间

2. 单击添加规则按钮。



单地域空间管理 / []

· 概览 · 基础设置 · 文件管理 · **生命周期** · 域名管理 · 镜像回源 · 跨区域复制 · 跨域设置 · 数据处理

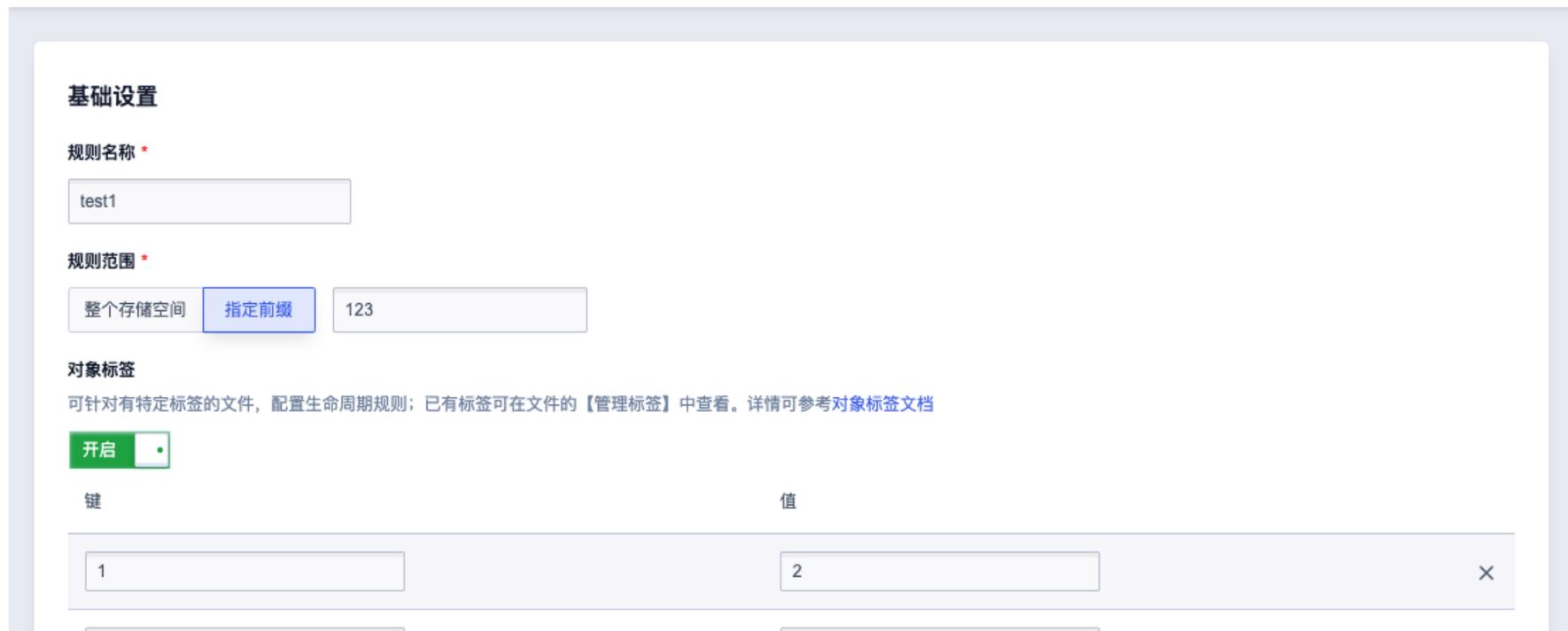
添加规则 删除

规则名称	规则范围	规则内容	状态	操作
[]	整个存储空间	7天后删除文件	● 已启用	编辑 禁用 删除
[]	指定前缀: test/	30天后删除文件	● 已启用	编辑 禁用 删除

< 1 > 10 条/页 /1

3. 设置生命周期规则界面。

添加生命周期规则



基础设置

规则名称 *

test1

规则范围 *

整个存储空间 指定前缀 123

对象标签

可针对有特定标签的文件，配置生命周期规则；已有标签可在文件的【管理标签】中查看。详情可参考[对象标签文档](#)

开启

键	值
1	2

×

+

规则配置

- 天后转为低频存储
- 天后转为归档存储
- 天后删除文件

字段	说明
规则名称	用户可以设置一个规则名称,记录自己创建的规则。
规则范围	指定规则的生效范围,选择整个存储空间时,生命周期规则对整个存储空间生效; 选择指定前缀时,仅对该存储空间下的指定前缀生效。
对象标签	生命周期规则仅针对拥有指定标签生效。对象前缀和对象标签、对象标签和对象标签之间都为“与”的关系(即所有条件需同时满足)。对象标签详情请参见对象标签。
规则配置	选择需要执行的生命周期规则和规则的触发时间,最小为 7 天,具体计算方式请参考备注说明。

注: 指定前缀时,无需在最前面添加额外的"/"。

4. 生命周期规则添加完成后默认生效,生效状态具体可见状态栏,用户可以通过操作栏的其他按钮,进行相应的修改以及关闭操作。



注: 单击禁用或者删除按钮以后, 状态修改在第二天生效, 不影响当天 00:00 已触发但未执行完成的任务继续执行。

计费说明

使用生命周期规则时, 会产生如下费用, 价格详情请参见产品价格:

1. 请求费用:

- 转换文件存储类型产生的请求费用, 即从标准存储类型转换为低频访问存储类型时, 按标准存储类型的请求次数收取请求费用。从低频存储类型转换为归档存储类型时, 按低频存储类型的请求次数收取请求费用。
- 删除文件产生的请求费用, 按文件被删除时的存储类型收取请求费用。

2. 存储费用: 根据用户存储的文件类型、大小和时长收取一定的存储费用。

- 文件最小计量单位限制: 单个文件小于64KB, 按照64KB计算; 大于或等于64KB, 按照实际大小计算。
- 不足规定时长存储费用:

- 通过生命周期将文件存储类型转换为低频访问,且在不足规定时长前删除文件:低频访问类型最低存储时间(30天)以文件存储在US3的时间开始计算。例如标准类型的文件在其创建10天后,通过生命周期将其转换为低频访问,过了2天后将其删除,则会产生18天的低频访问不足规定时长存储费用。
- 通过生命周期将文件存储类型转换为归档访问,且在不足规定时长前删除文件:归档访问类型最低存储时间(60天)以文件存储在US3的时间开始计算。例如标准类型的文件在其创建10天后,通过生命周期将其转换为归档访问,过了5天后将其删除,则会产生45天的低频访问不足规定时长存储费用。

3. 数据取回费用:

- 通过生命周期将存储类型从低频转换为归档时,会产生低频取回费用。例如某个低频类型的文件降冷转换为归档,则US3需要先获取原低频存储文件,从而产生低频取回费用。并且成功降冷后,原低频文件被删除也产生删除请求费用。

4. 对象标签费用:

- US3对象存储按照用户存储空间(Bucket)内对象标签的数量,按天计算对象存储标签费用,单价是0.00189 元/每万个标签。

备注

1. 低频存储和归档存储类型都有最短存储期限,早于最短存储期限删除、修改、覆盖文件,需要补足未满足最短存储期限的剩余天数的存储费用。
2. 生命周期规则的自动操作触发时间是如何计算的呢?US3 将文件上传时间与设置天数相加,计算得到时间的后一天的00:00 (CST: 中国标准时间)即为自动操作开始时间。例如文件在2015年1月1日 上午 10:00 CST 上传,设置在 7 天后被删除,那么会在 2015年1月9日 00:00 CST 后触发自动删除文件操作。
3. 当前生命周期规则里的对象标签仅越南、巴西支持,其他地域如有需要请联系技术支持。

静态网站托管

概念

静态网站是指网站的所有网站资源都由静态内容构成,如HTML、JavaScript、CSS、图片等文件。您可以通过控制台对已经绑定的自定义域名的US3存储桶配置静态网站托管策略

使用说明

使用该功能必须先创建的自定义域名。如使用若源站是us3域名,则需注意:

- cdn配置跟随源站域名:源站不需要自定义域名,但是不会识别成我们的cdn加速,一般源站的域名也会有服务。
- cdn配置跟随加速域名:需要源站自定义域名为cdn加速域名,这样服务才会正常

根据配置界面进行相应参数设置,其中必填项:

- 默认首页:该首页是您通过自定义域名访问US3存储桶时,返回US3存储桶首页。如果还开通了子目录首页,则子目录下也应该存在该文件,当然您可以根据自己的路径来自定义该文件的内容。
- 默认404页(但开通子项目首页,且文件404规则为NoSuchKey时,默认404不填)是当浏览器访问US3存储桶中的文件不存在(404)时,返回的错误页面。

US3存储桶必须为公共空间,即该存储桶必须是公共读私有写。另外通过US3默认的域名进行访问,会将静态网站以文件的形式进行下载。只有通过绑定US3存储桶的自定义域名来进行访问才会在浏览器渲染显示。

详细工作机制,参考举例说明。

举例说明

为Bucket开启静态网站托管后,您需要将与默认首页名称相同的文件(例如index.html)上传至目标Bucket,如果Bucket中包含了目录结构prefix/,则目录层级下也必须包含index.html文件。此外,您还需要将与默认404页名称相同的文件(例如error.html)上传至目标Bucket。Bucket的文件结构如下所示:

```
Bucket
├─ index.html
├─ error.html
├─ us3.png
├─ prefix/
├─ index.html
```

如果该Bucket绑定了自定义域名example.com,且配置的静态网站默认首页为index.html,默认404页为error.html。则通过自定义域名访问静态网站时,根据是否开通了子目录首页,访问规则如下:

- 未开通子目录首页
 - 当您访问<https://example.com/>和<https://example.com/prefix/>时,US3会返回<https://example.com/index.html>。
 - 当您访问<https://example.com/us3.png>时,正常获取us3.png文件。
 - 当您访问<https://example.com/helloworld>时,因helloworld不存在,US3会返回<https://example.com/error.html>。
- 已开通子目录首页
 - 当您访问<https://example.com/>时,US3会返回<https://example.com/index.html>。
 - 当您访问<https://example.com/prefix/>时,US3会返回<https://example.com/prefix/index.html>。
 - 当您访问<https://example.com/us3.png>时,正常获取us3.png文件。
 - 当您访问<https://example.com/helloworld>时,因helloworld不存在,US3会根据您设置的文件404规则返回对应信息:

- 如果文件404规则设置为Redirect (默认值), US3会继续检查helloworld/index.html是否存在。
 - 如果文件存在则返回302, 并将访问请求重定向为<https://example.com/helloworld/index.html>。
 - 如果文件不存在则返回404。
- 如果文件404规则设置为NoSuchKey, 则直接返回404。
- 如果文件404规则设置为Index, US3会继续检查helloworld/index.html是否存在。
 - 如果文件存在则返回200, 并直接返回文件内容。
 - 如果文件不存在则返回404。

域名管理

文件(Object)上传至存储空间(Bucket)后,US3会自动生成文件URL,用户可以直接通过文件URL访问该文件。若用户希望通过自己的域名或CDN访问US3上的文件,需要将自己的域名或CDN域名绑定至文件所在的存储空间。

- 实际生产环境推荐您绑定自定义域名创建 CDN 加速。

绑定自定义域名

前提条件:

- 确认自定义域名已经完成备案(若未进行备案,域名会被管局停止解析,导致业务不可用),备案相关请访问 [备案专区](#);
- 并已在DNS服务商处配置**CNAME**至US3的存储空间域名。

1. 登陆对象存储控制台。
2. 单击单地域空间管理,然后单击目标Bucket。
3. 在顶部导航栏,选择域名管理,单击绑定自定义域名。

绑定自定义域名 ✕

! 绑定的自定义域名请确保已备案，并已在DNS服务商处配置CNAME至US3的存储空间域名。

存储空间域名 fileos.com

自定义域名 *

CDN加速 ? 否 是

◦ CDN加速:CDN加速选项在UCDN产品页加速和管理,点击确定后将前往UCDN产品页面继续完成操作,跳转到 UCDN 产品页面继续完成基础配置和高级配置。UCDN加速相关请参考UCDN创建加速域名。

4. 在域名列表找到需要证书托管的域名,单击右侧证书托管,证书来源选择如下:

◦ 从**USSL**导入:选择已托管在USSL证书管理平台的证书

证书托管 ✕

证书来源
当前仅支持不加密的PEM格式证书,且会对证书有效性进行检查。

从USSL导入 本地导入 手动输入

证书名称 *

请输入证书名称 usecure_ussl-1 

取消 **确定**

- 从本地导入:按照公钥(public.crt)、私钥(private.key)格式上传文件

证书托管 ⓧ

证书来源

当前仅支持不加密的PEM格式证书,且会对证书有效性进行检查。

证书名称 *

授权证书 *

◦ 手动输入:按照公钥、私钥pem格式输入

证书托管 ⓧ

证书来源

当前仅支持不加密的PEM格式证书,且会对证书有效性进行检查。

从US3导入

本地导入

手动输入

证书名称 *

请输入证书名称

公钥 *

请按照PEM编码格式填写

私钥 *

请按照PEM编码格式填写

取消

确定

5. 通过自定义域名访问US3文件 绑定自定义域名后可直接通过HTTP协议访问文件,文件URL的格式为http://YourDomainName/ObjectName。如果用户需要通过HTTPS协议访问

文件,需完成步骤4的证书托管。

举例:目标存储空间examplebucket中存放了exampleobject.jpg的文件,自定义域名为www.demo.com ,且已完成证书托管,此时您可以使用https://www.demo.com/exampleobject.jpg 访问目标文件。

解绑自定义域名

当用户的自定义域名不再使用时,可以手动解除域名绑定。

1. 如果目标存储空间绑定的自定义域名启用了CDN加速服务,用户需要修改CDN加速服务的源站信息,使自定义域名不再指向US3域名。具体操作,请参见停用UCDN服务。
2. 在顶部导航栏,选择域名管理,在域名列表找到需要解绑的域名,单击右侧解绑,弹窗确定后即可完成。

单地域空间管理 /

概览 基础设置 文件管理 生命周期 域名管理 镜像回源 跨区域复制 跨域设置 数据处理

测试域名 ②

CDN加速状态: 已关闭 开启

自定义域名

绑定自定义域名

自定义域名	CDN加速	加速类型	https证书	加速区域	操作
www. .com	否	—	—	—	解绑 证书托管

< 1 > 10条/页 /1

解绑自定义域名

是否解绑当前自定义域名?

www. .com

取消 确定

镜像回源

US3 提供镜像回源功能,配置回源规则后,当您向 US3 存储空间请求的数据不存在时,可以通过回源设置,对于获取数据的请求以多种方式进行回源读取,满足您对于数据热迁移、特定请求重定向的需求。

设置回源规则后,可对 US3 的每条 GET 或 HEAD 请求的 URL 按设定的规则进行匹配,然后按照设定的规则进行回源。最多配置 5 条规则,顺序匹配,直到匹配到有效规则。回源类型为镜像方式。

- 镜像方式:如果配置了回源规则,则当对一个不存在的文件进行 GET 或 HEAD 操作时,会向源站地址请求这个文件,返回给用户,并同时写入到 US3。

- 镜像方式下,如果请求中携带 Range 头部字段,最终会将完整的回源文件写入 US3 而非仅写入 Range 所表明的部分数据。
- 图片处理请求不会触发镜像回源。

使用场景

您的大部分源文件存储在 US3 存储空间,少量文件在客户源站,但仍会有少量文件继续写到源站,可通过镜像回源功能直接到源站回源获取文件,从而实现业务的平滑迁移。

设置回源规则

用户指定文件前缀和源站地址,当 US3 没有文件时,US3 自动根据文件前缀到指定源站抓取文件保存,并返回给用户。

- 选择对应空间, 在右侧操作中选择镜像回源按钮。

对象存储 US3

切换旧版

单地域空间管理 | 全球化空间管理 | 统计报表 | 管理工具及SDK | 令牌管理

创建存储空间 | 更改业务组 | 删除存储空间

已选择 1 取消选择

存储空间域名	地域	业务组	空间类型	数据分析	创建时间	操作
[redacted].ufileos.com	北京	未分组	私有空间	[icon]	2019-03-19	详情 文件管理 域名管理 ...
[redacted].ufileos.com	北京	byh-test	公开空间	[icon]	2019-06-14	详情 文件管理 域名管理 ...
[redacted].ufileos.com	香港	未分组	公开空间	[icon]	2019-06-25	详情 文件管理 域名管理 ...
[redacted].fra.ufileos.com	法兰克福	hongshi-test	私有空间	[icon]	2019-07-31	详情 文件管理 域名管理 ...
[redacted].ufileos.com	新加坡	未分组	公开空间	[icon]	2019-10-22	详情 文件管理 域名管理 ...
[redacted].ufileos.com	首尔	未分组	公开空间	[icon]	2019-12-09	详情 文件管理 域名管理 ...
[redacted].ufileos.com	拉各斯	hongshi-test	私有空间	[icon]	2019-12-31	详情 文件管理 域名管理 ...
[redacted].ufileos.com	首尔	未分组	私有空间	[icon]	2020-06-25	详情 文件管理 域名管理 ...
[redacted].com	北京	import	公开空间	[icon]	2020-10-19	详情 文件管理 域名管理 ...
[redacted].ufileos.com	北京	未分组	私有空间	[icon]	2020-10-22	详情 文件管理 域名管理 ...

生命周期
镜像回源
跨域设置
日志管理
修改空间类型
更改业务组
删除存储空间

- 点击添加回源规则按钮。

< 单地域空间管理 / testnmd

概览

文件管理

域名管理

镜像回源

添加回源规则

什么是镜像回源?

文件前缀

回源地址

操作

aa

http://1test2.oss-cn-beijing.aliyuncs.com

编辑

删除

http://testcda1.hk.ufileos.com

编辑

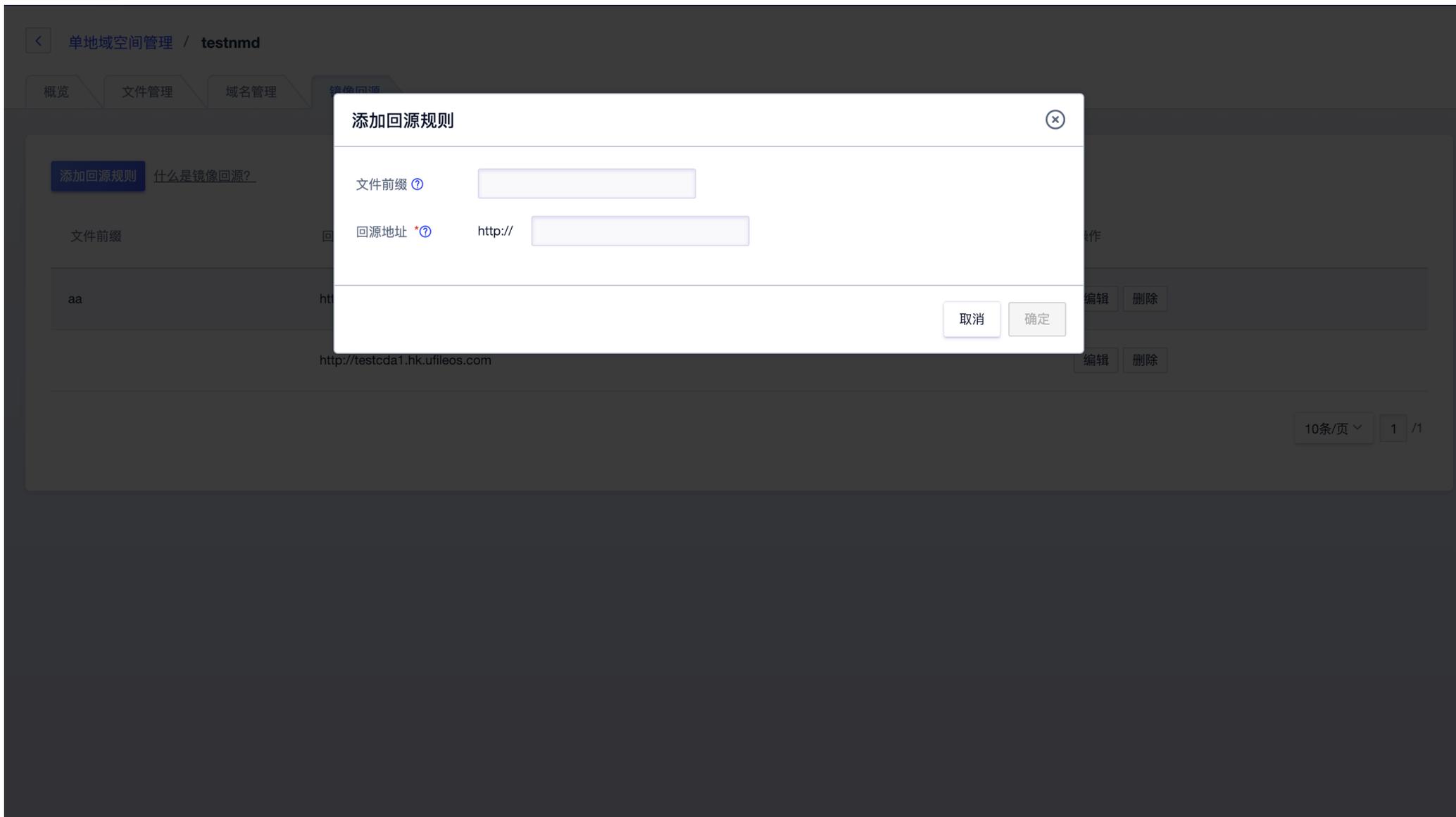
删除

10条/页

1

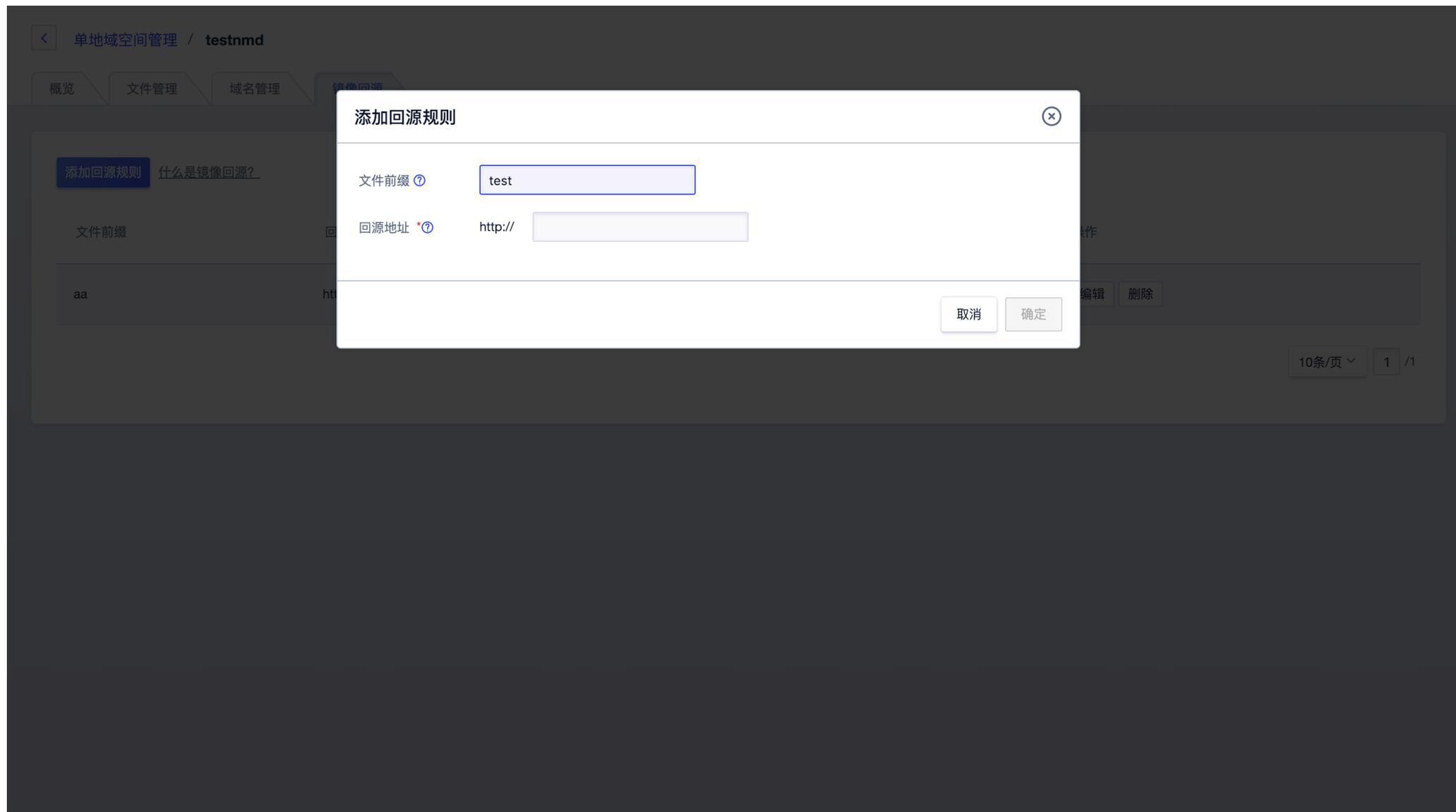
/1

- 添加回源规则界面。



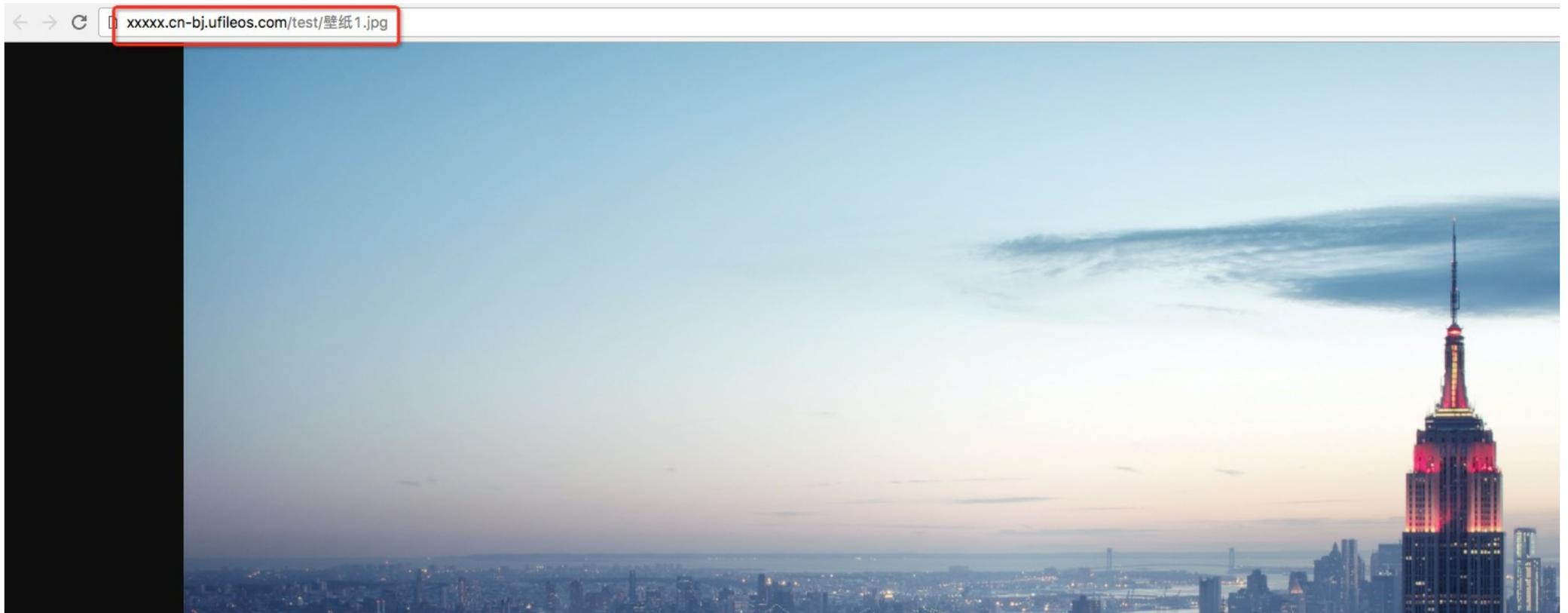
- 填写文件前缀和回源地址,文件前缀为指定源站真实路径,若为根目录,则文件前缀为空。

例:通过 US3 访问其他源站 test 文件夹下的文件,设置回源规则,文件前缀为 test,回源地址为源站地址。



注:回源地址只支持 HTTP 访问

- 通过US3访问其他源站该文件夹下的文件。



- US3也自动保存了该文件。

< 单地域空间管理 / testnmd

概览 文件管理 域名管理 镜像回源

上传文件 修改MIME-Type 删除

🔍 ⚙️ ↻️ ?

<input type="checkbox"/>	文件名 1l	文件大小	MIME-Type	更新时间	操作
<input type="checkbox"/>	1m.jpg1	810 B	application/octet-stream	2018-08-02 20:07:38	获取地址 下载 删除
<input type="checkbox"/>	7//timg (3).jpeg	47.77 KB	image/jpeg	2018-08-01 21:44:05	获取地址 下载 删除
<input type="checkbox"/>	7/timg.jpeg	68.85 KB	image/jpeg	2018-08-01 21:44:26	获取地址 下载 删除
<input type="checkbox"/>	aa/mirror-test.jpeg	47.77 KB	image/jpeg	2018-08-22 11:36:34	获取地址 下载 删除
<input type="checkbox"/>	index.html	76 B	text/html; charset=utf-8	2018-08-22 10:19:26	获取地址 下载 删除
<input type="checkbox"/>	test/mirror-test.jpeg	47.77 KB	image/jpeg	2018-12-29 13:46:28	获取地址 下载 删除
<input type="checkbox"/>	test/壁纸1.jpg	55.6 KB	image/jpeg	2018-12-29 13:48:50	获取地址 下载 删除
<input type="checkbox"/>	testWithPolicy	91 B	application/octet-stream	2018-08-19 15:33:56	获取地址 下载 删除
<input type="checkbox"/>	timg (3).jpeg	47.77 KB	image/jpeg	2018-08-13 13:31:07	获取地址 下载 删除

备注

1. 当前每个单地域存储空间(Bucket)下可创建 5 条回源规则,前缀唯一不可重复
2. 全球化空间管理暂不支持设置回源规则
3. 子账户需要开启 API 密钥权限(常见问题),若子账号未开启 API 密钥,US3 则不能将回源文件上传到用户的 Bucket
4. 不支持在图片处理请求中触发镜像回源,即不支持图片处理的目标文件为源站中的文件(用户可先发起镜像方式下的回源请求,待回源成功后,再发起图片处理请求)
5. 不支持源站为私有 US3 或其他第三方私有 Bucket
6. 不支持源站地址使用 US3 内网域名
7. 不支持 MD5 校验
8. 若源站返回 3xx 重定向响应,US3 会向新的源站地址重新发起请求,最大递归次数为 10 次

跨区域复制

US3 提供跨区域复制功能,您可以选择两个不同地域的存储空间(Bucket)进行绑定,进行文件的同步。

跨区域复制是跨不同 US3 数据中心(地域)的存储空间自动、异步复制文件,它会将文件的创建、更新和删除等操作从源存储空间复制到不同区域的目标存储空间。

跨区域复制功能提供了存储空间跨地域容灾能力,同时能够满足您数据复制的需求。目标存储空间的文件是源存储空间中文件的完整副本,它们具有相同的文件名、内容。

使用场景

- **合规性要求:**您的数据在合规性上必须要求数据跨一定距离保存一份副本。通过跨区域复制功能,可以在远距离的 US3 数据中心之间复制数据以满足这些合规性要求。
- **最大限度减少延迟:**您同时在两个地理位置存在业务需求,为了降低访问文件时的延迟,可以在地理位置与用户较近的 US3 数据中心中维护文件副本。
- **数据备份与容灾:**您对数据的安全性和可用性有极高的要求,以备 US3 数据中心因特大灾难,如地震、海啸等不可抗力导致损毁时,还能启用另一个 US3 数据中心的备份数据。
- **数据复制:**由于业务原因,需要将您的数据从 US3 的一个数据中心迁移到另一个数据中心。
- **操作原因:**您同时在两个不同数据中心具有计算资源,您可能通过跨区域复制实现在这些区域的数据中心上同时维护 US3 的文件。

注意事项

1. 对于处于同步状态的两个存储空间,由于您可以同时操作这两个存储空间,因此从存储空间复制过去的文件可能会覆盖目标存储空间中的同名文件,覆盖文件的操作是不可逆的,请根据您的需求合理设置跨区域复制规则,避免重要数据丢失。

2. 由于跨区域复制采用异步复制,数据复制到目标 Bucket 需要一定的时间,通常几分钟到几小时不等,取决于数据的大小。
3. 镜像回源与跨区域复制两个功能不能同时开启。
4. 如存储桶配置了生命周期规则,则生命周期配置仅对当前存储空间生效,开启跨区域复制的存储空间之间不会同步删除文件。
5. 跨区域复制相关费用:
 - 流量费用:使用跨区域复制功能将源存储空间的数据同步复制到目标存储空间时所产生的源端的外网流出费用。流量费用,请参见产品价格。
 - 请求费用:US3会计算请求次数并收取请求费用,包含源端的拉列表和下载请求以及目标端的上传请求。请求费用,请参见产品价格。

设置跨区域复制规则

选中对应空间,进入管理界面,选择跨区域复制功能。

< 单地域空间管理 /

概览 文件管理 生命周期 域名管理 镜像回源 跨区域复制 跨域设置

添加规则 [什么是跨区域复制?](#)

目标地域	目标存储空间	历史文件同步	创建时间	状态	操作
北京		否	2019-12-11	● 已启用	禁用 删除

点击添加规则按钮,弹出添加跨区域复制规则界面。

管理 镜像回源 跨区域复制 跨域设置

添加跨区域复制规则

目标地域

目标存储空间* 

历史文件是否同步?

1. 目标地域:用户可选择复制目标的存储空间所在地域。数据同步的两个存储空间必须分属两个地域,同地域的存储空间之间不能进行数据同步。
2. 目标存储空间:选择开启数据同步的目标存储空间。跨区域复制的两个存储空间都不能同时与其他任何存储空间存在互相同步的关系。例如已设置存储空间 A 的数据同步至存储空间 B,则 A 和 B 都不能和其他任何存储空间再建立数据同步关系。

跨区域复制规则添加完成后默认生效,生效状态具体可见状态栏,您可以通过操作栏下的其他按钮,进行相应的启用、禁用以及删除操作。

备注

1. 跨区域复制功能支持数据实时同步,对于数据的增加、删除、修改能够实时监控同步到目标地域的存储空间。对于 2MB 文件,能够做到分钟级别信息同步,保证两边数据的最终一致,不保证中间状态的数据一致。
2. 当前(迪拜 孟买 圣保罗 胡志明 拉各斯)暂不支持跨域复制。如有需求,请联系技术支持。
3. 跨区域复制功能目前不支持国内和国外地域之间的复制。
4. 归档类型文件不支持复制。

跨域访问

跨域访问(CORS)是指当用户从一个域名的网页去请求另一个域名的资源的操作。通常浏览器会出于安全考虑对这种跨域访问进行限制,为了解决这个问题,US3 提供跨域访问策略管理功能,便于您对跨域访问的权限进行限制。

当 US3 收到跨域请求,会读取存储空间配置过的跨域访问策略列表,根据来源(Origin)逐一匹配每一条策略,第一个符合条件的策略信息会添加到 CORS 相关的 Header 中返回给用户。

设置跨域访问策略

选中对应空间,在右侧操作中选择跨域设置按钮。

对象存储 US3

切换旧版

单地域空间管理 全球化空间管理 统计报表 管理工具及SDK 令牌管理

创建存储空间 更改业务组 删除存储空间

已选择 1 取消选择

存储空间域名	地域	业务组	空间类型	数据分析	创建时间	操作
...	北京	未分组	私有空间	📈	2019-03-19	详情 文件管理 域名管理 ...
...	北京	byh-test	公开空间	📈	2019-06-14	详情 文件管理 域名管理 ...
...	香港	未分组	公开空间	📈	2019-06-25	详情 文件管理 域名管理 ...
...	法兰克福	hongshi-test	私有空间	📈	2019-07-31	详情 文件管理 域名管理 ...
...	新加坡	未分组	公开空间	📈	2019-10-22	详情 文件管理 域名管理 ...
...	首尔	未分组	公开空间	📈	2019-12-09	详情 文件管理 域名管理 ...
...	拉各斯	hongshi-test	私有空间	📈	2019-12-31	详情 文件管理 域名管理 ...
...	首尔	未分组	私有空间	📈	2020-06-25	详情 文件管理 域名管理 ...
...	北京	import	公开空间	📈	2020-10-19	详情 文件管理 域名管理 ...
...	北京	未分组	私有空间	📈	2020-10-22	详情 文件管理 域名管理 ...

生命周期
镜像回源
跨域设置
日志管理
修改空间类型
更改业务组
删除存储空间

点击新建策略按钮,弹出新建跨域访问策略界面。



1. 来源(Origin):用户可填写允许跨域请求的来源,可以设置多个,每行一个,每行最多一个通配符(*)。
2. 允许Methods:用户可选择允许的跨域请求方法。
3. Allow-Headers:用户可填写允许的跨域请求Header,可以设置多个,每行一个,每行最多一个通配符(*)。
4. Expose-Headers:用户可填写允许从应用程序中访问的响应头,可以设置多个,每行一个,不允许使用通配符(*)。

跨域访问策略添加完成后默认生效,您可以通过操作栏下的其他按钮,进行相应的编辑及删除操作。

[< 单地域空间管理 /](#)

概览

文件管理

生命周期

域名管理

镜像回源

跨区域复制

跨域设置

新建策略

来源Origin	允许Methods	Allow-Headers	Expose-Headers	操作
example.com	GET,PUT	Origin Content-type		编辑 删除

10条/页 1 /1

备注

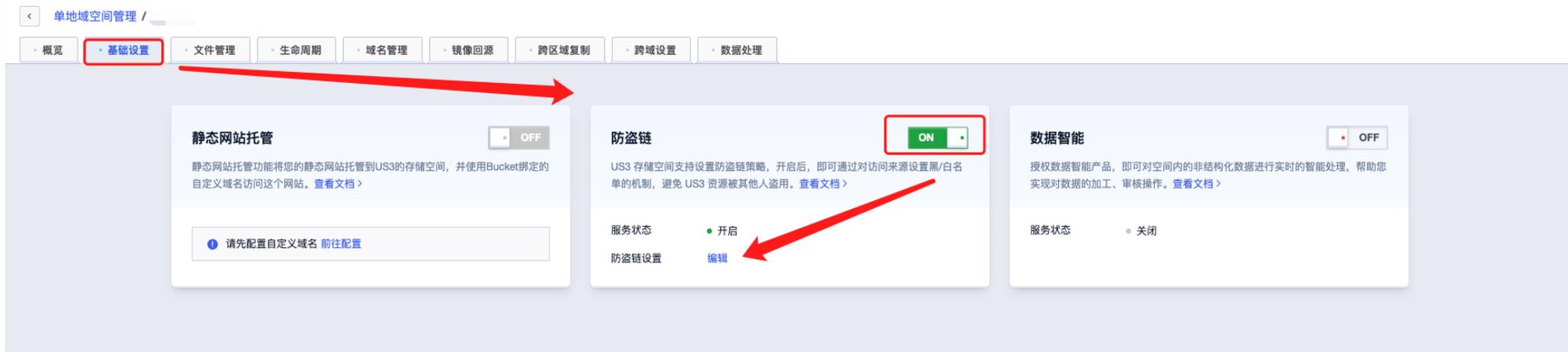
1. 每个存储空间最多配置 10 条跨域访问策略。

防盗链

US3 存储空间支持设置防盗链策略。开启后,您可通过对访问来源设置黑、白名单的机制,避免 US3 资源被其他人盗用,或被盗刷公网流量,造成不必要的损失。

设置防盗链

1. 登陆US3对象存储控制台。
2. 单击单地域空间管理,然后单击目标Bucket。
3. 在顶部导航栏,选择基础设置,打开防盗链开关后,单击编辑进入可编辑状态:



4. 根据实际需求设置,配置项说明如下:

防盗链设置



白名单Referer *

请输入域名或者表示域名URL的正则表达式，使用回车符或者英文逗号分隔多个Refer规则

http://www.ucloud.cn

黑名单Referer

请输入域名或者表示域名URL的正则表达式，使用回车符或者英文逗号分隔多个Refer规则

请输入黑名单Referer

允许空Referer

如果不允许空 Referer，则只有 HTTP 或 HTTPS header 中包含 Referer 字段的请求才能访问 US3 对象存储资源。

否

取消

确定

- **白名单Referer:** 允许名单内的域名访问存储空间的默认访问地址,若名单外的域名访问存储桶的默认访问地址,则返回403。
- **黑名单Referer:** 拒绝名单内的域名访问存储空间的默认访问地址,若名单内的域名访问存储桶的默认访问地址,则返回403。

请输入域名或者表示域名URL的正则表达式,使用回车符或者英文逗号分隔多个Refer规则。

- **允许空Referer:**
 - **是:**表示允许请求中不带Referer或Referer为空。

- 否:表示请求中必须携带Referer且Referer值不能为空。设置不允许空Referer后,则只有 HTTP 或 HTTPS header 中包含 Referer 字段的请求才能访问 US3 对象存储资源。

6. 单击确定完成防盗链设置。

统计报表

通过 US3 提供的统计报表功能,您可以快速查看 US3 存储空间的资源使用情况。您可在 US3 控制台点击顶部页签的“统计报表”,进入统计报表页面。

对象存储 US3

切换旧版

单地域空间管理

全球化空间管理

统计报表

管理工具及SDK

令牌管理

北京

香港

广州

上海二

洛杉矶

新加坡

雅加达

拉各斯

圣保罗

迪拜

胡志明市

台北

孟买

华盛顿

法兰克福

首尔

所有空间

最近7天

2020-10-16 — 2020-10-23



存储容量-标准 (总)

906.62 GB

存储容量-低频 (总)

0.02 GB

存储容量-归档 (总)

29.13 GB

CDN回源流量 (总)

0.00 GB

请求次数 (总)

11.2066 万

下载流量-闲时 (总)

0.00 GB

下载流量-忙时 (总)

2.22 GB

点击查看剩余配额 >

日期	存储容量-标准	存储容量-低频	存储容量-归档	下载流量	下载流量-闲时	下载流量-忙时	CDN回源流量	数据取回-低频	数据取回-归档	请求次数
2020-10-16	917.35GB	0.02GB	1.71GB	0.00GB	0.00GB	0.00GB	0.00GB	0.00GB	0.15GB	5.0146万次
2020-10-17	917.31GB	0.02GB	9.97GB	0.00GB	0.00GB	0.00GB	0.00GB	0.00GB	0.00GB	0.0000万次
2020-10-18	917.31GB	0.02GB	9.97GB	0.00GB	0.00GB	0.00GB	0.00GB	0.00GB	0.00GB	0.0000万次
2020-10-19	917.62GB	0.02GB	9.97GB	0.00GB	0.00GB	0.00GB	0.00GB	0.00GB	0.00GB	0.0265万次
2020-10-20	914.88GB	0.02GB	20.78GB	0.00GB	0.00GB	0.41GB	0.00GB	0.00GB	0.00GB	5.3176万次
2020-10-21	907.63GB	0.02GB	29.79GB	0.00GB	0.00GB	0.00GB	0.00GB	0.00GB	0.00GB	0.1191万次
2020-10-22	906.62GB	0.02GB	29.13GB	0.00GB	0.00GB	1.81GB	0.00GB	0.00GB	0.00GB	0.7288万次

字段	说明
----	----

存储容量-标准	标准存储类型的存储容量。
存储容量-低频	低频存储类型的存储容量。
存储容量-归档	归档存储类型的存储容量。
下载流量-闲时	00:00-08:00 (闲时) 通过互联网从 US3 下载数据到本地端所产生的流量。
下载流量-忙时	08:00-24:00 (忙时) 通过互联网从 US3 下载数据到本地端所产生的流量。
CDN 回源流量	通过 CDN 服务进行数据回源所产生的流量。
数据取回-低频	低频存储类型请求取回数据产生的流量。
数据取回-归档	归档存储类型请求取回数据产生的流量。
请求次数	访问存储空间进行 API 请求的次数。

令牌管理

US3 令牌功能可以根据用户的需求灵活的开放存储空间和文件管理权限。

一个令牌由一对特殊的公私钥组成。其中包括了:允许操作的存储空间列表、允许操作的文件前缀列表、操作权限和令牌过期时间等属性。

用户可以按需申请不同令牌来完成不同权限的管控。

用户可以通过两种方式管理 US3 令牌:

1. 登陆 UCloud 官方控制台, 进入 US3 - 令牌管理。
2. 通过调用 US3 API 进行创建、编辑等操作。

创建/编辑令牌

创建令牌 ✕

令牌名称 *

有效时长 * 月 ▼

授权存储空间 ① 选择存储空间 已选择: 华北一

黑名单 ①

白名单 ①

授权文件 ① 所有文件 设置前缀

令牌权限 上传 下载 删除 文件列表 图片处理

取消 确定

1. 令牌名称:用于标识令牌,用户自定义。
2. 有效时长:令牌生效的时间,首次设置完成后,可通过编辑令牌操作重新设置过期时间。
3. 授权的存储空间:存储空间可选一个或多个,表示该令牌只能操作这些存储空间。一个令牌不能存在多个地区的存储空间,比如只能选择北京或者上海的存储空间,不能同时选择两个地区。
4. 黑名单:可设置一个或多个IP黑名单,限制IP黑名单中的IP列表访问。
5. 白名单:可设置一个或多个IP白名单,授权IP白名单中的IP列表访问,如未设置白名单,则默认不对IP进行白名单控制,仅限制黑名单的IP地址访问。当前UHost内网由于采用IPv6网络,白名单暂时无法识别内网的IPv4的虚拟IP地址,故白名单暂时还不支持控制UHost主机的内部访问权限。

6. 授权文件:可选择授权所有文件,也可设置一个或多个文件前缀,表示该令牌只能访问这些前缀的文件。
7. 令牌权限:令牌的操作权限有上传、下载、删除、文件列表、图片处理,可多选或者不选。
8. 高级权限:当设置的令牌具有上传权限时,可设置禁止覆盖权限,禁止用户上传时覆盖同名文件。

注意:编辑令牌的生效等待时间约5分钟,请以实际生效时间为准。

示例

一个完整的令牌例子如下所示:

公钥:TOKEN_da044c8a-20bc-42a1-8b04-850535c75330

私钥:6318b15a-faf3-4577-890d-79855313dfd9

过期时间:2019-04-19 00:00:00

授权存储空间:test, test1, test2

授权文件前缀:file_prefix, file_prefix1, file_prefix2

访问白名单:

访问黑名单:

文件管理权限: 上传, 下载, 删除, 文件列表, 图片处理

高级权限:

对象标签

US3支持使用标签对存储空间(Bucket)中的对象(Object)进行分类,用户可以针对同标签的对象设置生命周期规则。生命周期相关请访问 [生命周期](#)。目前对象标签支持越南、巴西,如其他地域有需要,请联系技术支持。

使用说明

对象标签使用一组键值对(Key-Value)标记对象,用户可以在上传对象时添加标签,也可以为已有对象添加标签。

- 单个对象最多可设置10个标签,键(Key)不可重复,且键(key)和值(Value)不支持为空;
- 每个键(key)长度不超过128字符,每个值(Value)长度不超过256字符;
- 键(key)和值(Value)支持大小写字母、数字、空格、和特殊字符: + - = . _ : / @

注意:通过HTTP Header的方式设置标签且标签中包含任意字符时,需要对标签的Key和Value进行URL编码。

设置对象标签

1. 登陆US3对象存储控制台。
2. 单击单地域空间管理,然后单击目标Bucket。
3. 在顶部导航栏,选择文件管理,为对象设置标签:

1) 上传对象时:

单击上传文件,对象标签规则请参见使用说明指定标签的键和值。

The screenshot shows the US3 console interface for uploading an object. The left pane displays a list of objects with columns for '文件名' (File Name) and 'ETag'. The right pane shows the '上传文件' (Upload File) configuration page, which includes a warning banner, '基础设置' (Basic Settings), and '对象标签' (Object Tagging) configuration.

基础设置

上传位置:

选择存储类型:

US3提供标准、低频、归档三种存储类型。如何选择适合的存储类型?

对象标签

针对Object设置标签,详情可参考 [对象标签文档](#)

键	值
<input type="text" value="请输入文本"/>	<input type="text" value="请输入文本"/>
+	

文件上传

文件名	文件大小	存储类型	空间名称	操作
ges	84.71 KB	标准存储	li t	<input type="button" value="X"/>

2) 已上传对象:

在目标对象右侧操作栏下,选择...更多操作列表里的管理标签,支持查看、编辑或删除已添加的标签,对象标签规则请参见使用说明。

单地域空间管理 /

· 概览 · 基础设置 · 文件管理 · 生命周期 · 域名管理 · 镜像回源 · 跨区域复制 · 跨域设置 · 数据处理

上传文件 创建目录 添加解压规则 ...

输入文件前缀查找

文件名	ETag	文件大小	存储类型	MIME-Type	更新时间	操作
	--	--	--	--	--	删除
	--	--	--	--	--	删除
.png.png	AQAAA...	595.31 KB	标准存储	image/png	2023-10-25 13:59:54	预览 下载 ...
ig.png	AQAAA...	556.95 KB	标准存储	image/png	2023-10-25 13:59:53	预览 下载
2 : .png	AQAAA...	487.3 KB	标准存储	image/png	2023-11-07 11:04:33	预览 下载

修改存储类
管理标签
删除
获取URL

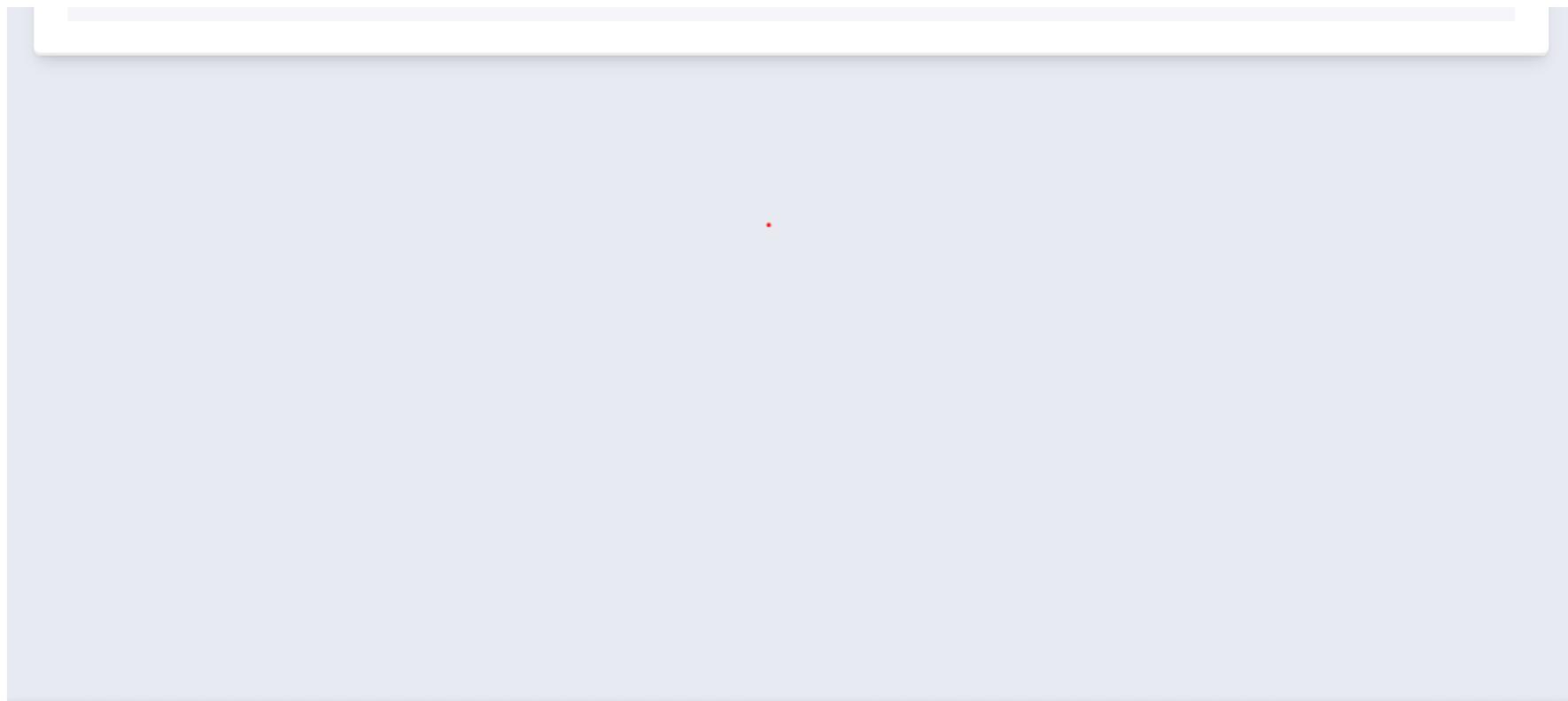
管理标签

您可以对文件设置标签，并根据标签进行生命周期管理，详情可查看[对象标签文档](#)。标签收费标准，请查看[计费文档](#)。

对象标签

键	值	
1	1	×
2	1	×

+



4. 单击确定。

计费说明

US3对象存储按照用户存储空间(Bucket)内对象标签的数量,按天计算对象存储标签费用。

有关US3计费项的定价详情, 请参见产品价格

计费项	单价 (元/每万个标签)
对象标签费用	0.00189

简介

图片处理服务是 UCloud 对外提供的海量、安全、低成本高可靠的图片处理服务。用户将原始图片上传保存在对象存储空间中,用户可以在任何时间、任何地点、任何设备上对图片进行处理。

图片处理参数

US3支持直接在图片URL后添加一个或多个参数处理图片,也支持将多个参数封装在一个样式内批量处理图片。如需了解图片处理参数的相关说明,请参见图片处理参数;图片样式的相关说明,请参见图片样式,目前图片样式支持新加坡、伦敦、首尔、香港、拉各斯、胡志明、迪拜以及台北地域,如其他地域有需要,请联系技术支持。

使用限制

- 图片处理服务仅在下载时触发;
- 图片处理不支持的地域,泰国、圣保罗、孟买、华盛顿、日本,雅加达,法兰克福,若该不支持地域有图片处理需求,请联系技术支持;
- 只有北京、上海、香港地域支持像素10000*10000的图片处理请求;
- 动态图片(例如 GIF 图片)的总像素计算方式为 宽 * 高 * 图片帧数;非动态图片(例如 PNG 图片)的总像素计算方式为 宽 * 高。
- 图片处理不支持range参数,加range参数会返回整个图片;
- 华北二、伦敦、洛杉矶、尼日利亚暂不支持外网水印,如需使用,可将图片先上传至US3的存储桶,图片水印的imageurl可以使用内网域名url;
- 每个存储空间Bucket下最多能创建50个样式,如用户有更多样式需求,请联系技术支持。

计费说明

使用图片处理服务时,会产生如下费用:

- 图片处理费用:包含基本图片处理以及图片高级压缩两种收费模式,价格详情请参见产品价格;
- 请求费用:处理图片时会产生一次GetFile请求,按请求次数收费,价格详情请参见产品价格;
- 流量费用:根据处理后的图片大小收取外网流出的流量费用,价格详情请参见产品价格。

图片处理参数

功能支持

图片处理服务功能列表：

1. 基本图片信息获取
2. EXIF 信息获取
3. 图片缩放
4. 图片裁剪
5. 索引切割
6. 内切圆
7. 圆角矩形
8. 图片旋转
9. 图片自适应方向
10. 图片亮度
11. 图片对比度
12. 图片锐化
13. 图片模糊
14. 获取图片主色调
15. 水印
16. 图片格式转换
17. 图片渐进显示
18. 管道顺序调用多种图片处理功能

基本图片信息获取

图片基本信息包括图片格式、图片大小。

在图片下载 URL 后附加 imageinfo 指示符 (区分大小写), 即可获取 JSON 格式的图片基本信息, 格式为: `https://bucketname.endpoint/objectname?iopcmd=imageinfo`

参数名	值	解释
iopcmd	imageinfo	主命令

获取示例:

```
{
  "width": 640,
  "height": 480,
  "format": "JPEG"
}
```

EXIF信息获取

EXIF (EXchangeable Image File Format) 是专门为数码相机的照片设定的可交换图像文件格式, 通过在图片下载 URL 后附加 exif 指示符 (区分大小写) 获取, 格式为: `https://bucketname.endpoint/objectname?iopcmd=exif`

注意: 缩略图等经过处理的新图片不支持该方法。

参数名	值	解释
-----	---	----

iopcmd	exif	主命令
--------	------	-----

获取示例:

```
{
  "BrightnessValue": "0/100",
  "ColorSpace": "1",
  "ComponentsConfiguration": "...",
  "ExifOffset": "126",
  "ExifVersion": "0220",
  "ExposureBiasValue": "0/6",
  "ExposureMode": "0",
  "ExposureProgram": "0",
  "FlashPixVersion": "0100",
  "ImageLength": "4512",
  "ImageWidth": "4512",
  "InteroperabilityOffset": "398",
  "LightSource": "21",
  "MaxApertureValue": "192/100",
  "MeteringMode": "2",
  "OffsetTime": "+08:00",
  "OffsetTimeOriginal": "+08:00",
  "Orientation": "1",
```

```
"PixelXDimension": "4512",
"PixelYDimension": "4512",
"ResolutionUnit": "2",
"SceneCaptureType": "0",
"SceneType": ".",
"SensingMethod": "0",
"XResolution": "72/1",
"YCbCrPositioning": "1",
"YResolution": "72/1",
"thumbnail:Compression": "6",
"thumbnail:InteroperabilityIndex": "R98",
"thumbnail:JPEGInterchangeFormat": "510",
"thumbnail:JPEGInterchangeFormatLength": "6020",
"thumbnail:ResolutionUnit": "2",
"thumbnail:XResolution": "72/1",
"thumbnail:YResolution": "72/1"
}
```

图片缩放

参数名	值	解释
iopcmd	thumbnail	主命令

type	1	按比例缩放
type	2	宽度不变,高度按百分比缩放
type	3	高度不变,宽度按百分比缩放
type	4	指定宽度,高度等比缩放
type	5	指定高度,宽度等比缩放
type	6	widthXheight,限定长边,短边自适应缩放
type	7	widthXheight,限定短边,长边自适应缩放
type	8	widthXheight,指定高和宽的最小尺寸,等比缩放,如只指定高或宽,则未指定边按照指定数值进行裁剪.但是超出指定矩形会被居中裁剪
type	9	以大的缩放比例进行等比缩放,缩放后可以小于指定的矩形,如高的缩放比例更大,以高为准等比缩放
type	10	以小的缩放比例进行等比缩放,缩放后可以大于指定的矩形,如宽的缩放比例更小,以宽为准等比缩放
type	11	使用指定的宽和高,会导致图像拉伸
type	12	原图尺寸小于 widthXheight 目标尺寸,原图居中,四周补 color 颜色
type	13	等比缩放,一边先缩到目标值,另一边居中裁剪,传 widthXheight
type	14	等比缩放,目标会小于等于 widthXheight
type	15	等比缩放,目标会等于 widthXheight,不足的边,补 color 颜色
color	十六进制或颜色名 red 等	
scale	最大 1000	缩放百分比

gifmode		该参数只对 gif 图片有效。若不传,则该参数默认为 1
gifmode	1	返回连续的图像帧,如果不能处理全部的图像帧则只返回部分,同 3
gifmode	2	返回第一帧静态图片,图像格式仍为 gif
gifmode	3	在系统处理的超时时间内,返回连续的图像帧,如果不能处理全部的图像帧则只返回部分
gifmode	4	在系统处理的超时时间内,如果不能处理全部的图像帧,则返回部分跳跃的图像帧,图像帧平均分布
gifmode	5	在系统处理的超时时间内,如果不能处理全部的图像帧,则返回部分跳跃的图像帧,图像帧平均分布,同时用前一帧填充跳跃的帧,保持处理前后总帧数数量不变
gifmode	6	在系统处理的超时时间内,如果不能处理全部的图像帧,则返回部分跳跃的图像帧,图像帧平均分布,同时修改图像帧的延迟时间,使播放速度和原图相同

图片裁剪

参数名	值	解释
iopcmd	crop	主命令
width	0-10000	裁剪后宽度
height	0-10000	裁剪后高度
ax	0-10000	锚点 x 坐标
ay	0-10000	锚点 y 坐标

gravity	NorthWest/North/NorthEast/West Center/East/SouthWest/South/SouthEast	位置偏移指示符,可以和锚点同时指定(比锚点优先使用)
---------	----------------------------------------------------------------------	----------------------------

索引切割

参数名	值	解释
iopcmd	indexcrop	主命令
x	1-图片宽度	x方向切割的每块区域长度
y	1-图片高度	y方向切割的每块区域长度
i	0-区域数,默认0表示第一块	选择切割后返回图片的区域

注意事项

- 如果指定索引值大于切割后形成的区域数量,将返回原图
- 当x和y同时指定值合法时,按照y的参数为准

内切圆

参数名	值	解释
iopcmd	circle	主命令
r	1-4096	指定内切圆半径

注意事项

- 如果图片的源格式是PNG、WebP或BMP等支持透明通道的图片,那么图片的非圆型区域将以透明补充,如果图片的源格式是JPG,图片的非圆形区域会以白色进行填充
- 当r的取值大于原图最小边的一半时,以原图最小边的一半为值返回内切圆

圆角矩形

参数名	值	解释
iopcmd	rounded-corners	主命令
r	1-4096	将图片切出圆角,指定圆角的半径

注意事项

- 如果图片的源标格式是PNG、WebP或BMP等支持透明通道的图片,那么图片的圆角以外区域将以透明补充,如果图片的源格式是JPG,图片的圆角以外区域会以白色进行填充

图片旋转

参数名	值	解释
iopcmd	rotate	主命令
degree	[0,360]	旋转度数(顺时针)

自适应方向

参数名	值	解释
iopcmd	auto-orient	主命令
value	0,1	0:不对图片进行自适应方向旋转,1:对图片进行自适应方向旋转

亮度

参数名	值	解释
iopcmd	bright	主命令
value	[-100,100]	<0:降低图片亮度,=0:不调整图片亮度,>0:提高图片亮度

注意事项

- 如果取值 ≥ 100 或者 ≤ -100 ,图片会变成全白或者全黑

对比度

参数名	值	解释
iopcmd	contrast	主命令

value	[-100,100]	<0:降低图片对比度,=0:不调整图片对比度,>0:提高图片对比度
-------	------------	-----------------------------------

注意事项

- 如果取值 ≥ 100 或者 ≤ -100 ,图片会变成全白或者全黑

锐化

参数名	值	解释
iopcmd	sharpen	主命令
radius	[0,10]	设置锐化半径
sigma	[0,300]	设置正态分布标准差

注意事项

- 锐化半径radius越大,越清晰,但是数值过大,计算会比较慢,可能会出现失真情况

模糊

参数名	值	解释
iopcmd	blur	主命令

type	1,2	1:普通模糊,2:高斯模糊,不传默认为普通模糊
radius	[0,50]	设置锐化半径,默认为0
sigma	[0,50]	设置正态分布标准差,不传默认为0
width	0-10000	局部模糊的宽度,不传默认为图片的宽度,为全部模糊
height	0-10000	局部模糊的宽度,不传默认为图片的宽度,为全部模糊
ax	0-10000	锚点 x 坐标
ay	0-10000	锚点 y 坐标
gravity	NorthWest/North/NorthEast/West Center/East/SouthWest/South/SouthEast	位置偏移指示符,可以和锚点同时指定(比锚点优先使用)

获取图片主色调

参数名	值	解释
iopcmd	averagehue	返回图片的主色调

水印

参数名	值	解释
iopcmd	watermark	主命令

type	1=文字水印,2=图片水印	水印类型
gravity	NorthWest/North/NorthEast/West Center/East/SouthWest/South/SouthEast	位置偏移指示符,可以和锚点同时指定(比锚点优先使用)
opacity	1-100,缺省为 100 (完全不透明)	透明度
ax	小于原图宽度,默认为 10	锚点 x 坐标
ay	小于原图高度,默认为 10	锚点 y 坐标
text	base64 URL encode 后的水印文字	base64 URL encode 后的水印文字,支持 UTF-8
imageurl	资源地址	可访问的图片资源URL,水印图片,必须是可以访问的资源地址,经过 base64 URL encode,后续会不支持外网水印URL,并且仅支持对象存储公开bucket内网域名水印的URL
fontsize	1-100	水印文字大小,缺省为10,字体大小
rotate	0-360	水印文字旋转角度,默认为0,不旋转
fill		RGB 颜色编码表,base64 URL encode 后的RGB格式,可以是颜色名称(比如red)或十六进制(比如#FF0000),缺省为白色。
font	字体格式编码类型参考下方编码对应表	base64 URL encode 后的字体名称,默认宋体

font参数中可选的文字类型以及编码对应表

字体	中文名	编码值
SimSun	宋体	5a6L5L2T / U2ItU3Vu
LiShu	隶书	TGIaHU=

FangZhengFangSong	方正仿宋	RmFuZ1poZW5nRmFuZ1Nvbmc=
FangZhengHeiTi	方正黑体	RmFuZ1poZW5nSGVpVGk=
FangZhengKaiTi	方正楷体	RmFuZ1poZW5nS2FpVGk=
FangZhengShuSong	方正书宋	RmFuZ1poZW5nU2h1U29uZw==
WenQuanWeiMiHei	微泉微米黑	V2VuUXVhbldlaU1pSGVp
WenQuanYiZhengHei	微泉驿正黑	V2VuUXVhblpoZW5nSGVp
DroidSansFallback	DroidSansFallback	RHJvaWRTYW5zRmFsbGJhY2s=

图片格式转换

参数名	值	解释
iopcmd	convert	主命令
dst	jpg/png/bmp/webp	目标格式
Q	0-100	图片压缩质量(绝对压缩比),目标格式为 jpg,webp 时有效
q	0-100	图片压缩质量(相对压缩比),即在图片原有压缩基础上按此比例进行压缩,目标格式为 jpg,webp 时有效,Q 与 q 同时出现时,取 Q
fr	0.0-1.0	需要格式转换的图像帧在图像总帧中的位置,默认为 0.0

图片高级压缩:支持将JPG、PNG等格式图片转码为WEBP格式,更高效地将图片转换为高压缩率格式,价格详情请参见产品价格。

渐进显示

参数名	值	解释
iopcmd	interlace	主命令
value	0,1	0:不对图片进行渐进显示,1:对图片进行渐进显示

注意事项

- 用以开启先模糊后清晰的呈现方式,非jpg格式的图片会转换为jpg格式再处理

管道顺序调用多种图片处理功能

管道顺序是一种可以实现多种图片处理任务顺序执行的机制。用户可以通过管道顺序在一次下载时中按照顺序完成对图像的不同处理。

如:

```
http://ufile-release.cn-bj.ufileos.com/US3pic/demo.jpg?iopcmd=rotate&degree=180|iopcmd=thumbnail&type=1&scale=40
```

上述请求会将 demobucket 这个空间中的 here-for-example.jpg 文件先旋转 180 度,然后按比例缩放为原来的 40% 大小。

注意:如果请求当前不支持的 iopcmd,则会报错,不会返回原图。

图片样式

US3支持将多个图片处理参数封装在一个样式 (Style) 内批量处理图片,可以快速满足复杂的图片处理需求,在控制台支持创建样式、编辑样式以及删除样式等操作,目前仅支持新加坡地域,其他地域若有需要,请联系技术支持。

创建样式

1. 登陆US3管理控制台。

对象存储 US3

· 单地域空间管理 · 统计报表 · 管理工具及SDK · 令牌管理

🔔 对象存储 US3 服务等级协议 (SLA) 于 2021年3月1日 更新, [了解详情](#)

创建存储空间 更改业务组 删除存储空间

<input type="checkbox"/>	存储空间域名	资源ID	地域 ▼	默认存储类型 ▼	空间类型 ▼	数据分
<input type="checkbox"/>	lambert.test03.ufileos.com	ufile-ogljybzj	TEST03	标准	私有空间	
<input type="checkbox"/>	test-cwb.test03.ufileos.com	ufile-ciTvT8s9kFX	TEST03	标准	公开空间	
<input type="checkbox"/>	test-ft.test03.ufileos.com	ufile-3dkllobd	TEST03	标准	公开空间	
<input type="checkbox"/>	uds-dst-test.test03.ufileos...	ufile-UFyyUpRxVks	TEST03	标准	私有空间	
<input type="checkbox"/>	lwtestforyy10.test03.ufileo...	ufile-0gqlbrzg	TEST03	标准	公开空间	
<input type="checkbox"/>	test-ft3.test03.ufileos.com	ufile-UZZtwRPNroV	TEST03	标准	私有空间	
<input type="checkbox"/>	s3proxy-test03.test03.ufile...	ufile-3cbr0kef	TEST03	低频	私有空间	

2. 在单地域空间管理, 单击目标Bucket。

对象存储 US3

· 单地域空间管理 · 统计报表 · 管理工具及SDK · 令牌管理

对象存储 US3 服务协议 (SLA) 于 2021年3月1日 更新, [了解详情](#)

创建存储空间 更改业务组 删除存储空间

搜索: ab; 搜索结果: 1条; [返回列表](#)

<input type="checkbox"/>	存储空间域名 🔗	资源ID	地域 ▼	默认存储类型 ▼	空间类型 ▼	数据分析 📊	创建时间 🕒	操作
<input type="checkbox"/>	abthony.test03.ufileos.com	ufile-UFyyTswADyq	TEST03	标准	公开空间	📊	2022-04-27	详情 文件管理 域名管理 ...

< 1 > 10

3. 选择数据处理>图片处理, 然后单击添加样式。



4. 在添加样式面板配置样式,可以使用可视化编辑和代码编辑两种方式添加样式:

< 单地域空间管理 / 图片处理 / 添加样式

可视化编辑

代码编辑

基本处理

格式转换

原图格式

渐进显示

用以开启先模糊后清晰的呈现方式。会将图片转换为jpg格式再处理。

OFF

自适应方向

对图片进行自适应方向旋转

OFF

图片尺寸

缩放方式

不进行缩放

裁剪方式

不进行裁剪

预览



取消

确定

预览



取消 确定

图片效果

图片旋转

90
 180
 270
 360
 度

图片亮度

图片对比度

图片锐化

锐化半径radius越大，越清晰，但是数值过大，计算会比较慢，可能会出现失真情况，标准差建议设置锐化半径的三分之一。

开启并设置参数

图片模糊

开启并设置参数

水印样式

水印类型

管理设置

样式名称



- 可视化编辑:通过可视化界面选择需要的图片处理功能,如基本处理、图片尺寸、图片效果、水印等。
 - 代码编辑:使用代码编辑图片处理功能,格式为*iopcmd=action1¶me=value1|iopcmd=action2¶me=value2...*。
5. 输入样式名称,点击确定即可

使用样式

图片样式创建完成后,用户可以直接将图片样式添加到图片的访问URL上,格式为:<https://bucketname.endpoint/objectname?iopstyle=stylename>

编辑样式

< 单地域空间管理 / abthony

概览 基础设置 文件管理 生命周期 域名管理 镜像回源 跨区域复制 跨域设置 数据处理

图片处理
ZIP包解压

添加样式 访问设置

样式名称 样式代码 操作

iop-indexcrop 	iopcmd=indexcrop&x=80&y=50&i=10	编辑 删除
iop-circle 	iopcmd=circle&r=200	编辑 删除
iop-clr 	iopcmd=indexcrop&x=80&i=2	编辑 删除
test-code 	iopcmd=rotate°ree=0 iopcmd=bright&value=0 iopcmd=contrast&value=0	编辑 删除
test-scale-color 	iopcmd=thumbnail&type=12&width=1100&height=1100&color=%238691cb	编辑 删除
test-wm 	iopcmd=watermark&type=1&opacity=100&gravity=Center&ax=10&ay=10&text...	编辑 删除
test-imgwm 	iopcmd=watermark&type=2&opacity=100&gravity=North&ax=10&ay=30&image...	编辑 删除

搜索 刷新 文档

删除样式

< 单地域空间管理 / anthony

· 概览 · 基础设置 · 文件管理 · 生命周期 · 域名管理 · 镜像回源 · 跨区域复制 · 跨域设置 · 数据处理

图片处理

ZIP包解压

添加样式 访问设置

样式名称	样式代码	操作
iop-indexcrop 	iopcmd=indexcrop&x=80&y=50&i=10 	编辑 删除
iop-circle 	iopcmd=circle&r=200	编辑 删除
iop-cir 	iopcmd=indexcrop&x=80&i=2	编辑 删除
test-code 	iopcmd=rotate°ree=0 iopcmd=bright&value=0 iopcmd=contrast&value=0	编辑 删除
test-scale-color 	iopcmd=thumbnail&type=12&width=1100&height=1100&color=%238691cb	编辑 删除
test-wm 	iopcmd=watermark&type=1&opacity=100&gravity=Center&ax=10&ay=10&text...	编辑 删除
test-imgwm 	iopcmd=watermark&type=2&opacity=100&gravity=North&ax=10&ay=30&image...	编辑 删除

注意事项

- 每个存储空间Bucket下最多能创建50个样式, 这些样式仅适用于该Bucket下的图片
- 可视化编辑尚未完全支持所有的样式功能, 所以代码编辑后再次进入编辑的时候, 可视化编辑的结果可能会丢失
- 当选择编辑框里面的参数, 若图片没有加载成功, 打开开发者模式检查对应返回错误, 修改对应参数。

原图保护

原图保护功能可以防止US3内允许匿名访问的图片文件被盗用,开启原图保护后,匿名者可以使用携带样式参数的请求或通过签名URL访问原图,目前仅支持新加坡、伦敦、首尔、香港、拉各斯、胡志明、迪拜以及台北地域,其他地域若有需要,请联系技术支持。

访问方式

用户可以通过以下方式访问开启原图保护的图片:

- 使用携带样式参数的文件URL访问,格式为:`https://bucketname.endpoint/objectname?iopstyle=stylename`
- 使用携带签名的文件URL访问,格式为:`https://bucketname.endpoint/objectname?signature`

操作步骤

1. 登陆US3管理控制台

对象存储 US3

· 单地域空间管理 · 统计报表 · 管理工具及SDK · 令牌管理

🔔 对象存储 US3 服务协议 (SLA) 于 2021年3月1日 更新, [了解详情](#)

创建存储空间 更改业务组 删除存储空间

<input type="checkbox"/>	存储空间域名	资源ID	地域 ▼	默认存储类型 ▼	空间类型 ▼	数据分
<input type="checkbox"/>	lambert.test03.ufileos.com	ufile-ogljybzj	TEST03	标准	私有空间	
<input type="checkbox"/>	test-cwb.test03.ufileos.com	ufile-ciTvT8s9kFX	TEST03	标准	公开空间	
<input type="checkbox"/>	test-ft.test03.ufileos.com	ufile-3dkllobd	TEST03	标准	公开空间	
<input type="checkbox"/>	uds-dst-test.test03.ufileos...	ufile-UFyyUpRxVks	TEST03	标准	私有空间	
<input type="checkbox"/>	lwtestforyy10.test03.ufileo...	ufile-0gqlbrzg	TEST03	标准	公开空间	
<input type="checkbox"/>	test-ft3.test03.ufileos.com	ufile-UZZtwRPNroV	TEST03	标准	私有空间	
<input type="checkbox"/>	s3proxy-test03.test03.ufile...	ufile-3cbr0kef	TEST03	低频	私有空间	

2. 在单地域空间管理, 单击目标Bucket

对象存储 US3

· 单地域空间管理 · 统计报表 · 管理工具及SDK · 令牌管理

对象存储 US3 服务协议 (SLA) 于 2021年3月1日 更新, [了解详情](#)

创建存储空间 更改业务组 删除存储空间

搜索: ab; 搜索结果: 1条; [返回列表](#)

<input type="checkbox"/>	存储空间域名 📄	资源ID	地域 ▼	默认存储类型 ▼	空间类型 ▼	数据分析 📊	创建时间 📅	操作
<input type="checkbox"/>	abthony.test03.ufileos.com	ufile-UFyyTswADyq	TEST03	标准	公开空间	📈	2022-04-27	详情 文件管理 域名管理 ...

< 1 > 10

3. 选择数据处理>图片处理, 然后单击访问设置。

< 单地域空间管理 / abthony

· 概览 · 基础设置 · 文件管理 · 生命周期 · 域名管理 · 镜像回源 · 跨区域复制 · 跨域设置 · 数据处理

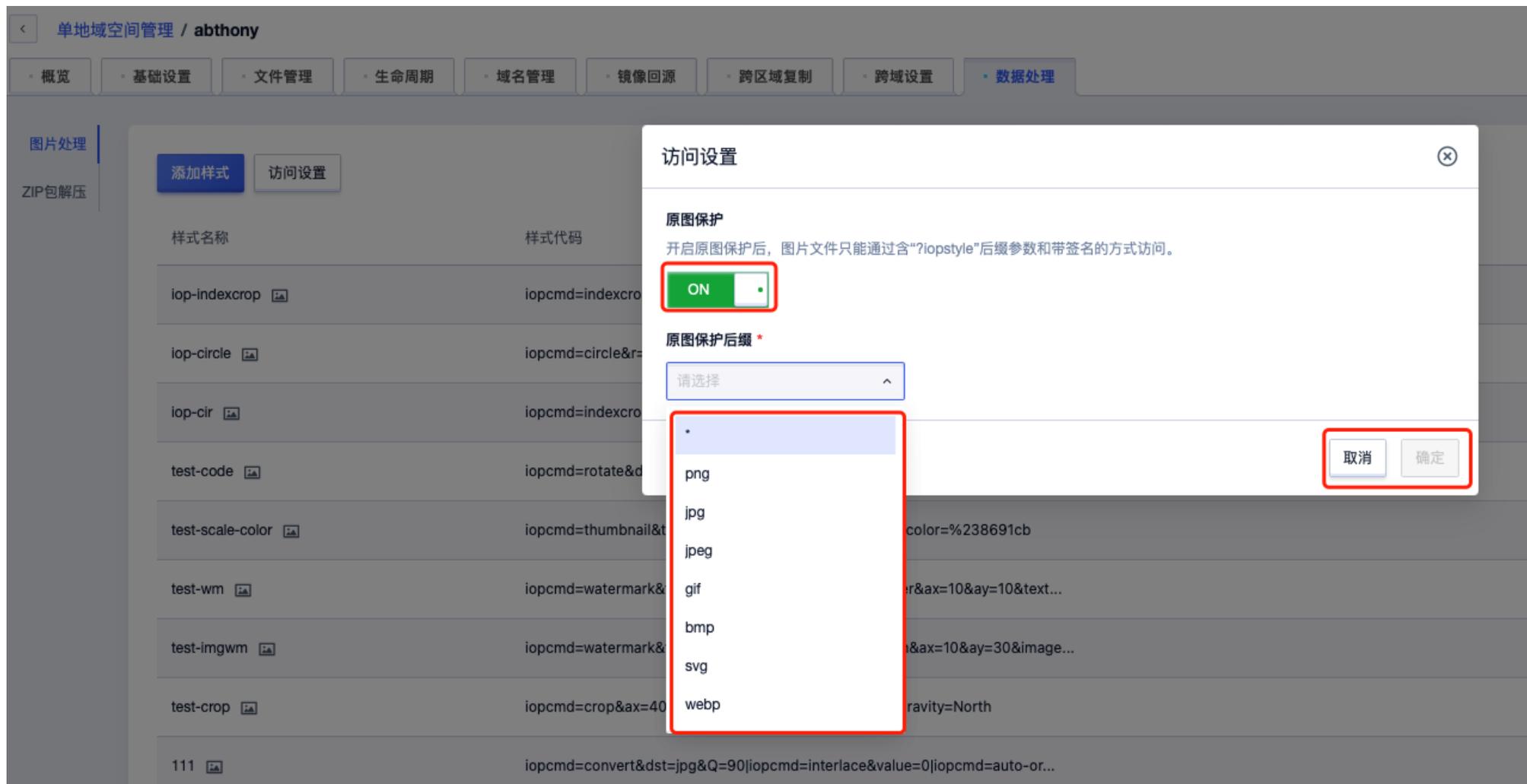
图片处理

ZIP包解压

添加样式 访问设置

样式名称	样式代码
lop-indexcrop 	iopcmd=indexcrop&x=80&y=50&i=10 
lop-circle 	iopcmd=circle&r=200
lop-cir 	iopcmd=indexcrop&x=80&i=2
test-code 	iopcmd=rotate°ree=0 iopcmd=bright&value=0 iopcmd=contrast&value=0
test-scale-color 	iopcmd=thumbnail&type=12&width=1100&height=1100&color=%238691cb
test-wm 	iopcmd=watermark&type=1&opacity=100&gravity=Center&ax=10&ay=10&text...

4. 在访问设置面板, 打开原图保护开关, 单击确定保存



- 原图保护后缀:在原图保护后缀下拉列表选择文件后缀,Bucket内所有对应后缀的图片文件都会被保护,(*为所有格式)。

注意事项

- 原图保护仅针对允许匿名访问的文件,使用签名URL访问图片属于非匿名访问。所以开启原图保护后,如果使用带签名的URL、或者CDN回源到了私有Bucket,仍然可以访问原图。

解压缩服务

解压缩服务是UCloud对外提供的低成本高可靠的解压服务,用户通过在控制台的存储桶设置解压缩规则,设置触发前缀,解压后的目标bucket及目标路径,实现在规则前缀下上传的*.zip文件进行自动解压。

注意事项

- 目前解压缩处于公测阶段,只支持上海和华北一地域,仅仅支持后缀zip包文件的解压
- 每个bucket最多只能创建十条解压缩规则
- 每个规则的筛选前缀之间不能互相包含,例如:有个规则的筛选前缀为 /a/b/c/d,则不能再创建/a/b/c和/a/b/c/d/e的规则前缀
- 目标目录默认不填,会自动解压到你上传的的key的前缀作为目录
- 源bucket令牌的权限要具备上传下载权限,如果令牌设置了prefix,则不能和自己规则的筛选前缀有冲突关系,否则会解压失败
- 目标bucket令牌的权限要具备上传下载权限,如果令牌设置了prefix,则不能和目标目录有冲突关系,否则会解压失败
- 注意目标目录的设置和筛选前缀的设置,解压后若还存在zip文件,可能还会触发解压行为
- 单个压缩包的最大解压时间为10分钟,超过10分钟未完成的任务会解压失败
- 创建规则包含两个入口,一个是存储桶的bucket文件管理的创建规则,另一个是数据处理下面的创建规则
- 解压缩目前不收费,后续会按照使用次数,资源使用以及公网出流量三个维度进行收费

配置解压缩

创建规则

- 1.登录控制台
- 2.点击打开目标bucket功能详情,点击【数据处理】或【文件管理】

- 3.在zip包解压列表点击【添加规则】
- 4.填写相关参数,确认创建规则

< 单地域空间管理 / abthony

概览 基础设置 文件管理 生命周期 域名管理 镜像回源 跨区域复制 跨域设置 **数据处理**

ZIP包解压

添加规则

规则名称	触发范围	解压格式	目标存储桶	目标路径	解压配置	操作
test-li	...	ZIP	...	/a/b	保留压缩文件名为路径目录	修改 删除
test-li2	...	ZIP	...		保留压缩文件名为路径目录	修改 删除
test-li3	前缀: appq	ZIP	...	/wahah/ksdl	直接解压到目标目录中	修改 删除
...	前缀: ...	ZIP	uc-dst-test	...	保留压缩文件名为路径目录	修改 删除
gen	前缀: nakelulu/	ZIP	...	/	直接解压到目标目录中	修改 删除

< 1 > 10条/页 1/1

< 单地域空间管理 / abthony

概览 基础设置 文件管理 生命周期 域名管理 镜像

ZIP包解压

添加规则

规则名称	触发范围	解压格式
test-li	前缀: lishisong/	ZIP
test-li2	前缀: test/	ZIP
test-li3	前缀: appq	ZIP

→

添加规则

触发范围 * 指定前缀 全部文件

/ 请输入前缀

指定的前缀在同一个Bucket下的不同触发条件不能重叠包含。

解压格式 ZIP

源Bucket令牌 * abthony ↕

目标Bucket * ued2333 ↕

目标Bucket目录 ⓘ 请输入目标Bucket目录

目标Bucket令牌 * ceshi ↕

解压配置 保留压缩文件名为路径目录 直接解压到目标目录中

若您的目标目录为A/, 压缩文件名为B.zip, 则解压后文件路径为A/B/...

规则名称 * 请输入规则名称

取消

确定

参数介绍

参数	是否必选	说明
触发前缀	是	解压触发前缀设置, 在该前缀目录下上传的*.zip文件均会自动触发解压, 规则的前缀之间不能存在互相包含的关系, 如果已经存在规则, 则不能选择全部文件作为触发条件, 若已经有规则的触发条件是全部文件, 则不能创建其他规则, 目录格式a/b/
源bucket令牌	是	bucket令牌需要具有上传下载权限, 若令牌权限设置了prefix, 则该prefix应该包含触发前缀, 否则会解压失败
目标bucket	是	选择解压后文件需要上传的bucket
目标bucket令牌	是	bucket令牌需要具有上传下载权限, 若令牌权限设置了prefix, 则该prefix应该包含目标目录, 否则会解压失败
目标路径	否	解压后上传的目标路径, 不传默认解压到上传的*.zip的key目录, 上传到根目录输入"/", 其他目录格式"a/b/c"
解压配置	是	解压后的目录是否保留文件名为最后一层目录, 目标目录为a/, 压缩文件名为b.zip, 若选择保留, 则解压后文件路径为a/b/, 若选择不保留, 则解压后的文件路径为a/
规则名字	是	解压缩规则名字

配置用例:

- 上传到src前缀下, 解压到dst目录, 不保留压缩文件名为目录, 前缀设置为src/, 目标目录设置为dst/
- 解压后的文件结构

```

bucket-src
├── src/
│   ├── a.zip
│   └── b.zip
bucket-dst
├── dst/

```

```
|— a.txt
|— b.txt
└— ...
```

- 上传到src前缀下,解压到dst目录,保留压缩文件名为目录,前缀设置为src/,目标目录设置为dst/
- 解压后的文件结构

```
bucket-src
└— src/
  |— a.zip
  └— b.zip
bucket-dst
└— dst/
  |— a/
  | |— a.txt
  |— b/
  |— b.txt
```

- 上传到src前缀下,解压到dst目录,保留压缩文件名为目录,前缀设置为src/,目标目录设置为空,默认解压到上传的*.zip文件的key目录下
- 解压后的文件结构

```
bucket-src
└— src/
  |— a.zip
  └— b.zip
bucket-dst
```

```
└── src/  
    ├── a/  
    │   └── a.txt  
    ├── b/  
    └── b.txt
```

修改规则

选择US3指定存储空间,在右侧操作中点击数据处理按钮,会展示所有的该存储空间的解压缩规则,选择想要修改的规则即可

< 单地域空间管理 / anthony

概览 基础设置 文件管理 生命周期 域名管理 镜像回源 跨区域复制 跨域设置 数据处理

ZIP包解压

添加规则

规则名称	触发范围	解压格式	目标存储桶	目标路径	解压配置	操作
test-li	前缀: lishisong/	ZIP	uds-dst-test	/a/b	保留压缩文件名为路径目录	修改 删除
test-li2	前缀: test/	ZIP	uds-dst-test		保留压缩文件名为路径目录	修改 删除
test-li3	前缀: appq	ZIP	uds-dst-test		保留压缩文件名为路径目录	修改 删除

< 1 > 10 条/页 /1

点击修改以后,会弹出修改窗口,修改想要修改对应窗口里面的字段即可

单地域空间管理 / abthony

概览 基础设置 文件管理 生命周期 域名管理 镜像

ZIP包解压

添加规则

规则名称	触发范围	解压格式
test-li	前缀: lishisong/	ZIP
test-li2	前缀: test/	ZIP
test-li3	前缀: appq	ZIP

编辑规则

触发范围 * 指定前缀 全部文件

/ lishisong/

! 指定的前缀在同一个Bucket下的不同触发条件不能重叠包含。 ×

解压格式 ZIP

源Bucket令牌 * abthony ↻

目标Bucket * uds-dst-test ↻

目标Bucket目录 ⓘ /a/b

目标Bucket令牌 * uds-dst-test ↻

解压配置 保留压缩文件名为路径目录 直接解压到目标目录中

! 若您的目标目录为A/, 压缩文件名为B.zip, 则解压后文件路径为A/B/... ×

规则名称 * test-li

取消 确定

删除规则

对于不需要的zip包解压规则,您可以手动删除,选择US3指定存储空间,点击数据处理按钮,选择想要删除的规则,删除过后,该规则不会自动触发解压

< 单地域空间管理 / abthony

概览 基础设置 文件管理 生命周期 域名管理 镜像回源 跨区域复制 跨域设置 数据处理

ZIP包解压

添加规则

规则名称	触发范围	解压格式	目标存储桶	目标路径	解压配置	操作
test-li	前缀: lishisong/	ZIP	uds-dst-test	/a/b	保留压缩文件名为路径目录	修改 删除
test-li2	前缀: test/	ZIP	uds-dst-test		保留压缩文件名为路径目录	修改 删除
test-li3	前缀: appq	ZIP	uds-dst-test		保留压缩文件名为路径目录	修改 删除

< 1 > 10条/页 /1

依据弹窗的提示选择确定删除或者取消操作

< 单地域空间管理 / abthony

概览 基础设置 文件管理 生命周期 域名管理 镜像回源 跨区域复制 跨域设置 数据处理

ZIP包解压

添加规则

规则名称	触发条件	操作
test-li	前缀:	修改 删除
test-li2	前缀: test/	ZIP uds-dst-test 保留压缩文件名为路径目录 修改 删除
test-li3	前缀: appq	ZIP uds-dst-test 保留压缩文件名为路径目录 修改 删除

1 10条/页 /1

确认删除规则

是否确认删除解压规则 test-li, 删除规则后将无法通过该规则解压文件。

取消 确定

文档预览

概念

文档预览功能支持文字文件、演示文件、表格文件、以及pdf文件的在线预览,便于您进行文档内容的查看。

注意事项

- 支持在线预览的文件类型
 - 支持以pdf的格式预览的文件类型:ppt、pptx、wps、doc、docx、xls、xlsx
 - 支持以h5格式预览的文件类型:pptx
- 文件大小限制
 - 不支持在线预览大于200MB的文件

调用参数

对于支持在线预览的文件格式,可在文件url上拼接参数 `x-udi-process=doc-preview&dstFormat=${format}` 进行预览

举例说明:

- 对于公共bucket

`http://{bucketName}.cn-bj.ufileos.com/sample.pptx?x-udi-process=doc-preview&dstFormat=h5`

- 对于私有bucket

`http://{bucketName}.cn-bj.ufileos.com/sample.pptx?UCloudPublicKey=${publicKey}&Signature=${signature}&Expires=1649832250&x-udi-process=doc-preview&dstFormat=h5`

dstFormat的值为pdf或h5

控制台预览

在文件管理tab页内,对于支持预览的文件,在操作一栏上会有预览按钮,点击即可进行文件预览

< 单地域空间管理 / doc-preview-test

· 概览 · 基础设置 · 文件管理 · 生命周期 · 域名管理 · 镜像回源 · 跨区域复制 · 跨域设置

上传文件 创建目录 修改MIME-Type 删除

输入文件前缀查找 🔍 ⚙️ ↻️ ?

📁 doc-preview-test

<input type="checkbox"/>	文件名	ETag	文件大小	存储类型	MIME-Type	更新时间	操作
<input type="checkbox"/>	 demo.jpeg	AgAAAODXTAgB9HpVJ0f59VXiRj6UyQmY	5.52 MB	标准存储	image/jpeg 	2022-04-13 13:54:23	预览 下载 ...
<input type="checkbox"/>	 demo.pptx	AgAAAht-kfP-KWkZn3NxnANzFTi7apl	6.31 MB	标准存储	application/vnd.openxmlform... 	2022-04-13 11:57:54	预览 下载 ...
<input type="checkbox"/>	 demo.zip	AQAAAIKtJAu5x2Ro0pwtyrU-7WoGTI3Z	1.03 MB	标准存储	application/zip 	2022-04-13 13:54:35	获取地址 下载 ...

< 单地域空间管理 / doc-preview-test

· 概览 · 基础设置 · 文件管理 · 生命周期 · 域名管理 · 镜像回源 · 跨区域复制 · 跨域设置

上传文件 创建目录 修改MIME-Type 删除

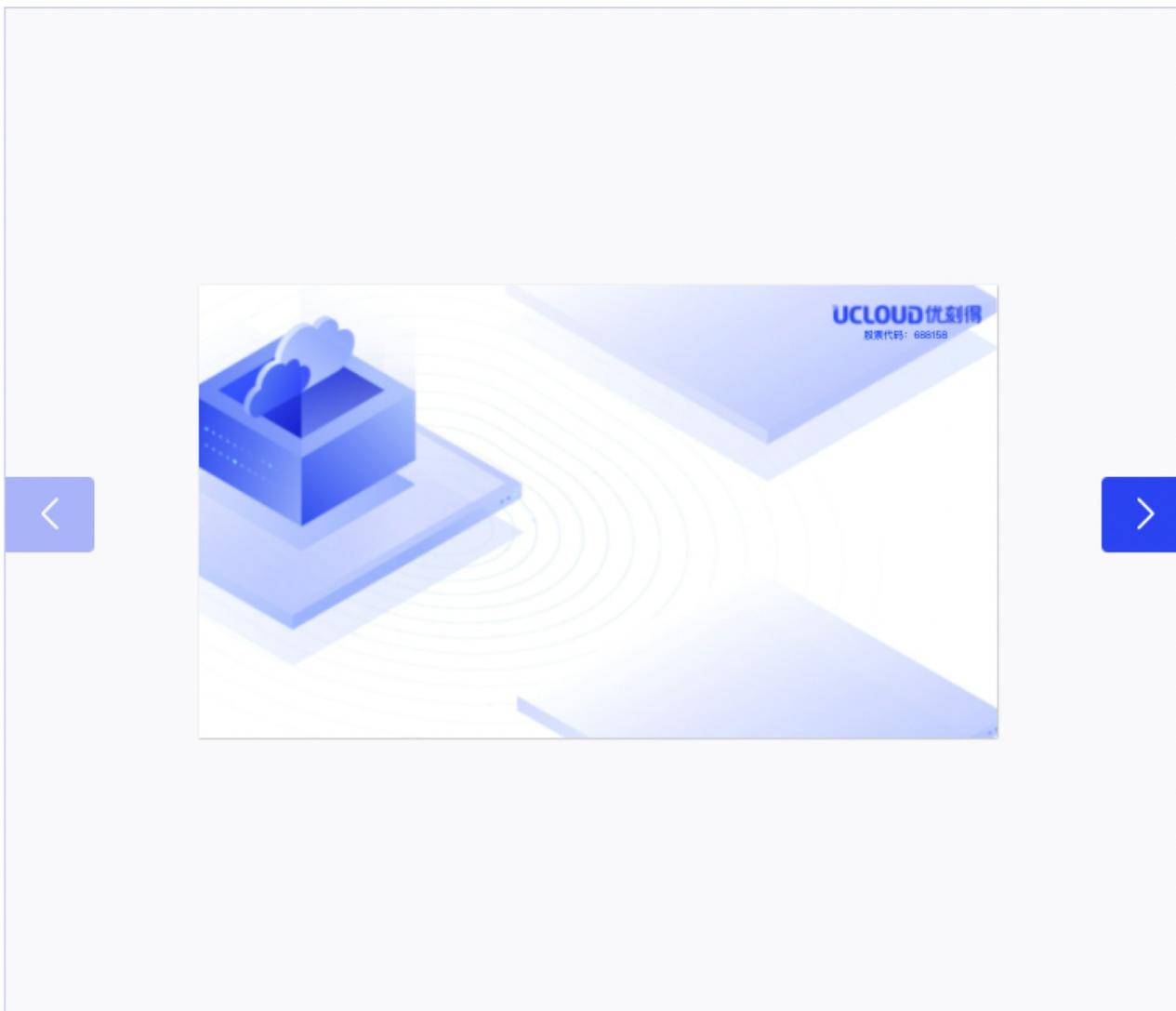
输入文件前缀查找 🔍 ⚙️ ↻️ ?

📁 doc-preview-test

demo.pptx 

打开预览链接 复制预览链接

- 文件名
-  demo.jpeg
-  demo.pptx
-  demo.zip
-  买菜战术手册



操作

预览 下载 ...

预览 下载 ...

获取地址 下载 ...

预览 下载 ...

关闭

开放地域

- 华北一
- 华北二
- 上海
- 广州

内容审核服务

概览

对象存储提供内容审核功能,可以对存储在US3中的图片进行违禁内容识别,识别的场景包括涉黄、涉恐、涉政。

详细介绍见UDI数据智能文档

操作步骤

1. 登陆UDI管理控制台。

The screenshot displays the '数据智能 UDI' (Data Intelligence UDI) interface. At the top, there are navigation elements including '全部产品' (All Products), 'Default', and '华北一' (North China 1). The main area features a table of tasks with columns for '任务名称' (Task Name), '存储空间' (Storage Space), '任务类型' (Task Type), '审核内容' (Review Content), '规则范围' (Rule Scope), '审核类型' (Review Type), '状态' (Status), and '操作' (Actions). The table contains eight rows of task data. Below the table, there is a pagination control showing '1' of 1 page and '10 条/页' (10 items per page).

任务名称	存储空间	任务类型	审核内容	规则范围	审核类型	状态	操作
		增量审核	图片	所有文件	涉黄、涉政、涉暴恐	进行中	查看结果 任务详情 暂停 删除
		存量审核	图片	所有文件	涉黄、涉政、涉暴恐	完成	查看结果 任务详情 删除
		存量审核	图片	所有文件	涉黄、涉政、涉暴恐	完成	查看结果 任务详情 删除
		存量审核	图片	所有文件	涉黄、涉政、涉暴恐	完成	查看结果 任务详情 删除
		增量审核	图片	所有文件	涉黄、涉政、涉暴恐	进行中	查看结果 任务详情 暂停 删除
		存量审核	图片	所有文件	涉黄、涉政、涉暴恐	完成	查看结果 任务详情 删除
		存量审核	图片	所有文件	涉黄、涉政、涉暴恐	完成	查看结果 任务详情 删除

2. 单击“创建任务”，选择对应的基础和审核设置，各参数说明请参见参数说明表。

[内容审核 / 创建任务](#)

基础设置

选择地域 华北一

存储空间 ▼ ↻

任务类型 增量审核 存量审核

审核内容 图片

规则范围 所有文件 部分文件

审核设置

图片格式 JPG PNG JPEG

审核类型 涉黄 涉政 涉暴恐

违规处理 处理并迁移

任务名称

取消 立即创建

3. 配置各项参数后,单击“立即创建”即为任务创建完成。其中首次使用UDI的存储空间需进行授权确认。



4. 已创建的内容审核任务支持“查看结果”、“任务详情”、“暂停”和“删除”操作。

1) 查看结果: 单击对应任务的“查看结果”, 该页面展示审核异常的详细记录, 分别标签异常图片的审核类型和结果, 同时也支持对异常图片的迁移、删除和忽略操作。

数据智能 UDI

创建任务 删除

任务名称	存储空间
<input type="checkbox"/> tt	0428
<input type="checkbox"/> test0428	0428
<input type="checkbox"/> 0428-2	0428
<input type="checkbox"/> test_censor	0428
<input type="checkbox"/> tt-aab	0428
<input type="checkbox"/> test0428-2	0428
<input type="checkbox"/> test_bj	0428
<input type="checkbox"/> test	0428
<input type="checkbox"/> test	0428

近 7 天审核异常记录

4	2	0
涉黄数量	涉政数量	涉暴恐数量

类型筛选

全选 涉黄 涉政 涉暴恐

结果筛选

全选 违规 疑似

迁移 删除 忽略

最近7天 2023-04-21 - 2023-04-28

全选 (已选 0 / 2)

涉暴恐 疑似

涉黄 疑似

2) 任务详情: 单击对应任务的“任务详情”, 该页面展示其基础信息和审核信息。



The screenshot displays a task details page with the following components:

- Buttons:** 查看结果 (View Results), 暂停 (Pause), 删除 (Delete).
- 基础信息 (Basic Information):**
 - 任务名称 (Task Name): [Redacted]
 - 地域 (Region): CN-BJ
 - 任务状态 (Task Status): 进行中 (In Progress)
 - 创建时间 (Creation Time): 2024-28
- 审核信息 (Review Information):**
 - 存储空间 (Storage Space): 31
 - 任务类型 (Task Type): 增量审核 (Incremental Review)
 - 审核内容 (Review Content): 图片 (Image)
 - 规则范围 (Rule Scope): 所有文件 (All Files)
 - 图片格式 (Image Format): JPG、PNG、JPEG
 - 审核类型 (Review Type): 涉黄、涉政、涉暴恐 (Involving pornography, politics, terrorism)
 - 违规处理 (Violation Handling): 不处理 (No Action)

3) 暂停: 支持暂停和启动“进行中”的任务状态。

4) 删除: 支持删除单个任务和批量删除多个任务。

参数说明表

参数	说明
地域	目前开放地域是华北一、上海二、台北。其他地域有开通需求, 请联系技术支持。

存储空间	US3对象存储空间(简称存储空间)是文件的组织管理单位,一个文件必然隶属于某个空间中。 用户可以选择在US3已创建的存储空间,或者单击“新建存储空间”跳转至US3对象存储完成创建。
任务类型	- 增量审核:针对该审核任务创建完成后,新上传至该存储空间的图片进行审核; - 存量审核:针对已存储在存储空间的图片进行审核。
审核内容	目前仅支持图片审核。
规则范围	- 所有文件:针对US3存储空间中上传的所有图片进行处理; - 部分文件:针对US3存储空间中上传的文件名以指定前缀的图片进行处理。
图片格式	目前仅支持的图片格式为:JPG、PNG、JPEG。 必选项,至少需要选择一项。
审核类型	涉黄/涉政/涉暴恐。 必选项,至少需要选择一项,且每种类型单独计费。
违规处理	- 不处理:则审核结果会展示在结果页面中,由用户决定如何处理; - 迁移文件至:审核的违规图片会被移动到选择的US3存储空间的指定前缀下。
任务名称	支持输入长度为1-63位字符,支持中英文、数字以及- _

开放地域

其他地域有开通需求,请联系技术支持。

- 华北一

- 上海二
- 台北

计费规则

图片内容审核的计费场景共3个(涉黄、涉政、涉暴恐),支持后付费模式,检测多个场景时会累积计费,后付费按照当日计费场景累计扫描量对应的档位阶梯计费。

后付费定价

当日总扫描量	单价 (元/千张)
>0	1.5
>5000	1.43
>50000	1.36
>130000	1.25
>260000	1.20
>850000	1.15

1. 图片内容审核有多种类型,每种类型单独计费。如使用了涉黄、涉政,则上传1张图片后会计入2次计费;
2. 数据智能UDI提供每月2000张的免费使用额度。

常用工具

除使用 US3 管理控制台之外,您还可以通过 US3 提供的各类常用工具对 US3 存储空间及空间内的文件对象进行管理,或帮助您接入各类场景,实现数据在云端的快速读写。

US3 产品底层升级后,推出 **us3cli**、**us3fs** 工具,便于用户对存储空间、文件进行管理操作,操作方式更灵活。推出 **us3hadoop** 大数据适配工具,便于各类大数据计算框架访问及读写 **US3** 上的数据。

管理工具

工具	说明
US3CLI 命令行工具	统一管理工具 US3CLI 可帮助用户进行 US3 存储空间和文件的统一管理,支持丰富的存储空间以及文件对象操作。
US3FS 文件挂载工具	文件系统挂载工具 US3FS 可帮助用户在 Linux 环境下以 POSIX 协议方式挂载文件目录,对文件对象进行操作。
US3SYNC 迁移工具	迁移工具 US3SYNC 可帮助用户将数据迁移至 US3 的存储空间中。
US3Hadoop 大数据适配工具	大数据适配工具 US3Hadoop 可帮助用户在 Hadoop 中以文件系统的方式访问 US3。

工具简介

概述

US3CLI 是一款通过命令行操作对 US3 进行数据管理的工具,支持丰富的存储空间(bucket)以及对象(object)操作,支持Windows、Linux、Mac平台。

通过本工具,您可以进行以下操作: 存储空间(bucket):创建、删除、查看存储量、查看存储空间列表等。对象(object):上传、下载、删除、移动、拉列表、计算 etag 等。

另外,本工具提供丰富的批量操作:上传,下载,删除,拷贝(不同存储空间之间),移动(同一存储空间内),数据取回等。

版本和运行环境

软件版本

当前版本:1.7.2

运行环境

- Windows(x86 32/64位)
- Linux(x86 32/64位, arm 32/64位)
- Mac(64位)

下载与环境准备

下载地址

I.Windows

- us3cli-windows.exe

II.Linux

x86架构

- us3cli-linux32
- us3cli-linux64

arm架构

- us3cli-arm32
- us3cli-arm64

III.Mac

- us3cli-mac

环境准备

Linux环境（Linux 64位版本）

1. 下载工具至本地。
2. 添加可执行权限。

```
chmod +x us3cli-linux64
```

3. 开始创建配置。

```
./us3cli-linux64 config
```

Windows环境

1. 下载工具至本地。
2. 以管理员身份打开cmd面板, 切换到us3cli-windows.exe文件所在路径。
3. 执行config命令创建配置。

```
us3cli-windows.exe config
```

注意事项:

1. 请使用管理员身份打开cmd命令行。

2. 下载后请不要直接点击执行us3cli-windows.exe文件, 该工具需要使用命令行工具打开执行。
3. 如果点击下载无反应, 请右键单击复制链接地址, 输入到浏览器地址输入框打开。

Mac环境

1. 下载工具至本地。
2. 添加可执行权限。

```
chmod +x us3cli-mac
```

3. 开始创建配置。

```
./us3cli-mac config
```

快速上手

- 视频教学
- 配置管理
- 创建存储空间
- 上传文件或文件夹
- 下载文件或文件夹
- 拷贝文件或文件夹
- 删除文件或文件夹
- 查询文件基本信息
- 文件基本信息修改

视频教学

观看以下视频,快速上手使用US3CLI

该内容为动态文件,无法展示

配置管理

此处主要介绍us3cli工具配置方式以及常见的使用场景,us3cli工具支持多种配置生成方式,对于不同配置项,支持配置项之间的切换、配置项删除、更新、查看。对于单个配置项,支持配置内

容加密(公私钥)、HTTPS、代理。

配置项:一个配置项代表一个不同权限的配置文件,便于不同操作权限区分管理

- 常用配置
 - 配置项创建
 - 配置项管理
- 临时配置

配置方式根据是否保存配置文件做区分,分为常用配置和临时配置,常用配置可进行更新、切换、查看等操作,临时配置作为参数在其他命令运行时输入,表示当前配置参数只在当前命令运行时生效。

常用配置

命令格式:

```
us3cli config [--ls][--su <配置名>][--rm <配置名>][--cat <配置名>][--encrypt][--ssl][--proxy [代理地址]]
[--accesskey <API/Token公钥>][--secretkey <API/Token私钥>][--endpoint <访问域名>]
[--language <ZH/EN>]
```

常用配置创建方式分为两种:交互式配置和一键配置,命令格式相同,使用方式不同

配置项创建

1. 交互式配置

使用示例:

```
#!/us3cli config
```

请输入语言(Please enter the language): ZH/EN, 默认为ZH(default is ZH) :ZH

请输入当前配置项名称: config1

开始创建新配置项:[config1]

是否启用配置加密(y or n) ? n

请输入API/Token公钥[当前:]: xxxxxxxxxxxxxxxxxxxxxxxxx

请输入API/Token私钥[当前:]: xxxxxxxxxxxxxxxxxxxxxxxxx

地区列表:

No. RegionName Region

0 北京 cn-bj

1 上海二 cn-sh2

2 广州 cn-gd

3 香港 hk

4 洛杉矶 us-ca

5 新加坡 sg

6 雅加达 idn-jakarta

7 台北 tw-tp

8 拉各斯 afr-nigeria

9 圣保罗 bra-saopaulo

10 迪拜 uae-dubai

11 法兰克福 ge-fra

12 胡志明市 vn-sng

13 华盛顿 us-ws

14 孟买 ind-mumbai

15 首尔 kr-seoul

16 东京 jpn-tyo

```
17 曼谷 th-bkk
18 华北二 cn-wlcb
19 英国 uk-london
请输入region编号: 0
内外网列表:
No. Network
0 外网
1 内网
请选择或输入内外网编号:0
您选择的endpoint是:[cn-bj.ufileos.com],[当前:],请输入回车确认或自定义endponit:
当前最终配置:
ConfigName: config1
AccessKey: xxxxxxxxxxxxxxxxxxxxxxxxx
SecretKey: xxxxxxxxxxxxxxxxxxxxxxxxx
Endpoint: cn-bj.ufileos.com
请检查后输入回车确认:
是否启用HTTPS (y or n) ? n
是否启用代理(y or n):n
配置文件 [ config1 ] 已修改
是否使用该配置作为默认配置(当前默认配置为:< config >)(y or n)?
```

2. 一键配置

使用示例:

```
./us3cli config config1 --accesskey TOKEN_13be86***** --secretkey BAtrQO8LYdgve1HS_benbK-MXNTI3***** --endpoint cn-bj.ufileos.com --language
```

ZH

配置项管理

#切换默认配置

```
./us3cli config --su config1
```

#删除指定配置(只支持指定配置项名称删除,不支持指定路径)

```
./us3cli config --rm config1
```

#配置项内容查看,包含accesskey、secretkey、endpoint

```
./us3cli config --cat config1
```

#查看当前所有配置项列表,默认配置会标明 "(default)",Authority表示当前配置是Token密钥类型还是API密钥类型(只判断类型,不进行权限验证)

```
./us3cli config --ls
```

临时配置

临时配置可以通过配置名、配置文件路径、配置项内容使用,具体示例如下:

#1.通过配置名指定临时配置

```
./us3cli ls us3://bucket1 --config config2
```

#2.通过配置文件路径指定临时配置

```
./us3cli ls us3://bucket1 --config ~/go/src/userconfig.yaml
```

#3.通过配置项内容直接指定临时配置

```
./us3cli ls us3://bucket1 --accesskey "xxxxxx" --secretkey "xxxxxx" --endpoint "xxxxxx"
```

注:当前版本支持自定义配置文件,但只支持与工具自动生成配置文件内容相同的配置文件

自定义配置文件内容如下,具体填写说明见常用命令页config命令介绍

```
accesskey: "user accesskey"  
secretkey: "user secretkey"  
endpoint: "ufile.cn-north-02.ucloud.cn"  
encrypt: "false"  
enablessl: "false"  
proxy: "http://ip:port or https://ip:port"  
language: "ZH"
```

创建存储空间

交互式创建

命令格式:

```
us3cli mb us3://<bucketname>
```

使用示例:

```
# ./us3cli mb us3://bucketTest
```

请输入要创建bucket的权限类型acl(private/public,默认为private):private

地区列表:

No. RegionName Region

0 北京 cn-bj

1 上海二 cn-sh2

2 广州 cn-gd

3 香港 hk

4 洛杉矶 us-ca

5 新加坡 sg

6 雅加达 idn-jakarta

7 台北 tw-tp

8 拉各斯 afr-nigeria

9 圣保罗 bra-saopaulo

10 迪拜 uae-dubai

11 法兰克福 ge-fra

12 胡志明市 vn-sng

13 华盛顿 us-ws

14 孟买 ind-mumbai

15 首尔 kr-seoul

16 东京 jpn-tyo

17 曼谷 th-bkk

18 华北二 cn-wlcb

19 英国 uk-london

请输入要创建bucket地区编号或地区代码(默认为北京:cn-bj):0

Region: cn-bj

当前账号下项目信息如下:

No. ProjectName ProjectId

1 Default org-orcwsj

请输入要创建bucket的项目编号:1

Number: 1

ProjectID: org-orcwsj

2020-11-24 17:52:56.973 INFO Make bucket [bucketTest] success

一键创建

#命令格式:

```
us3cli mb us3://<bucketname> --projectid <projectid> --region <region> --acl <acl>
```

#使用示例:

```
./us3cli mb us3://buckettest --projectid org-test --region cn-bj --acl public
```

上传文件或文件夹

上传单个文件

```
#命令格式:
#普通文件
us3cli cp <local filename> us3://<bucketname>/<us3key>
#流式文件
us3cli rcat us3://<bucketname>/<us3key>

#使用示例:
#普通上传文件, 不指定任何参数
./us3cli cp ./test.txt us3://buckettest/test.txt
#指定存储类型为IA上传 (不区分大小写)
./us3cli cp ./test.txt us3://buckettest/test.txt --storageclass IA
#指定并发数, 设置并发数为10(文件大于64MB时才会有效果, 因为64MB以上大文件才会使用分片上传)
./us3cli cp ./test.txt us3://buckettest/test.txt --parallel 10
#将本地文件test.txt cat到标准输入, 再上传至us3存储空间内
cat test.txt | ./us3cli rcat us3://buckettest/test.txt
#流式上传并指定失败重试次数为10, 设置并发数为10
cat test.txt | ./us3cli rcat us3://buckettest/test.txt --retrycount 10 --parallel 10
```

上传文件夹

```
#命令格式:
us3cli cp -r <localdir> us3://<bucketname>/<us3key>

#使用示例:
```

```
#普通上传文件夹
./us3cli cp -r ./testdir us3://buckettest/us3dir
#上传后缀名为".txt"的文件
./us3cli cp -r ./testdir us3://buckettest/us3dir --include "*.txt"
#上传文件名不包含test的文件
./us3cli cp -r ./testdir us3://buckettest/us3dir --exclude "*test*"
#上传文件夹并进行完整性校验
./us3cli cp -r ./testdir us3://buckettest/us3dir --check
```

增量上传文件夹

增量上传文件夹:对比本地文件夹和us3对应的文件夹,忽略已上传的文件,将未上传过的文件上传到us3文件夹内

```
#命令格式:
us3cli sync <localdir> us3://<bucketname>/<us3key>

#使用示例:
#遍历本地文件夹,以本地缓存为标准进行同步,如果文件修改时间晚于本地缓存保存的时间,或者未保存过,则上传文件,否则跳过
./us3cli sync ./testdir us3://buckettest/us3dir

#遍历本地文件夹,以本地缓存为标准进行同步,如果文件etag和本地缓存保存的etag不同,则上传文件,否则跳过
./us3cli sync ./testdir us3://buckettest/us3dir --ruler etag

#遍历本地文件夹,对比本地和us3所有文件,如果文件修改时间晚于us3文件的修改时间,则上传文件,否则跳过
#如果存在文件在us3的目录下存在,但在本地不存在,则删除us3中的该文件(该删除操作会默认询问,目前不提供强制删除功能)
```

```
./us3cli sync ./testdir us3://buckettest/us3dir --mode local
```

#遍历本地文件夹,对比本地和us3所有文件,如果文件etag和us3同一文件的etag不同,则上传文件,否则跳过

```
./us3cli sync ./testdir us3://buckettest/us3dir --mode local --ruler etag
```

#增量上传并指定存储类型为低频类型

```
./us3cli sync ./testdir us3://buckettest/us3dir --storageclass "IA"
```

#增量上传并指定mimetype

```
./us3cli sync ./testdir us3://buckettest/us3dir --mimetype "mimetype1"
```

下载文件或文件夹

下载单个文件

#命令格式:

#普通下载

```
us3cli cp us3://<bucketname>/<us3key> <local filename>
```

#流式下载

```
us3cli cat us3://<bucketname>/<us3key>
```

#使用示例:

#普通下载单个文件

```
./us3cli cp us3://buckettest/test.txt ./test.txt
```

#下载文件,每个分片大小指定为8M(不区分大小写,并且默认大小为4M,最小值为4M)

```
./us3cli cp us3://buckettest/test.txt ./test.txt --partsize 4M
```

#流式文件下载(文件会写入到标准输入)

```
./us3cli cat us3://buckettest/test.txt
```

#流式下载并指定并发数,重试次数为10

```
./us3cli cat us3://buckettest/test.txt --parallel 10 --retrycount 10
```

下载文件夹

#命令格式:

```
us3cli cp -r us3://<bucketname>/<us3key> <localdir>
```

#使用示例:**#普通下载文件夹**

```
./us3cli cp -r us3://buckettest/us3dir ./testdir
```

#下载文件夹并指定并发数为20

```
./us3cli cp -r us3://buckettest/us3dir ./testdir --parallel 20
```

```
# 下载文件夹并限速为100MB/s
./us3cli cp -r us3://buckettest/us3dir ./testdir --speedlimit 100MB
```

拷贝文件或文件夹

```
# 命令格式
us3cli cp us3://<bucketname>/<us3key>

# 使用示例
# 将文件从bucket1拷贝至bucket2中(两个bucket必须在同一地域中, 如需要不同地域的数据复制, 则需要使用跨区域复制功能)
./us3cli cp us3://bucket1/test.txt us3://bucket2/test.txt

# 将文件夹从bucket1拷贝到bucket2中
./us3cli cp -r us3://bucket1/test us3://bucket2/test

# 将bucket1中test文件夹中以".txt"结尾的文件拷贝到bucket2的test文件夹中
./us3cli cp -r us3://bucket1/test us3://bucket2/test --include "*.txt"
```

删除文件或文件夹

删除单个文件

#命令格式:

```
us3cli rm us3://<bucketname>/<us3key>
```

#使用示例:

#删除文件

```
./us3cli rm us3://buckettest/test.txt
```

#强制删除

```
./us3cli rm -f us3://buckettest/test.txt
```

删除文件夹

#命令格式:

```
us3cli rm -r us3://<bucketname>/<us3key>
```

#使用示例:

#删除文件夹

```
./us3cli rm -r us3://buckettest/test
```

#强制删除整个test文件夹

```
./us3cli rm -r -f us3://buckettest/test
```

#删除test文件夹下不含有test的文件

```
./us3cli rm -r -f us3://buckettest/test --exclude "*test"
```

#设置并发数为10,强制清空当前buckettest存储空间

```
./us3cli rm -r -f us3://buckettest --parallel 10
```

#删除文件夹,限制请求数量为每秒10个

```
./us3cli rm -r -f us3://buckettest/test --qps 10
```

查询文件基本信息

#命令格式

```
us3cli stat us3://<bucketname>/<us3key>
```

#使用示例

```
./us3cli stat us3://buckettest/test.txt
```

获取文件列表

#命令格式

```
us3cli ls us3://<bucketname>[/us3key]
```

#使用示例

```
./us3cli ls us3://buckettest
```

#拉取buckettest中的文件, 只显示10个

```
./us3cli ls us3://buckettest --limit 10
```

#拉取buckettest中的文件, 并以非目录形式展示

#非目录形式:目录下所有的文件都显示完整路径, 子目录下的文件也会列出

```
./us3cli ls us3://buckettest --flat
```

#列取buckettest中的文件, 并展示是否已取回, 以及取回时间

#数据取回:指归档类型的数据短暂激活, 可以下载

```
./us3cli ls us3://buckettest --restore
```

文件基本信息修改

#命令格式

```
us3cli modify us3://<bucketname>/us3key
```

#使用示例

#修改文件mimetype为xxx/yyy

```
./us3cli modify us3://buckettest/test.txt --mimetype xxx/yyy
```

```
#为文件增加metadata key为 "name" value为 "us3cli"
./us3cli modify us3://buckettest/test.txt --metadata name=us3cli

#清空当前文件的metadata
./us3cli modify us3://buckettest/test.txt --metadata "" --replace

#修改文件的存储类型为ARCHIVE(归档类型)
./us3cli modify us3://buckettest/test.txt --storageclass ARCHIVE
```

常用命令

操作	命令	描述
配置管理	config	管理上传所需的公私钥及endpoint等信息,包括配置项的创建、修改、删除、更新、切换
创建存储空间	mb	创建存储空间
删除存储空间	rb	删除存储空间(存储空间为空)
查看存储空间信息	stat	查看bucket的元数据信息
查看存储量	du	查看bucket的存储量(标准、低频、归档)
普通上传	cp	将本地文件或目录上传到存储空间中
增量上传	sync	增量上传目录到存储空间中
流式上传	rcat	将流式文件上传到存储空间中
创建目录	mkdir	在us3存储空间中创建一个空目录
普通下载	cp	将存储空间中的文件或目录下载到本地
流式下载	cat	将存储空间中的数据下载并写入到标准输入中
拷贝	cp	将一个存储空间中的文件拷贝到另一个存储空间中(同一地域)
移动	mv	将文件或目录移动到其他目录下(同一存储空间内)
删除	rm	删除存储空间中的文件或目录

列表	ls	列出US3存储空间列表或US3存储空间中的文件列表
获取下载URL	sign	获取存储空间中文件的下载链接
查看元数据	stat	查看存储空间中文件的元数据信息
修改元数据	modify	修改存储空间中文件的存储类型、MIMETYPE、MetaData
归档数据取回	restore	将归档类型的文件激活为可下载状态
数据完整性校验	etag	查看本地文件、标准输出、US3存储空间的文件etag
创建token	create-token	创建一个用于操作US3的token
删除token	delete-token	删除一个用于操作US3的token
更新token	update-token	更新一个用于操作US3的token
描述token	describe-token	列取并描述操作US3的token
版本更新	update	更新工具版本
版本特性	version	查看工具版本特性

config

config命令用于管理配置文件。

命令格式

```
us3cli config [--ls][--su <配置名>][--rm <配置名>][--cat <配置名>][--encrypt][--ssl][--proxy [代理地址]]
```

```
[--accesskey <API/Token公钥>][--secretkey <API/Token私钥>][--endpoint <访问域名>]  
[--language <ZH/EN>]
```

参数说明

-a, --accesskey string :用于访问us3的API密钥或Token公钥
--cat string :打印指定配置项内容
--encrypt :是否配置加密
--language :us3cli的语言配置,ZH/EN,可选,默认为ZH中文
-e, --endpoint string :固定域名,可在地域和域名页查看
-h, --help :查看当前命令帮助
--ls :列出当前所有配置项
--proxy string :代理地址(ip:port)
--rm string :删除指定配置项
-s, --secretkey string :用于访问us3的API私钥或Token私钥
--ssl :使用https
--su string :切换指定配置为默认配置

配置文件内容说明:

配置项	说明	填写说明
AccessKey	用于鉴权的bucket公钥	API公钥、Token公钥
SecretKey	用于鉴权的bucket私钥	API私钥、Token私钥

Endpoint	外网或内网域名	地域和域名
encrypt	是否使用配置加密	false 或 true
enablesl	是否使用https	false 或 true
proxy	代理地址	"ip:port"
language	工具显示语言	ZH 或 EN

自定义配置文件格式, 填写说明同上:

```
accesskey: "user accesskey"  
secretkey: "user secretkey"  
endpoint: "ufile.cn-north-02.ucloud.cn"  
encrypt: "false"  
enablesl: "false"  
proxy: "http://ip:port or https://ip:port"  
language: "EN"
```

使用示例

1. 交互式配置

- 创建配置项

```
#!/usr3cli config
```

```
请输入语言(Please enter the language): ZH/EN, 默认为ZH(default is ZH) : ZH
```

请输入当前配置项名称: config1

开始创建新配置项:[config1]

是否启用配置加密(y or n) ? n

请输入API/Token公钥[当前:]: xxxxxxxxxxxxxxxxxxxxxxxxx

请输入API/Token私钥[当前:]: xxxxxxxxxxxxxxxxxxxxxxxxx

地区列表:

No. RegionName Region

0 华北一 cn-bj

1 上海二 cn-sh2

2 广州 cn-gd

3 香港 hk

4 洛杉矶 us-ca

5 新加坡 sg

6 雅加达 idn-jakarta

7 台北 tw-tp

8 拉各斯 afr-nigeria

9 圣保罗 bra-saopaulo

10 迪拜 uae-dubai

11 法兰克福 ge-fra

12 胡志明市 vn-sng

13 华盛顿 us-ws

14 孟买 ind-mumbai

15 首尔 kr-seoul

16 东京 jpn-tyo

17 曼谷 th-bkk

```
18 华北二 cn-wlcb
19 英国 uk-london
请输入region编号: 0
内外网列表:
No. Network
0 外网
1 内网
请选择或输入内外网编号:0
您选择的endpoint是:[cn-bj.ufileos.com],[当前:],请输入回车确认或自定义endponit:
当前最终配置:
ConfigName: config1
AccessKey: xxxxxxxxxxxxxxxxxxxxxxxx
SecretKey: xxxxxxxxxxxxxxxxxxxxxxxx
Endpoint: cn-bj.ufileos.com
请检查后输入回车确认:
是否启用HTTPS (y or n) ? n
是否启用代理(y or n):n
配置文件 [ config1 ] 已修改
是否使用该配置作为默认配置(当前默认配置为:< config >)(y or n)?
```

注意:

1. 首次创建的配置文件时会自动将该配置作为默认配置
 2. 配置加密只加密到公私钥,并且当前配置文件只在初次创建时可选是否加密
 3. 填写代理地址时只需要填写“ ip:port”,客户端会根据填写的https启用状态补充需要的“https://”或“http://”头部信息
- 列出配置项列表

```
./us3cli config --ls
```

```
ConfigName ModTime FilePath Authority
config1 (Default) 2020-09-21 14:18:50 /root/.us3cliconfig/config1 Token
config2 2020-09-21 14:18:50 /root/.us3cliconfig/config2 Token
us3cli 2020-09-16 10:36:00 /root/.us3cliconfig/us3cli APIKey
```

说明:

- 1.Default标识表示该配置项是当前的默认配置
- 2.Authority表示权限分类,只用于快速区分Token和API密钥格式,不保证内容准确

- 切换配置项

```
./us3cli config --su config2
```

- 删除配置项

```
./us3cli config --rm config1
```

注意:以下所有命令的(y or n)选项规则均不区分大小写,输入yes或y表示确认,其他选项均表示取消

- 打印配置项

```
./us3cli config --cat config2
ConfigName:config2
```

```
AccessKey: TOKEN_13be86*****  
SecretKey: BAtRQO8LYdgve1HS_benbK-MXNTI3*****  
Endpoint: cn-bj.ufileos.com  
Language: ZH
```

2.非交互式配置

```
./us3cli config config3 --accesskey TOKEN_AAGASGAZVZV**** --secretkey USAsflmTAAF***** --endpoint cn-bj.ufileos.com --language EN  
Configuration file [ config3 ] has been updated
```

3.临时使用(对其他命令生效)

- 上传文件时临时使用配置项config3

```
./us3cli cp test.txt us3://bucket1 --config config3
```

- 上传文件时临时使用配置文件 /home/ubuntu/myconfig1

```
./us3cli cp test.txt us3://bucket1 --config /home/ubuntu/myconfig1
```

- 上传文件时使用自定义配置内容

```
./us3cli cp test.txt us3://bucket1 --accesskey LTAI4G3t3BTza47xxxxxxxxxx --secretkey gznFs9daMtKmUaTq9xpxxxxxxxxxxxxxx --endpoint cn-bj.ufileos.com
```

mb

该命令用于创建存储空间

命令格式

```
us3cli mb us3://<桶名字> [--acl <权限类型>][--region <桶所在地区>][--projectid <项目ID>]
```

参数说明

```
--accesskey <string> :用于访问us3的API公钥或Token公钥  
-a, --acl <string> :权限类型,可以设置为private、public,默认为private私有(不区分大小写)  
--config <string> :当前命令临时指定配置名/配置文件路径  
--endpoint <string> :固定域名,可通过地域和域名页查看,如:cn-bj.ufileos.com  
-h, --help :当前命令使用说明  
--projectid <string> :项目ID,当前bucket属项目ID,默认为Default  
-r, --region <string> :桶所在地区,可查看地域信息,默认地区为华北一  
--secretkey <string> :用于访问us3的API私钥或Token私钥
```

本命令提供命令输入和交互式输入二选一的操作,命令输入参数,就会自动跳过交互式输入。

使用示例

- 交互式创建bucket

```
./us3cli mb us3://us3cli-test
```

请输入要创建bucket的权限类型acl(private/public,默认为private):public

地区列表:

No. RegionName Region

0 华北一 cn-bj

1 上海二 cn-sh2

2 广州 cn-gd

3 香港 hk

4 洛杉矶 us-ca

5 新加坡 sg

6 雅加达 idn-jakarta

7 台北 tw-tp

8 拉各斯 afr-nigeria

9 圣保罗 bra-saopaulo

10 迪拜 uae-dubai

11 法兰克福 ge-fra

12 胡志明市 vn-sng

13 华盛顿 us-ws

14 孟买 ind-mumbai

15 首尔 kr-seoul

16 东京 jpn-ky

17 曼谷 th-bkk

18 华北二 cn-wlcb

19 英国 uk-london

```
请输入要创建bucket地区编号或地区代码(默认为华北一:cn-bj):0
```

```
Region: cn-bj
```

```
当前账号下业务组分组信息如下:
```

```
No. ProjectName ProjectId
```

```
1 Default org-orcwsj
```

```
请输入要bucket的项目编号:1
```

```
Number: 1
```

```
ProjectID: org-orcwsj
```

```
Make bucket [ us3cli-test ] success
```

- 非交互式创建bucket,输入acl,region以及projectid信息,acl,region是必填项,projectid可不填

```
./us3cli mb us3://us3cli-test --acl private --region cn-bj --projectid org-orcwsj
```

注意:若您的bucket不在默认项目下,bucket操作都需要加上--projectid参数,请创建bucket时自行记录projectid,或通过ls 命令查看projectid

rb

本命令用于删除存储空间

命令格式

```
us3cli rb us3://<桶名字> [--projectid <项目ID>][--force]
```

参数说明

```
--accesskey <string> :用于访问us3的API公钥或Token公钥
--config <string> :当前命令临时指定配置名/配置文件路径
--endpoint <string> :固定域名,可通过地域和域名页查看
-f, --force :强制删除
-h, --help :当前命令使用说明
--projectid <string> :指定当前bucket所在projectid,可以在控制台查看,默认项目可忽略
--secretkey <string> :用于访问us3的API私钥或Token私钥
```

使用示例

- 删除存储空间:存储空间必须为空,否则无法删除

```
./us3cli rb us3://bucket1
The bucket [bucket1] is being deleted, continue(y or n)? y
Delete bucket [bucket1] success
```

stat

本命令用于查看存储空间或文件信息

命令格式

```
us3cli stat us3://<桶名字>[/<文件key>][--projectid <项目ID>]
```

参数说明

```
--accesskey <string> :用于访问us3的API公钥或Token公钥  
--config <string> :当前命令临时指定配置名/配置文件路径  
--endpoint <string> :固定域名,可通过地域和域名页查看  
-h, --help :当前命令使用说明  
--projectid <string> :指定当前bucket所在projectid,可以在控制台查看,默认项目可忽略
```

使用示例

- 查看默认项目下存储空间bucket1的基本信息

```
./us3cli stat us3://bucket1
```

```
BucketName: bucket1
```

```
Region: cn-bj
```

```
BucketId: ufile-dpgjzcn
```

```
Type: public
```

```
CreateTime: 2020-09-15 18:17:24
```

```
ModifyTime: 2020-09-15 18:17:24
```

- 查看us3://bucket1/test.txt的文件信息

```
./us3cli stat us3://bucket1/aaa.txt
```

Name: aaa.txt

X-Ufile-Create-Time: Fri, 18 Sep 2020 10:09:05 GMT

X-Ufile-Storage-Class: STANDARD

Server: nginx/1.11.1

Date: Mon, 21 Sep 2020 11:17:56 GMT

Content-Type: application/octet-stream

Accept-Ranges: bytes

Etag: "AQAAAEpjpDD8COEdGg3uOeLfsR_ddQgc"

Content-Length: 4298

Last-Modified: Fri, 18 Sep 2020 10:09:05 GMT

Vary: Origin

du

该命令用于获取指定存储空间(Bucket)各存储类型的存储量大小,以及总存储量

注意:存储量结算时间为当前日期前一天0:00

命令格式

```
us3cli du us3://<桶名字>[/<key>] [--force][--projectid <项目ID>]
```

参数说明

```
--accesskey <string> :用于访问us3的API公钥或Token公钥
--config <string> :当前命令临时指定配置名/配置文件路径
--endpoint <string> :固定域名,可通过地域和域名页查看
-f, --force :强制更新存储量
-h, --help :当前命令使用说明
--projectid <string> :指定当前bucket所在projectid,可以在控制台查看,默认项目可忽略
```

使用示例

- 查看bucket1所占空间大小,内容包含各存储类型的存储量以及总存储量(在弹出的提示中选择N,需要使用API密钥且传入参数--projectid)

```
StorageClass Size
STANDARD: 1.05 GB
IA: 0 GB
ARCHIVE: 0 GB
Total storage: 1.05 GB
```

- 查看bucket1下的各类型存储量以及文件数量(在弹出的提示中选择Y)

```
FileCount FileSize
Sum 87 1.05 GB
STANDARD 87 1.05 GB
IA 0 0 B
```

```
ARCHIVE 0 0 B
```

- 查看bucket1下的testfolder目录下各类型存储量以及文件数量

```
#!/usr3cli du us3://bucket1/testfolder
```

```
FileCount FileSize
```

```
Sum 10000 99.77 MB
```

```
STANDARD 10000 99.77 MB
```

```
IA 0 0 B
```

```
ARCHIVE 0 0 B
```

```
UpdateTime: 2020-12-24 11:40:27 Thu
```

注意:当使用命令首次查看某个目录或前缀下的存储量时,会将数据保存在本地,24内使用非强制更新查看存储量时,都不会重新请求,只显示前24小时内拉取到的最新数据。有更新需要的可以添加-f/--force选项再次拉取存储量。

cp

该命令用于上传、下载、拷贝文件

命令格式

上传文件

```
us3cli cp <本地文件路径> us3://<桶名字>/<文件Key> [--check][--recursive][--reduce][--parallel <分片上传并发数>][--speedlimit <速度限制>][--storageclass <存
```

```
储类型>][--exclude <通配符表达式>][--rexclude <正则表达式>][--include <通配符表达式>][--rinclude <正则表达式>][--metadata <Key>=<value1>[,<key2>=<value2>]...][--mimetype <多媒体文件格式>]
```

下载文件

```
us3cli cp us3://<桶名字>/<文件Key> <本地文件路径> [--recursive][--reduce][--speedlimit <速度限制>][--exclude <通配符表达式>][--rexclude <正则表达式>][--include <通配符表达式>][--rinclude <正则表达式>][--parallel <分片下载并发数>][--partsize <下载分片大小>]
```

拷贝文件

```
us3cli cp us3://<桶名字>/<文件Key> us3://<桶名字>/<文件Key> [--recursive][--reduce][--exclude <通配符表达式>][--rexclude <正则表达式>][--include <通配符表达式>][--rinclude <正则表达式>][--metadata <Key>=<value1>[,<key2>=<value2>]...]
```

参数说明

```
--accesskey <string> :用于访问us3的API公钥或Token公钥  
-c, --check :用于上传文件夹结束后检查已上传文件的etag是否一致  
--config <string> :当前命令临时指定配置名/配置文件路径  
--endpoint <string> :固定域名,可通过地域和域名页查看  
--exclude <string> :不包含当前通配符的文件名  
-h, --help :当前命令使用说明  
--include <string> :包含当前通配符的文件名  
--metadata <string> :指定元数据信息(该参数仅限上传和拷贝) 多个元数据以","分隔,如 "key1=value,key2=value2",其他分隔符暂不支持  
--mimetype <string> :指定mimetype(该参数仅限上传)
```

```
--parallel <int> :分片并发数或批量并发数
--partsize <string> :下载分片大小,默认大小为4M,最小值为1M(该参数仅限下载)
-r, --recursive :递归文件夹中的所有文件及子目录下所有文件
--reduce :是否以精简模式执行(不展示进度)
--reexclude <string> :不包含当前正则表达式的文件名
--rinclue <string> :包含当前正则表达式的文件名
--secretkey <string> :用于访问us3的API私钥或Token私钥
-s, --speedlimit <string> :平均速度限制(单位可以是B,KB,MB,不带单位默认以B/s计算),默认200MB/s
--storageclass <string> :指定存储类型,对应有有效值:STANDARD, IA, ARCHIVE(该参数仅限上传),默认值:STANDARD
```

注意:

1.通配符表达式暂时只支持"*","?"两种字符,并且需要注意的是,四种表达式筛选均以当前目录下文件路径为准

如:us3://us3cli/test 目录下的test2/test3.txt 会以test2/test3.txt作为字符串筛选,而不是以test3.txt作为字符串进行筛选

2.以下所有speedlimit选项均描述为平均速度

3.单个文件上传时根据文件大小使用不同的请求,文件大于64MB使用分片上传,小于64MB时使用普通的put请求,故上传中的parallel参数仅限上传64MB以上大文件使用

4.由于windows环境下文件名存在不区分大小写的情况,要下载的文件已存在时,会在文件名及文件后缀名之间默认添加(1),(2)...等作为文件名后缀,如:

文件test.txt存在,则下载的文件保存为test(1).txt,若此时test(1).txt也存在,也保存为test(2).txt,依次类推

使用示例

- 上传单个文件

```
./us3cli cp ~/go/src/test.txt us3://bucket1/test
```

- 上传单个大文件(大于64MB)并设置分片并发数为10

```
./us3cli cp ~/go/src/test.mp4 us3://bucket1/test.mp4 --parallel 10  
# 文件大小小于64MB时该参数无效
```

- 下载单个文件

```
./us3cli cp us3://bucket1/test/test.txt ~/go/src/test.txt
```

- 拷贝单个文件

```
./us3cli cp us3://bucket1/test.txt us3://bucket2/test.txt
```

- 拷贝文件夹

```
./us3cli cp -r us3://bucket1/test us3://bucket2/test
```

- 下载文件夹

```
./us3cli cp -r us3://bucket/test ~/go/src/test
```

- 指定存储类型上传

上传单个文件并指定存储类型为IA(低频访问)类型

```
./us3cli cp ~/go/src/test.txt us3://bucket/path/test.txt --storageclass IA
```

- 限速上传

上传文件test.txt, 并设置速度为1024Kb/s

```
./us3cli cp ~/go/src/test.txt us3://bucket/test.txt --speedlimit 1024Kb
```

- 批量上传

上传所有格式为jpg的文件(通配符)

```
./us3cli cp -r ~/go/src/test us3://bucket/test --include "*.jpg"
```

上传所有a开头b结尾的文件(正则表达式)

```
./us3cli cp -r ~/go/src/test us3://bucket/test --rinclude "a*b"
```

上传所有不包括a开头b结尾的文件(正则表达式)

```
./us3cli cp -r ~/go/src/test us3://bucket/test --rexclude "a*b"
```

上传所有文件名不包括001的文件(通配符)

```
./us3cli cp -r ~/go/src/test us3://bucket/test --exclude "*001*"
```

- 上传单个文件并指定元数据信息

```
./us3cli cp ~/go/src/test us3://bucket/test --metadata key1=value1,key2=value2
```

- 指定mimetype为text/plain上传

```
./us3cli cp ~/go/src/test us3://bucket/test --mimetype text/plain
```

- 上传文件夹并检查etag是否一致

```
./us3cli cp -r ~/go/src/test us3://bucket/test --check
```

sync

本命令用于目录的增量上传或增量下载

增量上传

命令格式

```
us3cli sync <本地目录> us3://<桶名字>/<文件Key> [--reduce][--mode cache|local][--ruler modtime|etag][--speedlimit <速度限制>][--retrycount <重试次数>][--exclude <通配符表达式>][--rexclude <正则表达式>][--include <通配符表达式>][--rinclude <正则表达式>][--parallel <请求并发数>][--metadata <Key>=<value1>[,<key2>=<value2>]...][--mimetype <多媒体文件格式>][--storageclass <存储类型>][--force]
```

参数说明

```
--accesskey <string> :用于访问us3的API公钥或Token公钥
--config <string> :当前命令临时指定配置名/配置文件路径
--endpoint <string> :固定域名,可通过地域和域名页查看
--exclude <string> :不包含当前通配符的文件名
-f, --force :是否强制同步,在加入该选项后,同步删除时,不弹出确认信息
--no-delete :是否默认不删除源端不存在的对象,且不弹出信息,在与-f同时使用时,执行--no-delete
-h, --help :当前命令使用说明
--include <string> :包含当前通配符的文件名
--metadata <string> :指定元数据信息,多个元数据以","分隔,如 "key1=value,key2=value2",其他分隔符暂不支持
--mimetype <string> :指定mimetype上传
-m, --mode <string> :可选项 cache、local 默认值为cache
cache :以本地缓存为标准,检查基于缓存的增量文件,同步us3端对应目录的文件,默认为该模式
local :以本地文件系统为标准,检查本地文件以及us3不同步的文件,补全或删除us3端对应目录的文件
--parallel <int> :请求并发数,默认值为10
--reduce :是否以精简模式执行(不展示进度)
-r, --retrycount: <int> :失败重试次数,默认值:10
--rexclude <string> :不包含当前正则表达式的文件名
--rinclude <string> :包含当前正则表达式的文件名
--ruler <string> :可选项 modtime、etag 默认为modtime
modtime :在判断是否上传时采用文件最后修改时间作为判断标准,如果本地文件最后修改时间晚于us3,则进行上传请求,否则不上传
etag :在判断是否上传时采用文件etag作为判断标准,如果本地文件etag和us3中的etag不同,则进行上传请求,否则不上传
--secretkey <string> :用于访问us3的API私钥或Token私钥
-s, --speedlimit <string> :平均速度限制(单位可以是B,KB,MB,不带单位默认以B/s计算),默认200MB/s
--storageclass <string> :指定存储类型,对应有有效值:STANDARD, IA, ARCHIVE,默认值:STANDARD
```

增量模式说明:

1.cache模式使用本地缓存,从本地上传到bucket成功的文件,都会被记录为上传成功文件,如果需要重新上传,可以选择删除当前用户目录下的.us3cliconfig/leveldb文件夹,使用命令时会自动创建新的文件夹。

2.local模式下最终以本地文件为标准,保证bucket中的目标文件夹和本地同步,以下场景会进行文件删除,请慎用:

增量上传文件夹后,将本地文件删除,再次使用local模式增量,会将bucket中的文件删除以保持US3 Bucket和本地同步

使用示例

- cache模式,以本地缓存中保存的修改时间为标准,同步本地文件

```
./us3cli sync /root/test us3://bucket/path
```

- cache模式,以本地缓存中的文件etag信息为标准,同步本地文件

```
./us3cli sync /root/test us3://bucket/path --ruler etag
```

- local模式,对比本地文件夹与指定us3目录中的所有文件的修改时间,通过上传文件以及删除us3文件操作,将us3目录状态与本地保持一致(如us3中有相对于本地多余的文件,则会删除us3中对应多余文件,请慎用)

```
./us3cli sync /root/test us3://bucket/path --mode local
```

- local模式,对比本地文件夹与指定us3目录中的所有文件的etag信息,通过上传文件以及删除us3文件操作,将us3目录状态与本地保持一致(如us3中有相对于本地多余的文件,则会删除us3中对应多余文件,请慎用)

```
./us3cli sync /root/test us3://bucket/path --mode local --ruler etag
```

- 限速为1024 Kb/s上传

```
./us3cli sync /root/test us3://bucket/path --speedlimit 1024Kb
```

- 并发上传文件夹

```
./us3cli sync /root/test us3://bucket/path --parallel 5
```

- 指定mimetype上传

```
./us3cli sync /root/test us3://bucket/path --mimetype "mimetype"
```

- 指定存储类型上传

```
./us3cli sync /root/test us3://bucket/path --storageclass "ARCHIVE"
```

- 指定元数据上传

```
./us3cli sync /root/test us3://bucket/path --metadata "key1=value1,key2=value2"
```

增量下载

命令格式

```
us3cli sync us3://<桶名字>/<文件Key> <本地目录> [--update][--ignore-existing][--delete][--backup-dir][--speedlimit <速度限制>][--retrycount <重试次数>][--exclude <通配符表达式>][--rexclude <正则表达式>][--include <通配符表达式>][--rinclude <正则表达式>][--parallel <请求并发数>][--metadata <Key>=<value1>
```

```
[,<key2>=<value2>]...][--mimetype <多媒体文件格式>][--storageclass <存储类型>][--force]
```

参数说明

--accesskey <string> :用于访问us3的API公钥或Token公钥

--config <string> :当前命令临时指定配置名/配置文件路径

--endpoint <string> :固定域名,可通过地域和域名页查看

--exclude <string> :不包含当前通配符的文件名

-f, --force :是否强制同步,在加入该选项后,同步删除时,不弹出确认信息

-h, --help :当前命令使用说明

--include <string> :包含当前通配符的文件名

--metadata <string> :指定元数据信息,多个元数据以","分隔,如 "key1=value,key2=value2",其他分隔符暂不支持

--mimetype <string> :指定mimetype上传

--parallel <int> :请求并发数,默认值为10

--reduce :是否以精简模式执行(不展示进度)

-r, --retrycount: <int> :失败重试次数,默认值:10

--rexclude <string> :不包含当前正则表达式的文件名

--rinclude <string> :包含当前正则表达式的文件名

--update :是否覆盖本地的同名文件,默认覆盖且覆盖前会询问

--ignore-existing :优先级高于--update和--force,只要设置了必然不会对本地文件进行覆盖

--delete :是否删除本地存在而us3中不存在的文件,需要和--backup-dir一起使用

--backup-dir :--delete删除的文件备份的本地目录

```
--secretkey <string> :用于访问us3的API私钥或Token私钥  
-s, --speedlimit <string> :平均速度限制(单位可以是B,KB,MB,不带单位默认以B/s计算),默认200MB/s  
--storageclass <string> :指定存储类型,对应有有效值:STANDARD, IA, ARCHIVE,默认值:STANDARD
```

增量模式说明: 远端同步到本地文件成功后,会在缓存中记录本地文件的修改时间和etag。加上--update选项后,对于本地存在的同名文件,会执行如下判断: 下次再从远端同步时,如果:
1)本地文件的etag和修改时间和缓存中记录的一致;2)本地文件的修改时间晚于缓存中记录的修改时间。上述两种情况,不会再从远端同步。否则,如果遇到已存在文件,会询问是否从远端覆盖(如果设置了--force,不会询问)

使用示例

- 默认,只要是同名文件都会覆盖,覆盖前会询问

```
./us3cli sync us3://bucket/path /root/test
```

- 推荐用法,同名文件会按照【增量模式说明】中逻辑,判断是否覆盖

```
./us3cli sync us3://bucket/path /root/test --update --force
```

- 对于本地存在而us3中不存在的文件,会从本地目录删除并备份到--backup-dir

```
./us3cli sync us3://bucket/path /root/test --update --force --delete --backup-dir /root/backdir
```

rcat

本命令用于流式上传文件

命令格式

```
us3cli rcat us3://<桶名字>/<文件Key> [--reduce][--retrycount <重试次数>][--speedlimit <速度限制>][--parallel <并发数限制>]
```

参数说明

```
--accesskey <string> :用于访问us3的API公钥或Token公钥  
--config <string> :当前命令临时指定配置名/配置文件路径  
--endpoint <string> :固定域名,可通过地域和域名页查看  
-h, --help :当前命令使用说明  
--parallel <int> :分片并发数  
--reduce :是否以精简模式执行(不展示进度)  
--retrycount: <int> :失败重试次数,默认值:10  
--secretkey <string> :用于访问us3的API私钥或Token私钥  
-s, --speedlimit <string> :平均速度限制(单位可以是B,KB,MB,不带单位默认以B/s计算),默认200MB/s
```

使用示例

- 上传流式数据到us3中

```
cat test.txt | ./us3cli rcat us3://bucket1/test.txt
```

- 流式上传文件test.txt并设置限速为 2MB/s

```
cat test.txt | ./us3cli rcat us3://bucket1/test.txt --speedlimit 2MB
```

- 流式上传文件test.txt并设置重试次数为5次

```
cat test.txt | ./us3cli rcat us3://bucket1/test.txt --retrycount 5
```

- 流式上传文件test.txt并设置并发数为2

```
cat test.txt | ./us3cli rcat us3://bucket1/test.txt --parallel 2
```

mkdir

本命令用于在US3存储空间内创建目录

命令格式

```
us3cli mkdir [--parents] us3://<桶名字>/<目录名>[/<目录名>]
```

参数说明

--accesskey <string> :用于访问us3的API公钥或Token公钥

--config <string> :当前命令临时指定配置名/配置文件路径

--endpoint <string> :固定域名,可通过地域和域名页查看

```
-h, --help :当前命令使用说明  
-p, --parents :用于创建多级目录  
--secretkey <string> :用于访问us3的API私钥或Token私钥
```

使用示例

- 创建目录

```
./us3cli mkdir us3://bucket/dir
```

- 创建多级目录

```
./us3cli mkdir -p us3://bucket/dir1/dir2
```

cat

本命令用于流式下载文件

命令格式

```
us3cli cat us3://<bucketname>/<keyname> [--reduce][--retrycount <重试次数>][--speedlimit <速度限制>][--partsize <下载分片大小>]
```

参数说明

```
--accesskey <string> :用于访问us3的API公钥或Token公钥
--config <string> :当前命令临时指定配置名/配置文件路径
--endpoint <string> :固定域名,可通过地域和域名页查看
-h, --help :当前命令使用说明
--partsize :下载分片大小,默认大小为4M,最小值为1M(该参数仅限下载)
--reduce :是否以精简模式执行(不展示进度)
--retrycount: <int> :失败重试次数,默认值:10
--secretkey <string> :用于访问us3的API私钥或Token私钥
-s, --speedlimit <string> :平均速度限制(单位可以是B,KB,MB,不带单位默认以B/s计算),默认200MB/s
```

使用示例

- 查看test.txt文件内容

```
./us3cli cat us3://bucket1/test.txt
```

- 流式下载test.txt到本地

```
./us3cli cat us3://bucket1/test.txt > test.txt
```

- 流式下载并限速为1MB/s

```
./us3cli cat us3://bucket1/test.txt --speedlimit 1MB > test.txt
```

- 流式下载并设置重试次数为5次

```
./us3cli cat us3://bucket1/test.txt --retrycount 5 > test.txt
```

- 下载并设置分片大小为2M

```
./us3cli cat us3://bucket1/test.txt --partsize 2M > test.txt
```

mv

本命令用于移动文件

注意:本命令只能用于相同bucket之内,同时-f选项只支持文件

命令格式

```
us3cli mv us3://<桶名字>/<文件Key> us3://<桶名字>/<文件Key> [--force][--reduce][--exclude <通配符表达式>][--rexclude <正则表达式>][--include <通配符表达式>][--rinclude <正则表达式>][--parallel <并发数>]
```

参数说明

--accesskey <string> :用于访问us3的API公钥或Token公钥
--config <string> :当前命令临时指定配置名/配置文件路径
--endpoint <string> :固定域名,可通过地域和域名页查看
--exclude <string> :不包含当前通配符的文件名
-f, --force :存在同名文件时是否覆盖

```
-h, --help :当前命令使用说明
--include <string> :包含当前通配符的文件名
--parallel <int> :并发数
--reduce :是否以精简模式执行(不展示进度)
--rexclude <string> :不包含当前正则表达式的文件名
--rinclude <string> :包含当前正则表达式的文件名
--secretkey <string> :用于访问us3的API私钥或Token私钥
```

使用示例

- 移动文件

```
./us3cli mv us3://bucket/test.txt us3://bucket/test/test.txt
```

- 强制移动文件

```
./us3cli mv us3://bucket/test.txt us3://bucket/test/test.txt -f
```

- 移动文件夹 注意:目标文件夹存在时,文件夹会被移动到目标文件夹子目录中

```
./us3cli mv us3://bucket/test us3://bucket/test2
```

rm

本命令用于删除目录或者对象

命令格式

```
us3cli rm us3://<桶名字>/<文件Key> [--recursive][--force][--reduce][--exclude <通配符表达式>][--rexclude <正则表达式>][--include <通配符表达式>][--rinclude <正则表达式>][--qps <每秒最大请求数量>]
```

参数说明

```
--accesskey <string> :用于访问us3的API公钥或Token公钥  
--cache <string> :要删除的缓存类型 : upload|download|sync|du  
--config <string> :当前命令临时指定配置名/配置文件路径  
--endpoint <string> :固定域名,可通过地域和域名页查看  
--exclude <string> :不包含当前通配符的文件名  
-f, --force :强制删除  
-h, --help :当前命令使用说明  
--include <string> :包含当前通配符的文件名  
--parallel <int> :批量删除并发数,默认值为10  
-q, --qps <int> :限制每秒请求数量,默认50qps,取值范围:1~1000  
-r, --recursive :递归文件夹中的所有文件及子目录下所有文件  
--reduce :是否以精简模式执行(不展示进度)  
--rexclude <string> :不包含当前正则表达式的文件名  
--rinclude <string> :包含当前正则表达式的文件名  
--secretkey <string> :用于访问us3的API私钥或Token私钥
```

使用示例

- 删除文件

```
./us3cli rm us3://bucket/test.txt
```

- 强制删除文件

```
./us3cli rm -f us3://bucket/test.txt
```

- 强制删除文件夹

```
./us3cli rm -r -f us3://bucket/test
```

- 批量删除test文件夹中后缀名为.jpg的文件

```
./us3cli rm -r us3://bucket/test --include "*.jpg"
```

- 限制删除操作最多每秒10个请求,最大可以上调到1000qps,取值范围在1~1000

```
./us3cli rm -r us3://bucket/test --include "*.jpg" --qps 10
```

- 批量删除文件并设置并发数为30

```
./us3cli rm -r us3://bucket/test --parallel 30
```

- 清空bucket

```
./us3cli rm -r -f us3://bucket --parallel 10
```

- 删除sync产生的缓存

```
./us3cli rm -f --cache sync
```

ls

该命令用于列出存储空间(Bucket)、对象(Object)或ProjectID

命令格式

```
us3cli ls [us3://<桶名称>] [--limit <输出限制数>][--restore][--flat][--mimetype][--etag][--project][--projectid <项目ID>]
```

参数说明

--accesskey <string> :用于访问us3的API公钥或Token公钥

--config <string> :当前命令临时指定配置名/配置文件路径

--endpoint <string> :固定域名,可通过地域和域名页查看

--etag :是否展示etag信息

--flat :是否以非层级结构(展开所有目录)显示,默认层级显示

-h, --help :当前命令使用说明

-l, --limit <int> :需要同时列出的最大文件条数,默认值为100

```
--mimetype :是否展示mimetype信息
--project :列出当前账户下所有项目及项目ID
--projectid <string> :列出bucket时指定projectid,默认项目可忽略
-r, --restore :是否展示数据解冻信息
--secretkey <string> :用于访问us3的API私钥或Token私钥
--prefix-file-list :调用专有云的列表服务接口
```

注意:

ls 列举文件列表默认情况修改为只显示文件key,文件大小,存储类型,创建时间,可以自行通过参数添加restore、etag、mimetype等信息

使用示例

- 列出当前权限下所有bucket

```
./us3cli ls
BucketName Region Acl CreateTime
bucket1 cn-bj public 2020-03-27 17:20:56
bucket2 cn-bj private 2020-03-27 17:20:43
us3cli cn-bj public 2020-09-15 16:17:24
```

- 列出bucket下所有对象

```
./us3cli ls us3://bucket
Key FileSize StorageClass Etag CreateTime
us3://bucket/cf1 DIR IA AQAAANo5o-5ea0sNMIW_75VgGJCv2Acj 2020-07-16 18:51:04
```

```
us3://bucket/aa.txt 4KB STANDARD AQAAGNifLt9clbFawzU5caZm7aDZkho 2020-07-22 11:04:34
us3://bucket/aa.txt 4KB ARCHIVE AQAAGNifLt9clbFawzU5caZm7aDZkho 2020-07-22 11:04:34
```

- 列出当前账号下所有project

```
./us3cli ls --project
```

当前账号下项目信息如下:

```
No. ProjectName ProjectId
```

```
1 Default org-ddhm4o
```

```
2 Test org-nhrc4y
```

```
3 Test2 org-wr5pej
```

- 列出bucket下对象并限制显示文件数量为2

```
./us3cli ls us3://bucket --limit 2
```

- 列出bucket下对象并显示数据解冻状态信息

```
./us3cli ls us3://bucket --restore
```

- 递归当前目录列出非目录形式的文件列表(展开所有目录)

```
./us3cli ls us3://bucket --flat
```

sign

本命令用于文件下载url获取

命令格式

```
us3cli sign us3://<桶名字>/<文件Key> [--expires <时间>]
```

参数说明

```
--accesskey <string> :用于访问us3的API公钥或Token公钥  
--config <string> :当前命令临时指定配置名/配置文件路径  
--endpoint <string> :固定域名,可通过地域和域名页查看  
--expires <int> :url使用的过期时间,单位s  
-h, --help :当前命令使用说明  
--secretkey <string> :用于访问us3的API私钥或Token私钥
```

使用示例

- 获取url签名

```
./us3cli sign us3://bucket/test.txt
```

- 获取限时100s 的url签名

```
./us3cli sign us3://bucket/test.txt --expires 100
```

modify

该命令用于修改us3中的文件信息

命令格式

```
us3cli modify us3://<桶名字>/<文件Key> [--storageclass <存储类型>][--mimetype <多媒体文件格式>][--metadata <<key1=value1>[,<key2=value2>]...>][--replace]
```

参数说明

--accesskey <string> :用于访问us3的API公钥或Token公钥
--config <string> :当前命令临时指定配置名/配置文件路径
--endpoint <string> :固定域名,可通过地域和域名页查看
-h, --help :当前命令使用说明
--metadata <string> :指定元数据信息
--mimetype <string> :指定多媒体文件格式mimetype
--replace :是否清空原有元数据
--secretkey <string> :用于访问us3的API私钥或Token私钥
--storageclass <string> :指定存储类型,对应有有效值:STANDARD, IA, ARCHIVE

使用示例

- 存储类型改变

修改test.txt的存储类型为STANDARD标准型

```
./us3cli modify us3://bucket/test.txt --storageclass IA
```

- mimetype改变

修改test.txt文件的mimetype信息

```
./us3cli modify us3://bucket/test.txt --mimetype image/x-icon
```

- metadata改变

1.修改test.txt文件的元数据信息,以key=value形式作为参数,可修改多个元数据信息,中间以英文逗号","分隔

```
./us3cli modify us3://bucket/test.txt --metadata "key1=value1,key2=value2"
```

2.清除原有元数据,添加新的内容

```
./us3cli modify us3://bucket/test.txt --metadata "key1=value1" --replace
```

3.删除所有元数据

```
./us3cli modify us3://bucket/test.txt --metadata "" --replace
```

restore

本命令用于恢复冷冻状态的对象(object)为可读状态

命令格式

```
us3cli restore us3://<桶名字>/<文件Key> [--recursive][--reduce][--qps <每秒请求数上限>][--exclude <通配符表达式>][--rexclude <正则表达式>][--include <通配符表达式>][--rinclude <正则表达式>][--parallel <批量激活并发数>]
```

参数说明

--accesskey <string> :用于访问us3的API公钥或Token公钥
--config <string> :当前命令临时指定配置名/配置文件路径
--endpoint <string> :固定域名,可通过地域和域名页查看
--exclude <string> :不包含当前通配符的文件名
-h, --help :当前命令使用说明
--include <string> :包含当前通配符的文件名
--parallel <int> :批量激活并发数,默认值为10
-q, --qps <int> :限制每秒请求数量,默认50qps,取值范围:1~1000
-r, --recursive :递归文件夹中的所有文件及子目录下所有文件
--reduce :是否以精简模式执行(不展示进度)
--rexclude <string> :不包含当前正则表达式的文件名
--rinclude <string> :包含当前正则表达式的文件名
--secretkey <string> :用于访问us3的API私钥或Token私钥

使用示例

- 单个文件数据解冻

```
./us3cli restore us3://bucket/test.txt
```

- 数据解冻整个目录中所有文件

```
./us3cli restore -r us3://bucket/test
```

- 批量激活文件并设置并发数为30

```
./us3cli restore -r us3://bucket/test --parallel 30
```

etag

该命令用于计算文件etag值

命令格式

```
us3cli etag [<本地文件路径1>,[<本地文件路径2>] ...] [us3://<桶名字>/<文件Key>] [-]
```

参数说明

固定参数: "-" 表示计算标准输入中的数据流etag值

--accesskey <string> :用于访问us3的API公钥或Token公钥

--config <string> :当前命令临时指定配置名/配置文件路径

--endpoint <string> :固定域名,可通过地域和域名页查看

-h, --help :当前命令使用说明

--secretkey <string> :用于访问us3的API私钥或Token私钥

使用示例

- 计算本地文件etag值

```
./us3cli etag main.go
```

Name Etag

```
main.go AQAAAEpmwm87EANJQDpLTEMxsjR7-R0N
```

- 计算us3文件etag值

```
./us3cli etag us3://bucket1/test.txt
```

Name Etag

```
us3://bucket1/test.txt AQAAAEpmwm87EANJQDpLTEMxsjR7-R0N
```

- 计算标准输入etag值

```
cat test.txt | ./us3cli etag -
```

Name Etag

```
[-] SmbCbzsQA0IAOktMSbGyNHv5HQ0=
```

- 计算多个文件etag

```
cat test4.txt | ./us3cli etag us3://bucket1/test.txt us3://bucket1/test2.txt test3.txt -
```

Name Etag

```
us3://bucket1/test.txt AQAAAEpmwm87EANJQDpLTEMxsjR7-R0N
```

```
us3://bucket1/test2.txt AQAAAEpmwm87EANJQDpLTEMxsjR7-R0N
```

```
test3.txt AQAAHPuBI-6VRpzVHiBFjSOVhLrcsam
```

```
[-] SmbCbzsQA0IAOktMSbGyNHv5HQ0=
```

create-token

本命令用于创建Token

命令格式

```
us3cli create-token token-name [--flags]
```

参数说明

```
--black-list stringArray 黑名单ip  
-b, --buckets stringArray token允许操作的桶  
-e, --expire int token过期时间  
-h, --help 帮助  
-o, --operations stringArray token允许的操作  
--prefixes stringArray token允许操作的对象前缀  
-p, --projectid string token所属的项目id  
-r, --region string token所在的区域  
--white-list stringArray 白名单ip
```

使用示例

```
us3cli create-token test-token --projectid org-xxxxxx
```

```
2021-10-13 15:02:37.248 INFO Token created successfully, token-id: ea8e7799-3241-xxxxxxxxxx
```

delete-token

本命令用于删除Token

命令格式

```
us3cli delete-token --token-id token-id [--flags]
```

参数说明

```
-h, --help 帮助信息  
-p, --projectid string token的项目id  
-r, --region string token所在的区域  
--token-id string 想要删除的token的id
```

使用示例

```
us3cli delete-token --token-id ea8e7799-3241-xxxxxx --projectid org-xxxxxx --region cn-bj
```

```
2021-10-13 15:05:14.449 INFO Token deleted successfully
```

update-token

本命令用于更新Token

命令格式

```
us3cli update-token token-id [--flags]
```

参数说明

```
--black-list stringArray 黑名单ip  
-b, --buckets stringArray token允许操作的桶  
-e, --expire int token过期时间  
-h, --help 帮助  
-n, --name string token的名字  
-o, --operations stringArray token允许的操作  
--prefixes stringArray token允许操作的对象前缀  
-p, --projectid string token所属的项目id  
-r, --region string token所在的区域  
--white-list stringArray 白名单ip
```

使用示例

```
us3cli update-token c1fd74ad-9c04-4ee2-bb1e-xxxxxxxx --region cn-bj --projectid org-xxxxxx
```

```
2021-10-13 15:12:02.403 INFO Token updated successfully
```

describe-token

本命令用于列出和描述Token

命令格式

```
us3cli describe-token [--flags]
```

参数说明

```
-h, --help 帮助信息  
-p, --projectid string token的项目id  
-r, --region string token所在的区域  
--token-id string 想要删除的token的id
```

使用示例

- 列出某个project,某region下的所有token

```
us3cli describe-token --region cn-bj --projectid org-jhrz4y  
Token Id:  
2853e6eb-951d-4ba4-b2e9-xxxxxxxxxx  
Token Name:  
sourextoken
```

```
Allowed buckets:
[sourcecx]
Allowed operations:
[TOKEN_ALLOW_READ TOKEN_ALLOW_WRITE TOKEN_ALLOW_DELETE TOKEN_ALLOW_LIST TOKEN_ALLOW_IOP]
Allowed prefixes:
[*]
Exipre time:
1569742841
Private key:
4xxxxxxxxxxxxxxxxxxxxxxxxxxxx
Public key:
TOKEN_XXXXXXXXXXXXXXXXXXXXX
....
```

- 描述某个特定的token

```
us3cli describe-token --token-id c1fd74ad-9c04-4ee2-bb1e-xxxxxxx --region cn-bj --projectid org-xxxxx
Token Id:
c1fd74ad-9c04-4ee2-bb1e-xxxxxxx
Token Name:
test-token
Allowed buckets:
[*]
```

```
Allowed operations:
[TOKEN_ALLOW_NONE]
Allowed prefixes:
[*]
Exipre time:
86400
Private key:
dxxxxxxxxxxxxxxxxxxxxxxxxx
Public key:
TOKEN_XXXXXXXXXXXXXXXXXXXX
```

update

本命令用于版本更新

注意：

- 1.windows用户更新后会产生一个名为config.temp的临时文件,在下一次运行us3cli时会自动删除。
- 2.将本工具放入bin目录下执行时,更新操作会出现找不到文件的问题,具体解决方案请查看常见问题页面。

命令格式

```
us3cli update [--force]
```

参数说明

```
-f, --force :忽略版本,强制更新  
-h, --help :当前命令使用说明
```

使用示例

```
./us3cli update  
  
Your version is not the latest version, do you update it now(y or n):y  
Downloaded latest version package,saved as: us3cli-linux64
```

version

本命令用于查看工具版本

命令格式

```
us3cli version
```

使用示例

```
./us3cli version
```

```
us3cli version 1.2.0
```

常见问题

报错 `Open leveledb err: resource temporarily unavailable`

问题原因

1. 使用了老版本的us3cli($\leq v1.3.0$),老版本的us3cli不支持多个进程同时使用。
2. 在使用新版本us3cli时,同时执行同一条命令多次。

解决方案

针对原因1,您可以运行`us3cli update` 命令来更新工具

针对原因2,您可以查看后台是否有正在运行的us3cli进程,执行`kill -9 $旧us3cli进程pid`,关闭旧us3cli进程即可。

Bucket操作如`stat`、`ls`、`du`报错 `bucket not found`

问题原因

当前工具查找bucket范围默认为当前账户默认项目,如您的Bucket不在默认项目下,普通操作是无法找到的,需要使用`--projectid`选项填写您要操作的bucket所在项目ID,即projectid。

解决方案

在您想要操作的命令后加上--projectid 就可以操作当前项目下的bucket

#1.查看projectid

#方法1:mb命令创建时会展示当前账户所有projectid

#方法2:登录控制台查看左上角项目ID

#方法3: ./us3cli ls --projectid 展示当前账户下所有projectid

#2.指定projectid操作bucket

```
./us3cli stat us3://buckettest --projectid xxxx
```

```
./us3cli du us3://bucketttest --projectid xxxx
```

并发数parallel和qps有什么区别?

parallel代表同时进行任务的协程数, 不保证具体的请求速率。

qps代表每秒请求数量限制, 如qps为1, 那么该请求会限制在1秒1个。

上传文件不指定文件key, 上传成功但找不到文件

问题原因

在使用us3cli时, 需要将文件上传到文件夹内, 由于对象存储可以存在同名文件和文件夹的特性, 所以需要在文件夹名称后加上"/"才可以判断为上传文件至当前文件夹, 否则会识别为普通

上传文件。

解决方案

如:上传本地文件test1至存储空间bucketTest的test目录下时,以下方式1只会生成一个test文件在根目录下,而方式2和方式3都可以成功上传文件到test下,命名为test1。

```
方式1:./us3cli cp test1 us3://bucketTest/test
```

```
方式2: ./us3cli cp test1 us3://bucketTest/test/test1
```

```
方式3: ./us3cli cp test1 us3://bucketTest/test/
```

将本工具放入bin目录下全局使用时,使用update命令提示找不到文件的问题

问题原因

更新时工具会根据输入参数找到可执行文件位置,但参数中只有命令名,没有可执行文件路径,故出现找不到文件的问题。

解决方案

在bin目录下执行us3cli update。

报错 Signature VerifyAC Error

问题原因

US3令牌(公钥以TOKEN_为首)仅支持对bucket文件相关操作,在使用需要ApiKey鉴权的命令时会报错:ERROR Signature VerifyAC Error

解决方案

鉴权改用ApiKey方式

1.7.3

新增

- 增加"--language"非交互式创建配置语言选项,可选ZH/EN,其中ZH表示中文,EN表示英文,不使用此选项表示默认ZH中文
- 增加交互式创建配置提示选择语言ZH/EN,默认ZH中文

1.7.2

修复

- 现在cp命令在递归的上传整个文件夹时,当文件夹中任一个文件所有分片都上传完毕,会立刻调用mput finish,而不是等待整批文件都上传完
- 现在上传时会使用bucket配置的存储类型,而不是STANDARD

1.7.0

新增

- 增加"--debug"全局选项, 来打印额外的配置信息以及http请求信息

1.6.2

修复

- 现在cp命令在下载时,会正确的替换掉本地文件,而不是部分覆盖本地文件

1.6.1

修复

- 现在sign命令生成的url会有一个3600s的默认过期时间

1.6.0

新增

- 新增了sync命令的--no-delete选项,可以在同步时默认不删除源端不存在的文件。

1.5.2

修复

- 修复了一个list时的ui bug

1.5.1

新增

- 新增了sync命令的-f选项,可以在同步删除时不弹出确认信息
- 新增了针对503和500的错误提示

1.5.0

修改

- rcat命令可以指定storage class, mimetype和metadata
- cp命令在上传文件夹时, 如果指定了 [--include --exclude --rinclude --rexclude] 参数中的一个, 就不会把文件夹名作为对象键前缀的一部分。
- config命令在交互式配置时会优先询问是否需要自定义endpoint, 而不是先要求选择region

新增

- 新增了ls 命令的--format选项, 可以选择使用json或yaml格式来输出列表, 选项的参数大小写不敏感。(例如us3cli ls us3://bucket --format YAML)

1.4.0

新增

- 对专有云32M分片大小的支持
- 增加了对如下命令列取时携带--prefix-file-list参数来指定使用prefix-file-list接口列取的功能
 - cp
 - du
 - modify
 - mv
 - rm
 - restore
 - sync

1.3.0

修改

- cat命令目前可以并发下载
- 修复了du命令的一个bug,该bug会导致子账号拿不到桶的数据统计信息
- 现在可以并发的上传和下载,但是您无法同时执行两个同样的cp, sync,或du命令。(例如:您无法同时执行两个 cp ./test us3://bucket/prefix/test 命令)
- 现在modify命令在指定--recursive参数时可以批量的修改同一个prefix下的所有对象。
- 优化us3cli的升级逻辑,修复升级失败后再次打开cli出现的找不到可执行文件的错误
- 修复在根目录执行拷贝操作时通配符失效的问题
- 修复代理配置无法清除的问题

新增

- 新增了四个对于token的操作: create-token, delete-token, update-token, describe-token, 分别对应于token的增删改查。
- 新增了三个新的区域: jpn-tky, th-bkk, inspurcloud。
- 新增了ls命令的一个参数: --prefix-file-list, 设置这个参数将会调用prefix-file-list接口来拉取列表。

1.2.2

修改

- 修复上传重试失败问题
- 修复进度条展示出现负数问题
- 修复存在空目录时下载报错问题

新增

- 新增.svg文件的mimetype自动识别

1.2.0

新增

- 增加了动态进度条显示开关, 默认开启
- du命令增加查看目录数据量功能
- ls命令新增查看projecdid功能
- ls展示文件列表新增文件数目显示信息
- cp命令增加自动识别mimetype功能
- sync命令新增指定storageclass (存储类型), mimetype (多媒体文件格式), metadata (元数据) 上传功能
- sync命令新增mimetype自动识别功能
- 配置项新增内容加密、https以及代理功能

修改

- 修改ls下展示文件激活信息, 修改激活时间格式, 新增Restoring标志, 表示文件正在激活中
- 修改sync中的remote模式为local模式, 功能不变

删除

- 删除cp中的hit秒传功能

工具简介

概述

US3FS是一个在Linux/Windows系统环境中, 将US3的存储空间(Bucket)挂载到本地挂载点的工具, 挂载成功后, 您可以像操作本地文件一样操作存储空间(Bucket)中的文件。

版本和运行环境

软件版本

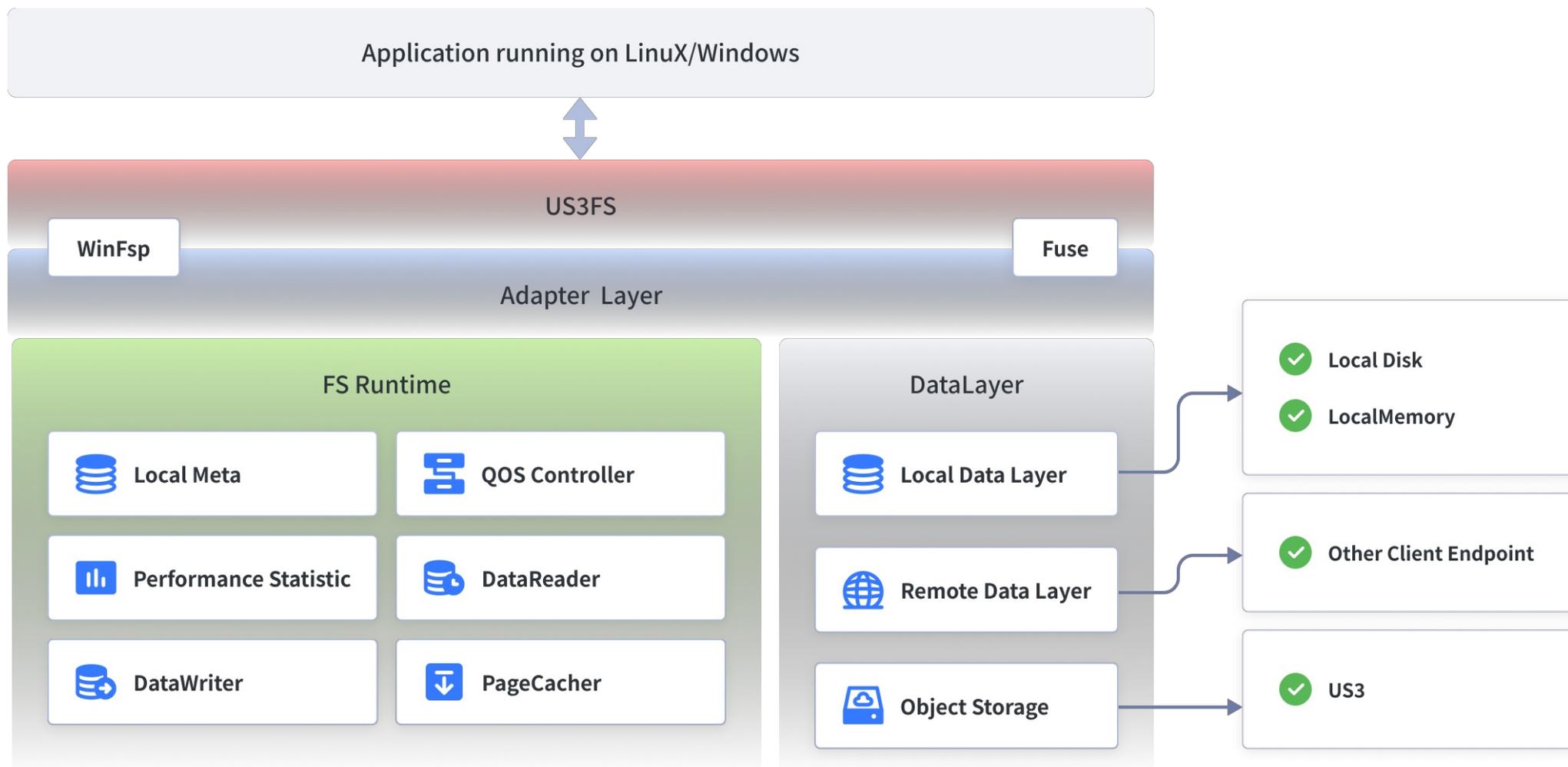
当前版本:

- linux: v2.1.0
- windows: v1.6.8

运行环境

- Linux:
 - CentOS 7.0 及以上 (可通过cat /etc/redhat-release查看)
 - Ubuntu 16.04 及以上 (可通过cat /etc/issue查看)
- Windows
 - 开启WinFsp服务

软件架构图



主要功能

- 支持POSIX文件系统的大部分功能,如读;顺序写;权限;UID/GID。
- 使用US3的分片上传功能上传大文件。
- 支持Etag和MD5校验,保证数据一致性。

使用限制

- 不支持读取归档类型的文件
- 不支持随机写/追加写
- rename非原子操作
- 不支持硬/软链接
- 多个客户端挂载同一个US3 Bucket时,需要用户自行维护数据一致性。

下载与环境准备

运行环境

US3FS基于Linux下fuse和Windows平台下winfsp实现, 您的机器需要支持fuse或winfsp。

建议您将US3FS运行在以下环境中:

- Linux
 - centos 7.0及以上 (可通过cat /etc/redhat-release查看)
 - ubuntu 16.04及以上 (可通过cat /etc/issue查看)
- Windows
 - 下载WinFsp Installer
 - 根据官方说明 进行安装

US3FS支持在UCloud内网以及互联网环境下使用, 在内网环境下, 你可以使用内网域名以提升性能和稳定性。

下载链接

Linux下载链接 或

```
curl -o us3fs https://ufile-release.cn-bj.ufileos.com/us3fs/us3fs_2.1.0
```

Windows下载链接

配置账号访问信息

Linux

编辑/etc/us3fs/us3fs.yaml并增加如下信息(如果没有该目录需要自行创建):

```
access_key: *****  
secret_key: *****  
endpoint: ufile.cn-north-02.ucloud.cn  
hosts: []
```

冒号后有单个空格

- **access_key**: 公钥,支持token密钥和api密钥两种模式,要有覆盖上传权限
- **secret_key**: 私钥,支持token密钥和api密钥两种模式,要有覆盖上传权限
- **endpoint**: 访问域名,详见地域和域名。填写域名为地域域名,并非具体的存储空间域名。
- **hosts**: 指定访问点IP列表,不会走DNS解析逻辑获取US3接入层IP。如果指定个数小于3个不会生效。如: hosts: [10.9.254.190, 117.50.123.23, 117.50.123.29, 117.50.123.8]

hosts指定的IP列表,在遇到异常(网络不可达)IP节点时会在5s检测周期内自动标记剔除,新的请求不受影响,但已经发起的请求且使用异常网络的链接,由于TCP采用退避指数重试

算法,默认重试次数为15次,所以最坏情况要到15min左右才能检测异常,建议在使用该参数时,修改Linux参数net.ipv4.tcp.retries2(或修改系统文件/proc/sys/net/ipv4/tcp_retries2)为6,可使如网络不可达异常在25s左右能检测到,从而剔除已建立异常链接;此外由于目前链接保活探测逻辑会占用一定量的文件描述符,建议调整系统设置,请参考[场景问题](#)中的系统日志出现**too many open file**问题项解决。

当需要在一台机器上挂载多个Bucket时,可以通过--passwd=passwd_file指定账号信息(默认路径为 /etc/us3fs/us3fs.yaml,不需要指定)。

下载US3FS后.使用chmod +x us3fs增加可执行权限,如果需要直接执行,可将us3fs移动到/bin目录下.示例:

```
chmod +x us3fs
./us3fs --passwd=passwd_file <bucket> <mountpoint>

# 移动到可执行目录下
mv us3fs /bin/us3fs
us3fs --passwd=passwd_file <bucket> <mountpoint>
```

Windows

配置信息内同linux,配置路径自定义。

下载好可执行文件后移动到us3fs的工作目录(自定义),然后通过按键windows+R打开运行窗口,输入cmd进入命令行工具界面(后续支持图形化界面),进入可执行文件us3fs.exe 所在路径。示例:

```
# 进入可执行文件所在盘符,这里是D盘
C:\Users\Administrator> D:
# 进入可执行文件所在路径
D:\>cd us3fs
```

```
D:\us3fs>dir
```

```
驱动器 D 中的卷没有标签。
```

```
卷的序列号是 5CAF-F66B
```

```
D:\us3fs 的目录
```

```
2021/09/09 21:16 <DIR> .
```

```
2021/09/09 21:16 <DIR> ..
```

```
2021/09/09 19:13 19,475,146 us3fs.exe
```

```
2021/08/26 11:29 157 us3fs.yaml
```

```
2 个文件 19,475,303 字节
```

```
2 个目录 213,768,716,288 可用字节
```

```
# 进行挂载操作
```

```
# * 这里挂载到x盘,并且指定uid,gid为0的用户,日志级别为debug,预读窗口为32MiB,挂载的US3桶名为rickwu
```

```
D:\us3fs>us3fs.exe --passwd=us3fs.yaml -o debug --uid=0 --gid=0 --level=debug --readahead=32m <bucket> x:
```

注意目前Windows下挂载只能前台挂载

快速上手

- 视频教学
- 使用方式
 - 配置访问权限
 - 设置挂载只读
 - 开启日志
 - 版本更新
- 使用帮助
- 选项列表
- 使用示例
- 场景化参数设置
 - 性能相关参数
 - 小文件场景
- 本地服务支持

视频教学

观看以下视频,快速上手使用US3FS

该内容为动态文件,无法展示

使用方式

- 挂载

```
us3fs [global options] <bucket> <mountpoint>
```

参数<bucket>和<mountpoint>必须依次作为最后两个参数,否则其他参数不能生效

- 卸载

```
umount <mountpoint>
```

windows 对cmd执行Ctrl+C

配置访问权限

us3fs挂载的默认访问权限为当前挂载用户,如果需要允许其他用户/用户组访问挂载点,可以使用如下参数:

- `-o allow_other`:允许任何用户都可以访问文件。
- `--uid=xxx`:指定默认的用户
- `--gid=xxx`:指定默认的用户组

- `--mp_mask`: 用来设置挂载点的权限掩码, 只有当`allow_other`选项设置后, 该选项才生效, 默认值为0000。使用方式基本于`umask`命令一致; 例如需要设置挂载点的权限为770, 则使用参数 `-o allow_other --mp_mask 0007`; 注意`root`用户会忽略非`root`用户挂载时设置的`mp_mask`。

可通过`id`命令获取用户的`uid/gid`信息, 示例如下:

```
// 在ubuntu账户下挂载默认用户和用户组为www的us3fs
ubuntu:~$ id www
uid=1001(www) gid=1001(www) groups=1001(www)

ubuntu:~$ us3fs --uid=1001 --gid=1001 -o allow_other <bucket> <mountpoint>
```

`mp_mask` 配置示例如下:

挂载目录权限	<code>mp_mask</code>
<code>rwxrwxrwx</code>	0000
<code>rwxrwx---</code>	0007
<code>rwx-----</code>	0077
<code>rwxr--r--</code>	0033
<code>rwxrwxr--</code>	0003
<code>rwxr-xr-x</code>	0022

- 如果挂载出现以下问题

```
stderr:
```

```
/bin/fusermount: option allow_other only allowed if 'user_allow_other' is set in /etc/fuse.conf
```

在/etc/fuse.conf中增加user_allow_other。

windows 下只支持--uid和--gid

设置挂载只读

挂载时时指定-o ro。

windows下不支持

开启日志

- --level=info/debug/error 开启指定级别的us3fs日志
- --debug_fuse 开启用户态fuse日志
 - centos 日志在/var/log/messages
 - ubuntu 日志在/var/log/syslog
- 挂载时指定-f, us3fs会以前台模式挂载, 日志会输出到屏幕上。

版本更新

执行如下命令：

```
us3fs --update
```

更新后的可执行文件放在/bin/目录下

```
windows无效
```

使用帮助

通过us3fs -h查看us3fs支持的参数

Linux

```
us3fs - a single posix file system based on us3
```

```
USAGE
```

```
us3fs [global options] bucket mountpoint
```

```
Version
```

```
US3FS Version: v2.1.0
```

```
Commit ID: 879f7d4
```

```
Build: 2024-10-28:19:16:23
```

```
Go Version: go1.17.13 linux/amd64
```

```
FUSE
```

-o value Specify fuse/winfsp option

--entry_timeout value How long to cache dentry for inode for fuse. (default: 5m0s)

--attr_timeout value How long to cache inode attr for fuse (default: 5m0s)

--disable_async_read Disable all read (even read-ahead) operations asynchronously

--wb Enable writeback mode, which is turned off by default

--max_background value Specify the max_background parameter of fuse kernel(>=7.13), currently fuse usespace supports up to 1024 (default: 64)

--congestion_threshold value Specify the congestion_threshold parameter of fuse kernel(>=7.13), currently fuse usespace supports up to 768 (default: 48)

--async_dio Enable the async_dio parameter of fuse kernel, async_dio is disabled by default

--keep_pagecache Turn on pagecache, when the file is opened, it will be decided whether to update according to the modification time of the inode, so please pay attention to the attr_timeout and dcache_timeout parameters will have a certain impact on this

OS

--dcache_timeout value How long to cache dentry for us3fs (default: 5m0s)

--retry value Number of times to retry a failed I/O (default: 5)

--parallel value Number of parallel I/O thread (default: 32)

--disable_remove Disable remove op, such as unlink, rmdir, rename

--debug Set debug level for fuse/winfsp

--level value Set log level: error/warn/info/debug (default: "info")

--log_dir value Set log dir

--log_max_age value Set log max age (default: 72h0m0s)

--log_rotation_time value Set log rotation time (default: 1h0m0s)

--enable_load_dentries enable auto init dentries in memory

--cache_db value specify cache db path, e.g.: dbtype:dbpath

--local_write write file to local and upload async

--max_local_file_size value specify local file max size (default: "32m")

```
--read_after_write_finish read operation will wait all write operation done
--finish_write_when_release all written data will be uploaded when release
--readahead value Readahead size. e.g.: 1m/1k/1 (default: "0")
--max_cache_per_file value Max cache per file when enable readahead. e.g.: 32m/64m/128m (default: "1024m")
--etag value Check etag for part. value is percent(0~100) (default: 50)
--passwd value specify access file (default: "/etc/us3fs/us3fs.yaml")
--enable_md5 Enable md5 in http header
--uid value Specify default uid (default: 0)
--gid value Specify default gid (default: 0)
--mp_mask value Specify mountpoint mask (default: 0)
--disable_check_vdir disable detection of virtual directories
--update Update us3fs to /bin/us3fs
-n Doesn't check access when mount us3fs
-l Enable local cache for small file
-p value Specify local cache location (default: "/tmp/us3fs/")
--prefix value Specify bucket prefix path
--open_rename Use rename api rather than copy and delete
--gfl Enable get_file_list
--direct_read Enable cache bypass read
--perf_dump value How long to output the performance dump (default: 1h0m0s)
--skip_ne_dir_lookup Skip non-essential directory checking, such as files ending in ".log", ".png", ".jpg", etc.
--storage_class value Storage type, including "STANDARD", "IA" (default: "STANDARD")
--enable_remote_cache
--cache_dirs value specify cache db path, e.g.: /tmp/read-cache1,/tmp/read-cache2
--cache_size_limit value specify cache size limit, unit is GB (default: 4)
```

```
--master_addr value master server addr
--data_port value if data_port is specified, then other clients will connect it to get chunk data (default: 0)
--page_size value (default: 1048576)
--fuse_session_cnt value (default: 0)
--enable_recursive_dir enable recursive read dir(default: false)
--pprof_listen_ip value pprof listen ip (default: "127.0.0.1")
```

MISC

```
--help, -h show help
-f foreground
```

Windows

us3fs - a single posix file system based on us3

USAGE

```
us3fs [global options] bucket mountpoint
```

Version

US3FS Version: v1.6.8

Commit ID: c87ec9c

Build: 2022-08-17:11:01:05

Go Version: go1.16.3 linux/amd64

WinFSP

```
-o value Specify fuse/winfsp option
```

```
--dir_info_timeout value The expiration time of the directory information, in seconds (default: 5)
--file_info_timeout value File information expiration time, in seconds (default: 5)
--volume_info_timeout value Volume information expiration time, in seconds (default: 5)
--case_insensitive Is case sensitive
--keep_filecache keep filecache
```

OS

```
--dcache_timeout value How long to cache dentry for us3fs (default: 5m0s)
--retry value Number of times to retry a failed I/O (default: 5)
--parallel value Number of parallel I/O thread (default: 32)
--debug Set debug level for fuse/winfsp
--level value Set log level: error/warn/info/debug (default: "info")
--readahead value Readahead size. e.g.: 1m/1k/1 (default: "0")
--etag value Check etag for part. value is percent(0~100) (default: 50)
--passwd value specify access file (default: "/etc/us3fs/us3fs.yaml")
--enable_md5 Enable md5 in http header
--uid value Specify default uid (default: -1)
--gid value Specify default gid (default: -1)
--disable_check_vdir disable detection of virtual directories
--update Update us3fs to /bin/us3fs
-n Doesn't check access when mount us3fs
-l Enable local cache for small file
-p value Specify local cache location (default: "/tmp/us3fs/")
--prefix value Specify bucket prefix path
--gfl Enable get_file_list
```

```

--direct_read Enable cache bypass read
--perf_dump value How long to output the performance dump (default: 1h0m0s)
--skip_ne_dir_lookup Skip non-essential directory checking, such as files ending in ".log",".png",".jpg", etc.
--storage_class value Storage type, including "STANDARD", "IA" (default: "STANDARD")
MISC
--help, -h show help
-f foreground

```

Linux和Windows的参数区别主要体现在FUSE和WinFsp

配置文件对应终端参数

- 部分终端挂载参数支持配置在配置文件, 配置文件的参数和终端参数的对应关系如下:

终端挂载参数名称	配置文件参数名称	配置文件样例
gfl	get_file_list	get_file_list: true
keep_pagecache	keep_pagecache	keep_pagecache: true
n	no_check	no_check: true
disable_check_vdir	disable_check_vdir	disable_check_vdir: true
async_dio	async_dio	async_dio: true

skip_ne_dir_lookup	skip_ne_dir_look	skip_ne_dir_look: true
l	enable_local	enable_local: true
wb	writeback	writeback: true
direct_read	direct_read	direct_read: true
enable_md5	enable_md5	enable_md5: true
debug	debug	debug: true
-o allow_other	allow_other	allow_other: true
enable_load_dentries	enable_load_dentries	enable_load_dentries:true
disable_async_read	disable_async_read	disable_async_read:true
retry	retry	retry: 66
parallel	parallel	parallel: 77
disable_remove	disable_remove	disable_remove: true
congestion_threshold	congestion_threshold	congestion_threshold: 88
max_background	max_background	max_background: 99
uid	uid	uid=100
gid	gid	gid :100
etag	etag	etag :100
dcache_timeout	dcache_timeout	dcache_timeout: 2h

entry_timeout	entry_timeout	entry_timeout: 3h
attr_timeout	attr_timeout	attr_timeout: 4h
perf_dump	perf_dump	attr_timeout: 5h
log_max_age	log_max_age	log_max_age: 6h
log_dir	log_dir	log_dir: /a/b/c
prefix	prefix	prefix: /a/b/c
level	level	level: debug
storage_class	storage_class	storage_class: STANDARD
p	local_path	local_path: /a/b/c
readahead	readahead	readahead: 8m
max_cache_per_file	max_cache_per_file	max_cache_per_file: 1024m
cache_db	cache_db	cache_db: leveldb:/data/us3fs_cachedb
local_write	local_write	local_write: true
max_local_file_size	max_local_file_size	max_local_file_size: 32M
finish_write_when_release	finish_write_when_release	finish_write_when_release: true
read_after_write_finish	read_after_write_finish	read_after_write_finish: true
enable_remote_cache	enable_remote_cache	enable_remote_cache: true
cache_dirs	cache_dirs	cache_dirs: /mnt/nvme01,/mnt/nvme02

cache_size_limit	cache_size_limit	cache_size_limit: 100
master_addr	master_addr	master_addr: <master_ip>:<master_port>
data_port	data_port	data_port: 3333
page_size	page_size	page_size: 8388608
fuse_session_cnt	fuse_session_cnt	fuse_session_cnt: 4
enable_recursive_dir	enable_recursive_dir	enable_recursive_dir: true
pprof_listen_ip	pprof_listen_ip	pprof_listen_ip: 127.0.0.1

- 挂载参数配置在配置文件样例 编辑/etc/us3fs/us3fs.yaml (如果没有该目录需要自行创建) 依据具体需求将挂载参数写在配置文件, 简化挂载命令

```
access_key: *****
secret_key: *****
endpoint: ufile.cn-north-02.ucloud.cn
hosts: []
```

选项列表

WinFsp

选项名称	描述
------	----

o	WinFsp支持的option参数
dir_info_timeout	目录缓存的超时时间, 默认5s
file_info_timeout	文件缓存的超时时间, 默认5s
volume_info_timeout	卷信息的超时时间, 默认5s
keep_filecache	是否把文件放入缓存

FUSE

选项名称	描述
o	FUSE支持的option参数
entry_timeout	指定fuse缓存被查找的文件名的时间 默认为5min
attr_timeout	指定fuse缓存文件/目录属性的时间 默认为5min
disable_async_read	关闭fuse kernel预读使用异步模式。默认开启
wb	指定写入使用writeback方式。不支持覆盖写/追加写
max_background	指定fuse kernel的max_background参数(fuse kernel版本 \geq 7.13), 目前fuse usespace最多支持1024(默认:64), 该参数能提升direct io的并行度

congestion_threshold	指定fuse kernel(fuse kernel版本 ≥ 7.13)的congestion_threshold参数, 目前fuse usespace最多支持768 (默认:48), 该参数会触发并行IO的拥塞控制
async_dio	开启fuse内核的async_dio参数, 默认关闭async_dio。该参数开启后, fuse kernel对direct io进行异步处理
keep_pagecache	开启pagecache, 文件打开时会根据inode的修改时间以及大小变化决定是否更新, 所以请注意entry_timeout和dcache_timeout参数对此会有一定影响, 使得未能及时感知到文件修改时间、大小变化

- fuse常用选项列表 (与-o一起使用)

选项名称	描述
allow_other	指定文件系统可以所有用户访问 默认关闭
ro	指定当前文件系统为只读

使用方式

```
-o option=value
```

OS(Object Storage)

选项名称	描述
------	----

dcache_timeout	dentry cache在us3fs中的缓存时长 默认为5min
retry	请求失败后重试次数, 默认5次
parallel	I/O并发线程数 默认20个
debug	指定用户态fuse日志级别为debuy 默认关闭
level	指定us3fs日志级别 默认为Info级别
readahead	预读大小。例如:1m/1k/1(默认:"0")
etag	检查上传数据的 etag所占百分比。值是百分比(0~100) (默认值:50)
passwd	指定账户文件, 默认路径/etc/us3fs/us3fs.yaml // windows 平台自定义
enable_md5	在http请求头中增加md5校验, 默认关闭
uid	指定文件所属的默认用户, 默认当前用户
gid	指定文件所属的默认用户组, 默认当前用户组
disable_check_vdir	禁用虚拟目录检测
update	更新us3fs版本, 新版本路径为/bin/us3fs
n	挂载时不检查bucket权限, 如果没有所在地域没有列表服务可开启

l	开启后对小文件使用本地目录做缓存,异步上传。具体使用示例见小文件场景
p	指定小文件异步上传的本地缓存目录
prefix	指定挂载的bucket前缀目录,默认为空
gfl	对于没有ListObjects API支持的Endpoint, 该参数可以绕过,通过PrefixFileList API模拟
direct_read	开启后,绕过us3fs内部缓存组织模块,直接读取us3数据, 对于被频繁访问的文件会有一定性能降低,反之有利于降低时延
perf_dump	指定时间周期输出时延统计信息,默认周期是1hour
skip_ne_dir_lookup	跳过非必要的目录检查,目前过滤支持".jpe"、".jpeg"、".png"、 ".gz"、".tgz"、".gz"、".tgz"、".log"、".plot"、".js"、".html"、 ".css"、".apk"为后缀的文件,需要确保bucket下没有用以上后缀 作为目录后缀的情况
storage_class	指定写入US3中文件的存储类型,支持STANDARD(标准), IA(低频)两种。(default: STANDARD)
cache_db	指定本地存储us3fs元数据cache的方式及路径,格式为:leveldb:/data/us3fs_cachedb
finish_write_when_release	开启后,支持文件异步结束上传,用于支持一个fd有多次flush的场景
read_after_write_finish	开启后,读取文件时,如果文件正在写入,会等待文件写入完成后才返回读的内容
local_write	开启后,写入的数据会暂存到本地文件系统后再异步上传到服务端
max_local_file_size	和 local_write搭配使用,指定能写入到本地文件系统的最大文件大小(默认值:32M)

MISC

选项名称	描述
help, h	查看帮助
f	挂载时启用前台模式, 相关输出会打印到标准输出

windows 下-f参数无效

使用示例

- **entry_timeout, attr_timeout, dcache_timeout:**

设置dcache_timeout可增加文件/目录属性在内存中的有效时间, 增强使用体验。建议entry_timeout , attr_timeout设置时间小于dcache_timeout

注: 开启缓存后, 可能造成用户读取目录的内容和实际bucket中的内容不一致。默认为开启, 需要关闭请设置为0s

示例: ls包含10000个文件的目录耗时

```
[root@10-9-120-211 ~]# us3fs --dcache_timeout=60s --entry_timeout=60s --attr_timeout=60s <your_bucket> <mountpoint>
[root@10-9-120-211 ~]# time ls -la <your_dir> | wc -l
10003

real 0m5.964s
user 0m0.033s
sys 0m0.232s
```

```
[root@10-9-120-211 ~]#  
[root@10-9-120-211 ~]#  
[root@10-9-120-211 ~]# time ls -la <your_dir> | wc -l  
10003  
  
real 0m0.872s  
user 0m0.029s  
sys 0m0.133s
```

- **disable_async_read**

默认读取模式为异步,同步读取性能较差。

示例如下:

```
[root@10-9-120-211 ~]# us3fs --disable_async_read <your_bucket> <mountpoint>  
[root@10-9-120-211 ~]# dd if=<your_file> of=/dev/null bs=4M count=10  
10+0 records in  
10+0 records out  
41943040 bytes (42 MB, 40 MiB) copied, 10.2345 s, 4.1 MB/s  
  
[root@10-9-120-211 ~]# us3fs <your_bucket> <mountpoint>  
[root@10-9-120-211 ~]# dd if=<your_file> of=/dev/null bs=4M count=10  
10+0 records in  
10+0 records out  
41943040 bytes (42 MB, 40 MiB) copied, 0.685801 s, 61.2 MB/s
```

- **parallel**

增大并发数可提升读写性能,相应的也行增加系统资源占用。

示例如下:

```
// 默认并发数20
[root@10-9-120-211 ~]# us3fs <your_bucket> <mountpoint>
[root@10-9-120-211 ~]# dd if=/dev/zero of=<your_file> bs=4M count=1024
1024+0 records in
1024+0 records out
4294967296 bytes (4.3 GB, 4.0 GiB) copied, 25.5351 s, 168 MB/s

// 调整并发数为32
[root@10-9-120-211 ~]# us3fs --parallel=32 <your_bucket> <mountpoint>
[root@10-9-120-211 ~]# dd if=/dev/zero of=<your_file> bs=4M count=1024
1024+0 records in
1024+0 records out
4294967296 bytes (4.3 GB, 4.0 GiB) copied, 18.3614 s, 234 MB/s
```

- **readahead**

调整预读窗口大小对大文件的顺序读有较大影响,建议在16m~32m,但会增加内存消耗,可以适当缩小预读窗口。

示例如下:

```
// 默认预读大小16MB
[root@10-9-120-211 ~]# dd if=<your_file> of=/dev/null bs=4M count=1024
```

```
1024+0 records in
1024+0 records out
4294967296 bytes (4.3 GB, 4.0 GiB) copied, 60.0498 s, 71.5 MB/s

// 调整预读大小为32MB
[root@10-9-120-211 ~]# us3fs --readahead=32m <your_bucket> <mountpoint>
[root@10-9-120-211 ~]# dd if=<your_file> of=/dev/null bs=4M count=1024
1024+0 records in
1024+0 records out
4294967296 bytes (4.3 GB, 4.0 GiB) copied, 37.6013 s, 114 MB/s
```

场景化参数设置

性能相关参数

- `parallel`: 设置并发线程, 对cpu负载有一定影响。建议设置在20~40较为合理
- `critical`: 写入文件时启用本地etag校验, 相比未开启会提高约50%的cpu占用。
- `readahead`: 预读窗口大小, 由于fuse自身有读写窗口的限制, 一定的预读大小对读取性能有显著提升。建议设置在16m~32m, 但会增加内存消耗, 可以适当缩小预读窗口。
- `keep_pagecache`: 尽可能缓存数据内容在vfs pagecache中, 直到文件的修改时间和大小发生变化, 才无效掉pagecache中的历史数据。
- `wb`: 该参数会使得写入的IO会尽量在pagecache中合并, 然后以大IO(默认128K)发送到us3fs。
- `fuse_session_cnt`: 设置与内核fuse模块通信的连接数, 对cpu和内存负载有一定影响, 建议在资源空闲的机器上可开启, 并将数量设置在CPU核数以内, 开启后读写性能都有一定程度的提升

小文件场景

对于大量小文件场景,如果对性能有要求,可指定-l开启本地本地缓存。当启用本地缓存后,us3fs挂载后首先会将指定缓存目录下已存在的所有小于等于4MB的文件按照其路径上传到bucket中。当写入文件大小不大于4MB,文件会尝试写入本地缓存目录,写入成功后即返回,后端异步上传到us3。写入失败(如权限不足,空间不足等)则仍然使用同步方式写入us3对象存储

注:异步上传可能出现写入后端失败,us3fs会一直重试直到写入成功。

高读吞吐场景

- `cache_dirs`: 设置本地读缓存磁盘,推荐使用本地nvme盘来存放读缓存内容,可设置多个盘,使用 , 号分割,注意缓存盘尽量不要使用系统盘,避免由于瞬间的IO上涨导致系统hang住,推荐使用独立的本地nvme盘(盘的读写吞吐能在2GB+)。
- `cache_size_limit`: 设置缓存盘存储量使用上限,当缓存的内容达到该上限时,会进行LRU淘汰
- `page_size`: 在开启本地读缓存功能时,需要将`page_size`从默认的1048576调整到8388608

共享读缓存

当多个挂载点之间需要共享读缓存内容时,可以通过执行

```
us3fs run-master --listen_addr 192.168.0.10:6667
```

命令来启动一个master节点运行,同时将其余的挂载点的 `master_addr` 参数设置为 192.168.0.10:6667,这样各个挂载点会同该master定期保持心跳,这一组挂载点将会成为一个小型的读缓存集群

- `enable_remote_cache`: 当一个挂载点开启该参数后,其余的挂载点就可能会连接上它检查是否有缓存数据可读,当发现目标数据存在时,会尝试从该挂载点读取数据内容
- `data_port`: 当一个挂载点开启`enable_remote_cache`后,暴露在哪个端口对外进行访问

自动挂载

推荐的最佳实践方案是采用systemctl来保证服务异常退出,或者机器重启后能重新挂载us3fs,但需要注意Linux(Windows平台暂未提供方案)的发行版本是否支持systemctl:

- Ubuntu >= 15.04
- Centos、RHEL >= 7

满足系统要求后,经过如下几个简单步骤即可:

1. 设置配置

在/etc/systemd/system/目录下创建名为us3fs.service的文件,并增加如下内容

```
[Unit]
Description=US3FS (User Space FileSystem for US3)
Requires=network-online.target
After=network-online.target

[Service]
Type=forking
User=<user>
Group=<group>
Restart=always
RestartSec=10
ExecStart=/bin/us3fs --passwd=/etc/us3fs/us3fs.yaml --keep_pagecache <your_bucket> <mountpoint>
ExecStop=/bin/umount <monutpoint>

[Install]
```

```
WantedBy=multi-user.target
```

- *User*为需要访问挂载点的用户。如root。
- *Group*为需要访问挂载点的用户组。如root。
- *ExecStart*为挂载命令,按照需要自行填写

执行systemctl daemon-reload配置则会生效。

2. 服务启用或停止

执行systemctl start us3fs.service启动服务;

执行systemctl stop us3fs.service停止服务;

执行systemctl restart us3fs.service重启服务;

执行systemctl status us3fs.service查看服务状态;

3. 设置为开机自启动

执行systemctl enable us3fs.service;

常见问题

适用于所有挂载失败的情况

挂载的时候,指定 `-f` 参数以前台进程方式运行us3fs,可直观地从标准输出中查看异常信息

挂载失败,日志出现“internal error”

所在地域不支持列表服务,可指定`--gfl`重新挂载。(不支持ListObjects API 地域)

挂载时出现403

- 日志ErrMsg信息出现"action not allow",则检查token相应权限,缺失并添加,过5min后测试
- 日志ErrMsg信息出现"invlid signature",先检查配置文件字段是否符合官方文档说明,其次检查token公私钥是否粘贴完整

异常退出后再次挂载报错

1. 执行:`umount <挂载点>`,如果失败,进入2:
2. 执行:`sudo umount <挂载点>`,如果失败,联系技术支持

读写发生错误

报错信息如:“Input/Output error“,进行如下处理:

1. 检查系统日志:
 1. Centos的路径:/var/log/message
 2. Ubuntu的路径:/var/log/syslog
2. 如果通过日志发现是有http状态码为500错误,请联系技术支持查询该Bucket请求报500原因。

当前用户挂载后其他用户无法访问

添加参数-o allow_other

挂载后发现跟控制台文件信息不一致

该问题是由于默认开启entry_timeout,attr_timeout,dcache_timeout参数,并5min过期时间,如果对一致性敏感,建议这些参数设置为0s来关闭

降低读写时延的方法

- 应用层系统调用写入尽量采用大IO($\geq 128\text{KiB}$, 如果内存允许使用 $\geq 1\text{MiB}$ 的提升吞吐效果更佳), 并尽量保持 4KiB 对齐
- 对于写入大文件, 设置parallel参数提升写入并发度, 提升整体写入吞吐
- 对于大文件顺序读, 建议设置参数readahead, 如 32m
- wb开启回写模式, 减少内核态与用户态上下文切换次数, 提升写入速度, 但注意通过wb模式写入的文件不能进行覆盖写
- 对于大量非热点小文件($< 256\text{KiB}$)顺序/随机读, 大文件随机读场景, 建议设置参数direct_read

- 对于热点小文件(<256KiB)顺序/随机读场景, 不设置参数direct_read
- 对于有高IOPS的业务需求, 建议采用异步IO, 同时使用DirectIO模式, 并根据实际情况调大max_background和congestion_threshold
- 开启keep_pagecache可以利用系统VFS pagecache, 对于热点文件特别是作为web静态资源存储加速明显;
- 开启参数skip_ne_dir_lookup, 这样可以减少lookup的操作时延;

内存使用量高, 导致OOM

如果us3fs异常退出, 通常是机器内存不足导致, 可以通过检查系统日志是否有OOM来判断, 如果发现了类似如下的日志, 则可以确定是发生了OOM

```
earlyoom: low memory! at or below SIGKILL limits: mem 2.00%, swap 100.00%
earlyoom: sending SIGKILL to process 5637 uid 0 "us3fs": badness 671, VmRSS 2550 MiB
```

建议解决方案:

- 由于us3fs读写都会占据机器内存, 要达到最佳的读写性能, 建议更换到内存充裕的机器上使用
- 如果对读写性能不是特别敏感, 可以通过降低parallel参数(默认为32)来降低us3fs的读写并发, 以达到减少内存使用的目的
- 如果内存本身紧张, 建议entry_timeout,attr_timeout,dcache_timeout参数设置为0s
- 检查是否开启readahead, 如果开启, 检查是否过大, 调小该值
- 如果不是顺序读大文件的场景, 建议设置direct_read参数

控制台有对应的目录, 但是us3fs挂载路径没有

证明这是一个虚拟目录, 需要在us3fs挂载路径执行mkdir <这个目录>来显示该目录, 或者开启check_virtual_dir该参数, 通过以这个目录为前缀来检测是否有这个前缀的文件, 来决定是否"应该"展示该目录, 但不建议这么操作, 会增加us3fs的时延。在v1.5.4版本以后check_virtual_dir参数功能默认开启, 并取消该参数, 如果需要关闭该功能可以设置disable_check_vdir参数。

系统日志出现‘too many open files’

该场景出现在有大量随机IO读的情况下,需要调大系统对进程打开文件描述符数量的限制(一般为1024)。通过ulimit -a检查open files项,设置成65535或以上。设置方式:

- - 单次bash环境有效: ulimit -n 65535; <us3fs 挂载命令>;
 - 系统生效: 参考Too many open files;

历史版本

US3FS v2.1.0

新特性

- 增加pprof_listen_ip参数设置,用于设置pprof的监听ip

US3FS v2.0.9

新特性

- 增加enable_recursive_dir参数设置,用于设置是否开启递归目录刷新,可能会增加拉取列表的延迟

US3FS v2.0.8

BUG修复

- 目录的ctime/mtime统一设置成20000401的时间,支持rsync命令设置目录的时间

US3FS v2.0.7

新特性

- 新增文件夹递归刷新的特性

US3FS v2.0.6

BUG修复

- 修复非root用户挂载us3fs 文件中的用户组权限为root的问题

US3FS v2.0.5

新特性

- 支持mv接口中文件大于100MB的场景

US3FS v2.0.3

BUG修复

- 修复部分场景下 GID, UID 显示错误

US3FS v2.0.2

BUG修复

- 修复小概率的大文件上传丢失最后一块分片的异常

US3FS v2.0.1

新特性

- 增加fuse_session_cnt参数设置,用于设置us3fs与内核fuse模块的连接数,开启后读写性能都会有一定程度的提升

US3FS v2.0.0

新特性

- 增加读缓存功能,提升读吞吐
- 增加同一个bucket多个挂载点之间操作同步功能

US3FS V1.7.12

BUG修复

- 修复部分网络连接异常问题

US3FS V1.7.11

新特性

- 增加mp_mask参数,可对挂载点进行权限控制

US3FS V1.7.10

BUG修复

- 修复开启finish_write_when_release时,对正在结束上传的文件做mv操作时,会mv到一个空文件

US3FS V1.7.9

BUG修复

- 修复使用prefix参数报错问题

US3FS V1.7.8

新特性

- 增加stats,gc,reload命令

改进

- 优化顺序读判断
- 优化读缓存淘汰的效率

US3FS V1.7.7

新特性

- 增加 `--reload` 命令,可以通知运行在同一台机器上的us3fs实例reload它们使用的配置文件(仅针对us3fs_1.7.7版本及以后的版本)
- 优化预读的读取效率和内存使用

US3FS V1.7.6

BUG修复

- 修复部分场景下小文件上传失败的问题

新特性

- 增加--disable_remove参数,用来禁用 unlink, rmdir, rename等操作

US3FS V1.7.5

BUG修复

- 修复部分场景下rename导致inode本地丢失的问题
- 修复预读场景下读取压力大时内存占用过大的问题

US3FS V1.7.4

BUG修复

- 修复文件句柄多次close导致异常退出的问题

新特性

- host参数可配置IP加端口

改进

- 适配高版本的fuse配置

US3FS V1.7.3

BUG修复

- 修复使用skip_ne_dir_lookup时异常退出的问题

US3FS V1.7.2

BUG修复

- 修复crash日志在某些场景下会被异常清理

US3FS V1.7.1

BUG修复

- 修复某些场景下的文件查找失败

US3FS V1.7.0

新特性

- 增加cache_db参数,支持将us3fs元数据缓存在磁盘
- 增加finish_write_when_release参数,支持异步结束文件上传
- 增加read_after_write_finish参数,支持读等待写入完成后才返回数据

改进

- 优化文件列表的加载速度

BUG修复

- 修复部分异常场景下的内存未回收问题

US3FS V1.6.8

新特性

- 支持写入时识别文件mimeType
- 增加enable_load_dentries参数,支持在us3fs挂载成功后提前加载目录信息,加快文件访问效率

改进

- 支持在配置文件里设置挂载参数
- bucket字段从配置文件移除,从而支持多个bucket使用同一份配置文件挂载
- 优化使用-gfl选项时拉文件列表的效率

BUG修复

- 修复部分场景下本地inode cache不会更新问题
- 修复部分场景下dcache_timeout配置不生效问题
- 修复flush后再次write进程异常crash的问题
- 修复部分写入场景下,写入的文件长度为0的异常
- 修复inode同步延迟的问题

US3FS V1.6.7

新特性

- 新增log_dir, log_max_age, log_rotation_time参数指定日志的输出路径

BUG修复

- 优化高并发读写情况下的内存使用

US3FS V1.6.6

新特性

- 支持低频存储

BUG修复

- 索引ttl更新异常导致的缓存淘汰时间不准确

US3FS V1.6.5

BUG修复

- 使用host参数时tcp链接无法复用问题
- 大文件分片上传时tcp链接无法复用问题

US3FS V1.6.4

BUG修复

- 修复专有云场景写入异常

US3FS V1.6.3

BUG修复

- 修复写入文件提交最后尾部数据时,后端报错无法透传给用户态问题

US3FS V1.6.2

改进

- 异常奔溃日志捕获

BUG修复

- 修复写入文件时的panic问题
- 默认配置路径修复

US3FS V1.6.0

新特性

- 支持通过命令行在windows平台挂载
- 支持指定ip列表进行访问

改进

- 清理逻辑

BUG修复

- 并发读写内部数据结构导致的奔溃问题
- fuse kernel触发的forget操作导致的路径信息丢失
- -o allow_other参数无效问题

US3FS v1.5.5

新特性

- 无

改进

- 客户端http请求负载均衡

BUG修复

- 上传压力大情况下,会导致上传分片不完整,从而导致失败
- 空指针导致奔溃

US3FS v1.5.4

新特性

- 新增max_background,congestion_threshold参数支持direct模式下高并发io
- 新增async_dio参数,允许fuse kernel对direct io异步
- 新增keep_pagecache参数对文件大小以及修改时间不变的文件在vfs中进行缓存

改进

- 自动补全缺失的"key"格式目录,目录只创建"key/"格式的文件,且路径解析时优先head "key/"格式的文件;减少目录索引创建,有效减少创建与查询时延问题
- 参数check_virtual_dir功能默认开启,且取消该参数,需要关闭使用disable_check_vdir参数;减少对虚拟目录进行判断带来的时延
- 新增skip_ne_dir_lookup参数来开启文件名后缀的过滤字典,减少head操作频率,目前过滤支持".jpe"、".jpeg"、".png"、".gz"、".tgz"、".gz"、".tgz"、".log"、".plot"、".js"、".html"、".css"、".apk"为后缀的文件,需要确保bucket下没有用以上后缀作为目录后缀的情况

BUG修复

- 无

US3FS v1.5.3

新特性

- 新增direct_read参数支持绕过us3fs cache直接读us3
- 简单的性能分析数据周期性输出, 默认1h输出一次, 可以通过perf_dump参数调整
- 创建目录的mime-type设置为"application/x-directory", 便于在console上进行区分

改进

- entry_timeout, attr_timeout, dcache_timeout参数默认开启, 且设置为5min
- readahead参数修改支持字符串解析, 如:16m(1MiB), 8k(8KiB), 512(512B)
- 自动补全缺失的"key"格式目录, 减少lookup目录时两次head操作("key"和"key/"格式的)为一次
- 默认开启参数nodirbl的功能, 并且取消该参数, 替换为语义化参数check_virtual_dir
- 新增文件名后缀的过滤字典, 减少head操作频率, 目前仅过滤以".jpe"和".jpeg"为后缀的文件

BUG修复

- 分配inode时未初始化指定的uid,gid信息, 以及未初始化mtime,ctime
- 目录成员过多情况下读取不完整问题
- allow_other参数未生效问题

US3FS v1.5.2

新特性

- 无

BUG修复

- 修复了后端连接重置后读取文件导致崩溃的问题

US3FS v1.5.1

新特性

- 支持使用GetFileList获取目录下文件;通过--gfl参数开启

BUG修复

- 修复了mv空目录报错的问题。
- 取消了预读最大64MB的限制

US3FS v1.5.0

新特性

- 支持US3后端动态分片
- 支持挂载时指定bucket前缀路径

US3FS v1.4.0

新特性

- 支持使用本地缓存异步上传
- 支持小文件读取读已之写 (Read-Your-Write)

改进

- 通过--level指定日志级别

BUG修复

- 修复了文件名中有特殊字符无法写入的问题。
- 修复了写入过程中可能出现的进程中断的问题。
- 修复了在对象存储中创建同名的目录和文件, ls会显示两个目录的问题
- 修复了初始化分片上传失败会阻塞的问题

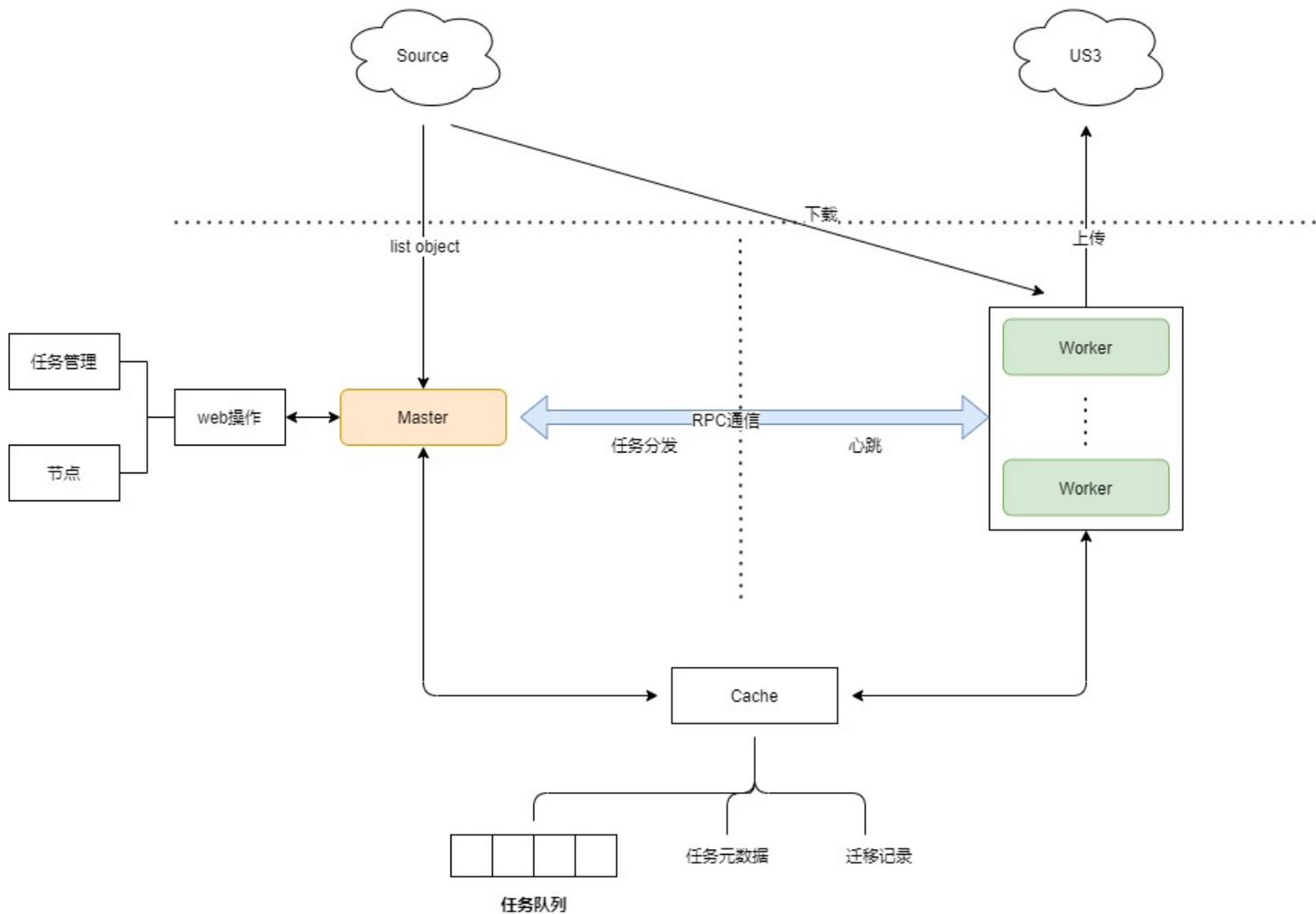
工具简介

概述

US3SYNC 是一款将不同源的数据同步到 US3 的迁移工具。通过将 US3SYNC 部署在本地或者云主机中,可以便捷地从本地或者其他云环境中将数据迁移到 US3 存储空间。US3SYNC工具可以创建两种模式的任务:

- **普通模式:** 普通同步任务可以将源端某个指定前缀或文件夹的文件批量的同步到US3上。
- **fetch模式:** fetch模式的同步任务是通过POST请求(描述需要同步的文件URL和目的位置)将文件同步到指定的位置上。可以指明文件同步成功和失败后的回调接口,US3SYNC会在同步任务完成后,发送相应的通知。

普通任务工作原理



图中master节点与worker节点功能：

- master节点：

单点部署,负责迁移任务的管理。其主要逻辑是从源端拉取文件列表,然后将需要迁移的文件派发给worker进程迁移。

- worker节点:

支持节点扩展,负责迁移文件。其主要逻辑是从源端下载文件,然后将文件上传到目的端。

master节点与worker节点可以部署在同一台机器,也可以部署在多台机器上,用户可以根据需要自行扩展worker节点,下面分别介绍:

- 部署在同一台机器:

master节点和worker节点通过启动时配置的内部通信监听地址进行通信。用户需要确保配置给worker节点的路径是单独的路径,不可与master路径以及其他worker路径重复。

- 部署在不同机器:

master节点和worker节点通过启动时配置的内部通信监听地址进行通信,确保该地址在worker机器上可以访问。用户需要确保配置给worker节点的路径是单独的路径,不可与master路径以及其他worker路径重复。

Fetch任务工作原理

Fetch类型的任务跟普通任务工作原理大致类似,不同的是同步任务的源端需要发送**POST**请求来指定,而不是根据配置的源端的位置自动拉取。

POST请求语法

```
POST /fetch/ HTTP 1.1
Authorization: Auth
Content-Type:application/json
```

请求头部

Authorization为必选项, content-type, date等都是可选项

例如: Authorization: UCloud xxxx-xxxx-xxxx-xxxx-xxxx:xxxxxxxxxxxxxxx 计算方法伪码如下:

```
method := "POST"
md5 := xxxxxx
contentType := xxxxxx
date := xxxxx
privateKey = xxxxx-xxxx-xxxxx-xxxx
publicKey = xxxx-xxxx-xxxx-xxxx
strToSign = method + "\n" + md5 + "\n" + contentType + "\n" + date + "\n"
signature = HmacSHA1(strToSign, privateKey)
signature = Base64(signature)
Authorization: "UCloud " + publicKey + ":" + signature
```

请求内容

名称	描述	类型	必选
Url	源站资源地址,需要url encode	String	是
Key	桶内文件路径,不需要url encode	String	是
Bucket	桶名	String	是
JobId	Fetch任务Id	String	是
SuccessCallbackUrl	回源拉取成功的回调地址	String	否
FailureCallbackUrl	回源拉取失败的回调地址	String	否

此处的JobId可以在界面中获取

返回内容

名称	描述	类型
RetCode	请求状态码	Int
ErrMsg	请求信息	String
TaskId	任务的唯一id	String

```
{
  "RetCode":,
  "ErrMsg":,
  "TaskId":
}
```

示例请求

```
POST /fetch/ HTTP/1.1
Authorization:Authorization: UCloud this-is-my-public-key:AAAArandomsignature=
Content-Type:application/json
Content-Length: 159

{
  "Url": "http://xxx.xxx.xxx/xxx/movie.mp4",
  "Bucket": "example_bucket",
  "Key":"movie.mp4",
  "JobId": "xxxxxxxxxxxxxxxxxxxxxx",
  "SuccessCallbackUrl":"http://xxx.xxx.xxx/xxx",
  "FailureCallbackUrl":"http://xxx.xxx.xxx/xxx"
}
```

返回

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Content-Length: 122
Connection: keep-alive
```

```
{
  "RetCode":0,
  "ErrMsg":"success",
  "TaskId": "d4d62b79-b292-411a-a1f2-47369e2b532f"
}
```

回调内容

您可以在请求中指定任务完成后的回调地址,US3SYNC会依照任务执行结果给相应的地址放松POST请求,请求内容示例如下:

Failure callback

```
{
  "Code":1,
  "TaskId": "d4d62b79-b292-411a-a1f2-47369e2b532f",
  "Message":"We encountered an internal error.",
  "Resource":"http://xxx.xxx.xxx/xxx/movie.mp4",
}
```

Success callback

```
{
  "Code": 0,
  "TaskId": "d4d62b79-b292-411a-a1f2-47369e2b532f",
  "ETag":"xxxxxxxx",
}
```

```
"Key": "xxxxxxxxxxxxxxxxx.mp4",  
"SHA1": "0bc51013e87869137a432200f57daf6affdd3d0c",  
"Size": 638304718  
}
```

主要功能

1. 支持从 s3, oss, qiniu, youpai, US3 云存储迁移数据到 US3 中。
2. 支持从 NAS 存储或者本地目录将数据迁移到 US3 中。
3. 支持指定 URL 资源列表将数据迁移到 US3 中。
4. 支持 web 管理, 通过 web 管理迁移任务, 迁移节点。
5. 支持源端管理
6. 支持配置任务定期执行
7. 支持失败文件列表导出

注: 暂不支持源端为归档类型的文件迁移到 **US3**。

文件结构

US3SYNC

├─ bin

│ └─ master # master 可执行程序

│ └─ worker # worker 可执行程序

└─ conf

```
| └─ config.toml # 配置文件
| └─ cert # https证书
| └─ log # master日志文件存放路径
| └─ pika # 依赖pika
└─ console.sh # 启动脚本
```

与原迁移工具对比

原迁移工具使用请参照:原迁移工具

US3SYNC	ufile-import
提供界面管理操作	只支持命令行操作
配置文件整合为单个	多个配置文件
数据不落盘, 提高迁移效率	数据落盘, 需要根据需要提供磁盘资源
使用 pika 缓存	使用 redis 缓存
按分片粒度并发, 带宽稳定	按文件粒度并发, 对大文件迁移不友好
支持按照大小进行数据校验	不支持校验

版本和运行环境

软件版本

当前版本:1.10.4

运行环境

- Linux:
 - CentOS 7.0 及以上 (可通过cat /etc/redhat-release查看)
 - Ubuntu 16.04 及以上 (可通过cat /etc/issue查看)

下载与环境准备

US3SYNC提供“通用部署”和“UHOST镜像启动”两种部署US3SYNC的方式。用户可根据现有资源情况选择“通用部署”或“UHOST镜像启动”US3SYNC服务。通用部署方案如下：

运行环境

- Linux:
 - CentOS 7.0 及以上 (可通过cat /etc/redhat-release查看)
 - Ubuntu 16.04 及以上 (可通过cat /etc/issue查看)

US3SYNC 依赖 telnet, expect, rsync 命令, 确保预装这些命令, 使用 yum 源做包管理可以使用以下命令安装:

```
yum install -y telnet expect rsync
```

下载并解压

```
wget -O US3SYNC.tgz "https://ufile-release.cn-bj.ufileos.com/US3SYNC/v1.10.5/US3SYNC.tgz"  
tar xzf US3SYNC.tgz  
cd ./US3SYNC
```

启动master服务

```
./console.sh start
请设置缓存服务监听端口[9000]:
请设置缓存服务密码[user-passwd]:
# 云主机一般绑定EIP,建议ip地址使用0.0.0.0
请设置web 服务监听地址[0.0.0.0:443]:
# 内部通信地址不提供外网服务,建议ip地址使用机器内网ip
请设置内部通信监听地址[x.x.x.x:8080]:
请设置报错重试次数[10]:
请设置web 登录使用的用户名[root]:
请设置web 登录使用的密码[passwd]:
US3SYNC start success!
# 查看master服务
./console.sh show
# 结束master服务
./console.sh stop
# 验证
# 查看进程是否正常启动
./console.sh show
```

添加worker节点

1. 服务启动后,在浏览器中打开:https://<web服务监听IP>:<web服务监听端口>/
注: 使用云主机部署迁移服务时,这里需要使用**EIP**,而不是**0.0.0.0**。
2. 页面登录,使用启动时设置的用户名和密码。
3. 添加工作节点,参考 创建节点界面说明,需要为每个节点提供唯一工作路径。每个工作节点需要提供唯一工作路径,如果路径不存在会自动创建对应目录。
注: 建议使用内网**ip**。

视频参考

部署与使用视频教学请参考【快速上手】

快速上手

- 快速上手
 - 视频教学
 - 界面说明
 - 登录界面说明
 - 任务管理界面说明
 - 错误日志导出
 - 创建任务界面说明
 - 可以在此处获取jobid来构建Fetch请求
 - 创建迁移源界面说明
 - 创建节点界面说明
 - 使用说明
 - 添加Endpoint
 - 创建任务
 - 启动迁移
 - 暂停迁移
 - 重试
 - 导出失败文件列表
 - 缓存说明
 - 迁移类型说明
 - S3
 - local
 - http

- 使用说明
 - 添加Endpoint
 - 创建任务
 - 启动迁移
 - 暂停迁移
 - 重试
 - 导出失败文件列表
- 缓存说明
- 迁移类型说明
 - S3
 - local
 - http
- 使用说明
 - 添加Endpoint
 - 创建任务
 - 启动迁移
 - 暂停迁移
 - 重试
 - 导出失败文件列表
- 缓存说明
- 迁移类型说明
 - S3
 - local
 - http
- 使用说明

- 添加Endpoint
 - 创建任务
 - 启动迁移
 - 暂停迁移
 - 重试
 - 导出失败文件列表
- 缓存说明
 - 迁移类型说明
 - S3
 - local
 - http

视频教学

观看以下视频,快速上手部署与使用US3SYNC

该内容为动态文件,无法展示

界面说明

服务启动后,通过浏览器打开页面<<https://<web服务监听IP>:<web服务监听端口>>>,即可访问US3SYNC>界面,在界面上进行数据的迁移操作。

登录界面说明

* 用户名:

* 密码: 

任务管理界面说明

US3SYNC

[迁移任务](#) [迁移节点](#) [EndPoint](#)

创建任务

任务名称	源端EndPoint	目的端EndPoint	覆盖类型	md5校验	任务状态	迁移情况	操作
∨ test0621	src	dst	全覆盖	未开启	迁移完成	等待执行:0 正在执行:0 迁移总数:2160 迁移成功:2160 迁移失败:0	暂停 管理

错误日志导出

US3SYNC

迁移任务 迁移节点 EndPoint

创建任务

任务名称	源端EndPoint	目的端EndPoint	覆盖类型	md5校验	任务状态	迁移情况	操作
test0621	src	dst	全覆盖	未开启	迁移完成	等待执行:0 正在执行:0 迁移总数:2160 迁移成功:2160 迁移失败:0	暂停 管理 重试 删除 克隆 导出
test0622	src	dst	全覆盖	未开启	迁移完成	等待执行:0 正在执行:0 迁移总数:2160	暂停 管理

如果任务出错，点击此处可以导出错误日志

- 任务操作:

开始迁移:开始文件迁移

开始校验:基于大小校验源端与目录端文件,会从源端与目的端全量拉取文件列表对比

管理-重试:迁移文件后有失败记录,通过该按钮将失败文件重新迁移

管理-导出:迁移文件后有失败记录,通过该按钮将失败文件列表导出到目的端bucket的.US3SYNC/details/[jobid]/error.log 位置

管理-删除:删除任务,会清理任务缓存记录

管理-克隆:打开创建任务界面,并填写当前任务信息

- 统计信息如下:

=====
迁移总数: 此次迁移文件总数

等待迁移: 处于待迁移队列的文件个数

正在迁移: 正在执行迁移的文件个数

迁移成功: 已经迁移成功的文件个数

迁移失败: 已经迁移失败的文件个数
=====

- 错误日志导出:

如果您的迁移任务出错, 请您在任务接受后点击图上按钮将错误日志导出, 日志将会被us3sync上传到us3上, 并在界面上提供给您一个可以下载的url。

您可以在节点所在机器上使用wget下载, 并参考: <https://cms-docs.ucloudadmin.com/ufile/tools/us3sync/questions> 进行排查。

创建任务界面说明

创建任务

任务名称:

任务类型: **任务类型选择**

覆盖类型:

源端Bucket:

源端EndPoint:

源站前缀:

正则匹配对象键: [] 不支持用于创建任务或清理源端列表

目标Bucket:

目标EndPoint:

目标前缀:

存储类型:

定期执行: **配置定时执行**

过滤规则:

md5校验:

同步删除: **删除掉目的端存在而源端不存在的文件**

前缀自动拆分:

如果您的目录层级较深，请慎选此功能。此选项将不会同步空文件夹

可以在此处获取jobid来构建Fetch请求

[迁移任务](#) 迁移节点 EndPoint

创建任务

任务名称	源端EndPoint	目的端EndPoint
点击此处以获取任务id		
^ fetch		dst-internal
源站Bucket: 源站Prefix: 目标Bucket: 目标Prefix: 任务ID: __FETCHJOB__-917dcf27-46c5-11ec-bf50-5254007d849f		

创建迁移源界面说明

创建EndPoint ×

名称:

类型:

公钥:

私钥:

上传/下载超时时间: s

存储空间列表:

前缀列表:

EndPoint域名:

使用Https:

使用PrefixFileList列取: 专有云用户请勾选此处

创建节点界面说明

添加节点 ×

ip:

user:

密码:
免密登陆时可以为空

工作路径:

限速(MB/s):

QPS(每秒):

使用说明

添加Endpoint

添加迁移源, 参考 [创建迁移源界面说明](#), 点击创建Endpoint按钮, 在对话框中填写对应信息, 点击确定。

创建任务

创建任务,参考 [创建任务界面说明](#),点击创建任务按钮,在对话框中填写对应信息,点击确定。如果您想要创建fetch任务,请您点击界面上的Fetch任务选项。另外JobId请您参考 [获取任务Id](#)

说明:

迁移过程会将指定的源端前缀下的文件迁移到指定的目的端前缀下面,例如:

1. 源端前缀指定了:A/ 目的端前缀指定了 B/
此时会将源端 A/ 下的所有文件迁移到目的端 B/ 下
如果源端有一个文件A/a.txt 迁移后在目录端会有 B/a.txt
2. 源端前缀指定了:A/ 目的端前缀指定了空路径
此时会将源端 A/ 下的所有文件迁移到目的端根目录下
如果源端有一个文件A/a.txt 迁移后在目录端根目录下会有 a.txt

覆盖条件选项说明,目前支持三种方式:

- > 全覆盖:源端存在的文件全部强制上传到US3
- > 条件覆盖:根据指定字段判断是否需要迁移文件
 - >> 条件覆盖目前支持三个字段:文件类型,文件时间戳,文件大小
 - >> 文件类型和文件大小通过判断 源端文件与目的端文件对应属性是否一致
 - >> 文件时间戳通过判断 源端文件比目的端文件更新则重新迁移该文件
- > 不覆盖:发现目的端已经存在同名文件,跳过该文件

定期任务配置说明:

> 如果需要定期执行迁移,配置每隔几天,什么时间进行迁移

增量过滤条件说明,根据配置过滤待迁移文件,支持两种规则:

1. 根据绝对时间过滤,用户指定起始时间。
2. 根据最近时间过滤,用户指定迁移最近几个小时的文件。如果配置定期任务执行,则每次定期任务启动,都会根据当时时间进行计算。

正则匹配说明:

1. 若配置此项,则拉取源端列表时会匹配正则表示式,只有命中的文件才可能被迁移。
2. 此表达式不会匹配源端前缀,因此和前缀配合使用可以匹配到某前缀下命中该正则表达式的文件。

添加告警

- 当任务的结束,worker宕机通过post请求发送消息
- 收集迁移以及fetch任务的错误数据指标
- 允许用户通过规则字符串的形式配置规则job错误数据告警,规则匹配时,通过post请求发送消息

监控告警模块界面说明

- 创建告警

US3SYNC

[迁移任务](#) [迁移节点](#) [EndPoint](#) [告警](#)创建告警

告警名称	告警类型	告警任务	告警事件	告警条件	告警数据周期	告警目标	操作
test	系统告警		worker宕机			http://10.9.24.35:5000	删除

< 1 >

创建告警



名称:

类型:

选择针对某个job创建告警或者针对集群创建告警

告警事件:

http请求方法:

请求url:

请求头:

消息模版:

创建告警



创建告警

名称:

类型:

任务告警



点击此处编辑复杂逻辑

选择告警逻辑

编辑复杂告警条件:

统计指标



判断逻辑



请输入一个数字

告警周期:

请输入一个数字 一个周期内至多告警一次

分钟

任务名:

选择告警的任务



创建成功

http请求方法:

请求url:

请求头:

添加

消息模版:

请使用%s代表告警信息, eg: 告警! 事件: %s

取消

确认

复杂告警逻辑可以自行编辑, 支持的变量如下:

- RESOURCE_ERR_4XX: 源端4xx错误数
- RESOURCE_ERR_5XX: 源端5xx错误数
- TARGET_ERR_4XX: 目的端4xx错误数
- TARGET_ERR_5XX: 目的端5xx错误数
- LOCAL_ERR: 其它错误数

- Q_LEN: fetch任务正在等待的迁移的文件数量

支持的操作符如下:

- ||: 或逻辑
- &&: 与逻辑
- (): 括号内的判断逻辑优先级最高

例如 (RESOURCE_ERR_4XX > 100 || RESOURCE_ERR_5XX > 200) && TARGET_ERR_4XX > 150 && Q_LEN > 150

告警请求的header可以在如下位置添加

创建告警 ×

名称:

类型:

告警事件:

http请求方法:

请求url:

请求头:

消息模版:

消息模版处可以填写自定义的请求消息模版,使用%s标识告警信息,例如"alert! error: %s" 在告警时会发送 alert! error: number of resource bucket error 4xx is 100, greater than 50

- 总体示例

创建告警



名称: src-internal

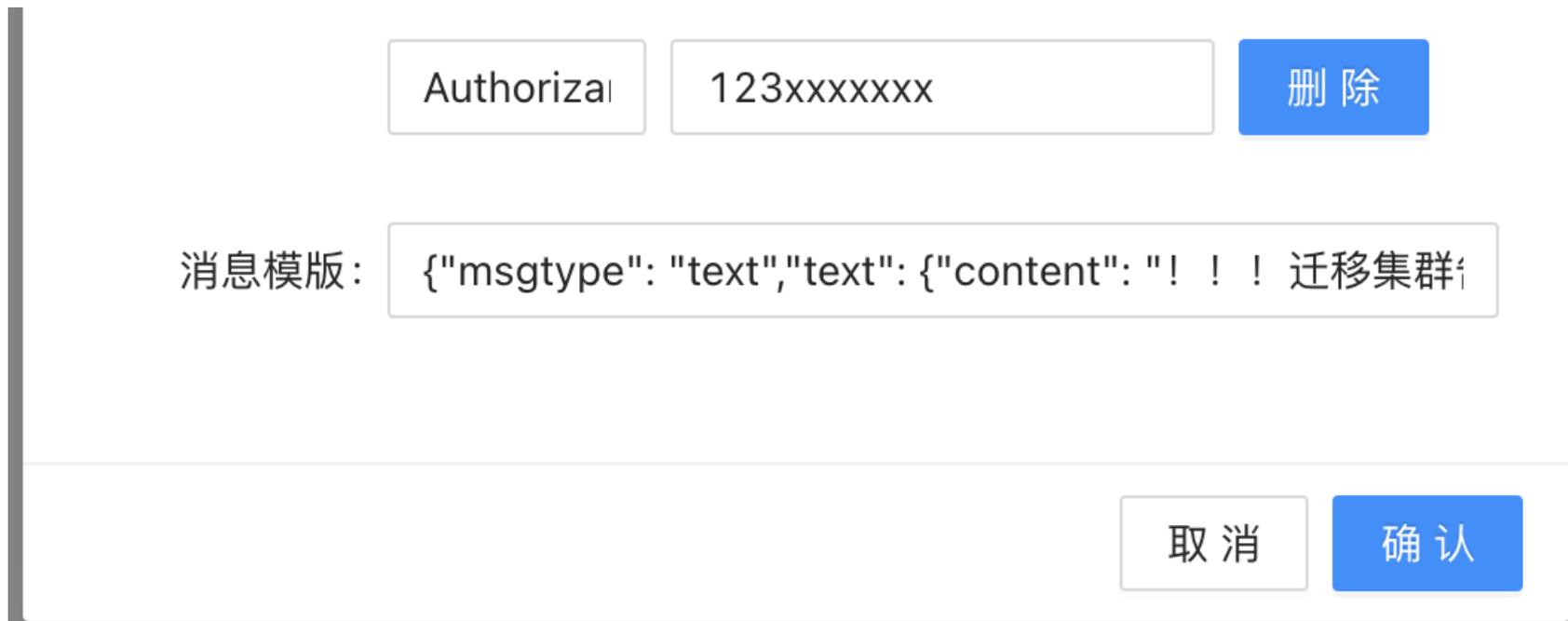
类型: 系统告警

告警事件: 任务结束

http请求方法: POST

请求url: https://localhost:5000

请求头: [添加](#)



Authoriza 123xxxxxxx 删除

消息模版: {"msgtype": "text", "text": {"content": "!!! 迁移集群"}}

取消 确认

启动迁移

参考 [任务管理界面说明](#), 点击开启迁移按钮。

暂停迁移

参考 [任务管理界面说明](#), 点击暂停按钮。

重试

如果迁移后有失败记录, 可以对迁移失败的文件进行重试, 参考 [任务管理界面说明](#), 点击管理-重试按钮。

导出失败文件列表

如果迁移后有失败记录,可以对迁移失败的文件列表导出,参考 [任务管理界面说明](#),点击管理-导出按钮。

缓存说明

US3SYNC 使用 pika 提供缓存服务。主要缓存以下信息:

- 任务详细信息
- 迁移统计信息
- 迁移记录

迁移类型说明

迁移类型	说明
S3	实现基于 S3 接口的迁移,支持 S3 协议的云平台皆可基于此类型迁移
OSS	实现 OSS 云存储接口
qiniu	实现 qiniu 云存储接口
youpai	实现 youpai 云存储接口
us3	实现 us3 云存储接口
NAS	实现 NAS 系统或者本地文件迁移

URL	实现 基于 URL 列表迁移
-----	----------------

S3

AWS S3源需要额外提供待迁移空间的 region 字段,参考:https://docs.aws.amazon.com/zh_cn/general/latest/gr/s3.html。

第三方实现S3协议参考对应厂商S3文档说明。

local

local 源需要提供待迁移目录的路径

http

http 源需要提供待迁移文件列表的 url 地址。

每行代表一个待迁移资源,待迁移文件列表格式如下:

文件url地址

文件url地址

常见问题

master部署后，浏览器无法看到界面

问题原因

Linux防火墙屏蔽了相应的网络端口。

解决方案

检查机器防火墙的web服务监听端口是否开启,将防火墙关闭或对端口设置IP白名单。

问题原因

由于us3sync使用了自签证书,因此不被chrome等浏览器承认。

解决方案

在chrome报错页面上输入thisisunsafe即可。

worker部署后，启动任务后正在迁移的任务一直显示为0

问题原因

worker节点启动失败,通常为worker节点与master节点网络不通导致。

解决方案

通过ping检查网络连通性,检查worker节点IP地址是否配置正确,确保master和worker节点网络连通。

任务有报错,在导出的任务日志中,报错为"**Client.Timeout or context cancellation**"

问题原因

worker节点试图上传/下载文件,但是由于网络原因或者服务端未响应,导致请求超时,从而引起该错误。

解决方案

通过ping或mtr等手段检查endpoint到worker机器的网络连通性,如果网络正常,则可以点击重试按钮,尝试重新传输失败的文件,也可以考虑增加endpoint的超时时间。

在导出的任务日志中,报错状态码为**403**

问题原因

endpoint上配置的公私钥没有上传,下载或列表的权限

解决方案

请您更新下公私钥的权限,确认其带有上传下载和列表权限

v1.10.4

1. 修复了在使用delimit时,导致部分文件没有迁移

v1.10.1

1. 修复在error时worker会panic的错误

v1.10.0

1. 新增告警系统,用户可以在web界面上配置告警

v1.9.1

1. 修复了fetch任务回调请求没有带Content-Type: application/json的错误

v1.9.0

1. 针对body长度不正确的http请求会进行重试
2. 增加节点错误率监控, 以及实时传输速度显示
3. 增加了节点掉线提示, 现在掉线的节点会在web界面上标红

v1.8.0

1. 修复了s3类型endpoint无法使用https的问题
2. 增加了endpoint的请求超时配置

v1.7.0

1. 修复了前缀自动发现式列取时, 校验阶段仅校验顶层前缀的bug
2. 在不勾选前缀自动发现时, 会同步源端的空目录到目的端
3. 支持了七牛专有云的endpoint

v1.6.1

1. 修复了在专有云上无法创建endpoint的bug
2. 修复了使用前缀自动发现功能时可能会卡住的bug
3. 修复了url列表模式的endpoint可能无法创建的bug

v1.6.0

1. 优化了数据的统计方式
2. 修复了在使用s3接口拉取列表时,返回结果中nextmarker为空导致的panic
3. 修复了重试时由于分段上传请求已经被取消而导致的上传id失效
4. 修复了一个list时始终使用同一个marker的bug
5. 新增了如果启动时不设置用户名密码,可以在首次登录界面时设置用户名密码的功能

v1.5.0

1. 新增fetch任务模式,可以通过向server的/fetch路径下发送POST请求来增量同步文件
2. 现在可以在启动时指定--sentinel-addr参数的方式,让节点使用哨兵模式连接pika数据库
3. 支持了在使用aws,us3,oss,七牛以及url列表类型的endpoint时的https下载以及拉列表功能,可以在创建endpoint时进行配置
4. 支持了在使用aws,us3,oss,七牛类型的endpoint拉列表时,自动发现新公共前缀的功能,在这个模式下,拉列表会针对各个前缀并发列取
5. 支持了在使用us3类型的endpoint时使用prefix-file-list接口进行拉列表,该功能通常用在专有云环境
6. 支持了根据初始化分片上传任务时获取的返回结果来选择分片上传的分片大小的功能。
7. 现在源端404的任务同样会被认为是失败
8. 现在worker在丢失掉master心跳后并不会立刻退出,而是会以指数退避的策略放缓心跳,最大心跳间隔为32秒一次,如果重新捕获到主节点心跳,会恢复心跳间隔
9. 现在worker会正确的abort掉无法完成的分片上传任务
10. 现在worker在下载文件时遇到404,503,500,504状态码,会以指数退避的形式重试5次
11. 更改了分配任务给worker的流程,现在worker获取的任务可以在worker宕机后快速恢复
12. 优化web界面,目前可以在任务列取和审计阶段看到文件列表操作的进度
13. 现在可以在创建任务时,使用正则表达式过滤源端对象键,但是此对象键不包括配置在任务中的桶名和前缀

v1.4.0

1. 新增同步删除功能,目前可以在创建任务时勾选此选项,将源端不存在,而目的端存在的文件删除
2. 优化web界面,将同步,删除,以及正在执行的指标分隔显示

v1.3.1

1. 忽略迁移中源端不存在的文件
2. 优化URL列表拉取源列表文件
3. 修复 不覆盖选项导致进程panic

v1.3.0

1. 多DB优化列取方式
2. 优化local源列取
3. 优化http源列取
4. 修复oss FileGroup类型迁移失败
5. 修复空文件迁移失败
6. 非定时任务支持校验新文件

v1.2.0

1. 定时任务取消立即启动
2. 增加MD5校验支持, 创建任务指定
3. endpoint支持prefix校验, 针对指定prefix的token
4. URL列表迁移, 支持跳过404链接
5. 任务分发优化, 迁移速度提升

v1.1.2

1. 修复 NAS,URL 列表类型 endpoint 无法创建
2. 修复 uarchive 类型迁移, mime type 校验失败

v1.1.1

1. 修复任务阻塞问题
2. 修复任务无法正常结束
3. 启动脚本校验预装命令
4. 前端支持必填项校验

v1.1.0

1. 支持迁移源管理
2. 支持定期任务
3. 支持失败文件导出
4. 支持文件覆盖选项
5. 支持限速, QPS

v1.0.0

1. 支持迁移源:s3,oss,qiniu,youpai,US3,NAS,URL列表。
2. 支持 web 管理,通过 web 管理迁移任务,迁移节点。

工具简介

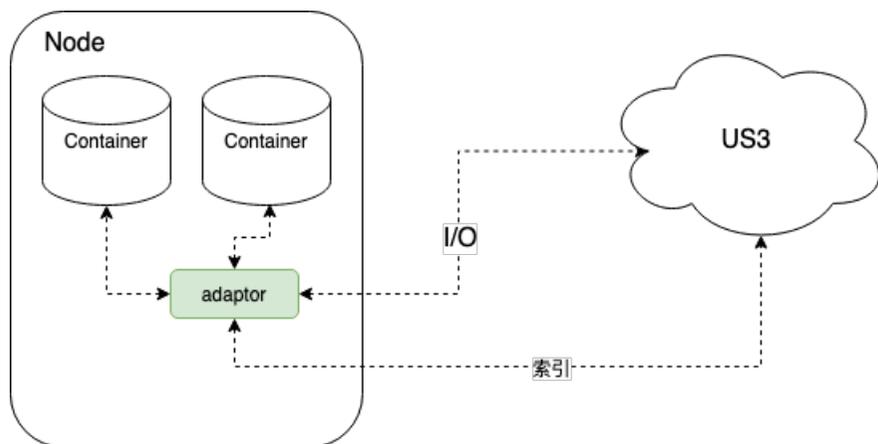
概述

US3Hadoop适配器是为了解决Hadoop访问UCloud对象存储US3的问题,实现了标准的Hadoop文件系统,支持Hive、Spark、Flink等大数据计算框架可以像访问HDFS文件系统一样,来读写存储在US3上的数据。

适配器当前已开源,用户可以通过github获取适配器的源代码。

原理说明

适配器是实现了Hadoop提供的存储访问接口FileSystem的适配器组件,类似HDFS实现的DistributedFileSystem和基于AWS S3协议实现的S3AFileSystem。适配器直接把IO,索引请求发给US3,架构如下图所示。



该架构适合备份场景和小规模计算分析场景。

版本和运行环境

软件版本

当前版本:1.4.1

运行环境

- hadoop-2.6.0
- hadoop-2.8.3
- hadoop-2.8.5
- hadoop-3.1.1

下载与环境准备

下载与安装

获取适配器

US3大数据适配器通过jar包形式提供其功能, US3大数据适配器jar包格式为us3-bigdata-adaptor-\${hadoop version}-\${adaptor version}.jar;

适配器版本

- v1.4.0
 - hadoop-2.6.0
 - hadoop-2.8.3
 - hadoop-2.8.5
 - hadoop-3.1.1

如有其他版本**Hadoop**对接需求, 请联系技术支持。

安装适配器

1. 配置各个节点的core-site.xml参数项, 参数项参考快速上手-参数说明;

2. 将us3-bigdata-adaptor-\${hadoop version}-\${adaptor version}.jar拷贝到\$HADOOP_HOME/share/hadoop/common/lib/下;

接入方式具有侵入性，适用于小规模计算分析场景，大数据备份需求场景请参考快速上手-场景示例。

快速上手

- 视频教学
- 使用说明
 - 参数说明
 - Hadoop配置
 - 使用方式
- 场景示例

视频教学

观看以下视频,快速上手使用US3Hadoop

该内容为动态文件,无法展示

使用说明

参数说明

属性Key	属性说明	默认值
-------	------	-----

fs.us3.impl	US3对FileSystem的实现类名,请确保为:cn.ucloud.us3.fs.US3FileSystem	无
fs.AbstractFileSystem.us3.impl	对AbstractFileSystem 的实现类,固定为 cn.ucloud.us3.fs.US3Fs。	无
fs.us3.access.key	访问US3的API公钥或者Token公钥	无
fs.us3.secret.key	访问US3的API私钥或者Token私钥	无
fs.us3.socket.receive.buffer	US3读取流Buffer,在HiveSQL、Spark-SQL等场景,如果分析的数据格式是ORC, Parquet等,建议调小该参数,比如16348(16KB),如果分析的数据格式是普通的文本格式,建议调大该参数,比如1048576(1MB)	65536(64KB)
fs.us3.log.level	US3大数据适配器日志级别,支持info, error, debug, trace级别。目前日志统一输出到标准输出	info
fs.us3.endpoint	US3内网域名后缀,如:ufile.cn-north-02.ucloud.cn。参考地域和域名,域名需去除www.前缀。	无
fs.us3.async.wio.use	单个流写入是否使用异步IO,可以提高写入速度,特别是对单个大文件有效,但会消耗一定CPU资源,可能会降低任务的并行度	false
fs.us3.async.wio.parallel	在fs.us3.async.wio.use为true情况下生效,表示单个文件写入4MB分片的最大并行	2
fs.us3.metadata.use	是否启用索引缓存服务器,来加速US3索引性能,该服务需要使用者管理,目前正在公测中	false

fs.us3.metadat a.host	在fs.us3.metadata.usetrue情况下生效,可以直接指定IP:Port的地址形式,这种方式不需要配置/etc/hosts文件。用户也可以指定自定义域名,需要在/etc/hosts中配置解析地址,或者配置DNS解析。该参数处于测试中。	无
fs.us3.metadat a.zookeeper.ad drs	高可用vmds集群方案中使用的zookeeper节点信息,参数格式为以","分割的zookeeper服务器地址和端口号,例如:uhadoop-q4mbrxk2-master1:2181,uhadoop-q4mbrxk2-master2:2181,uhadoop-q4mbrxk2-core1:2181。如果您配置了fs.us3.metadata.host参数,则本参数无效。该功能需要 US3适配器版本>=1.3.0 。	无
fs.us3.generat e.md5	默认为false,如果为true则在写入US3时,客户端会计算一个MD5,并在最后会以md5-hash为key,base64编码的md5值为value写入到文件的元数据中。开启会增加>=30%的写入时延。 该功能需要 US3适配器的版本>=1.0.2	false
fs.us3.retryTim es	请求重试次数 该功能需要 US3适配器的版本>=1.4.0	30
fs.us3.timeout	请求超时时间 该功能需要 US3适配器的版本>=1.4.0	11

Hadoop配置

1 需要在core-site.xml中配置以上相关配置项,如:

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
```

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License. See accompanying LICENSE file.

-->

```
<configuration>
```

```
<property>
```

```
<name>fs.defaultFS</name>
```

```
<value>hdfs://192.168.0.221:9000</value>
```

```
</property>
```

.....

```
<property>
```

```
<name>dfs.datanode.data.dir</name>
```

```
<value>/data/hd3/datanode</value>
```

```
</property>
```

```
<property>
```

```
<name>fs.us3.impl</name>
```

```
<value>cn.ucloud.us3.fs.US3FileSystem</value>
```

```
</property>
```

```
<property>
```

```
<name>fs.AbstractFileSystem.us3.impl</name>
```

```
<value>cn.ucloud.us3.fs.US3Fs</value>
</property>
<property>
<name>fs.us3.access.key</name>
<value>TOKEN_XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX</value>
</property>
<property>
<name>fs.us3.secret.key</name>
<value>XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX</value>
</property>
<property>
<name>fs.us3.socket.recv.buffer</name>
<value>16348</value>
</property>
<property>
<name>fs.us3.thread.pool.size</name>
<value>8</value>
</property>
<property>
<name>fs.us3.log.level</name>
<value>info</value>
</property>
<property>
<name>fs.us3.endpoint</name>
<value>ufile.cn-north-02.ucloud.cn</value>
```

```
</property>
<property>
<name>fs.us3.async.wio.use</name>
<value>>false</value>
</property>
<property>
<name>fs.us3.async.wio.parallel</name>
<value>2</value>
</property>
<property>
<name>fs.us3.metadata.use</name>
<value>>false</value>
</property>
<property>
<name>fs.us3.metadata.host</name>
<value></value>
</property>
<property>
<name>fs.us3.metadata.zookeeper.addrs</name>
<value>uhadoop-q4mbrxk2-master1:2181,uhadoop-q4mbrxk2-master2:2181,uhadoop-q4mbrxk2-core1:2181</value>
</property>
<property>
<name>fs.us3.retryTimes</name>
<value>5</value>
</property>
```

```
<property>
<name>fs.us3.timeout</name>
<value>60</value>
</property>
.....
</configuration>
```

使用方式

Hadoop文件系统访问

使用格式为 `hadoop fs -ls us3://<bucket name>/<path>`。使用式样如下：

```
[root@task10 ~]# hadoop fs -ls us3://bigdata-us3/HiBench/Aggregation/Input/rankings | more
Found 10241 items
-rw-r--r-- 1 root supergroup 0 2021-01-26 13:46 us3://bigdata-us3/HiBench/Aggregation/Input/rankings/_SUCCESS
-rw-r--r-- 1 root supergroup 6375702 2021-01-26 13:30 us3://bigdata-us3/HiBench/Aggregation/Input/rankings/part-00000
-rw-r--r-- 1 root supergroup 6379753 2021-01-26 13:30 us3://bigdata-us3/HiBench/Aggregation/Input/rankings/part-00001
-rw-r--r-- 1 root supergroup 6372962 2021-01-26 13:30 us3://bigdata-us3/HiBench/Aggregation/Input/rankings/part-00002
.....
```

其他使用方式同 **hdfs** 操作

Hive使用US3

在创建外部表的时候指定location 为对应的US3路径即可,如... location 'us3://<bucket name>/hive/tpc-ds/1002-orc/customer'

Spark使用US3

无论是通过spark-submit还是pyspark、spark-shell、spark-sql等读写US3时,需要指定文件路径的格式为: us3://<bucket name>/path

Flink使用US3

只需指定US3格式的路径即可;

注意

- 已有存储空间的文件对象会返回默认权限;
- 不支持包含特殊字符的文件名,如:;

场景示例

假设需要把HDFS集群的/var目录数据备份至US3的bigdata-us3存储空间的/distcp/前缀下,可以在安装好前面的适配器后通过hadoop自带的distcp执行拷贝,如:

```
[root@master1 tools]# hadoop distcp -bandwidth 15 -m 20 -skipcrccheck -update -pugp /var us3://bigdata-us3/distcp/var
21/01/26 16:51:50 INFO tools.DistCp: Input Options: DistCpOptions{atomicCommit=false, syncFolder=true, deleteMissing=false, ignoreFailures=false,
overwrite=false, skipCRC=true, blocking=true, numListstatusThreads=0, maxMaps=20, mapBandwidth=15, sslConfigurationFile='null',
copyStrategy='uniformsize', preserveStatus=[USER, GROUP, PERMISSION], preserveRawXattrs=false, atomicWorkPath=null, logPath=null,
sourceFileListing=null, sourcePaths=[/var], targetPath=us3://bigdata-us3/distcp/var, targetPathExists=false, filtersFile='null'}
21/01/26 16:51:51 INFO tools.SimpleCopyListing: Paths (files+dirs) cnt = 9226; dirCnt = 851
21/01/26 16:51:51 INFO tools.SimpleCopyListing: Build file listing completed.
```

```
21/01/26 16:51:51 INFO Configuration.deprecation: io.sort.mb is deprecated. Instead, use mapreduce.task.io.sort.mb
21/01/26 16:51:51 INFO Configuration.deprecation: io.sort.factor is deprecated. Instead, use mapreduce.task.io.sort.factor
21/01/26 16:51:51 INFO tools.DistCp: Number of paths in the copy list: 9226
21/01/26 16:51:52 INFO tools.DistCp: Number of paths in the copy list: 9226
21/01/26 16:51:53 INFO mapreduce.JobSubmitter: number of splits:25
21/01/26 16:51:53 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1611108331895_0847
21/01/26 16:51:54 INFO impl.YarnClientImpl: Submitted application application_1611108331895_0847
21/01/26 16:51:54 INFO mapreduce.Job: The url to track the job: http://master2:23188/proxy/application_1611108331895_0847/
21/01/26 16:51:54 INFO tools.DistCp: DistCp job-id: job_1611108331895_0847
21/01/26 16:51:54 INFO mapreduce.Job: Running job: job_1611108331895_0847
21/01/26 16:52:01 INFO mapreduce.Job: Job job_1611108331895_0847 running in uber mode : false
21/01/26 16:52:01 INFO mapreduce.Job: map 0% reduce 0%
21/01/26 16:52:18 INFO mapreduce.Job: map 1% reduce 0%
21/01/26 16:52:19 INFO mapreduce.Job: map 3% reduce 0%
21/01/26 16:52:20 INFO mapreduce.Job: map 6% reduce 0%
21/01/26 16:53:07 INFO mapreduce.Job: map 7% reduce 0%
21/01/26 16:53:31 INFO mapreduce.Job: map 8% reduce 0%
...
```

注意

- `-p`参数不要带上**b**, 因为US3实际的块大小跟HDFS不一致, 如果重新执行该命令会导致已有文件会重新被覆盖上传;
- `-skipcrccheck -update`请保证开启, 因为US3的完整性校验通过另一套机制完成;

如果不想安装适配器即可进行备份或者文件系统操作(避免侵入已有服务带来的风险), 可以按照如下格式进行执行操作:

```
hadoop fs|distcp|... -Dfs.us3.impl=cn.ucloud.us3.fs.US3FileSystem \  
-Dfs.us3.access.key=${访问US3的API公钥或者Token公钥} \  
-Dfs.us3.secret.key=${访问US3的API私钥或者Token私钥} \  
-Dfs.us3.socket.recv.buffer=65536 \  
-Dfs.us3.log.level=${日志级别} \  
-Dfs.us3.endpoint=${访问点} \  
-Dfs.us3.async.wio.use=${是否开启写入异步IO} \  
-Dfs.us3.async.wio.parallel=${异步写入IO并行度} \  
-Dmapreduce.job.user.classpath.first=true \  
-Dmapreduce.task.classpath.user.precedence=true \  
-libjars ${适配器jar包路径} \  
${对应hadoop子命令的执行参数} \  

```

配置参数参考参数说明

如拉取US3的某个目录:

```
[root@master1 tools]# hadoop fs -Dfs.us3.impl=cn.ucloud.us3.fs.US3FileSystem -Dfs.us3.access.key=TOKEN_7468973e-d192-xxxx-xxxx-xxxxxxxxxxxxx -  
Dfs.us3.secret.key=9a632eb8-3938-xxxx-xxxx-xxxxxxxxxxxxx -Dfs.us3.socket.recv.buffer=65536 -Dfs.us3.thread.pool.size=8 -Dfs.us3.log.level=info -  
Dfs.us3.endpoint=internal-cn-bj.ufileos.com -Dfs.us3.bucket.bigdata.only=false -Dfs.us3.async.wio.use=true -Dfs.us3.async.wio.parallel=2 -  
Dmapreduce.job.user.classpath.first=true -Dmapreduce.task.classpath.user.precedence=true -libjars ./us3-bigdata-adaptor-2.8.5.0.jar -ls us3://bigdata-  
us3/distcp/var/log/hadoop-yarn/apps/root/logs  
Found 846 items  
drwxrwxrwx - root supergroup 0 2021-01-26 16:52 us3://bigdata-us3/distcp/var/log/hadoop-yarn/apps/root/logs/application_1611108331895_0001  
drwxrwxrwx - root supergroup 0 2021-01-26 16:53 us3://bigdata-us3/distcp/var/log/hadoop-yarn/apps/root/logs/application_1611108331895_0002
```

```
drwxrwxrwx - root supergroup 0 2021-01-26 16:52 us3://bigdata-us3/distcp/var/log/hadoop-yarn/apps/root/logs/application_1611108331895_0003
drwxrwxrwx - root supergroup 0 2021-01-26 16:53 us3://bigdata-us3/distcp/var/log/hadoop-yarn/apps/root/logs/application_1611108331895_0004
drwxrwxrwx - root supergroup 0 2021-01-26 16:53 us3://bigdata-us3/distcp/var/log/hadoop-yarn/apps/root/logs/application_1611108331895_0005
drwxrwxrwx - root supergroup 0 2021-01-26 16:53 us3://bigdata-us3/distcp/var/log/hadoop-yarn/apps/root/logs/application_1611108331895_0006
drwxrwxrwx - root supergroup 0 2021-01-26 16:53 us3://bigdata-us3/distcp/var/log/hadoop-yarn/apps/root/logs/application_1611108331895_0007
...
```

常见问题

对US3进行HDFS命令操作时报错

如报错信息为Is: No FileSystem for scheme: us3, 请检查配置项fs.us3.impl是否配置。

对US3进行操作时报错

如报错信息为java.lang.RuntimeException: java.lang.ClassNotFoundException: Class cn.ucloud.us3.fs.US3FileSystem not found, 请检查适配器jar包是否放置在Hadoop查找的CLASS_PATH路径中。

v1.4.1

修复

- 重新打开流时发生offset错误

v1.4.0

新增

- 适配器增加超时和重试的配置项
- 现在会在发现下载流断开时重新打开下载流

v1.3.0

新增

- 适配器增加从zookeeper获取vmds集群主节点信息的功能

v1.2.0

重构

- 使用线上2.6.6版本的us3 sdk

v1.1.0

功能

- 支持distcp校验拷贝功能

修复

- 日志格式打印错误

v1.0.2

功能

- 支持开启md5计算,并保留其经过base64编码后结果到文件的元数据中

修复

- listStatus的Path是文件时,会返回空列表,正确的语义应该是只包含这个文件的列表
- 写入流的字节统计不准确
- 重命名时有相同文件名失败
- 列表服务响应变化导致的确保一致性误判而超时和目录偏移越界的问题
- 在缺失以/结尾的目录文件时,getFilsStatus触发的dead loop问题
- rename的目的端存在时抛出FileAlreadyExistsException,语义应该为false

优化

- 通知us3vmds逻辑从写入流解耦

v1.0.1

功能

- getFileStatus
- listStatus
- mkdirs
- rename
- delete
- create
- open
- setOwner
- setPermission
- setReplication
- close
- write of OutputStream
- close of OutputStream
- read of InputStream
- seek of InputStream
- getPos of InputStream
- close of InputStream

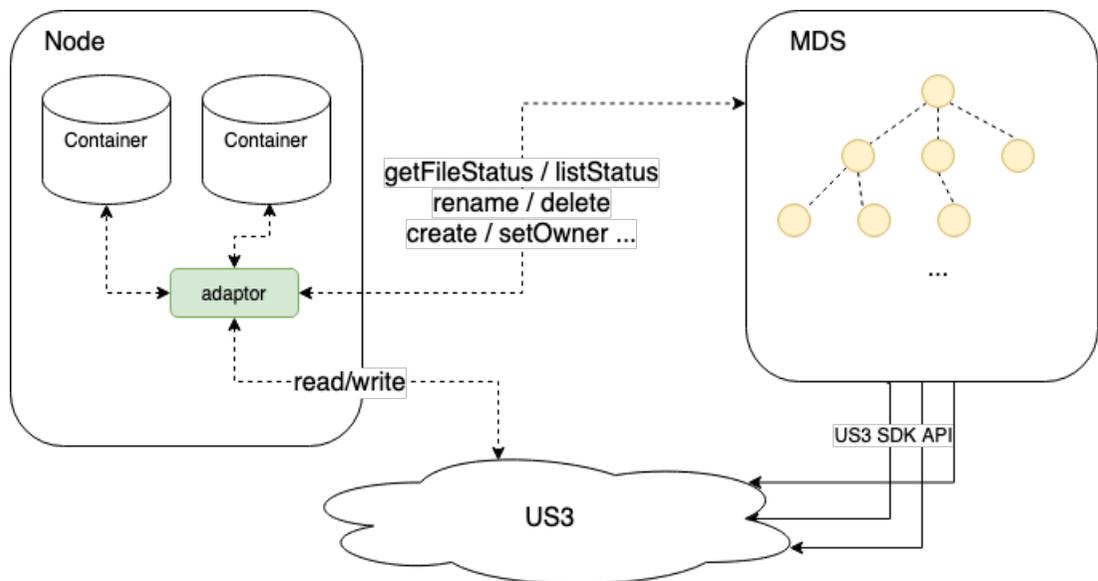
工具简介

概述

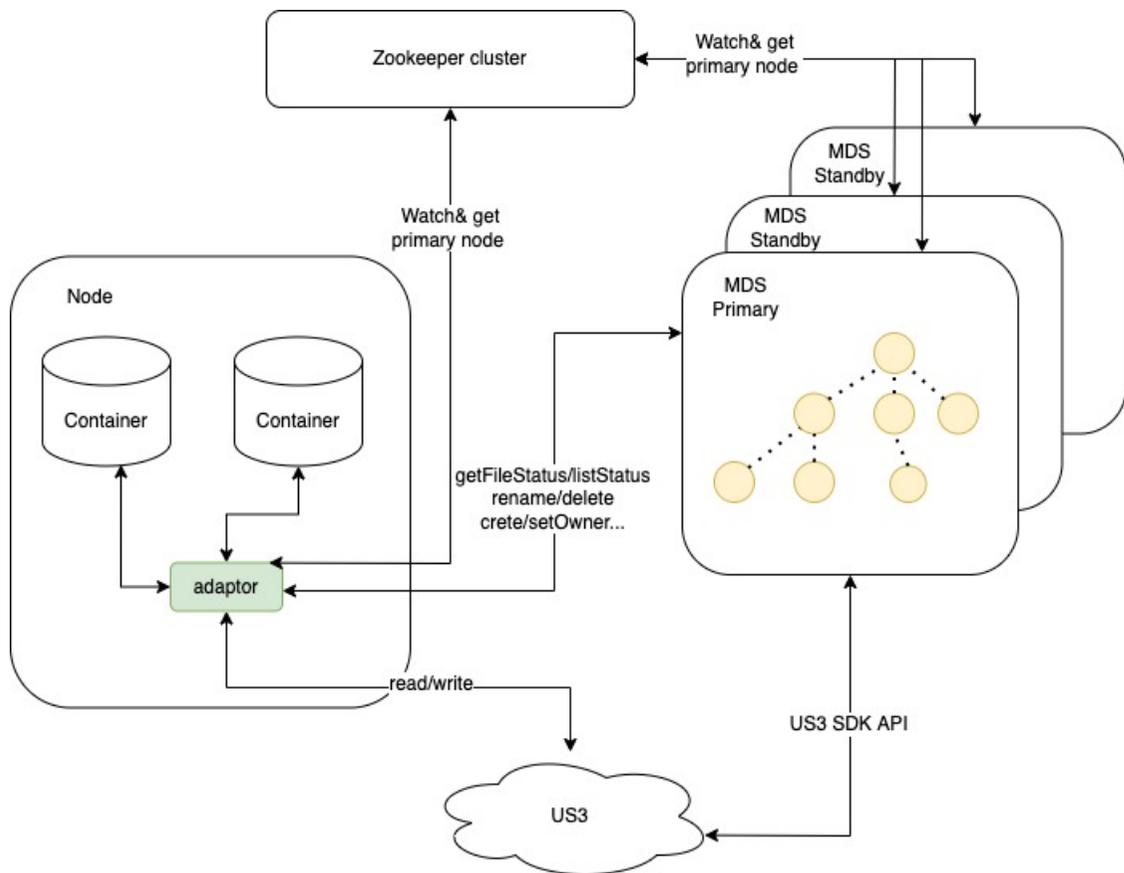
US3VMDS (全称:US3 Virtual Metadata Service) 工具是为了解决Hadoop访问UCloud对象存储US3索引速度较慢的问题而引入的一个服务组件。因为Hadoop对分布式文件系统的访问是基于目录树的结构,而对象存储的索引只是简单的KV结构,所以直接通过US3来模拟目录树的移动目录,删除目录等操作时时延高。US3VMDS在内存中基于US3的存储空间索引构建目录树结构,这样客户端可以通过单次交互来实现复杂的逻辑,并可以利用全内存的索引加速如Hadoop对分布式文件系统的getFileStatus, listStatus等操作。

原理说明

US3VMDS实现了us3适配器用到的Head、PUT、ListObjects、Delete、Rename、Copy几个API来处理索引请求,并在内存中构建了目录树,其位置类似HDFS实现的NameNode。US3适配器IO请求则直接发给US3,只是Write IO成功后会通知US3VMDS(时延在150us左右,影响可以忽略不计)。整个架构如下图所示。



大于等于v0.1.0版本的US3VMDS同样支持高可用集群部署,集群中同时存在一个主节点和多个备节点。主节点会如上面描述的那样正常提供缓存服务,备节点则仅会处理索引请求,但是不会将结果在内存中缓存。集群中的各个节点以及US3Hadoop适配器会监听zookeeper中的US3VMDS节点信息变动,一旦发现主节点离线(主节点崩溃,网络不通等问题),则将一个备用节点晋升为主节点来提供缓存服务。架构如下图:



该架构适合加速备份场景和中小规模计算分析场景。

运行环境

- Linux
- Mac

下载与环境准备

下载与安装

获取可执行文件

us3vmds通过二进制可执行文件提供,文件格式为us3vmds-`{version}`-`{OS}`;

公测版本

- us3vmds-v0.0.1-linux
- us3vmds-v0.0.1-macOS
- us3vmds-v0.1.0-linux
- us3vmds-v0.1.0-macOS

正式版本

待发布...

注意: 对应的US3适配版本需要 \geq v1.0.2

安装可执行文件

1. 下载可执行文件并移动到/usr/bin目录下,如`mv us3vmds-v0.0.1 /usr/bin;`
2. 为最新的us3vmds建立软链接,如`ln -s /usr/bin/us3vmds-v0.0.1 /usr/bin/us3vmds;`
3. us3vmds配置参数,参考以下【使用说明】;
4. us3适配器配置参数,参考文档US3Hadoop 大数据适配器工具 中`fs.us3.metadata.use`和`fs.us3.metadata.host`参数项配置说明;

建议: 部署节点选择**NameNode**节点

高可用集群部署

注意: 对应的**US3**适配器版本需要 \geq **v1.3.0**, **vmds**本身的版本需要 \geq **v0.1.0**, 另外由于容器需要去**zookeeper**上获取**vmds**节点信息, 高可用部署会造成一定的性能损失, 与容器数目相关, 与数据量关系不大, 通常是分钟级别的性能下降。

步骤:

1. 根据上文配置好相应的jar包, 以及相应的core-site.xml配置文件。US3Hadoop大数据适配器默认会根据如下优先级寻找vmds服务节点信息:

1. 在core-site.xml文件中配置的`fs.us3.metadata.host`参数, 直接使用standalone单节点vmds服务。
2. 根据在core-site.xml文件中配置的`fs.us3.metadata.zookeeper.addrs`获取zookeeper信息, 从zookeeper中获取vmds服务主节点信息。
3. 根据core-site.xml文件中的`ha.zookeeper.quorum`参数, 使用hadoop中自带的zookeeper, 并从其中获取vmds服务主节点信息。

这里我们直接使用hadoop中自带的zookeeper。

2. 启动vmfs服务。启动参数之中增加-z选项,并附加zookeeper的host:port信息(如果您不提供port,则会使用默认的zookeeper端口2181)。如果您有多台zookeeper则需要重复使用-z选项。例如:

```
./us3vmfs- start -z 10.0.0.1 -z 10.0.0.2 -z 10.0.0.3
```

这里的zookeeper节点信息可以在core-site.xml文件中使用ha.zookeeper.quorum参数提供的地址。注意如果您不提供-z选项,则zookeeper将使用单节点模式。

3. 到这里高可用的vmfs服务集群就启动好了,您可以使用hadoop fs相关命令进行测试。

快速上手

- 使用说明
 - 参数配置
 - 开始使用

使用说明

参数配置

有两种参数配置方式。通过交互式命令配置：

```
→ bin X us3vmds config
```

```
输入监听IP地址[当前为: 127.0.0.1]:192.168.185.220 // 如果需要修改地址则填入修改值, 否则直接输入enter采用默认值127.0.0.1
```

```
监听IP地址修改为: 192.168.185.220
```

```
输入监听端口[当前为: 5888]:6666 // 如果需要修改端口则填入修改值, 否则直接输入enter采用默认值5888
```

```
监听IP地址修改为: 6666
```

```
输入API/Token公钥[当前为: ]:xxxx-xxxx-xxxx-xxxx // 如果需要修改公钥则填入修改值, 否则直接输入enter, 默认为空
```

```
API/Token公钥修改为: xxxx-xxxx-xxxx-xxxx
```

```
输入API/Token私钥[当前为: ]:yyyy-yyyy-yyyy-yyyy // 如果需要修改私钥则填入修改值, 否则直接输入enter, 默认为空
```

```
API/Token私钥修改为: yyyy-yyyy-yyyy-yyyy
```

```
输入存储空间Bucket[当前为: ]:bigdata-us3 // 如果需要修改存储空间填入修改值, 否则直接输入enter, 默认为空
```

```
存储空间Bucket修改为: bigdata-us3
```

输入访问Endpoint[当前为:]:ufile.cn-north-02.ucloud.cn // 如果需要修改endpoint填入修改值, 否则直接输入enter, 默认为空

访问Endpoint修改为: ufile.cn-north-02.ucloud.cn

输入日志路径[当前为: /Users/rick.wu/.us3vmds]:/var/log/us3vmds // 如果需要修改日志路径填入修改值, 否则直接输入enter, 默认为/Users/rick.wu/.us3vmds

日志路径修改为: /var/log/us3vmds

输入日志级别(debug|error|info)[当前为: info]:debug // 如果需要修改日志级别则填入修改值, 否则直接输入enter采用默认值info, 建议始终采用debug级别, 方便问题跟踪

日志级别修改为: debug

输入热点路径同步US3周期(单位:s)[当前为: 180 s]:3600 // 如果需要修改热点路径自动同步周期值则填入修改值, 否则直接输入enter采用默认值180, 如果只考虑分析从大数据集群写入的数据时, 建议调大该值, 避免频繁同步带来的性能开销

热点路径同步US3周期: 3600

通过文件配置, us3vmds使用配置文件路径始终在\$HOME/.us3vmds/us3vmds.yaml, 配置参数有:

属性Key	属性说明	默认值
accesskey	访问US3的API公钥或者Token公钥	无
secretkey	访问US3的API私钥或者Token私钥	无
bucket	存储空间名	无
endpoint	US3内网域名后缀, 如:ufile.cn-north-02.ucloud.cn。参考 这里 , 注意要去掉“www.”。	无
host	us3vmds监听的IP地址	127.0.0.1
port	us3vmds监听的端口	80
logdir	输出的日志路径, 日志名固定为us3mvds.log或者历史日志名格式如us3vmds-2021-02-23T02-32-21.424.log, 建议该日志路径至少有100G以上存储容量	\$HOME/.us3vmds
loglevel	日志基本, 可填:debug, error, info, 建议目前调成debug方便后续问题跟踪	info

hotspotsyncperiod	热点路径同步周期,如果只考虑分析从大数据集群写入的数据时,建议调大该值,避免频繁同步带来的性能开销	180(单位:s)
isbigdataonly	如果只考虑分析从大数据集群写入的数据时,强烈建议该参数置为true以提高getFileStatus性能,该参数在交互式命令中不显示需要手动修改配置文件	false

对应的配置文件例子如下:

```
→ bin X cat $HOME/.us3vmds/us3vmds.yaml
accesskey: TOKEN_7468973e-d192-4378-8253-xxxxxxx
bucket: bigdata-us3
endpoint: internal-cn-bj.ufileos.com
host: 172.16.16.237
hotspotsyncperiod: 3600
logdir: /data/us3vmds/log
loglevel: debug
port: 80
secretkey: 9a632eb8-3938-43a0-a7e1-xxxxxxx
isbigdataonly: true
```

还可以通过命令展示配置信息:

```
→ bin X us3vmds config --cat
配置路径为: /root/.us3vmds/us3vmds.yaml
监听IP地址为: 172.16.16.237
```

监听端口为: 80

API/Token公钥: TOKEN_7468973e-d192-4378-8253-xxxxxxx

API/Token私钥: 9a632eb8-3938-43a0-a7e1-xxxxxxx

存储空间Bucket: bigdata-us3

访问Endpoint: internal-cn-bj.ufileos.com

日志路径为: /data/us3vmds/log

日志级别为: debug

热点路径同步US3周期: 3600s

开始使用

后台启动

```
→ bin X us3vmds start
```

停止重启

```
→ bin X us3vmds restart
```

停止

```
→ bin X us3vmds stop
```

如果需要调试,也可以:

前台启动

```
→ bin X us3vmds start -f
2021-02-23T20:10:19.454+0800 DEBUG mds/cache.go:49 [uuid:70a82150-3bee-4bda-8bb6-01401a05c1e8] insert / to cach
2021-02-23T20:10:19.455+0800 INFO mds/sync_cache.go:418 hotspot sync period set 3600 seconds
...
```

当启动了us3vmds服务后,通过hadoop操作US3,并享受us3vmds带来的索引加速能力。除了以上功能操作,us3vmds还有以下功能:

热点路径强制同步

```
→ bin X us3vmds sync //HiBench2 // 这里强制us3vmds对/和/HiBench2目录进行同步
```

注意: 该同步路径必须已经在**us3vmds**中加载过, 如果不确定可以先通过**hadoop**命令对该路径进行拉取

性能数据与分析

us3vmds基于Go语言开发, 自带了Go的profile功能, 通过以下命令, 可以在指定地址开启Go的运行时性能分析HTTP服务:

```
→ bin X us3vmds pprof --open 127.0.0.1 8081
→ bin X netstat -tunpl | grep 8081
tcp 0 0 127.0.0.1:8081 0.0.0.0:* LISTEN 3312/us3vmds
→ bin X us3vmds pprof --close // 关闭运行时性能分析HTTP服务
→ bin X netstat -tunpl | grep 8081
```

Go的运行时分析可以参考Go的官方文档Package pprof

测试环境

硬件环境

部署服务	数量	操作系统	CPU	内存	数据盘 (数量/单盘容量/规格)
NameNode		Centos 7.6 64 位	32C	128G	1/100G/RSSD云盘
ResourceManager<>Zookeeper	3				
JournalNode					

DataNode	10	Centos 7.6 64	32C	128G	10/300G/RSSD云盘
NodeManager		位			

软件环境

软件	版本号	限制约束
Hadoop	2.8.5	标准社区版本
Hive	2.3.7	标准社区版本
Spark	2.4.3	标准社区版本
Zookeeper	3.5.6	标准社区版本

关键参数

Hadoop

配置文件	配置项	配置值
hdfs-site.xml	dfs.replication	3

HiBench测试

HiBench是 Intel 开源的大数据基准测试工具,可帮助评估速度,吞吐量和系统资源利用率方面的不同大数据框架。它包含一组Hadoop, Spark和流工作负载,包括 Sort, WordCount, TeraSort, Repartition, Sleep, SQL, PageRank, Nutch indexing, Bayes, Kmeans, NWeight和enhanced DFSIO等。它还包含一些用于Spark流的流工作负载, Flink, Storm和Gearpump。HiBench中总共有29个工作负载。工作负载分为6类,分别是micro, ml (machine learning), sql, graph, websearch和stream。

本次HiBench测试的版本为：

v7.0.0

公共配置信息为：

配置文件	配置项	配置值
hibench.conf	hibench.scale.profile	mydata
hibench.conf	hibench.default.map.parallelism	40960
hibench.conf	hibench.default.shuffle.parallelism	10240
spark.conf	hibench.yarn.executor.num	50
spark.conf	hibench.yarn.executor.cores	4
spark.conf	spark.executor.memory	16g
spark.conf	spark.driver.memory	8g

测试数据

测试选择了其中的7个工作负载进行,且分别基于us3(只采用适配器工具)、us3vmds(适配器和us3vmds)、hdfs三种底层存储进行：

Micro

sort

测试参数调整为：

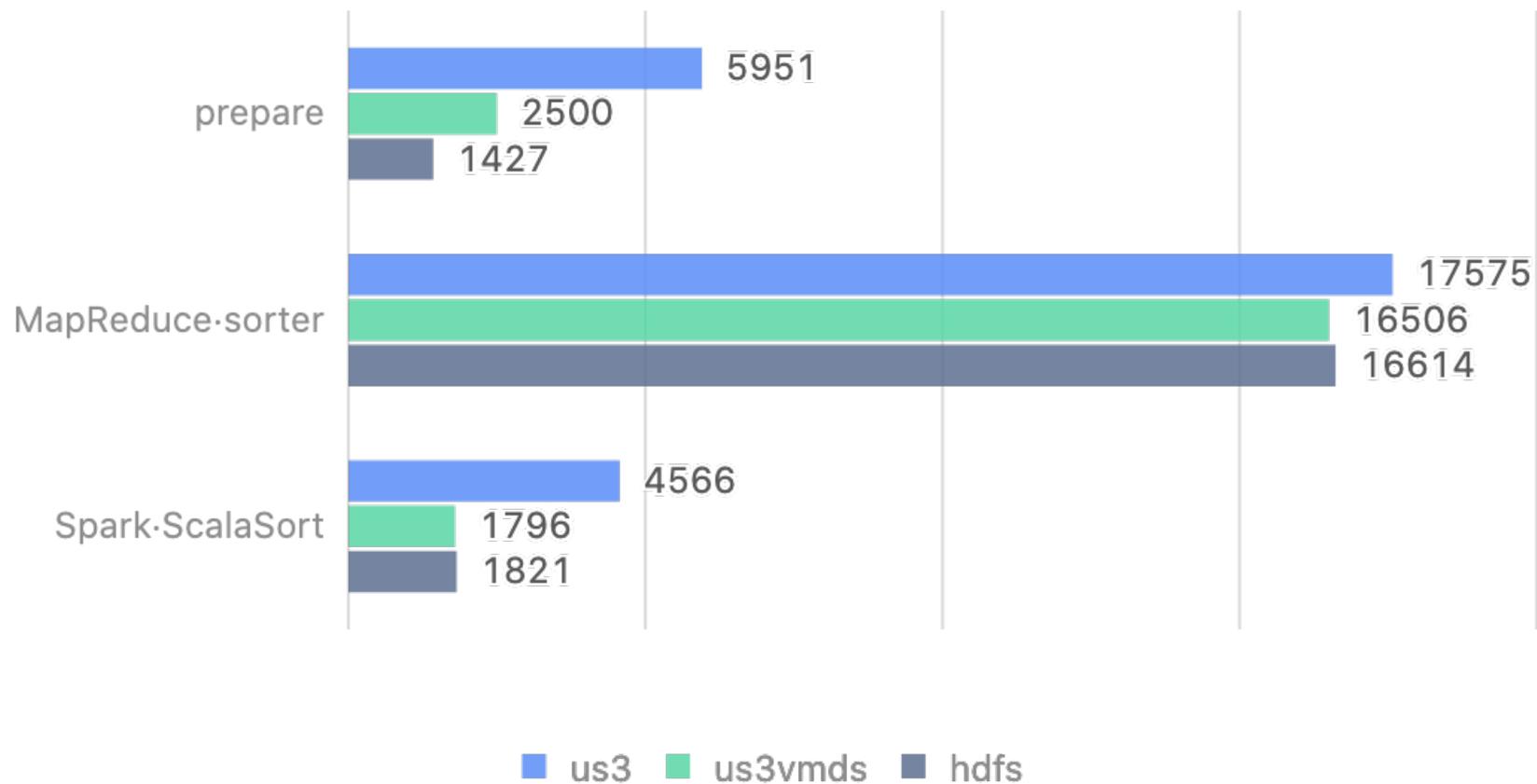
- hibench.sort.mydata.datasize: 3000000000000

该参数表示总体数据量,约为3TB

测试数据:

HiBench

micro.sort工作负载测试(单位:s)



terasort

测试参数调整为:

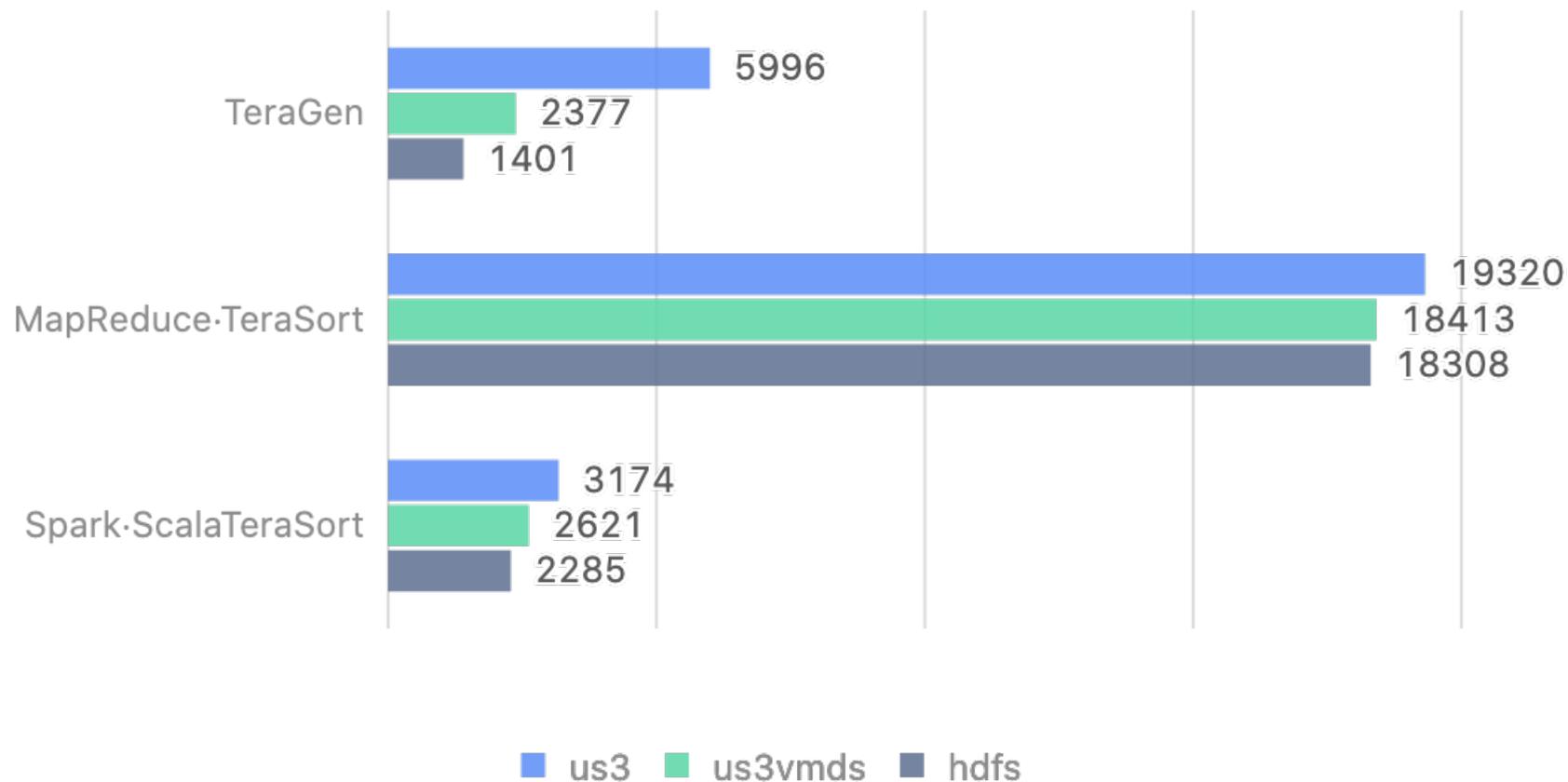
- hibench.sort.mydata.datasize: 30000000000

该参数表示生成的多少行数据,结合源码可知,随机生成每行数据大小为100字节,所以为了跟micro.sort总体测试数据量保持一致设置为30000000000。

测试数据:

HiBench

micro.terasort工作负载测试(单位:s)



wordcoun

测试参数调整为:

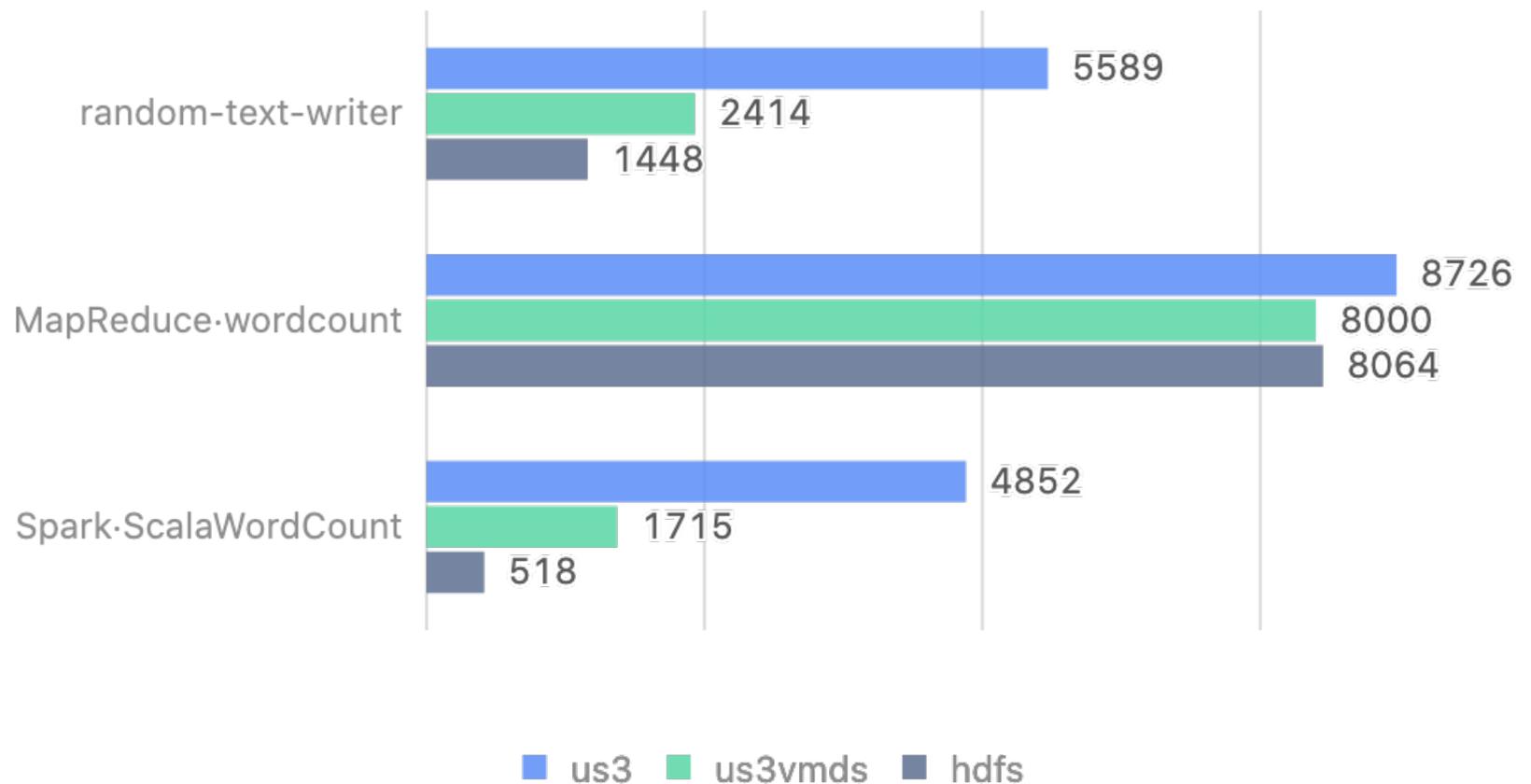
- hibench.dfsioe.mybigdata.read.number_of_files 44704
- hibench.dfsioe.mybigdata.read.file_size 64 // 64MiB
- hibench.dfsioe.mybigdata.write.number_of_files 44704
- hibench.dfsioe.mybigdata.write.file_size 64 // 64MiB

测试的读写数据量约为3TB

测试数据:

HiBench

micro.wordcount工作负载测试(单位:s)



dfsioe

测试参数调整为:

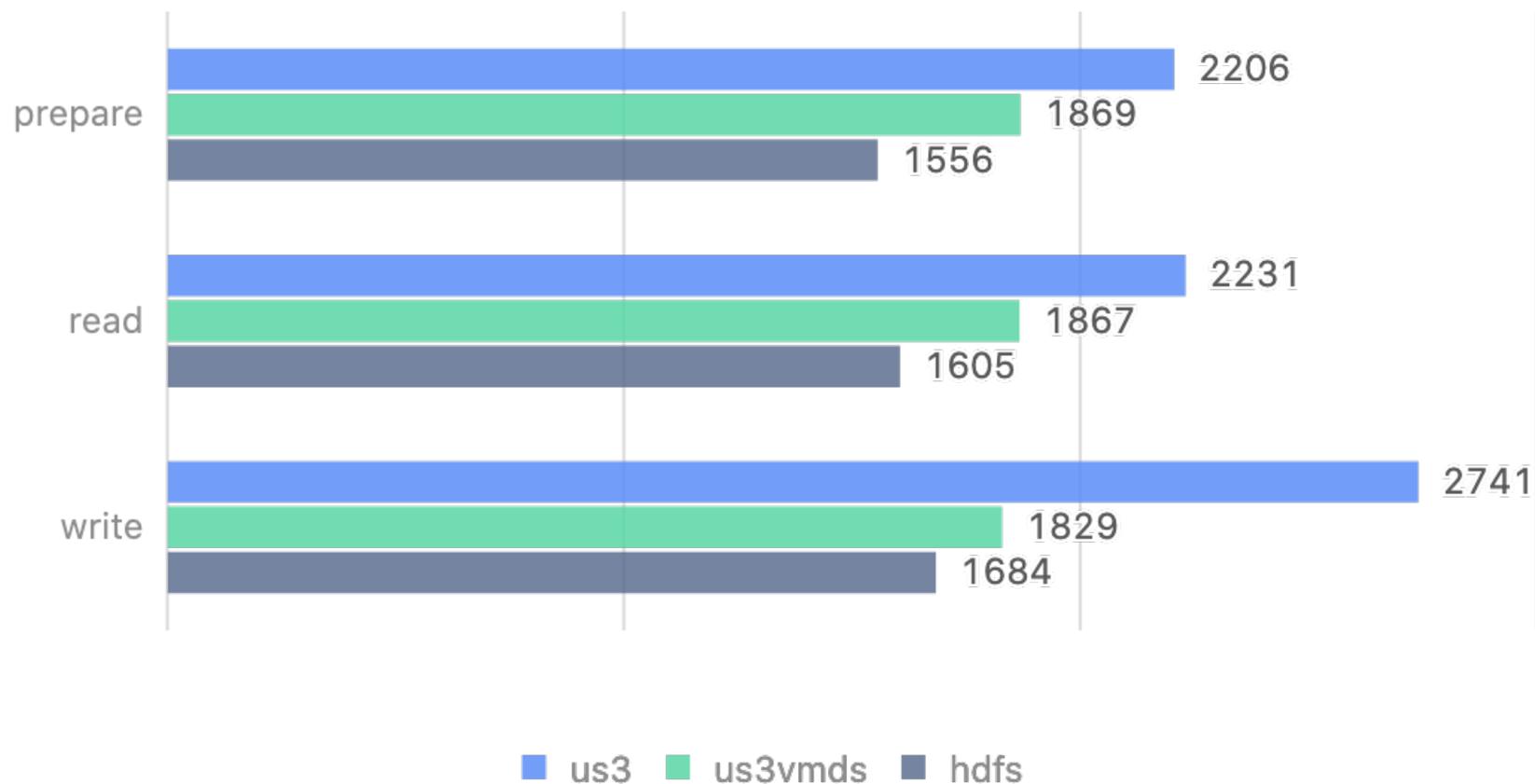
- hibench.dfsioe.mybigdata.read.number_of_files 44704
- hibench.dfsioe.mybigdata.read.file_size 64 // 64MiB
- hibench.dfsioe.mybigdata.write.number_of_files 44704
- hibench.dfsioe.mybigdata.write.file_size 64 // 64MiB

测试的读写数据量约为3TB

测试数据:

HiBench

micro.dfsioe工作负载测试(单位:s)



SQL

scan

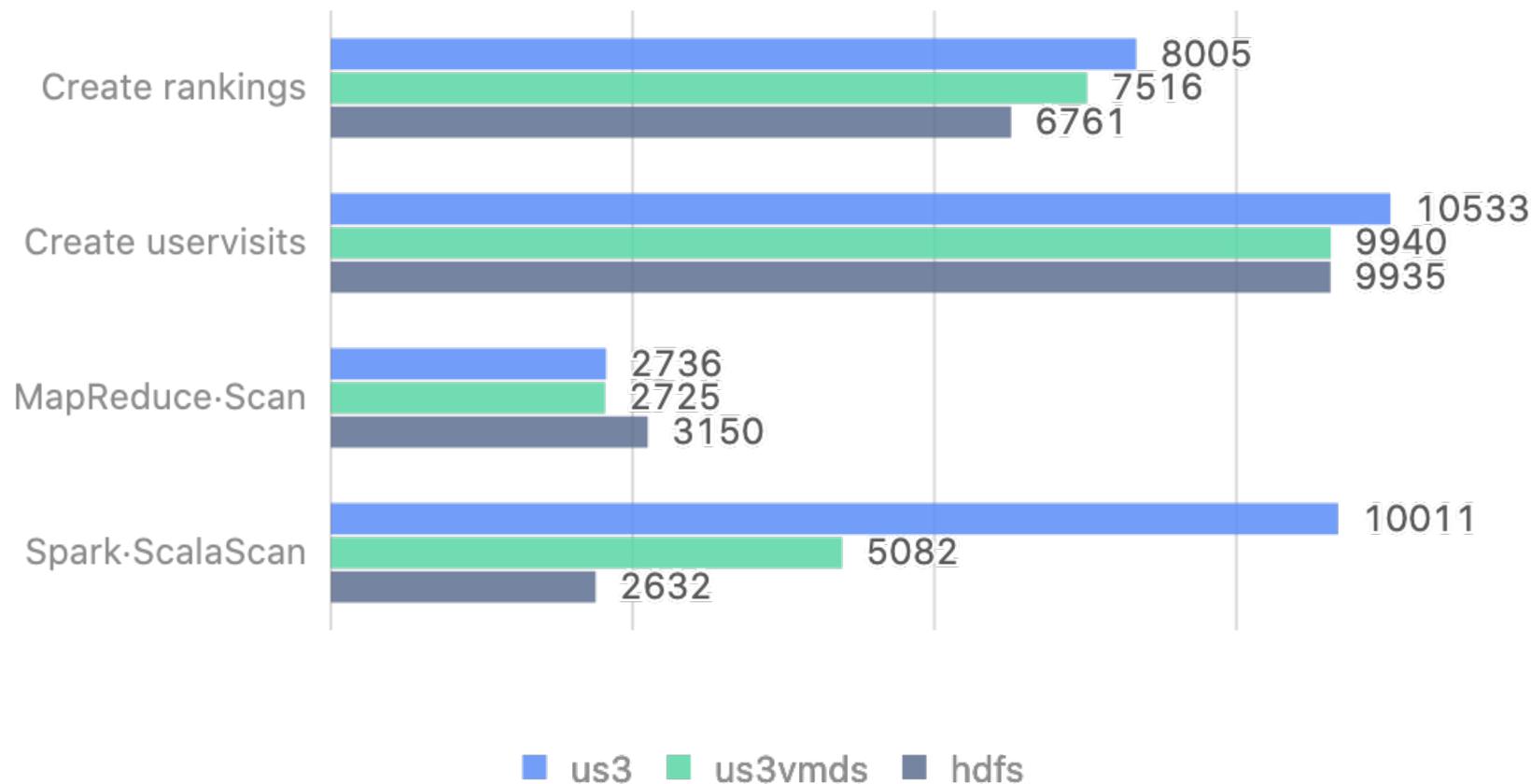
测试参数调整为:

- hibench.scan.mybigdata.uservisits 2000000000
- hibench.scan.mybigdata.pages 100000000

测试数据:

HiBench

sql.scan工作负载测试(单位:s)



join

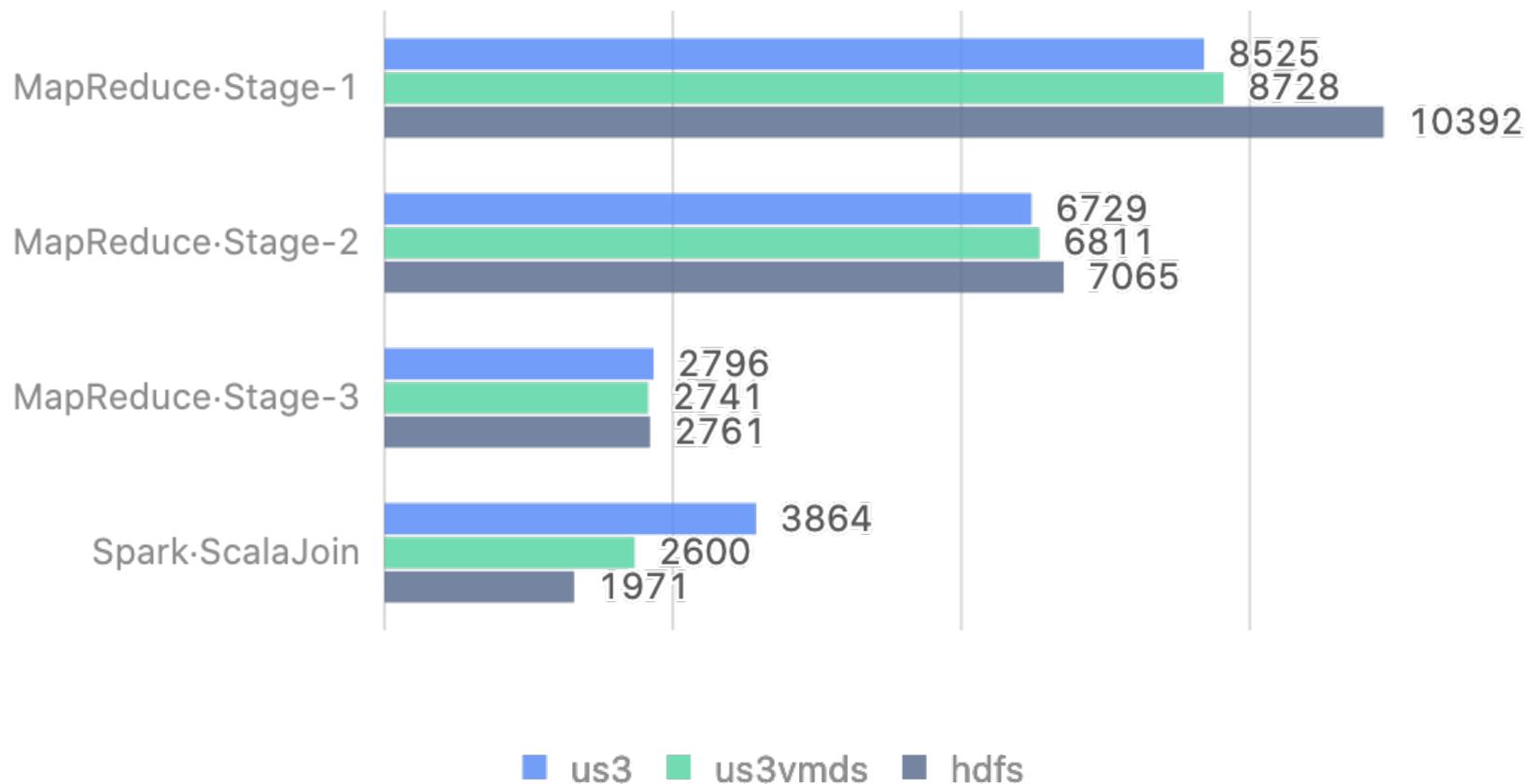
测试参数调整为:

- hibench.scan.mybigdata.uservisits 2000000000
- hibench.scan.mybigdata.pages 100000000

测试数据:

HiBench

sql.join工作负载测试(单位:s)



注意: Create rankings和Create uservisits任务同sql.scan,所以忽略

aggregation

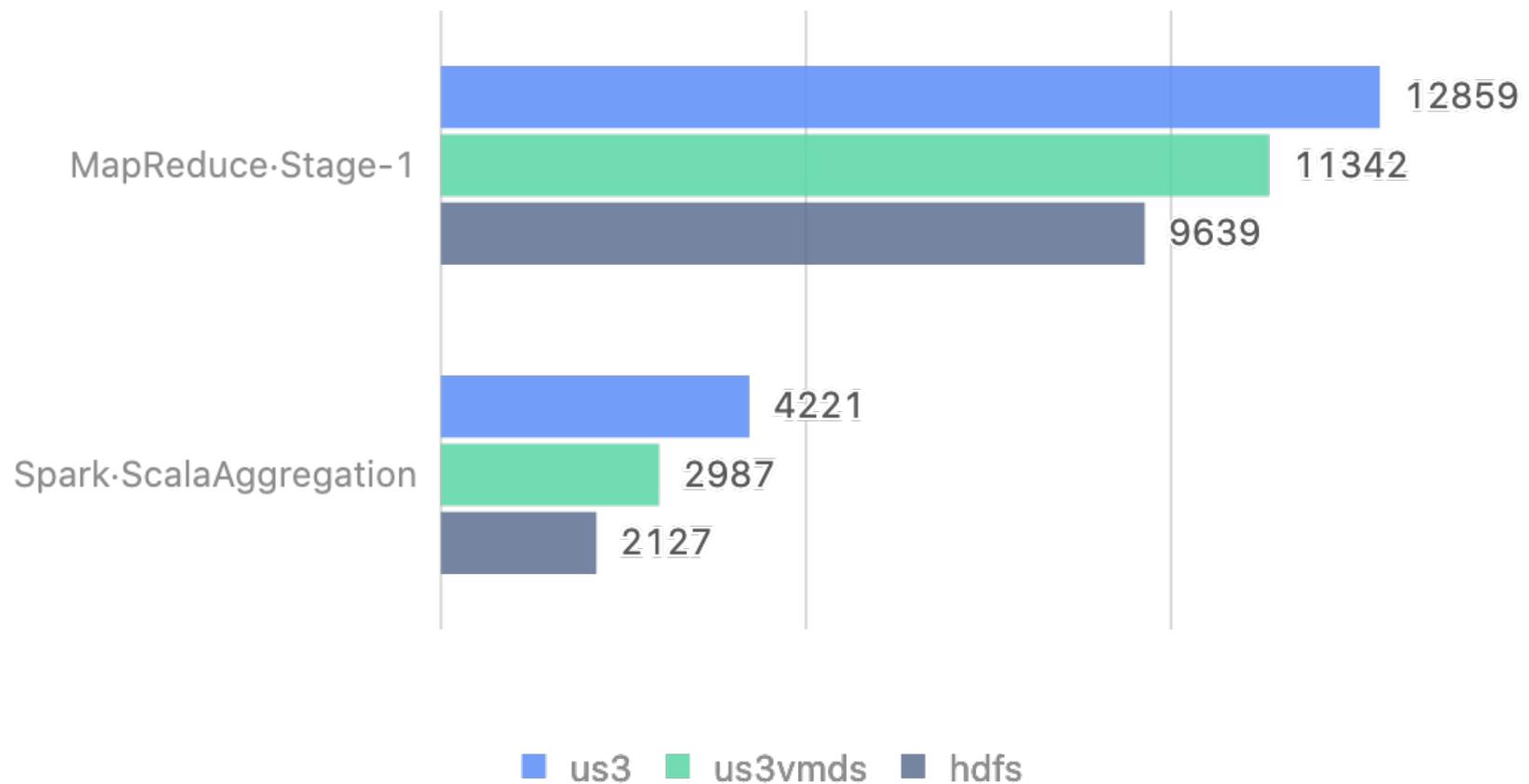
测试参数调整为:

- hibench.scan.mybigdata.uservisits 2000000000
- hibench.scan.mybigdata.pages 100000000

测试数据:

HiBench

sql.aggregation工作负载测试(单位:s)



注意: Create rankings和Create uservisits任务同sql.scan,所以忽略

总体来看,基于us3目前的方案相对原生分布式存储hdfs有一定差距,但基于us3vmds的方案在大部分场景性能有了大幅提高,后续也会不断改进,推出媲美甚至超越hdfs的方案。

缺陷

虽然us3vmds目前确实能给基于US3存储的大数据方案带来进一步性能提高,但us3vmds目前的设计还是有不少缺陷,主要在体现在几个方面:

1. 单点架构使得可靠性降低。不过us3vmds是无状态的设计,如果由于某些原因导致us3vmds退出,不会导致索引的丢失,只要快速重启,还是能根据us3的索引重建目录树物化视图,一般在服务过程中us3vmds退出10s内,不会对大数据任务产生影响,因为us3大数据适配器工具对错误做了一定时长的重试。只要及时保证us3vmds处于启动状态,就可以使得因us3vmds异常退出导致的可靠性问题大幅减少。当然由于us3vmds所在节点故障需要迁移而导致us3vmds需要更新服务地址影响时延相对会长,这个时候可以简单的修改配置,暂时改回只使用us3大数据适配器的方案。
2. 某些操作不能保证原子性。比如移动目录、删除目录,在us3vmds中的操作会变成批量操作这个目录前缀下的相关文件,中间任何一个文件即使加上了重试逻辑也可能会失败。目前us3vmds对这类问题只通过尽可能的重试来解决。
3. 非大数据集群写入数据,us3vmds索引不能保证强一致性。由于通过非us3大数据适配器方式(比如SDK)写入us3的数据不能及时通知us3vmds,所以us3vmds只能依赖大数据集群主动触发同步操作去同步最新索引,只能做到弱一致性。通过us3vmds方案中us3大数据适配器方式写入us3的数据,由于在写入us3成功后会主动通知us3vmds,所以us3vmds能及时更新索引信息,做到强一致性。

因此目前us3vmds不满足ACID,只满足BASE。

常见问题

内存消耗情况如何?

目前us3vmds测试保存超过100W文件索引时,内存在1.7g左右,其中包含了不少目录序列化的物化视图,理论实际值相对更少。平均每个索引的大小为1.7k左右,这个值确实相对较高,这也是后续优化点之一。

us3vmds启动后,发现通过hadoop命令看到的目录下文件与us3控制台上看到不一致,该怎么办?

有两个方法,第一种是直接暴力重启us3vmds,第二种是通过us3vmds的sync命令来进行强制同步。

工具简介

概述

US3Hadoop备份工具(简称US3Distcp)是基于MapReduce构建的自动检测,并能自动同步HDFS中的数据在US3中不存在或不一致文件的工具,因此US3Distcp可以直接替换Hadoop自带的distcp来做数据备份至US3(目前建议通过Hadoop distcp同步,用US3Distcp校验)。虽然工作原理类似于Hadoop自带distcp,但Hadoop distcp的文件完整性校验是基于HDFS的checksum,Hadoop distcp无法直接利用checksum来对比HDFS的数据与US3中的数据是否一致。因此要校验从HDFS中迁移到US3中文件的完整性,需要使用US3Distcp。

原理说明

US3Distcp同步的源目的相对位置关系始终跟distcp保持一致。工具会把整个执行阶段分为两个阶段:

- 检测阶段:检测阶段会先构建要同步的文件列表,提前构建好需检测的源与目的的文件,然后根据检测策略去决定任务的拆分,目前提供的策略有以下几种:
 - 目的是否存在:该策略是默认进行且不可跳过,如果目的端不存在则把该记录写入reduce输出文件,并附上dst not found表明是由于目标不存在的原因需要进行同步;
 - 长度是否相等:同是否存在的策略一样,是默认进行且不可跳过,该策略在源目的的大小不等的情况下会记录下来,并附上size not match表明需要进行同步的原因;
 - 目的修改时间是否最新:该策略可选,如果开启会比较源和目的的修改时间,如果源修改时间晚于目的修改时间,则会记录该源目的,并附上modification time not match表明需要同步的原因;
 - checksum校验:此checksum非hdfs中checksum的概念,只是对文件指纹信息的一种统称,目前支持crc32c和md5两种算法,同时考虑到此种校验方式需要消耗带宽并两端计算,所以考虑成本的因素,提供了两种校验方式的选择:
 - 全量校验:这种方式会用选择的算法对两端文件进行整体计算,然后对比最后计算结果;
 - 抽样校验:这种方式会对头尾1MiB数据分别进行校验,然后对文件中间部分随机抽取4个1MiB进行计算校验(小于6MiB的文件自动采用全量校验)。这种校验方式主要考虑到大数据场景一些格式化数据文件的头尾一般包含整个文件很重要的自描述信息,确保了不会因为头尾数据损坏导致整个文件无法使用的问题,而文件中间一般会有同步点,即使出现几个损坏点也不会导致整个文件不可使用。

- 同步阶段:同步阶段会根据检测阶段给出的待同步列表启动mapreduce任务进行自动同步,在全部同步成功之前会有三次job级别的重试。目前任务的拆分方式也是自动化的,每个任务负责处理不超过4.5GiB的文件同步,但如果单个文件超过4.5GiB,则会用一个map任务处理这个文件。该阶段任务拆分逻辑跟检测阶段进行checksum全量校验时的一致。该阶段也是可选阶段。

运行环境

- Windows
- Linux
- MacOS

下载与安装

参考【US3Hadoop 大数据适配工具】中US3适配器下载安装

注意:对应的**US3**适配版本需要 **$\geq v1.1.0$**

快速上手

- 参数说明
- 使用同步
- 使用校验
- 增量迁移
- 缺陷

参数说明

```
hadoop jar us3-bigdata-adaptor-2.8.5-1.1.0.jar cn.ucloud.us3.fs.distcp.DistCp
```

```
usage: distcp OPTIONS [source_path...] <target_path>
```

OPTIONS

-workspace <arg> the workspace keeps a directory of state data for the entire verification synchronization phase

-modtime check the modification time.when the modification time of the source end is later than the destination end, it is considered that it needs to be copied

-checksum <arg> to calculate the checksum, 'randome' only compares the head and tail 4MB checksums, and randomly takes 4 1MB fragments in the middle of the file for verification.

'all' will calculate the checksum of the entire file and compare it. not on by default

```
-algorithm <arg> checksum algorithm, the default is crc32, you can also  
choose 'md5'  
-dump <arg> Display the corresponding input information, input,  
checkout, cpout  
-enforce If the status result of the check or copy exists,  
whether to enforce the check or copy  
-onlycheck Only get the list of files that fail the integrity  
check, no need to copy  
-skipcheck skip all checks. Including file size, modification  
time, checksum
```

- `-workspace` : 该参数为必填项, 会在该参数指定的目录中创建input, checkout, cpout三个目录, 三个目录分别保存各个阶段的一些状态信息, input负责保留校验阶段的输入信息, 该信息由执行程序输入的参数转换而来, checkout负责保留校验阶段输出的待同步文件结果, 并作为同步阶段的输入, cpout保存的是同步阶段失败的文件, 如果文件中的记录为空, 则证明整个同步是成功的。
- `-modtime`: 可选, 如果开启则会启用目的修改时间是否最新的校验策略;
- `-checksum`: 可选, 如果开启则会启用checksum校验的校验策略, 其值为all则会进行文件内容全量校验, 为random则会进行文件内容的抽样校验;
- `-algorithm`: 可选, 该参数用于选择校验算法, 目前只提供md5, crc32c, 如果考虑校验时延, 可以采用crc32c, 默认为crc32c
- `-dump`: 用于列取workspace中input, checkout, cpout中的信息。
- `-enforce`: 一般在进行过校验或同步阶段后, 不能再次执行, 但该参数可以强制再次进行校验和同步;
- `-onlycheck`: 该参数会只有校验阶段, 没有同步阶段;
- `-skipcheck`: 该参数会跳过校验阶段, 只进行同步阶段;

此外还需要通过GenericOptionsParser的-D选项传入以下适配器的关键参数:

- `fs.us3.imp`: 始终为cn.ucloud.us3.fs.US3FileSystem。
- `fs.us3.access.key`: API或Token公钥。

- fs.us3.secret.key: API或Token私钥。
- fs.us3.endpoint: US3内网域名后缀,如:ufile.cn-north-04.ucloud.cn。参考地域和域名,域名需去除www.前缀。
- fs.us3.async.wio.use: 单个流写入是否使用异步IO,可以提高写入速度,特别是对单个大文件有效,但会消耗一定CPU资源,可能会降低任务的并行度,默认为false。
- fs.us3.async.wio.parallel: 在fs.us3.async.wio.use为true情况下生效,表示单个文件写入4MB分片的最大并行,默认为2。
- fs.us3.metadata.use: 是否启用索引缓存服务器,来加速US3索引性能,该服务需要使用者管理,目前正在公测中。
- fs.us3.metadata.host: 在fs.us3.metadata.usetrue情况下生效,可以直接指定IP:Port的地址形式,这种方式不需要配置/etc/hosts文件。用户也可以指定自定义域名,需要在/etc/hosts中配置解析地址,或者配置DNS解析。该参数处于测试中。
- fs.us3.generate.md5: 默认为false,如果为true则在写入US3时,客户端会计算一个MD5,并在最后会以md5-hash为key,md5值为value写入到文件的元数据中。开启会增加写入时延。该功能需要US3适配器的版本 $\geq 1.0.2$
- mapreduce.job.user.classpath.first: 始终为true。
- mapreduce.task.classpath.user.precedence: 始终为true。

最后还需通过-libjars选项传入适配器jar包路径。

使用同步

例如同步hdfs的/var/log/hadoop-yarn/apps/root/logs/application_1613870792268_0067 目录下文件到us3的存储空间bigdata-us3的hdfs_backup前缀下

```
→ bin X hadoop jar ./us3-bigdata-adaptor-2.8.5-1.1.0.jar \  
cn.ucloud.us3.fs.distcp.DistCp \  
-Dfs.us3.impl=cn.ucloud.us3.fs.US3FileSystem \  
-Dfs.us3.access.key=TOKEN_7468973e-d192-4378-8253-xxxxxx \  
-Dfs.us3.secret.key=9a632eb8-3938-43a0-a7e1-xxxxxx \  
-Dfs.us3.endpoint=internal-cn-bj.ufileos.com \  
-Dfs.us3.async.wio.use=false \  
-Dfs.us3.async.wio.parallel=2 \  

```

```
-Dfs.us3.metadata.use=true \  
-Dmapreduce.job.user.classpath.first=true \  
-Dmapreduce.task.classpath.user.precedence=true \  
-libjars ./us3-bigdata-adaptor-2.8.5-1.1.0.jar \  
-workspace /workspace/us3distcptest /var/log/hadoop-yarn/apps/root/logs/application_1613870792268_0067 us3://bigdata-us3/hdfs_backup  
21/03/03 16:28:54 INFO distcp.DistCp: Prepare to enter data, on workspace:/workspace/us3distcptest  
21/03/03 16:28:54 INFO distcp.DistCp: Prepare to enter data by sources:  
21/03/03 16:28:54 INFO distcp.DistCp: - /var/log/hadoop-yarn/apps/root/logs/application_1613870792268_0067  
21/03/03 16:28:54 INFO distcp.DistCp: Prepare to enter data by target: us3://bigdata-us3/hdfs_backup  
21/03/03 16:28:54 INFO distcp.CopyListing: check input file:/workspace/us3distcptest/input/input  
21/03/03 16:28:55 INFO distcp.DistCp: us3 distcp need check size/count is 10  
21/03/03 16:28:55 INFO distcp.DistCp: do check job, num of maps:1  
21/03/03 16:28:55 INFO distcp.DistCpInputFormat: the size of files count by DistCp, size is 10  
21/03/03 16:28:55 INFO distcp.DistCpInputFormat: Average size of map: 10, Number of maps:1, total size:10  
21/03/03 16:28:56 INFO mapreduce.JobSubmitter: number of splits:1  
21/03/03 16:28:56 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1614496134003_0032  
21/03/03 16:28:56 INFO impl.YarnClientImpl: Submitted application application_1614496134003_0032  
21/03/03 16:28:56 INFO mapreduce.Job: The url to track the job: http://master2:23188/proxy/application_1614496134003_0032/  
21/03/03 16:28:56 INFO mapreduce.Job: Running job: job_1614496134003_0032  
21/03/03 16:29:02 INFO mapreduce.Job: Job job_1614496134003_0032 running in uber mode : false  
21/03/03 16:29:02 INFO mapreduce.Job: map 0% reduce 0%  
21/03/03 16:29:07 INFO mapreduce.Job: map 100% reduce 0%  
21/03/03 16:29:14 INFO mapreduce.Job: map 100% reduce 100%  
21/03/03 16:29:14 INFO mapreduce.Job: Job job_1614496134003_0032 completed successfully  
21/03/03 16:29:14 INFO mapreduce.Job: Counters: 56
```

File System Counters

FILE: Number of bytes read=0

FILE: Number of bytes written=350391

FILE: Number of read operations=0

FILE: Number of large read operations=0

FILE: Number of write operations=0

HDFS: Number of bytes read=1853

HDFS: Number of bytes written=1867

HDFS: Number of read operations=17

HDFS: Number of large read operations=0

HDFS: Number of write operations=2

US3: Number of bytes read=0

US3: Number of bytes written=0

US3: Number of read operations=0

US3: Number of large read operations=0

US3: Number of write operations=0

Job Counters

Launched map tasks=1

Launched reduce tasks=1

Other local map tasks=1

Total time spent by all maps in occupied slots (ms)=193914

Total time spent by all reduces in occupied slots (ms)=329522

Total time spent by all map tasks (ms)=3078

Total time spent by all reduce tasks (ms)=2701

Total vcore-milliseconds taken by all map tasks=3078

```
Total vcore-milliseconds taken by all reduce tasks=2701
Total megabyte-milliseconds taken by all map tasks=6192936
Total megabyte-milliseconds taken by all reduce tasks=10544704
Map-Reduce Framework
Map input records=10
Map output records=10
Map output bytes=1502
Map output materialized bytes=1528
Input split bytes=116
Combine input records=0
Combine output records=0
Reduce input groups=10
Reduce shuffle bytes=1528
Reduce input records=10
Reduce output records=10
Spilled Records=10
Shuffled Maps =1
Failed Shuffles=0
Merged Map outputs=1
GC time elapsed (ms)=190
CPU time spent (ms)=1700
Physical memory (bytes) snapshot=1480015872
Virtual memory (bytes) snapshot=8427098112
Total committed heap usage (bytes)=2974810112
Shuffle Errors
```

```
BAD_ID=0
CONNECTION=0
IO_ERROR=0
WRONG_LENGTH=0
WRONG_MAP=0
WRONG_REDUCE=0
cn.ucloud.us3.fs.distcp.DistCpMapper$US3_DISTCP_CHECK
NUMBER_OF_DST_NOT_FOUND=10
TOTAL_NUMBER_OF_INPUT_FILES=10
File Input Format Counters
Bytes Read=1737
File Output Format Counters
Bytes Written=1867
21/03/03 16:29:14 INFO distcp.DistCp: us3 distcp check taken: 0 Days 0 Hours 0 Minutes 11 Seconds 893 Milliseconds
21/03/03 16:29:14 WARN distcp.CopyListing: hdfs://Ucluster/workspace/us3distcptest/checkout/_SUCCESS read
err:hdfs://Ucluster/workspace/us3distcptest/checkout/_SUCCESS not a SequenceFile
21/03/03 16:29:14 ERROR distcp.DistCp: us3 distcp, some files failed the integrity check, need to copy ~ o(ᑦ~ᑦ)o
21/03/03 16:29:14 WARN distcp.CopyListing: hdfs://Ucluster/workspace/us3distcptest/checkout/_SUCCESS read
err:hdfs://Ucluster/workspace/us3distcptest/checkout/_SUCCESS not a SequenceFile
21/03/03 16:29:14 INFO distcp.DistCp: us3 distcp need copy size is 5063546
21/03/03 16:29:14 INFO distcp.DistCp: calculate map num, total size:5063546 ,size_per_task:4294967296
21/03/03 16:29:14 INFO distcp.DistCp: calculate map num:1
21/03/03 16:29:14 INFO distcp.DistCp: do copy job, num of maps:1
21/03/03 16:29:15 INFO distcp.DistCpInputFormat: the size of files count by DistCp, size is 5063546
21/03/03 16:29:15 INFO distcp.DistCpInputFormat: Average size of map: 5063546, Number of maps:1, total size:5063546
```

```
21/03/03 16:29:15 WARN distcp.DistCpInputFormat: hdfs://Ucluster/workspace/us3distcptest/checkout/_SUCCESS read
err:hdfs://Ucluster/workspace/us3distcptest/checkout/_SUCCESS not a SequenceFile
21/03/03 16:29:15 INFO mapreduce.JobSubmitter: number of splits:1
21/03/03 16:29:15 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1614496134003_0033
21/03/03 16:29:15 INFO impl.YarnClientImpl: Submitted application application_1614496134003_0033
21/03/03 16:29:15 INFO mapreduce.Job: The url to track the job: http://master2:23188/proxy/application_1614496134003_0033/
21/03/03 16:29:15 INFO mapreduce.Job: Running job: job_1614496134003_0033
21/03/03 16:29:23 INFO mapreduce.Job: Job job_1614496134003_0033 running in uber mode : false
21/03/03 16:29:23 INFO mapreduce.Job: map 0% reduce 0%
21/03/03 16:29:30 INFO mapreduce.Job: map 100% reduce 0%
21/03/03 16:29:34 INFO mapreduce.Job: map 100% reduce 100%
21/03/03 16:29:35 INFO mapreduce.Job: Job job_1614496134003_0033 completed successfully
21/03/03 16:29:35 INFO mapreduce.Job: Counters: 57
File System Counters
FILE: Number of bytes read=0
FILE: Number of bytes written=348845
FILE: Number of read operations=0
FILE: Number of large read operations=0
FILE: Number of write operations=0
HDFS: Number of bytes read=5065539
HDFS: Number of bytes written=85
HDFS: Number of read operations=27
HDFS: Number of large read operations=0
HDFS: Number of write operations=2
US3: Number of bytes read=0
```

```
US3: Number of bytes written=5063546
US3: Number of read operations=0
US3: Number of large read operations=0
US3: Number of write operations=0
Job Counters
Launched map tasks=1
Launched reduce tasks=1
Other local map tasks=1
Total time spent by all maps in occupied slots (ms)=298179
Total time spent by all reduces in occupied slots (ms)=321714
Total time spent by all map tasks (ms)=4733
Total time spent by all reduce tasks (ms)=2637
Total vcore-milliseconds taken by all map tasks=4733
Total vcore-milliseconds taken by all reduce tasks=2637
Total megabyte-milliseconds taken by all map tasks=9522796
Total megabyte-milliseconds taken by all reduce tasks=10294848
Map-Reduce Framework
Map input records=10
Map output records=0
Map output bytes=0
Map output materialized bytes=6
Input split bytes=126
Combine input records=0
Combine output records=0
Reduce input groups=0
```

```
Reduce shuffle bytes=6
Reduce input records=0
Reduce output records=0
Spilled Records=0
Shuffled Maps =1
Failed Shuffles=0
Merged Map outputs=1
GC time elapsed (ms)=148
CPU time spent (ms)=2660
Physical memory (bytes) snapshot=1519132672
Virtual memory (bytes) snapshot=8458067968
Total committed heap usage (bytes)=3032481792
Shuffle Errors
BAD_ID=0
CONNECTION=0
IO_ERROR=0
WRONG_LENGTH=0
WRONG_MAP=0
WRONG_REDUCE=0
cn.ucloud.us3.fs.distcp.DistCpMapper$US3_DISTCP_COPY
READ_SIZE_OF_SOURCE=5063546
TOTAL_NUMBER_OF_INPUT_FILES=10
WRITE_SIZE_OF_DISTINATION=5063546
File Input Format Counters
Bytes Read=1867
```

File Output Format Counters

Bytes Written=85

21/03/03 16:29:35 INFO distcp.DistCp: us3 distcp copy taken: 0 Days 0 Hours 0 Minutes 11 Seconds 872 Milliseconds

21/03/03 16:29:35 WARN distcp.CopyListing: hdfs://Ucluster/workspace/us3distcptest/cpout/_SUCCESS read

err:hdfs://Ucluster/workspace/us3distcptest/cpout/_SUCCESS not a SequenceFile

21/03/03 16:29:35 INFO distcp.DistCp: us3 distcp check & copy succ!!

可以看到任务application_1614496134003_0032即为检查阶段,由于没有指定校验策略,所以默认只采用了目的是否存在和长度是否相等的策略,可以看到统计指标cn.ucloud.us3.fs.distcp.DistCpMapper\$US3_DISTCP_CHECK的两项:

- TOTAL_NUMBER_OF_INPUT_FILES=10:表明待校验的有10个文件。
- NUMBER_OF_DST_NOT_FOUND=10:表明这10个文件在目的端都不存在

然后会自动对10个文件进行同步,任务application_1614496134003_0033则负责同步,可以看到最后同步已经成功。

使用校验

对上面的同步进行校验,校验不通过的自动同步。这里示例除了采用默认的目的是否存在,长度是否相等 策略,还增加了checksum校验 策略,其校验算法采用 md5。

```
→ bin X hadoop jar ./us3-bigdata-adaptor-2.8.5-1.1.0.jar \  
cn.ucloud.us3.fs.distcp.DistCp \  
-Dfs.us3.impl=cn.ucloud.us3.fs.US3FileSystem \  
-Dfs.us3.access.key=TOKEN_7468973e-d192-4378-8253-xxxxxx \  
-Dfs.us3.secret.key=9a632eb8-3938-43a0-a7e1-xxxxxx \  
-Dfs.us3.endpoint=internal-cn-bj.ufileos.com \  
-Dfs.us3.async.wio.use=false \  

```

```
-Dfs.us3.async.wio.parallel=2 \  
-Dfs.us3.metadata.use=true \  
-Dmapreduce.job.user.classpath.first=true \  
-Dmapreduce.task.classpath.user.precedence=true \  
-libjars ./us3-bigdata-adaptor-2.8.5-1.1.0.jar \  
-workspace /workspace/us3distcptest -checksum all -algorithm md5 -enforce /var/log/hadoop-yarn/apps/root/logs/application_1613870792268_0067  
us3://bigdata-us3/hdfs_backup  
21/03/03 16:34:53 INFO distcp.DistCpOptions: set check sum mode:all  
21/03/03 16:34:53 INFO distcp.DistCpOptions: set check sum alogrithm:md5  
21/03/03 16:34:54 INFO distcp.DistCp:  
=====
```

```
21/03/03 16:34:54 INFO distcp.DistCp: The last task has not been processed yet, on workspace:/workspace/us3distcptest  
21/03/03 16:34:54 INFO distcp.DistCp: Start processing the last task.....  
21/03/03 16:34:54 INFO distcp.DistCp:  
=====
```

```
21/03/03 16:34:54 INFO distcp.DistCp: us3 distcp need check size/count is 5063546  
21/03/03 16:34:54 INFO distcp.DistCp: us3 distcp, drop workspace:/workspace/us3distcptest/checkout  
21/03/03 16:34:54 INFO distcp.DistCp: do check job, num of maps:1  
21/03/03 16:34:54 INFO distcp.DistCpInputFormat: the size of files count by DistCp, size is 5063546  
21/03/03 16:34:54 INFO distcp.DistCpInputFormat: Average size of map: 5063546, Number of maps:1, total size:5063546  
21/03/03 16:34:54 INFO mapreduce.JobSubmitter: number of splits:1  
21/03/03 16:34:54 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1614496134003_0034  
21/03/03 16:34:54 INFO impl.YarnClientImpl: Submitted application application_1614496134003_0034  
21/03/03 16:34:54 INFO mapreduce.Job: The url to track the job: http://master2:23188/proxy/application_1614496134003_0034/  
21/03/03 16:34:54 INFO mapreduce.Job: Running job: job_1614496134003_0034
```

```
21/03/03 16:35:01 INFO mapreduce.Job: Job job_1614496134003_0034 running in uber mode : false
```

```
21/03/03 16:35:01 INFO mapreduce.Job: map 0% reduce 0%
```

```
21/03/03 16:35:07 INFO mapreduce.Job: map 100% reduce 0%
```

```
21/03/03 16:35:15 INFO mapreduce.Job: map 100% reduce 100%
```

```
21/03/03 16:35:15 INFO mapreduce.Job: Job job_1614496134003_0034 completed successfully
```

```
21/03/03 16:35:15 INFO mapreduce.Job: Counters: 57
```

```
File System Counters
```

```
FILE: Number of bytes read=0
```

```
FILE: Number of bytes written=348871
```

```
FILE: Number of read operations=0
```

```
FILE: Number of large read operations=0
```

```
FILE: Number of write operations=0
```

```
HDFS: Number of bytes read=5065399
```

```
HDFS: Number of bytes written=85
```

```
HDFS: Number of read operations=27
```

```
HDFS: Number of large read operations=0
```

```
HDFS: Number of write operations=2
```

```
US3: Number of bytes read=0
```

```
US3: Number of bytes written=0
```

```
US3: Number of read operations=0
```

```
US3: Number of large read operations=0
```

```
US3: Number of write operations=0
```

```
Job Counters
```

```
Launched map tasks=1
```

```
Launched reduce tasks=1
```

```
Other local map tasks=1
Total time spent by all maps in occupied slots (ms)=252819
Total time spent by all reduces in occupied slots (ms)=469334
Total time spent by all map tasks (ms)=4013
Total time spent by all reduce tasks (ms)=3847
Total vcore-milliseconds taken by all map tasks=4013
Total vcore-milliseconds taken by all reduce tasks=3847
Total megabyte-milliseconds taken by all map tasks=8074156
Total megabyte-milliseconds taken by all reduce tasks=15018688
Map-Reduce Framework
Map input records=10
Map output records=0
Map output bytes=0
Map output materialized bytes=6
Input split bytes=116
Combine input records=0
Combine output records=0
Reduce input groups=0
Reduce shuffle bytes=6
Reduce input records=0
Reduce output records=0
Spilled Records=0
Shuffled Maps =1
Failed Shuffles=0
Merged Map outputs=1
```

```
GC time elapsed (ms)=170
CPU time spent (ms)=2570
Physical memory (bytes) snapshot=1501753344
Virtual memory (bytes) snapshot=8458620928
Total committed heap usage (bytes)=3024093184
Shuffle Errors
BAD_ID=0
CONNECTION=0
IO_ERROR=0
WRONG_LENGTH=0
WRONG_MAP=0
WRONG_REDUCE=0
cn.ucloud.us3.fs.distcp.DistCpMapper$US3_DISTCP_CHECK
SIZE_OF_DISTINATION_CHECKSUM_READ=5063546
SIZE_OF_SOURCE_CHECKSUM_READ=5063546
TOTAL_NUMBER_OF_INPUT_FILES=10
File Input Format Counters
Bytes Read=1737
File Output Format Counters
Bytes Written=85
21/03/03 16:35:15 INFO distcp.DistCp: us3 distcp check taken: 0 Days 0 Hours 0 Minutes 13 Seconds 9 Milliseconds
21/03/03 16:35:15 WARN distcp.CopyListing: hdfs://Ucluster/workspace/us3distcptest/checkout/_SUCCESS read
err:hdfs://Ucluster/workspace/us3distcptest/checkout/_SUCCESS not a SequenceFile
21/03/03 16:35:15 INFO distcp.DistCp: us3 distcp, all file integrity verification passed, no need to copy ~ ^_^
```

最后提示校验通过。

增量迁移

需要先删除workspace指定目录,然后按照前面*使用同步*的方法进行

缺陷

目前的缺陷有以下几点:

- 同步文件不能就近访问。不能把待同步的hdfs上相同节点的文件归并到指定任务中,并让任务分配到对应的hdfs节点。
- 不支持正则,通配等过滤规则。
- 不能对单个任务限流。但目前同步写入速度稳定在30MiB/s左右。
- 全量checksum校验会导致大量流量穿越。后续可以通过支持US3的Etag校验,把流量限定在hdfs集群内部。
- 任务分配算法不够均匀。

常见问题

Q. 大规模迁移备份时, 怎么避免hdfs集群的其他业务不受影响?

A. 分配固定资源的任务队列, 并通过-Dmapreduce.job.queueName传递给us3distcp。任务队列资源容量是个经验值, 需要结合整个备份数据规模, 特征来综合考虑。

工具简介

概述

为了解决clickhouse写入UCloud对象存储US3的问题,US3基于官方clickhouse版本v20.8.7.15-lts开发,增加了clickhouse中disk类型为us3对象存储的支持,支持前缀挂载。

ClickHouse介绍

clickhouse是一个用于OLAP的开源列式数据库,Yandex开发。具体介绍可参见官网。

下载与配置使用

源码下载

- 下载源码

```
git clone https://github.com/us3-epoch/ClickHouse
```

- 切换到指定分支

```
git checkout us3_support_v20.8.7.15-lts
```

- 下载依赖的子模块

```
git submodule update --init --recursive
```

编译

clickhouse编译依赖gcc/llvm, cmake, ninja。如使用gcc, 请确保版本在10及以上。可按官方说明准备编译环境。

编译：

```
cd ClickHouse
mkdir build
cd build
cmake ..
ninja
```

*注:如使用gcc,请将cmake ..替换为cmake -DENABLE_EMBEDDED_COMPILER=0 -DUSE_INTERNAL_LLVM_LIBRARY=0 -DWERROR=0 ..

配置与使用

如需使用us3作为后端存储,需在配置文件中增加disk配置。配置文件的详细设置请参考[官方链接](#)

在配置文件的disks中增加如下配置:

```
<disks>
<your_name>
<type>us3</type>
<endpoint>ufile.cn-north-02.ucloud.cn</endpoint>
<bucket>your-bucket</bucket>
<access_key>*****</access_key>
<secret_key>*****</secret_key>
<prefix>/clickhouse/</prefix>
</your_name>
</disks>
```

policies中增加如下配置:

```
<policies>
<your_name>
<volumes>
<main>
<disk>your_disk_name</disk>
</main>
</volumes>
</your_name>
</policies>
```

创建表时增加如下语句

```
SETTINGS your_setting,
storage_policy = 'your-policy-name';
```

即可创建使用us3作为存储后端的表。

可通过如下命令查看策略是否创建成功。

```
clickhouse-client

select * from system.storage_policies
```

输出如下(省略部分内容):

policy_name	volume_name	volume_priority	disks	volume_type	max_data_part_size	move_factor
default	default	1	['default']	JBOD	0	0
testpolicy	main	1	['testdiskname']	JBOD	0	0.1

性能分析

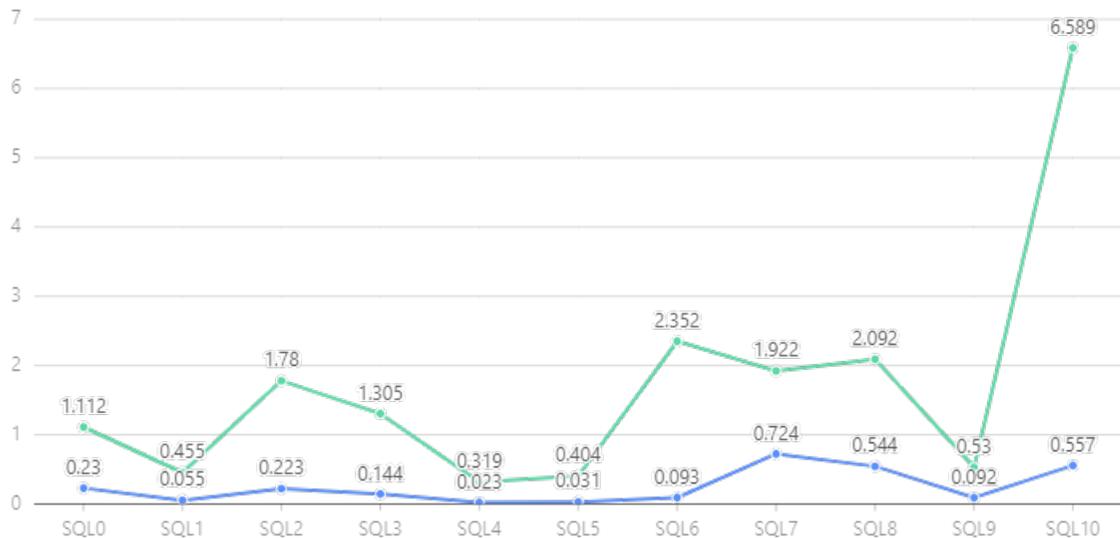
相比使用本地存储,性能上会有一些的损耗,时延大约是本地的8~9倍左右。

通过官方ontime测试数据集进行单表测试,测试结果如下:

本地vsUS3性能对比

SQL耗时 (单位: 秒)

— 本地 — US3



其中本地测试按照官网步骤执行;US3测试建表语句如下,SQL操作同本地

US3建表语句

```
CREATE TABLE `ontime_us3` (  
  `Year` UInt16,  
  `Quarter` UInt8,  
  `Month` UInt8,  
  `DayOfMonth` UInt8,  
  `DayOfWeek` UInt8,  
  `FlightDate` Date,  
  `UniqueCarrier` FixedString(7),  
  `AirlineID` Int32,  
  `Carrier` FixedString(2),  
  `TailNum` String,  
  `FlightNum` String,  
  `OriginAirportID` Int32,  
  `OriginAirportSeqID` Int32,  
  `OriginCityMarketID` Int32,  
  `Origin` FixedString(5),  
  `OriginCityName` String,  
  `OriginState` FixedString(2),  
  `OriginStateFips` String,  
  `OriginStateName` String,  
  `OriginWac` Int32,  
  `DestAirportID` Int32,  
  `DestAirportSeqID` Int32,  
  `DestCityMarketID` Int32,  
  `Dest` FixedString(5),
```

```
`DestCityName` String,  
`DestState` FixedString(2),  
`DestStateFips` String,  
`DestStateName` String,  
`DestWac` Int32,  
`CRSDepTime` Int32,  
`DepTime` Int32,  
`DepDelay` Int32,  
`DepDelayMinutes` Int32,  
`DepDel15` Int32,  
`DepartureDelayGroups` String,  
`DepTimeBlk` String,  
`TaxiOut` Int32,  
`WheelsOff` Int32,  
`WheelsOn` Int32,  
`TaxiIn` Int32,  
`CRSArrTime` Int32,  
`ArrTime` Int32,  
`ArrDelay` Int32,  
`ArrDelayMinutes` Int32,  
`ArrDel15` Int32,  
`ArrivalDelayGroups` Int32,  
`ArrTimeBlk` String,  
`Cancelled` UInt8,  
`CancellationCode` FixedString(1),
```

```
`Diverted` UInt8,  
`CRSElapsedTime` Int32,  
`ActualElapsedTime` Int32,  
`AirTime` Int32,  
`Flights` Int32,  
`Distance` Int32,  
`DistanceGroup` UInt8,  
`CarrierDelay` Int32,  
`WeatherDelay` Int32,  
`NASDelay` Int32,  
`SecurityDelay` Int32,  
`LateAircraftDelay` Int32,  
`FirstDepTime` String,  
`TotalAddGTime` String,  
`LongestAddGTime` String,  
`DivAirportLandings` String,  
`DivReachedDest` String,  
`DivActualElapsedTime` String,  
`DivArrDelay` String,  
`DivDistance` String,  
`Div1Airport` String,  
`Div1AirportID` Int32,  
`Div1AirportSeqID` Int32,  
`Div1WheelsOn` String,  
`Div1TotalGTime` String,
```

```
`Div1LongestGTime` String,  
`Div1WheelsOff` String,  
`Div1TailNum` String,  
`Div2Airport` String,  
`Div2AirportID` Int32,  
`Div2AirportSeqID` Int32,  
`Div2WheelsOn` String,  
`Div2TotalGTime` String,  
`Div2LongestGTime` String,  
`Div2WheelsOff` String,  
`Div2TailNum` String,  
`Div3Airport` String,  
`Div3AirportID` Int32,  
`Div3AirportSeqID` Int32,  
`Div3WheelsOn` String,  
`Div3TotalGTime` String,  
`Div3LongestGTime` String,  
`Div3WheelsOff` String,  
`Div3TailNum` String,  
`Div4Airport` String,  
`Div4AirportID` Int32,  
`Div4AirportSeqID` Int32,  
`Div4WheelsOn` String,  
`Div4TotalGTime` String,  
`Div4LongestGTime` String,
```

```
`Div4WheelsOff` String,  
`Div4TailNum` String,  
`Div5Airport` String,  
`Div5AirportID` Int32,  
`Div5AirportSeqID` Int32,  
`Div5WheelsOn` String,  
`Div5TotalGTime` String,  
`Div5LongestGTime` String,  
`Div5WheelsOff` String,  
`Div5TailNum` String  
) ENGINE = MergeTree  
PARTITION BY Year  
ORDER BY (Carrier, FlightDate)  
SETTINGS index_granularity = 8192,  
storage_policy = 'us3';
```

US3Browser 下线说明

US3Browser于2023年4月17日下线。**US3Browser**已不再维护更新,为避免因使用US3Browser造成数据损坏等影响,请尽快切换为US3管理控制台或其他US3常用工具。

如有可视化界面工具的需求,推荐您使用S3Browser请参见AWS客户端工具

概述

US3Browser是对象存储推出的一款可视化界面工具,支持Windows和Mac平台,让使用US3的用户可以采用更加简单轻松的交互方式对US3的Bucket进行管理,包括但不限于对目录/文件的创建、删除、上传、下载等操作。

版本和运行环境

软件版本

当前版本:v1.0.1

运行环境

支持Windows和Mac平台

主要功能

- 目录的创建、删除、批量删除、过滤等功能
- 文件的上传、下载、删除、批量删除
- 存储类型修改, 文件解冻等功能

迁移工具（历史版本）

介绍

ufile-import 是对象存储 US3 提供的一款将数据迁移至 US3 存储空间(Bucket) 的工具。您可以将 ufile-import 部署在本地服务或者云主机上, 轻松将您其他云存储的数据迁移到 US3。

全新 US3SYNC 迁移工具已正式发布, 请参照:[US3SYNC 迁移工具](#)

适用情况

- 阿里云对象存储数据迁移到 US3 对象存储
- 七牛云对象存储数据迁移到 US3 对象存储
- US3 对象存储不同 Bucket 之前的数据迁移
- 支持 S3 协议的对象存储迁移到 US3 对象存储

准备工作

1. 根据需要迁移的文件的总大小, 选择硬盘合适的云主机。必须要保证硬盘存储量大于文件迁移数据量, 否则可能会因为硬盘大小不够, 而造成迁移数据不完整。

举例: 假设您有总量为 100G 的文件需要迁移, 设置的处理文件并发数是 40 (即ufile-import.json中的"concurrent"参数, 可以往下阅读, 了解该参数的使用。), 在全部的迁移文件中, 单个文件最大大小为 1G 左右, 则您至少需要大小为: $40 / * 1 / * 2 = 80G$ 左右大小的硬盘, 来缓存下载过程中的临时文件, 来保证迁移过程中, 有足够的硬盘容量, 来缓存临时文件, 否则可能会造成迁移文件的不完整。

安装步骤

1. 下载安装包 Linux 64 位操作系统请下载 下载地址: <https://github.com/ufilesdk-dev/ufile-import/archive/master.zip>
2. 安装程序。

进入下载安装包目录,解压文件.

```
unzip master.zip
cd ufile-import-master
tar zxvf ufile-import.tgz
```

3. 启动 redis 服务。

服务依赖于 redis 服务,安装包中已经还有 redis 服务的相关配置,直接启动即可。

```
1 cd ufile-import
2 cd redis
3 ./start.sh
```

可以通过执行 `./ps.sh` 命令来查看 redis 服务状态,redis 服务正常启动,状态如下:

```
root 5318 1 0 14:50 pts/0 00:00:15 ./redis-server 127.0.0.1:6379
```

工具使用详情参照 US3 迁移工具使用说明

创建存储空间

在上传文件(Object)到 US3 之前,您需要使用 US3 API 中的 CreateBucket 接口来创建一个用于存储文件的存储空间(Bucket),存储空间具有各种配置属性,包括地域、访问权限以及其他元数据。或者使用 US3 控制台来创建一个存储空间(Bucket),并设置存储空间的访问权限。

操作方式

操作方式	说明
控制台	Web 应用程序,直观易用
管理工具	管理工具
API	API
SDK	SDK

使用限制

- 同一用户创建的存储空间总数不能超过 20 个。
- 每个存储空间的名字全局唯一,否则会创建失败。
- 存储空间的名称需要符合命名规范。
- 存储空间一旦创建成功,名称和所处地域不能修改。

设置空间读写权限

您可以在创建存储空间(Bucket)时设置存储空间的访问权限(ACL),也可以在创建 Bucket 后根据自己的业务需求修改存储空间的 ACL,该操作只有存储空间的拥有者可以执行。

存储空间有两种访问权限:

私有空间: 只有该存储空间的拥有者可以对该存储空间内的文件进行写操作,所有文件须获得拥有者的API密钥授权才能访问。

公有空间: 只有该存储空间的拥有者可以对该存储空间内的文件进行写操作,任何人(包括匿名访问者)都可以对该存储空间中的文件进行读操作。

获取存储空间信息

您可以通过 US3 API 的 DescribeBucket 接口获取存储空间(Bucket)的详细信息。获取存储空间信息的 API 详情请参考 DescribeBucket。

操作方式

操作方式	说明
控制台	进入控制台后在列表中查看存储空间信息
管理工具	管理工具
API	API
SDK	SDK

删除存储空间

您可以通过 US3 API 的 DeleteBucket 接口删除您创建的存储空间。删除存储空间的 API 详细信息可参考DeleteBucket

如果存储空间不为空（存储空间中有文件或者是尚未完成的分片上传），则存储空间无法删除。

必须删除存储空间中的所有文件和未完成的分片文件后，存储空间才能成功删除。如果想删除存储空间内部所有的文件，建议设置生命周期对存储桶内文件进行删除。

操作方式

操作方式	说明
控制台	Web 应用程序, 直观易用
管理工具	管理工具
API	API
SDK	SDK

上传文件

简单上传

简单上传指的是使用 US3 API 中的 PutObject 方法上传单个文件(Object)。简单上传适用于一次 HTTP 请求交互即可完成上传的场景,比如小文件(小于 512MB)的上传。

使用说明

- 简单上传的 API 接口的详细信息请参见 PutFile。
- 大文件(大于 512MB)的上传请使用分片上传。
- 命名限制:
 - 使用 UTF-8 编码。
 - 长度必须在 1-1023 字节之间。
 - 不能以正斜线 (/) 或者反斜线 (\) 字符开头。
- 为了防止第三方未经授权往您的 Bucket 里上传数据,US3 提供了 Bucket 和 Object 级别的访问权限控制。
- 在文件上传到 US3 上后,您可以通过上传回调来向指定的应用服务器发起回调请求,进行下一步操作。
- 如果上传的是图片,您还可以进行图片处理。

表单上传

表单上传是指使用 US3 API 中的 PostFile 请求来完成 Object 的上传,上传的 Object 不能超过 1GB。

备注: 表单上传的 **API** 接口详细信息请参见 **PostFile**。

适用场景

表单上传非常适合嵌入在 HTML 网页中来上传 Object,比较常见的场景是网站应用,以招聘网站为例:

	不使用表单上传	表单上传
流程对比	网站用户上传简历。	网站用户上传简历。
⋮	网站服务器回应上传页面。	网站服务器回应上传页面。
⋮	简历被上传到网站服务器。	简历上传到US3。
⋮	网站服务器再将简历上传到US3。	

从流程上来说,使用表单上传,少了一步转发流程,更加方便。从架构上来说,原来的上传都统一走网站服务器,上传量过大时,需要扩容网站服务器。采用表单上传后,直接从客户端上传数据到 US3,上传量过大时,压力都在US3上,由 US3 来保障服务质量。

分片上传

US3 提供的分片上传 (Multipart Upload) 功能,可以将要上传的文件分成多个数据块 (US3 里又称之为 Part) 来分别上传,上传完成之后再调用 US3 的接口将这些 Part 组合成一个 Object 来达到断点续传的效果。

适用场景

当使用简单上传(PutFile)功能来上传较大的文件到 US3 的时候,如果上传的过程中出现了网络错误,那么此次上传失败,重试必须从文件起始位置上传。针对这种情况,您可以使用分片上传来达到断点续传的效果。

相对于其他的上传方式,分片上传适用于以下场景:

1. 恶劣的网络环境:如手机端,当出现上传失败的时候,可以对失败的 Part 进行独立的重试,而不需要重新上传其他的 Part。
2. 加速上传:要上传到 US3 的本地文件很大的时候,可以并行上传多个 Part 以加快上传。
3. 流式上传:可以在需要上传的文件大小还不确定的情况下开始上传。这种场景在视频监控等行业应用中比较常见。

分片上传流程

将要上传的文件按照一定的大小分片。

1. 初始化一个分片上传任务 InitiateMultipartUpload。
2. 逐个或并行上传分片 UploadPart。
3. 完成上传 FinishMultipartUpload。

注意事项

- 除了最后一块 Part,其他 Part 的大小不能小于 4MB,否则会导致调用 FinishMultipartUpload 接口时失败。
- 要上传的文件切分成 Part 之后,文件顺序是通过上传过程中指定的 partNumber 来确定的,实际执行中并没有顺序要求,因此可以实现并发上传。
- 具体的并发个数并不是越多速度越快,要结合用户自身的网络情况和设备负载综合考虑。
- 默认情况下,已经上传但还没有调用 FinishMultipartUpload 的 Part 是不会自动回收的,因此如果要终止上传并删除占用的空间请调用 AbortMultipartUpload。

上传后续操作

- 在文件上传到 US3 上后,您可以通过上传回调来向指定的应用服务器发起回调请求,进行下一步操作。
- 如果上传的是图片,您还可以进行 [图片处理](#) 操作。

上传回调

US3 在上传文件完成的时候可以提供回调(Callback)给应用服务器。您只需要在发送给 US3 的请求中携带相应的 Callback 参数,即能实现回调。

下载文件

简单下载

简单下载是通过 US3 API 的 GetObject 接口, 下载已经上传的文件 (Object), Object 下载是使用 HTTP 的 GET 请求来完成的。

- 简单下载的 API 接口详细信息请参见 GetFile。
- Object 的 URL 生成规则请参考 US3 的访问。
- 如果需要使用自定义域名来访问 Object, 请参考自定义域名访问 US3。

断点续传下载

US3 提供了从 Object 指定的位置开始下载的功能, 在下载大的 Object 的时候, 可以分多次下载。如果下载中断, 重启的时候也可以从上次完成的位置开始继续下载。

和简单上传类似, 您也需要对该 Object 有读权限。通过设置参数 Range 来支持断点续传, 对于比较大的 Object 建议使用该功能。Range 的定义可参考 HTTP RFC。如果在请求头中使用 Range 参数, 则返回消息中会包含整个文件的长度和此次返回的范围。例如 Content-Range: bytes 0-9/44, 表示整个文件长度为 44, 此次返回的范围为 0-9。如果不在范围内, 则传送整个文件, 并且不在结果中提及 Content-Range, 返回码为 206。

查看文件列表

您可以通过 US3 API 中的 PrefixFileList 接口列出您在存储空间(Bucket)中上传的文件(Object)。

查看文件列表的 API 详细信息请参考 PrefixFileList。

删除文件

删除文件即删除上传到存储空间(Bucket)中的文件(Object)。

US3 允许您执行如下删除动作：

- 单个删除:指定某个 Object 进行删除。
- 自动删除:如果需要删除的 Object 数目很多,而且删除的 Object 有一定的规律,比如定期删除某些天之前的 Object,或者是要清空整个Bucket,推荐使用生命周期管理来自动删除 Object。设置了生命周期规则之后,US3 会根据规则自动删除到期的 Object,从而极大减少您发送删除请求的次数,提高删除效率。

备注:该功能处于内测阶段,如需使用请联系技术支持。

API 签名算法

US3 REST API 基于 HMAC (哈希消息身份验证码) 密钥使用自定义 HTTP 方案进行身份验证。要对请求进行身份验证,您首先需要合并请求的选定元素以形成一个字符串。然后,您可以使用 UCloud 私有访问密钥来计算该字符串的 HMAC。通常我们将此过程称为“签署请求”并且我们将输出 HMAC 算法称为“签名”,因为它会模拟真实签名的安全属性。最后,您可以使用本部分中介绍的语法,作为请求的参数添加此签名。

系统收到经身份验证的请求时,将提取您申领的 UCloud 私有访问密钥,并以相同的使用方式将它用于计算已收到的消息的签名。然后,它会将计算出的签名与请求者提供的签名进行对比。如果两个签名相匹配,则系统认为请求者必须拥有对 UCloud 私有访问密钥的访问权限,因此充当向其颁发密钥的委托人的颁发机构。如果两个签名不匹配,那么请求将被丢弃,同时系统将返回错误消息。

基于以上的原理,我们对空间管理和文件管理两套接口提供两种原理一样但是细节不同的鉴权过程。

空间管理签名算法

账户的公私钥可以在 UCloud 控制台中 API 产品 - API 密钥,点击显示 API 密钥获取。

空间管理的签名使用查询字符串的签名方式,客户端通过把签名字段作为一个 query 字段传递给服务端请求授权验证。携带了签名 (Signature) 的创建空间请求具有如下的格式:

```
GET
/Type=public&BucketName=demobucket&PublicKey=uclouddemo@mail.com45207436768156091&Action=CreateBucket&Signature=13f7989d4a5a8ued83
c0e57ah43b3607bc506c7c HTTP/1.1
```

其中 Signature 的伪代码计算方式如下:

```
//生成请求的 query 字典
querys = {"PublicKey" : publicKey} + {其他 query 字段}
//对 query 字典按照字典排序方式(lexicographical order)排序
querys = querys.sort()
//生成 signstring
signstring = ""
for key, value in querys {
signstring += key + value
}
//将私钥加入签名
signstring += privateKey
//按照SHA1(RFC 3174)计算签名
signature = sha1(signstring)
//以16进制显示签名
signature = HEX(signature)
```

其他空间管理 API 的签名均可使用同样的计算方式生成。

文件管理签名算法

文件管理的签名有两种不同的方式通过 HTTP 请求传递给服务器,分别是身份验证标头和查询字符串请求身份验证替代项。

身份验证标头

该方式使用 Authorization 头部字段传递签名数据,并放置于各 HTTP 请求的报文头中,如下图所示,身份验证标头具有以下形式:

```
Authorization: UCloud UCloudPublicKey:Signature
```

其中,Signature 是一个哈希值,具体为请求中特定元素的 HMAC-SHA1 (RFC2104),因此 Signature 会因请求不同而异。如果客户端请求中随附的 Signature 与服务端计算出的 Signature 相匹配,则证明请求者拥有 UCloud 允许的访问权限。以下是 Authorization 身份验证标头构造的伪代码:

```
Authorization = "UCloud" + " " + UCloudPublicKey + ":" + Signature
Signature = Base64( HMAC-SHA1( UCloudPrivateKey, UTF-8-Encoding-Of( StringToSign ) ) )
StringToSign = HTTP-Verb + "\n" +
Content-MD5 + "\n" +
Content-Type + "\n" +
Date + "\n" +
CanonicalizedUCloudHeaders +
CanonicalizedResource
CanonicalizedUCloudHeaders = <described below>
CanonicalizedResource = "/" + Bucket + "/" + Key
```

StringToSign 中包括两类标头元素:

一类是位置标头,仅有 3 个,分别是 Content-MD5、Content-Type 和 Date,在 StringToSign 中不包括这些标头的名称,仅包括它们在请求中的值,如果请求中这些标头不存在,需要用空字符串("")代替;另一类是 UCloud 附加标头,以X-UCloud-开头,此类标头需要按照下面指定的方法构造 CanonicalizedUCloudHeaders 字符串后加入到 StringToSign 中。

备注:

1. 如果位置标头不在请求中(例如,Content-Type 或 Content-MD5 对于 PUT 请求是可选的,并且对于GET请求没有任何意义),必须使用空字符串""替换该位置;
2. BASE64 使用 standardbase64,不是 URLSafe 的 base64 算法,下同;

3. 当使用 POST 表单上传时,签名使用的 Content-Type 字段应该是 form 参数中的 Content-Type 字段(即文件本身的 mimetype),而非 HTTP 请求的 Content-Type。
4. UCloudPublicKey 和 UCloudPrivateKey 对应 令牌管理 中创建的令牌,用户也可以用 UCloud 私有访问密钥进行访问。
5. Key使用原始字符串即可,无需经过url编码

计算CanonicalizedUCloudHeaders步骤

1. 将每个以X-UCloud-开头的 HTTP 标头名称转换为小写。例如X-UCloud-Date”改为“x-ucloud-date。
2. 根据标头名称按字典顺序排列标头集。
3. 按照 RFC2616 中第 4.2 节中的规定,将相同名称的标头字段合并为一个header-name:comma-separated-value-list对,各值之间不留空格。

例如,可以将元数据标头x-ucloud-meta-username:fred和x-ucloud-meta-username:barney合并为单个标头x-ucloud-meta-username:fred,barney。

4. 通过将折叠空格(包括换行符)替换为单个空格,“展开”跨多个行的长标头(按照 RFC2616 中第 4.2 节允许的方式)。
5. 删除标头中冒号周围的空格。例如,标头x-ucloud-meta-username:fred,barney改为x-ucloud-meta-username:fred,barney。
6. 最后,请向生成的列表中的每个标准化标头附加换行字符(U+000A)。通过将此列表中所有的标头规范化为单个字符串,构建 CanonicalizedUcloudHeaders 元素。

示例

1. 确定使用 API 接口,例如使用 PUTFile 接口,签名前的请求如下

```
PUT /demokey HTTP/1.1 Host: demobucket.ufile.ucloud.cn Content-Length: 11434 Content-Type: image/jpeg X-UCloud-Foo: foo X-UCloud-Bar: bar1 X-UCloud-Bar: bar2
```

2. 拼接签名字符串对照 step1 的请求中的各个参数,程序中的各变量取值如下(采用伪代码描述)

```
bucket = "demobucket"  
key = "demokey"  
http_verb = "PUT"
```

```
content_md5 = ""
content_type = "image/jpeg"
date = ""
canonicalized_ucloud_headers = "x-ucloud-foo:foo" + "\n" + "x-ucloud-bar:bar1,bar2" + "\n"
canonicalized_resource = "/" + "demobucket" + "/" + "demokey"
string2sign = "PUT" + "\n"
+ "" + "\n"
+ "image/jpeg" + "\n"
+ "" + "\n"
+ "x-ucloud-foo:foo" + "\n" + "x-ucloud-bar:bar1,bar2" + "\n"
+ "/demobucket/demokey"
```

即

```
string2sign = "PUT\n\nimage/jpeg\n\nx-ucloud-foo:foo\nx-ucloud-bar:bar1,bar2\n/demobucket/demokey"
```

3. hmac-sha1 运算

使用 UCloud 分配给您的私钥对签名字符串做 hmac-sha1 运算

```
hmacstring = hmac-sha1(privateKey, string2sign)
```

4. base64 运算

对生成的 hmacstring 做 base64 运算

```
base64string = base64(hmacstring) = S5FVD2w613MKb/hisjaqHdjvn9U=
```

5. 生成最终签名格式

```
signature = UCloud demouser@ucloud.cn13424346821929713944:S5FVD2w613MKb/hisjaqHdjvn9U=
```

6. 生成带签名的 HTTP 请求

```
PUT /demokey HTTP/1.1 Host: demobucket.ufile.ucloud.cn Content-Length: 11434 Content-Type: image/jpeg X-UCloud-Foo: foo X-UCloud-Bar: bar1 X-UCloud-Bar: bar2 Authorization: UCloud demouser@ucloud.cn13424346821929713944:S5FVD2w613MKb/hisjaqHdjvn9U=
```

查询字符串请求身份验证替代项

可以通过传递请求信息作为查询字符串参数,而不是使用 AuthorizationHTTP 标头来验证特定类型的请求。这在允许第三方浏览器直接访问您的 US3 私有空间的文件,而无需代理请求时非常有用。其概念是构建一个“预签名”的请求并将其编码为最终用户浏览器可检索的 URL。此外,您还可以通过指定过期时间来限制预签名请求。

以下是查询字符串经身份验证的 US3 REST 请求示例。

```
GET /demokey.jpg?UCloudPublicKey=AKIAIOSFODNN7EXAMPLE&Expires=1141889120&Signature=vjbyPxybdZaNmGa%2ByT272YEAiv4%3D HTTP/1.1
Host: demobucket.s3.ucloud.cn
Date: Mon, 26 Mar 2007 19:37:58 +0000
```

查询字符串请求身份验证方法与普通的方法稍有差异,不同之处仅在于 Signature 请求参数和 StringToSign 元素的格式。下面的伪语法演示了查询字符串请求身份验证方法。

```
Signature = URL-Encode( Base64( HMAC-SHA1( UCloudPrivateKey, UTF-8-Encoding-Of( StringToSign ) ) ) );
StringToSign = HTTP-VERB + "\n" +
"\n" +
"\n" +
Expires + "\n" +
```

```
CanonicalizedUCloudHeaders +  
CanonicalizedResource;
```

请注意,在 StringToSign 中,HTTPDate 位置元素已替换为 Expires。CanonicalizedUCloudHeaders 和 CanonicalizedResource 是相同的。

文件 ETag 生成方法

文件 ETag 可用于查询文件是否可以秒传。

具体算法的伪代码如下:

```
if filesize<=4MB:  
hash = base64_url_safe(blkcnt, sha(file))  
else:  
hash = base64_url_safe(blkcnt, sha(sha(blk0), sha(blk1)...))
```

字段含义

blkcnt:文件以4MB为一个块进行切分后的块个数,占4个字节,以小端模式保存。

blkN:第N(N>=0)个数据块的数据。

签名工具

由于生成签名比较繁琐,用户可以借助签名工具来帮助生成签名,或者作为签名校验工具。

签名工具链接

目前签名工具支持:

1. 文件管理签名。
2. 私有空间文件下载地址构造。

US3 同时提供用户自行部署私有化的签名工具,适用于以下场景:

1. 用 java sdk 进行 Andorid 应用开发;进行用户图片、文件上传功能,需要使用公私钥或者 Token 进行签名。
2. 用 ios sdk 进行 IOS 应用开发;进行用户图片、文件上传功能,需要使用公私钥或者Token 进行签名。

在这类场景下,如果在开发时在 APP 里内置公私钥或者 Token,调用上传下载接口实现签名和上传,APP 可能会被破解,导致公私钥、Token 泄露,建议采用以下方式:

1. 用户上传前,把上传的参数,传递给 APP 的服务器。
2. 服务器收到用户请求,进行登录态校验,生成文件上传签名,返回给APP。
3. APP 使用服务器返回的签名,进行文件上传操作。

签名服务代码

在URL中包含签名

除了使用Authorization Header,您还可以在URL中加入签名信息,以便将该URL转给第三方实现授权访问。

注意事项

- 使用在URL中签名的方式,会将授权的数据在过期时间内暴露在互联网上,请预先评估使用风险。
- US3不支持同时在URL和Header中包含签名。
- PUT和GET请求都支持在URL中签名。
- 您可以为PUT操作生成一个预签名的URL,该URL检查用户是否上传了正确的内容.SDK对请求进行预签名时,将计算请求正文的校验和,并生成包含在预签名URL中的MD5校验和。用户必须上传与SDK生成的MD5校验和相同的内容,否则操作失败。如果要验证MD5,只需在请求中增加Content-MD5头即可。

签名实现

- 签名示例

```
https://****.cn-bj.ufileos.com/us3-api.pdf?UCloudPublicKey=TOKEN_2368f919-0e7e-4e6c-affb-083fa53110e8&Signature=ExPEr%2FBkI8AQZ2RP87UAgn%2BwkCk%3D&Expires=1656487334
```

- 参数说明

名称	类型	是否必选	描述
UCloudPublic Key	字符串	是	指定URL签名中使用的UCloudPublicKey。

Expires	数字	是	Unix时间戳(自UTC时间1970年01月01号开始的秒数),用于标识该URL的超时时间。如果US3接收到该URL请求的时间晚于签名中包含的Expires参数时,则返回请求超时的错误码。例如,当前时间是1141889060,开发者希望创建一个60秒后自动失效的URL,则可以设置Expires时间为1141889120。
Signature	字符串	是	<p>签名信息。格式如下:</p> <pre>StringToSign = HTTP-Verb + "\n" + Content-MD5 + "\n" + Content-Type + "\n" + Expires + "\n" + CanonicalizedUCloudHeaders + CanonicalizedResource</pre> <ul style="list-style-type: none"> * 所有US3支持的请求和各种Header参数,在URL中进行签名的算法和在Header中包含签名的算法类似。 * 生成URL中的签名字符串时,除了将Date参数替换为Expires参数外,仍包涵其他签名Header * 使用URL签名时,US3会先验证请求时间是否晚于Expires时间,然后再验证签名。

上传策略说明

US3 上传策略 (PutPolicy) 用于上传对象的同时, 完成一些指定的操作, 这些操作将会在上传动作完成后 (部分动作是在开始上传前执行) 触发并完成。

可以使用 PutPolicy 的 API 包括: PutFile、FinishMultipartUpload。

US3 上传回调

上传策略可实现回调其他服务 (callback), 对上传的文件进行处理。

该策略指定在上传文件完成后 (携带可能的参数) 去请求一个用户指定的服务地址 (目前仅支持 http, 且仅支持请求一个指定服务)。在得到用户服务器的回应后 (必须是 application/json 格式), 将用户服务器的返回值透传给用户。回调服务的地址使用 json 格式进行封装, 格式如下:

```
{
  "callbackUrl" : "http://test.ucloud.cn", //指定回调服务的地址
  "callbackBody" : "key1=value1&key2=value2" //传递给回调服务的参数
}
```

携带上传策略的API请求, 授权字段 Authorization 部分与不携带上传策略有所区别。

不携带上传策略, 上传时的 Authorization 格式为:

```
Authorization: UCloud publickey:signature
```

使用上传策略,则格式为:

```
Authorization: UCloud publickey:signature:encodedPutPolicy
```

其中,encodedPutPolicy = base64(json_encode(put_policy)) (注意: **json** 格式请使用压缩后的格式,不要携带空白字符,除非 **key/value** 本身是含有空白字符的字符串。这里的 **base64** 是 **URLSafe** 的 **base64**)

此外,旧有方式的签名字符串计算方式是:

```
signstring = HTTP-Verb + "\n" +  
Content-MD5 + "\n" +  
Content-Type + "\n" +  
Date + "\n" +  
CanonicalizedUCloudHeaders +  
CanonicalizedResource
```

当上传请求需要执行上传策略时,签名字符串其他部分不变,需要在末尾追加上传策略的base64字符串,即:

```
signstring\_with\_putpolicy = signstring + base64(json_encode(put_policy))
```

用法示例

若上传的文件为:flower.jpg,上传策略为:

```
"callbackUrl" : "<http://inner.umedia.ucloud.com.cn/CreateUmediaTask>",  
"callbackBody" : "url=<http://demo.ufile.ucloud.cn/flower.jpg&patten_name=mypolicy>"
```

没有上传策略的上传请求:

```
PUT /flower.jpg HTTP/1.1  
Content-Length: 123456  
Content-Type: image/jpeg  
Host: test.ufile.ucloud.cn  
Authorization: UCloud aGVsbHdvZGhhZGhhc2RoYWRzZGFkaHNkaGFkaGhkaGxrc2Rh:bTgzdWhkZGlsYS9kLmFkYWRhc2Ruaw==
```

携带上传策略的上传请求:

```
PUT /flower.jpg HTTP/1.1  
Content-Length: 123456  
Content-Type: image/jpeg  
Host: test.ufile.ucloud.cn  
Authorization: UCloud  
aGVsbHdvZGhhZGhhc2RoYWRzZGFkaHNkaGFkaGhkaGxrc2Rh:ZGFkLHBwMz0xZGthZGFkYXNkYQ==:XCJjYWxsYmFja1VybFwiOlwiIGh0dHA6Ly9pbm5lci51bWVkaWEudWNsb3VkLmNvbS5jbi9DcmVhdGVVbWVkaWFUYXNrXCIsXCJjYWxsYmFja0JvZHIcljpcInVybD1odHRwOi8vZGVtby51ZmlsZS51Y2xvdWQuY24vdGVzdC5tcDQmIHdHdHRIbI9uYW1IPW15cG9saWN5XCI=
```

备注: 签名与bucket相关, 示例中签名仅作参考。

AWS S3 协议支持说明

概述

S3协议是AWS推出,在对象存储行业成为事实标准,US3产品在自有标准的基础上,增加了针对S3 v4协议标准的兼容支持。

该内容为动态文件,无法展示

支持的 API

US3 目前的 S3 协议模块对标准 S3 协议的支持如下表:

编号	API名字	备注说明
1	HeadBucket	检测 Bucket 是否存在以及您是否有权限访问该 Bucket
2	ListBuckets	获取 Bucket 列表,只能获取公私钥或者 Token 拥有者创建的 Bucket
3	GetBucketLocation	返回所在地域名,不建议依赖该 API
4	GetBucketAcl	没有太多意义,主要为了支持 S3 Browser 而实现,响应体中的 Permission 字段永远为 FULL_CONTROL

5	GetBucketVersioning	没有太多意义, 主要为了支持 S3 Browser 而实现, 响应体中的 Status 字段永远为空字符串
6	PutBucketLifecycleConfiguration	为 Bucket 创建新的生命周期配置或替换现有的生命周期配置规则。注意, 这将覆盖现有的所有生命周期配置规则
7	GetBucketLifecycleConfiguration	获取 Bucket 中设置的生命周期配置规则
8	DeleteBucketLifecycle	删除 Bucket 中设置的所有生命周期配置规则。注意, 不支持删除 Bucket 中的指定某个或多个生命周期配置规则
9	GetObjectAcl	获取 Object 的访问权限信息
10	PutObjectAcl	设置 Object 的访问权限信息
11	HeadObject	参考US3 兼容 S3 API - v2.2.pdf
12	PutObject	参考US3 兼容 S3 API - v2.2.pdf
13	PostObject	参考US3 兼容 S3 API - v2.2.pdf
14	CopyObject	参考US3 兼容 S3 API - v2.2.pdf
15	GetObject	参考US3 兼容 S3 API - v2.2.pdf
16	ListObjects/ListObjectsV2	参考US3 兼容 S3 API - v2.2.pdf
17	DeleteObject/DeleteObjects	参考US3 兼容 S3 API - v2.2.pdf
18	CreateMultipartUpload	参考US3 兼容 S3 API - v2.2.pdf
19	UploadPart	参考US3 兼容 S3 API - v2.2.pdf
20	UploadPartCopy	参考US3 兼容 S3 API - v2.2.pdf
21	CompleteMultipartUpload	参考US3 兼容 S3 API - v2.2.pdf
22	AbortMultipartUpload	参考US3 兼容 S3 API - v2.2.pdf

23	ListMultipartUploads	获取正在执行的分片上传请求 ID
24	ListParts	获取正在执行分片上传的分片信息
25	RestoreObject	解冻处于归档状态的文件

注意:

- PutObject 目前仅支持 1GB 大小文件,如果需要上传大于 1GB 的文件,请采用分片上传的 API
- PostObject 目前仅支持最大 32MB 文件的上传
- CopyObject 目前仅支持最大 100MB 文件的拷贝
- UploadPart 目前仅支持 8MB 定长分片大小(最后一个分片允许小于 8MB)。若有不定长分片的需求,请联系技术支持
- US3 S3 对 AWS S3 兼容的存储类型及其转换规则参考 [存储类型转换规则](#)
- US3 的 ETag 计算方式与 AWS S3 存在部分差异,建议不依赖该 ETag
- 目前不支持 S3 API 的 MD5 校验,建议关闭:

例如AWS S3 Java SDK:

```
System.setProperty(SkipMd5CheckStrategy.DISABLEGETOBJECTMD5VALIDATION_PROPERTY,"");
```

```
System.setProperty(SkipMd5CheckStrategy.DISABLEPUTOBJECTMD5VALIDATION_PROPERTY,"");
```

- US3 的访问权限 (ACL) 定义与 AWS S3 存在差异,具体参考 [访问权限定义 \(ACL\)](#)
- 目前文件访问权限控制 API (GetObjectAcl、PutObjectAcl) 仅在部分地域支持
- 目前生命周期配置规则控制 API (PutBucketLifecycleConfiguration、GetBucketLifecycleConfiguration、DeleteBucketLifecycle) 仅在部分地域支持

- 目前 UploadPartCopy 处于内测阶段。若有使用需求, 请联系技术支持
- 目前不支持多版本功能 (Versioning)
- 目前不支持标签功能 (Tagging)
- ListObjects请求中的max-keys参数(请求返回对象的最大数量)最大值为5000

访问权限定义 (ACL)

US3 ACL	AWS S3 Canned ACL
private	private
public-read	public-read
public-read-write	public-read-write
不支持	aws-exec-read authenticated-read bucket-owner-read bucket-owner-full-control log-delivery-write

仅支持签名 V4

支持 V4 签名的场景:

1. URL 中携带参数 (URL Query 部分的 x-amz-credential 字段);

2. POST(表单中 x-amz-credential 域);
3. Header 中携带参数(Authorization字段);

S3 的 AccessKeyID 和 SecretAccessKey 说明

S3 的 AccessKeyID(或称AccessKey)和 SecretAccessKey(或称SecretKey)对应就是 UCloud 的 API 公钥和私钥,或者是 US3服务提供的 Token 公钥和 Token 私钥;

注意: 要求无论是 **API** 公私钥还是 **Token** 公私钥,要求操作的 **bucket**, 必须满足以下条件:

- 创建该 bucket 的账户与 API 公私钥的拥有者必须一致;
- 创建该 bucket 的账户与创建 Token 的账户必须一致;

S3的分片大小说明

1. 为了达到更好的传输性能,默认情况下仅支持8M大小的分片。
2. 部分地域已开通动态分片功能,如果固定8M分片无法满足需求,可联系技术支持开通动态分片。

API支持路径风格和虚拟主机风格

路径风格格式为: **http://\${Endpoint}/\${bucket名字}/\${key名字}**, **bucket** 名字作为路径使用的一部分。例如,AWS S3 Java SDK 在 UCloud 北京地域走外网使用 US3 S3 服务则设置如下:

```
"AWSCredential credentials = new BasicAWSCredentials(ACCESS_KEY,
SECRET_KEY);
ClientConfiguration clientConfig = new ClientConfiguration();
...
S3ClientOptions clientOptions = S3ClientOptions.builder().build();
```

```
clientOptions.setPathStyleAccess(true); // 表明使用路径风格API
AmazonS3 conn = new AmazonS3Client(credential, clientConfig);
conn.setS3ClientOptions(clientOptions);
conn.setEndpoint("s3-cn-bj.ufileos.com");"
```

虚拟主机风格: **http://\${bucket名字}.\${Endpoint}/\${key名字}**, 类似US3目前使用的URL形式。

访问域名 (Endpoint)

编号	地域	外网Endpoint	内网Endpoint
1	华北一	s3-cn-bj.ufileos.com	internal.s3-cn-bj.ufileos.com
2	华北二	s3-cn-wlcb.ufileos.com	internal.s3-cn-wlcb.ufileos.com
3	上海	s3-cn-sh2.ufileos.com	internal.s3-cn-sh2.ufileos.com
4	广州	s3-cn-gd.ufileos.com	internal.s3-cn-gd.ufileos.com
5	香港	s3-hk.ufileos.com	internal.s3-hk.ufileos.com
6	洛杉矶	s3-us-ca.ufileos.com	internal.s3-us-ca.ufileos.com
7	新加坡	s3-sg.ufileos.com	internal.s3-sg.ufileos.com
8	雅加达	s3-idn-jakarta.ufileos.com	internal.s3-idn-jakarta.ufileos.com
9	台北	s3-tw-tp.ufileos.com	internal.s3-tw-tp.ufileos.com
10	拉各斯	s3-afr-nigeria.ufileos.com	internal.s3-afr-nigeria.ufileos.com

11	圣保罗	s3-bra-saopaulo.ufileos.com	internal.s3-bra-saopaulo.ufileos.com
12	迪拜	s3-uae-dubai.ufileos.com	internal.s3-uae-dubai.ufileos.com
13	法兰克福	s3-ge-fra.ufileos.com	internal.s3-ge-fra.ufileos.com
14	胡志明市	s3-vn-sng.ufileos.com	internal.s3-vn-sng.ufileos.com
15	华盛顿	s3-us-ws.ufileos.com	internal.s3-us-ws.ufileos.com
16	孟买	s3-ind-mumbai.ufileos.com	internal.s3-ind-mumbai.ufileos.com
17	首尔	s3-kr-seoul.ufileos.com	internal.s3-kr-seoul.ufileos.com
18	东京	s3-jpn-ky.ufileos.com	internal.s3-jpn-ky.ufileos.com
19	曼谷	s3-th-bkk.ufileos.com	internal.s3-th-bkk.ufileos.com
20	伦敦	s3-uk-london.ufileos.com	internal.s3-uk-london.ufileos.com
21	莫斯科	s3-rus-mosc.ufileos.com	internal.s3-rus-mosc.ufileos.com

注意: 目前华北一, 香港, 胡志明, 首尔, 圣保罗, 洛杉矶, 华盛顿地域已经支持https协议, 其他地域可支持路径风格https, 后续支持虚拟主机风格https (所有地域内网不支持https)

回调扩展功能支持

请求形式 API 名字	PUT Object	POST Object	Complete Multipart Upload
在 URL 中携带参数	√	×	√
在 Header 中携带参数	√	×	√

在 POST 请求的 body 中使用表单域来携带参数	×	√	×
-------------------------------------------	---	---	---

√:支持

×:不支持

图片操作支持

参考 [图片处理服务](#)

存储类型转换规则

US3存储类型	S3存储类型	US3对应S3默认存储类型
STANDARD	STANDARD STANDARD_IA	STANDARD
IA	ONEZONE_IA INTELLIGENT_TIERING REDUCED_REDUNDANCY	ONEZONE_IA
ARCHIVE	GLACIER DEEP_ARCHIVE	GLACIER

AWS S3 协议应用场景

文件浏览器工具

功能说明

S3 Browser 是一种易于使用和强大的 Amazon S3 免费客户端。它提供了一个简单的 Web 服务接口,可用于存储和检索任意数量的数据,无论任何时候从任何地方。可以通过相关配置,直接操控 US3 对象存储的 Bucket 中的文件,进行上传,下载,删除等操作。

安装和使用

适用的操作系统: **Windows**

安装步骤

1. 下载安装包

下载地址: <http://s3browser.com>

2. 安装程序

进入下载页面,点击 Download S3 Browser,按照提示,进行安装即可。

使用方法

1. 增加用户

点击左上角 Accounts 按钮, 在下拉框中, 点击 Add new account

在 Add new account 页面中, 需要填写的项描述如下:

Account Name: 账户名称, 用户自定义。

Account Type: 账户类型, 选择 S3 Compatible Storage

REST Endpoint: 固定域名, 填写参考支持 AWS S3 协议说明。比如:s3-cn-bj.ufileos.com

Signature Version: 签名版本, 选择 Signature V4。

Access Key ID: Api 公钥, 或者 Token。具体获取请参考 S3 的 AccessKeyID 和 SecretAccessKey 说明。

Secret Access Key: API 私钥。具体获取请参考 S3 的 AccessKeyID 和 SecretAccessKey 说明。

Encrypt Access Keys with a password: 请勿勾选。

Use secure transfer(SSL/TSL): 目前仅中国-北京二, 中国-香港, 越南-胡志明, 韩国-首尔, 巴西-圣保罗, 美国-洛杉矶, 美国-华盛顿地域支持 HTTPS, 其他区域请勿勾选。

具体配置填写如下:

Add New Account

 **Add New Account** [online help](#)

Enter new account details and click Add new account

Account Name: **账户名称**

Assign any name to your account.

Account Type: **账户类型**

S3 Compatible Storage

Choose the storage you want to work with. Default is Amazon S3 Storage.

REST Endpoint: **固定域名**

Specify S3-compatible API endpoint. It can be found in storage documentation. Example: rest.server.com:8080

Access Key ID: **API或TOKEN公钥**

Required to sign the requests you send to Amazon S3, see more details at <https://s3browser.com/keys>

Secret Access Key: **API或TOKEN私钥**

Required to sign the requests you send to Amazon S3, see more details at <https://s3browser.com/keys>

Encrypt Access Keys with a password:

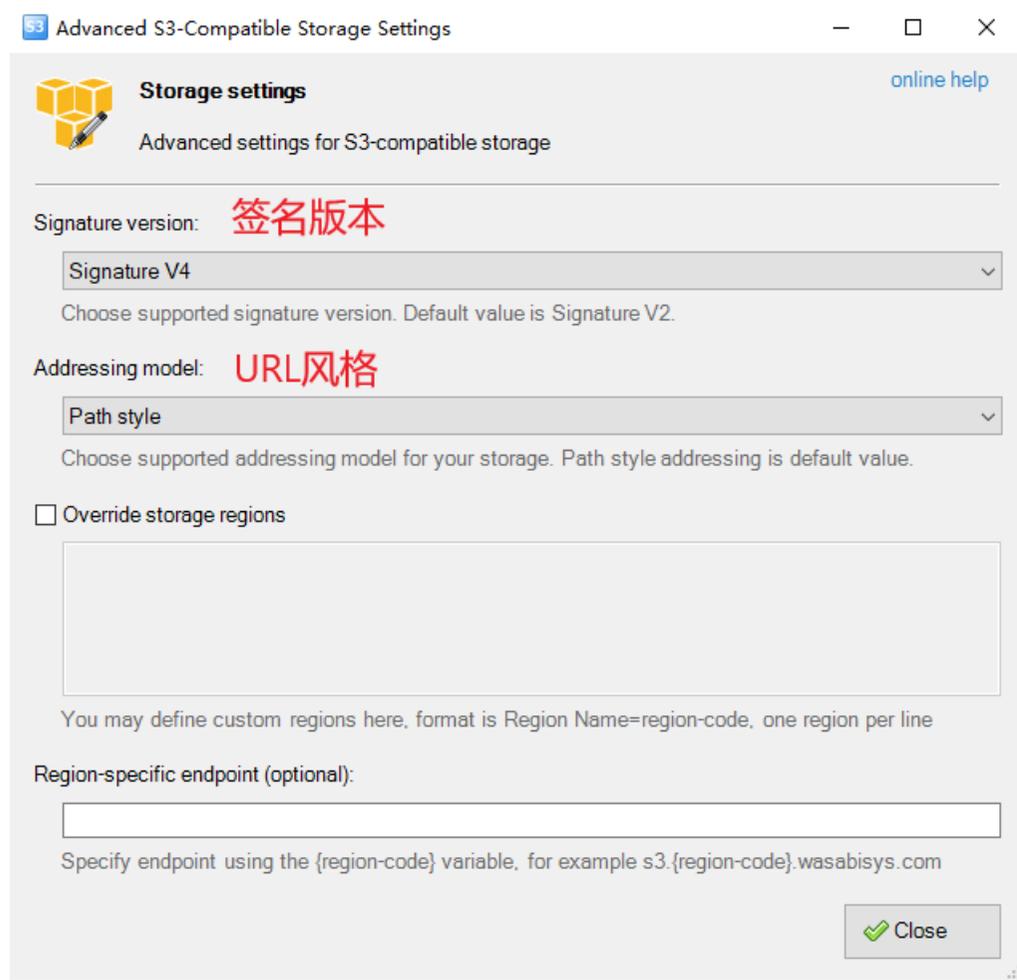
Turn this option on if you want to protect your Access Keys with a master password.

Use secure transfer (SSL/TLS)

If checked, all communications with the storage will go through encrypted SSL/TLS channel

签名版本及URL风格设置
[Advanced S3-compatible storage settings](#)

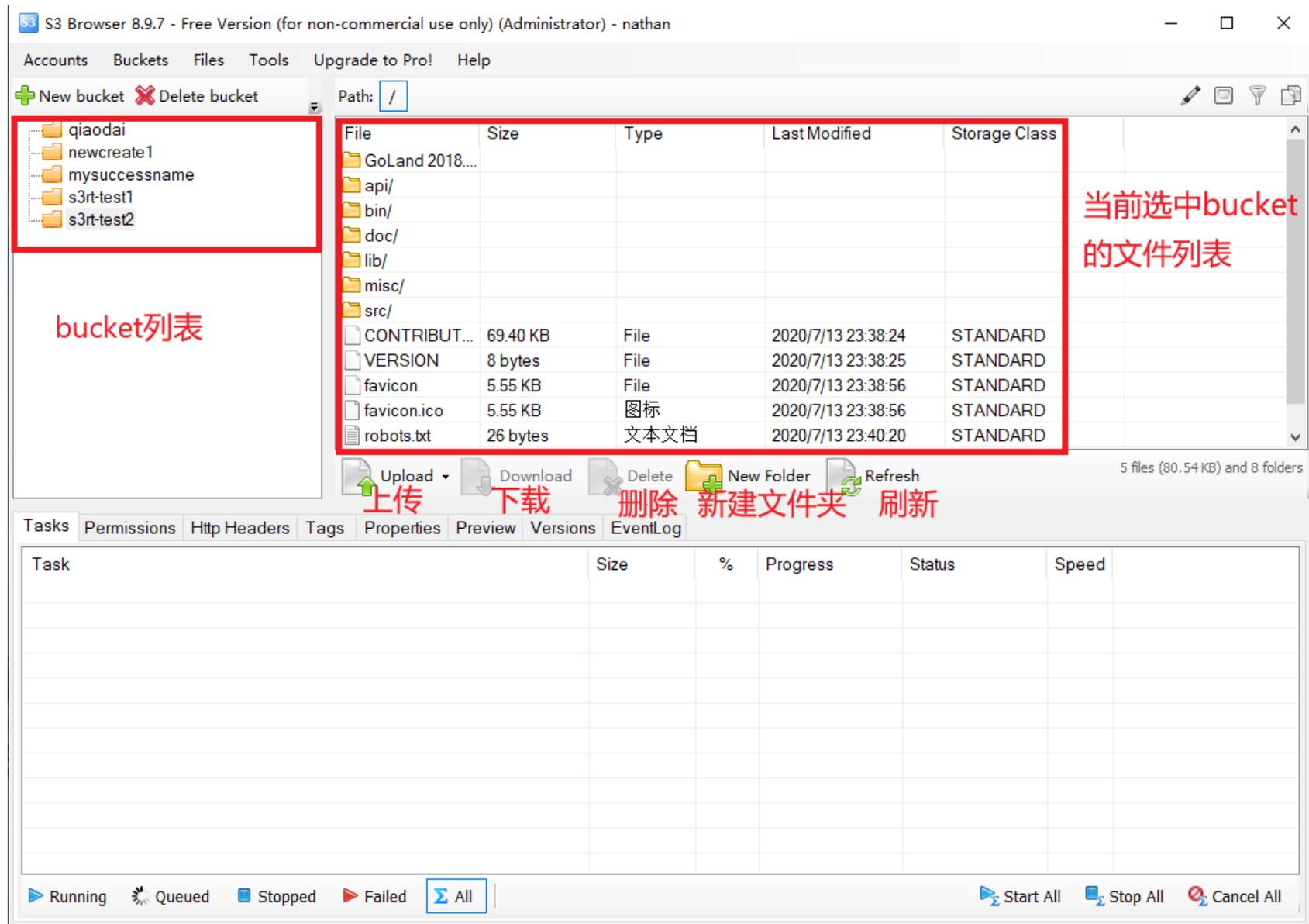
点击左下角的Advanced S3-compatible storage settings配置签名版本以及URL风格



修改成功点击Close 关闭当前设置,点击Add new account 保存配置,则成功创建用户。

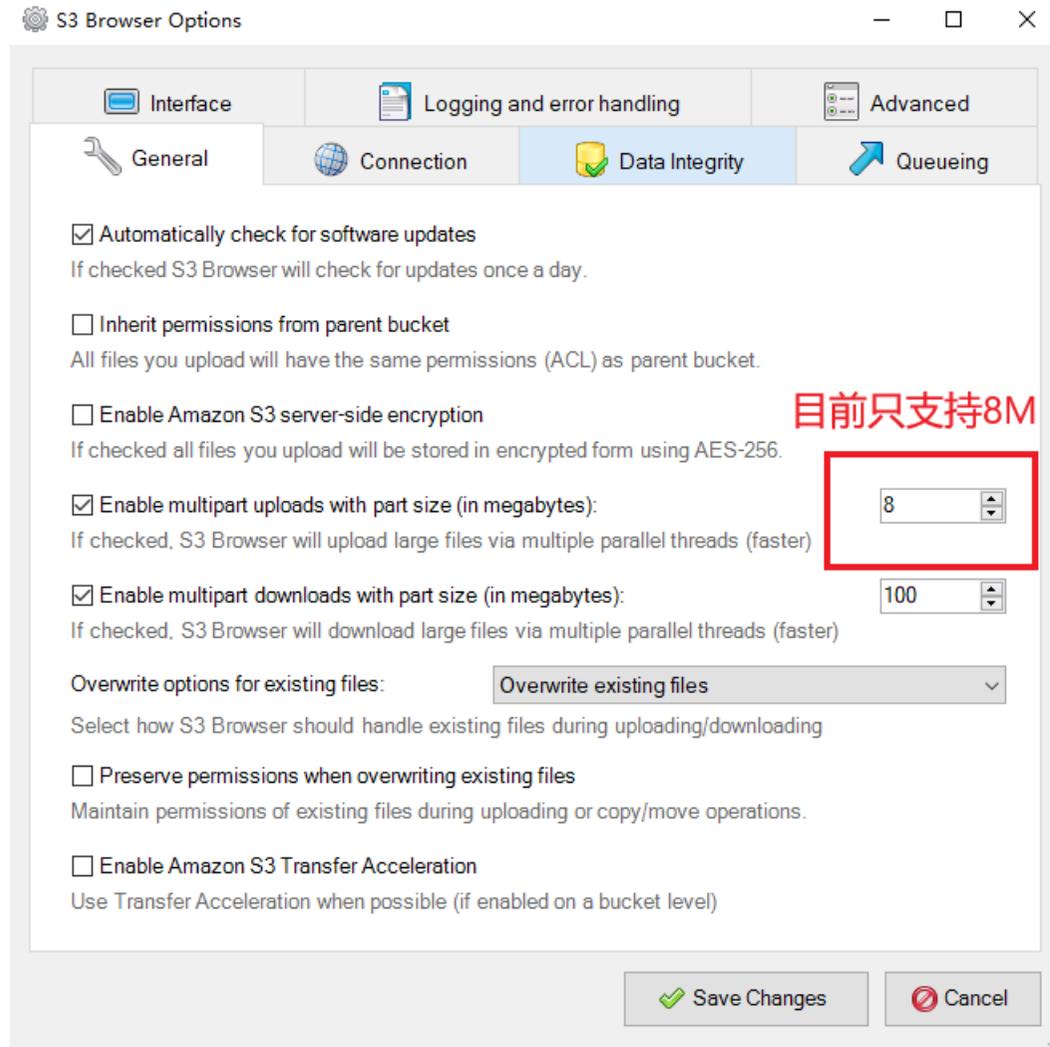
2.对象操作

控制台功能说明



特别说明: 目前分片大小只支持 8M 具体配置如下: 1. 点击上方工具栏 Tools, 在下拉列表中选择 Options, 选择 General, 在弹出页面中, 设置 Enable multipart uploads with size

(in megabytes) 为 8,如下图所示:



常见问题

1. 上传文件超过78G,报错400.显示分片大小为16MB

问题原因:

s3分片数限制为1万条,设置分片为8M是,只能满足78G左右以下文件的上传,如果文件大小超过78G,会自动调整分片大小,保证分片数小于1万条。目前us3后端s3协议只支持8M的分片,如果分片大小不对,会返回400错误。

解决方案:

- 使用us3cli工具进行上传(us3协议)。使用方式参考 [us3cli 工具简介](#)

网络文件系统 S3FS

功能说明

s3fs 工具支持将 Bucket 挂载到本地,像使用本地文件系统一样直接操作对象存储中的对象。

安装和使用

适用的操作系统 **Linux、MacOS**

适用s3fs版本: **v1.83及以上**

安装步骤

MacOS 环境

```
brew cask install osxfuse  
brew install s3fs
```

RHEL 和 CentOS 7 或更新版本通过 EPEL:

```
sudo yum install epel-release  
sudo yum install s3fs-fuse
```

Debian 9 和 Ubuntu 16.04 或更新版本

```
sudo apt-get install s3fs
```

CentOS 6 及其以下版本

需要编译 s3fs , 并且安装该程序

获取源码

首先, 您需要从 上将源码下载到指定目录, 以 /data/s3fs 为例:

```
1. cd /data  
2. mkdir s3fs  
3. cd s3fs  
4. wget https://github.com/s3fs-fuse/s3fs-fuse/archive/v1.83.zip
```

安装依赖项

CentOS 系统下安装依赖软件：

```
sudo yum install automake gcc-c++ git libcurl-devel libxml2-devel  
fuse-devel make openssl-devel fuse unzip
```

编译和安装 s3fs

进入安装目录,执行如下命令进行编译和安装：

```
1. cd /data/s3fs  
2. unzip v1.83.zip  
3. cd s3fs-fuse-1.83/  
4. ./autogen.sh  
5. ./configure  
6. make  
7. sudo make install  
8. s3fs --version #查看 s3fs版本号
```

可以查看 s3fs 的版本号,到此,s3fs 已经安装成功。

备注：在执行第五步,./configure 的过程中,可能会遇到以下的问题。汇总为：

报错：configure: error: Package requirements (fuse >= 2.8.4 libcurl >= 7.0 libxml-2.0 >= 2.6) were not met:

原因：fuse 版本过低,此时,您需要手动安装 fuse 2.8.4 及以上版本,安装命令示例如下：

1. yum -y remove fuse-devel #卸载当前版本的 fuse
2. wget https://.com/libfuse/libfuse/releases/download/fuse_2_9_4/fuse-2.9.4.tar.gz
3. tar -zxvf fuse-2.9.4.tar.gz
4. cd fuse-2.9.4
5. ./configure
6. make
7. make install
8. export
PKG_CONFIG_PATH=/usr/lib/pkgconfig:/usr/lib64/pkgconfig:/usr/local/lib/pkgconfig
9. modprobe fuse #挂载 fuse 内核模块
10. echo "/usr/local/lib" >> /etc/ld.so.conf
11. ldconfig #更新动态链接库
12. pkg-config --modversion fuse #查看 fuse 版本号,当看到“2.9.4”时,表示 fuse 2.9.4 安装成功

s3fs 使用方法

配置密钥文件

在 `${HOME}/` 目录中创建 `.passwd-s3fs` 文件。文件格式为 `[API 公钥:API 密钥]`。

公私钥获取方式具体请参考获取请参考 S3 的 `AccessKeyID` 和 `SecretAccessKey` 说明。

例如:

```
[root@10-9-42-233 s3fs-fuse-1.83]# cat ~/.passwd-s3fs
AKdDhQD4Nfyrr9nGPJ+d0iFmJGwQlgBTwxXXXXXXXXXXXX:7+LPnkPdPWhX2Aj+p/B1XVFi8bbbbbbbbbbbbbbbbbb
```

将文件设置读写权限。

```
chmod 600 ${HOME}/.passwd-s3fs
```

执行挂载操作

操作指令解释:

- 建立 US3 挂载文件路径 `${LocalMountPath}`
- 获取已创建的存储空间 (Bucket) 名称 `${UFileBucketName}`
注意:空间名称不带域名后缀,比如 US3 空间名称显示为test.cn-bj.ufileos.com,则`${UFileBucketName}=test`
- 根据 US3 存储空间所在地域,本地服务器是否在 UCloud 内网,参考支持 AWS S3 协议说明
- 执行命令。

参数说明如下:

```
s3fs ${UFileBucketName} ${LocalFilePath}
-o url={UFileS3URI} -o passwd_file=~/.passwd-s3fs
-o dbglevel=info
```

```
-o curldbg,use_path_request_style,allow_other
-o retries=1 //错误重试次数
-o multipart_size="8" //分片上传的大小为 8MB,目前仅支持该值 -o
multireq_max="8" //当上传的文件大于 8MB 是采用分片上传,目前UFile 的 S3
接入层不允许 PUT 单个文件超过 8MB,所以该值建议必填
-f //表示前台执行,后台执行则省略
-o parallel_count="32" //并行操作数,可以提高分片并发操作,建议不要超过 128
```

示例:

```
s3fs s3fs-test /data/vs3fs -o url=http://internal.s3-cn-bj.ufileos.com -o passwd_file=~/.passwd-s3fs -o dbglevel=info -o
curldbg,use_path_request_style,allow_other,nomixupload -o retries=1 -o multipart_size="8" -o multireq_max="8" -o parallel_count="32"
```

挂载效果

执行 `df -h` 指令,可以看到 s3fs 程序的运行。效果如下:

```
[root@10-9-25-74 test]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/vda1       20G   1.7G   17G   9% /
tmpfs           1.9G   0     1.9G   0% /dev/shm
/dev/vdb        20G   73M   19G   1% /data
s3fs            256T   0     256T   0% /data/test
```

此时,可以看到 /data/vs3fs 目录下的文件和指定 bucket 的文件,保持一致。也可以通过 `tree` 执行,查看文件结构。安装指令:`yum install -y tree` 效果如下:

```
[root@10-9-25-74 test]# tree
.
├── 100k
├── 11
├── 11.txt
├── 20150311083924_SSSAk.jpeg
├── aa
│   ├── timg (1).jpeg
│   ├── timg (1)\347\232\204\345\211\257\346\234\254.jpeg
│   ├── timg (2).jpeg
│   ├── timg.jpeg
│   ├── timg\347\232\204\345\211\257\346\234\254.jpeg
│   └── uCVwU0BmK1kGEpWR==Fm7N4qsbVLXTbsvsYAYM3cu6fY01543765067643compressflag.jpg
├── af2abb6523f7385c14ec7f4d846cbc544b6d62e4.jpg
├── cr-mgr\346\265\213\350\257\225.pages
├── hzw13_19202.jpg
├── New Folder
├── timg (1).jpeg
├── timg (1)\347\232\204\345\211\257\346\234\254.jpeg
├── timg (2).jpeg
├── timg.jpeg
├── timg\347\232\204\345\211\257\346\234\254.jpeg
├── uCVwU0BmK1kGEpWR==Fm7N4qsbVLXTbsvsYAYM3cu6fY01543765067643compressflag.jpg
├── \344\274\201\344\270\232\345\276\256\344\277\241\346\210\252\345\233\276_5402fd74-8353-4a7a-814c-fd4b68d373d4.png
└── \346\263\260\345\233\275\346\270\270\347\216\251\345\200\274\345\276\227\350\264\255\344\271\260\347\232\204\344\270\234\350\245\277.pages

2 directories, 21 files
```

文件上传和下载

挂载 US3 存储空间和后,可以像使用本地文件夹一样使用 US3 存储空间。

1. 拷贝文件到 `${LocalMountPath}` ,即是上传文件。
2. 将文件从 `${LocalMountPath}` 拷贝到其他路径,即下载文件。

注意:

1. 路径不符合 Linux 文件路径规范的路径,可以在 US3 管理控制台看到,但不会在 Fuse 挂载的 `\${LocalMountPath}` 下显示。
2. Fuse 使用枚举文件清单会比较缓慢,建议直接使用指定到具体文件的命令,如 `vim`、`cp`、`rm` 指定具体文件。

删除文件

将文件从 `\${LocalMountPath}` 删除掉,则 US3 存储空间中,该文件也被删除掉。

卸载US3存储空间

```
sudo umount \${LocalMountPath}
```

性能数据

写入吞吐量40MB/s左右 读取吞吐量能达到166 MB/s(跟并发量相关)

goofys

功能说明

goofys 工具同 s3fs, 也支持将 Bucket 挂载到本地,像使用本地文件系统一样直接操作对象存储中的对象。性能方面比 s3fs 更优.

安装与使用

适用的操作系统: **Linux, MacOS**

使用步骤

下载可执行文件:

Mac X86-64 Linux X86-64

使用如下命令解压到指定目录:

```
tar -xzvf goofys-0.21.1.tar.gz
```

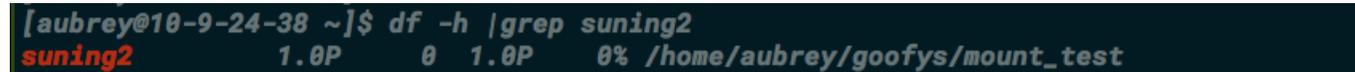
默认在 `$HOME/.aws/credentials` 文件里面配置 bucket 的公私钥, 格式如下:

```
[default]
aws_access_key_id = TOKEN_*****9206d
aws_secret_access_key = 93614*****b1dc40
```

执行挂载命令 `./goofys --endpoint your_ufile_endpoint your_bucket your_local_mount_dir`, 例如:

```
./goofys --endpoint http://internal.s3-cn-bj.ufileos.com/suning2 ./mount_test:
```

挂载效果如图:



```
[aubrey@10-9-24-38 ~]$ df -h |grep suning2
suning2 1.0P 0 1.0P 0% /home/aubrey/goofys/mount_test
```

测试挂载是否成功, 可以拷贝一个本地文件到 `mount_test` 目录, 看是否上传到 US3。

其它操作（删除,上传,获取,卸载）

同 s3fs, 可参考上面的 s3fs 操作

性能数据

4核8G 的 UHost 虚拟机, 上传 500MB 以上的文件, 平均速度可达140MB/s

基于 US3 的 FTP 服务

功能说明

对象存储支持通过 FTP 协议直接操作 Bucket 中的对象和目录, 包括上传文件、下载文件、删除文件 (不支持进入文件夹)。

安装和使用

适用的操作系统: **Linux**

安装步骤

搭建环境

使用 s3fs 工具将 Bucket 挂载到本地。具体安装方式步骤参考基于 S3FS、US3 搭建网络文件系统的内容。

安装依赖项

先检查下本地是否有 FTP 服务,执行命令 `rpm -qa | grep vsftpd`,如果显示未安装,则执行以下命令,安装 FTP。

运行以下命令安装 vsftpd。

```
yum install -y vsftpd
```

开启本地 **fpd** 服务

执行以下命令,开启 ftp 服务。

```
service vsftpd start
```

S3FS 使用方法

添加账户

1. 运行以下命令创建 `ftptest` 用户,并且设置指定目录。

```
useradd ${username} -d {SpecifiedDirectory}
```

(删除用户命令:`sudo userdel -r newuser`)

2. 运行以下命令修改 `ftptest` 用户密码。

```
passwd ${username}
```

客户端使用

此时,您可以在外部任何一台机器上连接该服务器,输入您的用户名和密码,来管理 bucket 的文件

```
ftp ${ftp_server_ip}
```

s3cmd

功能说明

s3cmd是一个免费的命令行工具,用于使用S3协议上传、检索和管理数据,它最适合熟悉命令程序的用户,广泛用于批处理脚本和自动备份。

安装和使用

适用的操作系统: **Linux、MacOS、Windows**

安装步骤

1. 下载安装包

<https://s3tools.org/download> ,这里以目前最新版本2.1.0为例

2. 解压安装包

```
tar xzvf s3cmd-2.1.0.tar.gz
```

3. 移动路径

```
mv s3cmd-2.1.0 /usr/local/s3cmd
```

4. 创建软连接

```
ln -s /usr/local/s3cmd/s3cmd /usr/bin/s3cmd (权限不足可以使用sudo)
```

5. 执行配置命令,填写必要信息(直接跳过也可以,可以放在下一步手动填写)

```
s3cmd --configure
```

6. 填写配置

```
vim ~/.s3cfg
```

打开当前配置,填写以下参数

```
access_key = "TOKEN公钥/API公钥"
```

```
secret_key = "TOKEN私钥/API私钥"
```

```
host_base = "s3协议域名,例如: s3-cn-bj.ufileos.com"
```

```
host_bucket = "请求风格,例如: %(bucket)s.s3-cn-bj.ufileos.com"
```

```
multipart_chunk_size_mb = 8 "us3 支持的s3协议分片大小为8M,所以这里只能填8"
```

access_key: 参考Token公钥/API公钥

secret_key: 参考 Token私钥/API私钥

host_base: 参考 s3协议域名

示例配置项

```
[default]
access_key = "TOKEN_xxxxxxxxx"
access_token =
add_encoding_exts =
add_headers =
bucket_location = US
```

```
check_ssl_certificate = True
check_ssl_hostname = True
connection_pooling = True
content_disposition =
content_type =
default_mime_type = binary/octet-stream
delay_updates = False
delete_after = False
delete_after_fetch = False
delete_removed = False
dry_run = False
enable_multipart = True
encrypt = False
expiry_date =
expiry_days =
expiry_prefix =
follow_symlinks = False
force = False
get_continue = False
gpg_passphrase =
guess_mime_type = True
host_base = s3-cn-bj.ufileos.com
host_bucket = %(bucket)s.s3-cn-bj.ufileos.com
human_readable_sizes = False
invalidate_default_index_on_cf = False
```

```
invalidate_default_index_root_on_cf = True
invalidate_on_cf = False
kms_key =
limit = -1
limitrate = 0
list_md5 = False
log_target_prefix =
long_listing = False
max_delete = -1
mime_type =
multipart_chunk_size_mb = 8
multipart_max_chunks = 10000
preserve_attrs = True
progress_meter = True
proxy_host =
proxy_port = 80
public_url_use_https = False
put_continue = False
recursive = False
recv_chunk = 65536
reduced_redundancy = False
requester_pays = False
restore_days = 1
restore_priority = Standard
secret_key = "xxxxxxxxxxxxxxxxxxxxxx"
```

```
send_chunk = 65536
server_side_encryption = False
signature_v2 = False
signurl_use_https = False
simpledb_host = sdb.amazonaws.com
skip_existing = False
socket_timeout = 300
stats = False
stop_on_error = False
storage_class =
throttle_max = 100
upload_id =
urlencoding_mode = normal
use_http_expect = False
use_https = False
use_mime_magic = True
verbosity = WARNING
website_index = index.html
```

使用方法

1. 上传文件

```
s3cmd put test.txt s3://bucket1
```

2. 删除文件

```
s3cmd del s3://bucket1/test.txt
```

3. 下载文件

```
s3cmd get s3://bucket1/test.txt
```

4. 拷贝文件

```
s3cmd cp s3://bucket1/test.txt s3://bucket2/test.txt
```

其他常用操作

1. 上传文件夹

```
s3cmd put -r ./dir s3://bucket1/dir1
```

2. 下载文件夹

```
s3cmd get -r s3://bucket1/dir1 ./dir1
```

3. 删除文件夹

```
s3cmd del s3://bucket1/dir1
```

4. 列取bucket列表

```
s3cmd ls
```

5. 列取文件列表

```
s3cmd ls s3://bucket1
```

6. 归档文件取回

```
s3cmd restore s3://bucket1
```

rclone

功能说明

rclone是一个命令行程序,用于管理云存储上的文件,支持s3协议

安装和使用

安装步骤

```
curl https://rclone.org/install.sh | sudo bash
```

//参考 <https://rclone.org/install/>

配置

```
rclone config
```

配置参考

```
[s3] //这里可以填写s3,ucloud等自定义名,作为命令前缀
type = s3
provider = Other
env_auth = false
access_key_id = xxxxxxxx
secret_access_key = xxxxxxxxxxxx
endpoint = http://s3-cn-bj.ufileos.com //参考
location_constraint = cn-bj
acl = private
bucket_acl = private // public/private
chunk_size = 8M //目前只支持8M分片
```

access_key_id: 参考Token公钥/API公钥

secret_access_key: 参考 Token私钥/API私钥

endpoint: 参考 s3协议域名

使用方法

说明: 以下命令前缀 (配置内的中括号内容) 以remote为例, 使用过程中需要自行修改

1. 查看所有bucket

```
rclone lsd remote:
```

2. 列取文件列表

```
rclone ls remote:bucket
```

3. 上传文件

```
rclone copy ./test.txt remote:bucket
```

4. 删除文件

```
rclone delete remote:bucket/test.txt
```

Cyberduck

功能说明

Cyberduck是一个跨平台的文件管理器,支持s3协议,适用于macos系统

安装和使用

安装步骤

下载地址:<https://cyberduck.io/download/>

打开软件即可使用

配置

Cyberduck配置目录为：

```
~/Library/Group Containers/G69SCX94XU.duck/Library/Application Support/duck/
```

在配置目录下创建命名为“default.properties”的文件

配置参考

文件内容为：

```
s3.upload.multipart.lookup=false
```

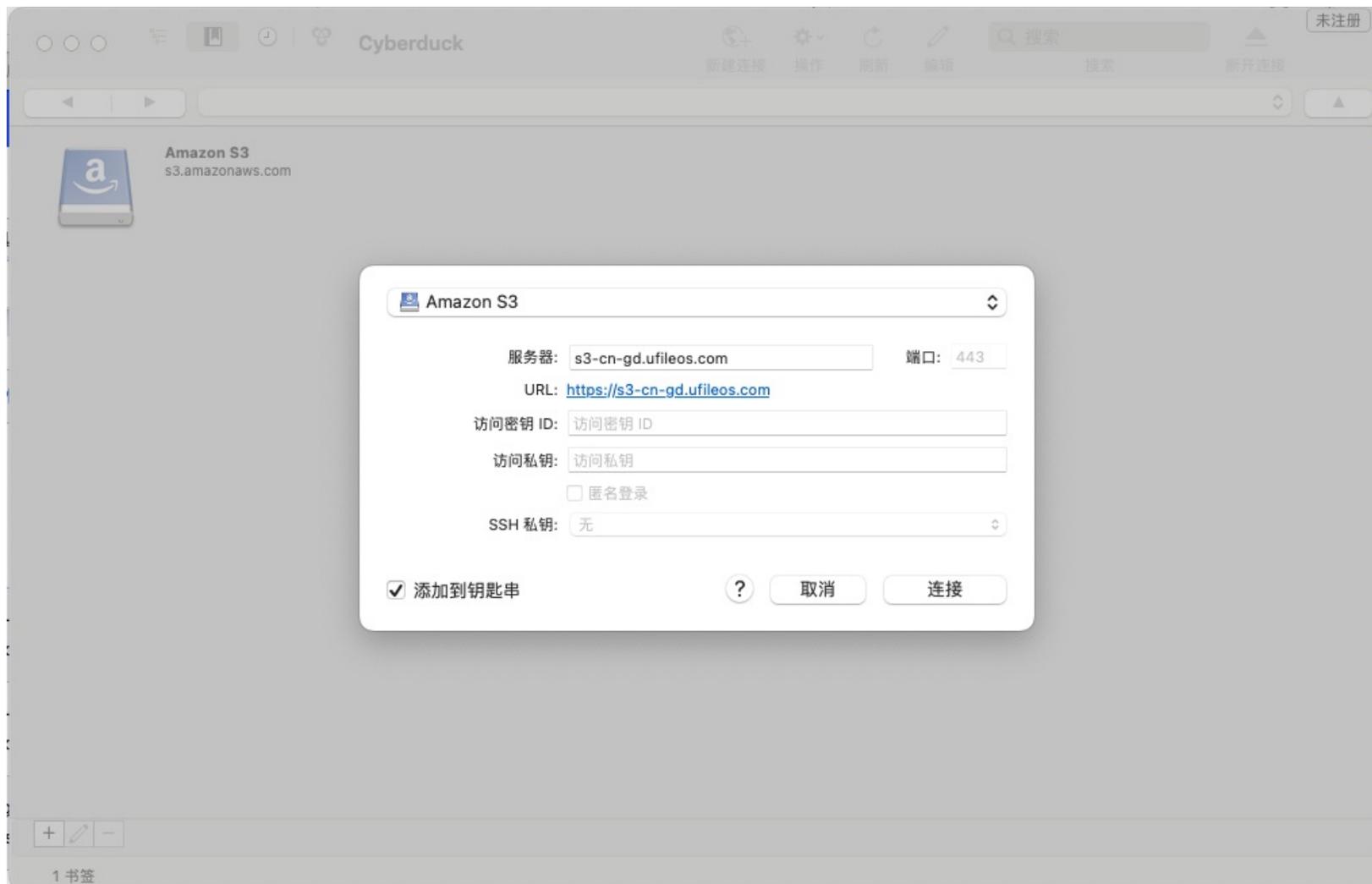
```
s3.upload.multipart.size=8388608
```

完成后效果如下图所示

□

使用方法

按照下图所示配置



access_key_id: 参考Token公钥/API公钥

secret_access_key: 参考 Token私钥/API私钥

endpoint: 参考 s3协议域名

支持操作说明

支持上传, 下载, 删除, 列表, 同步, 预签名url等操作。

不支持移动, 拷贝, 重命名, 新建加密库, 恢复, 归档等。

API 列表

对象存储 US3 提供以下相关 API:

管理类 API

创建Bucket-CreateBucket

创建US3令牌-CreateUFileToken

删除Bucket-DeleteBucket

删除令牌-DeleteUFileToken

获取Bucket信息-DescribeBucket

获取令牌信息-DescribeUFileToken

更改Bucket属性-UpdateBucket

更新令牌-UpdateUFileToken

数据类 API

获取目录文件列表-ListObjects

上传文件-PutFile

表单上传-PostFile

秒传文件-UploadHit

下载文件-GetFile

查询文件基本信息-HeadFile

删除文件-DeleteFile

初始化分片-InitiateMultipartUpload

上传分片-UploadPart

完成分片-FinishMultipartUpload

放弃分片-AbortMultipartUpload

其他API请参考:[对象存储 API 文档](#)

SDK 列表

US3 提供了各种主流语言的 SDK,用于对象存储相关功能的调用,各个 SDK 的 Git 仓库链接如下:

JS SDK

C SDK

PHP SDK

Python SDK

Go SDK

C# SDK

NodeJS SDK

Java SDK

iOS SDK

Android SDK

C++ SDK

备注:使用 JS SDK 需要指派跨域配置的工单至技术支持,请参考 [常见问题](#)。

在微信小程序端上传文件到US3

简介

本文将介绍并梳理使用小程序原生方法上传文件到US3的流程。

预先准备

1. 在控制台上创建一个存储桶, 以及具有操作该桶的权限的令牌。
2. 在控制台上获取桶名, 终端节点(endpoint)以及令牌的公钥和私钥。

上传步骤

接下来我们将以一个图片文件为例, 使用PUT 和POST 两种方法来实现文件上传。

1. 首先我们需要拼接好上传文件的url, 如果您选择的是PUT方法上传, url通常如下:

```
https://<bucket_name>.<endpoint>/<file_name_with_prefix_on_US3>
```

而POST方法上传的话, 由于文件在US3上的前缀(prefix)和文件名通过POST表单参数来传递, 因此我们不需要在url中写入这两个部分:

```
https://<bucket_name>.<endpoint>/
```

- 上传文件的接口需要我们使用公钥和私钥对http请求的资源计算一个签名,以供服务端查看请求使用的公私钥是否有对应的权限。计算方法请参考 API签名算法 。这里提供一个计算签名的js代码demo:

```
const sign = (method, publicKey, privateKey, md5, contentType, date, bucketName, fileName) =>{
const CryptoJS = require("crypto-js") //这里使用了crypto-js加密算法库,安装方法会在后面说明
const CanonicalizedResource = `/${bucketName}/${fileName}`
const StringToSign = method + "\n"
+ md5 + "\n"
+ contentType + "\n"
+ date + "\n"
+ CanonicalizedResource //此处的md5以及date是可选的,contentType对于PUT请求是可选的,对于POST请求则是必须的
let Signature = CryptoJS.HmacSHA1(StringToSign, privateKey)
Signature = CryptoJS.enc.Base64.stringify(Signature)
const Authorization = "UCloud" + " " + publicKey + ":" + Signature
return Authorization
}
```

上方demo中的crypto-js是一个常用的javaScript加密算法库,我们建议您使用npm进行安装,具体步骤如下:

1. 在小程序的开发机上安装 nodejs ,完成后尝试在命令行界面运行 npm ,以验证安装以及配置是否成功。
 2. 在小程序的跟目录下运行 npm init ,来初始化一个npm项目。
 3. 运行 npm i crypto-js 命令来安装crypto-js。也可以参考 npm 的官方页面
 4. 最后,使用小程序构建一下npm包,请参考 构建npm
3. 接下来将给出两个demo以介绍如何使用PUT和POST方法来上传文件:

PUT方法

参考文档 使用PUT方法上传文件

```
function uploadByPut(){
const publicKey = "xxxx-xxxxx-xxxx-xxxxx-xxxxx"
const privateKey = "xxxxx-xxx-xxxx-xxxx-xxxxx"
const fileName = "filename"
const bucketName = "bucketname"
wx.chooseImage({
count: 1, // 设置最多1张
sizeType: ['original', 'compressed'], //所选的图片的尺寸
sourceType: ['album', 'camera'], //选择图片的来源
success(res) {
const image = wx.getFileSystemManager().readFileSync(res.tempFilePaths[0])
//由于这一代码仅是demo,此处将md5和date两个参数设置为空字符串,在正式的小程序开发中,建议填写这两个参数
const auth = sign("PUT", publicKey, privateKey, "", "image/jpeg", "", bucketName, fileName)
wx.request({
url: `http://${bucketName}.cn-bj.ufileos.com/${fileName}`,
method: "PUT",
header: {
'Authorization':auth,
'content-type': 'image/jpeg'
},
data: image,
```

```
success: function (reg) {  
  console.log(reg)  
}  
})  
}  
})  
}
```

POST方法

此处我们使用小程序原生的wx.uploadFile函数来实现上传。该方法使用的content-type为multipart/form-data, 调用的接口也同样是POST上传的US3接口, 请参考 [使用POST表单上传文件](#)

```
function uploadByUplodaFile(){  
  const publicKey = "xxxx-xxxxx-xxxx-xxxxx-xxxxx"  
  const privateKey = "xxxxx-xxx-xxxx-xxxx-xxxxx"  
  const fileName = "filename"  
  const bucketName = "bucketname"  
  wx.chooseImage({  
    count: 1, // 设置最多1张  
    sizeType: ['original', 'compressed'], //所选的图片的尺寸  
    sourceType: ['album', 'camera'], //选择图片的来源  
    success (res) {  
      const tempFilePaths = res.tempFilePaths  
      //注意此处的签名方法为POST, 签名计算所使用的content-type仍然为文件本身的mime-type, 即"image/jpeg"
```

```
const auth = sign("POST", publicKey, privateKey, "", "image/jpeg", "", bucketName, fileName)
wx.uploadFile({
  url: `https://${bucketName}.cn-bj.ufileos.com/`,
  filePath: tempFilePaths[0],
  name: 'file',
  //此处FormData为表单参数,在此处指定要上传的文件名,以及鉴权签名
  formData: {
    'FileName': fileName,
    'Authorization': auth
  },
  success (res){
    console.log(res)
  }
})
}
```

那么如果情况不允许使用微信的原生文件上传方法,那么也可以通过拼接multipart/form-data的方式来进行上传。这种方式会麻烦一些,因此如无特殊需求,还是建议使用原生方法进行上传。

```
function uploadByPost(){
  const publicKey = "xxxx-xxxxx-xxxx-xxxxx-xxxxx"
  const privateKey = "xxxxx-xxx-xxxx-xxxx-xxxxx"
  const fileName = "filename"
```

```
const bucketName = "bucketname"
// boundary可以自定义,这里使用US3官方文档中示例所用的boundary字符串
const boundary = "----UCloudPOSTFormBoundary"
// 将string类型数据转换成binary数组
const strToBinary = (str) => {
  let res = []
  for (var i = 0; i < str.length; i++) {
    res.push(str.charCodeAt(i));
  }
  return res
}
wx.chooseImage({
  count: 1, // 设置最多1张
  sizeType: ['original', 'compressed'], //所选的图片的尺寸
  sourceType: ['album', 'camera'], //选择图片的来源
  success(res) {
    //将文件内容读取出来并转换为Uint8Array
    const image = new Uint8Array(wx.getFileSystemManager().readFileSync(res.tempFilePaths[0]))
    //这里的签名规则和原生请求的规则相同
    const auth = sign("POST", publicKey, privateKey, "", "image/jpeg", "", bucketName, fileName)
    //拼接请求体的前半部分,可见和原生请求中的FormData内容相对应。
    //需要将这部分也同样转换成binary数组
    const start = strToBinary(
      `--${boundary}` + '\r\n' +
      'Content-Disposition: form-data; name="FileName"' + '\r\n' +
```

```
'\r\n' +
fileName + '\r\n' +
`--${boundary}` + '\r\n'+
'Content-Disposition: form-data; name="Authorization"' + '\r\n'+
'\r\n' +
auth + '\r\n' +
`--${boundary}` + '\r\n' +
'Content-Disposition: form-data; name="file"; filename="MyFilename.jpg"' + '\r\n' +
'Content-Type: image/jpeg' + '\r\n' +
'\r\n')
//请求体的结尾,转换为binary数组
const end = strToBinary('\r\n' + `--${boundary}--`)
//将前半部分的内容,文件的数据,以及结尾拼接在一起,形成整个请求的binary数组
const request = start.concat(Array.prototype.slice.call(image), end);
//将整个request转换成Uint8Array,并通过.buffer参数获取ArrayBuffer对象,wx.request函数最终将请求转化为对应的字符串
const data = new Uint8Array(totalArray).buffer
wx.request({
url: `https://${bucketName}.cn-bj.ufileos.com/`,
method: "POST",
header: {
'content-type': `multipart/form-data; boundary=${boundary}`,
},
data: data,
success: function (reg) {
console.log(reg)
```

```
}  
}  
}  
}  
}
```

如何将US3上的文件打包下载

简介

此方案主要是为了解决将US3上Bucket里的文件打包后下载的问题。通过打包服务将US3上指定的文件打包成ZIP压缩包,从而方便将多个文件批量下载到本地。

预先准备

- 在云主机控制台上创建一台linux操作系统的UHost云主机。
- 在US3控制台上拿到具有对于目标桶上传下载,以及列取权限的令牌。

基本原理

工具的工作原理如下:

1. 工具本身会拉起一个HTTP服务器
2. 用户发送POST请求以提交打包任务,您会在请求的response中拿到最终将会生成的zip压缩包的Key
3. 打包工具根据您提交的请求中的信息,将US3中的文件下载到本地,并进行打包
4. 将本地打包好的zip压缩包上传到US3中的指定位置
5. 您可以根据第二条中获取的zip压缩包Key下载文件

操作步骤

我们这里假定您已经创建了云主机UHost,并且在US3控制台上拿到了对应的令牌。

1. 下载打包工具 工具包
2. 将工具包解压缩 `unzip US3-PACK.zip`
3. 修改工具中的server_conf.json配置文件,配置文件如下:

```
{
  "log": {
    "LogDir": "logs",
    "LogPrefix": "zip_",
    "LogSuffix": ".log",
    "LogSize": 50,
    "LogLevel": "DEBUG"
  },
  "http": {
    "Ip": "0.0.0.0",
    "Port": 80
  }, //服务监听的端口和ip
  "us3_config": {
    "public_key": "xxxxxxxxxxxxxxxx", //Token中的公钥
    "private_key": "xxxxxxxxxxxxxxxx" //Token中的私钥
  }
}
```

```
}
```

4. 执行./US3-PACK以启动服务,您也可以使用后台进程来执行该服务nohup ./US3-PACK &
5. 此时您可以发送POST请求到服务的根url (例如http://xxx.xxx.xxx.xxx),请求参数有两种类型,分别对应指定某个前缀下的所有文件进行打包的任务,以及指定具体文件进行打包的任务。

注意,如果您申请的UHost云主机只有内网IP,那么请您在同一台云主机上,或者同一VPC内部发送打包的POST请求。

指定前缀进行打包

```
{  
  "action": "GetUFileZipRequest",  
  "prefix": "prefix",  
  "bucket_name": "BucketName",  
  "file_host": "internal-cn-sh2-01.ufileos.com"  
}
```

其中:

- action字段指定request类型
- bucket_name对应您所需打包文件所在存储桶的桶名
- prefix为您所需打包文件所在的前缀(文件夹)路径

- file_host为您访问桶所使用的endpoint,请您参考 [地域和域名](#)

指定文件列表进行打包

```
{
"action": "GetUFileZipByListRequest",
"file_list": "prefix/key1,prefix/key2,prefix/key3",
"bucket_name":"BucketName",
"file_host":"internal-cn-sh2-01.ufileos.com"
}
```

其中:

- file_list字段指定要打包的文件名,注意此处文件名包括文件的前缀,但不包括桶名
- 其他字段同上

我们在文件包中提供了请求的json示例,您可以在压缩包中找到event.json文件,根据上文中的请求格式以及您想要提交的打包任务情况来更改json中的内容,并在同一台云主机上通过这一命令进行测试: curl -X POST -d@event.json localhost

在发送完请求后,您会在返回中收到压缩包的地址。请求返回格式如下:

```
{
"Action": "GetUFileZipByListRequest",
"prefix": "prefix",
```

```
"RetCode": 0,  
"ErrMsg": "",  
"Key": "output/xxxx-xxxx-xxxxx-xxxx-xxxxxxx.zip"  
}
```

其中key字段即为打包请求处理完毕后,工具上传到US3中的压缩包的对象名

获取压缩包

您可以通过logs/文件夹下的日志文件来查看任务是否完成

最后,您可以使用http客户端工具下载这一文件,例如:

```
wget http://bucket.internal-cn-sh2-01.ufileos.com/output/xxxx-xxxx-xxxxx-xxxx-xxxxxxx.zip
```

性能测试

在使用1核1G内存的UHost主机,内网传输数据的情况下,打包55个20M文件(总大小1.15G),的时间大概为30S。

数据迁移方案

背景

对象存储具有容量大、成本低、高扩展性以及高可靠性等优势,面对增量数据导致的存储容量瓶颈、成本增加等问题。用户可通过数据迁移的方式,将第三方的存储集群数据迁移至 US3 的存储空间中,充分利用对象存储按需收费,无须购买额外硬件资源的优势,对数据进行归档,以实现成本最优。本文介绍了如何从其他源站将数据迁移到 US3 的方式。

US3 镜像回源功能

用户在 US3 创建存储空间后,通过配置存储空间的回源配置,US3 将在接收到访问请求时,会先在 US3 本地查找文件,如果文件不存在,则会回源到用户指定的源站获取文件,并且在 US3 本地存储一份。下一次访问同一个文件时就会直接从 US3 本地直接返回给用户。

具体设置方法请参照：[镜像回源](#)

优点：

1. 客户的请求可以快速无缝的切换到 US3。
2. 数据迁移到 US3 的过程无感知。

缺点：

1. 部分未访问到的文件无法迁移到 US3。
2. 访问 US3 不存在的文件需要回源,访问延时会略有增加。

US3 数据迁移工具

介绍

US3SYNC 是对象存储 US3 提供的一款将数据迁移至 US3 存储空间(Bucket)的工具。您可以将 US3SYNC 部署在本地服务或者云主机上,轻松将您其他云存储的数据迁移到 US3。

适用情况

阿里云对象存储数据迁移到 US3 对象存储 七牛云对象存储数据迁移到 US3 对象存储 US3 对象存储不同 bucket 之前的数据迁移 支持 s3 协议的对象存储迁移到 US3 对象存储

详情请参照:US3SYNC 迁移工具

数据库备份方案

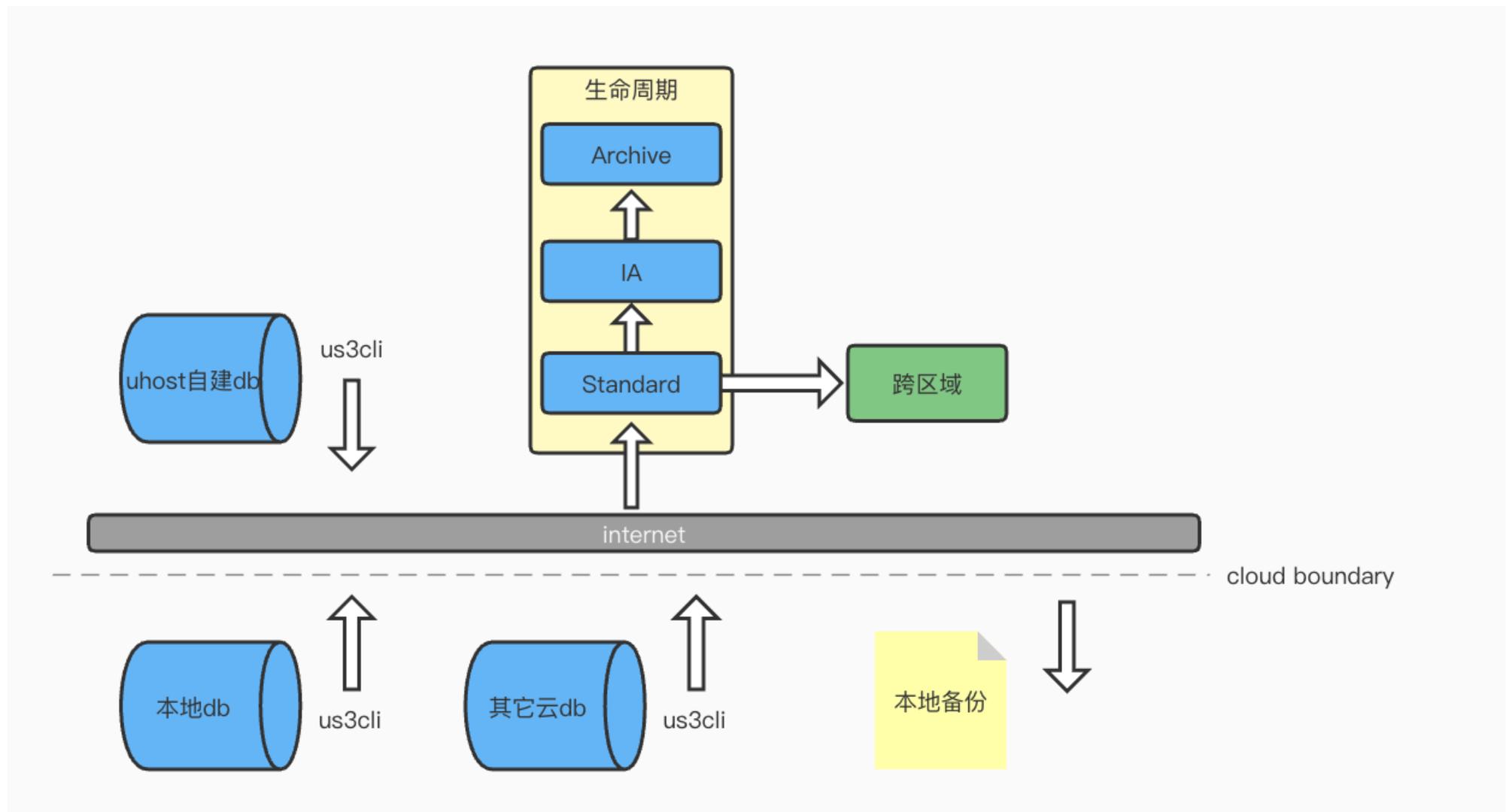
背景

对象存储作为海量非结构化数据的云存储应用,面对日益上涨的数据库备份场景,可以有效帮助用户缩减备份流程,降低备份成本,有效提升用户体验。本文介绍了如何基于 US3 完成数据库备份。

应用场景

目前 US3 数据库备份场景主要有以下三类:

1. 备份与恢复:备份方式推荐使用 us3cli 备份到 US3 中。us3cli 支持本地备份恢复与流式备份恢复,通过流式功能可以帮助用户完成数据不落地备份与恢复。
2. 分级存储:针对需要定时清理备份、缩减备份成本的用户,US3 支持生命周期功能。通过控制台指定生命周期规则,可以帮助用户完成:1、定期清理;2、定期转入低频;3、定期转入归档;
3. 异地备份:针对需要更高安全级别的用户,US3 支持跨区域复制功能。通过控制台配置跨区域复制功能,可以帮助用户在上传备份的同时,完成数据的异地备份。



方案优势

1. 使用 us3cli 进行流式备份以及流式恢复,完成不落地备份与恢复,可以避免落盘操作。
2. 使用 US3 生命周期 功能,配合定期删除、低频存储、归档存储可以实现数据分级存储,帮助用户节约存储成本。
3. 使用 US3 跨区域复制,为备份数据进行异地容灾,提高备份数据安全性。

方案实施

使用us3cli进行流式备份，流式恢复

1. 下载 us3cli
2. 使用 us3cli config 命令配置好密钥, endpoint等参数。[参考文档](#)
3. 使用 us3cli 进行备份恢复,此处展示最简命令,其他备份命令请结合自己业务类比实现

```
# 注意如果,欲使用低频存储(IA)或者冷存储(ARCHIVE),请在命令参数storageclass中指定,支持三种值:STANDARD, IA, ARCHIVE
```

```
# 注意如果,备份时指定了storageclass参数为ARCHIVE,需要提前对该文件restore
```

```
./us3cli restore us3://<bucketName>/<backupKey>
```

```
# 逻辑备份
```

```
# 全库备份
```

```
mysqldump -A | ./us3cli rcat us3://<bucketName>/<backupKey> --storageclass <storage-class>
```

```
# 分库备份
```

```
mysqldump -B database1 database2 | ./us3cli rcat us3://<bucketName>/<backupKey> --storageclass <storage-class>
```

逻辑备份恢复

```
./us3cli cat us3://<bucketName>/<backupKey> --storageclass <storage-class> | mysql
```

```
# -----
```

xtrabackup物理备份

全量备份

```
innobackupex --stream=tar | ./us3cli rcat us3://<bucketName>/<backupKey> --storageclass <storage-class>
```

增量备份

```
innobackupex --stream=tar --extra-lsdir=/data/backup/chkpoint /data/backup/tmp/ | ./us3cli rcat us3://<bucketName>/<backupKey> --storageclass
```

```
<storage-class> # 全备
```

```
innobackupex --stream=xbstream --incremental --extra-lsdir=/data/backup/chkpoint --incremental-basedir=/data/backup/chkpoint /data/backup/tmp/ |
```

```
./us3cli rcat us3://<bucketName>/<backupKey> --storageclass <storage-class> # 增备
```

```
# xtrabackup物理备份恢复, 需要提前转移原来的DB数据, 备份恢复后需要重启服务
```

全量备份恢复

```
# full-backupKey为全量备份使用的key:
```

```
./us3cli cat us3://<bucketName>/<backupKey> --storageclass <storage-class> | tar xf - -C /data/backup/full/
```

```
innobackupex --apply-log /data/backup/full/
```

```
innobackupex --copy-back --rsync /data/backup/full/
```

增量备份恢复

```
# full-backupKey为全量备份使用的key, incre-backupKey为增量备份使用的key:
```

```
./us3cli cat us3://<bucketName>/<backupKey> --storageclass <storage-class> | tar xf - -C /data/backup/base/
```

```
innobackupex --apply-log --redo-only /data/backup/base/
```

```
./us3cli cat us3://<bucketName>/<backupKey> --storageclass <storage-class> | xbstream -x -C /data/backup/incre
```

```
innobackupex --apply-log /data/backup/base --incremental-dir=/data/backup/incre
```

```
innobackupex --copy-back --rsync /data/backup/base
```

lvm snapshot物理备份恢复

备份,/snap-lvm0为备份lv 挂载点,/data-lvm0为源lv挂载点

```
tar czf - /snap-lvm0/* | ./us3cli rcat us3://<bucketName>/<backupKey> --storageclass <storage-class>
```

恢复,本地数据被清空,--strip-components 用来去除压缩快照时产生的第一层目录

```
./us3cli cat us3://<bucketName>/<backupKey> --storageclass <storage-class> | tar xzf - -C /data-lvm0/ --strip-components 1
```

合并历史快照, /snap-lvm0为快照lv挂载点

```
./us3cli cat us3://<bucketName>/<backupKey> --storageclass <storage-class> | tar xzf - -C /snap-lvm0/ --strip-components 1
```

```
lvconvert --merge <vg>/<snap-lv>
```

针对有特殊需求的用户,这边提供了相应的逻辑备份命令,其他备份命令请类比实现:

压缩

备份

```
mysqldump -A | gzip | ./us3cli rcat us3://<bucketName>/<backupKey> --storageclass <storage-class>
```

恢复

```
./us3cli cat us3://<bucketName>/<backupKey> --storageclass <storage-class> | gzip -d | mysql
```

加密

备份,使用aes256,指定密码文件key file在备份路径中进行压缩

```
mysqldump -A | openssl enc -e -aes256 -in - -out - -kfile <key file> | ./us3cli rcat us3://<bucketName>/<backupKey> --storageclass <storage-class>
```

恢复

```
./us3cli cat us3://<bucketName>/<backupKey> --storageclass <storage-class> | openssl enc -d -aes256 -in - -out - -kfile <key file> | mysql
```

备注： 如果不希望异常情况终止任务,可以使用 --retrycount 参数来将retry次数设置为一个比较大的值,默认为 10。每次执行失败会开始重试,第 5 次重试开始每次重试会等待 5s,请合理计算重试次数。

使用生命周期实现定期删除

1. 打开对象存储控制台,进入备份使用的 bucket 详情页



2. 点击生命周期 tab,进入生命周期配置页



3. 为该 bucket 配置定期删除任务
4. 当备份文件超过配置期限, 文件会被自动删除

使用跨区域复制进行异地容灾

1. 打开对象存储控制台, 进入备份使用的 bucket 详情页



2. 点击开区域复制 tab, 进入跨区域复制配置页



3. 为该 bucket 配置跨区域复制任务

4. 当该 bucket 下产生备份文件时, 文件会被自动同步到配置好的异地 bucket 中

数据归档方案

背景

在数据归档领域,传统的磁带库或是蓝光盘库介质在过往一直是首选,这些磁带或者光盘一旦存储了数据,就意味着数据进入到数据中心某个不起眼的角落中,如没必要将通常会进入到“沉睡”阶段,有些数据甚至几十年都不再被读取使用。如今数字经济的背景下,冷数据价值的挖掘受到了越来越多的关注,灵活的数据检索,准实时的数据取回能力也成为了新时代数据归档场景的核心需求,同时用户也在 TCO 成本优化的角度开始关注数据分层带来的价值。

本文介绍了如何灵活地使用 US3 的不同存储类型存储数据,以达到降低数据长期存储成本的目的。

US3 归档类型

UCloud 归档类型实现了 0.024元/GB/月 的存储单价,比市场同类型产品价格下降 30%,同时可以保障数据11个9以上的可靠性,数据可用性达到99.9%以上。归档存储类型在使用数据前需要进行解冻数据操作,解冻数据会解冻类型收取解冻费用,数据解冻操作将根据文件大小以及解冻类型需要不同的等待时间,请以实际解冻时间为准。使用 RESTORE 进行解冻操作,解冻完成后,数据取回不再收取额外费用,仅收取正常流出流量费用。解冻操作完成后数据保留 3 天,已解冻的数据取回不收取取回费用,超过 3 天则无法下载,需要重新进行解冻操作。如您需要详细了解 US3 归档类型的定价标准,建议您参考 [产品价格](#) 以及 [计费案例](#) 中的案例三。

使用 US3 归档类型存储冷数据

如您需要使用 US3 归档类型存储冷数据,US3 支持您在上传时指定存储类型的方式,将数据直接存储至归档类型,或采用文件存储类型转换方式进行手动的存储类型转换。US3 同时支持

您对存储在 US3 标准类型或低频类型中的数据配置生命周期策略,以达到定期自动降冷的效果。

直接上传文件至归档存储

1. 通过 US3 网页控制台上传文件

选择指定存储空间,进入文件管理页后,在文件管理页点击“上传文件”,弹出上传文件窗口。



您可以在添加所需上传的文件后,在“选择存储类型”选项右侧,勾选归档存储,执行上传操作,文件即可被上传至归档存储。

2. 通过上传类 API 进行文件上传

US3 提供的上传类 API 中,在请求头(Request Headers)中以 X-Ufile-Storage-Class 字段来进行存储类型判断,默认请求头中该字段为空时,文件会被默认上传至标准类型的存储空间中,按标准类型计费。如您需要将文件直接上传至归档存储,需指定 X-Ufile-Storage-Class 的值为 ARCHIVE。

如您需要详细了解 US3 上传类 API,您可以参考 [上传文件](#)、[表单上传](#)、[初始化分片](#) API 文档。

手动修改存储类型为归档存储

1. 通过 US3 网页控制台修改存储类型

选择指定存储空间,进入文件管理页后,在单个文件右侧操作下拉列表中点击“修改存储类型”。



您可以在弹出窗口中选择“归档存储”,即可手动修改存储类型为归档存储。

2. 通过文件存储类型转换 API 修改存储类型

US3 提供 [文件存储类型转换 API](#),您可以使用该 API 转换文件的存储类型。

设置生命周期策略将文件转换为归档存储

您可以通过开通存储空间(Bucket)生命周期功能,实现存储空间内所有文件或特定前缀文件定期自动转换为归档存储,从而节省存储费用。

如您需要详细了解生命周期功能的设置方式,请参考 [生命周期](#)

归档存储适用场景

多媒体归档场景

在线直播、视频监控目前已逐渐普及,在这些场景中,一个 1080P 的高清摄像头存储一天就需要 45G 的容量,一个视频网站每天产生的数据量可达 TB 以上;UCloud 某广电客户此前采用蓝光存储的方式,预计到 2024 年,其存储数据量将达 16.4PB,大概需要 8 个蓝光盘柜,占用机房一整排机柜空间,对客户来说,是一笔巨大的成本支出。

UCloud 新一代归档存储可提供不亚于标准存储的写入带宽,实现分钟级数据异步取回,在线回看;并采用纠删码冗余策略来保障数据安全可靠。结合 US3 不同存储类型间生命周期转换功能,用户还可快速实现数据由热至温再到冷的存储类型转换,完成自动化的数据生命周期管理。

历史数据合规存储

面对日益上涨的企业数据库备份场景,US3 提供的 [数据库备份方案](#) 能够有效帮助用户缩减备份流程。针对需要定时清理备份、缩减备份成本的用户,US3 支持数据生命周期管理功能,可以实现自动化的数据定期清理、定期转入归档存储。针对需要更高安全级别的用户,US3 可支持跨区域复制功能,帮助用户完成数据的异地备份。

在电商平台的日志归档场景中,US3 还提供了 [ElasticSearch 接入和数据库备份功能](#),当数据量增大后,对历史数据进行统一归档至归档存储的方式,以降低存储成本。

大数据、AI分析数据归档

根据研究机构提供的数据,2020 年生物经济规模已达 15 亿美元。拿肿瘤疾病的基因测序为例,单个患者的 DNA 样本数据能达到 560GB,如果按照每年 1800 多万的癌症病例来计算,使用基因分析技术后每年就会产生 10PB 的肿瘤基因样本数据。而中国一家三甲医院每年的影像数据就有 20TB 左右,全国 3 万多家医院的数据量,也是非常庞大的数据。US3 归档存储

能够为大量的生物信息、IoT 实时分析数据等场景提供长期归档存储,为未来的医疗研究、工业智能储备数据资料。

新基建背景下,随着新技术与新场景不断融合,在线教育、云游戏、自动驾驶、智慧社区、智能制造等行业,都将产生越来越多的数据。US3 归档存储面向未来的数据分层存储场景,采用全新的自研存储架构,降低用户的硬件成本和运营成本,让用户以更低的价格、更可靠的方式,存储数据资产,为未来挖掘数据生产要素的价值积累财富。

使用STS临时访问凭证访问US3

您可以通过STS服务给其他用户颁发一个临时访问凭证。该用户可使用临时访问凭证在规定时间内访问您的US3资源。临时访问凭证无需透露您的长期密钥,使您的US3资源访问更加安全。

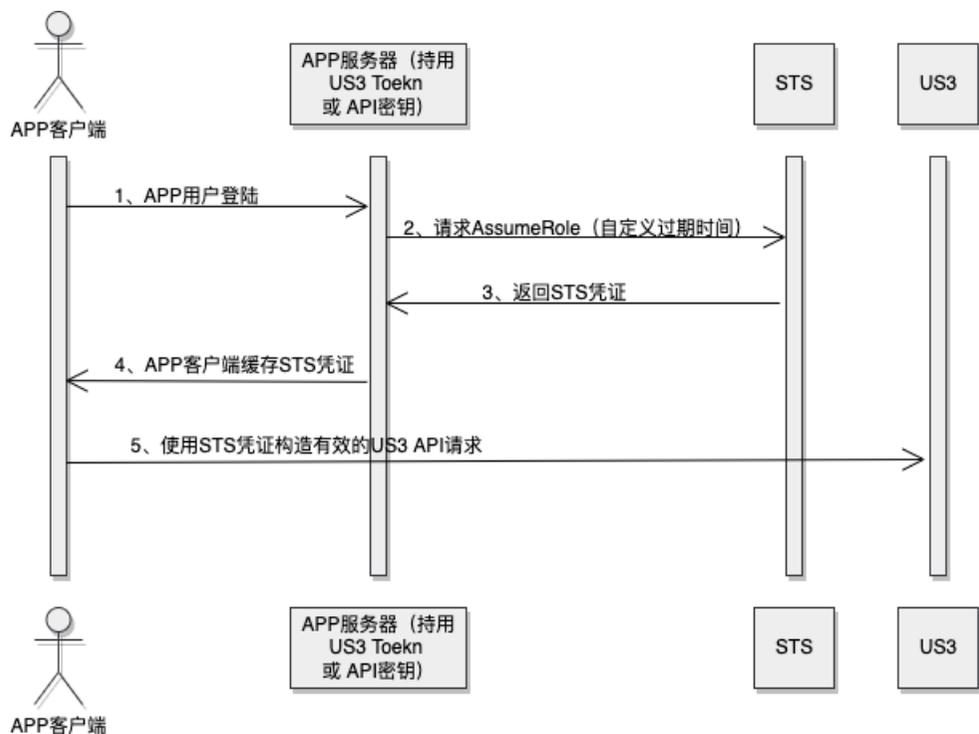
前提条件

已确保当前账号为被授予 STSFullAccess 权限的子账户或者主账户。关于为子用户/角色授权的具体步骤,请参见为子账户用户授权。

适用场景

假设您是一个移动App开发者,希望使用US3服务来保存App的终端用户数据,并且要保证每个App用户之间的数据隔离。此时,您可以使用STS授权用户直接访问US3。

使用STS授权用户直接访问US3的流程如下:



1. APP用户登录。APP用户和云账号无关，它是APP的终端用户，APP服务器支持APP用户登录。对于每个有效的APP用户来说，需要APP服务器能定义出每个APP用户的最小访问权限。
2. APP服务器请求STS服务获取一个安全令牌 (SecurityToken)。在调用STS之前，APP服务器需要确定App用户的最小访问权限 (用RAM Policy来自定义授权策略) 以及凭证的过期时间。然后通过扮演角色 (AssumeRole) 来获取一个代表角色身份的安全令牌 (SecurityToken)。
3. STS返回给APP服务器一个临时访问凭证，包括一个安全令牌 (SecurityToken)、临时访问密钥 (AccessKeyId和AccessKeySecret) 以及过期时间。
4. APP服务器将临时访问凭证返回给App客户端，APP客户端可以缓存这个凭证。当凭证失效时，APP客户端需要向APP服务器申请新的临时访问凭证。例如，临时访问凭证有效期为1小时，那么APP客户端可以每30分钟向APP服务器请求更新临时访问凭证。
5. APP客户端使用本地缓存的临时访问凭证去请求US3 API。US3收到访问请求后，会通过STS服务来验证访问凭证，正确响应用户请求。

步骤一:创建子用户

1. 登录控制台。

2. 在左侧导航栏,选择用户管理。
3. 单击邀请用户,并选择 "API访问"。
4. 输入邮箱/用户名称。
5. 然后单击确定。
6. 创建完成完成之后需要给改用户添加 STSFullAccess 权限

添加权限

应用范围
全局级范围为账号全局服务,不区分项目,不支持分项目授权;项目级生效范围为指定项目

应用策略

可选策略

<input type="checkbox"/>	策略名称	描述
<input type="checkbox"/>	STSCreateOnlyAccess	安全令牌服务 (STS) 新增...
<input checked="" type="checkbox"/>	STSFullAccess	安全令牌服务 (STS) 管理...

已选策略*

① 请从左侧列表选择

步骤二：创建用于获取临时访问凭证的角色

1. 在左侧导航栏,选择角色管理。

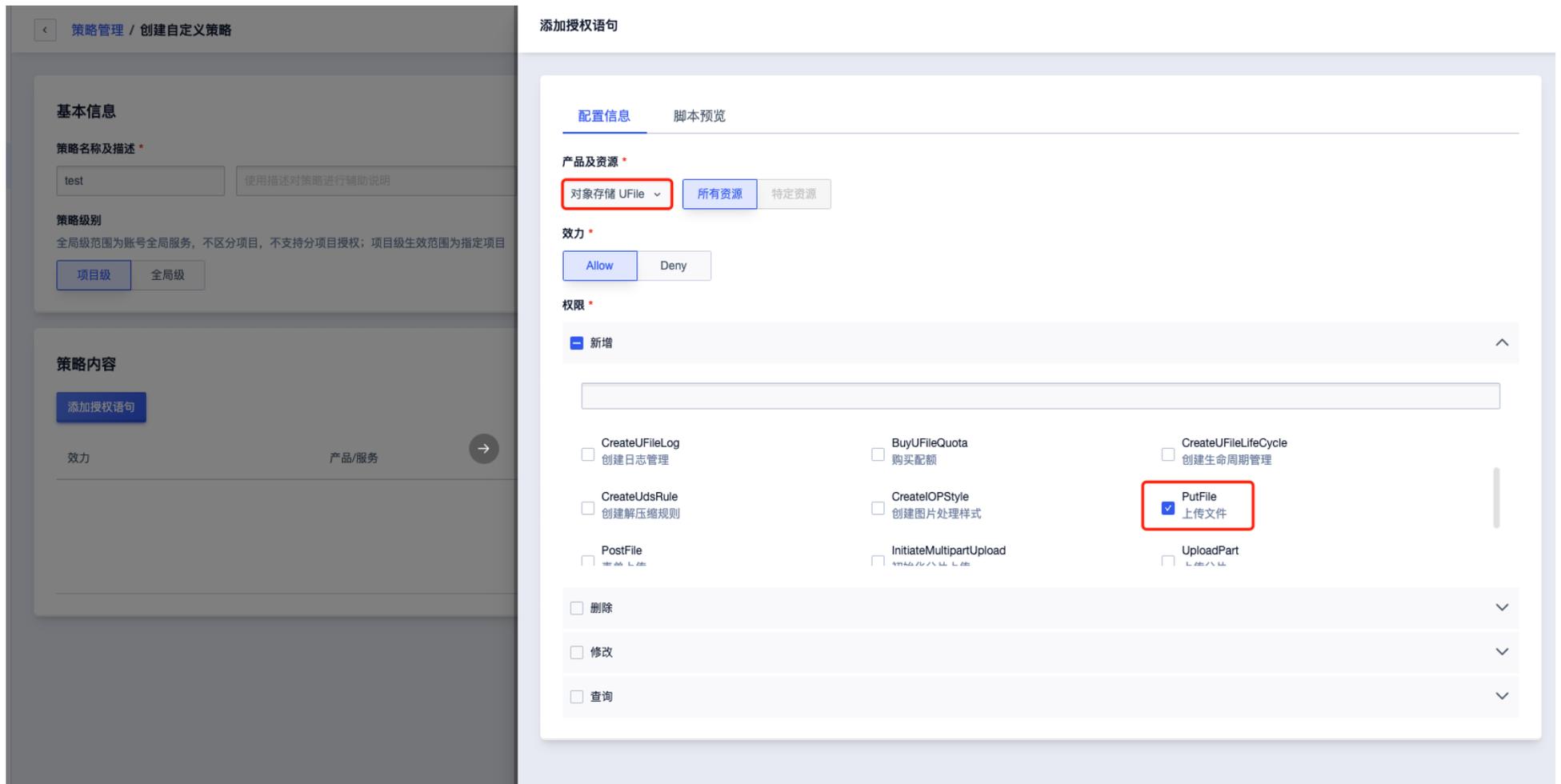
2. 单击**创建角色**。
3. 在创建角色对话框,角色名称填写为RamOssTest。
4. 单击**完成**。角色创建完成后,单击**关闭**。



角色名称	描述	URN	创建时间	操作
RamOssTest		ucs:iam::55936045:role/RamOssTest	2023-07-10 17:23:24	详情 删除

步骤三：为角色授予上传文件的权限

1. 创建上传文件的自定义权限策略。
 1. 在左侧导航栏,选择**权限管理>权限策略**。
 2. 在**权限策略**页面,单击**创建权限策略**。
 3. 在**创建权限策略**页面,单击**授权语句**并选择**对象存储**,然后在Allow的新增类权限中选择PutFile的权限。具体配置示例如下。



4. 点击确定。

2. 为RAM角色RamOssTest授予自定义权限策略。

1. 在左侧导航栏,选择角色管理。
2. 在角色页面,找到目标RAM角色RamOssTest。
3. 单击RAM角色RamOssTest页面的添加权限。
4. 在添加权限页面下的自定义策略页签,选择已创建的自定义权限策略test。

5. 单击确定。

步骤四:获取临时访问凭证

- 使用 REST API

使用子账户的API密钥调用****AssumeRole(获取扮演角色的临时身份凭证),****请确保已为调用者授予STS的管理权限,当前API访问地址为https://api.ucloud.cn

请求参数:

名称	类型	必填	描述	示例值
Action	string	是	动作:AssumeRole	AssumeRole
DurationSeconds	int	否	有效期.单位:秒。	43200
PublicKey	string	是	API公钥。	*****
RoleSessionName	string	是	角色会话名称。	alice
RoleUrn	string	是	要扮演的RAM角色ARN。ucs:iam::55936045:role/RamOssTest	
Signature	string	是	签名.签名方法	*****

返回参数:

名称	类型	描述	示例值
Message	string	状态消息。	success
SecurityToken	string	安全令牌。	AAHoTYM*****

AccessKeyId	string	访问密钥ID。	STS.3ho71eJpBoyQvgD****
AccessKeySecret	string	访问密钥。	MwSLKHDpz72KsKBQ****
Expiration	string	Token到期失效时间。	2023-07-11T20:15:58Z

示例：

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings Cookies

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL **JSON** Beautify

```

1  {
2  ... "Action": "AssumeRole",
3  ... "DurationSeconds": 43200,
4  ... "PublicKey": "F6bpUq4IXL03KHVp...iRG6ZF",
5  ... "RoleSessionName": "xiaobo",
6  ... "RoleArn": "ucs:iam::55936045:role/xiaobo",
7  ... "Signature": "1abfe30eec5b8168de...J5a"
8  }

```

Body Cookies Headers (17) Test Results Status: 200 OK Time: 264 ms Size: 1.16 KB Save as Example

Pretty Raw Preview Visualize **JSON**

```

1  {
2  "RetCode": 0,
3  "Message": "success",
4  "Credentials": {
5  "SecurityToken":
6  "AAE2D_AC3cHmRd3iDiykxsZA...X172wGANHxxIUT1bKV7kuRVEGpQZQUqFvqqX1NiZb_UF8RH1-nH93h-ud-KHZUeJJqhDYCfZD20ayvfqR_Pwt-khtVIAg20xwu0raZS-mN11E3nh8
7  ON5ktfXm02s9hItoXeSVug0jz...kSp1-uloen67GJpM2Q3bH045U4LXzL9tvNqLjCsU8Yp_80tioEIs7GDPEbaBj4XT10ZHHiMwqrQ",
8  "AccessKeyId": "STS.3nka...1s",
9  "AccessKeySecret": "...Dkq",
10 "Expiration": "2023-07-10T19:19:37Z"
11 }

```

步骤五:使用临时访问凭证上传文件至US3

以Go SDK 为例,将本地/data/路径下的exampletest.txt文件上传至存储空间examplebucket下的src目录的示例代码如下:

```
func PutFile() {
    config := &ufsdk.Config{
        PublicKey: "STS.*****",
        PrivateKey: "*****",
        FileHost: "cn-bj.ufileos.com",
        BucketName: "examplebucket",
    }
    head := http.Header{}
    head.Set("SecurityToken", "AAFtszppallV8y23F0T2tCpAJmtEZ6CpljO4bCmCCU*****")
    req, err := ufsdk.NewFileRequestWithHeader(config, head, nil)
    if err != nil {
        panic(err.Error())
    }
    err = req.PutFile("/data/exampletest.txt", "src/exampletest.txt", "")
    if err != nil {
        panic(err.Error())
    }
}
```

归档转标准/低频存储类型

背景

- 提供方法要将归档类型文件转换为标准类型或者低频类型文件。

方案

- 使用激活加copy的方式创建出指定存储类型的新文件。
- 具体流程:
 1. 存储空间上传归档文件file1
 2. 对file1解冻
 3. 使用copy操作在同一存储空间中创建出指定文件类型的新文件file2
 4. 根据实际场景需要删除原文件file1

参考代码

```
import (  
  "net/http"  
  ufsdk "github.com/ufilesdk-dev/ufile-gosdk"  
  "github.com/ufilesdk-dev/ufile-gosdk/example/helper"
```

```
"log"
"os"
"time"
)

const (
    FakeFilePath = "./FakeFile.txt"
    FileSize = 5 * 1024 * 1024 * 1024
    ConfigFile = "config.json"
    srcFileKey = "test_archive_5g.txt"
    newFileKey = "test_standard_5g.txt"
)

func MPutCopy(req *ufsdk.UFileRequest, dstkeyName, srcBucketName, srcKeyName string) error {
    state, err := req.InitiateMultipartUpload(dstkeyName, "") //初始化分片上传
    if err != nil {
        return err
    }
    parts, err := ufsdk.SplitFileByPartSize(FileSize, int64(state.BlkSize))
    if err != nil {
        log.Fatalf("SplitFileByPartSize Err:%v", err)
    }

    for _, part := range parts {
        // 拷贝分片
```

```
err = req.UploadPartCopy(state, part.Number, srcBucketName, srcKeyName, part.Offset, part.Size)
if err != nil {
    req.AbortMultipartUpload(state)
    return err
}

return req.FinishMultipartUpload(state) // 完成分片上传
}

func main() {
    log.SetFlags(log.Lshortfile)
    if _, err := os.Stat(FakeFilePath); os.IsNotExist(err) {
        helper.GenerateFakefile(FakeFilePath, FileSize)
    }
    config, err := ufsdk.LoadConfig(ConfigFile)
    if err != nil {
        panic(err.Error())
    }

    header := make(http.Header)
    req, err := ufsdk.NewFileRequestWithHeader(config, header, nil)
    if err != nil {
        panic(err.Error())
    }
}
```

```
//1、上传归档存储类型文件
//存储类型,目前支持的类型分别是标准:"STANDARD"、低频:"IA"、冷存:"ARCHIVE"
header.Set("X-Ufile-Storage-Class", "ARCHIVE")
log.Println("正在上传归档存储类型文件。。。。")
err = req.AsyncMPut(FakeFilePath, srcFileKey, "")
if err != nil {
log.Fatalln("文件上传失败,失败原因:", err.Error())
}
log.Println("文件上传成功,文件名:", srcFileKey)

//2、解冻归档存储类型文件
log.Println("正在解冻归档存储类型文件。。。。")
err = req.Restore(srcFileKey)
if err != nil {
log.Fatalln("文件解冻失败,失败原因:", err.Error())
}
log.Println("文件解冻成功,文件名:", srcFileKey)
//解冻需要几秒的生效时间
time.Sleep(10 * time.Second)

//3、copy创建新文件并指定存储类型为标准类型
log.Println("正在创建标准类型的新文件。。。。")
header.Set("X-Ufile-Storage-Class", "STANDARD")

// 大于100M的文件使用UploadPartCopy
```

```
if FileSize > 100 * 1024 * 1024 {
    err = MPutCopy(req, newFileKey, req.BucketName, srcFileKey)
} else { // 小文件copy
    err = req.Copy(newFileKey, req.BucketName, srcFileKey)
}
if err != nil {
    log.Fatalln("文件转换存储类型失败, 失败原因:", err.Error())
}
log.Println("创建标准类型的新文件成功, 新文件名:", newFileKey)

//4、检查新文件存储类型
log.Println("正在获取文件列表。。。")
list, err := req.PrefixFileList("test_", "", 10)
if err != nil {
    log.Fatalln("获取文件列表失败, 错误信息为:", err.Error())
}
log.Printf("获取文件列表返回的信息是:\n%s\n", list)

//5、删除原文件
log.Println("正在删除原文件。。。")
//err = req.DeleteFile(srcFileKey)
if err != nil {
    log.Fatalln("文件删除失败。")
}
log.Println("原文件删除成功。")
```

```
}
```

测试用例

- 北京100k文件

```
user@userdeMacBook-Air example % go run demo_restore.go
demo_restore.go:61: 正在上传归档存储类型文件。。。。
demo_restore.go:66: 文件上传成功, 文件名: test_archive_100m.txt
demo_restore.go:69: 正在解冻归档存储类型文件。。。。
demo_restore.go:74: 文件解冻成功, 文件名: test_archive_100m.txt
demo_restore.go:77: 正在创建标准类型的新文件。。。。
demo_restore.go:89: 创建标准类型的新文件成功, 新文件名: test_standard_100m.txt
demo_restore.go:92: 正在获取文件列表。。。。
demo_restore.go:97: 获取文件列表返回的信息是:
{
  "BucketName": "test-restore-bj",
  "BucketId": "ufile-18bpxilztrxz",
  "DataSet": [
    {
      "BucketName": "test-restore-bj",
      "FileName": "test_archive_100m.txt",
      "Hash": "GQAAADW944yct-b21efQT1YChk8IANuE",
      "MimeType": "text/plain; charset=utf-8",
      "Size": 104857600,
```

```
    "CreateTime": 1739417311,  
    "ModifyTime": 1739417358,  
    "StorageClass": "ARCHIVE",  
    "RestoreStatus": "Restored"  
  },  
  {  
    "BucketName": "test-restore-bj",  
    "FileName": "test_standard_100m.txt",  
    "Hash": "GQAAADW944yct-b21efQTlYChk8IANuE",  
    "MimeType": "text/plain; charset=utf-8",  
    "Size": 104857600,  
    "CreateTime": 1739417369,  
    "ModifyTime": 1739417369,  
    "StorageClass": "STANDARD"  
  }  
]  
}
```

- 北京5g文件

常见问题

对象存储空间和 Key 是什么?

对象存储空间(简称存储空间)是文件的组织管理单位,一个文件必然位于某个空间中。空间名称全局唯一,且无法进行修改。

文件名是对应文件的名称,在存储空间中全局唯一,每个文件名在存储空间均标识了一个文件,写入文件时,用户可以自定义文件名。上传同样文件名的文件,会导致原文件名文件被覆盖。

公共空间和私有空间有何区别?

公共空间指任何人都可以直接通过URL访问该空间里的文件,不需要授权签名。

私有空间需要根据API公私钥生成正确的签名才能访问文件。

如何查看及管理上传的文件?

用户可以通过控制台的文件管理页查看已上传的文件,同时也可以使用文件管理工具或API查看已经上传的文件。

是否支持目录、文件列表?

对象存储空间没有目录的概念,所以不能按照目录列出文件列表。

但上传文件时,Key 依然沿用目录形式,便于特定的用户使用场景需要。

例如:demobucket.ufile.ucloud.cn/test/a.jpg 这里的 key=test/a.jpg 。

如何使用对象存储空间提供的两个域名?

每个存储空间默认提供一个存储空间域名与一个 CDN 加速域名。

文件上传操作必须将请求发往存储空间域名。

文件下载操作可以通过访问存储空间域名或 CDN 加速域名进行。文件下载时建议使用 CDN 加速域名进行下载,以获得更好的下载体验。

我可以存储多少数据? 对象存储空间是否有容量上限?

您可以存储的总数据容量和对象个数不受限制,您可以按需使用。

文件大小有什么限制?

单个文件大小上限为 5TB。

对象存储空间如何通过内网访问?

1. 空间管理的 API,内网访问时域名同公网,使用api.ucloud.cn。
2. 文件管理的 API,需要使用内网专用域名<bucket_name>.ufile.cn-north-02.ucloud.cn。

例如,bucket 名称为 demobucket,则其内网域名为demobucket.ufile.cn-north-02.ucloud.cn。

3. 文件管理的 API 列表如下:PutFile, PostFile, UploadHit, GetFile, DeleteFile, InitiateMultipartUpload, UploadPart, FinishMultipartUpload, AbortMultipartUpload。
4. 命令行工具,通过内网访问,需要将配置文件中的 proxy_host 改成 "proxy_host":'www.ufile.cn-north-02.ucloud.cn'。
5. SDK(以 phpSDK 为例)通过内网访问,需要将配置文件中 \$UCLLOUD_PROXY_SUFFIX 改成 \$UCLLOUD_PROXY_SUFFIX = 'ufile.cn-north-02.ucloud.cn'(其他 SDK 的配置文件通常改 proxy_suffix)。

如何删除大量对象?

您可以通过设置 生命周期 的方式,对存储桶内文件进行删除操作。

US3 域名被第三方平台告知有安全风险怎么处理?

第三方平台的安全检测是基于泛域名做检测及封禁处理,而 US3 不同客户的域名使用同一个泛域名,因此只要有一个客户有违规内容,整个 US3 默认域名都会被安全软件封禁。

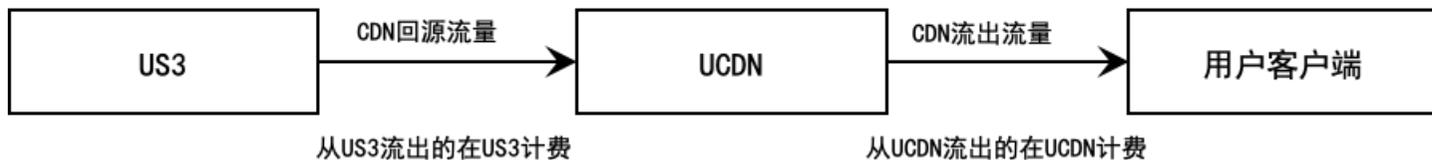
这个问题我们和第三方平台沟通过多次,对方以按照逐个域名做封禁成本太大为由拒绝。因此遇到封禁情况的客户,建议通过使用自定义域名解决。配置方法参考 [域名管理](#)。

有跨域需求时如何申请?

如需在 US3 配置跨域,需要指派工单至技术支持,工单内注明:bucket 名称、US3 域名、Origin 地址和要跨域的 http method。

CDN 回源 US3 的流量如何计费?

CDN 回源,流量从 US3 流向 UCDN,这部分流量 UCDN 不计费,由 US3 计费,请参考下图:



计费价格详见:计量计费

镜像回源支持的空间类型

镜像回源没有签名的过程,目前只支持公开空间。

我的账号为何无法进行存储空间或文件操作?

如果您的账号在控制台操作中提示291:[xxx]该账户没有执行对应 Action 和产品类型的权限,表示您当前使用的子账号未被授权进行相关 US3 对象存储操作权限,请联系主账号管理员开通相关权限。

如何存储额外的文件元数据信息?

US3 API 支持用户存储不超过 8KB 的自定义元数据, 在用户调用 API 进行文件上传请求时, 可在请求头中增加 X-Ufile-Meta-* 字段, 如增加文件 MD5 信息, 可增加请求头 X-Ufile-Meta-MD5 , 在执行 Head、Get 请求时, 可从 Response Header 中获取 X-Ufile-Meta-MD5:[*] 的内容。更多内容可参考对象存储API文档

注意: **Header** 请求中 **X-Ufile-Meta-xxx** 限制 **xxx** 只能包含英文字母、数字和连接符 (短横线 '-')

子帐号授权后进入文件管理页面提示: 非法授权

1. 添加子账号时, 如果没有选择 "API访问" 那么这个子账号默认就是没有密钥的

邀请子用户 ⊗

访问方式 *

控制台访问 API访问

控制台访问即子用户通过邮箱和密码登录UCloud控制台；
API访问即为子用户分配一对公私钥，支持通过API、SDK、CLI等开发工具访问。

邮箱 *

用于登录控制台、接收邮件等

用户名 *

长度限制0-64个字符, 只能包含英文、数字、下划线 (_)

昵称

长度限制0-64个字符, 只能包含中英文、数字、下划线

2. 有两个修改入口

- 有管理权限的账号:到用户管理页面,进入具体某个用户的详情页,然后创建密钥

访问控制

- 用户管理
- 用户组管理
- 策略管理
- 项目管理
- 授信管理

< 用户管理 / test

控制台访问

访问能力

登录保护

定期更改密码

所属用户组

加入到用户组

输入关键字

用户组名称	备注	加入时间
暂时数据为空		

< 1 > 10条/页 /1

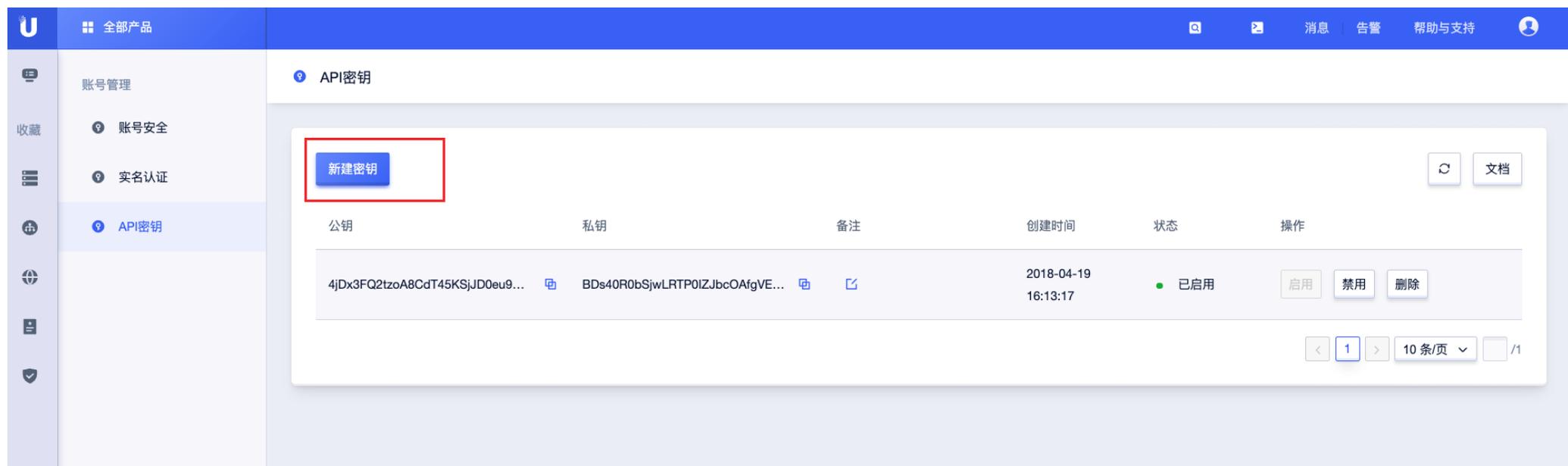
API密钥

新建密钥

公钥	私钥	备注	创建时间	状态	操作
暂无API密钥, 请创建					

< 1 > 10条/页 /1

- 为当前登录的账号创建密钥:到当前账号的账号管理页面,点击API密钥菜单,进行创建密钥



第1种适合管理人员操作,第2种适合子账号给自己创建密钥

为什么使用Firefox浏览器时,无法批量解冻以及下载已解冻的归档文件等解冻相关操作?

使用Firefox浏览器无法批量解冻、批量或单文件下载已解冻的归档文件、以及解冻状态展示的原因是Firefox浏览器存在兼容性问题。

解决方法:

- 更换浏览器(如Chrome浏览器),确认该文件的解冻状态:
 1. 如“未解冻”,即可以操作解冻按钮,用户可以重试解冻后,再对已解冻的文件进行相关操作;
 2. 如“已解冻”,即控制台直接展示该状态,用户无需重复操作解冻,可以直接访问该已解冻文件。