

基因容器

# 最佳实践

文档版本 01  
发布日期 2025-11-26



版权所有 © 华为云计算技术有限公司 2025。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

## 商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

## 注意

您购买的产品、服务或特性等应受华为云计算技术有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为云计算技术有限公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

---

## 目录

---

1 使用命令行工具执行测序.....	1
2 使用第三方数据执行测序.....	7
3 基于 GCS 构建基因测序平台.....	12

# 1 使用命令行工具执行测序

本示例通过指导您创建CCI命名空间、绑定文件存储卷，使用命令行工具查询和设置环境、查询和设置SFS存储卷、创建execution执行测序及查看计算结果，帮助您学习如何使用命令行工具在云容器实例环境中执行测序。

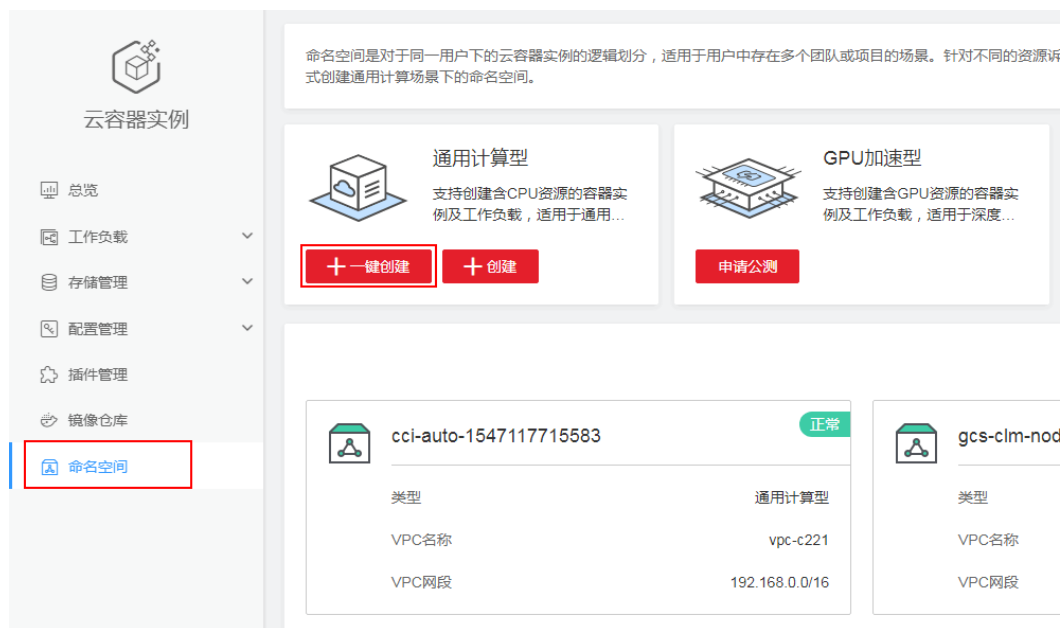
## 操作步骤

使用命令行工具执行测序的操作步骤如下：

### 步骤1 创建CCI命名空间。

登录[云容器实例控制台](#)，左侧导航栏中选择“命名空间”，单击“一键创建”创建命名空间。创建成功的CCI环境命名空间如图1-1，本步骤创建的命名空间将为后续测序提供计算资源。

图 1-1 创建命名空间



### 步骤2 创建文件存储卷，并绑定到CCI命名空间下。

1. 登录**弹性文件服务控制台**，在页面右上角单击“创建文件系统”。进入文件存储卷创建界面，根据界面提示配置参数，如**图1-2**。配置完成后，单击“立即创建”。

### 须知

文件存储卷的区域应与CCI命名空间的区域一致，文件存储卷的虚拟私有云应与创建CCI命名空间时使用的VPC一致。

图 1-2 创建文件存储卷

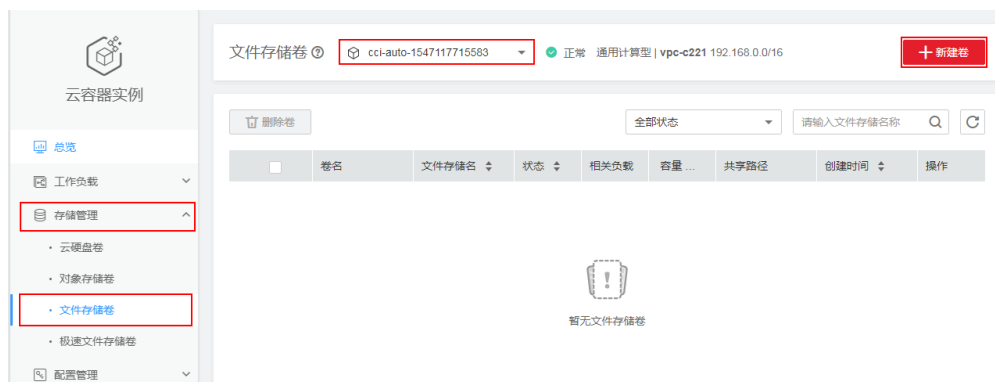
创建成功的SFS存储如**图1-3**，本步骤创建的文件存储将作为后续测序原始数据存放地址以及中间数据和结果数据的存放地址。

图 1-3 SFS 存储创建成功

名称	可用区	状态	类型	共享协议	可用容量(GB)
sfs-176a	可用区2	可用	SFS容量型	NFS	100.00

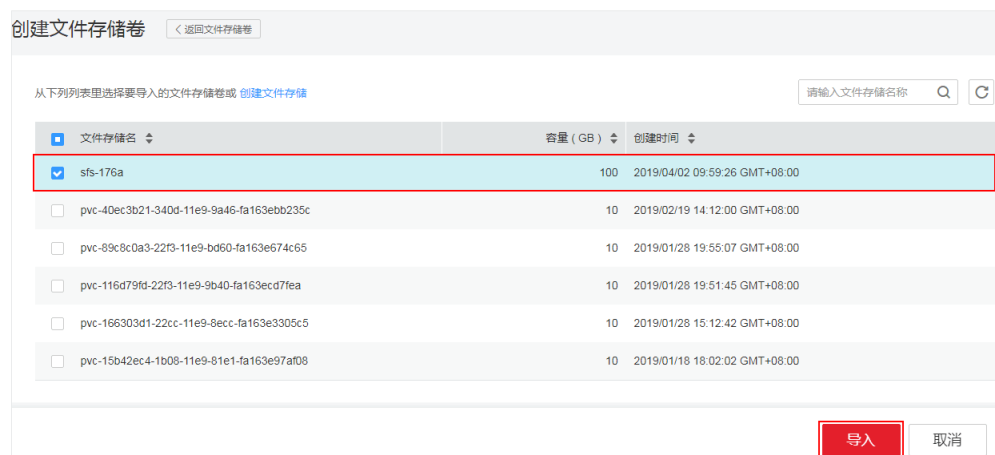
2. 登录**云容器实例控制台**，选择左侧导航栏的“存储管理>文件存储卷”，在右侧页面单击选择步骤一创建的命名空间，单击“新建卷”。

图 1-4 新建卷



3. 选择刚创建的SFS存储卷，并单击“导入”，完成SFS存储卷和命名空间的绑定。

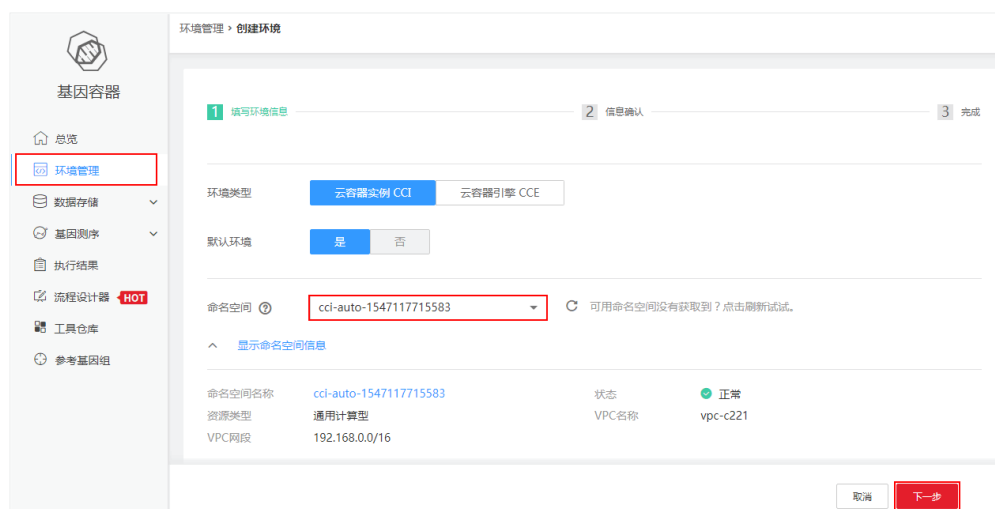
图 1-5 导入 SFS 存储卷



**步骤3** 创建云容器实例环境。

1. 登录[基因容器控制台](#)，选择左侧导航栏的“环境管理”，在右侧页面单击“创建环境”。
2. 选择“云容器实例 CCI”类型，并选定步骤1创建的命名空间。

图 1-6 创建 CCI 环境



- 单击“下一步”，确认环境信息配置后单击“提交”，完成云容器实例环境的创建。本步骤创建的环境封装了CCI命名空间，将为后续测序提供计算资源。

**步骤4** 制作镜像，并上传镜像至华为云私有镜像仓库，添加GCS私有工具。

本示例使用nginx镜像，对应工具为nginx:latest。

**步骤5** 编写流程文件Workflow，具体方法请参见[流程语法参考](#)。

本示例将使用gcs-demo.yaml文件，内容如下：

```
version: genecontainer_0_1
workflow:
  gcs-demo:
    tool: 'nginx:latest'
    type: GCSJob
    resources:
      memory: 2G
      cpu: 1C
    commands:
      - echo "gcs-demo" > /sfs/pkg/gcs-demo.yaml
volumes:
  gensfs:
    mount_path: '/sfs'
    mount_from:
      pvc: '${GCS_SFS_PVC}'
```

测序完成后“gcs-demo”将写入存储目录/pkg下的gcs-demo.yaml文件中。

**步骤6** 将步骤2创建的SFS存储卷挂载到本地，方便上传和查看数据。

本示例将创建的文件存储卷挂载至/root/gcs目录下。

**步骤7** 创建目录，将数据放置到SFS存储中。

根据步骤5编写的workflow，在SFS存储下创建pkg目录：

```
[root@ecs-6454 gcs]# ls -lt
total 17284
-rw----- 1 root root 12667 Mar 6 16:01 gcs-cli.log
-rw----- 1 root root 278 Mar 6 15:58 gcs-demo.yaml
drwx----- 2 root root 4096 Mar 6 15:32 pkg
-rw----- 1 root root 4271407 Feb 28 14:13 cloud-cli-linux-gcs-3.0.42.zip
-rwxrwxr-x 1 root root 13400278 Feb 28 10:41 gcs
[root@ecs-6454 gcs]#
```

**步骤8** 下载并配置GCS命令行工具，方法请参见[安装命令行工具](#)。

**步骤9** 使用命令行工具查询环境。

执行./gcs get environment 查看当前账号下可用的环境列表：

```
[root@ecs-6454 gcs]# ./gcs get environment
{
  "environments": [
    {
      "guid": "5b2fe838-3d68-0ee2-b029-cfb4c799fca1",
      "project_id": "0ca",
      "domain_id": "d5d4",
      "name": "gcs-env-cci-auto-1547117715583",
      "description": "Gene container environment from cci",
      "status": "Running",
      "create_at": "2019-03-06T07:01:28Z",
      "update_at": "2019-03-06T07:01:28Z",
      "cluster_id": "a061d",
      "cluster_name": "cci-auto-1547117715583",
      "cpu": 0,
      "memory": 0,
      "source": "cci",
      "stack_id": "",
      "is_gene_cluster": false,
      "bucket_name": "",
      "default_env": true,
      "order_id": ""
    }
  ],
  "total": 1
}
```

#### 步骤10 设置环境。

执行./gcs set env设置后续测序使用的环境：

```
./gcs set env gcs-env-cci-auto-1547117715583
```

```
[root@ecs-6454 gcs]# ./gcs set env gcs-env-cci-auto-1547117715583
current environment is gcs-env-cci-auto-1547117715583
please set sfs-pvc-name for environment gcs-env-cci-auto-1547117715583
[root@ecs-6454 gcs]#
```

#### 步骤11 查询当前环境下的文件存储卷。

执行./gcs get sfs查询当前环境填的SFS存储卷列表：

```
./gcs get sfs
```

```
please set sfs-pvc-name for environment gcs-env-cci-auto-1547117715583
[root@ecs-6454 gcs]# ./gcs get sfs
current environment is gcs-env-cci-auto-1547117715583
[
  "pvc-sfs-jswuo09d-j6cf"
]
[root@ecs-6454 gcs]#
```

#### 步骤12 设置SFS。

执行./gcs set sfs设置后续使用的SFS：

```
./gcs set sfs pvc-sfs-jswuo09d-j6cf
```

```
[root@ecs-6454 gcs]# ./gcs set sfs pvc-sfs-jswuo09d-j6cf
current environment is gcs-env-cci-auto-1547117715583
set environment gcs-env-cci-auto-1547117715583 default sfs-pvc-name pvc-sfs-jswuo09d-j6cf succeed
[root@ecs-6454 gcs]#
```

**步骤13** 创建Workflow。

执行./gcs create workflow 创建workflow:

```
./gcs create workflow gcs-demo.yaml --name pvc-demo-workflow
```

```
[root@ecs-6454 gcs]# ./gcs create workflow gcs-demo.yaml --name gcs-demo-workflow
create workflow succeed !
create workflow succeed! workflow id : 48b51171-d836-fd8e-a23e-7b985576861d
[root@ecs-6454 gcs]#
```

**步骤14** 创建execution，执行测序。

执行./gcs create execution WORKFLOWID使用步骤13创建的workflow返回的workflowID:

```
./gcs create execution 48b51171-d836-fd8e-a23e-7b985576861d --name gcs-demo-execution
```

```
[root@ecs-6454 gcs]# ./gcs create execution 48b51171-d836-fd8e-a23e-7b985576861d --name gcs-demo-execution
current environment is gcs-env-cci-auto-1547117715583
create execution succeed
execution [0] : gcs-demo-execution succeed
{
  "id": "f1fd6525-3fe6-11e9-a947-0255ac109f1a"
}
batch create executions finish
```

**步骤15** 查看创建的execution。

执行./gcs get execution executionID 使用步骤14创建execution返回的ID:

```
./gcs get execution f1fd6525-3fe6-11e9-a947-0255ac109f1a
```

```
batch create executions finish
[root@ecs-6454 gcs]# ./gcs get execution f1fd6525-3fe6-11e9-a947-0255ac109f1a
job_name      namespace      status      submit_time      duration
gcs-demo-0    cci-auto-1547117715583  Succeeded    2019-03-06 16:07:55  0:00:30
[root@ecs-6454 gcs]#
```

**步骤16** 查看存储目录下是否有计算结果。

进入/root/gcs/pkg目录下，发现“gcs-demo”文字已经写入对应区域:

```
[root@ecs-6454 pkg]# ls -lt
total 4
-rw-r--r-- 1 root root 9 Mar  6 16:08 gcs-demo.yaml
[root@ecs-6454 pkg]# cat gcs-demo.yaml
gcs-demo
[root@ecs-6454 pkg]#
```

----结束

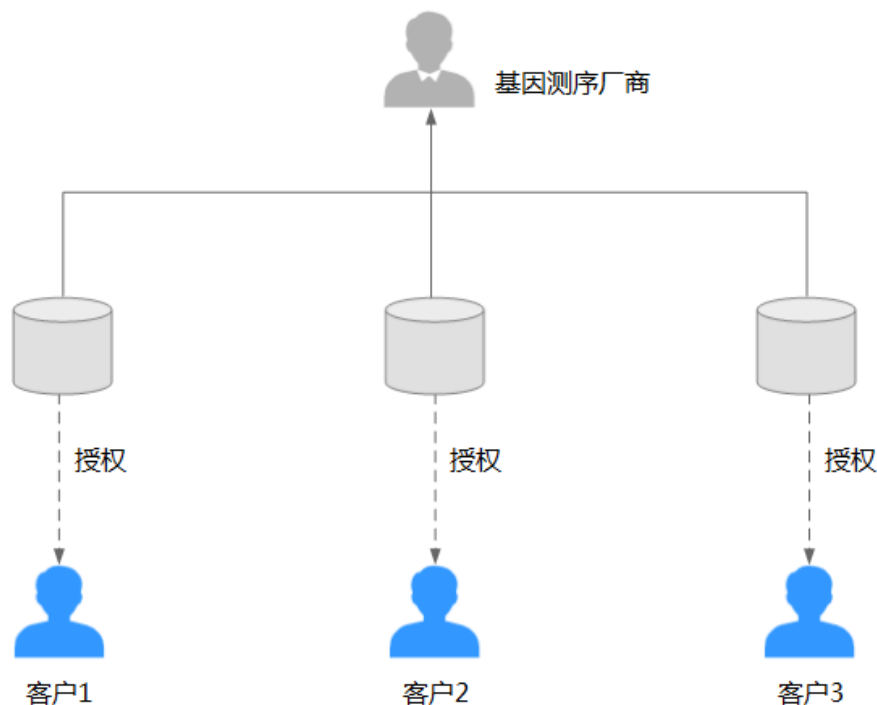
# 2 使用第三方数据执行测序

华为云对象存储服务（Object Storage Service，OBS）支持资源隔离和权限控制，OBS桶是资源隔离和对象存储的逻辑载体，基本原理如图2-1。

假设基因测序厂商账户下已有n个OBS桶，这些桶可分别授权给第三方客户使用。每个第三方账号均为独立的华为云账号，共享桶与第三方账户之间是一一对应的关系，从而实现资源隔离。基因测序厂商可在授权时对桶做精确的权限配置，实现对第三方使用OBS桶的权限控制。第三方只需要登录自己的账号，通过OBS客户端实现原始数据的上传和分析结果的下载。

本章将介绍如何在使用基因容器服务时通过OBS的权限控制来实现资源隔离。

图 2-1 基本原理图



## 基本概念

- 基因测序厂商

- 基因测序服务的提供者，即华为云基因容器服务的客户。
- 第三方  
基因测序服务的使用者，即基因测序厂商的客户。

## 操作步骤

基因测序厂商进行桶配置、权限控制及第三方添加外部桶的操作步骤如下：

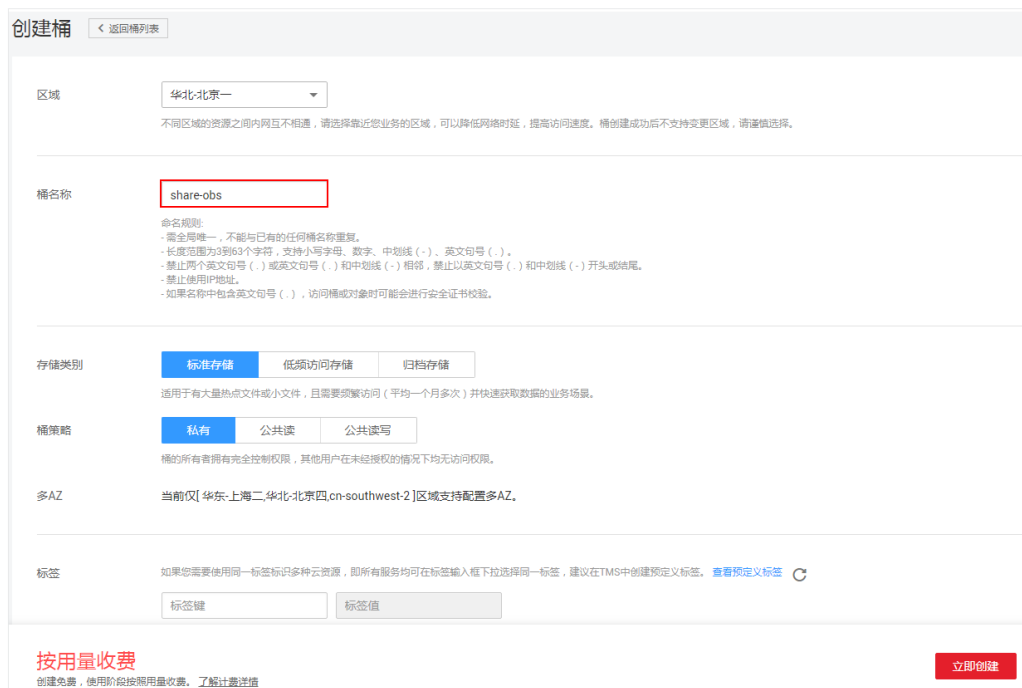
**步骤1** 使用基因测序厂商账号登录**OBS管理控制台**，单击“创建桶”。

图 2-2 OBS 管理控制台



**步骤2** 设置桶参数：

图 2-3 创建 OBS 桶



1. 选择“区域”、“存储类别”，并输入“桶名称”。

### 须知

桶创建成功后，不能修改名称，创建时，请设置合适的桶名。桶名称建议使用第三方账户名，便于区分。

- 桶策略：基因测序厂商可以根据界面提示为桶配置私有、公共读或公共读写策略。
- 多AZ：基因测序厂商可以开启或关闭多AZ。关闭多AZ时，桶内数据存储存储在单个AZ中；开启多AZ时，桶内数据在上传时同时复制双份，保存在3个AZ中。
- 标签：标签用于标识OBS对象存储中的桶，以此达到对OBS对象存储中的桶进行分类的目的。OBS对象存储以键值对的形式来描述标签，每个标签有且只有一对键值。有关添加标签的信息，请参见[标签简介](#)。
- 单击“立即创建”。

**步骤3** OBS桶创建完成后，默认跳转至桶列表。单击桶名称进入桶详情页，单击左侧导航栏中的“权限”，进行桶策略配置。

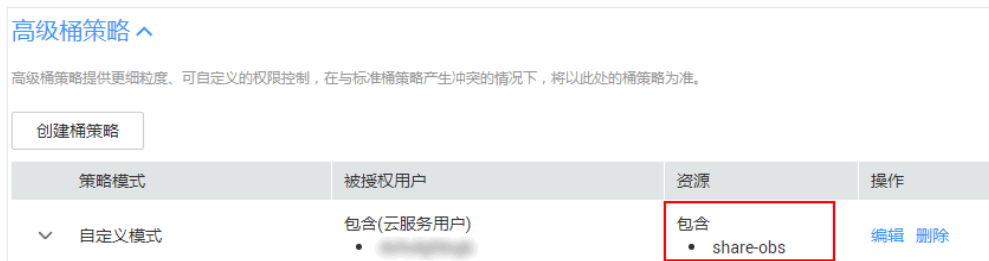
**步骤4** 创建桶级别授权策略：

- 选择“高级桶策略”，单击“创建桶策略”。

图 2-4 创建桶策略



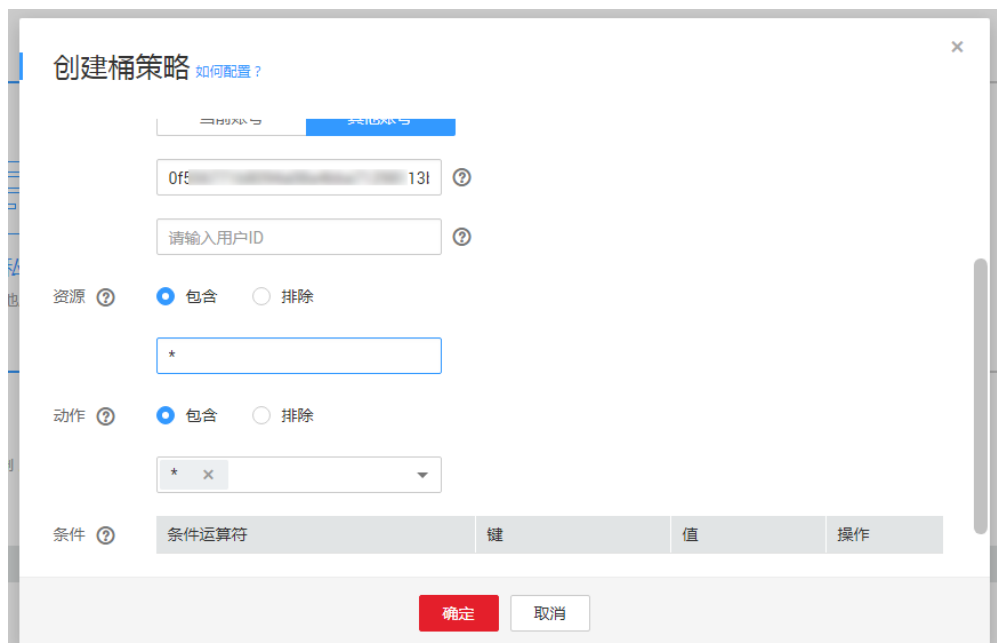
- 在弹出的对话框中，选择“自定义模式”，单击“其他账号”，输入第三方的账号ID（用户ID可省略不填，获取账户ID和用户ID的方法请参见[注意事项](#)）。资源名称设置为空，表示对桶权限的控制。动作列表中选择“\*”，表示授予所有权限（如需精确控制权限，可勾选各权限接口）。
- 单击“确定”，可查看刚创建的桶级别授权策略。



**步骤5** 创建对象级别授权策略：

- 选择“高级桶策略”，单击“创建桶策略”。
- 选择“自定义模式”，单击“其他账号”，输入第三方的账号ID（用户ID可省略不填）。资源名称输入“\*”，表示对桶中所有对象权限的控制。“动作”列表中选择“\*”，表示授予所有权限（如需精确控制权限，可勾选各权限接口）。

图 2-5 创建对象级别授权策略



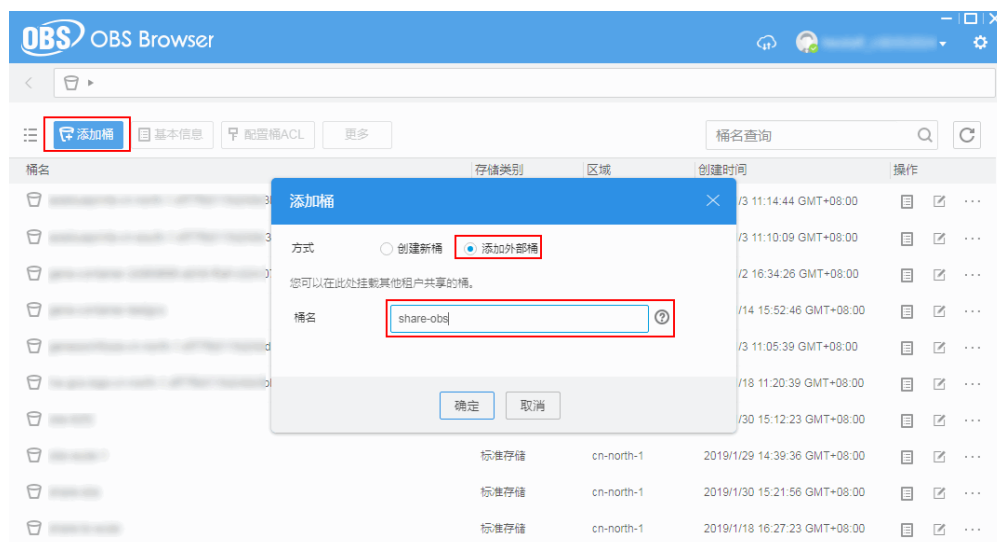
3. 单击“确定”，可查看刚创建的对象级别授权策略。



**步骤6** 第三方添加桶：

基因测序厂商创建完桶并配置好桶策略后，将桶名告知第三方。第三方可使用自己账号的AK/SK登录OBS客户端。登录成功后单击“添加桶”，选择“添加外部桶”，输入基因测序厂商提供的桶名，即可查看到列表中出现的桶。

图 2-6 添加外部桶



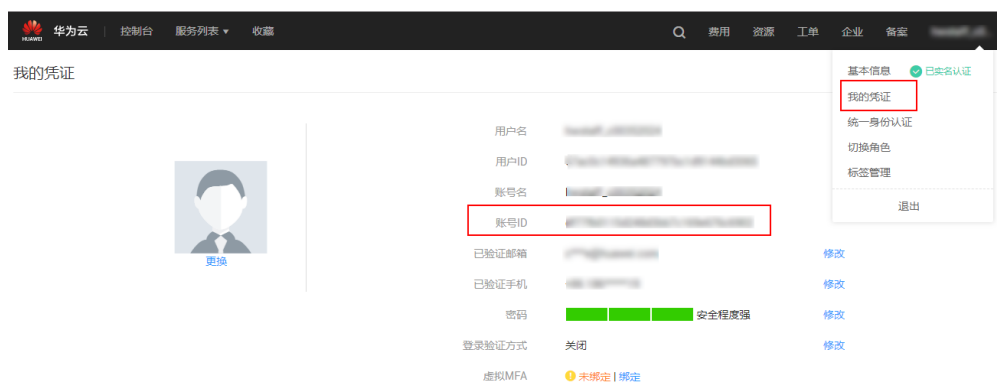
桶添加成功后，第三方即可通过OBS Browser上传和下载数据，基因测序厂商可获取到第三方上传的原始数据进行测序计算，输出分析结果。

----结束

## 注意事项

- OBS桶的拥有者始终为基因测序厂商。测序服务完成后，基因测序厂商可根据需要删除桶或修改桶策略，收回共享权限。
- 基因测序厂商创建的桶必须配置2条高级桶策略，1条是桶级别策略，1条是对象级别策略。
- 基因测序厂商创建桶前，第三方需要将自己的账户ID告知基因测序厂商，账户ID可通过界面右上角账户名下“我的凭证”中查看。

图 2-7 查看账户 ID



- 基因测序厂商创建完桶并配置好桶策略后，将桶名告知第三方。
- 由于单个账号在OBS管理控制台最多创建100个OBS桶，本方案适合第三方数量较少的场景。

# 3 基于 GCS 构建基因测序平台

本通过一个示例Demo介绍如何基于GCS构建一个基因测序平台。

## Demo 下载地址

[https://mirrors.huaweicloud.com/gcs-sdk/GCS\\_CloudOnCloud\\_SDK.rar](https://mirrors.huaweicloud.com/gcs-sdk/GCS_CloudOnCloud_SDK.rar)

### 📖 说明

示例demo源码只实现了最基础的功能，不可直接作为商业软件使用。

示例demo源码为示例源码，开源目的为方便第三方基于此继续进行二次开发，实现定制化需求。

示例demo开源后不会再提供任何维保和版本更新服务。

对于使用过程中的任何问题，华为云保留最终解释权。

## 原理说明

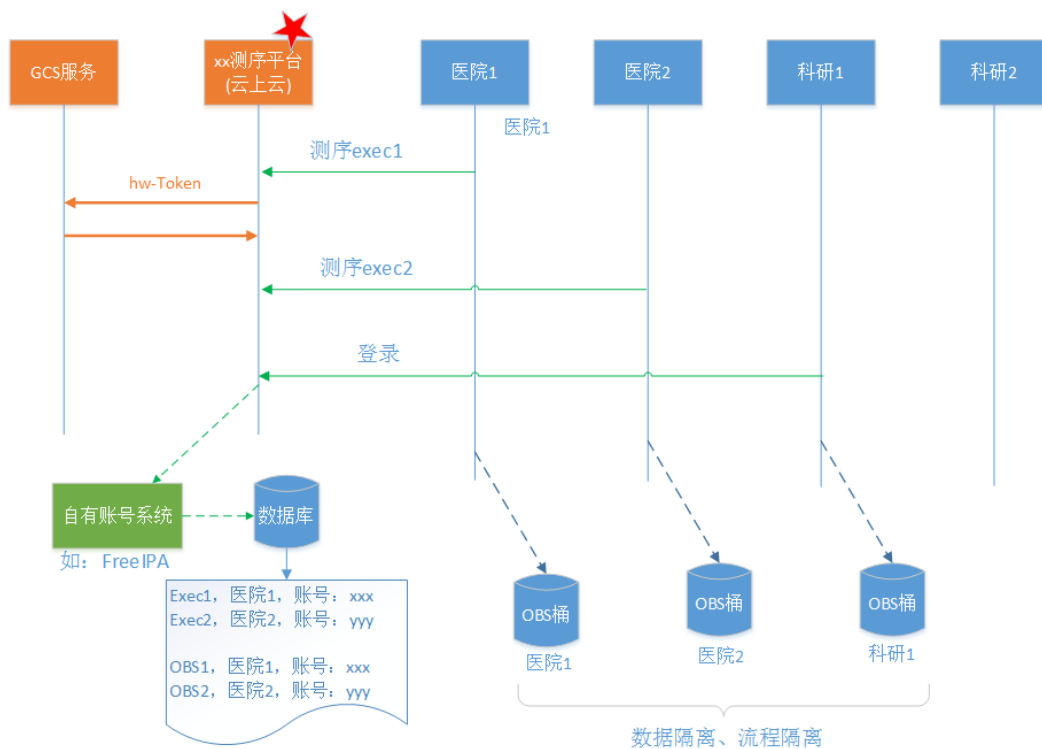
GCS提供了**REST API**，调用API可以使用GCS提供的所有功能。通过调用GCS的API，您可以构建您自己的测序平台，使得用户无需感知GCS。

**图3-1**展示了测序平台（Demo系统）的调用流程，共有如下三种角色。

- 测序平台的用户：使用测序平台的用户，像医院、科研单位等。
- 测序平台：面向用户提供测序服务，通过调用GCS的API响应用户的测序需求。当用户想发起测序请求时，调用GCS的API执行测序，待测序完成后获取测序结果并返回给用户。
- GCS服务：提供底层测序API供测序平台调用。

从上面的描述可以看出，测序平台相当于用户与GCS之间的一个代理，将用户的测序请求转发给GCS处理，然后获取测序结果返回给用户。当然测序平台处理转发测序请求外，还可以做很多其他事情，比如提供用户管理、数据查询等功能，您可以根据自身需求构建。

图 3-1 基于 GCS 构建测序平台



## 代码介绍

示例Demo基于开源框架Beego开发，Beego 是一个快速开发 Go 应用的 HTTP 框架，他可以用来快速开发 API、Web 及后端服务等各种应用。

Beego是一种MVC ( model、view、controller ) 架构的框架。

- 控制器 ( Controller ) - 负责转发请求，对请求进行处理。
- 视图 ( View ) - 界面设计人员进行图形界面设计。
- 模型 ( Model ) - 程序员编写程序应有的功能 ( 实现算法等等 )、数据库专家进行数据管理和数据库设计(可以实现具体的功能)。

