利用阿里云下载国外镜像

在国内下载 k8s.gcr.io 的镜像时，可能会遇到连接超时或下载失败的问题。本文将介绍如何通过阿里云容器镜像服务来顺畅地下载这些镜像。

一、问题描述

在配置使用k8s部署 rook-ceph 集群时，需要下载以下六个镜像，但可能会遇到下载失败的问题：

docker复制

```
ROOK_CSI_CEPH_IMAGE: "quay.io/cephcsi/cephcsi:v3.13.0"
ROOK_CSI_REGISTRAR_IMAGE: "registry.k8s.io/sig-storage/csi-node-driver-registrar:v2.13.0"
ROOK_CSI_RESIZER_IMAGE: "registry.k8s.io/sig-storage/csi-resizer:v1.13.1"
ROOK_CSI_PROVISIONER_IMAGE: "registry.k8s.io/sig-storage/csi-provisioner:v5.1.0"
ROOK_CSI_SNAPSHOTTER_IMAGE: "registry.k8s.io/sig-storage/csi-snapshotter:v8.2.0"
ROOK_CSI_ATTACHER_IMAGE: "registry.k8s.io/sig-storage/csi-attacher:v4.8.0"
```

错误信息如下：

```
Error response from daemon: Head "https://registry.k8s.io/v2/sig-storage/csi-node-driver-registrar/manifests/v2.13.0": dial tcp 34.96.108.209:443: i/o timeout
```

二、解决方法

通过 GitHub 和阿里云容器镜像服务来解决这个问题。

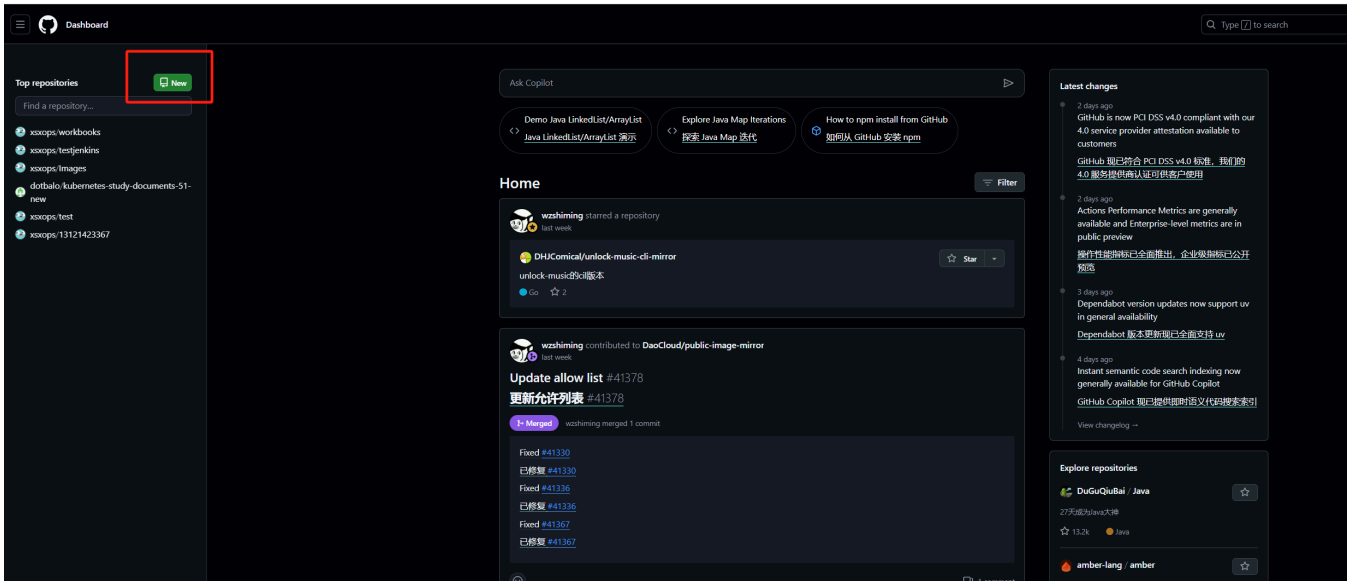
1. 在 GitHub 配置 Dockerfile

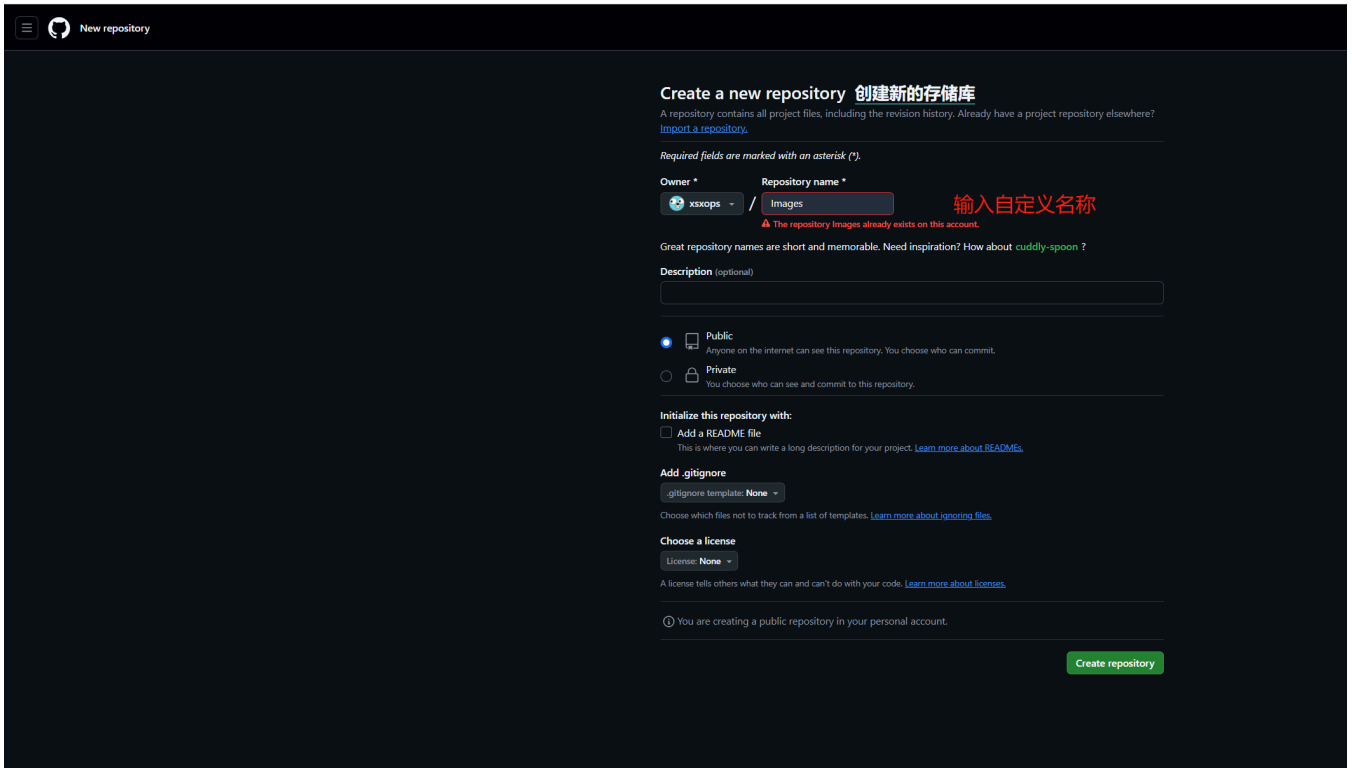
1. 登录 GitHub 并创建一个仓库,比如名称 Images
2. 为每个想要拉取的镜像创建一个目录，目录名称为该镜像的名称并在每个目录下创建一个 Dockerfile。
3. 在 Dockerfile 中指定相应的镜像路径。

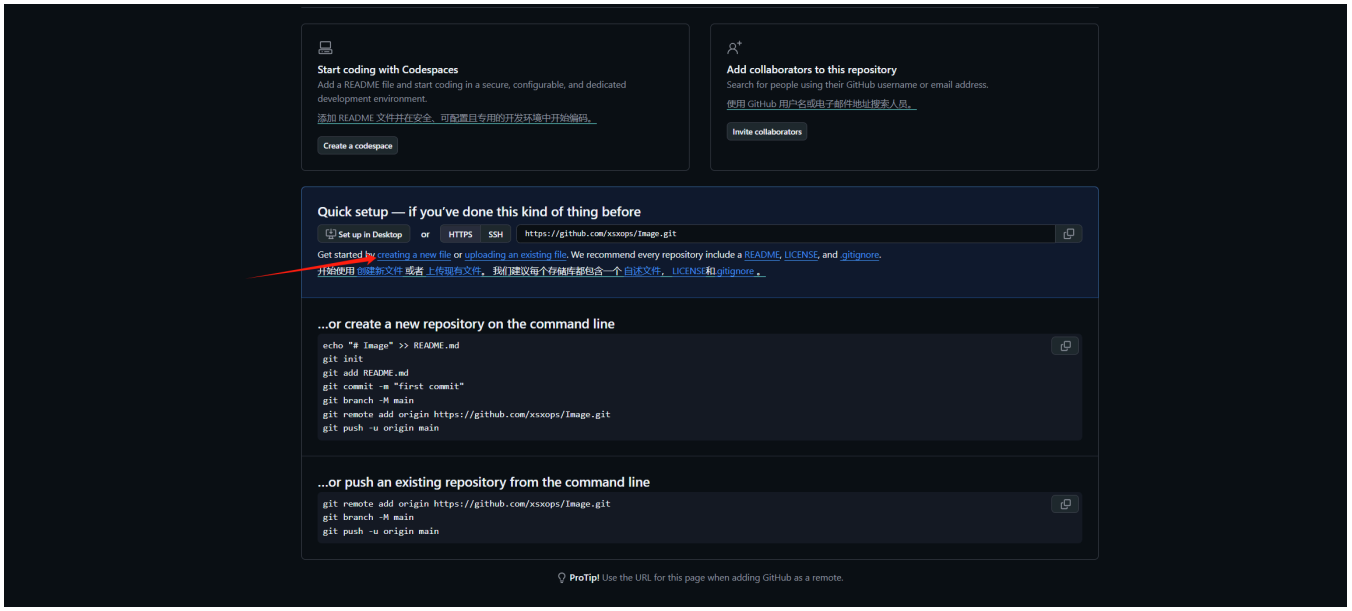
例如，创建六个目录分别对应六个镜像，并在每个目录下创建 Dockerfile：

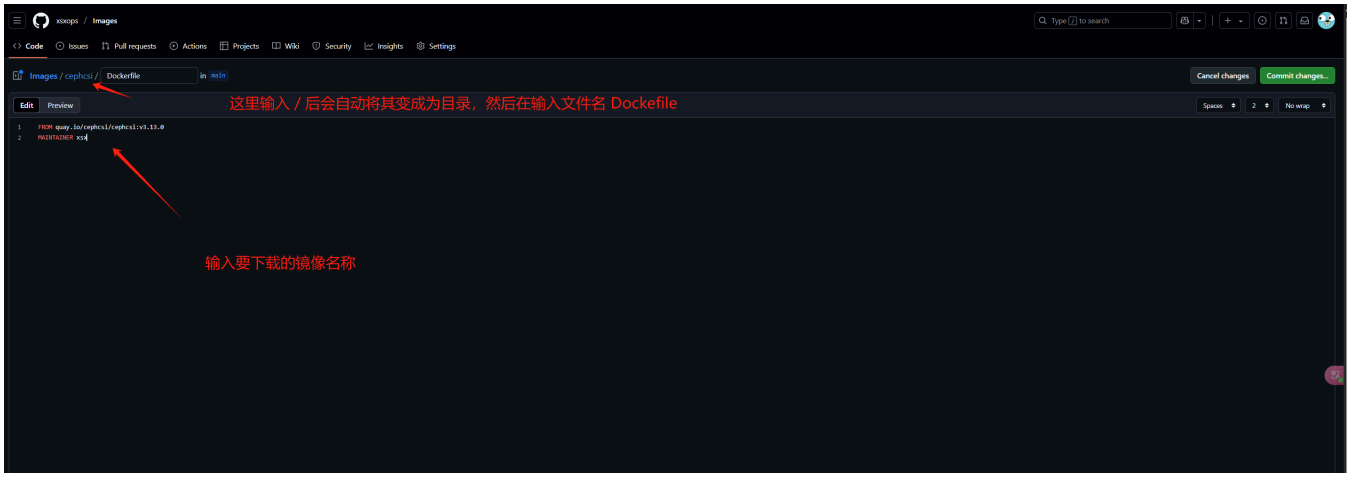
```
# cephcsi/Dockerfile
FROM quay.io/cephcsi/cephcsi:v3.13.0
MAINTAINER xsx

# csi-node-driver-registrar/Dockerfile
FROM registry.k8s.io/sig-storage/csi-node-driver-registrar:v2.13.0
MAINTAINER xsx

# csi-resizer/Dockerfile
FROM registry.k8s.io/sig-storage/csi-resizer:v1.13.1
MAINTAINER xsx

# csi-provisioner/Dockerfile
FROM registry.k8s.io/sig-storage/csi-provisioner:v5.1.0
```

```
MAINTAINER xsx

# csi-snapshotter/Dockerfile
FROM registry.k8s.io/sig-storage/csi-snapshotter:v8.2.0
MAINTAINER xsx

# csi-attacher/Dockerfile
FROM registry.k8s.io/sig-storage/csi-attacher:v4.8.0
MAINTAINER xsx
```







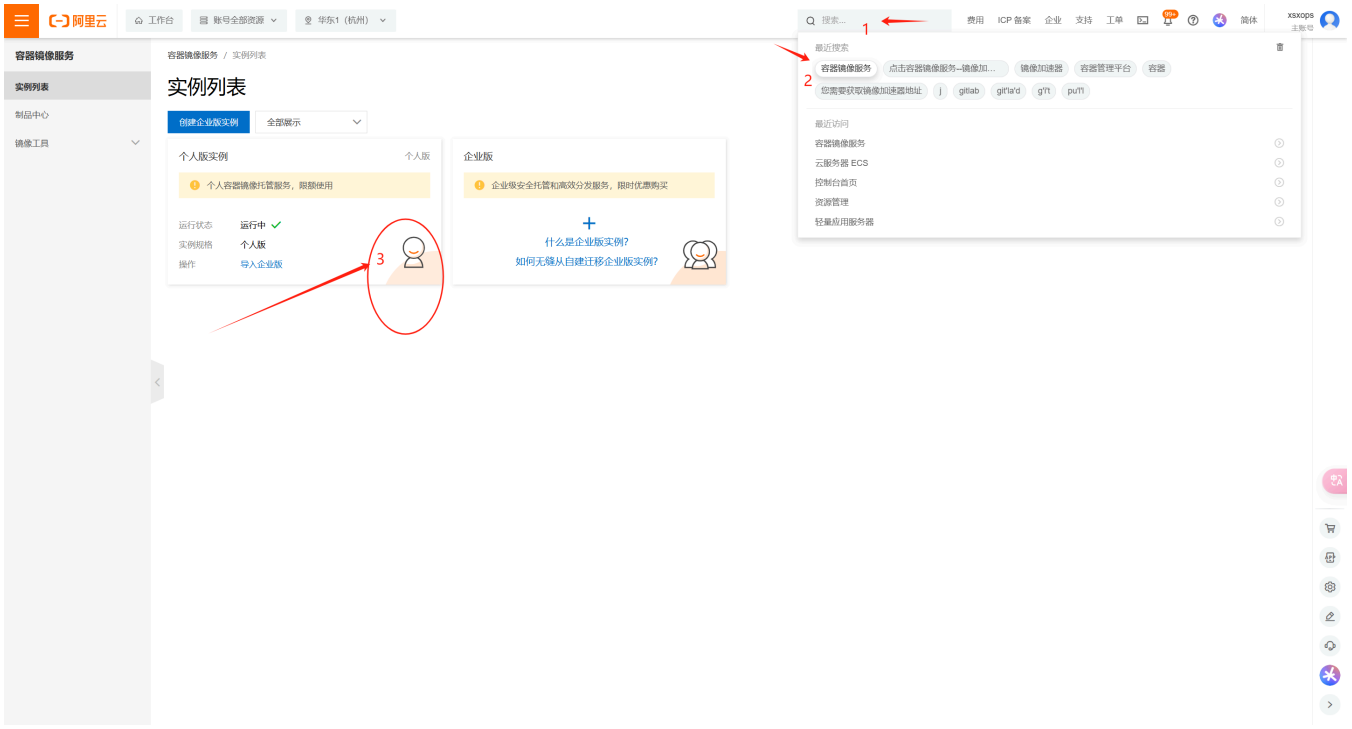


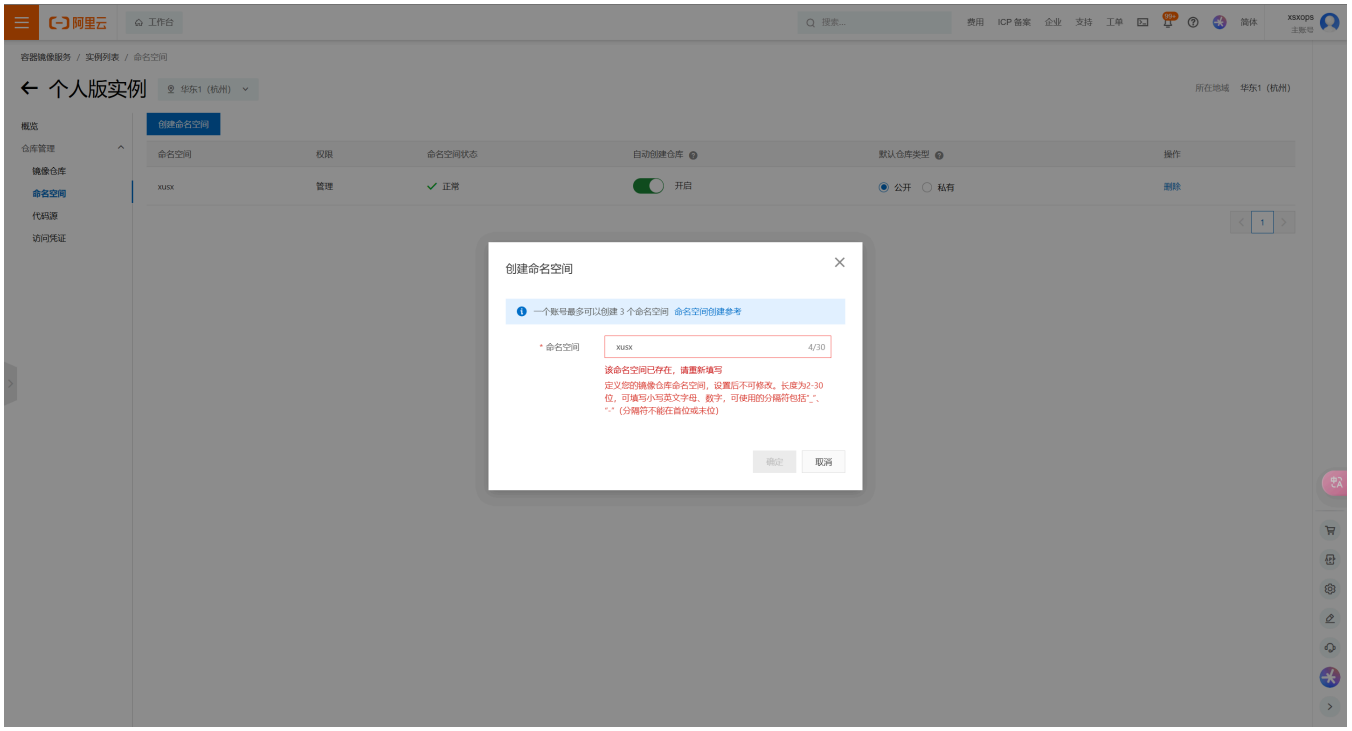
2. 配置阿里云镜像代理

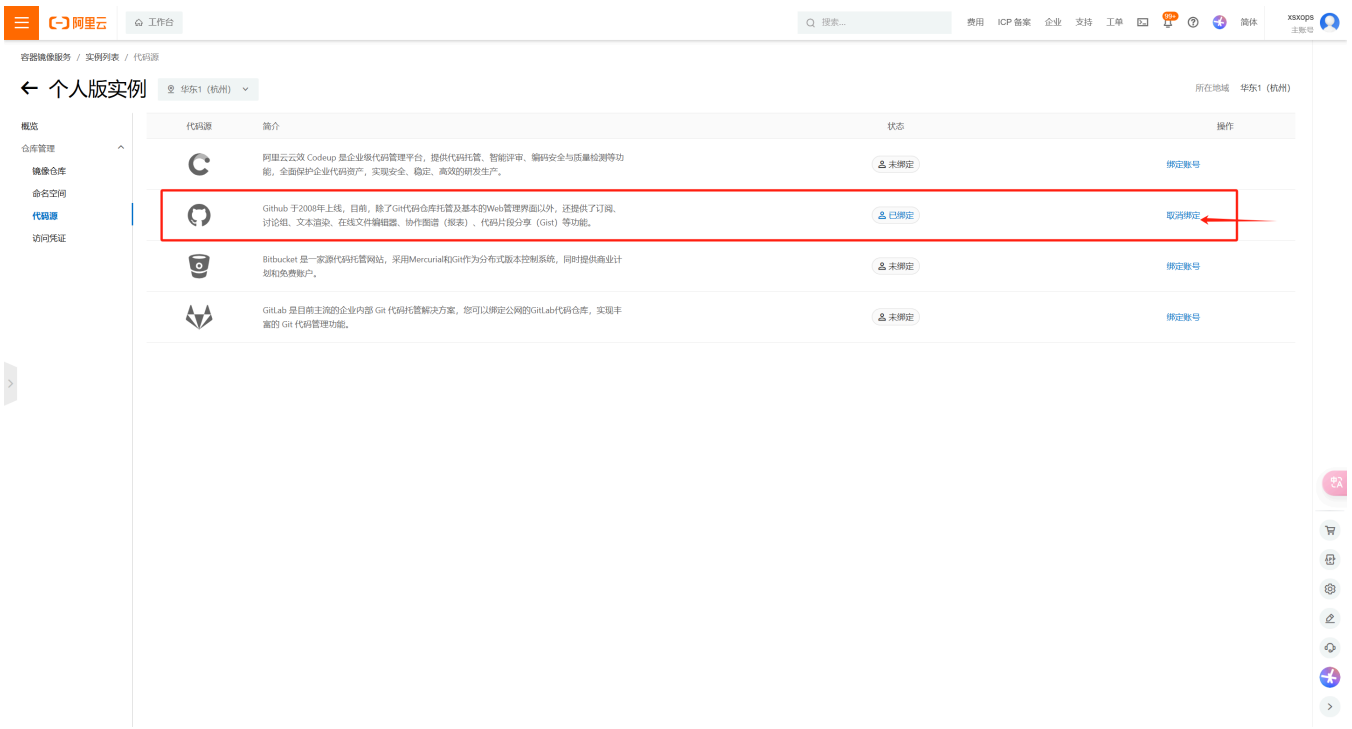
1. 登录阿里云控制台，进入容器镜像服务。
2. 创建一个命名空间（如果尚未创建）。如名称：`xusx`
3. 创建镜像仓库，并授权 GitHub 仓库。
4. 配置构建规则，指定版本号。

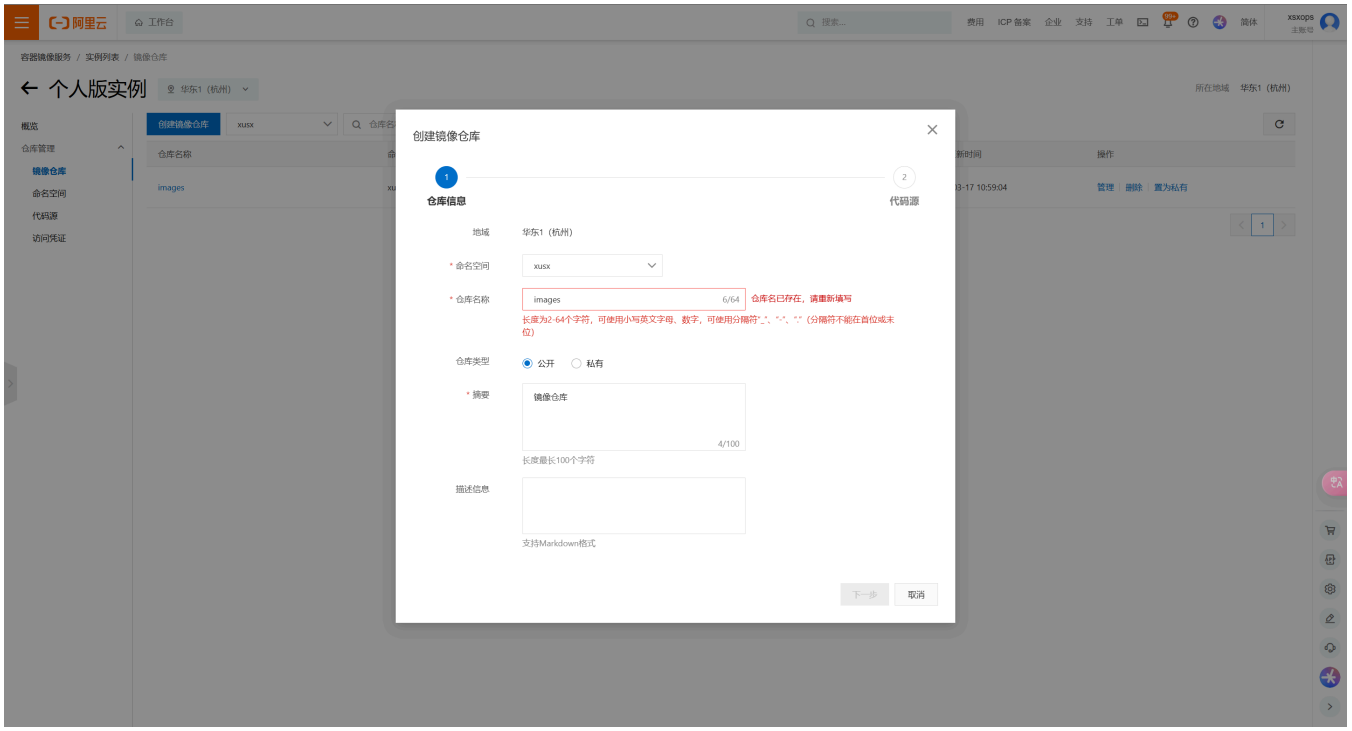
3. 构建镜像

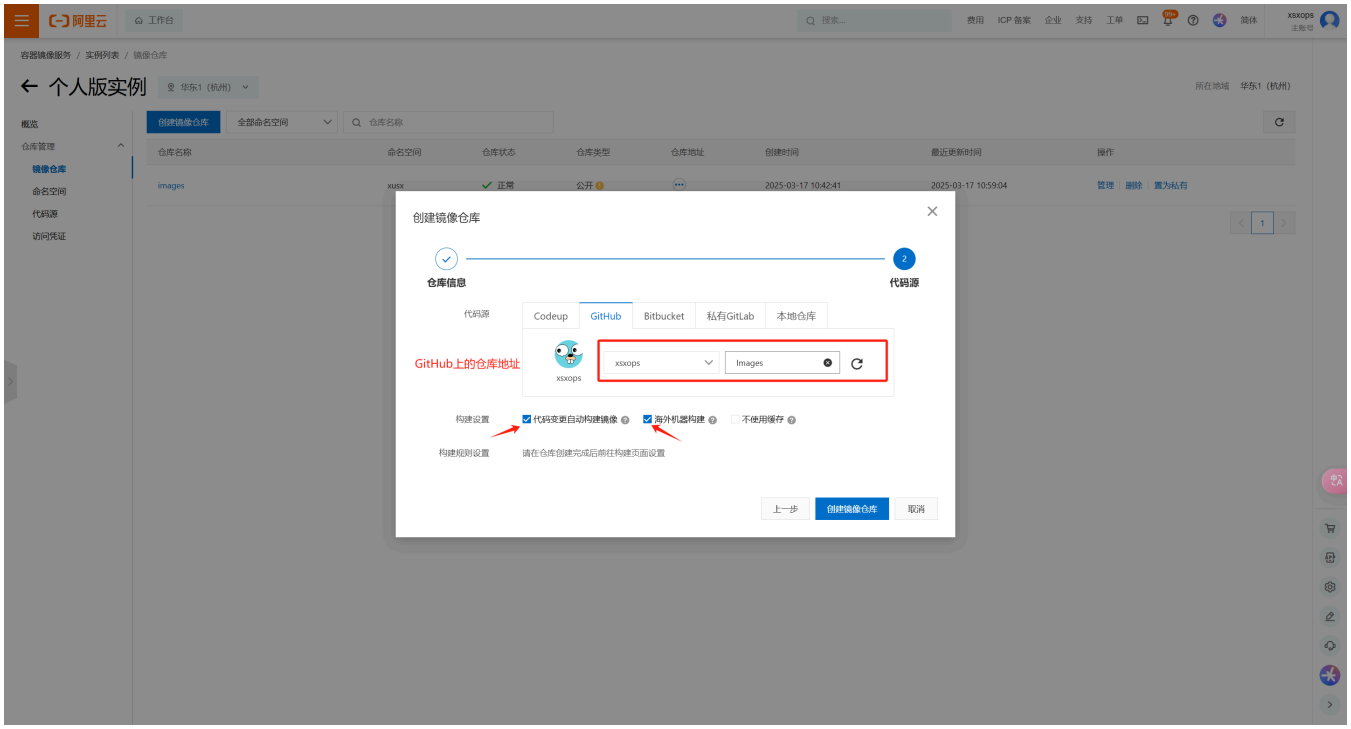
在阿里云容器镜像服务中点击“立即构建”，等待构建完成。

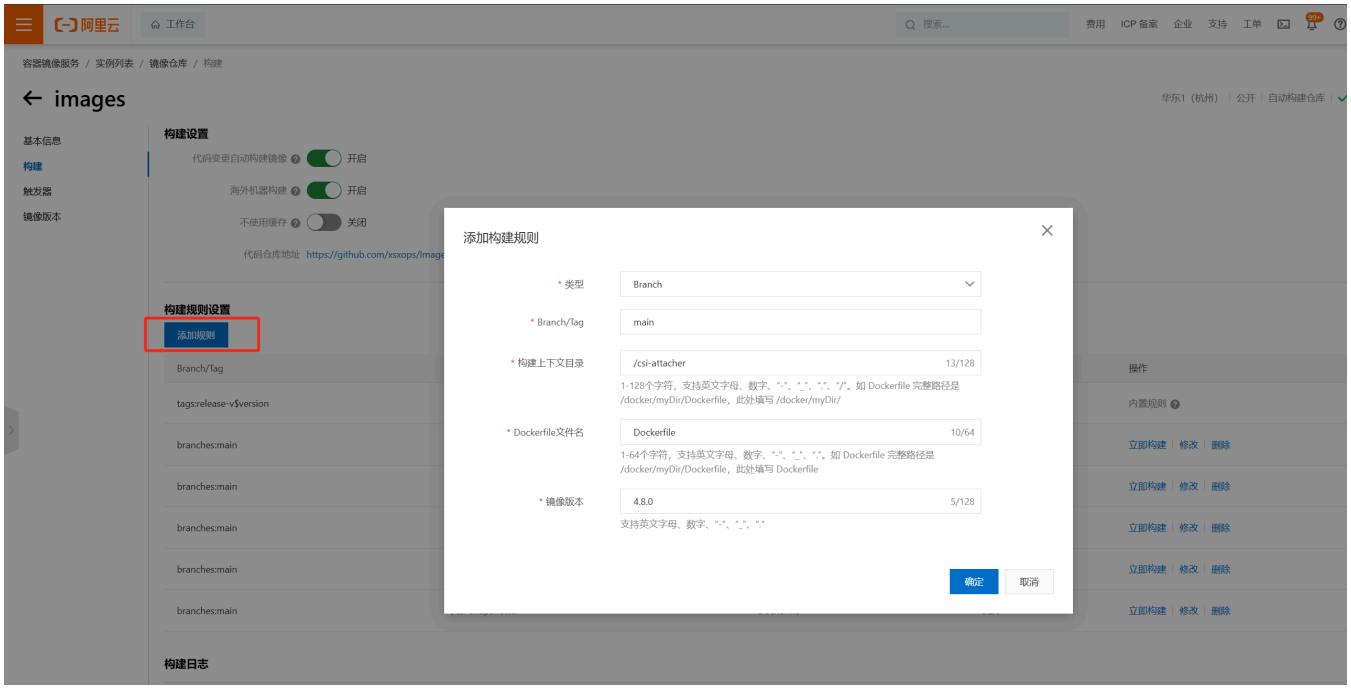

阿里云
工作台
账号全部资源
华东1（杭州）
搜索...
费用
ICP 备案
企业
支持
工单
简体
xsxops
容器镜像服务
实例列表
制品中心
镜像工具
容器镜像服务 / 实例列表
实例列表
创建企业版实例
全部展示
个人版实例
个人版
个人容器镜像托管服务，限额使用
运行状态
运行中
实例规格
个人版
操作
导入企业版
企业版
企业级安全托管和高效分发服务，限时优惠购买
什么是企业版实例?
如何无缝从自建迁移企业版实例?
1
2
3
最近搜索
容器镜像服务
点击容器镜像服务--镜像加...
镜像加速器
容器管理平台
容器
您需要获取镜像加速器地址
j
gitlab
git'la'd
g'ft
pu'ft
最近访问
容器镜像服务
云服务器 ECS
控制台首页
资源管理
轻量应用服务器

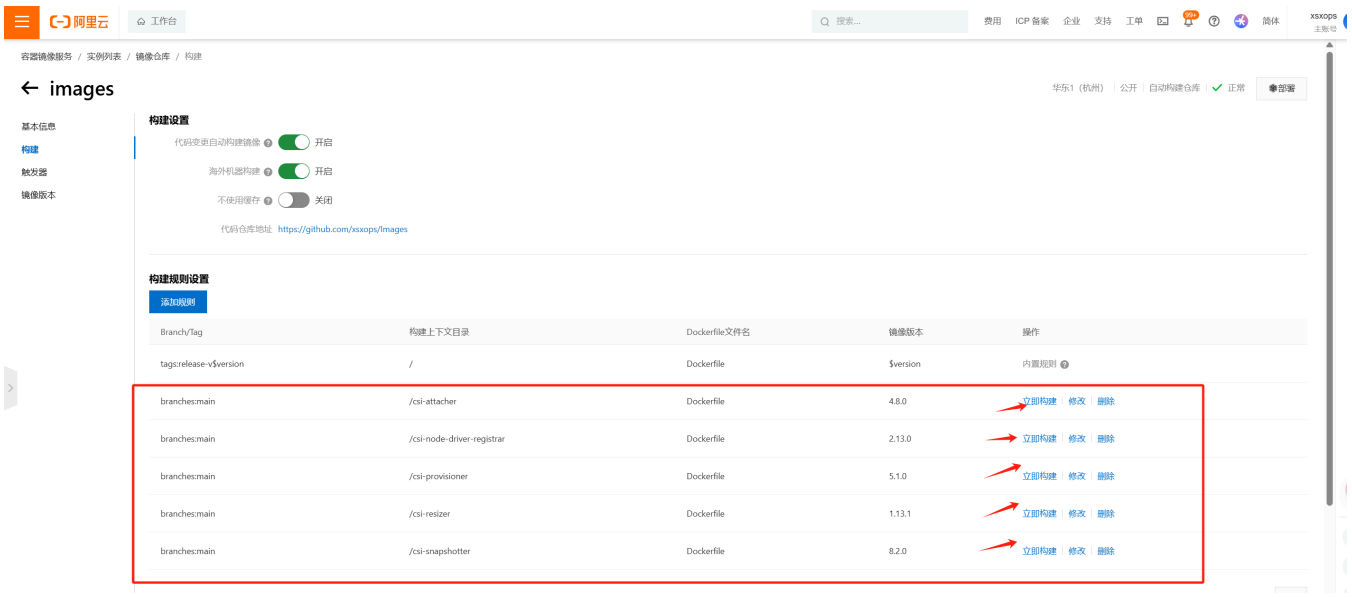

阿里云
工作台
搜索...
费用
ICP 备案
企业
支持
工单
简体
xsxops
容器镜像服务 / 实例列表 / 命名空间
个人版实例
华东1（杭州）
所在地域 华东1（杭州）
概览
仓库管理
镜像仓库
命名空间
代码源
访问凭证
创建命名空间
命名空间
权限
命名空间状态
自动创建仓库
默认仓库类型
操作
xusx
管理
正常
开启
公开
私有
删除
创建命名空间
一个账号最多可以创建 3 个命名空间 命名空间创建参考
* 命名空间
xusx
4/30
该命名空间已存在，请重新填写
定义您的镜像仓库命名空间，设置后不可修改。长度为2-30位，可填写小写英文字母、数字，可使用的分隔符包括"_"、"."（分隔符不能在首位或末位）
确定
取消


阿里云
工作台
搜索...
费用
ICP 备案
企业
支持
工单
简体
xsxops
容器镜像服务 / 实例列表 / 代码源
← 个人版实例
华东1（杭州）
所在地域 华东1（杭州）
概览
仓库管理
镜像仓库
命名空间
代码源
访问凭证
代码源
简介
状态
操作
阿里云云效 Codeup 是企业级代码管理平台，提供代码托管、智能评审、编码安全与质量检测等功能，全面保护企业代码资产，实现安全、稳定、高效的研发生产。
未绑定
绑定账号
Github 于2008年上线，目前，除了Git代码仓库托管及基本的Web管理界面以外，还提供了订阅、讨论组、文本渲染、在线文件编辑器、协作图谱（报表）、代码片段分享（Gist）等功能。
已绑定
取消绑定
Bitbucket 是一家源代码托管网站，采用Mercurial和Git作为分布式版本控制系统，同时提供商业计划和免费账户。
未绑定
绑定账号
GitLab 是目前主流的企业内部 Git 代码托管解决方案，您可以绑定公网的GitLab代码仓库，实现丰富的 Git 代码管理功能。
未绑定
绑定账号


阿里云
工作台
搜索...
费用
ICP 备案
企业
支持
工单
简体
xsxops
容器镜像服务 / 实例列表 / 镜像仓库
← 个人版实例
华东1（杭州）
所在地域 华东1（杭州）
概览
仓库管理
镜像仓库
命名空间
代码源
访问凭证
创建镜像仓库
xusx
仓库名称
images
管理
删除
置为私有
创建镜像仓库
仓库信息
代码源
地域
华东1（杭州）
* 命名空间
xusx
* 仓库名称
images
6/64
仓库名已存在，请重新填写
长度为2-64个字符，可使用小写英文字母、数字，可使用分隔符"_"、"-"、"."（分隔符不能在首位或末位）
仓库类型
公开
私有
* 摘要
镜像仓库
4/100
长度最长100个字符
描述信息
支持Markdown格式
下一步
取消


个人版实例
创建镜像仓库
仓库信息
代码源
Codeup
GitHub
Bitbucket
私有GitLab
本地仓库
GitHub上的仓库地址
xsxops
Images
构建设置
代码变更自动构建镜像
海外机器构建
不使用缓存
构建规则设置
请在仓库创建完成后前往构建页面设置
上一步
创建镜像仓库
取消


images
构建设置
代码变更自动构建镜像
开启
海外机器构建
开启
不使用缓存
关闭
构建规则设置
添加规则
Branch/Tag
操作
tags:release-v$version
branches:main
构建日志
添加构建规则
* 类型
Branch
* Branch/Tag
main
* 构建上下文目录
/csi-attacher
13/128
1-128个字符，支持英文字母、数字、"-"、"_"、"."、"/"。如 Dockerfile 完整路径是 /docker/myDir/Dockerfile，此处填写 /docker/myDir/
* Dockerfile文件名
Dockerfile
10/64
1-64个字符，支持英文字母、数字、"-"、"_"、"."。如 Dockerfile 完整路径是 /docker/myDir/Dockerfile，此处填写 Dockerfile
* 镜像版本
4.8.0
5/128
支持英文字母、数字、"-"、"_"、"."
确定
取消



4. 下载镜像

使用 Docker 命令下载镜像：

```
#登录阿里云Docker Registry
docker login --username=xsxops registry.cn-hangzhou.aliyuncs.com

#从Registry中拉取镜像,根据刚刚输入的版本号去判断是那个镜像
docker pull registry.cn-hangzhou.aliyuncs.com/xusx/images:2.13.0

[root@centos7 ~]# docker images
REPOSITORY                                                TAG             IMAGE ID
CREATED          SIZE
registry.cn-hangzhou.aliyuncs.com/xusx/images             2.13.0          91db6594b285    2
months ago     30.1MB
```

```
ctr images pull registry.cn-hangzhou.aliyuncs.com/xusx/images:2.13.0
```

```
crictl pull registry.cn-hangzhou.aliyuncs.com/xusx/images:2.13.0
```

5. 镜像打标签并上传到 Harbor

为下载的镜像打标签和上传到内部 Harbor 仓库：

```
docker tag registry.cn-hangzhou.aliyuncs.com/xusx/images:v2.13.0 registry.k8s.io/sig-
storage/csi-node-driver-registrar:v2.13.0

docker rmi registry.cn-hangzhou.aliyuncs.com/xusx/images:v2.13.0
```

```
ctr images tag registry.cn-hangzhou.aliyuncs.com/xusx/images:2.13.0 registry.k8s.io/sig-
storage/csi-node-driver-registrar:v2.13.0

ctr image rm registry.cn-hangzhou.aliyuncs.com/xusx/images:2.13.0
```

```
crictl tag registry.cn-hangzhou.aliyuncs.com/xusx/images:2.13.0 registry.k8s.io/sig-storage/csi-node-driver-registrar:v2.13.0
```

上传镜像

```
docker push registry.cn-hangzhou.aliyuncs.com/xusx/images:[镜像版本号]
```

三、快速推送镜像

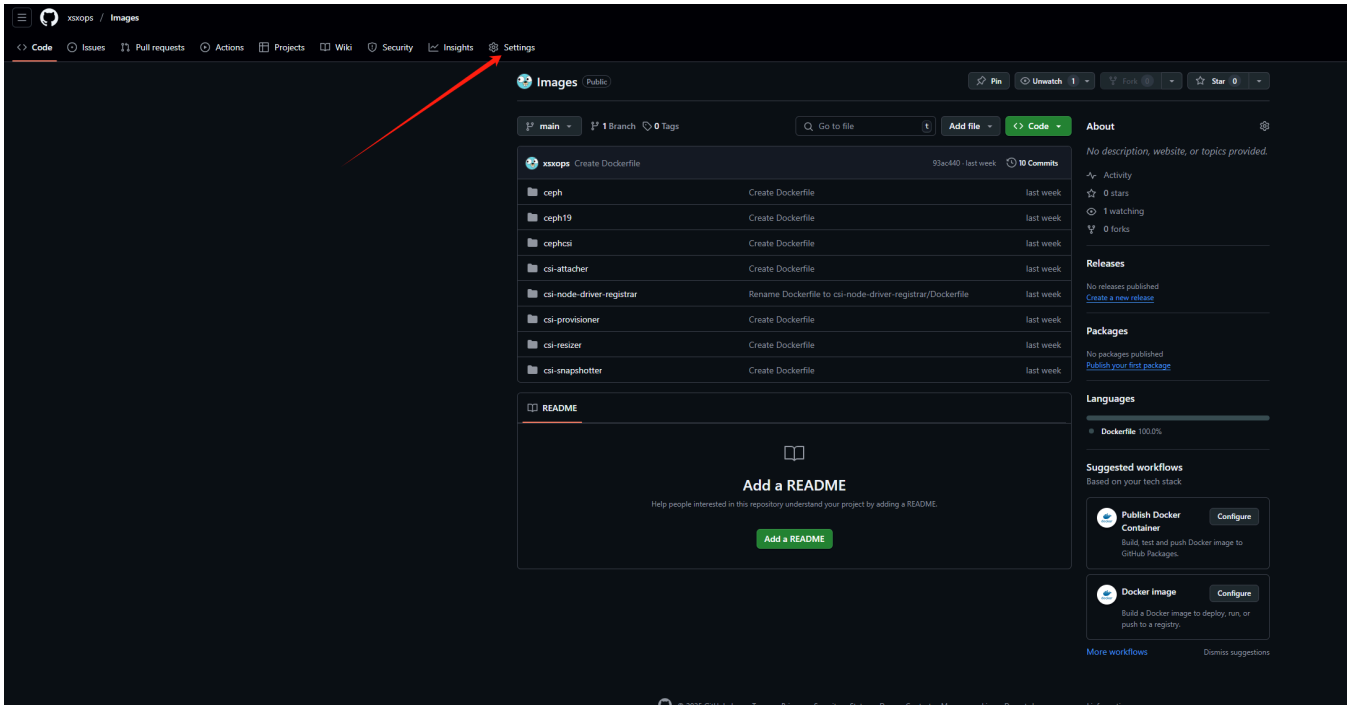
1. 在电脑克隆镜像仓库

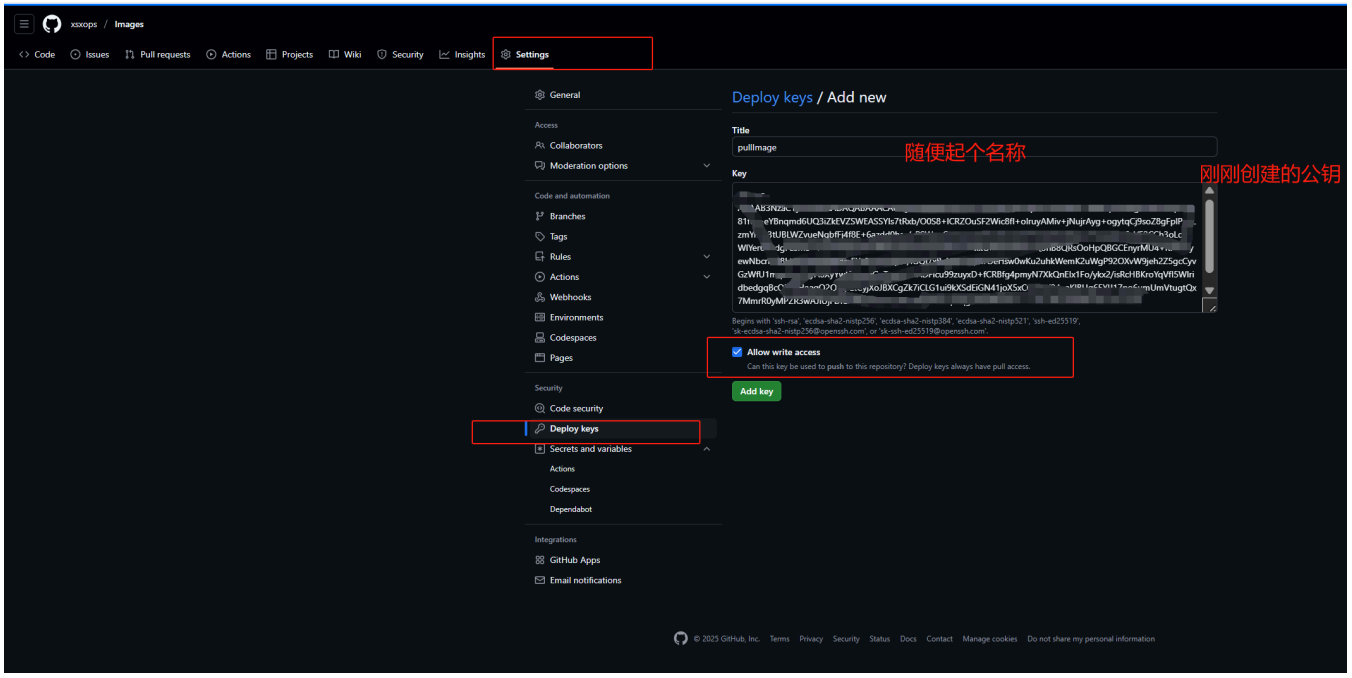
```
git clone https://github.com/xsxops/Images.git
```

```
13121@▓▓▓ MINGW64 /d/Desktop/镜像Hub仓库
$ git clone https://github.com/xsxops/Images.git
Cloning into 'Images'...
remote: Enumerating objects: 36, done.
remote: Counting objects: 100% (36/36), done.
remote: Compressing objects: 100% (24/24), done.
remote: Total 36 (delta 6), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (36/36), 9.51 KiB | 70.00 KiB/s, done.
Resolving deltas: 100% (6/6), done.

13121@▓▓▓ MINGW64 /d/Desktop/镜像Hub仓库
$ ssh-keygen -t rsa -b 4096 -C "13121423367@163.com.com"
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/13121/.ssh/id_rsa):
Created directory '/c/Users/13121/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/13121/.ssh/id_rsa
Your public key has been saved in /c/Users/13121/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:Luh4Ni+XHHefyS84UALnE9FSLIvohzrXiIhmWch9ETs 13121423367@163.com.com
The key's randomart image is:
+---[RSA 4096]----+
|         .=.     |
|      .. = o     |
|     .o= =       |
|    .E. = .      |
|. o. .o S+       |
| o oooo.o .      |
|..ooo=.+.o + o   |
|o++oB =.  o *    |
|o .=.=.    . o.  |
+----[SHA256]-----+

13121@▓▓▓ MINGW64 /d/Desktop/镜像Hub仓库
```

2.对代码仓库新建密钥

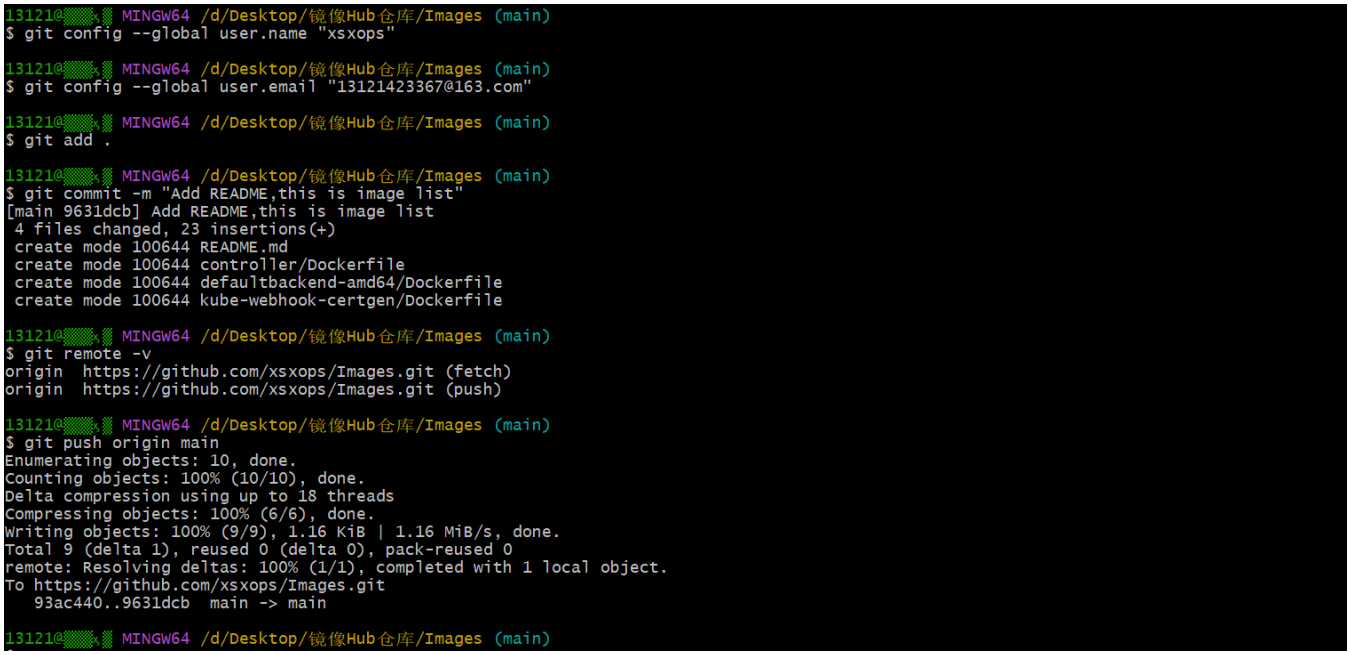
```
cat ~/.ssh/id_rsa.pub
```


xsxops / Images
Code
Issues
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
Images
Public
Pin
Unwatch 1
Fork 0
Star 0
main
1 Branch
0 Tags
Go to file
Add file
Code
About
No description, website, or topics provided.
Activity
0 stars
1 watching
0 forks
xsxops Create Dockerfile
93ac440 · last week
10 Commits
ceph Create Dockerfile last week
ceph19 Create Dockerfile last week
cephcsi Create Dockerfile last week
csi-attacher Create Dockerfile last week
csi-node-driver-registrar Rename Dockerfile to csi-node-driver-registrar/Dockerfile last week
csi-provisioner Create Dockerfile last week
csi-resizer Create Dockerfile last week
csi-snapshotter Create Dockerfile last week
Releases
No releases published
Create a new release
Packages
No packages published
Publish your first package
Languages
Dockerfile 100.0%
README
Add a README
Help people interested in this repository understand your project by adding a README.
Add a README
Suggested workflows
Based on your tech stack
Publish Docker Container
Configure
Build, test and push Docker image to GitHub Packages.
Docker image
Configure
Build a Docker image to deploy, run, or push to a registry.
More workflows
Dismiss suggestions


xsxops / Images
Code
Issues
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
General
Access
Collaborators
Moderation options
Code and automation
Branches
Tags
Rules
Actions
Webhooks
Environments
Codespaces
Pages
Security
Code security
Deploy keys
Secrets and variables
Actions
Codespaces
Dependabot
Integrations
GitHub Apps
Email notifications
Deploy keys / Add new
Title
pullImage
随便起个名称
Key
刚刚创建的公钥
Allow write access
Can this key be used to push to this repository? Deploy keys always have pull access.
Add key
© 2025 GitHub, Inc. Terms Privacy Security Status Docs Contact Manage cookies Do not share my personal information

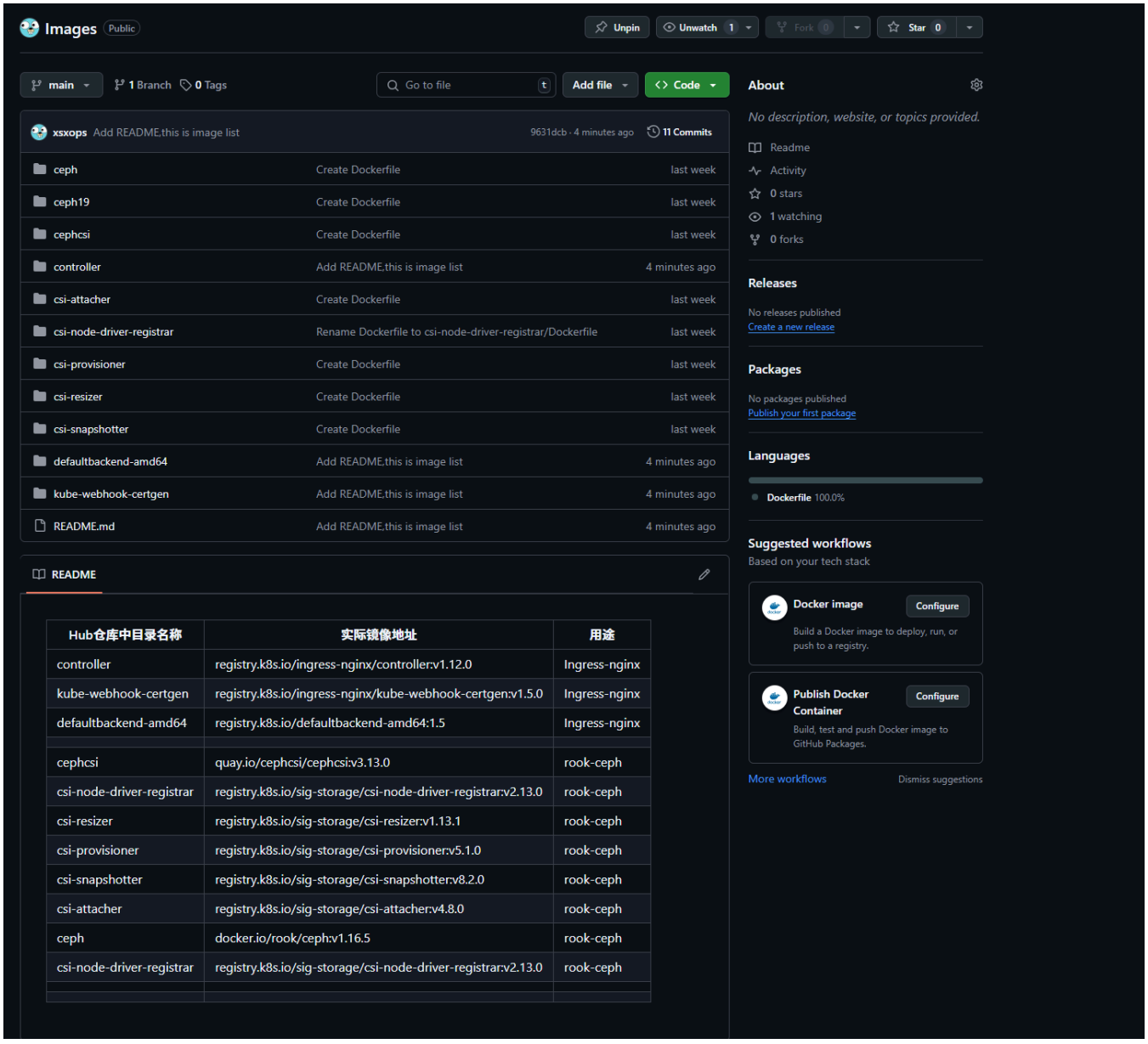
3.配置仓库信息

```
# 配置用户名和邮箱
git config --global user.name "xsxops"
git config --global user.email "13121423367@163.com"

# 查看本地远端仓库的配置
git remote -v
git config --list


#新增镜像配置内容后，推送到远端仓库
git add .
git commit -m "Add README,this is image list"
git push origin main
```



4.验证

打开GitHub 上发现，push的代码已生效



Hub仓库中目录名称	实际镜像地址	用途
controller	registry.k8s.io/ingress-nginx/controller:v1.12.0	Ingress-nginx
kube-webhook-certgen	registry.k8s.io/ingress-nginx/kube-webhook-certgen:v1.5.0	Ingress-nginx
defaultbackend-amd64	registry.k8s.io/defaultbackend-amd64:1.5	Ingress-nginx
cephcsi	quay.io/cephcsi/cephcsi:v3.13.0	rook-ceph
csi-node-driver-registrar	registry.k8s.io/sig-storage/csi-node-driver-registrar:v2.13.0	rook-ceph
csi-resizer	registry.k8s.io/sig-storage/csi-resizer:v1.13.1	rook-ceph
csi-provisioner	registry.k8s.io/sig-storage/csi-provisioner:v5.1.0	rook-ceph
csi-snapshotter	registry.k8s.io/sig-storage/csi-snapshotter:v8.2.0	rook-ceph
csi-attacher	registry.k8s.io/sig-storage/csi-attacher:v4.8.0	rook-ceph
ceph	docker.io/rook/ceph:v1.16.5	rook-ceph
csi-node-driver-registrar	registry.k8s.io/sig-storage/csi-node-driver-registrar:v2.13.0	rook-ceph