

容器镜像服务

常见问题

文档版本 01
发布日期 2025-09-19



版权所有 © 华为技术有限公司 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。

安全声明

漏洞处理流程

华为公司对产品漏洞管理的规定以“漏洞处理流程”为准，该流程的详细内容请参见如下网址：

<https://www.huawei.com/cn/psirt/vul-response-process>

如企业客户须获取漏洞信息，请参见如下网址：

<https://securitybulletin.huawei.com/enterprise/cn/security-advisory>

目录

1 通用类	1
1.1 什么是容器镜像服务？	1
1.2 产品咨询	1
1.3 如何制作容器镜像？	2
1.4 如何制作镜像包？	6
1.5 SWR 的配额是多少？	6
2 登录问题	8
3 镜像推送	9
4 镜像拉取	11
5 镜像同步	13
5.1 是否支持跨区域同步镜像？	13
5.2 为什么 IAM 用户无法设置镜像同步？	13
5.3 为什么已有镜像自动同步不成功？	13
5.4 他人共享的镜像，是否可以再进行镜像同步？	14
5.5 为什么镜像自动同步后在同步区域看不到镜像？	14
6 故障类	15
6.1 为什么登录指令执行失败？	15
6.2 为什么使用客户端上传镜像失败？	17
6.3 为什么通过页面上传镜像失败？	18
6.4 为什么 docker pull 指令执行失败？	20
6.5 如何解决内网下载镜像失败？	21
6.6 为什么创建组织失败？	22
7 其他	23
7.1 为什么 CCE 工作负载拉取 SWR 内的镜像异常，且提示为“未登录”？	23
7.2 SWR 私有镜像最多可以共享给多少个租户？	23
7.3 为什么通过客户端和页面上传的镜像大小不一样？	24
7.4 控制台页面的镜像可以下载到本地吗？	24

1 通用类

1.1 什么是容器镜像服务？

容器镜像服务（SoftWare Repository for Container，简称SWR）是一种支持容器镜像全生命周期管理的服务，提供简单易用、安全可靠的镜像管理功能，帮助用户快速部署容器化服务。

通过使用容器镜像服务，您无需自建和维护镜像仓库，即可享有云上的镜像安全托管及高效分发服务，并且可配合[云容器引擎 CCE](#)使用，获得容器上云的顺畅体验。

欲了解更多请参见[产品介绍](#)。

1.2 产品咨询

容器镜像服务是否收费？

容器镜像服务企业版收费，基础版免费。

请问 SWR 是否支持查询一个镜像的 cpu 架构（x86 or ARM）？

- 对于公共镜像，您可以登录[SWR控制台](#)，进入镜像中心，搜索资源并查看镜像的详细信息，包括该镜像支持的架构。
- 对于您的私有镜像，您可以使用`docker inspect` [镜像名称：版本名称]，查询该镜像的架构。

示例：`docker inspect openjdk:7`。

图 1-1 示例

```
},
  "Architecture": "amd64",
  "Os": "Linux",
  "Size": 621209007,
  "VirtualSize": 621209007,
  "GraphDriver": {
    "Data": {
      "LowerDir": "/var/lib/docker/overlay2/93ff8e16f44919ca8ec9e5c5077ea166c79a3f5f1dbf46288390a77d32cedf7b/diff:/var/lib/docker/overlay2/579337e171a09f26649010fadac542b9b050079fda9a97837450c20ebc36076e/diff:/var/lib/docker/overlay2/d9b3e45a133974fcb00a8a78c4462f066b56ba3d3af23fa4ba59cc1cf6efafb2/diff",
      "MergedDir": "/var/lib/docker/overlay2/09ead26ee61e1ecfd8556963d768e888a63b8f51dc506b4c806c7408e1b76852/merged",
      "UpperDir": "/var/lib/docker/overlay2/09ead26ee61e1ecfd8556963d768e888a63b8f51dc506b4c806c7408e1b76852/diff",
      "WorkDir": "/var/lib/docker/overlay2/09ead26ee61e1ecfd8556963d768e888a63b8f51dc506b4c806c7408e1b76852/work"
    },
    "Name": "overlay2"
  },
  "RootFS": {
    "Type": "layers",
    "Layers": [
      "sha256:174f5685490326fc8a1c8f5570b8663732189b327007e47ff13d2ca59673db02",
      "sha256:fd431e3bcbf54e46ee912e04becffa279cd866c278c7f74dd0a15fca05dcdb9b",
      "sha256:f0200dd2ca0ff94a8beacb8df805ff2c9c4843dac363abea052181063b535445",
      "sha256:0f137c758756a06bc5bd64d05ceb636525f1267fbef49b8147f651a062b54ea7"
    ]
  }
}
```

SWR 最多能存储多少个镜像？

SWR对存储的镜像数量有配额限制，详见[swr_faq_0006.xml#swr_faq_0006/section626111350212](#)。

容器镜像服务的带宽多大？

容器镜像服务的带宽会根据用户使用情况动态变化。

为什么无法查看容器交付流水线页面？

因业务调整，容器交付流水线（ContainerOps）功能暂停开通，如您在2020-06-25之前已经开通了流水线功能，则可以继续使用。为您带来不便，敬请谅解。

1.3 如何制作容器镜像？

自己制作容器镜像，主要有两种方法：

- 制作快照方式获得镜像（偶尔制作的镜像）：在基础镜像上（比如Ubuntu），先登录镜像系统并安装容器引擎软件，然后整体制作快照。适用于后续镜像没有变化的场景。
- Dockerfile方式构建镜像（经常更新的镜像）：将软件安装的流程写成Dockerfile，使用docker build构建容器镜像。适用于后续镜像需要经常更换的场景。

方法一：制作快照方式获得镜像

如果后续镜像没有变化，可采用[方法一](#)制作镜像。



具体操作如下：

1. 找一台主机，安装容器引擎软件。
2. 启动一个空白的基础容器，并进入容器。
例如：启动一个CentOS的容器。

docker run -it centos

3. 执行安装任务。

yum install XXX

git clone <https://github.com/lh3/bwa.git>

cd bwa;make

📖 说明

请预先安装好Git，并检查本机是否有ssh key设置。

4. 输入**exit**退出容器。
5. 制作快照。

docker commit -m "xx" -a "test" container-id test/image:tag

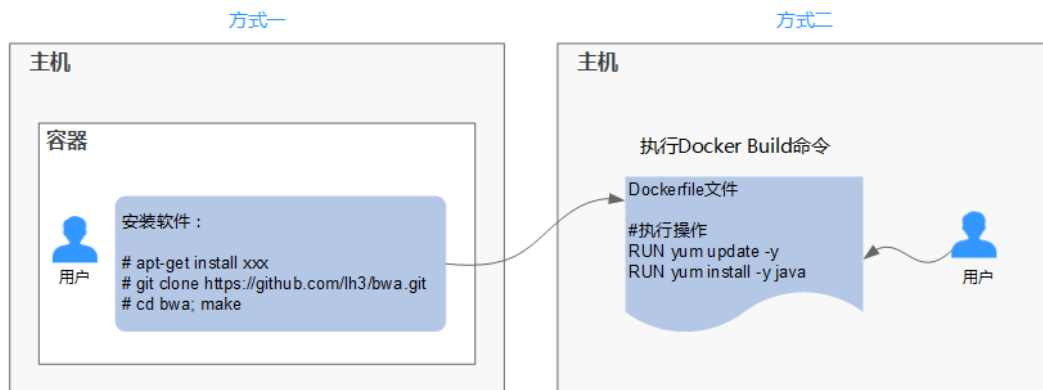
- a：提交的镜像作者。
- container-id：步骤2中的容器id。可以使用**docker ps -a**查询得到容器id。
- m：提交时的说明文字。
- test/image:tag：仓库名/镜像名:TAG名。

6. 执行**docker images**可以查看到制作完成的容器镜像。

方法二：使用 Dockerfile 方式构建

如果后续镜像经常变更（例如某个软件更新版本），则需要采用方法二制作镜像。若仍采用方法一制作镜像，则每次变更都需要重新执行方法一的命令，操作过程比较繁琐，所以建议使用自动化制作镜像的方法。

其实就是将方法一制作镜像的方法，用文件方式写出来（文件名为Dockerfile）。然后执行：**docker build -t test/image:tag**命令（命令中“.”表示Dockerfile文件的路径），自动完成镜像制作。



简单的Dockerfile示例：

📖 说明

如果依赖外部网络，请搭建网络环境，并保证网络可用。

```
#Version 1.0.1
FROM centos:latest

#设置root用户为后续命令的执行者
USER root

#执行操作
RUN yum update -y
RUN yum install -y java

#使用&&拼接命令
RUN touch test.txt && echo "abc" >>abc.txt

#对外暴露端口
EXPOSE 80 8080 1038

#添加网络文件
ADD https://www.baidu.com/img/bd_logo1.png /opt/

#设置环境变量
ENV WEBAPP_PORT=9090

#设置工作目录
WORKDIR /opt/

#设置启动命令
ENTRYPOINT ["ls"]

#设置启动参数
CMD ["-a", "-l"]

#设置卷
VOLUME ["/data", "/var/www"]

#设置子镜像的触发操作
ONBUILD ADD . /app/src
ONBUILD RUN echo "on build executed" >> onbuild.txt
```

Dockerfile 基本语法

- FROM:

指定待扩展的父级镜像（基础镜像）。除注释之外，文件开头必须是一个FROM指令，后面的指令便在这个父级镜像的环境中运行，直到遇到下一个FROM指令。通过添加多个FROM命令，可以在同一个Dockerfile文件中创建多个镜像。

- **MAINTAINER:**
声明创建镜像的作者信息：用户名、邮箱，非必须参数。
- **RUN:**
修改镜像的命令，常用来安装库、安装程序以及配置程序。一条RUN指令执行完毕后，会在当前镜像上创建一个新的镜像层，接下来的指令会在新的镜像上继续执行。RUN 语句有两种形式：
 - **RUN yum update:** 在/bin/sh路径中执行的指令。
 - **RUN ["yum", "update"]:** 直接使用系统调用exec来执行。
 - **RUN yum update && yum install nginx:** 使用&&符号将多条命令连接在同一条RUN语句中。
- **EXPOSE:**
指明容器内进程对外开放的端口，多个端口之间使用空格隔开。
运行容器时，通过设置参数-P（大写）即可将EXPOSE里所指定的端口映射到主机上其他的随机端口，其他容器或主机可以通过映射后的端口与此容器通信。
您也可以通过设置参数-p（小写）将Dockerfile中EXPOSE中没有列出的端口设置成公开。
- **ADD:**
向新镜像中添加文件，这个文件可以是一个主机文件，也可以是一个网络文件，也可以是一个文件夹。
 - 第一个参数：源文件（夹）。
 - 如果是相对路径，必须是相对于Dockerfile所在目录的相对路径。
 - 如果是URL，会将文件先下载下来，然后再添加到镜像里。
 - 第二个参数：目标路径。
 - 如果源文件是主机上的zip或者tar形式的压缩文件，容器引擎会先解压缩，然后将文件添加到镜像的指定位置。
 - 如果源文件是一个通过URL指定的网络压缩文件，则不会解压。
- **VOLUME:**
在镜像里创建一个指定路径（文件或文件夹）的挂载点，这个容器可以来自主机或者其它容器。多个容器可以通过同一个挂载点共享数据，即便其中一个容器已经停止，挂载点也仍可以访问。
- **WORKDIR:**
为接下来执行的指令指定一个新的工作目录，这个目录可以是绝对目录，也可以是相对目录。根据需要，WORKDIR可以被多次指定。当启动一个容器时，最后一条WORKDIR指令所指的目录将作为容器运行的当前工作目录。
- **ENV:**
设置容器运行的环境变量。在运行容器的时候，通过设置-e参数可以修改这个环境变量值，也可以添加新的环境变量。
例如：
docker run -e WEBAPP_PORT=8000 -e WEBAPP_HOST=www.example.com ...
- **CMD:**
用来设置启动容器时默认运行的命令。

- **ENTRYPOINT:**
用来指定容器启动时的默认运行的命令。区别在于：运行容器时添加在镜像之后的参数，对ENTRYPOINT是拼接，CMD是覆盖。
 - 若在Dockerfile中指定了容器启动时的默认运行命令为**ls -l**，则运行容器时默认启动命令为**ls -l**，例如：
 - **ENTRYPOINT ["ls", "-l"]**：指定容器启动时的程序及参数为**ls -l**。
 - **docker run centos**：当运行centos容器时，默认执行的命令是**docker run centos ls -l**
 - **docker run centos -a**：当运行centos容器时拼接了-a参数，则默认运行的命令是**docker run centos ls -l -a**
 - 若在Dockerfile中指定了容器启动时的默认运行命令为**--entrypoint**，则在运行容器时如果需要替换默认运行命令，可以通过添加**--entrypoint**参数来替换Dockerfile中的指定。例如：
docker run gutianlangyu/test --entrypoint echo "hello world"
- **USER:**
为容器的运行及RUN、CMD、ENTRYPOINT等指令的运行指定用户或UID。
- **ONBUILD:**
触发器指令。构建镜像时，容器引擎的镜像构建器会将所有的ONBUILD指令指定的命令保存到镜像的元数据中，这些命令在当前镜像的构建过程中并不会执行。只有新的镜像使用FROM指令指定父镜像为当前镜像时，才会触发执行。
使用FROM以这个Dockerfile构建出的镜像为父镜像，构建子镜像时：
ONBUILD ADD . /app/src：自动执行**ADD . /app/src**

1.4 如何制作镜像包？

使用docker save命令可将容器镜像制作成tar或tar.gz镜像包，具体命令格式如下：

docker save [OPTIONS] IMAGE [IMAGE...]

OPTIONS说明：--output、-o，表示导出到文件。

示例：

```
$ docker save nginx:latest > nginx.tar
$ ls -sh nginx.tar
108M nginx.tar

$ docker save php:5-apache > php.tar.gz
$ ls -sh php.tar.gz
372M php.tar.gz

$ docker save --output nginx.tar nginx
$ ls -sh nginx.tar
108M nginx.tar

$ docker save -o nginx-all.tar nginx
$ docker save -o nginx-latest.tar nginx:latest
```

1.5 SWR 的配额是多少？

容器镜像服务对镜像数量没有配额限制，您可以根据需要上传镜像。

容器镜像服务对单个用户的组织数量限定了配额，如表1-1所示。

表 1-1 容器镜像服务配额

资源类型	配额
组织	5

2 登录问题

首次登录 SWR 服务时，弹窗授权失败可能的原因

问题现象：用户首次登录容器镜像服务SWR时，单击授权弹框的确定按钮后授权失败。

授权说明

使用 SWR(容器镜像) 服务需要授予访问以下云资源的权限:

- 访问 CCE(云容器引擎) 服务
容器镜像服务需要您授权我们访问您的云容器引擎服务集群，以便将您的镜像部署到集群上
- 访问 CCI(云容器实例) 服务
容器镜像服务需要您授权我们访问您的云容器实例，以便将您的镜像部署到实例上

确定

解决办法

- 您的账号未实名认证，请进行实名认证。
- 您的账号欠费，请进行[续费](#)。

长期有效的登录指令与临时登录指令的区别是什么？

- 临时的登录指令代指6个小时后会过期失效，不能再被使用的登录指令。可应用在临时使用，对外单次授权等场景中，对安全性要求较高的生产集群也可通过定时刷新的方式进行使用。
- 长期有效的登录指令有效期为永久。可应用在前期测试、CI/CD流水线及容器集群拉取镜像等场景中。
- 获取了长期有效的登录指令后，在有效期内的临时登录指令仍然可以使用。
- 长期有效的登录指令与临时登录指令均不受限制，可以多台机器多人同时登录。

3 镜像推送

如何通过 API 推送镜像到 SWR?

SWR暂时没有开放镜像上传的API，您可以参考[推送镜像到镜像仓库](#)章节的两种方式：容器引擎客户端的方式或页面上传方式来完成镜像推送。

为什么通过客户端上传和页面上传的镜像大小不一样?

问题描述

假设在本地Docker客户端上制作了一个nginx镜像，版本号为v2.0.0，使用**docker images**命令查询SIZE为**22.8MB**：

```
$ docker images
REPOSITORY          TAG         IMAGE ID      CREATED       SIZE
nginx                v2.0.0      22f2bf2e2b4f 9 days ago    22.8MB
```

1. 通过docker容器引擎客户端（即执行docker push命令）上传该镜像至SWR镜像仓库，查询大小为**9.5MB**。
2. 在本地Docker客户端将镜像打包为tar格式，将nginx.tar下载至本地后，通过页面方式上传至SWR镜像仓库，查询大小为**23.2MB**。

可以发现，通过客户端和页面上传的镜像大小不一样。

原因分析

使用客户端上传的镜像每一层layer都进行了tgz压缩，而页面上传的镜像包是经过docker打包的，每一层layer只进行了打包，没有压缩。所以两种方式上传的镜像大小显示会不一致。

SWR 镜像仓库支持 ARM 镜像上传么?

SWR镜像仓库对镜像的内核架构是没有任何限制的，您上传ARM架构的镜像和上传x86的镜像是没有区别的，直接上传即可。

docker push 使用什么协议将镜像推送到 SWR?

HTTPS协议。

同名同 tag 的镜像上传后会覆盖之前的镜像吗?

会覆盖，保留最新上传的镜像。

SWR 单个 layer 的最大限制是多少？

使用客户端向SWR上传镜像，镜像的每个layer大小不能超过10G。

SWR 公网上传单租户流控限制是多少？

为了确保SWR镜像上传租户间不会互相影响，当前对单个租户流控为镜像层上传20QPS，超过后会被限流，docker会收到503并会自动重试限流拦截的请求。

SWR 支持断点续传镜像文件吗？

暂不支持。

4 镜像拉取

如何通过 API 拉取镜像？

SWR暂不支持通过API拉取镜像。如果要拉取镜像，请通过容器引擎客户端来进行拉取操作。docker容器引擎使用docker pull命令拉取，containerd容器引擎使用ctrctl pull命令拉取。

docker pull 下载的镜像存放在什么地方？

docker pull将镜像下载到本地节点上，您可以通过docker save命令将镜像保存成tar归档文件。

是否支持跨区域下载镜像？

支持。

SWR当前支持跨区域公网下载，请确保获取正确的登录指令。

容器镜像服务有内网吗？通过内网上传下载镜像会不会收取流量费？

如果您的机器与镜像仓库在同一区域，访问仓库是通过内网访问。内网访问不需要额外流量费，因为您购买服务器、EIP的费用已支付。

如果您的机器与镜像仓库在不同区域，通过公网才能访问仓库，下载跨区域仓库的镜像需要节点可以访问公网。

控制台页面的镜像可以下载到本地吗？

控制台页面的镜像不能直接下载至本地，您可以参考以下方法操作：

1. 在镜像详情页获取镜像的下载指令。
2. 在安装了Docker客户端的机器上执行上一步的指令下载镜像。

示例：

```
docker pull swr.ap-southeast-1.myhuaweicloud.com/group/nginx:v1
```

3. 将镜像保存为.tar或.tar.gz格式的文件。

示例：

```
docker save nginx:v1 > nginx.tar
```

4. 下载文件至本地。

拉取镜像慢可能的原因

1. 网络原因。
2. 镜像可能是多层的。
3. 拉取镜像任务可能是串行的，前面的拉取镜像任务未完成的情况下后面的拉取任务将会等待。前面的拉取任务的超时时间是30分钟。

同名同 tag 的镜像重复拉取会覆盖原有旧的镜像吗？

如果镜像manifest未发生变化，不会覆盖；如果镜像manifest发生变化，会覆盖。

5 镜像同步

5.1 是否支持跨区域同步镜像？

- 目前支持跨区域同步镜像的局点有：“华北-北京一”、“华北-北京四”、“华北-乌兰察布一”、“华东-上海一”、“华东-上海二”、“华南-广州”、“西南-贵阳一”、“中国-香港”、“亚太-曼谷”、“亚太-新加坡”、“亚太-雅加达”、“非洲-约翰内斯堡”区域同步镜像。
- 其他区域的用户如果想把最新推送的镜像同步到其他区域镜像仓库内，可以先将镜像下载到本地，然后上传到需要同步区域的镜像仓库中。

5.2 为什么 IAM 用户无法设置镜像同步？

当前仅账号及具有管理员权限的IAM用户才能设置镜像同步。

5.3 为什么已有镜像自动同步不成功？

镜像自动同步是帮助您把最新推送的镜像自动同步到其他区域镜像仓库内，后期镜像有更新时，目标仓库的镜像也会自动更新，但已有的镜像不会自动同步。

已有镜像的同步，需要手动操作：

1. 在镜像详情页面的“镜像版本”页签选择具体的镜像版本后，单击“镜像同步”，如图5-1。

图 5-1 已有镜像的同步



2. 在弹出的对话框中，选择镜像同步的目标区域、目标组织及是否覆盖。

图 5-2 镜像同步

镜像同步

* 目标区域 华北-乌兰察布-二零三

* 目标组织 aaaab

覆盖 ☒

目标组织中具有相同名称或标签的不同镜像将被覆盖，并且最后的更新时间也会发生变化

取消 确定

5.4 他人共享的镜像，是否可以再进行镜像同步？

问题描述：是否可以把账号A共享给账号B的镜像，再同步给账号B下的不同区域？

原因分析：不可以。账号A共享给账号B的镜像，被共享者只有该镜像的只读权限，即只能下该镜像。而同步该镜像需要账号及具有管理员权限的用户才能操作。

5.5 为什么镜像自动同步后在同步区域看不到镜像？

问题描述

镜像自动同步后在同步区域看不到镜像。例如：创建自动同步任务，将regionA的nginx_01镜像同步到regionB区域，同步成功后，在regionB区域看不到nginx_01镜像。

原因分析

当手动同步单个版本镜像或批量手动同步多个版本镜像时，同步才是及时生效的。

如果您创建的是自动同步任务，同步是在下一次版本变动或者有镜像变动时才会同步到另外一个区域，所以当您上传一个新版本nginx_01镜像后，您在regionB区域才能看到新上传的镜像。

6 故障类

6.1 为什么登录指令执行失败？

登录指令执行失败有以下几种情况：

1. 容器引擎未安装正确，报如下所示错误：

“docker: command not found”

解决方法：重新安装容器引擎，安装方法请参见[安装容器引擎](#)。

- 由于容器镜像服务支持18.06及以上版本的Docker容器引擎上传镜像，建议下载对应版本。
- 安装容器引擎需要连接互联网，内网服务器需要绑定弹性公网IP后才能访问。

2. 临时登录指令已过期或登录指令中区域项目名称、AK、登录密钥错误，报如下所示错误：

“unauthorized: authentication required”

解决方法：登录容器镜像服务控制台，在左侧菜单栏选择“我的镜像”，单击右侧“客户端上传”在弹出的页面单击“生成登录指令”。

- a. 获取临时的登录指令：默认进入“临时登录指令”页签，单击 复制登录指令。
- b. 获取长期有效的登录指令：切换到“长期有效登录指令”页签进行获取，具体方法请参见[获取长期有效登录指令](#)。

3. 登录指令中镜像仓库地址错误，报如下所示错误：

“Error logging in to v2 endpoint, trying next endpoint: Get https://{{endpoint}}/v2/: dial tcp: lookup {{endpoint}} on xxx.xxx.xxx.xxx:53 : no such host”

解决方法：

- a. 修改登录指令中的镜像仓库地址。
镜像仓库地址格式：swr.区域项目名称.myhuaweicloud.com，如“华北-北京四”对应的镜像仓库地址为swr.cn-north-4.myhuaweicloud.com。
- b. 获取临时的登录指令：方法请参见[2](#)。

4. **x509: certificate has expired or is not yet valid**

长期有效登录指令中AK/SK被删除导致，请使用有效的AK/SK生成登录指令。

5. **x509: certificate signed by unknown authority****问题原因：**

容器引擎客户端和SWR之间使用HTTPS的方式进行通信，客户端会对服务端的证书进行校验。如果服务端证书不是权威机构颁发的，则会报如下错误：x509: certificate signed by unknown authority

```
[root@luster01-fd0rb ~]# docker login -u cn-north-1@HWUW7N1AIXLW8UGPF -p Scad07ba37ba0b2956c29f5fe1173a5e00f220fa509f7ab14dc85d74956902 registry.cn-north-1.huaweiclouds.com
Error response from daemon: Get https://registry.cn-north-1.huaweiclouds.com/v1/users/: x509: certificate signed by unknown authority
```

解决方法：

如果用户信赖服务端，跳过证书认证，那么可以手动配置Docker的启动参数，配置方法如下：

- CentOS:

修改“/etc/docker/daemon.json”文件（如果没有，可以手动创建），在该文件内添加如下内容：

```
{
  "insecure-registries": ["{镜像仓库地址}"]
}
```

- Ubuntu:

修改“/etc/default/docker”文件，在DOCKER_OPTS配置项中增加如下内容：

```
DOCKER_OPTS="--insecure-registry {镜像仓库地址}"
```

- EulerOS:

修改“/etc/sysconfig/docker”文件，在INSECURE_REGISTRY配置项中增加如下内容：

```
INSECURE_REGISTRY="--insecure-registry {镜像仓库地址}"
```

说明

镜像仓库地址支持域名和IP形式。

- 域名：swr.区域项目名称.myhuaweicloud.com。例如“华北-北京四”的镜像仓库地址为：swr.cn-north-4.myhuaweicloud.com。
- IP：可通过ping镜像仓库地址（域名形式）获取。

配置完成后，执行**systemctl restart docker**重启容器引擎。

6. **denied: Authenticate Error**

用户无编程访问权限，需要使用管理员账号登录IAM服务，单击用户名称进入用户详情页面，单击访问方式参数后面的✎按钮，修改访问方式同时勾选编程访问和管理控制台访问。关于IAM服务如果想了解更多请参见IAM产品介绍。



6.2 为什么使用客户端上传镜像失败？

denied: you do not have the permission

问题现象：使用客户端上传镜像，报如下所示错误：

“denied: you do not have the permission”

问题原因：

- 该组织名已被其他用户注册或当前SWR组织数量已超过配额。
- docker login命令使用IAM用户的AK、SK生成，没有对应组织的权限。

解决方法：

- 该组织名已被其他用户注册时：建议您先创建组织然后再上传镜像，创建组织的方法请参见[创建组织](#)。
- SWR组织数量超过配额时：单个用户的组织数量限制为5个，您可以将镜像上传到已存在的组织下。
- 没有对应组织的权限可通过以下两种方法进行授权：

方法一：本地授权：使用具有SWR管理员权限的用户在组织管理页面授予要上传镜像的用户目标组织的管理权限。授权操作请参见[授权管理](#)

方法二：使用IAM细粒度授权，为用户授予自定义的权限策略，包含镜像推送的权限，具体操作如下：

- a. 登录[管理控制台](#)。
- b. 在页面上方选择“区域”后，单击页面左上方的，选择“管理与监管 > 统一身份认证服务 IAM”。
- c. 选择左侧导航树的“权限管理 > 权限”，单击右上角的“创建自定义策略”填写策略参数。策略名称自定义，策略配置方式选择“JSON视图”。填写策略内容如下（请将namespace1替换为您要上传镜像所到的组织），配置完成后单击“确定”。

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "swr:repo:upload"
      ],
      "Resource": [
        "SWR:*:*:repo:namespace1/*"
      ]
    }
  ]
}
```
- d. 选择左侧导航树的“用户组”，单击您的用户所在的用户组名称进入用户组详情页面，单击“授权”，将3创建的权限添加给用户所在的用户组即可。
- e. 稍等1分钟待权限生效后进行重新上传。

denied: Image organization does not exist, you should create it first

问题现象：使用客户端上传镜像，报如下所示错误：

“denied: Image organization does not exist, you should create it first”

问题原因：docker push命令中的组织名称不存在。

解决方法：请先创建组织，再重新上传。

tag does not exist: xxxxxx 或 An image does not exist locally with the tag: xxxxxx

问题现象：使用客户端上传镜像，报如下所示错误：

“tag does not exist: xxxxxx”

或

“An image does not exist locally with the tag: xxxxxx”

问题原因：上传的镜像或镜像版本不存在。

解决方法：通过docker images查看本地镜像，确认要上传的镜像名称及版本后，重新上传镜像。

name invalid: 'repository' is invalid

问题现象：使用客户端上传镜像，报如下所示错误：

“name invalid: 'repository' is invalid”

问题原因：组织命名或镜像命名不规范。

解决方法：以下分别是组织名（namespace）和仓库名（repository）的命名正则表达式：

namespace: ^([a-z]+(?:_(?:[a-z0-9]+)?)?)\$, 长度范围为：1-64；

repository: ^([a-z0-9]+(?:_(?:[a-z0-9]+)?)?)\$, 长度范围为：1-128。

您可以按照上述命名规范，重新指定上传的组织 and 镜像名称。

偶现推送镜像超时

问题现象：偶现推送镜像超时

问题原因：您在国内机器往国外推镜像，网速慢，偶现连接失败。

6.3 为什么通过页面上传镜像失败？

SWR对镜像的命名和地址有严格的规范。如果镜像的命名不规范或镜像地址不规范都会导致镜像上传失败。

镜像格式不合法或鉴权失败

问题现象：通过页面上传镜像，出现“镜像格式不合法”或“鉴权失败”的报错。

问题原因：

1. 上传时长超过15分钟，前端token过了有效期。

2. 镜像地址不规范，导致上传失败。

镜像地址各个部分的含义如下，最后的tag（版本号）可省略，如果省略则表示latest版本，其余部分均不可省略，且不可多余。

样例：swr.regionid.*****.com/repo_namespace/repo_name:tag

- swr.regionid.*****.com为容器镜像服务的镜像仓库地址。
- repo_namespace为组织名称，命名正则表达式为 $^([a-z]+(?:[:_]|_|[-])*[a-z0-9]+)+?)$$ ，长度范围为：1-64。
- repo_name:tag为镜像名称和版本号，镜像命名正则表达式为 $^([a-z0-9]+(?:[:_]|_|[-])*[a-z0-9]+)+?)$$ ，长度范围为：1-128。

您可以将镜像解压，打开文件manifest.json文件查看RepoTags字段的值是否符合上述规范。



解决方法：

1. 针对token过期的场景，建议使用[容器引擎客户端](#)进行上传。
2. 针对镜像地址不规范的场景，请按照命名规范，重新给镜像打tag，然后使用docker save命令保存镜像，然后再使用页面上上传。

须知

SWR判定镜像名是否合法不是以用户上传镜像时的文件名作为依据，而是依据镜像包中的repositories和manifest.json文件。

一直卡在上传界面直到超时

问题现象：通过页面上传镜像，一直卡在上传界面直到超时。

问题原因：

- 镜像命名不规范，导致上传失败。
- 页面上传走公网链路，可能会因为网络不稳定导致传输异常。

解决方法：

- 您可以按照镜像命名规范修改镜像名称后，重新上传镜像。
- 建议更换网络环境，或者通过容器引擎客户端上传。

6.4 为什么 docker pull 指令执行失败？

x509: certificate signed by unknown authority

问题现象：使用docker pull拉取镜像，报错“x509: certificate signed by unknown authority”。

问题原因：

- 容器引擎客户端和SWR之间使用HTTPS的方式进行通信，客户端会对服务端的证书进行校验。如果客户端安装的根证书不完整，会报如下错误：“x509: certificate signed by unknown authority”。
- 容器引擎客户端配置了Proxy导致。

解决方法：

- 如果您信赖服务端，跳过证书认证，那么可以手动配置容器引擎的启动参数，配置如下（其中地址配置成需要的即可，选择一个配置即可）：
 - /etc/docker/daemon.json（如果没有可以手动创建），在该文件内添加如下配置（注意缩进，2个空格）：

```
{  
  "insecure-registries":["镜像仓库地址"]  
}
```

- /etc/sysconfig/docker:
INSECURE_REGISTRY='--insecure-registry=镜像仓库地址'

添加配置后执行如下命令重启：**systemctl restart docker**或**service restart docker**。

- 执行**docker info**命令，检查Proxy配置是否正确，修改为正确的Proxy配置。

Error: remote trust data does not exist

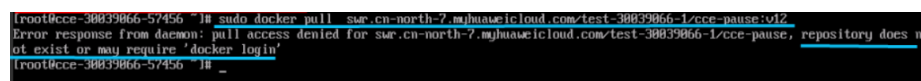
问题现象：使用docker pull拉取镜像，报错“Error: remote trust data does not exist”。

问题原因：客户端开启镜像签名验证，而镜像没有镜像签名层。

解决方法：查看“/etc/profile”文件中的环境变量是否设置了**DOCKER_CONTENT_TRUST=1**，如果设置了，请将其改为**DOCKER_CONTENT_TRUST=0**，然后执行**source /etc/profile**生效。

pull access deny

问题现象：拉取镜像的时候报错，详细报错见下图。



```
[root@cce-30839866-57456 ~]# sudo docker pull swr.cn-north-7.myhuaweicloud.com/test-30839866-1/cce-pause:012  
Error response from daemon: pull access denied for swr.cn-north-7.myhuaweicloud.com/test-30839866-1/cce-pause, repository does not exist or may require 'docker login'  
[root@cce-30839866-57456 ~]#
```

问题原因：您的账号没有拉取镜像的权限swr:repo:download或对应组织的权限。

解决办法：

给您的用户添加镜像拉取的权限以及对应组织的权限，可通过以下两种方法进行授权：

方法一：本地授权：使用具有SWR管理员权限的用户在组织管理页面授予要拉取镜像的用户目标组织的管理权限。授权操作请参见[授权管理](#)

方法二：使用IAM细粒度授权，为用户授予自定义的权限策略，包含镜像拉取的权限，具体操作如下：

步骤1 登录[管理控制台](#)。

步骤2 在页面上方选择“区域”后，单击页面左上方的，选择“管理与监管 > 统一身份认证服务 IAM”。

步骤3 选择左侧导航树的“权限管理 > 权限”，单击右上角的“创建自定义策略”填写策略参数。策略名称自定义，策略配置方式选择“JSON视图”。填写策略内容如下（请将namespace1替换为您要拉取镜像所在的组织），配置完成后单击“确定”。

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "swr:repo:download"
      ],
      "Resource": [
        "SWR:*:repo:namespace1/*"
      ]
    }
  ]
}
```

步骤4 选择左侧导航树的“用户组”，单击您的用户所在的用户组名称进入用户组详情页面，单击“授权”，将[步骤3](#)创建的权限添加给用户所在的用户组即可。

步骤5 稍等1分钟待权限生效后进行重新拉取。

----结束

6.5 如何解决内网下载镜像失败？

内网下载镜像失败基本上都是由于DNS配置问题导致的，可以采用以下两种方法进行修改。

- **方法一：**

编辑“/etc/resolv.conf”文件，在/etc/resolv.conf中新增一个内网DNS服务器地址，具体的内网DNS地址请参见[华为云提供的内网DNS地址是多少？](#)。

说明

新增的DNS服务器地址必须位于所有原有的DNS服务器地址之前。

DNS配置操作在保存“/etc/resolv.conf”文件的修改操作后立即生效。

- **方法二：**

“/etc/resolv.conf”文件的修改操作在弹性云服务器重启后会失效，需要重新进行配置。如果用户不希望每次重启弹性云服务器后都重新配置DNS，可以按如下步骤修改虚拟私有云的子网信息，将DNS服务器地址添加到弹性云服务器对应的子网中。

须知

对虚拟私有云的子网信息的修改会影响所有使用该子网创建的弹性云服务器。

- a. 登录管理控制台，选择“网络 > 虚拟私有云”。
- b. 参考[怎样切换内网DNS?](#) 修改VPC子网的DNS服务器地址。
- c. 重启弹性云服务器，查看“/etc/resolv.conf”文件的内容，确认其中包含待配置的DNS服务器地址，并且新增的DNS服务器地址位于其他DNS服务器地址之前。

6.6 为什么创建组织失败？

问题现象：创建组织失败，页面提示该组织已经存在，但在“组织管理”页面没有查询到该组织。

问题原因：组织名称全局唯一，即当前区域下，组织名称唯一。创建组织时如果提示组织已存在，可能该组织名称已被其他用户使用。

解决办法：请重新设置一个组织名称。

7 其他

7.1 为什么 CCE 工作负载拉取 SWR 内的镜像异常，且提示为“未登录”？

当CCE工作负载无法正常拉取SWR的镜像，且提示“未登录”时，请排查该工作负载的yaml文件中是否存在“imagePullSecrets”字段，且name参数值需固定为default-secret。

示例：

```
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  name: nginx
spec:
  replicas: 1
  selector:
    matchLabels:
      app: nginx
  strategy:
    type: RollingUpdate
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - image: nginx
          imagePullPolicy: Always
          name: nginx
          imagePullSecrets:
            - name: default-secret
```

7.2 SWR 私有镜像最多可以共享给多少个租户？

500个。

7.3 为什么通过客户端和页面上传的镜像大小不一样？

问题描述

假设在本地Docker客户端上制作了一个nginx镜像，版本号为v5，使用**docker images**命令查询SIZE为**22.8MB**：

```
$ docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
nginx                v5                 22f2bf2e2b4f       9 days ago         22.8MB
```

1. 通过客户端（即执行**docker push**命令）上传该镜像至SWR镜像仓库，查询大小为**9.5MB**。

☐ 镜像版本	大小	下载指令	更新时间
☐ v5	9.5 MB	docker pull swr.cn-north	2024/05/10 18:10:16 GMT+08:00

2. 在本地Docker客户端将镜像打包为tar格式，将nginx.tar下载至本地后，通过页面方式上传至SWR镜像仓库，查询大小为**23.2MB**。

☐ 镜像版本	大小	下载指令	更新时间
☐ v5	23.2 MB	docker pull swr.cn-north	2024/05/11 09:29:01 GMT+08:00

可以发现，通过客户端和页面上传的镜像大小不一样。

原因分析

使用客户端上传的镜像每一层layer都进行了tgz压缩，而页面上传的镜像包是经过docker打包的，每一层layer只进行了打包，没有压缩。所以两种方式上传的镜像大小显示会不一致。

7.4 控制台页面的镜像可以下载到本地吗？

控制台页面的镜像不能直接下载至本地，您可以参考以下方法操作：

1. 在镜像详情页获取镜像的下载指令。
2. 在安装了Docker客户端的机器上执行上一步的指令下载镜像。

示例：

```
docker pull swr.ap-southeast-1.myhuaweicloud.com/group/nginx:v1
```

3. 将镜像保存为.tar或.tar.gz格式的文件。

示例：

```
docker save nginx:v1 > nginx.tar
```

4. 下载文件至本地。