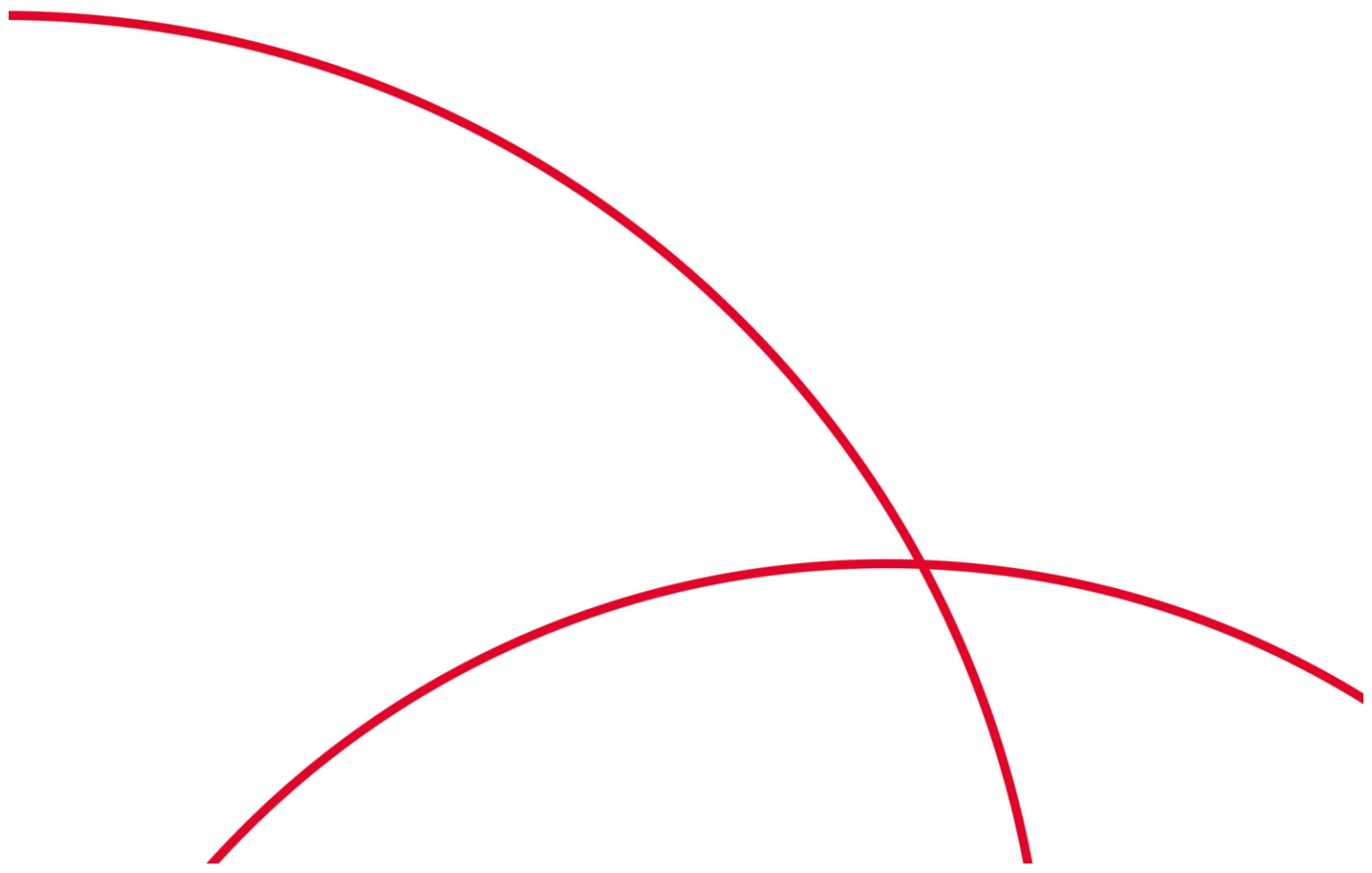




容器镜像服务

用户使用指南

天翼云科技有限公司



目录

1	产品简介.....	3
1.1	产品定义.....	3
1.2	产品优势.....	3
1.3	功能特性.....	4
1.4	应用场景.....	5
1.5	使用配额.....	7
2	快速入门.....	8
2.1	上传并使用私有镜像.....	8
3	用户指南.....	11
3.1	镜像构建.....	11
3.2	镜像清理.....	14
4	常见问题.....	16
4.1	计费类.....	16
4.2	购买类.....	16
4.3	操作类.....	17
4.4	镜像管理类.....	23

1 产品简介

1.1 产品定义

容器镜像服务（CT-CCR, Cloud Container Registry）是一种支持容器镜像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。您可以通过Docker CLI上传、下载Docker镜像，并对接天翼云容器服务部署应用，也可单独作为容器镜像仓库使用。

基本概念

- 项目（Project）

项目定义了属性与镜像存储空间。依据属性划分，用户可以创建属性为公有的项目，或者属性为私有的项目，在项目内创建的镜像仓库会继承项目的属性，在仓库下的镜像属性与镜像仓库相同。同时项目还定义了您存储镜像的空间范围，如果超出空间限制，将会上传失败。

- 镜像仓库（Repository）

库是用来保存所有创建好的镜像统一存储的位置，用户在创建容器应用前，需要将应用所需的镜像上传到镜像仓库。仓库一般分为两种属性，公有和私有两种：

公有：任何租户、用户均可以下载。

私有：仅当前租户或租户下的用户可用。

- 镜像（Image）

镜像是一个模板，用于打包应用程序及其依赖的环境。在进行应用部署容器化的过程中，需要指定镜像。

1.2 产品优势

简单易用

界面化操作简单，提供仓库创建及镜像生命周期管理功能，无需用户再自行搭建和运维仓库，即可实现快速推拉取容器镜像，提高业务容器化效率，降低用户运维成本。

支持多地域资源池创建，完美整合天翼云容器引擎、Serverless容器服务，实现平台同步更新镜像，一键

快速实现业务容器化部署。

海量镜像

镜像中心为用户提供了丰富的官方镜像资源，覆盖操作系统、微服务中间件等近百项常用容器镜像，保障了安全可靠的镜像来源，用户无需在准备镜像上耗费时间。

容器镜像服务的镜像共享功能，建立了容器镜像传递流通的良好通道，为用户提供了更多的镜像源选择。

用户不仅可以共享个人镜像，同时也可以使用他人共享的镜像，方便企业团队成员间镜像文件的传递。

安全可靠

云平台镜像的传输过程采用了HTTPS安全协议，通过SSL+HTTP实现加密传输、身份认证。可防止数据在传输过程中被窃取、改变，提升了用户访问仓库，传输镜像的安全水平。为用户提供了可靠的容器传输通道，满足具有信息安全保密的业务，如金融、电信等行业。

1.3 功能特性

多地域云上仓库

- 多地域

容器镜像服务支持中卫2、晋中等多地域开通，为用户提供了一套安全简便的镜像存储、管理平台。用户借助CCR，即可省去搭建、运维本地镜像仓库的烦恼，可以随时随地登陆服务完成镜像操作。

- 多镜像

平台支持用户自定镜像，镜像中心镜像、他人共享镜像，镜像种类丰富，满足用户各类需求。镜像中心由天翼云平台提供多类型多版本官方容器镜像，如httpd、redis、centos、golang等。同时，用户可查看、使用天翼云其它用户公开方向镜像，用户也可通过dockerfile构建自定义镜像，并上传平台，进行分享。

- 多方面安全保障

一键公私有项目设置，镜像采用https传输协议保障数据安全传输。可查询项目镜像操作日志，快速定位可疑操作。

镜像极速下拉

容器镜像服务与天翼云容器服务平台，包括容器引擎CCE、Serverless容器服务完美整合，平台同步更新

镜像资源，根据页面操作，快速下拉镜像到用户业务集群，完成从选择镜像到应用创建的一键化部署流程。用户无需在镜像准备、多节点镜像上传操作上耗费时间，通过CCR+CCE/CSK的搭配使用，实现镜像的稳定推拉。

镜像仓库管理

- 支持用户管理镜像仓库

包括创建、设置属性与删除，创建镜像仓库的过程由创建项目与创建仓库两步完成，首先创建项目并完成属性设置（公有或私有），在项目下创建仓库时继承该属性。

- 支持私有镜像管理

包括上传和下载等基本操作，用户可从将本地私有镜像或容器引擎上的镜像上传至镜像仓库中，以创建容器应用。

镜像仓库不扫描用户上传的镜像，不负责对用户上传的镜像进行安全性验证。上传的镜像中请不要包含未加密的口令，密码等隐私信息，以避免隐私泄露。下载公有镜像时，应确定数据来自于可信的仓库源，以避免下载到恶意软件。

如果使用自定义镜像，请确保镜像来源可信，不在容器镜像内安装不必要的软件，在升级时使用安全补丁升级镜像。使用第三方镜像，造成的后果（例如，环境不可用）用户需自己承担。

1.4 应用场景

用户需要自建本地镜像仓库，但涉及到组件安装、权限配置、集群整合等复杂的搭建流程。同时，后期仓库和镜像的管理将涉及到大量麻烦、耗时的后台操作，需要持续长期的进行维护，时间、人力成本耗费高。天翼云容器镜像服务为用户提供云上镜像仓库及全面的镜像管理平台，帮助用户快速完成仓库搭建，快捷实现镜像上传、下拉、删除等基本操作，大幅度提升镜像管理效率，用户无需维护仓库，由平台托管，大幅度降低人力、时间成本。同时，容器镜像服务提供的一键公私有项目设置，镜像HTTPS协议安全传输，可查操作日志能力，保障了用户数据安全。

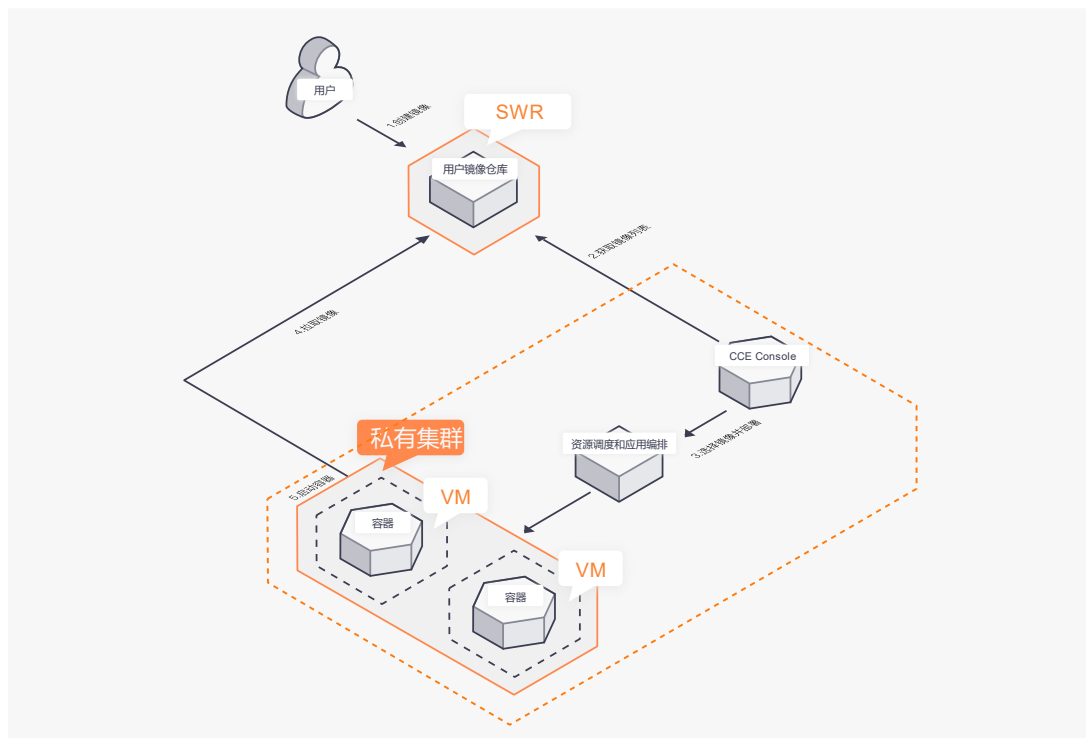
私有仓库搭建、镜像管理场景

应用：政务、医疗、教育、金融、工业制造、等大型央企国企客户

场景特点：具有搭建、管理企业或个人所有的镜像仓库需求，存在对镜像具有频繁操作，考虑降低用人成

本的用户。

使用场景：政务、医疗、教育、金融、工业制造、等大型央企国企用户：容器镜像服务能帮助用户快速建企业私有的镜像仓库，实现镜像的快件管理，并能保证用户仓库的安全性及稳定性，为用户降低成本



用户需要容器化部署自己的业务，首先在本地完成镜像上传，当集群节点数多，单个节点涉及业务种类多，且业务更新频繁时，后台镜像操作将变得繁琐费时，业务容器化效率低。需要简化操作，提升业务部署效率。容器镜像服务，为用户提供了健全的云上仓库，用户可随时随地进行镜像操作，并实现一次镜像上传，多节点快速下拉的一劳永逸的效果。除此以外，用户还可以免费享受天翼云专业的云上存储能力，以及丰富的Docker镜像源，和开放的镜像共享平台。同时，容器镜像服务平台完美与天翼云容器服务缝合，用户可搭配容器引擎和Serverless容器服务，一键实现业务容器化部署，效率翻倍。

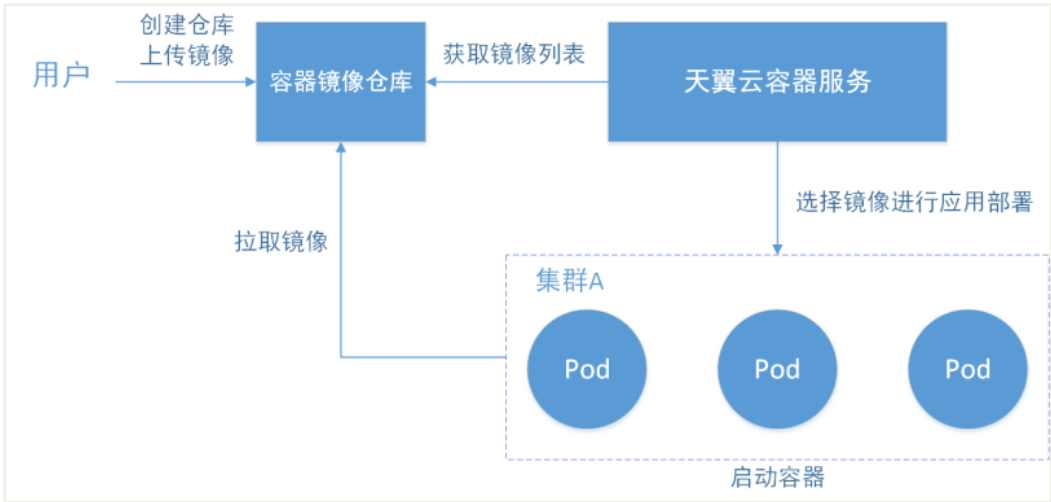
镜像统一下发

应用：政务、医疗、教育、金融、工业制造、等大型央企国企客户

场景特点：具有大规模集群，业务部署流程中镜像上传耗时长，效率低下。或已使用天翼云容器服务，建议搭配容器镜像服务一起使用。

使用场景：政务、医疗、教育、金融、工业制造、等大型央企国企客户:提供镜像统一下发的能力，实现应

用快速容器化，并提供丰富的镜像资源与免费的存储能力，帮助用户提升业务部署效率。



1.5 使用配额

容器镜像服务对单个用户创建的项目数量、项目内镜像存储空间进行了限制，具体如下表所示:

资源类型	释义	默认配额	可调配额
项目	单用户可创建的项目数量	10G存储空间，当前用户可创建属性为公有的项目与私有的项目各1个。	无法调整
镜像仓库	单个项目可创建的镜像仓库数量	数量无限制	不涉及
镜像	单个镜像仓库可上传的镜像数量	数量无限制，全量镜像空间不超过项目存储空间	不涉及

2 快速入门

2.1 上传并使用私有镜像

操作须知

- 1、在使用本产品之前，您需要先注册天翼云门户的账号。获取账号后，通过控制台总览，找到容器镜像服务，点击进入容器镜像服务控制台首页，即可开启操作。
- 2、在使用本产品之前，您需要在本地或申请的天翼云主机ECS上安装Docker。



操作步骤

- 创建项目

1、进入容器镜像服务平台，点击【我的项目】>点击【创建项目】，进入创建项目界面。

2、填写项目名称，并设置项目的属性，点击【创建】完成项目建立，点击【取消】则返回。

其中，项目属性包括公有或私有，这决定着在这个项目内镜像仓库的属性，决定着您上传的私有镜像可用范围，参数说明如下所示：

参数	参数含义
私有	镜像只有自己可在创建应用过程使用
公有	镜像全网容器用户均可在创建应用过程使用

● 创建镜像仓库

点击左侧导航栏【我的项目】，进入项目列表>点击已创建好的项目，进入仓库列表>点击【创建仓库】，进入仓库创建界面，输入仓库名称，点击【创建】>返回仓库列表，可查看创建的仓库。

● 上传镜像

上传镜像可通过【总览】界面的【上传镜像】，或【我的镜像】的【上传镜像】两个操作入口进入，点击【上传镜像】即可查看上传镜像操作步骤的说明，按照说明中的操作步骤即可完成镜像上传（详细示例请见4.1.1客户端上传私有镜像）。

● 应用部署

用户可通过拉取镜像在本地实现应用部署，或者通过直接使用容器引擎、serverless容器服务平台实现集群中的应用部署。

1、本地拉取镜像

(1) 登陆本地环境，若已购买天翼云云主机产品，可通过【远程登陆】进入环境。

(2) 确保本地已安装docker，且网络环境的稳定与联通。通过以下命令可实现镜像拉取：

```
docker login -u {username} -p {pwd} zhongwei2.ccr.ctyun.cn
```



```
docker pull zhongwei2.ccr.ctyun.cn/{项目名称}/{仓库名}/{镜像名}:{标签名}
```

其中用户名及密码等信息，可在容器服务镜像平台上进行查看。

2、集群应用部署

用户可通过容器引擎或Serverless容器服务中的【创建应用】及【应用升级】操作，完成镜公共镜像、私有镜像、共享镜像的选择及应用的部署，容器引擎及serverlerss服务的具体使用说明，请见：

<https://www.ctyun.cn/h5/help2/10000068>

<https://www.ctyun.cn/document/10015676>

点击【创建应用】>进入【容器设置】页面>点击【选择镜像】,进入镜像选择页面>点击【我的镜像】,通过对项目名称、镜像仓库的逐步确定，可以完成对于用户私有镜像的选择。

3 用户指南

3.1 镜像构建

3.1.1 客户端镜像构建

容器镜像服务提供了容器镜像管理的平台，具备安全、稳定、高效等特性。用户可使用Dockerfile文件完成自定义容器镜像的制作，随后再将其从上传到容器镜像仓库，进行统一的镜像管理及应用部署。本文将介绍如何使用容器镜像服务在客户端完成镜像构建流程，在使用本产品之前，您需要在本地或申请的天翼云主机ECS环境上完成Docker的安装。

3.1.1.1 镜像构建

平台目前还未提供页面形式的镜像构建能力，用户可在本地通过Dockerfile文件完成容器镜像的构建后再进行镜像上传。Dockerfile 是一个用来构建镜像的文本文件，文本内容包含了一条构建镜像所需的指令和说明。下文将简单介绍如何通过Dockerfile完成镜像构建。

操作步骤

1、用户可再本地执行以下命令创建Dockerfile文件或下载相关Dockerfile文件进行编辑

```
vi Dockerfile
```

2、编写Dockerfile文件

部分关键字说明：

FROM：定制的镜像都是基于 FROM 的镜像。

COPY：复制指令，从上下文目录中复制文件或者目录到容器里指定路径。

EXPOSE：声明端口。

CMD：指定容器运行时的默认命令。

RUN：用于执行后面跟着的命令行命令。

示例：

```
FROM centos:7

RUN yum -y --setopt=tsflags=nodocs update && \
    yum -y --setopt=tsflags=nodocs install httpd && \
    yum clean all

EXPOSE 80

ADD run-httpd.sh /run-httpd.sh
RUN chmod -v +x /run-httpd.sh

CMD ["/run-httpd.sh"]
```

3、通过命令构建镜像

执行以下命令，完成用户容器镜像的构建：

`docker build -t 镜像名称:镜像标签 .`

-t 为镜像命名

. 代表本次执行的上下文路径

4、查看镜像

执行以下命令，查看制作的Docker镜像：

`docker images`

3.1.1.2 创建项目

操作步骤

1、进入容器镜像服务平台，点击【我的项目】>点击【创建项目】，进入创建项目界面。

2、填写项目名称，并设置项目的属性，点击【创建】完成项目建立，点击【取消】则返回。

其中，项目属性包括公有或私有，这决定着在这个项目内镜像仓库的属性，决定着您上传的私有镜像可用范围，参数说明如下所示：

参数	参数含义
----	------

私有	镜像只有自己可在创建应用过程使用
公有	镜像全网容器用户均可在创建应用过程使用

3.1.1.3 创建镜像仓库

操作步骤

点击左侧导航栏【我的项目】，进入项目列表>点击已创建好的项目，进入仓库列表>点击【创建仓库】，进入仓库创建界面，输入仓库名称，点击【创建】>返回仓库列表，可查看创建的仓库。

※注：镜像仓库的属性（公有/私有）与所在项目保持一致

3.1.1.4 上传镜像

通过客户端上传私有镜像，指的是使用命令将镜像上传到容器镜像服务的镜像仓库。上传镜像可通过【总览】界面的【上传镜像】，或【我的镜像】的【上传镜像】两个操作入口进入，以下选择宁夏-中卫2资源池，以nginx:1.10镜像为例，介绍如何通过客户端上传私有镜像。

前置工作

- 1、已安装Docker客户端，并确定Docker服务已启动，可联通公网；
- 2、请确保镜像的正确性，能够成功后台启动；
- 3、已创建名为nginx的容器镜像仓库，操作过程请参见步骤三创建容器镜像仓库。

操作步骤

- 1、点击【上传镜像】，可获得镜像仓库登陆指令，其中用户名可直接获取，密码可通过查看密码获取；
- 2、打开本地环境，在命令行内输入步骤1获取的docker login命令，系统将返回successful的提示词，表示仓库已登陆成功。
- 3、使用以下命令查看本地需要上传的镜像，并获取相应的镜像。

docker images

- 3、使用命令为需要上传的镜像打上tag，docker tag命令可通过步骤1获取，以nginx: 1.10 为例，输入命令

```
docker tag nginx zhongwei2.ccr.ctyun.cn/private/nginx/nginx:1.10
```

其中，nginx为容器镜像本地的名称，也可以使用镜像ID。zhongwei2.ccr.ctyun.cn为容器镜像服务资源池仓库地址，可在步骤1上传镜像步骤中查看，nginx:1.10为用户为镜像的新命名及标签（版本号）。

*注意：请提前确保容器镜像服务平台，已完成private项目及nginx仓库的创建，否则镜像上传将发生错误。

4、通过docker push命令将打完tag的镜像上传到镜像仓库中，docker push命令可通过步骤1获取，以nginx: 1.10 为例，输入命令

```
docker push zhongwei2.ccr.ctyun.cn/private/nginx/nginx:1.10
```

终端显示如下信息，表明 push 镜像成功。

```
6d6b9812c8ae: Pushed
695da0025de6: Pushed
fe4c16cbf7a4: Pushed
1.10: digest: sha256:eb7e3bbd8e3040efa71d9c2cacfa12a8e39c6b2ccd15eac12bdc49e0b66cee63
size: 948
```

5、回到容器镜像服务平台，进入对应项目仓库，执行刷新操作后可查看到上传的镜像。

3.2 镜像清理

用户可登陆容器镜像服务平台，快速完成对容器镜像的删除，删除的镜像数据将无法找回，请用户谨慎操作。

3.2.1 删除镜像仓库

前置工作

删除镜像仓库前，需要您将该镜像仓库下的镜像全部清除。

操作步骤

- 1、 进入容器镜像服务页面，点击导航栏【我的项目】>点击进入待删除镜像仓库所在的项目
- 2、 选择待删除的镜像仓库，在操作列点击【删除】，将弹出删除提示框
- 3、 点击提示框中的【确定】按键，完成镜像仓库删除。

3.2.2 删除镜像



镜像删除可以通过【我的项目】入口，找到镜像所在的仓库，进入仓库进行删除，也可以通过【我的镜像】进行直接删除。请注意删除的镜像无法找回，请您谨慎操作。

操作步骤

- 1、点击【我的镜像】，可看到全量容器镜像。对需要删除的镜像，点击操作列的【删除】。如有必要，可在上方项目、仓库内进行筛选，或按镜像名称搜索，以便快速定位需要删除的镜像。
- 2、弹出删除确认提示框，点击【确定】，完成删除。

4 常见问题

4.1 计费类

Q：容器镜像服务是否收费？

A：容器镜像服务目前暂不收费，可以免费使用。

说明：具体收费方式可参考【计费说明】部分。

4.2 购买类

Q：什么是项目？

A：项目定义了属性与镜像存储空间。用户需要首先建立项目，才能在项目下建立镜像仓库，并上传镜像。

说明：依据属性划分，用户可以创建1个属性为公有的项目，以及1个属性为私有的项目，在项目内创建的镜像仓库会继承项目的属性，在仓库下的镜像属性与镜像仓库相同。同时项目还定义了您存储镜像的空间范围，如果超出空间限制，将会上传失败。

Q：容器镜像服务最多能存储多少个镜像？

A：容器镜像服务对存储的像数量没有限制，但是镜像所占用空间之和无法超过镜像所在项目的总空间，您可以根据需要合理安排项目空间的使用。

说明：详情参考【使用限制】部分。

Q：容器镜像服务的配额是多少？

A：容器镜像服务仅限制用户创建项目的数量，用户最多可创建属性为公有的项目1个，属性为私有的项目1个，且无法调整。镜像仓库与镜像均无数量限制，镜像的所占空间之和不超过项目的存储空间。

说明：详情参考【快速入门】【使用限制】部分。

Q：docker push将镜像推送到容器镜像服务使用的什么协议？

A：docker push将镜像推送到容器镜像服务使用的是HTTPS协议。Https是一种安全超文本传输协议，通

过计算机网络进行安全通信的传输协议，HTTPS 利用 SSL/TLS 来加密数据包，提供身份认证、保护交换数据的隐私与完整性。

说明：具体可查看【产品优势】

4.3 操作类

Q：如何在容器引擎中使用？

A：用户再容器引擎产品中创建应用与应用升级的过程中，选择镜像时会看到上传过的私有镜像。

说明：容器引擎创建应用操作说明详见：<https://www.ctyun.cn/h5/help2/10000068/10001129>

Q：用户如何制作容器镜像？

A：自己制作容器镜像，主要有两种方法：

方法一：制作快照方式获得镜像

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，然后整体制作镜像。如果后续镜像没有变化，可采用方法一制作镜像。



操作步骤：

- 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，使用以下命令进入容器。

```
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
```

注意：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.使用以下命令，制作镜像。

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

-a：提交的镜像作者。

container-id：操作步骤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。

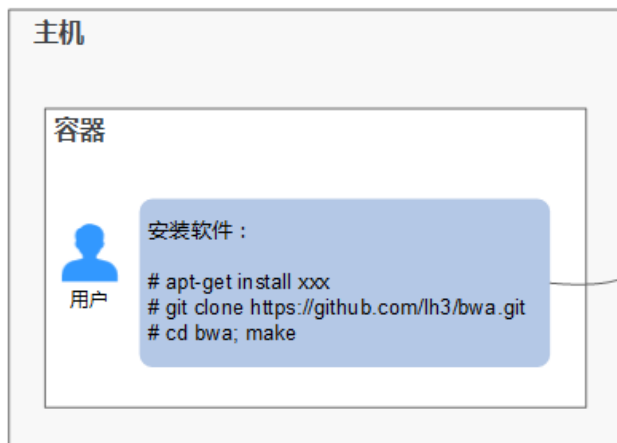
tsj/image:tag：仓库名/镜像名:TAG名。

6.执行docker images命令可以查看到制作完成的容器镜像

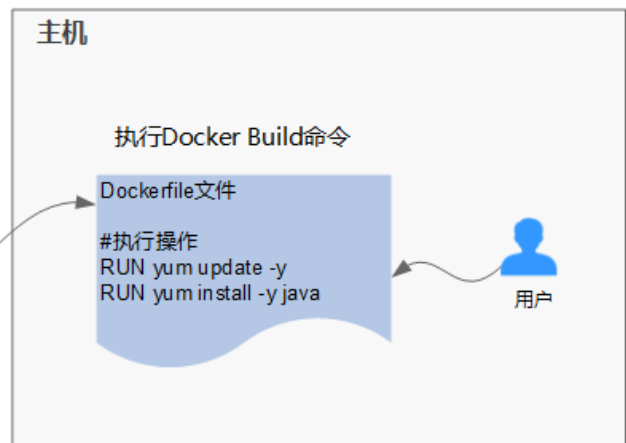
方法二：使用Dockerfile方式构建

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。Dockerfile方式构建镜像是将软件安装的流程写成DockerFile，使用Docker build构建容器镜像，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 tsj/image:tag.命令`（命令中“.”表示DockerFile文件的路径），自动完成镜像制作。

方式一



方式二



以下给出简单的DockerFile书写示例，注意如果依赖外部网络，请并保证网络可用。

```
#Version 1.0.1
```

```
FROM centos:latest
```

```
MAINTAINER ***u "***u@163.com"
```

```
#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 excuted" >> onbuild.txt

下表列出了DockerFile中的常用关键字

关键字	释义
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，接下来对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路径中执行的指令命令。 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，也可以使一个文件夹。 ● 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： <pre>docker run -e WEBAPP_PORT=8000 -e WEBAPP_HOST=www.example.com ...</pre>
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 <pre>docker run centos ls -l -a</pre> ● 若在DockerFile中指定了指定了容器启动时的默认运行命令为--entrypoint，则在运行容器时若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

Q：如何制作镜像压缩包？

A：使用docker save命令可将容器镜像制作成tar或tar.gz文件压缩包，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
```

说明：本产品不支持上传镜像压缩包，因此不建议您使用这种方法制作并上传。

4. 4 镜像管理类

Q：为什么创建项目时，提示项目已存在？

A：容器镜像服务中项目名称是所有账号共用的，比如A用户创建了“projectA”，B用户就没法使用“projectA”。因此创建项目时如果提示项目已存在，可能该项目名称已被其他用户使用，请重新设置一个项目名称。

说明：推荐用户使用实际的项目名称为项目命名，方便对镜像进行合理分类与管理。

Q：为什么使用客户端上传镜像失败？

A：在使用docker push命令上传镜像的过程中，可能会出现下列失败情况，下文列出问题现象与解决方法

问题现象1：提示 denied: you do not have the permission

引发原因：push的路径中，镜像服务域名、项目名称、仓库名称存在问题，均会引发此问题。

解决方法：请用户仔细核对用户、密码、镜像仓库地址等登录信息是否正确

问题现象2：提示 tag does not exist: xxxx 或 an image does not exist locally with the tag:

xxxx

问题原因：上传的镜像或镜像版本不存在。

解决方法：检查镜像打tag的路径是否存在问题，或镜像名称是否正确，否则重新执行镜像上传步骤。

说明：若已上解决方法任然无法解答您的疑问，你可以提交问题详细描述及截图到共但系统或社区，进行询问咨询。

Q：如何通过页面下载容器镜像？

A：目前容器镜像服务不支持通过界面直接下载容器镜像，您可以通过docker客户端使用如下命令下载镜像。

docker pull {节点域名}/[项目名称]/[仓库名称]/[镜像名称:版本名称]

以中卫2为例：

docker pull zhongwei2.ccr.ctyun.cn/[项目名称]/[仓库名称]/[镜像名称:版本名称]

域名可在【上传镜像】说明中获得。

说明：用户拉取的镜像将存放到本地环境，用户可通过docker images命令查看镜像，并通过docker save命令将镜像保存成tar归档文件。