

SWR 快速入门

文档版本 01
发布日期 2026-09-09



版权所有 © 华为云计算技术有限公司 2026。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。

华为云计算技术有限公司

地址：贵州省贵安新区黔中大道交兴功路华为云数据中心 邮编：550029

网址：<https://www.huaweicloud.com/>

目 录

1 镜像的整个生命周期.....	1
2 使用云容器引擎 CCE 拉取镜像并部署应用.....	10

1 镜像的整个生命周期

入门指引

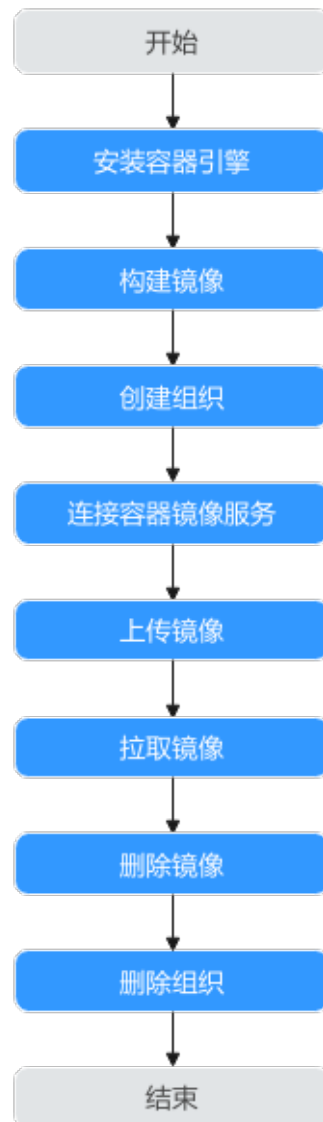
容器镜像服务是一种支持容器镜像全生命周期管理的服务，提供简单易用、安全可靠的镜像管理功能，帮助用户快速部署容器化服务。本文档以一个nginx镜像为例，帮助您学习如何安装容器引擎并构建镜像，以及如何使用容器引擎客户端推送镜像到容器镜像仓库。

说明

- 推送镜像到容器镜像仓库支持两种方式：使用容器引擎客户端和console页面，console页面上传请参见[页面上传](#)。
- 暂未开放镜像推送的API。

您将按以下流程学习如何使用容器镜像服务。

图 1-1 入门流程



准备工作

在使用容器镜像服务前，您需要完成注册华为云的准备工作。

如果您还没有华为云账号，请参考以下步骤创建。

1. 打开<https://www.huaweicloud.com/>，单击“注册”。
2. 根据提示信息完成注册，详细操作请参见[注册华为账号并开通华为云](#)和[个人实名认证](#)。注册成功后，系统会自动跳转至您的个人信息界面。

本文以一个nginx镜像为例，讲述根据该应用编写Dockerfile文件构建镜像并推送至容器镜像服务的操作。您可以编写一个Dockerfile文件，以nginx为基础镜像，来构建一个新nginx容器镜像。

一、安装容器引擎

首先，您需要准备一台虚拟机并安装docker容器引擎，请确保容器引擎为18.06及以上版本。

在安装容器引擎前，请了解容器引擎的基础知识，具体请参见[Docker Documentation](#)。

购买并登录弹性云服务器

参考[购买并登录Linux弹性云服务器](#)创建一台带有弹性公网IP的Linux弹性云服务器。

作为演示，弹性云服务器和公网IP的规格不需要太高，例如弹性云服务器的规格为“1vCPUs | 2GiB”、公网IP带宽为“1 Mbit/s”即可，操作系统以选择“CentOS 7.5”为例。

说明

- 您也可以使用其他机器安装容器引擎，不创建弹性云服务器。
- 如果您使用的操作系统为EulerOS，请参考[EulerOS操作系统下安装](#)。
- 如果您使用的是Centos操作系统，建议选择CentOS7、CentOS7.2、CentOS7.3、CentOS7.4、CentOS7.5、CentOS7.6操作系统版本，否则可能导致安装异常。

创建完成后返回弹性云服务器列表，单击操作列的“远程登录”，以root用户登录弹性云服务器。

安装 Docker 容器引擎

使用如下命令快速安装容器引擎，具体请参见[Docker 安装指南](#)。

```
curl -fsSL get.docker.com -o get-docker.sh
sh get-docker.sh
sudo systemctl daemon-reload
sudo systemctl restart docker
```

二、构建镜像

构建镜像的内容可以是源代码，也可以是已打包成压缩包的镜像。

构建源代码镜像

步骤1 创建一个名为Dockerfile的文件。

```
mkdir mynginx
cd mynginx
touch Dockerfile
```

步骤2 编辑Dockerfile。

```
vim Dockerfile
```

增加文件内容如下：

```
FROM nginx
RUN echo '<h1>Hello, SWR!</h1>' > /usr/share/nginx/html/index.html
```

Dockerfile指令介绍如下。

- FROM语句：表示使用nginx镜像作为基础镜像，一个Dockerfile中FROM是必备的指令，并且必须是第一条指令。

- RUN语句：格式为RUN <命令>，表示执行echo命令，在显示器中显示一段“Hello, SWR!”的文字。

按“Esc”，输入:wq，保存并退出。

步骤3 使用docker build [选项] <上下文路径> 构建镜像。

```
docker build -t nginx:v1 .
```

- -t：指定镜像的名称和版本。
- .：指定Dockerfile所在目录，镜像构建命令将该路径下所有的内容打包给容器引擎帮助构建镜像。

有关 Dockerfile 的更多信息，请参考 [Dockerfile 参考](#)。

步骤4 执行以下命令，可查看到已成功部署的nginx镜像，版本为v1。

```
docker images
```

----结束

构建压缩包镜像

下面指导您通过docker load命令和ctr images import命令将镜像压缩包导入为一个镜像。

制作镜像压缩包

本节指导您将容器镜像制作成tar或tar.gz文件压缩包。

步骤1 以root用户登录容器引擎所在机器。

步骤2 执行如下命令查看镜像。

```
docker images
```

查看需要导出的镜像及tag。

步骤3 执行如下命令制作镜像压缩包。

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

- OPTIONS：--output或-o，表示导出到文件。
- 压缩包格式为：.tar或.tar.gz。

注意

使用docker save制作镜像压缩包时，请用{image}:{tag}（请替换为实际镜像名和版本号），不要用image id（镜像id），否则无法在SWR页面上上传。

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 # 将nginx所有版本打包
docker save -o nginx-latest.tar nginx:latest
```

----结束

导入镜像文件

使用Docker客户端导入，执行以下命令：

```
docker load < 路径/文件名.tar
docker load --input或者-i 路径/文件名.tar
```

示例：

```
docker load --input fedora.tar
```

三、创建组织

组织用于隔离镜像，并为账号下的IAM用户指定不同的权限（读取、编辑、管理）。

- 步骤1 登录[SWR控制台](#)。
- 步骤2 单击控制台左上角的，选择区域和项目。
- 步骤3 在左侧导航栏单击“组织管理”，进入组织管理页面。
- 步骤4 单击页面右上角的“创建组织”按钮，在弹框中填写“组织名称”，然后单击“确定”。

创建组织

 规则

1. 组织名称，全局唯一。

2. 当前租户最多可创建5个组织。

3. 推荐您创建的每一个组织对应一个公司、下属部门或者个人用户。

示例

1. 以公司、部门作为组织：cloud-hangzhou、cloud-develop。

2. 以个人作为组织：john。

组织名称

请输入名称

表 1-1 创建组织参数说明

参数	示例	参数说明
组织名称	cloud-develop	组织用于隔离镜像仓库，每个组织可对应一个公司或部门，将其拥有的镜像集中在该组织下。 小写字母开头，后面跟小写字母、数字、点、下划线或中划线（其中下划线最多允许连续两个，中划线允许连续多个，点、下划线、中划线不能直接相连），小写字母或数字结尾，1-64个字符。

说明

- 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，请重新设置一个组织名称。
- 用户在IAM中被授予SWR Admin或Tenant Administrator策略才有创建组织的权限。
- 不允许创建名称为library的组织，该组织为系统预留，请更换一个其他的名称。

----结束

四、连接容器镜像服务

步骤1 登录[SWR控制台](#)。

步骤2 选择左侧导航栏的“总览”，单击页面右上角的“登录指令”，在弹出的页面中单击  复制登录指令。

图 1-2 登录指令



说明

临时登录指令支持用户自定义临时登录有效时间，支持设置有效期是15分钟-24小时，过期会失效，若需要长期有效的登录指令，请参考[获取长期有效登录指令](#)。

Docker客户端需要获取登录指令，Containerd客户端无需获取登录指令。

步骤3 在安装容器引擎的虚拟机中执行上一步复制的登录指令。

登录成功会显示“Login Succeeded”。

----结束

五、推送镜像

本节介绍使用Docker客户端推送镜像的方法。

步骤1 在安装容器引擎的虚拟机上执行以下命令，为nginx镜像打标签。

```
docker tag [镜像名称1:版本名称1] [镜像仓库地址]/[组织名称]/[镜像名称2:版本名称2]
```

其中，

- **[镜像名称1:版本名称1]**：请替换为您所要推送的实际镜像的名称和版本名称。
- **[镜像仓库地址]**：可在SWR控制台上查询，[四、连接容器镜像服务](#)中登录指令末尾的域名即为镜像仓库地址。
- **[组织名称]**：请替换为[三、创建组织](#)中创建的组织。
- **[镜像名称2:版本名称2]**：请替换为您期待的镜像名称和镜像版本。

示例：

```
docker tag nginx:v1 swr.cn-north-4.myhuaweicloud.com/cloud-develop/nginx:v1
```

步骤2 推送镜像至镜像仓库。

```
docker push [镜像仓库地址]/[组织名称]/[镜像名称2:版本名称2]
```

示例：

```
docker push swr.cn-north-4.myhuaweicloud.com/cloud-develop/nginx:v1
```

终端显示如下信息，表明推送镜像成功。

```
The push refers to repository [swr.cn-north-4.myhuaweicloud.com/cloud-develop/nginx]
fbce26647e70: Pushed
fb04ab8effa8: Pushed
8f736d52032f: Pushed
009f1d338b57: Pushed
678bbd796838: Pushed
d1279c519351: Pushed
f68ef921efae: Pushed
v1: digest: sha256:0cdfc7910db531bfa7726de4c19ec556bc9190aad9bd3de93787e8bce3385f8d size: 1780
```

返回SWR管理控制台，在“我的镜像”页面，执行刷新操作后可查看到对应的镜像信息。

----结束

六、拉取镜像

拉取镜像（也叫下载镜像），当您需要使用镜像仓库中的镜像时，您需要从镜像仓库拉取镜像。常搭配华为云产品云容器引擎CCE部署工作负载或者云容器实例CCI部署实例使用。

步骤1 登录[SWR控制台](#)。

步骤2 单击左侧导航栏的“我的镜像”进入镜像列表页面。

步骤3 单击镜像名称进入镜像详情页面。

步骤4 单击Pull/Push指南页签，选择docker容器引擎，复制页面生成的docker pull指令，在终端执行该指令。

示例：

```
docker pull swr.cn-east-3.myhuaweicloud.com/group/nginx:v2.0.0
```

步骤5 执行docker images查看已拉取的镜像。

```
docker images
```

----结束

七、删除镜像

容器镜像服务支持镜像的全生命周期管理，包括镜像的推送、拉取、删除等。

通过控制台删除镜像

步骤1 登录[SWR控制台](#)。

步骤2 在左侧导航栏选择“我的镜像”。

步骤3 找到需要删除的镜像，单击右侧操作列的更多，下拉菜单单击“删除”。

步骤4 在弹出的确认框中输入“DELETE”，单击“确定”，完成删除。

----结束

Docker 客户端删除镜像

如果您需要删除本地机器上的镜像，可以使用以下命令：

```
# 查看本地镜像
docker images
# 删除本地镜像
docker rmi [镜像名称:版本名称]
```

示例：

```
docker rmi swr.cn-north-4.myhuaweicloud.com/cloud-develop/nginx:v1
```

或删除本地构建的镜像：

```
docker rmi nginx:v1
```



注意

删除镜像前，请确保没有容器正在使用该镜像。您可以先执行docker ps -a查看所有容器，并停止或删除使用该镜像的容器。

Docker容器引擎客户端只删除本机器的镜像，不会删除SWR仓库里的镜像，SWR仓库的镜像要通过控制台删除方式。

八、删除组织

步骤1 登录[SWR控制台](#)。

步骤2 单击控制台左上角的，选择区域和项目。

步骤3 在左侧导航栏选择“组织管理”。

步骤4 单击待删除组织右上角“删除”按钮，单击“确定”。

图 1-3 删除组织



----结束

2 使用云容器引擎 CCE 拉取镜像并部署应用

您可以使用镜像快速创建一个可公网访问的单实例工作负载。本章节将指导您基于云容器引擎CCE快速部署Nginx容器应用。

前提条件

1. **首次开通CCE并进行授权**
 - 详细步骤请参见[首次开通CCE并进行授权](#)。
2. **创建集群**
 - 详细操作步骤请参见[创建集群](#)。
3. **在集群中创建节点池和节点**
 - 详细操作步骤请参见[创建节点池和节点](#)。
 - 集群至少包含1个4核8G节点。
 - 集群内节点已绑定弹性公网IP。

创建并访问工作负载

您可以登录[CCE控制台](#)，使用控制台和[kubectl](#)命令行方式部署工作负载，本文中使用的NGINX镜像作为示例。

使用控制台方式

步骤1 在集群控制台的左侧菜单栏选择“工作负载”，单击右上角“创建工作负载”。

步骤2 配置工作负载基本信息。

本示例中填写以下关键参数，其它参数可保持默认。关于配置参数的详细说明请参见[创建无状态负载（Deployment）](#)。

参数	示例	参数说明
负载类型	无状态负载 Deployment	工作负载是在 Kubernetes 上运行的应用程序。Kubernetes提供了一些内置的工作负载资源，不同工作负载的功能特性及应用场景不同。关于工作负载类型的介绍请参见 工作负载概述 。
负载名称	nginx	请填写工作负载的名称。

参数	示例	参数说明
命名空间	default	命名空间是Kubernetes集群中的抽象概念，可以将集群中的资源或对象划分为一个组，且不同命名空间中的数据彼此隔离。 集群创建后将默认生成一个default命名空间，您可以直接使用该命名空间。
实例数量	1	工作负载中的Pod数量。

步骤3 配置容器信息。

填写以下关键参数，其它参数可保持默认。

容器配置

告警信息

容器 - 1

+ 添加容器

基本信息

生命周期

健康检查

环境变量

数据挂载

安全策略

容器日志

容器名称

container-1

镜像名称

nginx

更新镜像

CPU配额

申请

0.25

cores

限制

0.25

cores

GPU配额

暂未支持GPU组件，无法使用此功能。点此安装组件

特权容器

☐

更新策略

☐ 白名单拉取镜像

镜像版本

latest

内存配额

申请

512.00

MB

限制

512.00

MB

HPA配额

暂未支持HPA组件，无法使用此功能。点此安装组件

初始化容器

☐

镜像访问凭证

default-secret

创建策略

参数	示例	参数说明
镜像名称	使用nginx镜像，镜像版本为latest	单击“选择镜像”，在弹出的窗口中切换至“镜像中心”，选择一个公共镜像。
CPU配额	申请值为0.25Core，限制值为0.25Core	- 申请：容器预分配的CPU值，默认0.25Core。 - 限制：容器允许使用的CPU资源最大上限，默认与申请值相同。如果设置限制值大于申请值，则表示在突发场景可临时突破预分配的资源限制。 详细说明请参见 设置容器规格 。
内存配额	申请值为512MiB，限制值为512MiB	- 申请：容器预分配的内存值，默认512MiB。 - 限制：容器允许使用的内存资源最大上限，默认与申请值相同。如果设置限制值大于申请值，则表示在突发场景可临时突破预分配的资源限制。 详细说明请参见 设置容器规格 。

步骤4 配置访问信息。

单击“服务配置”下的加号，创建服务（Service）用于从外部访问负载。本示例将创建一个负载均衡类型的Service，请在右侧弹窗中配置如下参数。

创建服务

×

Service名称

nginx

访问类型

集群内访问

通过集群的内部IP暴露服务，只能在集群内部访问

节点访问

通过每个节点上的IP和静态端口（NodePort）暴露服务

负载均衡

通过ELB负载均衡对外部提供服务，高可用，超高性能，稳定安全

DNAT网关

通过NAT网关暴露集群节点访问类型服务，支持多个节点共享使用弹性IP

集群外访问推荐选择负载均衡访问类型

服务亲和

集群级别

节点级别

?

负载均衡器

独享型

网络型（TCP/UDP）

自动创建

将基于如下配置自动创建负载均衡实例，当前资源删除时实例会被自动删除（若被多个资源关联，可能残留）。
自动创建的负载均衡实例按需计费。 [价格详情](#)

实例名称

默认随机生成

企业项目

default

新建企业项目

?

可用区

可用区1

×

?

前端子网

subnet-A02 (172.16.1.0/24)

查看子网

?

后端子网

与前端子网保持一致

?

网络规格

弹性规格

固定规格

?

单可用区实例最大支持400,000新建连接数、20,000,000并发连接数、10,000,000Mbit/s带宽的处理能力，实例性能随可用区数量叠加。

弹性公网IP

暂不使用

自动创建

?

参数	示例	参数说明
Service名称	nginx	输入服务的名称。
访问类型	负载均衡（LoadBalancer）	选择服务类型，即服务访问的方式。不同的服务类型的差别请参见 服务概述 。
负载均衡器	- 类型：选择独享型。 - 可用区：至少选择一个可用区，如可用区1。 - 弹性公网IP：选择自动创建。 其余参数可保持默认。	如果已有负载均衡（ELB）实例，可以选择已有ELB。 如果没有可选择“自动创建”，创建一个负载均衡器，并同时绑定弹性公网IP。关于参数的详细说明，请参见创建负载均衡类型的服务。

参数	示例	参数说明
端口配置	- 协议：TCP - 容器端口：80 - 服务端口：8080	- 协议：负载均衡监听器端口协议。 - 容器端口：容器中应用启动监听的端口，该容器端口需和应用对外提供的监听端口一致，使用nginx镜像请设置为80。 - 服务端口：ELB将会使用该端口创建监听器，提供外部流量访问入口。您可以自定义对外访问的端口。

步骤5 单击右下角“创建工作负载”。

等待工作负载创建成功。创建成功后在无状态负载下会显示一个运行中的工作负载。

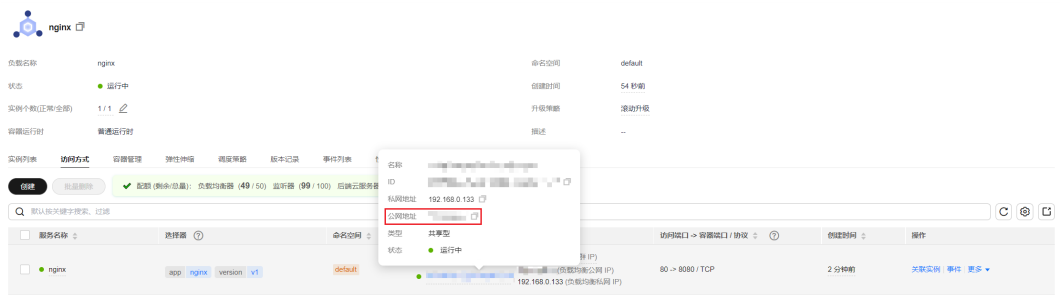
图 2-1 创建工作负载成功



步骤6 获取Nginx的外部访问地址。

单击Nginx工作负载名称，进入工作负载详情页。在“访问方式”页签下可以看到nginx的IP地址，其中公网地址就是外部访问地址。

图 2-2 获取外部访问地址



步骤7 在浏览器中输入“外部访问地址:服务端口”，即可成功访问应用。其中“服务端口”对应步骤4中配置的服务端口值，本例中使用8080。

图 2-3 访问 nginx 应用



----结束

使用 kubectl 命令行方式

须知

如果您使用kubectl命令行方式连接集群，您需要在**集群所在VPC**下准备一台已绑定弹性公网IP的ECS虚拟机。

步骤1 登录ECS虚拟机，详情请参见[Linux ECS登录方式概述](#)。

步骤2 在ECS虚拟机上安装kubectl命令行工具。

您可以尝试执行**kubectl version**命令判断是否已安装kubectl，如果已经安装kubectl，则可跳过此步骤。

本文以Linux环境为例安装和配置kubectl，更多安装方式请参考[安装kubectl](#)。

1. 下载kubectl。

```
cd /home
curl -LO https://dl.k8s.io/release/{v1.29.0}/bin/linux/amd64/kubectl
```

其中{v1.29.0}为指定的版本号，请根据集群版本进行替换。

2. 安装kubectl。

```
chmod +x kubectl
mv -f kubectl /usr/local/bin
```

步骤3 为kubectl命令行工具配置访问Kubernetes集群的凭证。

1. 请登录[CCE控制台](#)，并单击集群名称进入集群总览页。

2. 在集群总览页中找到“连接信息”版块。单击kubectl后的“配置”按钮，查看kubectl的连接信息。

3. 在弹出页面中选择“内网访问”，然后下载对应的配置文件。

4. 登录已安装kubectl客户端的虚拟机，将上一步中下载的配置文件的（以kubeconfig.yaml为例）复制到/home目录下。

5. 将kubectl认证文件保持至\$HOME/.kube目录下的config文件中。

```
cd /home
mkdir -p $HOME/.kube
mv -f kubeconfig.yaml $HOME/.kube/config
```

6. 执行kubectl命令验证集群的连通性。

以查看集群信息为例，执行以下命令。

```
kubectl cluster-info
```

回显如下：

```
Kubernetes master is running at https://*:*:5443
CoreDNS is running at https://*:*:5443/api/v1/namespaces/kube-system/services/coresdns:proxy
To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
```

步骤4 创建一个名为nginx-deployment.yaml的YAML文件。其中，nginx-deployment.yaml为自定义文件名称，您可以随意命名。

```
vi nginx-deployment.yaml
```

文件内容如下：

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx
spec:
  replicas: 1
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - image: nginx:alpine
          name: nginx
      imagePullSecrets:
        - name: default-secret
```

步骤5 执行以下命令，部署工作负载。

```
kubectl create -f nginx-deployment.yaml
```

回显如下表示已开始创建工作负载。

```
deployment "nginx" created
```

步骤6 执行以下命令，查看工作负载状态。

```
kubectl get deployment
```

回显如下，表示已创建成功。

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
nginx	1/1	1	1	4m5s

回显中的参数解析如下：

- NAME：工作负载名称。
- READY：表示工作负载的可用状态，显示为“可用Pod个数/期望Pod个数”。
- UP-TO-DATE：指当前工作负载已经完成更新的Pod数。
- AVAILABLE：工作负载可用的Pod个数。
- AGE：工作负载已经运行的时间。

步骤7 创建一个名为nginx-elb-svc.yaml的YAML文件。将selector修改为nginx-deployment.yaml示例应用文件中matchLabels的值（本示例为app: nginx），从而将该服务关联至后端应用。

```
vi nginx-elb-svc.yaml
```

示例如下，其中参数详细描述请参见[负载均衡\(LoadBalancer\)](#)。

```
apiVersion: v1
kind: Service
```

```

metadata:
  annotations:
    kubernetes.io/elb.class: union
    kubernetes.io/elb.autocreate:
      {
        "type": "public",
        "bandwidth_name": "cce-bandwidth",
        "bandwidth_chargemode": "bandwidth",
        "bandwidth_size": 5,
        "bandwidth_sharetype": "PER",
        "eip_type": "5_bgp"
      }
  labels:
    app: nginx
    name: nginx
spec:
  ports:
    - name: service0
      port: 8080
      protocol: TCP
      targetPort: 80
  selector:
    app: nginx
  type: LoadBalancer

```

步骤8 执行以下命令创建服务。

```
kubectl create -f nginx-elb-svc.yaml
```

回显如下，表示服务已创建。

```
service/nginx created
```

步骤9 执行以下命令查看服务。

```
kubectl get svc
```

回显如下，表示工作负载访问方式已设置成功，工作负载可访问。

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
kubernetes	ClusterIP	10.247.0.1	<none>	443/TCP	3d
nginx	LoadBalancer	10.247.130.196	**.*.*.*	8080:31540/TCP	51s

步骤10 在浏览器中输入访问地址，例如输入**.*.*.*:8080。其中**.*.*.*为负载均衡实例的弹性公网IP地址，8080为访问端口。

图 2-4 通过负载均衡访问 nginx

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

----结束