

云容器引擎

# 快速入门

文档版本 01  
发布日期 2024-09-29



版权所有 © 华为云计算技术有限公司 2024。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

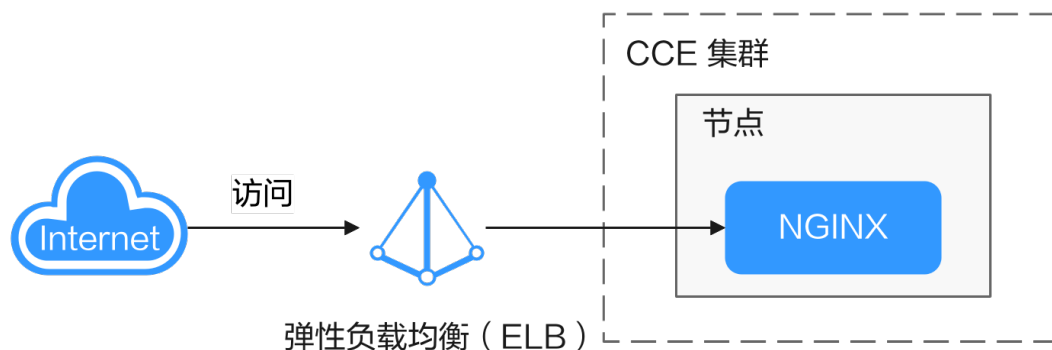
---

1 在 CCE 集群中部署 NGINX 无状态工作负载.....	1
2 在 CCE 集群中部署 WordPress 有状态工作负载.....	16
3 在 CCE 集群中通过 Helm 模板部署应用程序.....	39

# 1 在 CCE 集群中部署 NGINX 无状态工作负载

无状态工作负载（Deployment）是Kubernetes中的一种工作负载类型，通常适用于不需要考虑数据一致性和持久性的应用程序，例如Web服务器、应用服务器等典型应用场景。在无状态工作负载中，每个应用实例都是独立的，且实例之间没有运行状态的差异，因此即使某个实例发生故障，它负责接受的请求也可以被重新分配给其他健康的实例，从而保证服务的连续性。由于无状态负载实例相互独立、可替换的特性，您可以根据实时的业务需求灵活调整实例数量，比如在流量高峰期增加实例数量来分担压力。

本文将使用轻量级的Web服务器NGINX作为示例，在CCE集群中部署无状态工作负载。



## 操作流程

操作步骤	说明
<b>准备工作</b>	您需要注册华为账号，并为账户充值。
<b>步骤一：首次开通CCE并进行授权</b>	当您的账号在当前区域中首次使用CCE时，您需要为CCE进行授权。
<b>步骤二：创建集群</b>	您需要在CCE服务中创建一个集群，以提供Kubernetes服务。
<b>步骤三：在集群中创建节点池和节点</b>	您需在集群中创建节点，即可在节点上运行您的容器化应用。

操作步骤	说明
<a href="#">步骤四：创建并访问工作负载</a>	在集群中创建工作负载以运行您的容器，并为其创建一个服务，然后您就可以从公网访问您的应用。
<a href="#">后续操作：释放资源</a>	如果您在完成实践后不需要继续使用集群，请及时清理资源以免产生额外扣费。


## 准备工作

- 在开始操作前，请您先注册华为账号并完成实名认证，详情请参见[注册华为账号并开通华为云和个人实名认证](#)。

## 步骤一：首次开通 CCE 并进行授权

由于CCE在运行中对计算、存储、网络以及监控等各类云服务资源都存在依赖关系，因此当您首次登录CCE控制台时，CCE将自动请求获取当前区域下的云资源权限，从而更好地为您提供服务。如果您在当前区域已完成授权，可忽略本步骤。

**步骤1** 使用华为账号登录[CCE控制台](#)。

**步骤2** 单击管理控制台左上角的，选择区域。

**步骤3** 在首次登录某个区域的CCE控制台时将跳出“授权说明”，请您在仔细阅读后单击“确定”。

当您同意授权后，CCE将在IAM中创建名为“cce\_admin\_trust”委托，统一对您的其他云服务资源进行操作，并且授予其Tenant Administrator权限。Tenant Administrator拥有除IAM管理外的全部云服务管理员权限，用于对CCE所依赖的其他云服务资源进行调用，且该授权仅在当前区域生效。您可前往“IAM控制台 > 委托”页签，单击“cce\_admin\_trust”查看各区域的授权记录。关于资源委托详情，您可参考[委托](#)进行了解。

### 说明

由于CCE对其他云服务有许多依赖，如果没有Tenant Administrator权限，可能会因为某个服务权限不足而影响CCE功能的正常使用。因此在使用CCE服务期间，请不要自行删除或者修改“cce\_admin\_trust”委托。

----结束

## 步骤二：创建集群

**步骤1** 登录[CCE控制台](#)。

- 如果您的账号还未创建过集群，请在引导页面中单击页面上方的“购买集群”。
- 如果您的账号已经创建过集群，请在左侧菜单栏选择集群管理，单击右上角“购买集群”。

**步骤2** 配置集群基础参数。

本示例中仅解释必要参数，其余大多数配置可保留默认值。关于配置参数的详细说明请参见[创建集群](#)。



参数	示例	参数说明
集群类型	<b>CCE Standard 集群</b>	<p>CCE支持多种类型的集群创建，以满足您各种业务需求，可为您提供高可靠、安全的商业级容器集群服务。</p> <p>根据需求选择“CCE Standard集群”或“CCE Turbo集群”。</p> <ul style="list-style-type: none"> <li>● CCE Standard集群：标准版本集群，提供高可靠、安全的商业级容器集群服务。</li> <li>● CCE Turbo集群：拥有更高性能的云原生网络，提供云原生混部调度能力，可实现更高的资源利用率和更广的全场景覆盖。</li> </ul> <p>关于集群类型差异详情，请参见<a href="#">集群类型对比</a>。</p>
计费模式	<b>按需计费</b>	<p>选择集群的计费模式。</p> <ul style="list-style-type: none"> <li>● 包年/包月：预付费模式，按订单的购买周期计费，适用于可预估资源使用周期的场景，价格比按需计费模式更优惠。选择该计费模式时，需要设置“购买时长”并选择是否自动续费（按月购买时自动续费周期为1个月，按年购买时自动续费周期则为1年）。</li> <li>● 按需计费：后付费模式，按资源的实际使用时长计费，可以随时开通/删除资源。</li> </ul> <p>更多计费相关信息请参见<a href="#">计费模式概述</a>。</p>
集群名称	<b>cce-test</b>	自定义新建集群的名称。
企业项目	<b>default</b>	<p>企业项目是一种云资源管理方式，企业项目管理服务提供统一的云资源按项目管理，以及项目内的资源管理、成员管理。了解更多企业项目相关信息，请查看<a href="#">企业管理</a>。</p> <p>该参数仅对开通企业项目的企业客户账号显示，不显示时请忽略。</p>
集群版本	<b>选择推荐版本，如v1.29</b>	集群安装的Kubernetes软件版本，建议选择最新的商用版本，可为您提供更多稳定可靠的新特性。

参数	示例	参数说明
集群规模	50节点	集群可以管理的最大节点数量，请根据业务场景选择。创建完成后支持扩容，不支持缩容。
集群 master 实例数	3实例（高可用）	<p>选择集群控制平面的节点（master实例）数量。控制平面节点由系统自动托管，会部署 Kubernetes 集群的管控面组件，如 kube-apiserver, kube-controller-manager, kube-scheduler 等组件。</p> <ul style="list-style-type: none"> <li>3实例（高可用）：创建3个控制平面节点，确保集群高可用。</li> <li>单实例：您的集群只会创建一个控制平面节点。</li> </ul> <p>集群创建完成后不支持修改。</p>

### 步骤3 配置集群网络参数。

参数	示例	参数说明
虚拟私有云	vpc-cce	<p>为新建集群选择一个虚拟私有云。</p> <p>若没有可选虚拟私有云，请单击“新建虚拟私有云”进行创建，完成创建后单击刷新按钮。操作步骤详情请参见<a href="#">创建虚拟私有云和子网</a>。</p>
子网	subnet-cce	<p>选择一个子网，集群中的节点将会使用此子网中的IP。</p>

参数	示例	参数说明
容器网络模型	VPC网络	支持选择“VPC网络”和“容器隧道网络”，默认可选择“VPC网络”。 不同容器网络模型间的差异请参见 <a href="#">容器网络模型对比</a> 。
容器网段	10.0.0.0/16	设置容器使用的网段，该网段大小决定了集群中容器的数量上限。
服务网段	10.247.0.0/16	为集群配置ClusterIP地址的范围，该网段决定了Service资源数量的上限，且创建后不可修改。

**步骤4** 单击“下一步：插件选择”，选择创建集群时需要安装的插件。

本示例中，仅选择默认安装的必选插件。



**步骤5** 单击“下一步：插件配置”，默认插件可无需配置。

**步骤6** 单击“下一步：规格确认”，显示集群资源清单，确认无误后，单击“提交”。

等待集群创建成功，创建集群预计需要5-10分钟左右，请耐心等待。

创建成功后在集群管理下会显示一个运行中的集群，且集群节点数量为0。

图 1-1 集群创建成功



----结束

## 步骤三：在集群中创建节点池和节点

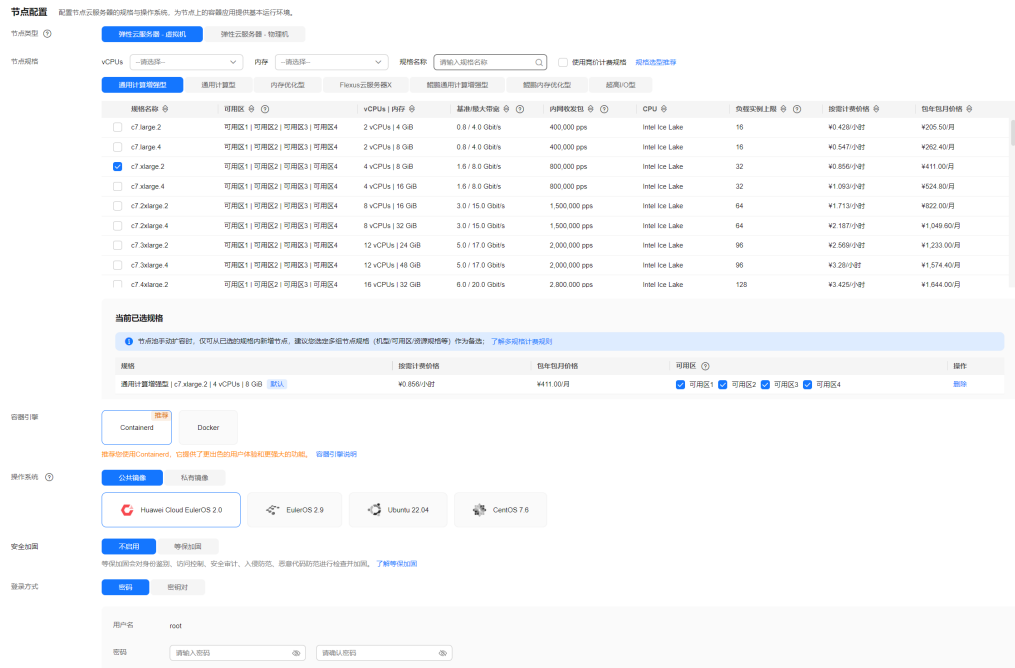
**步骤1** 单击新建的集群名称进入集群控制台。



**步骤2** 在左侧菜单栏中选择“节点管理”，单击右上角“创建节点池”。

**步骤3** 配置节点池参数。

本示例中仅解释必要参数，其余大多数配置可保留默认值。关于配置参数的详细说明请参见[创建节点池](#)。



参数	示例	参数说明
节点类型	弹性云服务器-虚拟机	请根据不同的业务诉求选择节点类型，“节点规格”列表中将自动为您筛选该类型下可部署容器服务的规格，供您进一步选择。
节点规格	4 vCPUs   8 GiB	请根据业务需求选择相应的节点规格。为保证集群中的组件可以正常运行，建议的节点规格为：vCPU≥4核，内存≥8 GiB。
容器引擎	Containerd	请根据业务需要选择相应的容器引擎。不同的容器引擎差别请参见 <a href="#">容器引擎</a> 。
操作系统	Huawei Cloud EulerOS 2.0	请选择节点对应的操作系统。

参数	示例	参数说明
登录方式	选择“密码”方式登录，密码请您自行设置。	<ul style="list-style-type: none"> <li>选择“密码”：用户名默认为“root”，请输入登录节点的密码，并确认密码。请妥善保管密码，登录节点时需要使用该密码，系统无法获取您设置的密码内容。</li> <li>选择“密钥对”：在选项框中选择用于登录本节点的密钥对，并单击勾选确认信息。密钥对用于远程登录节点时的身份认证。若没有密钥对，可单击选项框右侧的“创建密钥对”来新建，创建密钥对操作步骤请参见<a href="#">创建密钥对</a>。</li> </ul>

**步骤4** 配置“存储配置”和“网络配置”，本示例中均可保持默认值，您只需要在“网络配置”中勾选确认节点池关联安全组可用，然后单击“下一步: 规格确认”。



**步骤5** 查看节点规格无误后，阅读页面上的使用说明，确认后单击“提交”。

**步骤6** 等待节点池创建成功后，节点池中默认节点数为0，您需要单击节点池所在行的“扩缩容”，为节点池创建节点。



**步骤7** 设置本次扩容的节点数量为2，即在节点池中创建2个节点。

**节点池扩缩容**

节点池名称: cce-test-pool1

当前数量: 0

扩缩容:

扩容规格:    
使用选择的规格扩容节点, 如果规格资源不足会导致扩容失败

计费模式:

本次扩容节点数:    
当前集群管理规模(50 节点)下还可以创建 48 个节点

**步骤8** 等待节点创建成功, 创建节点预计需要5-10分钟左右, 请耐心等待。



---结束

## 步骤四：创建并访问工作负载

您可以使用控制台和**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	<b>无状态负载 Deployment</b>	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 <a href="#">工作负载概述</a> 。
负载名称	<b>nginx</b>	请填写工作负载的名称。
命名空间	<b>default</b>	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	<b>1</b>	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。



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	<ul style="list-style-type: none"><li>● 申请：容器预分配的CPU值，默认0.25Core。</li><li>● 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。</li></ul> 详细说明请参见 <a href="#">设置容器规格</a> 。
内存配额	申请值为512MiB，限制值为512MiB	<ul style="list-style-type: none"><li>● 申请：容器预分配的内存值，默认512MiB。</li><li>● 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。</li></ul> 详细说明请参见 <a href="#">设置容器规格</a> 。

### 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: 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)

服务亲和: 集群级别

负载均衡器: 独享型, 网络型 (TCP/UDP), 自动创建

实例名称: 默认随机生成

企业项目: default

可用区: 可用区1

前端子网: subnet-A02 (172.16.1.0/24)

后端子网: 与前端子网保持一致

网络规格: 弹性规格

弹性公网IP: 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 <a href="#">服务概述</a> 。
负载均衡器	<ul style="list-style-type: none"> <li>类型: 选择<b>独享型</b>。</li> <li>可用区: 至少选择一个可用区, 如<b>可用区1</b>。</li> <li>弹性公网IP: 选择<b>自动创建</b>。</li> </ul> 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 <a href="#">创建负载均衡类型的服务</a> 。

参数	示例	参数说明
端口配置	<ul style="list-style-type: none"> <li>协议：TCP</li> <li>容器端口：<b>80</b></li> <li>服务端口：<b>8080</b></li> </ul>	<ul style="list-style-type: none"> <li>协议：负载均衡监听器端口协议。</li> <li>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。</li> <li>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。</li> </ul>

**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。

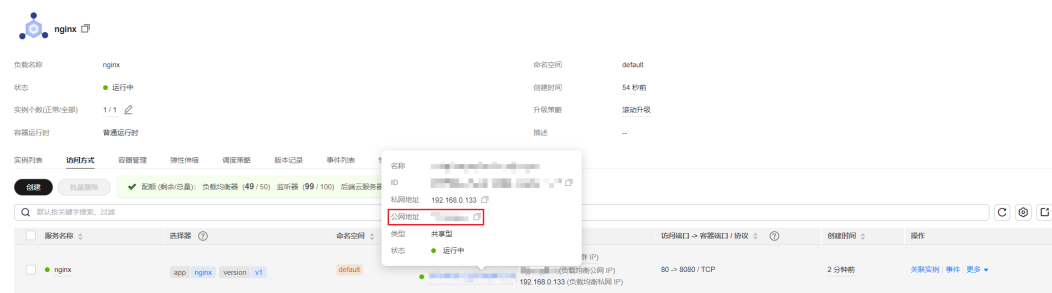
图 1-2 创建工作负载成功



**步骤6** 获取Nginx的外部访问地址。

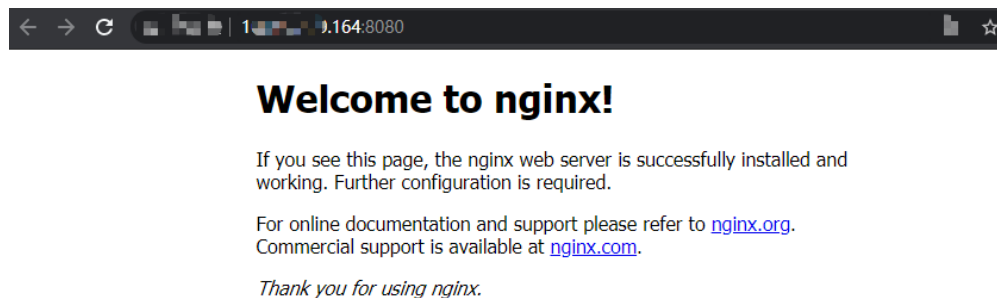
单击Nginx工作负载名称，进入工作负载详情页。在“访问方式”页签下可以看到nginx的IP地址，其中公网地址就是外部访问地址。

图 1-3 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 1-4 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 1-5 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](http://nginx.org).  
Commercial support is available at [nginx.com](http://nginx.com).

*Thank you for using nginx.*

----结束

## 后续操作：释放资源

如果您无需继续使用集群，请及时释放资源，避免产生额外的费用。详细操作步骤请参见[删除集群](#)。

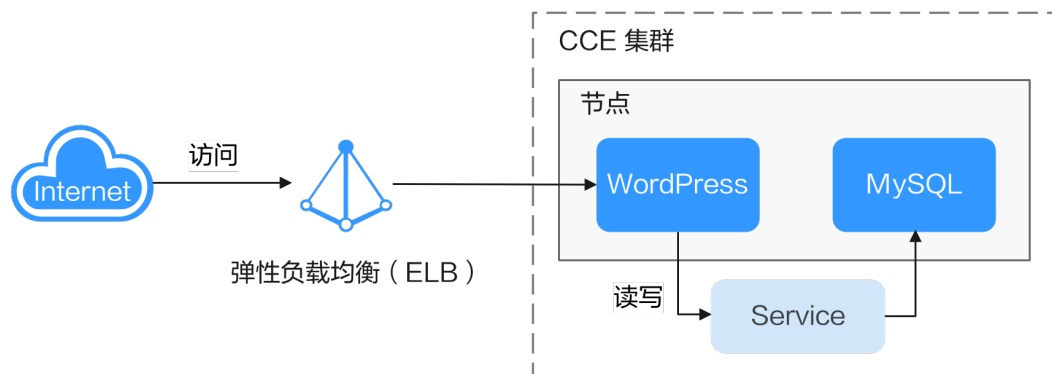
# 2 在 CCE 集群中部署 WordPress 有状态工作负载

## 负载

有状态工作负载（StatefulSet）是Kubernetes中用于管理有状态应用的一种工作负载类型。与无状态工作负载不同，有状态工作负载需要保持数据的一致性和持久性，且每个应用实例具有独立的标识符，需要按顺序部署和扩展。典型的有状态应用场景包括数据库（如MySQL）、消息队列（如Kafka）等。本文将使用WordPress博客应用及MySQL数据库作为示例，在CCE集群中部署有状态工作负载。

WordPress是使用PHP语言和MySQL数据库开发的博客平台，并逐步演化成一款内容管理系统软件，您可以在CCE集群中使用容器搭建属于自己的博客网站。更多关于WordPress的信息请通过[WordPress官方网站](#)进行了解。

WordPress一般配合数据库（本文中MySQL）一起使用，WordPress作为内容管理程序运行，而数据库则用于存储网站数据。在容器化部署时，WordPress和MySQL通常会分别运行在两个不同的容器中，并通过集群内的Service进行访问。



### 操作流程

操作步骤	说明
准备工作	您需要注册华为账号，并为账户充值。
步骤一：首次开通CCE并进行授权	当您的账号在当前区域中首次使用CCE时，您需要为CCE进行授权。

操作步骤	说明
<a href="#">步骤二：创建集群</a>	您需要在CCE服务中创建一个集群，以提供Kubernetes服务。
<a href="#">步骤三：在集群中创建节点池和节点</a>	您需在集群中创建节点，即可在节点上运行您的容器化应用。
<a href="#">步骤四：部署MySQL</a>	在集群中创建MySQL工作负载，并为其创建一个集群内部访问服务，可供WordPress应用访问。
<a href="#">步骤五：部署WordPress</a>	在集群中创建WordPress工作负载，并为其创建一个可供公网访问的服务。
<a href="#">步骤六：访问WordPress</a>	您可以从公网访问刚刚创建的WordPress网站，开始您的博客之旅。
<a href="#">后续操作：释放资源</a>	如果您在完成实践后不需要继续使用集群，请及时清理资源以免产生额外扣费。


## 准备工作

- 在开始操作前，请您先注册华为账号并完成实名认证，详情请参见[注册华为账号并开通华为云](#)和[个人实名认证](#)。

## 步骤一：首次开通 CCE 并进行授权

由于CCE在运行中对计算、存储、网络以及监控等各类云服务资源都存在依赖关系，因此当您首次登录CCE控制台时，CCE将自动请求获取当前区域下的云资源权限，从而更好地为您提供服务。如果您在当前区域已完成授权，可忽略本步骤。

**步骤1** 使用华为账号登录[CCE控制台](#)。

**步骤2** 单击管理控制台左上角的，选择区域。

**步骤3** 在首次登录某个区域的CCE控制台时将跳出“授权说明”，请您在仔细阅读后单击“确定”。

当您同意授权后，CCE将在IAM中创建名为“cce\_admin\_trust”委托，统一对您的其他云服务资源进行操作，并且授予其Tenant Administrator权限。Tenant Administrator拥有除IAM管理外的全部云服务管理员权限，用于对CCE所依赖的其他云服务资源进行调用，且该授权仅在当前区域生效。您可前往“IAM控制台 > 委托”页签，单击“cce\_admin\_trust”查看各区域的授权记录。关于资源委托详情，您可参考[委托](#)进行了解。

### 说明

由于CCE对其他云服务有许多依赖，如果没有Tenant Administrator权限，可能会因为某个服务权限不足而影响CCE功能的正常使用。因此在使用CCE服务期间，请不要自行删除或者修改“cce\_admin\_trust”委托。

----结束

## 步骤二：创建集群

### 步骤1 登录CCE控制台。

- 如果您的账号还未创建过集群，请在引导页面中单击页面上方的“购买集群”。
- 如果您的账号已经创建过集群，请在左侧菜单栏选择集群管理，单击右上角“购买集群”。

### 步骤2 配置集群基础参数。

本示例中仅解释必要参数，其余大多数配置可保留默认值。关于配置参数的详细说明请参见[创建集群](#)。



参数	示例	参数说明
集群类型	<b>CCE Standard 集群</b>	<p>CCE支持多种类型的集群创建，以满足您各种业务需求，可为您提供高可靠、安全的商业级容器集群服务。</p> <p>根据需求选择“CCE Standard集群”或“CCE Turbo集群”。</p> <ul style="list-style-type: none"> <li>• CCE Standard集群：标准版本集群，提供高可靠、安全的商业级容器集群服务。</li> <li>• CCE Turbo集群：拥有更高性能的云原生网络，提供云原生混部调度能力，可实现更高的资源利用率和更广的全场景覆盖。</li> </ul> <p>关于集群类型差异详情，请参见<a href="#">集群类型对比</a>。</p>
计费模式	<b>按需计费</b>	<p>选择集群的计费模式。</p> <ul style="list-style-type: none"> <li>• 包年/包月：预付费模式，按订单的购买周期计费，适用于可预估资源使用周期的场景，价格比按需计费模式更优惠。选择该计费模式时，需要设置“购买时长”并选择是否自动续费（按月购买时自动续费周期为1个月，按年购买时自动续费周期则为1年）。</li> <li>• 按需计费：后付费模式，按资源的实际使用时长计费，可以随时开通/删除资源。</li> </ul> <p>更多计费相关信息请参见<a href="#">计费模式概述</a>。</p>

参数	示例	参数说明
集群名称	<b>cce-test</b>	自定义新建集群的名称。
企业项目	<b>default</b>	企业项目是一种云资源管理方式，企业项目管理服务提供统一的云资源按项目管理，以及项目内的资源管理、成员管理。了解更多企业项目相关信息，请查看 <a href="#">企业管理</a> 。 该参数仅对开通企业项目的企业客户账号显示，不显示时请忽略。
集群版本	<b>选择推荐版本，如v1.29</b>	集群安装的Kubernetes软件版本，建议选择最新的商用版本，可为您提供更多稳定可靠的新特性。
集群规模	<b>50节点</b>	集群可以管理的最大节点数量，请根据业务场景选择。创建完成后支持扩容，不支持缩容。
集群 master 实例数	<b>3实例（高可用）</b>	选择集群控制平面的节点（master实例）数量。控制平面节点由系统自动托管，会部署 Kubernetes 集群的管控面组件，如 kube-apiserver, kube-controller-manager, kube-scheduler 等组件。 <ul style="list-style-type: none"> <li>3实例（高可用）：创建3个控制平面节点，确保集群高可用。</li> <li>单实例：您的集群只会创建一个控制平面节点。</li> </ul> 集群创建完成后不支持修改。

### 步骤3 配置集群网络参数。

The screenshot displays the 'Network Configuration' (网络配置) section of the CCE console. It includes the following settings:

- 集群网络配置 (Cluster Network Configuration):**
  - 虚拟私有云 (Virtual Private Cloud): vpc-cce (192.168.0.0/16)
  - 子网 (Subnet): subnet-cce (192.168.0.0/24)
  - 启用 IPv6 (Enable IPv6): Disabled (如何搭建 IPv4/IPv6 双栈集群)
  - 默认安全组 (Default Security Group): 自动生成 (Automatic generation)
- 容器网络配置 (Container Network Configuration):**
  - 容器网络模型 (Container Network Model): VPC 网络 (Selected) and 容器隧道网络 (Container Tunneling Network)
  - 容器网络 (Container Network): 10.0.0.0/16
  - 每个节点预留的容器 IP 个数 (Number of container IP addresses reserved per node): 128
- 服务网络配置 (Service Network Configuration):**
  - 服务网络 (Service Network): 10.247.0.0/16

参数	示例	参数说明
虚拟私有云	vpc-cce	为新建集群选择一个虚拟私有云。 若没有可选虚拟私有云，请单击“新建虚拟私有云”进行创建，完成创建后单击刷新按钮。操作步骤详情请参见 <a href="#">创建虚拟私有云和子网</a> 。
子网	subnet-cce	选择一个子网，集群中的节点将会使用此子网中的IP。
容器网络模型	VPC网络	支持选择“VPC网络”和“容器隧道网络”，默认可选择“VPC网络”。 不同容器网络模型间的差异请参见 <a href="#">容器网络模型对比</a> 。
容器网段	10.0.0.0/16	设置容器使用的网段，该网段大小决定了集群中容器的数量上限。
服务网段	10.247.0.0/16	为集群配置ClusterIP地址的范围，该网段决定了Service资源数量的上限，且创建后不可修改。

**步骤4** 单击“下一步：插件选择”，选择创建集群时需要安装的插件。

本示例中，仅选择默认安装的必选插件。



**步骤5** 单击“下一步：插件配置”，默认插件可无需配置。

**步骤6** 单击“下一步：规格确认”，显示集群资源清单，确认无误后，单击“提交”。

等待集群创建成功，创建集群预计需要5-10分钟左右，请耐心等待。

创建成功后在集群管理下会显示一个运行中的集群，且集群节点数量为0。

图 2-1 集群创建成功



----结束

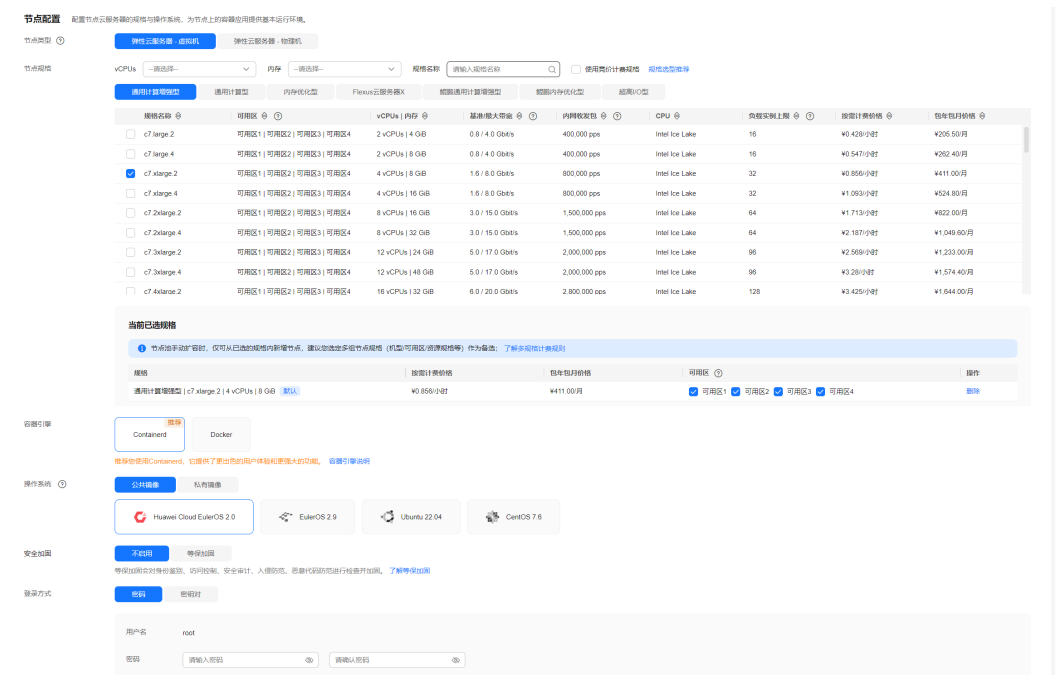
### 步骤三：在集群中创建节点池和节点

**步骤1** 单击新建的集群名称进入集群控制台。

**步骤2** 在左侧菜单栏中选择“节点管理”，单击右上角“创建节点池”。

**步骤3** 配置节点池参数。

本示例中仅解释必要参数，其余大多数配置可保留默认值。关于配置参数的详细说明请参见[创建节点池](#)。



参数	示例	参数说明
节点类型	弹性云服务器-虚拟机	请根据不同的业务诉求选择节点类型，“节点规格”列表中将自动为您筛选该类型下可部署容器服务的规格，供您进一步选择。
节点规格	4 vCPUs   8 GiB	请根据业务需求选择相应的节点规格。 为保证集群中的组件可以正常运行，建议的节点规格为：vCPU≥4核，内存≥8 GiB。
容器引擎	Containerd	请根据业务需要选择相应的容器引擎。不同的容器引擎差别请参见 <a href="#">容器引擎</a> 。
操作系统	Huawei Cloud EulerOS 2.0	请选择节点对应的操作系统。



参数	示例	参数说明
登录方式	选择“密码”方式登录，密码请您自行设置。	<ul style="list-style-type: none"> <li>选择“密码”：用户名默认为“root”，请输入登录节点的密码，并确认密码。请妥善保管密码，登录节点时需要使用该密码，系统无法获取您设置的密码内容。</li> <li>选择“密钥对”：在选项框中选择用于登录本节点的密钥对，并单击勾选确认信息。密钥对用于远程登录节点时的身份认证。若没有密钥对，可单击选项框右侧的“创建密钥对”来新建，创建密钥对操作步骤请参见<a href="#">创建密钥对</a>。</li> </ul>

**步骤4** 配置“存储配置”和“网络配置”，本示例中均可保持默认值，您只需要在“网络配置”中勾选确认节点池关联安全组可用，然后单击“下一步: 规格确认”。



**步骤5** 查看节点规格无误后，阅读页面上的使用说明，确认后单击“提交”。

**步骤6** 等待节点池创建成功后，节点池中默认节点数为0，您需要单击节点池所在行的“扩缩容”，为节点池创建节点。



**步骤7** 设置本次扩容的节点数量为2，即在节点池中创建2个节点。

**节点池扩缩容**

节点池名称: cce-test-pool1

当前数量: 0

扩缩容:

扩缩规格:    
使用选择的规格扩容节点, 如果规格资源不足会导致扩容失败

计费模式:

本次扩容节点数:

当前集群管理规模(50 节点)下还可以创建 48 个节点

**步骤8** 等待节点创建成功, 创建节点预计需要5-10分钟左右, 请耐心等待。



规格	可用区	状态	节点数量(实际/期望)	包年包月节点数	按需节点数	弹性伸缩	操作
c5.large.4.12.vCPU.18.5GB	可用区3	正常	2 / 2	0	2	关闭	<a href="#">节点列表</a> <a href="#">扩容</a>

----结束

## 步骤四：部署 MySQL

您可以使用不同的方式部署MySQL工作负载。

### 使用控制台方式

**步骤1** 登录[CCE控制台](#)。

**步骤2** 单击集群进入集群控制台。

**步骤3** 在左侧菜单栏选择“工作负载”，单击右上角“创建工作负载”。

**步骤4** 填写工作负载基本信息。

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[创建有状态负载（StatefulSet）](#)。

#### 基本信息

负载类型:  无状态负载 Deployment  有状态负载 StatefulSet  守护进程集 DaemonSet  普通任务 Job

 切换负载类型会导致已填写的部分关联数据被清空, 请谨慎切换

负载名称:

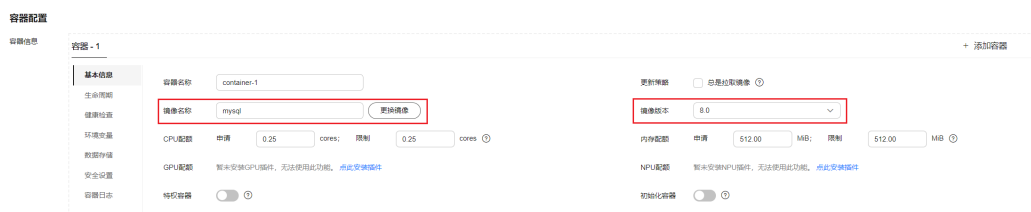
命名空间:  [创建命名空间](#)

实例数量:

时区同步:  开启后容器与节点使用相同时区 (时区同步功能依赖容器中挂载的本地磁盘, 请勿修改删除)

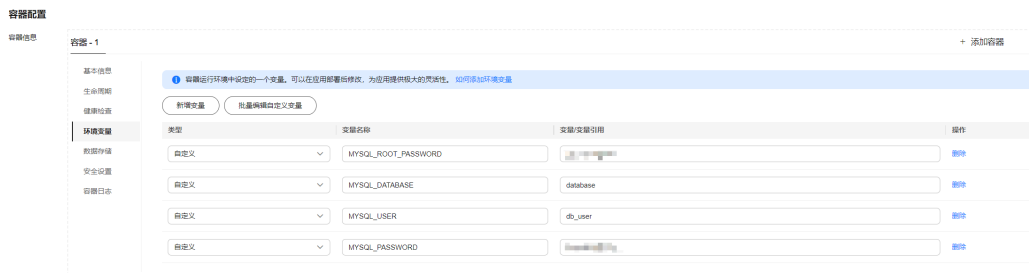
参数	示例	参数说明
负载类型	<b>有状态负载 StatefulSet</b>	工作负载是在 Kubernetes 上运行的应用程序。Kubernetes提供了一些内置的工作负载资源，不同工作负载的功能特性及应用场景不同。关于工作负载类型的介绍请参见 <a href="#">工作负载概述</a> 。
负载名称	<b>mysql</b>	请填写工作负载的名称。
命名空间	<b>default</b>	命名空间是Kubernetes集群中的抽象概念，可以将集群中的资源或对象划分为一个组，且不同命名空间中的数据彼此隔离。 集群创建后将默认生成一个default命名空间，您可以直接使用该命名空间。
实例数量	<b>1</b>	工作负载中的Pod数量。

### 步骤5 填写容器基本信息。



参数	示例	参数说明
镜像名称	使用 <b>mysql</b> 镜像，镜像版本为 <b>8.0</b>	在基本信息中单击“选择镜像”，在弹出的窗口中选择“镜像中心”，并搜索“mysql”，选择mysql镜像，并设置镜像版本为“8.0”。
CPU配额	申请值为 <b>0.25Core</b> ，限制值为 <b>0.25Core</b>	<ul style="list-style-type: none"> <li>● 申请：容器预分配的CPU值，默认0.25Core。</li> <li>● 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。</li> </ul> 详细说明请参见 <a href="#">设置容器规格</a> 。
内存配额	申请值为 <b>512MiB</b> ，限制值为 <b>512MiB</b>	<ul style="list-style-type: none"> <li>● 申请：容器预分配的内存值，默认512MiB。</li> <li>● 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。</li> </ul> 详细说明请参见 <a href="#">设置容器规格</a> 。

### 步骤6 在“环境变量”下添加如下环境变量，此处一共需要设置四个环境变量。关于MySQL支持的环境变量请参见[MySQL](#)。



环境变量	取值示例	说明
MYSQL_ROOT_PASSWORD	请您自行设置密码	MySQL的root用户密码，可自定义。
MYSQL_DATABASE	database	镜像启动时要创建的数据库名称，可自定义。
MYSQL_USER	db_user	数据库用户名称，可自定义。
MYSQL_PASSWORD	请您自行设置密码	数据库用户密码，可自定义。

**步骤7** 在“数据存储”下选择“动态挂载 (VolumeClaimTemplate)”，添加云硬盘存储作为MySQL的存储。

单击“创建存储卷声明PVC”，并填写以下关键参数，其余参数可保持默认。

#### 创建存储卷声明 PVC

存储卷声明类型  云硬盘  本地持久卷  专属存储

PVC 名称

创建方式  动态创建  静态创建   
创建底层存储会收取资源费用，并占用底层资源的配额。价格详情 [🔗](#)

存储类

存储卷名称前缀 (可选)    
不填写时默认值为"pvc"，实际创建的存储卷名称为存储卷名称前缀与PVC UID的拼接组合

可用区  可用区1 (1)  可用区2 (3)   
当前可用区有1台节点。云硬盘只能挂载到同一可用区的节点上，创建后不支持更换可用区，请谨慎选择。

云硬盘类型  高IO  通用SSD  超高IO  极速型SSD   
 通用型SSD V2  高IO增程版  通用型SSD增程版  超高IO增程版   
 极速型SSD增程版

容量 (GiB)

计费模式  按需计费  包年/包月

访问模式  ReadWriteOnce  ReadWriteMany  ReadOnlyMany

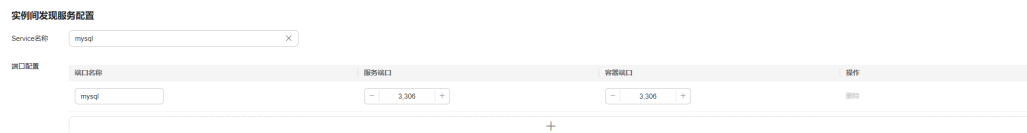
参数	示例	参数说明
存储卷声明类型	云硬盘	PVC使用的底层存储类型。
PVC名称	mysql	自定义PVC名称，如mysql。
存储类	csi-disk	默认为csi-disk。
可用区	可用区1	选择一个可用区，云硬盘只能挂载到同一可用区的节点上，创建后不支持更换可用区，请谨慎选择。
云硬盘类型	通用SSD	请根据需求自定义选择合适的云硬盘类型。
容量 (GiB)	10GiB	请根据需求填写容量，默认为10GiB。

单击“创建”，然后填写存储挂载到容器的路径，MySQL默认使用的路径为“/var/lib/mysql”。



### 步骤8 在“实例间发现服务配置”设置Headless Service。

Headless Service用于有状态负载实例间发现，它会生成每个Pod的集群DNS地址，可以实现对有状态负载某个特定实例的访问。对于多副本具有主副关系的MySQL数据库，可使用Headless Service对MySQL主服务器进行读写，并对其他副本进行数据复制。本示例中MySQL为单实例，关于多实例MySQL的示例请参见[运行一个有状态的应用程序](#)。



参数	示例	参数说明
Service名称	mysql	自定义Headless Service名称。
端口名称	mysql	自定义端口名称，用于区分同一个Service中的不同端口，本示例中仅使用一个端口。
服务端口	3306	Service对外提供访问的端口，可自定义设置，本示例中与容器端口保持一致。

参数	示例	参数说明
容器端口	3306	容器中的应用程序实际监听的端口，根据应用程序镜像对外开放的端口确定。例如MySQL对外开放3306端口。

**步骤9** 单击右下角“创建工作负载”。

等待工作负载创建成功，创建成功后在有状态负载下会显示一个运行中的工作负载。



----结束

## 使用 kubectl 命令行方式

### 须知

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** 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2** 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kubect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3 创建一个名为mysql.yaml的描述文件。其中，mysql.yaml为自定义名称，您可以随意命名。

```
vi mysql.yaml
```

文件内容如下：

```
apiVersion: apps/v1
kind: StatefulSet
metadata:
  name: mysql
  namespace: default
spec:
  replicas: 1
  selector:
    matchLabels:
      app: mysql
      version: v1
  template:
    metadata:
      labels:
        app: mysql
        version: v1
    spec:
      containers:
        - name: container-1
          image: mysql:8.0
          env:
            - name: MYSQL_ROOT_PASSWORD # MySQL的root用户密码，可自定义
              value: *****
            - name: MYSQL_DATABASE # 镜像启动时要创建的数据库名称，可自定义
              value: database
            - name: MYSQL_USER # 数据库用户名称，可自定义
              value: db_user
            - name: MYSQL_PASSWORD # 数据库用户密码，可自定义
              value: *****
          resources:
            requests:
              cpu: 250m
              memory: 512Mi
            limits:
              cpu: 250m
              memory: 512Mi
          volumeMounts:
            - name: mysql
              mountPath: /var/lib/mysql
          imagePullSecrets:
            - name: default-secret
      serviceName: mysql
      volumeClaimTemplates: #动态挂载云硬盘示例
        - apiVersion: v1
          kind: PersistentVolumeClaim
          metadata:
            name: mysql
            namespace: default
          annotations:
            everest.io/disk-volume-type: SSD # 云硬盘的类型
          labels:
            failure-domain.beta.kubernetes.io/region: ap-southeast-1 #云硬盘所在的区域
            failure-domain.beta.kubernetes.io/zone: #云硬盘所在的可用区，必须和工作负载部署的节点可用区一
```

```
致
spec:
  accessModes:
    - ReadWriteOnce # 云硬盘必须为ReadWriteOnce
  resources:
    requests:
      storage: 10Gi
      storageClassName: csi-disk # StorageClass的名称, 云硬盘为csi-disk
---
apiVersion: v1
kind: Service
metadata:
  name: mysql
  namespace: default
  labels:
    app: mysql
    version: v1
spec:
  selector:
    app: mysql
    version: v1
  clusterIP: None
  ports:
    - name: mysql
      protocol: TCP
      port: 3306
      targetPort: 3306
  type: ClusterIP
```

#### 步骤4 创建MySQL。

```
kubectl apply -f mysql.yaml
```

回显如下表示已开始创建负载。

```
statefulset "mysql" created
```

#### 步骤5 查看工作负载状态。

```
kubectl get statefulset
```

回显如下，表示工作负载已创建成功。

NAME	READY	AGE
mysql	1/1	4m5s

#### 步骤6 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。

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
kubernetes	ClusterIP	10.247.0.1	<none>	443/TCP	3d
mysql	ClusterIP	None	<none>	3306/TCP	51s

----结束

## 步骤五：部署 WordPress

您可以使用不同的方式部署WordPress工作负载。

### 使用控制台方式

步骤1 登录[CCE控制台](#)。

步骤2 单击集群进入集群控制台。



**步骤3** 在左侧菜单栏选择“工作负载”，单击右上角“创建负载”。

**步骤4** 填写工作负载基本信息。

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[创建有状态负载（StatefulSet）](#)。

**基本信息**

负载类型

 无状态负载  
Deployment

 有状态负载  
StatefulSet

 守护进程集  
DaemonSet

 普通任务  
Job

⚠️ 切换负载类型会导致已填写的部分关联数据被清空，请谨慎切换

负载名称

命名空间  [创建命名空间](#)

实例数量

参数	示例	参数说明
负载类型	<b>无状态负载 Deployment</b>	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 <a href="#">工作负载概述</a> 。
负载名称	<b>wordpress</b>	请填写工作负载的名称。
命名空间	<b>default</b>	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	<b>1</b>	工作负载中的Pod数量。

**步骤5** 填写容器基本信息。

**容器配置**

容器-1

**基本信息**

容器名称

镜像名称  [更换镜像](#)

镜像版本

CPU配额 申请  cores; 限制  cores

内存配额 申请  MiB; 限制  MiB

GPU配额 暂未安装GPU插件，无法使用此功能。 [点此安装插件](#)

NPV配额 暂未安装NPV插件，无法使用此功能。 [点此安装插件](#)

特权容器

初始化容器

限制策略  总是拉取镜像

参数	示例	参数说明
镜像名称	使用 <b>wordpress</b> 镜像，镜像版本为 <b>latest</b>	在基本信息中单击“选择镜像”，在弹出的窗口中选择“镜像中心”，并搜索“wordpress”，选择wordpress镜像，并设置镜像版本为“latest”。

参数	示例	参数说明
CPU配额	申请值为 <b>0.25Core</b> ，限制值为 <b>0.25Core</b>	<ul style="list-style-type: none"> <li>● 申请：容器预分配的CPU值，默认0.25Core。</li> <li>● 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。</li> </ul> 详细说明请参见 <a href="#">设置容器规格</a> 。
内存配额	申请值为 <b>512MiB</b> ，限制值为 <b>512MiB</b>	<ul style="list-style-type: none"> <li>● 申请：容器预分配的内存值，默认512MiB。</li> <li>● 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。</li> </ul> 详细说明请参见 <a href="#">设置容器规格</a> 。

**步骤6** 在环境变量下添加如下环境变量，此处一共需要设置四个环境变量，为WordPress添加MySQL数据库的信息。



环境变量	取值示例	说明
WORDPRESS_DB_HOST	<b>mysql:3306</b>	数据库的访问地址。本示例中需填写mysql工作负载的访问方式，即 <a href="#">步骤四：部署MySQL</a> 中的Headless Service，可以使用集群内部域名mysql.default.svc.cluster.local:3306访问，其中.default.svc.cluster.local可以省略，即使用 <b>mysql:3306</b> 。
WORDPRESS_DB_USER	<b>db_user</b>	访问数据的用户名，此处需要与 <a href="#">步骤四：部署MySQL</a> 中的 <b>MYSQL_USER</b> 参数值保持一致，即使用这个用户去连接MySQL。
WORDPRESS_DB_PASSWORD	<b>您自行设置的数据库密码</b>	访问数据库的密码，此处需要与 <a href="#">步骤四：部署MySQL</a> 中的 <b>MYSQL_PASSWORD</b> 参数值保持一致。
WORDPRESS_DB_NAME	<b>database</b>	访问数据库的名称，此处需要与 <a href="#">步骤四：部署MySQL</a> 中的 <b>MYSQL_DATABASE</b> 参数值保持一致。

**步骤7** 在“数据存储”下选择“已有存储卷声明 (PVC)”，添加云硬盘存储作为WordPress的存储。

单击“创建存储卷声明PVC”，并填写以下关键参数，其余参数可保持默认。

**创建存储卷声明 PVC** [YAML创建](#)

存储卷声明类型: **云硬盘** (文件存储, 对象存储, 极速文件存储, 本地持久卷, 专属存储)

PVC 名称:

命名空间: **default**

创建方式: **动态创建** (已有存储卷 PV, 新建存储卷 PV) <sup>①</sup>  
创建底层存储会收取资源费用, 并占用底层资源的配额。价格详情 [🔗](#)

存储类:

存储卷名称前缀 (可选):   
不填写时默认值为"pvc", 实际创建的存储卷名称为存储卷名称前缀与PVC UID的拼接组合

可用区: **可用区1 (1)** (可用区2 (3), 可用区3 (0), 可用区4 (0))  
当前可用区有1台节点。云硬盘只能挂载到同一可用区的节点上, 创建后不支持更换可用区, 请谨慎选择。

云硬盘类型: **通用SSD** (高IO, 超高IO, 极速型SSD, 通用型SSD V2, 高IO增强版, 通用型SSD增强版, 超高IO增强版, 极速型SSD增强版)

容量 (GiB):

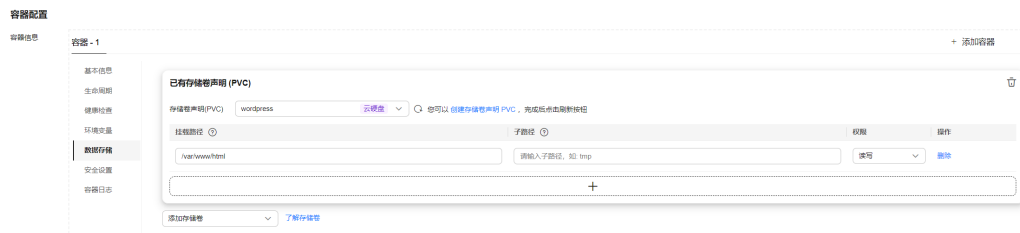
计费模式: **按需计费** (包年/包月)

访问模式: **ReadWriteOnce** <sup>②</sup>

参数	示例	参数说明
存储卷声明类型	<b>云硬盘</b>	PVC使用的底层存储类型。
PVC名称	<b>wordpress</b>	自定义PVC名称。
创建方式	<b>动态创建</b>	本例中选择“动态创建”，将自动为您创建存储卷声明PVC、存储卷PV和底层存储。适用于无可用的底层存储卷，需要新创建的场景。
存储类	<b>csi-disk</b>	默认为csi-disk。
可用区	<b>可用区1</b>	选择一个可用区，云硬盘只能挂载到同一可用区的节点上，创建后不支持更换可用区，请谨慎选择。
云硬盘类型	<b>通用SSD</b>	请根据需求自定义选择合适的云硬盘类型。

参数	示例	参数说明
容量 (GiB)	10GiB	请根据需求填写容量，默认为10GiB。

单击“创建”，然后填写存储挂载到容器的路径，WordPress默认使用的路径为“/var/www/html”。



### 步骤8 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。



参数	示例	参数说明
Service名称	wordpress	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 <a href="#">服务概述</a> 。
负载均衡器	<ul style="list-style-type: none"> <li>类型：选择<b>独享型</b>。</li> <li>可用区：至少选择一个可用区，如<b>可用区1</b>。</li> <li>弹性公网IP：选择<b>自动创建</b>。</li> </ul> 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 <a href="#">创建负载均衡类型的服务</a> 。
端口配置	<ul style="list-style-type: none"> <li>协议：<b>TCP</b></li> <li>容器端口：<b>80</b></li> <li>服务端口：<b>8080</b></li> </ul>	<ul style="list-style-type: none"> <li>协议：负载均衡监听器端口协议。</li> <li>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wordpress镜像请设置为80。</li> <li>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。</li> </ul>

**步骤9** 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。



----结束

## 使用 kubectl 命令行方式

**步骤1** 登录安装kubectl命令行工具的ECS虚拟机。

**步骤2** 创建一个名为wordpress-deployment.yaml的描述文件。其中，wordpress-deployment.yaml为自定义名称，您可以随意命名。

```
vi wordpress-deployment.yaml
```

文件内容如下：

```
apiVersion: apps/v1
kind: Deployment
```

```
metadata:
  name: wordpress
  namespace: default
spec:
  replicas: 1
  selector:
    matchLabels:
      app: wordpress
      version: v1
  template:
    metadata:
      labels:
        app: wordpress
        version: v1
    spec:
      containers:
        - name: container-1
          image: wordpress:latest
          env:
            - name: WORDPRESS_DB_HOST
              value: mysql:3306
            - name: WORDPRESS_DB_USER
              value: db_user
            - name: WORDPRESS_DB_PASSWORD
              value: ****
            - name: WORDPRESS_DB_NAME
              value: database
          resources:
            requests:
              cpu: 250m
              memory: 512Mi
            limits:
              cpu: 250m
              memory: 512Mi
          volumeMounts:
            - name: wordpress
              readOnly: false
              mountPath: /var/www/html
      imagePullSecrets:
        - name: default-secret
      volumes:
        - name: wordpress
          persistentVolumeClaim:
            claimName: wordpress
---
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: wordpress
  namespace: default
  annotations:
    everest.io/disk-volume-type: SSD
    everest.io/enterprise-project-id: '0'
  labels:
    failure-domain.beta.kubernetes.io/region: ap-southeast-1 #云硬盘所在的区域
    failure-domain.beta.kubernetes.io/zone: #云硬盘所在的可用区，必须和工作负载部署的节点可用区一致
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 10Gi
  storageClassName: csi-disk
```

### 步骤3 创建WordPress。

```
kubectl apply -f wordpress-deployment.yaml
```

查看工作负载状态。

```
kubectl get deployment
```

回显如下，表示工作负载已创建成功。

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

**步骤4** 创建一个名为wordpress-service.yaml的描述文件。其中，wordpress-service.yaml为自定义名称，您可以随意命名。

```
vi wordpress-service.yaml
```

文件内容如下：

```
apiVersion: v1
kind: Service
metadata:
  name: wordpress
  namespace: default
  annotations:
    kubernetes.io/elb.class: union
    kubernetes.io/elb.autocreate:
      {
        "type": "public",
        "bandwidth_name": "cce-wordpress",
        "bandwidth_chargemode": "bandwidth",
        "bandwidth_size": 5,
        "bandwidth_sharetype": "PER",
        "eip_type": "5_bgp"
      }
spec:
  selector:
    app: wordpress
  externalTrafficPolicy: Cluster
  ports:
    - name: cce-service-0
      targetPort: 80
      nodePort: 0
      port: 8080
      protocol: TCP
  type: LoadBalancer
```

**步骤5** 创建服务。

```
kubectl create -f wordpress-service.yaml
```

回显如下，表示服务已创建。

```
service/wordpress created
```

**步骤6** 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。您可通过LoadBalancer服务从公网访问WordPress工作负载。其中\*\*.\*\*.\*\*.\*\*为负载均衡实例的弹性公网IP地址，8080为访问端口。

```
NAME          TYPE          CLUSTER-IP    EXTERNAL-IP    PORT(S)          AGE
kubernetes    ClusterIP     10.247.0.1    <none>         443/TCP          3d
mysql         ClusterIP     10.247.202.20 <none>         3306/TCP         8m
wordpress    LoadBalancer 10.247.130.196 **.**.**.** 8080:31540/TCP  51s
```

----结束

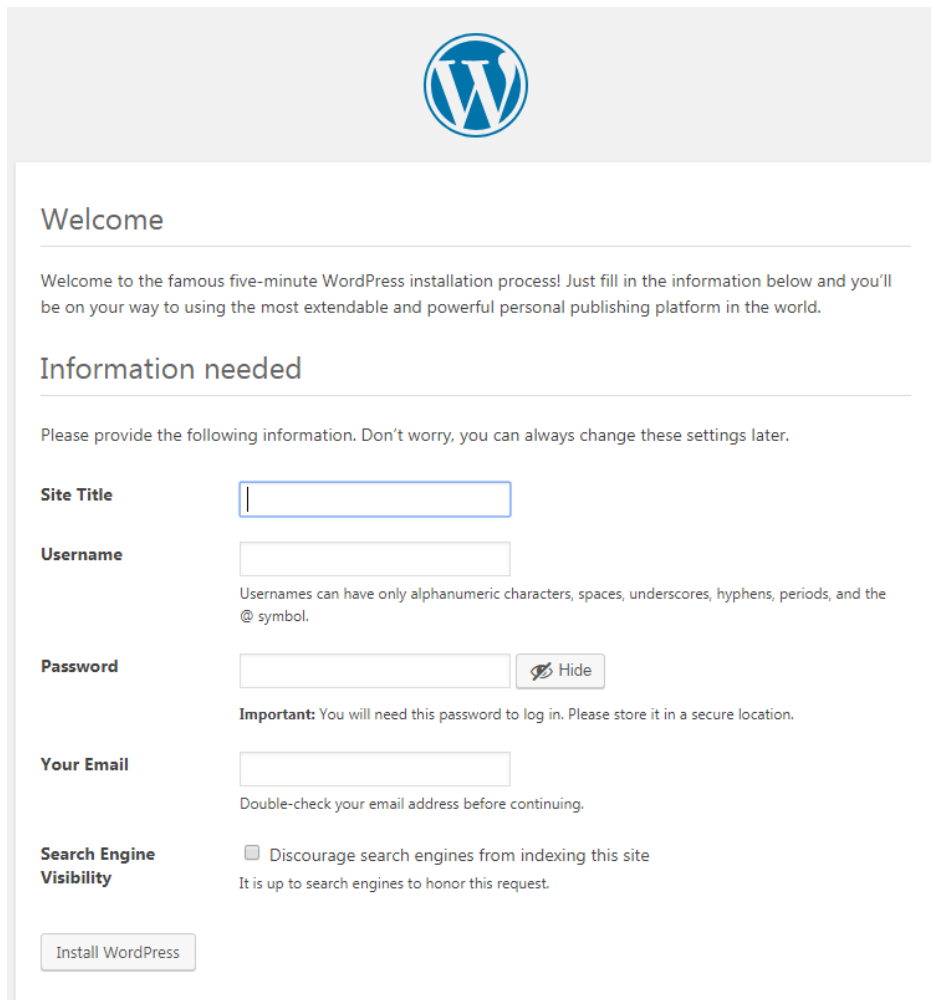
## 步骤六：访问 WordPress

**步骤1** 获取WordPress的外部访问地址。

单击WordPress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WordPress的IP地址，其中公网地址就是外部访问地址。



**步骤2** 在浏览器中输入“外部访问地址:端口”，即可成功访问应用，其中端口为**步骤8**中配置的服务端口值，为8080。



----结束



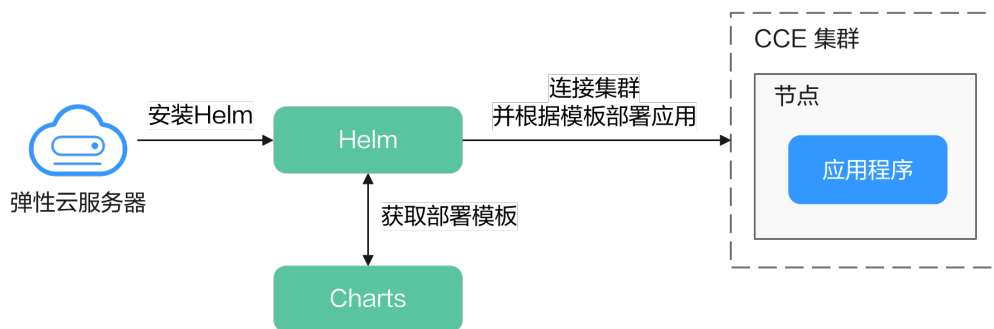
## 后续操作：释放资源

如果您无需继续使用集群，请及时释放资源，避免产生额外的费用。详细操作步骤请参见[删除集群](#)。

# 3 在 CCE 集群中通过 Helm 模板部署应用程序

Helm是一个Kubernetes应用程序包管理器，它可以简化部署、升级和管理Kubernetes应用程序的过程。Helm使用Charts（一种定义Kubernetes资源的打包格式）来封装Kubernetes部署的所有元素，包括应用程序代码、依赖项、配置文件和部署指令，使得复杂的Kubernetes应用能够以更加高效且一致的方式进行分发和部署。同时，Helm支持应用的升级和回滚，使得应用的生命周期管理更加容易。

本文介绍如何通过Helm部署WordPress应用。



## 操作流程

操作步骤	说明
<b>准备工作</b>	您需要注册华为账号，并为账户充值。
<b>步骤一：首次开通CCE并进行授权</b>	当您的账号在当前区域中首次使用CCE时，您需要为CCE进行授权。
<b>步骤二：创建集群</b>	您需要在CCE服务中创建一个集群，以提供Kubernetes服务。
<b>步骤三：在集群中创建节点池和节点</b>	您需在集群中创建节点，即可在节点上运行您的容器化应用。
<b>步骤四：使用kubectl命令行工具连接集群</b>	在使用Helm模板前，您需要在一台虚拟机上使用kubectl命令行工具连接刚刚创建的集群。
<b>步骤五：安装Helm</b>	在安装kubectl工具的虚拟机上，您需要继续安装Helm工具。

操作步骤	说明
<a href="#">步骤六：部署模板</a>	使用Helm安装命令在集群中创建WordPress工作负载，并为其创建一个可供公网访问的服务。
<a href="#">步骤七：访问WordPress</a>	您可以从公网访问刚刚创建的WordPress网站，开始您的博客之旅。
<a href="#">后续操作：释放资源</a>	如果您在完成实践后不需要继续使用集群，请及时清理资源以免产生额外扣费。


## 准备工作

- 在开始操作前，请您先注册华为账号并完成实名认证，详情请参见[注册华为账号并开通华为云](#)和[个人实名认证](#)。

## 步骤一：首次开通 CCE 并进行授权

由于CCE在运行中对计算、存储、网络以及监控等各类云服务资源都存在依赖关系，因此当您首次登录CCE控制台时，CCE将自动请求获取当前区域下的云资源权限，从而更好地为您提供服务。如果您在当前区域已完成授权，可忽略本步骤。

**步骤1** 使用华为账号登录[CCE控制台](#)。

**步骤2** 单击管理控制台左上角的，选择区域。

**步骤3** 在首次登录某个区域的CCE控制台时将跳出“授权说明”，请您在仔细阅读后单击“确定”。

当您同意授权后，CCE将在IAM中创建名为“cce\_admin\_trust”委托，统一对您的其他云服务资源进行操作，并且授予其Tenant Administrator权限。Tenant Administrator拥有除IAM管理外的全部云服务管理员权限，用于对CCE所依赖的其他云服务资源进行调用，且该授权仅在当前区域生效。您可前往“IAM控制台 > 委托”页签，单击“cce\_admin\_trust”查看各区域的授权记录。关于资源委托详情，您可参考[委托](#)进行了解。

### 说明

由于CCE对其他云服务有许多依赖，如果没有Tenant Administrator权限，可能会因为某个服务权限不足而影响CCE功能的正常使用。因此在使用CCE服务期间，请不要自行删除或者修改“cce\_admin\_trust”委托。

---结束

## 步骤二：创建集群

**步骤1** 登录[CCE控制台](#)。

- 如果您的账号还未创建过集群，请在引导页面中单击页面上方的“购买集群”。
- 如果您的账号已经创建过集群，请在左侧菜单栏选择集群管理，单击右上角“购买集群”。

**步骤2** 配置集群基础参数。

本示例中仅解释必要参数，其余大多数配置可保留默认值。关于配置参数的详细说明请参见[创建集群](#)。



参数	示例	参数说明
集群类型	<b>CCE Standard 集群</b>	<p>CCE支持多种类型的集群创建，以满足您各种业务需求，可为您提供高可靠、安全的商业级容器集群服务。</p> <p>根据需求选择“CCE Standard集群”或“CCE Turbo集群”。</p> <ul style="list-style-type: none"> <li>CCE Standard集群：标准版本集群，提供高可靠、安全的商业级容器集群服务。</li> <li>CCE Turbo集群：拥有更高性能的云原生网络，提供云原生混部调度能力，可实现更高的资源利用率和更广的全场景覆盖。</li> </ul> <p>关于集群类型差异详情，请参见<a href="#">集群类型对比</a>。</p>
计费模式	<b>按需计费</b>	<p>选择集群的计费模式。</p> <ul style="list-style-type: none"> <li>包年/包月：预付费模式，按订单的购买周期计费，适用于可预估资源使用周期的场景，价格比按需计费模式更优惠。选择该计费模式时，需要设置“购买时长”并选择是否自动续费（按月购买时自动续费周期为1个月，按年购买时自动续费周期则为1年）。</li> <li>按需计费：后付费模式，按资源的实际使用时长计费，可以随时开通/删除资源。</li> </ul> <p>更多计费相关信息请参见<a href="#">计费模式概述</a>。</p>
集群名称	<b>cce-test</b>	自定义新建集群的名称。
企业项目	<b>default</b>	<p>企业项目是一种云资源管理方式，企业项目管理服务提供统一的云资源按项目管理，以及项目内的资源管理、成员管理。了解更多企业项目相关信息，请查看<a href="#">企业管理</a>。</p> <p>该参数仅对开通企业项目的企业客户账号显示，不显示时请忽略。</p>

参数	示例	参数说明
集群版本	<b>选择推荐版本，如v1.29</b>	集群安装的Kubernetes软件版本，建议选择最新的商用版本，可为您提供更多稳定可靠的新特性。
集群规模	<b>50节点</b>	集群可以管理的最大节点数量，请根据业务场景选择。创建完成后支持扩容，不支持缩容。
集群 master 实例数	<b>3实例（高可用）</b>	<p>选择集群控制平面的节点（master实例）数量。控制平面节点由系统自动托管，会部署 Kubernetes 集群的管控面组件，如 kube-apiserver, kube-controller-manager, kube-scheduler 等组件。</p> <ul style="list-style-type: none"> <li>3实例（高可用）：创建3个控制平面节点，确保集群高可用。</li> <li>单实例：您的集群只会创建一个控制平面节点。</li> </ul> <p>集群创建完成后不支持修改。</p>

### 步骤3 配置集群网络参数。

The screenshot shows the 'Network Configuration' (网络配置) section of the CCE console. It includes the following settings:

- 集群网络配置 (Cluster Network Configuration):**
  - 虚拟私有云 (Virtual Private Cloud): vpc-ccc (192.168.0.0/16)
  - 子网 (Subnet): subnet-ccc (192.168.0.0/24)
  - 应用IPv6 (Application IPv6): Disabled
  - 默认安全组 (Default Security Group): Auto-generated
- 容器网络配置 (Container Network Configuration):**
  - 容器网络模型 (Container Network Model): VPC Network (Selected)
  - 容器网络 (Container Network): 10.0.0.0/16
  - 每个节点预留的容器IP个数 (Number of container IP addresses reserved per node): 128
- 服务网络配置 (Service Network Configuration):**
  - 服务网段 (Service CIDR): 10.247.0.0/16

参数	示例	参数说明
虚拟私有云	<b>vpc-ccc</b>	<p>为新建集群选择一个虚拟私有云。</p> <p>若没有可选虚拟私有云，请单击“新建虚拟私有云”进行创建，完成创建后单击刷新按钮。操作步骤详情请参见<a href="#">创建虚拟私有云和子网</a>。</p>

参数	示例	参数说明
子网	subnet-cce	选择一个子网，集群中的节点将会使用此子网中的IP。
容器网络模型	VPC网络	支持选择“VPC网络”和“容器隧道网络”，默认可选择“VPC网络”。 不同容器网络模型间的差异请参见 <a href="#">容器网络模型对比</a> 。
容器网段	10.0.0.0/16	设置容器使用的网段，该网段大小决定了集群中容器的数量上限。
服务网段	10.247.0.0/16	为集群配置ClusterIP地址的范围，该网段决定了Service资源数量的上限，且创建后不可修改。

**步骤4** 单击“下一步：插件选择”，选择创建集群时需要安装的插件。

本示例中，仅选择默认安装的必选插件。



**步骤5** 单击“下一步：插件配置”，默认插件可无需配置。

**步骤6** 单击“下一步：规格确认”，显示集群资源清单，确认无误后，单击“提交”。

等待集群创建成功，创建集群预计需要5-10分钟左右，请耐心等待。

创建成功后在集群管理下会显示一个运行中的集群，且集群节点数量为0。

**图 3-1 集群创建成功**



----**结束**

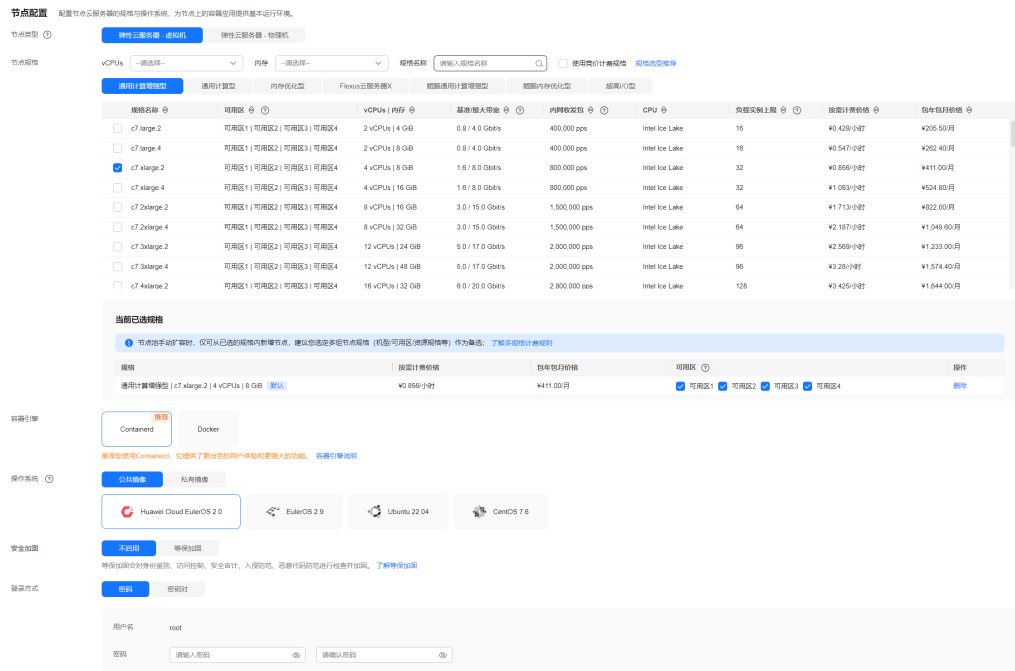
### 步骤三：在集群中创建节点池和节点

**步骤1** 单击新建的集群名称进入集群控制台。

**步骤2** 在左侧菜单栏中选择“节点管理”，单击右上角“创建节点池”。

**步骤3** 配置节点池参数。

本示例中仅解释必要参数，其余大多数配置可保留默认值。关于配置参数的详细说明请参见[创建节点池](#)。



参数	示例	参数说明
节点类型	弹性云服务器-虚拟机	请根据不同的业务诉求选择节点类型，“节点规格”列表中将自动为您筛选该类型下可部署容器服务的规格，供您进一步选择。
节点规格	4 vCPUs   8 GiB	请根据业务需求选择相应的节点规格。 为保证集群中的组件可以正常运行，建议的节点规格为：vCPU≥4核，内存≥8 GiB。
容器引擎	Containerd	请根据业务需要选择相应的容器引擎。不同的容器引擎差别请参见 <a href="#">容器引擎</a> 。
操作系统	Huawei Cloud EulerOS 2.0	请选择节点对应的操作系统。

参数	示例	参数说明
登录方式	选择“密码”方式登录，密码请您自行设置。	<ul style="list-style-type: none"> <li>选择“密码”：用户名默认为“root”，请输入登录节点的密码，并确认密码。请妥善保管密码，登录节点时需要使用该密码，系统无法获取您设置的密码内容。</li> <li>选择“密钥对”：在选项框中选择用于登录本节点的密钥对，并单击勾选确认信息。密钥对用于远程登录节点时的身份认证。若没有密钥对，可单击选项框右侧的“创建密钥对”来新建，创建密钥对操作步骤请参见<a href="#">创建密钥对</a>。</li> </ul>

**步骤4** 配置“存储配置”和“网络配置”，本示例中均可保持默认值，您只需要在“网络配置”中勾选确认节点池关联安全组可用，然后单击“下一步: 规格确认”。



**步骤5** 查看节点规格无误后，阅读页面上的使用说明，确认后单击“提交”。

**步骤6** 等待节点池创建成功后，节点池中默认节点数为0，您需要单击节点池所在行的“扩缩容”，为节点池创建节点。



**步骤7** 设置本次扩容的节点数量为2，即在节点池中创建2个节点。



**节点池扩容**

节点池名称: cce-test-pool1

当前数量: 0

扩缩容: 扩容节点 缩容节点

扩容规格: am7.large.8 | 可用区3  
使用选择的规格扩容节点, 如果规格资源不足会导致扩容失败

计费模式: 按需计费 包年/包月

本次扩容节点数:  + -  
当前集群管理规模(50节点)下还可以创建 48 个节点

**步骤8** 等待节点创建成功, 创建节点预计需要5-10分钟左右, 请耐心等待。



规格	可用区	状态	节点总数(实际期望)	包年包月节点数	按需节点数	弹性伸缩	操作
c7.large.4.12.xCPU.18.GB	可用区3	正常	2 / 2	0	2	关闭	节点列表 扩容

---结束

## 步骤四: 使用 kubectl 命令行工具连接集群

### 须知

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** 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2** 为kubectl命令行工具配置访问Kubernetes集群的凭证。

- 请登录**CCE控制台**, 并单击集群名称进入集群总览页。
- 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。
- 在弹出页面中选择“内网访问”, 然后下载对应的配置文件。
- 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配置文件(以kubeconfig.yaml为例)复制到/home目录下。

5. 将kubect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/coredns:dns/proxy
To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
```

----结束

## 步骤五：安装 Helm

本文以Helm v3.7.0为例进行演示。如需选择其他合适的版本，请访问<https://github.com/helm/helm/releases>。

- 步骤1 在连接集群的虚拟机上下载Helm客户端。

```
wget https://get.helm.sh/helm-v3.7.0-linux-amd64.tar.gz
```

- 步骤2 解压Helm包。

```
tar -xzvf helm-v3.7.0-linux-amd64.tar.gz
```

- 步骤3 将Helm复制到系统path路径下，以下为/usr/local/bin/helm。

```
mv linux-amd64/helm /usr/local/bin/helm
```

- 步骤4 查看Helm版本。

```
helm version
version.BuildInfo{Version:"v3.7.0",GitCommit:"eeac83883cb4014fe60267ec6373570374ce770b",GitTreeState:"
clean",GoVersion:"g01.16.8"}
```

----结束

## 步骤六：部署模板

本文将使用WordPress模板作为示例进行部署。

- 步骤1 添加WordPress官方Repo。

```
helm repo add bitnami https://charts.bitnami.com/bitnami
```

- 步骤2 执行如下命令，创建WordPress应用。

```
helm install myblog bitnami/wordpress \
--set mariadb.primary.persistence.enabled=true \
--set mariadb.primary.persistence.storageClass=csi-disk \
--set mariadb.primary.persistence.size=10Gi \
--set persistence.enabled=false
```

其中myblog为自定义的实例名称，其余参数功能如下：

- WordPress对接的Mariadb数据库使用持久化存储卷保存数据，持久化存储通过StorageClass自动化创建，选用云硬盘类型（csi-disk），大小为10GiB。
- WordPress不需要进行数据持久化，数据卷persistence.enabled配置为false。

上述命令执行后回显如下：

```
coalesce.go:223: warning: destination for mariadb.networkPolicy.egressRules.customRules is a table.
Ignoring non-table value ({})
```

```
NAME: myblog
LAST DEPLOYED: Mon Mar 27 11:47:58 2023
NAMESPACE: default
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
CHART NAME: wordpress
CHART VERSION: 15.2.57
APP VERSION: 6.1.1

** Please be patient while the chart is being deployed **

Your WordPress site can be accessed through the following DNS name from within your cluster:

    myblog-wordpress.default.svc.cluster.local (port 80)

To access your WordPress site from outside the cluster follow the steps below:

1. Get the WordPress URL by running these commands:

    NOTE: It may take a few minutes for the LoadBalancer IP to be available.
          Watch the status with: 'kubectl get svc --namespace default -w myblog-wordpress'

    export SERVICE_IP=$(kubectl get svc --namespace default myblog-wordpress --template "{{ range
(index .status.loadBalancer.ingress 0) }}{. . }}{{ end }}")
    echo "WordPress URL: http://$SERVICE_IP/"
    echo "WordPress Admin URL: http://$SERVICE_IP/admin"

2. Open a browser and access WordPress using the obtained URL.

3. Login with the following credentials below to see your blog:

    echo Username: user
    echo Password: $(kubectl get secret --namespace default myblog-wordpress -o
jsonpath="{.data.wordpress-password}" | base64 -d)

----结束
```

## 步骤七：访问 WordPress

### 步骤1 修改WordPress Service配置。

由于CCE中使用负载均衡Service需要额外的annotation配置，而bitnami/wordpress中并无此配置，因此需要手动修改。

```
kubectl edit svc myblog-wordpress
```

在metadata.annotations下添加kubernetes.io/elb.autocreate和kubernetes.io/elb.class两行并保存，如下所示。这两个annotation的作用是创建共享型负载均衡，这样就能通过负载均衡的弹性IP访问WordPress应用。

```
apiVersion: v1
kind: Service
metadata:
  name: myblog-wordpress
  namespace: default
  annotations:
    kubernetes.io/elb.autocreate: '{ "type": "public", "bandwidth_name": "myblog-wordpress",
"bandwidth_chargemode": "bandwidth", "bandwidth_size": 5, "bandwidth_sharetype": "PER", "eip_type":
"5_bgp" }'
    kubernetes.io/elb.class: union
spec:
  ports:
    - name: http
  ...
```

**步骤2 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。您可通过LoadBalancer服务从公网访问WordPress工作负载。其中\*\*.\*.\*.\*为负载均衡实例的弹性公网IP地址，80为访问端口。

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
kubernetes	ClusterIP	10.247.0.1	<none>	443/TCP	3d
myblog-mariadb	ClusterIP	10.247.202.20	<none>	3306/TCP	8m
<b>myblog-wordpress</b>	<b>LoadBalancer</b>	<b>10.247.130.196</b>	<b>**.*.*.*</b>	<b>80:31540/TCP</b>	<b>8m</b>

**步骤3 访问WordPress。**

- 访问WordPress网页：在浏览器地址栏中输入<负载均衡公网IP>:80访问WordPress网页。

User's Blog!

Sample Page

## Mindblown: a blog about philosophy.

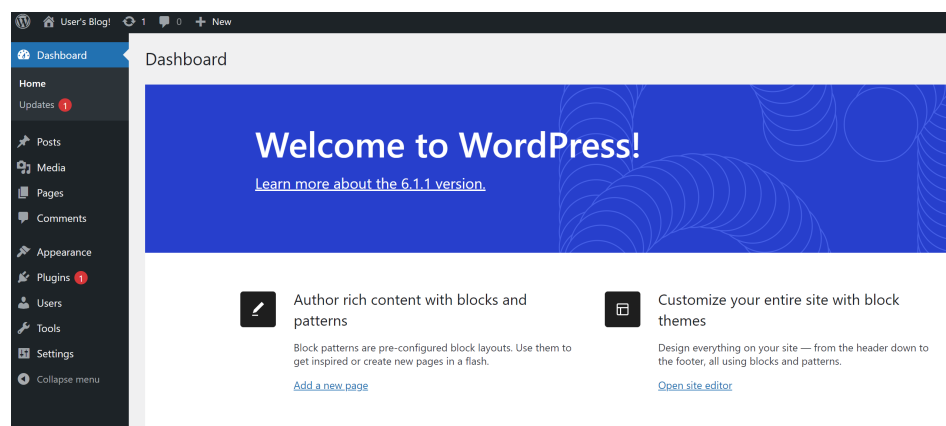
Hello world!

Welcome to WordPress. This is your first post.  
Edit or delete it, then start writing!

March 27, 2023

- 访问WordPress管理控制台：
  - 执行如下命令获取user用户的密码。

```
kubectl get secret --namespace default myblog-wordpress -o jsonpath="{.data.wordpress-password}" | base64 -d
```
  - 在浏览器地址栏中输入<负载均衡公网IP>:80/login 访问WordPress后台，用户为user，密码为上一步中获取的字符串。

**----结束**

## 后续操作：释放资源

如果您无需继续使用集群，请及时释放资源，避免产生额外的费用。详细操作步骤请参见[删除集群](#)。