

容器服务

快速入门

产品文档



腾讯云

【版权声明】

©2013-2024 腾讯云版权所有

本文档著作权归腾讯云单独所有，未经腾讯云事先书面许可，任何主体不得以任何形式复制、修改、抄袭、传播全部或部分本文档内容。

【商标声明】

及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。

【服务声明】

本文档意在向客户介绍腾讯云全部或部分产品、服务的当时的整体概况，部分产品、服务的内容可能有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

文档目录

快速入门

- 新手指引

- 快速创建一个标准集群

- 快速创建一个容器实例

入门示例

- 创建简单的 Nginx 服务

- 手动搭建 Hello World 服务

- 单实例版 WordPress

- 使用 TencentDB 的 WordPress

- 构建简单 Web 应用

快速入门 新手指引

最近更新时间：2023-05-22 17:27:45

本文将帮助您快速了解腾讯云容器服务（Tencent Kubernetes Engine，TKE），您可根据指引快速上手容器服务 TKE。

1. 什么是容器服务？

腾讯云容器服务（Tencent Kubernetes Engine，TKE）基于原生 kubernetes 提供以容器为核心的、高度可扩展的高性能容器管理服务，与腾讯云 IaaS 产品紧密结合，助力客户快速实现业务容器化，请参见 [产品概述](#)。

腾讯云容器服务支持您通过 [容器服务控制台](#) 或 [云 API](#) 操作集群与服务。

2. 了解容器服务的计费

腾讯云容器服务支持创建不同类型的 Kubernetes 集群，不同类型集群的计费项和计费标准不同。关于收费模式和具体价格，请参见 [购买指南](#)。

3. 使用容器服务

3.1 注册与认证

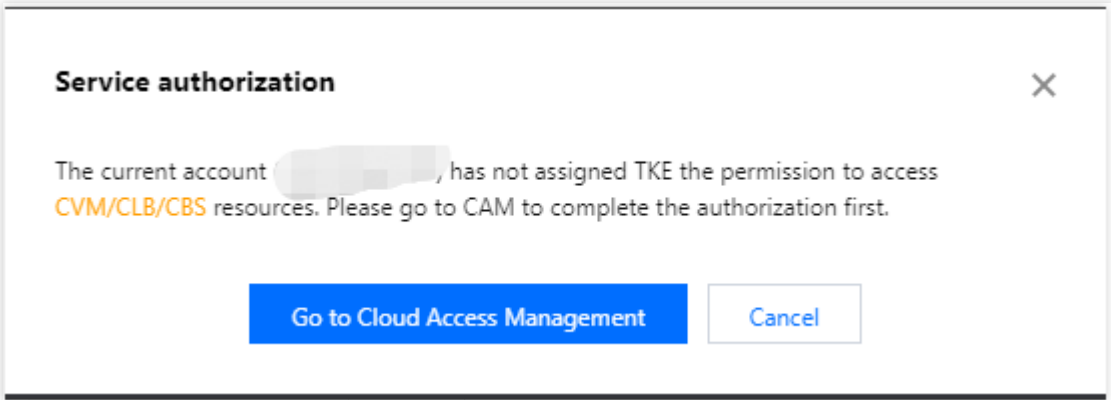
在使用腾讯云容器服务之前，您需要 [注册腾讯云账号](#) 并完成 [实名认证](#)。

3.2 角色授权

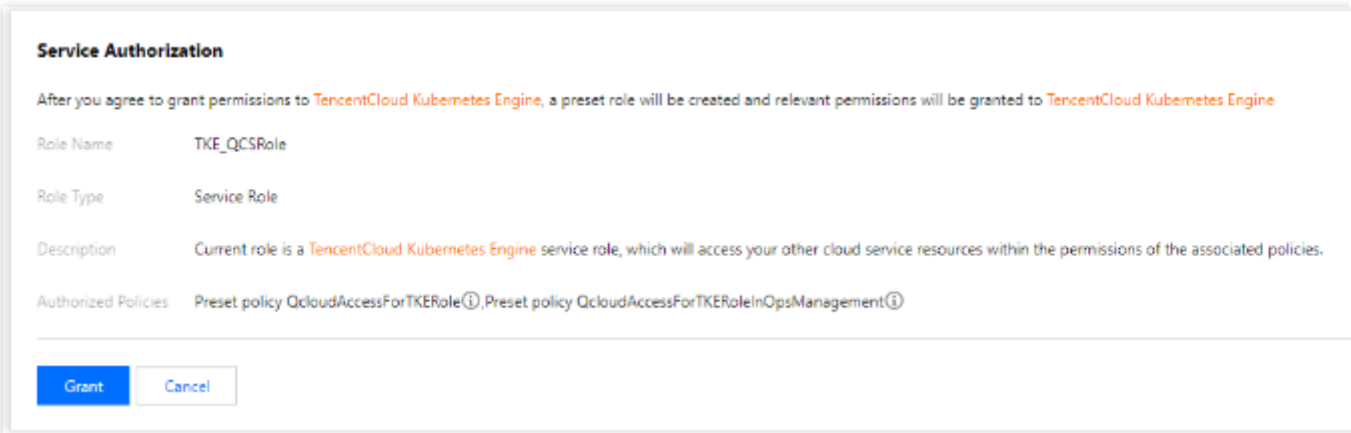
您需要为当前服务角色授权，赋予容器服务操作权限后才能正常地访问您的其他云服务资源。

在腾讯云控制台中，选择 [云产品 > 容器服务](#)，进入 [容器服务控制台](#)，按照界面提示为容器服务授权。完成服务授权获取相关资源操作权限，即可开始创建集群。操作步骤如下：

1. 查看弹出的“服务授权”窗口中的信息，并单击 [前往访问管理](#)。如下图所示：



2. 在“角色管理”页面，仔细阅读角色相关信息。如下图所示：



3. 单击**同意授权**即可完成授权操作，您可立即前往 [腾讯云容器服务购买页](#) 购买相关产品。

3.3 创建集群

您可参考 [快速创建一个标准集群](#) 文档，了解快速创建标准托管版集群的方法。可参考 [创建集群](#) 文档，了解创建标准托管版集群的完整过程。

如果您需要使用更多类型的集群，请参见 [创建 Serverless 集群](#)、[创建容器实例](#) 和 [创建边缘集群](#) 文档。

3.4 部署工作负载

容器服务支持使用镜像部署、YAML 文件编排两种方式部署工作负载。

如果您想通过镜像模板部署无状态负载（Deployment），操作详情请参见 [创建简单的 Nginx 服务](#) 或 [创建单实例版 WordPress](#)。

如果您想通过自定义镜像部署工作负载，操作详情请参见 [手动搭建 Hello World 服务](#)。

3.5 集群运维

容器服务 TKE 作为集群、应用、存储、网络等模块的管理平台，如果您需获取更多信息或需进行更多实践操作，请参考以下内容进一步了解并使用。

如果您想	您可以阅读
通过 Kubernetes 命令行工具 Kubectl 从本地客户端机器连接到 TKE 集群	连接集群

对运行中的 Kubernetes 集群进行升级	升级集群
在已创建的 Kubernetes 集群中添加实例	新增节点
管理 Kubernetes 集群内节点	创建节点池
通过控制台直接操作原生 Kubernetes 对象	Kubernetes 对象管理
采用 Service 方式为一组容器提供固定的访问入口	Service 基本功能
通过 Ingress 资源来配置不同的转发规则	Ingress 管理
使用容器服务的存储能力	存储管理概述
为集群内容容器分配在容器网络地址范围内的 IP 地址	容器网络概述
对 Kubernetes 集群内服务日志进行存储和分析	日志采集
对集群进行监控	监控实例管理
使用容器镜像服务内托管的私有镜像进行应用部署	使用 TCR 企业版实例内容容器镜像创建工作负载

4. 入门必读

我想在基础网络中使用 TKE，可以吗？

容器服务当前仅支持私有网络，不支持基础网络。

是否可以选择已有的云服务器加入集群？

支持，可以创建集群完成后添加已有云服务器。详情请参见 [添加已有节点](#)。

为什么我的服务一直在启动中？

服务内容容器若无持续运行的进程，会导致服务一直处于启动中，更多服务启动的问题见 [事件常见问题](#)。

创建集群前如何规划网络？

创建集群时，集群网络和容器网络网段不能重叠。通常情况下，您可以选择私有网络 VPC 中的子网用于集群的节点网络。详情见 [容器网络与集群网络说明](#)。

创建好的服务如何访问？

不同的访问方式提供不同的访问入口，详情见 [服务访问方式](#)。

容器怎么访问公网？

若容器所在主机有公网 IP 和带宽，则容器可直接访问外网，若容器所在主机无公网 IP 和带宽，则可以通过 NAT 网关访问外网。

我不会制作镜像，是否可以使用容器服务 TKE？

TKE 集成的 Helm 3.0 相关功能，为您提供创建 helm chart、容器镜像、软件服务等各种产品和服务的能力。已创建

的应用将在您指定的集群中运行，为您带来相应的能力。详情见 [应用管理](#)。

我的业务需要配置很多文本或环境变量，该怎么管理？

您可以通过 [配置项](#) 来管理配置文件。

服务之间想互访该怎么办？

集群下相同 Namespace 的服务可以直接相互访问，不同 Namespace 的服务需要通过 `<service-name>.<namespace-name>.svc.cluster.local` 的形式来访问。

5. 反馈与建议

使用腾讯云容器服务产品和服务中有任何问题或建议，您可以通过以下渠道反馈，将有专人跟进解决您的问题：

如果发现产品文档的问题，如链接、内容、API 错误等，您可以单击文档页右侧**文档反馈**或选中存在问题的内容进行反馈。

如果遇到产品相关问题，您可通过 [提交工单](#) 寻求帮助。

快速创建一个标准集群

最近更新时间：2023-09-26 15:36:36

本文向您介绍如何使用容器服务快速创建一个容器集群。

步骤1：注册腾讯云账号

在使用腾讯云容器服务之前，您需要 [注册腾讯云账号](#) 并完成 [实名认证](#)。

步骤2：在线充值

腾讯云容器服务 TKE 针对不同规格的托管集群，会收取相应的集群管理费用，以及用户实际使用的云资源费用。关于收费模式和具体价格，请参阅 [容器服务计费概述](#)。本文中我们创建的是“托管集群”，该模式下您依然要为集群的工作节点、持久化存储以及服务绑定的负载均衡等服务付费。购买前，需要在账号中进行充值。具体操作请参考 [在线充值](#) 文档。

步骤3：服务授权

在 [腾讯云控制台](#) 中，选择云产品 > **容器服务**，进入容器服务控制台，按照界面提示为容器服务授权。（如果您已为容器服务授权，请跳过该步骤。）

步骤4：新建集群

登录 [容器服务控制台](#)，开始创建集群。

集群信息

在**集群信息**页面，填写**集群名称**、选择**集群所在地域**、选择**集群网络**和**容器网络**，保持其他默认选项，并单击**下一步**。如下图所示：

1 Cluster information

2 Select model

3 CVM configuration

4 Component configurations

5 Confirm information

1

To use TKE, you need to create a cluster. A cluster consists several nodes (CVMs) on which services are running. To learn more, please see [Cluster Overview](#).

Cluster name

Enter the cluster name (up to 50 c

CPU architecture

X86 clusterARM cluster

Project of new-added resource

DEFAULT PROJECT

New added resources (CVM, CLB) will be allocated to this project automatically.[Instruction](#)

Kubernetes version

1.24.4

The super node is supported in clusters of v1.18, v1.20, and v1.22.
From January 4, 2023 (UTC +8), v1.16.3 is discontinued officially. For more information, see [Version Maintenance Mechanism](#).

Runtime components

containerdSuggestions

Select Containerd for the runtime when creating a node in a Kubernetes 1.24 cluster. Images built with Docker can still be used.
containerd is a more stable runtime component. It supports OCI standard and does not support docker API.

Region

GuangzhouShenzhenQingyuanShanghaiJinan ecHangzhou ecNanjingFuzhou ecHefei ecBeijingShijiazhuang ec
Wuhan ecChangsha ecChongqingChengduXi'an ecShenyang ecHong Kong, ChinaTaiwan, ChinaTorontoSeoulTokyo
SingaporeBangkokJakartaSilicon ValleyFrankfurtNortheastern EuropeMumbaiVirginiaSão Paulo

Tencent Cloud resources in different regions cannot communicate via private network. The region cannot be changed after purchase. Please choose a region close to your end-users to minimize access latency and improve download speed.

Cluster network

CIDR: 10.0.0.0/16

If the current networks are not suitable, please [create a VPC](#).

Container network add-on

Global RouterVPC-CNICilium-OverlaySuggestions

Developed by TKE, Global Router is a container network plugin based on VPC routing. It can be used to create a container IP range that parallelized to VPC.

Container network

CIDR: 172.16.0.0/16Instruction

Conflicts with CIDR blocks of other clusters in the same VPC CIDR_CONFLICT_WITH_OTHER_CLUSTER [cidr 172.16.0.0/16 is conflict with cluster id: cls-5u97apjy]

It cannot be modified after the creation.

Pod allocation mode

Max Pods per node64

Max Services in the cluster1024

Under the current container network configuration, the cluster can have a maximum of 1008 nodes.
You cannot modify max Pods per node and max Services in the cluster after creating them.

Image provider

Public imageMarketplace

Operating system

TencentOS Server 3.1 (TK4)Choosing an image (TencentOS Server is recommended)

Cancel

Next

集群名称：输入要创建的集群名称。本文以“test”为例。

所在地域：选择与您最近的一个地区，例如我在“深圳”，地域选择“广州”。

集群网络：为集群内主机分配在节点网络地址范围内的 IP 地址。这里我们选择已有的 VPC 网络。

容器网络：为集群内容器分配在容器网络地址范围内的 IP 地址。这里我们选择可用的容器网络。

选择机型

在**选择机型**页面，确认**计费模式**、选择**可用区**及**对应的子网**、**确认节点的机型**，并单击**下一步**。如下图所示：

Node source: [Add node](#) Existing nodes

Cluster type: [Managed cluster](#) Self-deployed cluster

The Master components and Etcd components of the cluster are managed and operated by Tencent Cloud. For more information, see [Cluster Hosting Mode Instruction](#).

Cluster specification: [L5](#) L20 L50 L100 L200 L500 L1000 L3000 L5000

Up to 5 nodes, 150 Pods, 128 ConfigMap and 150 CRDs are allowed under the current cluster specification. Please read [Choosing Cluster Specification](#) carefully before you make the choice. You can adjust the cluster specification manually, or enable Auto Cluster Upgrade to have it adjusted automatically.

☒ **Enable Auto Cluster Upgrade**

After the feature is enabled, it upgrades the cluster specification automatically when the load on control plane components reaches the threshold or the number of nodes reaches the upper limit. You can check details of configuration modification on the cluster details page. During the upgrade, the management plane (master node) components are updated on a rolling basis, which may cause temporary disruption. We recommend that you stop other operations (such as creating a workload) during the period.

Billing mode: [Pay-as-you-go](#)

Worker node configurations

Availability zone: [Guangzhou Zone 3](#) [Guangzhou Zone 4](#) [Guangzhou Zone 5](#) [Guangzhou Zone 6](#) [Guangzhou Zone 7](#)

Node network: [242/253 subnet IPs available](#)

CIDR: 10.0.0.0/16

If the current networks are not suitable, please go to the console to [create a VPC](#) or [create a subnet](#).

Model: SA2.MEDIUM2(Standard SA2, 2 core 2GB)

System disk: [Balanced SSD 50GB](#)

Data disk: [Purchase later](#)

Public network bandwidth: [Bill by traffic usage 1Mbps](#)

Node name: [Auto-generated](#)

CVM quantity: [-](#) 1 [+](#)

VPC network limit: Up to 242 IPs available for current node network

[Advanced settings](#)

[Confirm](#) [Cancel](#)

[Add model](#)

[Previous](#) [Next](#)

节点来源：提供**新增节点**和**已有节点**两个选项。这里我们选择“新增节点”。

集群类型：提供**托管集群**和**独立集群**两种集群模式选择。这里我们选择“托管集群”。

集群规格：提供多种集群规格。这里我们选择“L5”。

计费模式：提供**按量计费**的计费模式。

Worker 配置：该模块下只需选择可用区及对应的子网并确认节点的机型，其他设置项保持默认。

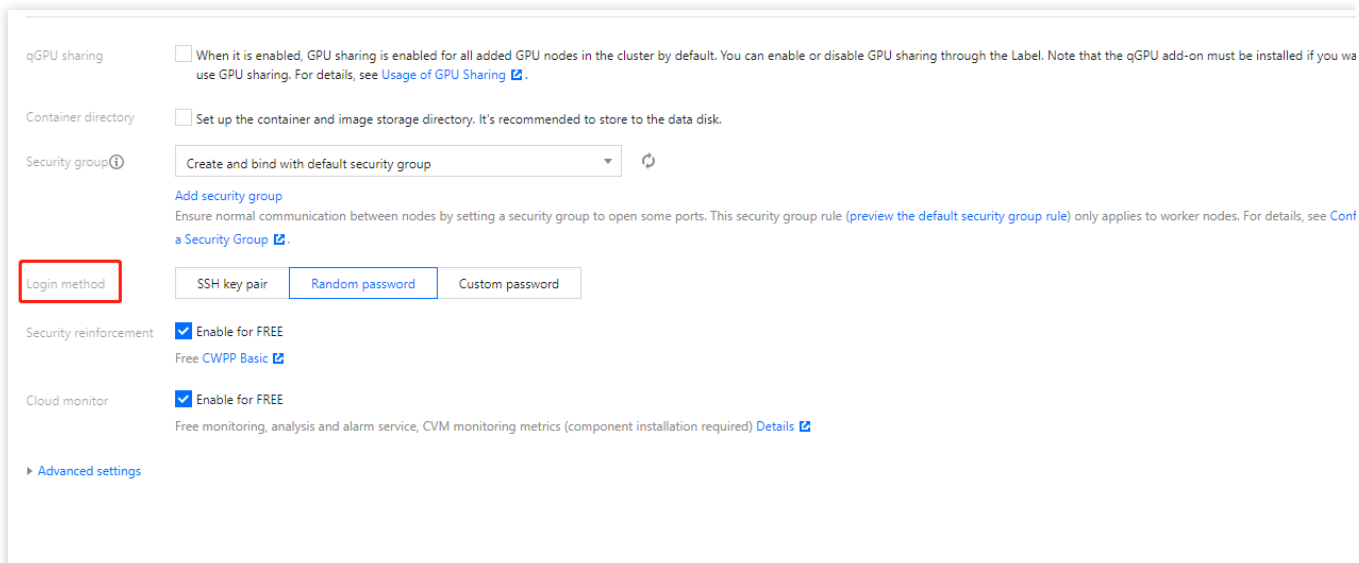
可用区：这里我们选择“广州六区”。

节点网络：这里我们选择当前 VPC 网络下的子网。

机型：这里我们选择“SA2.MEDIUM2(标准型SA2,2核2GB)”。

云服务器配置

在**云服务器配置**页面，选择登录方式，其他设置项保持默认，并单击**下一步**。如下图所示：



qGPU sharing ☐ When it is enabled, GPU sharing is enabled for all added GPU nodes in the cluster by default. You can enable or disable GPU sharing through the Label. Note that the qGPU add-on must be installed if you want to use GPU sharing. For details, see [Usage of GPU Sharing](#).

Container directory ☐ Set up the container and image storage directory. It's recommended to store to the data disk.

Security group 

[Add security group](#)
Ensure normal communication between nodes by setting a security group to open some ports. This security group rule ([preview the default security group rule](#)) only applies to worker nodes. For details, see [Configure a Security Group](#).

Login method

Security reinforcement ☒ Enable for FREE
Free CWPP Basic [Details](#)

Cloud monitor ☒ Enable for FREE
Free monitoring, analysis and alarm service, CVM monitoring metrics (component installation required) [Details](#)

[Advanced settings](#)

登录方式：提供**立即关联密钥**、**自动生成密码**和**设置密码**三种登录方式。这里我们选择“自动生成密码”。

组件配置

在**组件配置**页面，组件包含存储、监控、镜像等，您可按需选择。若无需安装，可单击**下一步**。这里我们选择不安装组件，其他设置项保持默认。

信息确认

在“信息确认”页面，确认集群的已选配置信息和费用。当您阅读并同意容器服务服务等级协议后，单击**完成**。如下图所示：

Selected configuration

Cluster name	test
Region	South China(Guangzhou)
Container network	GR...service/cluster, 64 Pod/node, up to 1008 nodes
Operating system	TencentOS Server 3.1 (TK4)
Cluster type	Managed cluster
Billing mode	Pay-as-you-go
Operating system ⓘ	TencentOS Server 3.1 (TK4) Public image -Basic image
Worker node	AZ:Guangzhou Zone 6 Model:SA2.MEDIUM2(Standard SA2,2 core2GB) System disk:Balanced SSD 50GB Data disk: purchase later Public bandwidth:Bill by traffic usage 1Mbps Amount:1
Add-on	CBS Tencent Cloud CBS
Cluster Auditing	Disabled
TMP	-

Fees

Terms of Service ☒ I have read and agree to TKE Service Level Agreement [🔗](#).

当您付费完成后，即可创建您的第一个集群。接下来，您可以在 [容器服务控制台](#) 查看您已创建的 TKE 标准集群。

步骤5：查看集群

创建完成的集群将出现在 [集群列表](#) 中。您可单击集群 ID 进入集群详情页面。在集群的**基本信息**页面中，您可查看集群信息、节点和网络信息、集群 API Server 信息等。如下图所示：

The screenshot displays the 'Basic information' page for a cluster named 'test(cis-fgfnr0k2)' in the 'Guangzhou' region. The page is divided into two main sections: 'Cluster information' and 'Node and Network Information'.

Cluster information:

- Cluster name: test
- Cluster ID: cis-fgfnr0k2
- Deployment type: Managed cluster
- Status: Running
- Region: South China(Guangzhou)
- Project of new-added resource: DEFAULT PROJECT
- Cluster specification: L5
- Auto Cluster Upgrade: Enabled
- Kubernetes version: Master 1.24.4-tke.5 (Latest version)
- Node 1.24.4-tke.5
- Runtime components: containerd 1.6.9
- Cluster description: N/A
- Tencent Cloud tags: -
- Deletion Protection: Enabled
- Time created: 2023-03-06 11:52:32

Node and Network Information:

- Number of nodes: 1
- Default OS: tlinux3.1x86_64
- qGPU sharing: Disabled
- System image source: Public image - Basic image
- Node hostname naming rule: Auto-generated
- Node network: vpc-5cu6x4bz
- Container network add-on: Global Router
- Container network: CIDR block
- Network mode: cni
- VPC-CNI mode: Disabled
- Service CIDR block: -
- Kube-proxy proxy mode: iptables

步骤6：删除集群

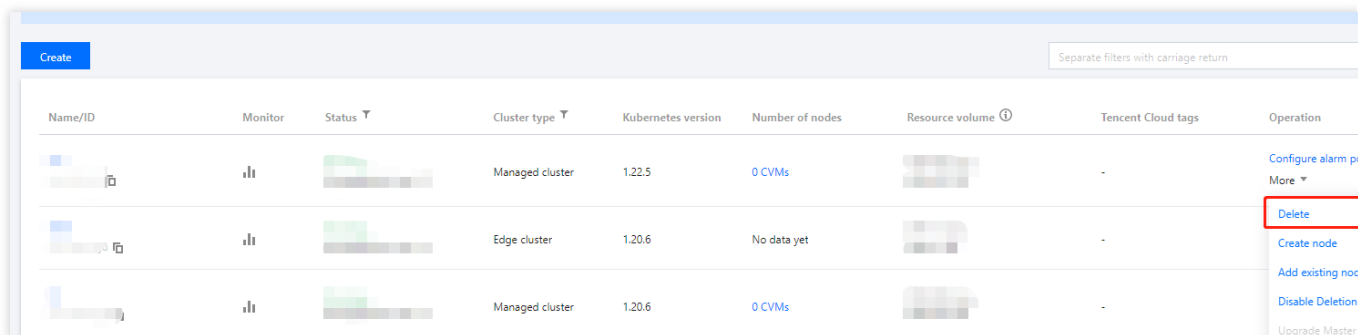
集群启动后即开始消耗资源，为避免产生不必要的费用，此步骤向您介绍如何清除所有资源。

1. 选择左侧导航栏中的**集群**，在“集群管理”页面选择需删除集群所在行右侧**更多 > 关闭集群删除保护**。如下图所示：

The screenshot displays the 'Cluster Management' page in the Tencent Cloud Container Service console. It features a table with columns for Name/ID, Monitor, Status, Cluster type, Kubernetes version, Number of nodes, Resource volume, Tencent Cloud tags, and Operation. The 'Operation' column for the first cluster (Managed cluster) shows a dropdown menu with options: Configure alarm p, More, Delete, Create node, Add existing noc, and Disable Deletion (highlighted with a red box).

Name/ID	Monitor	Status	Cluster type	Kubernetes version	Number of nodes	Resource volume	Tencent Cloud tags	Operation
[Cluster Name]	[Monitor Icon]	Running	Managed cluster	1.22.5	0 CVMs	[Resource Volume]	-	Configure alarm p More Delete Create node Add existing noc Disable Deletion
[Cluster Name]	[Monitor Icon]	Running	Edge cluster	1.20.6	No data yet	[Resource Volume]	-	
[Cluster Name]	[Monitor Icon]	Running	Managed cluster	1.20.6	0 CVMs	[Resource Volume]	-	

2. 选择集群所在行右侧**更多 > 删除**。如下图所示：



3. 在删除集群弹窗中确认信息后，单击**确定**即可删除集群。

后续操作：使用集群

通过本文，您已经了解如何在腾讯云容器服务中创建和删除集群。在已创建的集群中，您可以设置工作负载和创建服务。常用的任务包括：

[创建简单的 Nginx 服务](#)

[单实例版 WordPress](#)

[使用 TencentDB 的 WordPress](#)

[手动搭建 Hello World 服务](#)

[构建简单 Web 应用](#)

遇到问题？

使用容器服务控制台创建标准集群的完整步骤请参考 [创建集群](#)。如果您在使用时遇到问题，您可以 [联系我们](#) 来寻求帮助。

快速创建一个容器实例

最近更新时间：2022-12-13 17:16:57

步骤1：注册和充值

1. [注册腾讯云账号](#)，并进行实名认证。
如果您已在腾讯云注册，可忽略此步骤。
2. [在线充值](#)。
TKE Serverless 集群提供两种类型的计费模式：按量计费和预留券。购买容器实例前，需要在账号中进行充值。
具体操作请参考 [在线充值](#) 文档。

步骤2：服务授权

1. 容器实例目前正在内测中，您可以 [提交工单](#) 申请使用。
2. 开通白名单后，在容器实例控制台中按照页面提示为容器实例授权。（如果您已为容器实例授权，请跳过该步骤。）

步骤3：快速创建一个容器实例

登录容器实例控制台，在“快速创建实例”页面，参考以下提示进行配置。

配置项	说明
所在地域	选择与您最近的一个地区，例如我在“深圳”，地域选择“广州”。
容器网络	为容器实例分配在容器网络地址范围内的 IP 地址。 注意 子网决定了可用区，每个可用区所支持的资源类型不同，例如 AMD、GPU-T4、GPU-V100，请根据提示选择支持所需资源类型的子网。
安全组	安全组具有防火墙的功能，可限制容器实例的网络通信，默认为 default。
实例规格	实例支持的规格请参见 资源规格 。

镜像	支持选择容器镜像服务企业版、容器镜像服务个人版、Dockerhub 及其他第三方镜像仓库的镜像。						
镜像版本（Tag）	不填默认为 latest。						
镜像仓库凭证	当选择除 Dockerhub 以外的第三方镜像时，必须填写镜像凭证，即仓库的访问地址、用户名及密码。						
数据卷（选填）	为容器提供存储，目前支持文件存储 NFS、腾讯云硬盘 CBS，还需挂载到容器的指定路径中。 <table><tr><th>数据卷类型</th><th>描述</th></tr><tr><td>云硬盘 CBS</td><td>您可以指定一块腾讯云 CBS 云硬盘挂载到容器的某一路径下，当容器迁移时，云硬盘会随之迁移。使用云硬盘数据卷适用于数据的持久化保存，可用于 Mysql 等有状态服务。设置云硬盘数据卷的服务，实例数量最大为1。</td></tr><tr><td>文件存储 NFS</td><td>只需填写 NFS 路径，您可以使用腾讯云文件存储 CFS，也可使用自建的文件存储 NFS。使用 NFS 数据卷适用于多读多写的持久化存储，也适用于大数据分析、媒体处理、内容管理等场景。</td></tr></table>	数据卷类型	描述	云硬盘 CBS	您可以指定一块腾讯云 CBS 云硬盘挂载到容器的某一路径下，当容器迁移时，云硬盘会随之迁移。使用云硬盘数据卷适用于数据的持久化保存，可用于 Mysql 等有状态服务。设置云硬盘数据卷的服务，实例数量最大为1。	文件存储 NFS	只需填写 NFS 路径，您可以使用腾讯云文件存储 CFS，也可使用自建的文件存储 NFS。使用 NFS 数据卷适用于多读多写的持久化存储，也适用于大数据分析、媒体处理、内容管理等场景。
数据卷类型	描述						
云硬盘 CBS	您可以指定一块腾讯云 CBS 云硬盘挂载到容器的某一路径下，当容器迁移时，云硬盘会随之迁移。使用云硬盘数据卷适用于数据的持久化保存，可用于 Mysql 等有状态服务。设置云硬盘数据卷的服务，实例数量最大为1。						
文件存储 NFS	只需填写 NFS 路径，您可以使用腾讯云文件存储 CFS，也可使用自建的文件存储 NFS。使用 NFS 数据卷适用于多读多写的持久化存储，也适用于大数据分析、媒体处理、内容管理等场景。						
环境变量（选填）	支持配置容器的环境变量。						
实例数量（选填）	支持一次创建多个实例。当数据卷类型选择云硬盘 CBS 时，只能创建一个副本。						

完成所需配置项后，确认资源规格及配置费用，单击**创建实例**即可。接下来，您可以查看您已创建的容器实例。

步骤4：查看容器实例事件

- 方式1
 - 方式2
- 登录容器实例控制台。
 - 在容器实例列表页中，单击需要查看事件的实例右侧的**更多 > 查看事件**。

步骤5：查看容器日志

- 方式1
- 方式2

1. 登录容器实例控制台。
2. 在容器实例列表页中，单击需要查看事件的实例右侧的**日志**。

此处只可查看容器标准输出日志，关于标准输出日志及容器文件日志的采集，详情请参见 [配置日志采集](#)。

入门示例

创建简单的 Nginx 服务

最近更新时间：2023-05-06 19:41:07

操作场景

本文档旨在帮助您了解如何快速创建一个容器集群内的 Nginx 服务。

前提条件

已 [注册腾讯云账户](#)。

已创建集群。操作详情请参见 [创建集群](#)。

操作步骤

创建 Nginx 服务

1. 登录容器服务控制台，选择左侧导航栏中的 [集群](#)。
2. 在**集群管理**页面，选择需创建服务的集群 ID，进入集群的基本信息页面。
3. 在**工作负载 > Deployment**页面，单击**新建**。参数详情见 [创建 Deployment](#)。
4. 在**新建 Deployment** 页面，根据以下信息，设置工作负载基本信息。如下图所示：

The screenshot shows a configuration form for a container. It includes fields for Name, Description, Namespace, Labels, OS type, and Volume (optional). The Name field has a placeholder "Please enter a name" and a note: "Up to 63 characters, including lowercase letters, numbers, and hyphens ('-'). It must begin with a lowercase letter, and end with a number or lowercase letter." The Description field has a placeholder "Up to 1000 characters". The Namespace field is set to "default". The Labels field shows "k8s-app" = "nginx" with an "Add" button and a note: "The key name cannot exceed 63 chars. It supports letters, numbers, '/' and '-'. '/' cannot be placed at the beginning. A prefix is supported. Learn more". The OS type field is set to "Linux" with a refresh button and a note: "Configurations are initialized when you change the OS type for the container." The Volume (optional) field has a link "Add volume" and a note: "It provides storage for the container. It can be a node path, cloud disk volume, file storage NFS, config file and PVC, and must be mounted to the specified path of".

工作负载名称：本文以 nginx 为例。

描述：填写工作负载的相关信息。

标签：本例中标签默认值为 `k8s-app = nginx`。

命名空间：根据实际需求进行选择。默认为 `default`。

数据卷：根据实际需求设置工作负载的挂载卷，详情见 [存储卷使用说明](#)。

5. 参考以下信息设置“实例容器”。如下图所示：

The screenshot shows the "Containers in the Pod" configuration form. It includes fields for Name, Image, Image tag, Pull image from remote registry, Environment variable, CPU/memory limit, GPU resource, and Target port. The Name field is set to "test" with a note: "Up to 63 characters. It supports lower case letters, numbers, and hyphen ('-') and cannot start or end with '-'." The Image field is set to "nginx" with a "Select image" button. The Image tag field has a placeholder "latest" is used if it's left empty. The Pull image from remote registry field has buttons for "Always", "IfNotPresent", and "Never". The Environment variable field has a link "Add variable" and a note: "To enter multiple key-value pairs in a batch, you can paste multiple lines of key-value pairs (key=value or key:value) in the 'Variable name' field. They will be filled accordingly." The CPU/memory limit field has a table with columns for CPU limit and Memory limit. The CPU limit table has rows for "request" (0.25) and "limit" (0.5) with a unit of "-core". The Memory limit table has rows for "request" (256) and "limit" (1024) with a unit of "MiB". The GPU resource field has a "Number of cards" field set to "0" with a note: "Configure the minimum GPU resource usage of this workload. Please make sure that the cluster has enough GPU resource." The Target port field has a link "Add container port". There is also a link "Advanced settings" at the bottom.

主要参数信息如下：

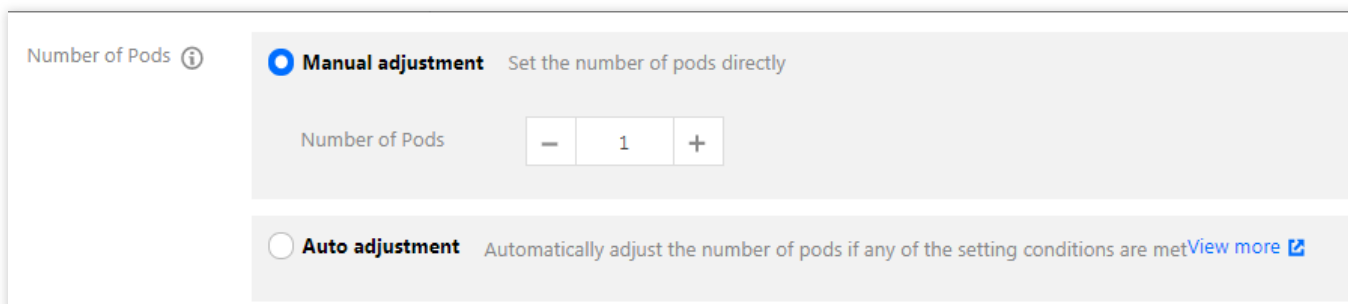
名称：输入实例内容容器名称，本文以 `test` 为例。

镜像：单击**选择镜像**，在弹框中选择**Docker Hub镜像** > **nginx**，并单击**确定**。

镜像版本（Tag）：使用默认值 `latest`。

镜像拉取策略：提供 `Always`、`IfNotPresent`、`Never` 三种策略，请按需选择，本文以**不进行设置使用默认策略**为例。

6. 在“实例数量”中，根据以下信息设置服务的实例数量。本文以**手动调节**为例，实例数量设置为1。如下图所示：



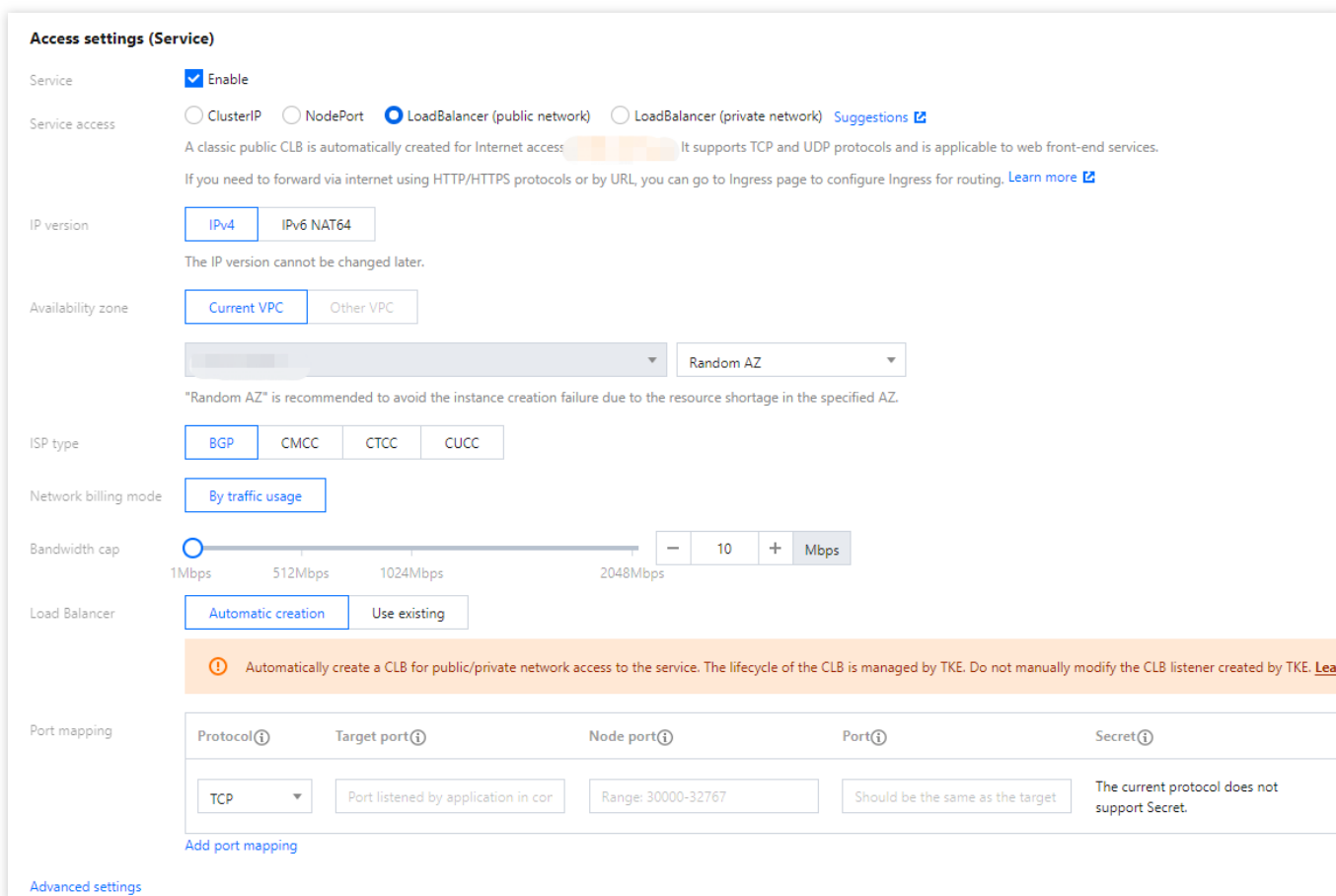
Number of Pods ⓘ

☒ **Manual adjustment** Set the number of pods directly

Number of Pods 1

☐ **Auto adjustment** Automatically adjust the number of pods if any of the setting conditions are met [View more](#) ⓘ

7. 根据以下提示，进行工作负载的访问设置。如下图所示：



Access settings (Service)

Service ☒ Enable

Service access ☐ ClusterIP ☐ NodePort ☒ LoadBalancer (public network) ☐ LoadBalancer (private network) [Suggestions](#) ⓘ

A classic public CLB is automatically created for Internet access. It supports TCP and UDP protocols and is applicable to web front-end services.

If you need to forward via internet using HTTP/HTTPS protocols or by URL, you can go to Ingress page to configure Ingress for routing. [Learn more](#) ⓘ

IP version

The IP version cannot be changed later.

Availability zone

"Random AZ" is recommended to avoid the instance creation failure due to the resource shortage in the specified AZ.

ISP type

Network billing mode

Bandwidth cap 10 Mbps

Load Balancer

ⓘ Automatically create a CLB for public/private network access to the service. The lifecycle of the CLB is managed by TKE. Do not manually modify the CLB listener created by TKE. [Learn more](#) ⓘ

Port mapping

Protocol ⓘ	Target port ⓘ	Node port ⓘ	Port ⓘ	Secret ⓘ
TCP	Port listened by application in cor	Range: 30000-32767	Should be the same as the target	The current protocol does not support Secret.

[Add port mapping](#)

[Advanced settings](#)

Service：勾选“启用”。

服务访问方式：选择“公网LB访问”。

负载均衡器：根据实际需求进行选择。

端口映射：选择 `TCP` 协议，将容器端口和服务端口都设置为 `80`。

协议：选择需要通信的协议类型。

容器端口：容器内应用程序监听的端口，端口范围 `1~65535`。

主机端口：可通过云服务器 IP + 主机端口访问服务，端口范围30000~32767，不填自动分配。

服务端口：集群外通过负载均衡域名或 IP + 服务端口访问服务；集群内通过服务名 + 服务端口访问服务。

Secret：仅当选择 TCP SSL 协议时才需要选择。

注意

服务所在集群的安全组需要放通节点网络及容器网络，同时需要放通30000 - 32768端口，否则可能会出现容器服务无法使用问题，详情请参见 [容器服务安全组设置](#)。

8. 单击**创建Workload**，完成 Nginx 服务的创建。

访问 Nginx 服务

可通过以下两种方式访问 Nginx 服务。

通过负载均衡 IP 访问 Nginx 服务

1. 单击左侧导航栏中 [集群](#)，进入“集群管理”页面。
2. 单击 Nginx 服务所在的集群 ID，选择**服务与路由** > **Service**。
3. 在 Service 列表页面，复制 Nginx 服务的负载均衡 IP，如下图所示：

Basic information

Node management

Namespace

Workload

HPA

Service and route

• Service

Create

default

You can enter only one keyword to search by name

Name	Labels	Type	Selector	IP address	Time created	Operation
kubernetes	componentapiserver providerkubernetes	ClusterIP	-	IP (IPv4) (Service IP)	2023-03-06 11:53:32	Update configuration Delete

Page 1

20

4. 在浏览器地址栏输入负载均衡 IP，按“**Enter**”即可访问服务。

通过服务名称访问 Nginx 服务

集群内的其他服务或容器可以直接通过服务名称访问。

验证 Nginx 服务

服务创建成功，访问服务时直接进入 Nginx 服务器的默认欢迎页。如下图所示：

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.

更多 Nginx 设置

若容器创建失败，可查看 [事件常见问题](#)。

手动搭建 Hello World 服务

最近更新时间：2023-05-06 19:41:07

操作场景

本文档旨在帮助您了解如何快速创建一个容器集群内的 Hello World 的 Node.js 版的服务。更多关于如何构建 Docker 镜像的教程可参阅 [如何构建 docker 镜像](#)。

前提条件

已创建集群，详情请参见 [创建集群](#)。

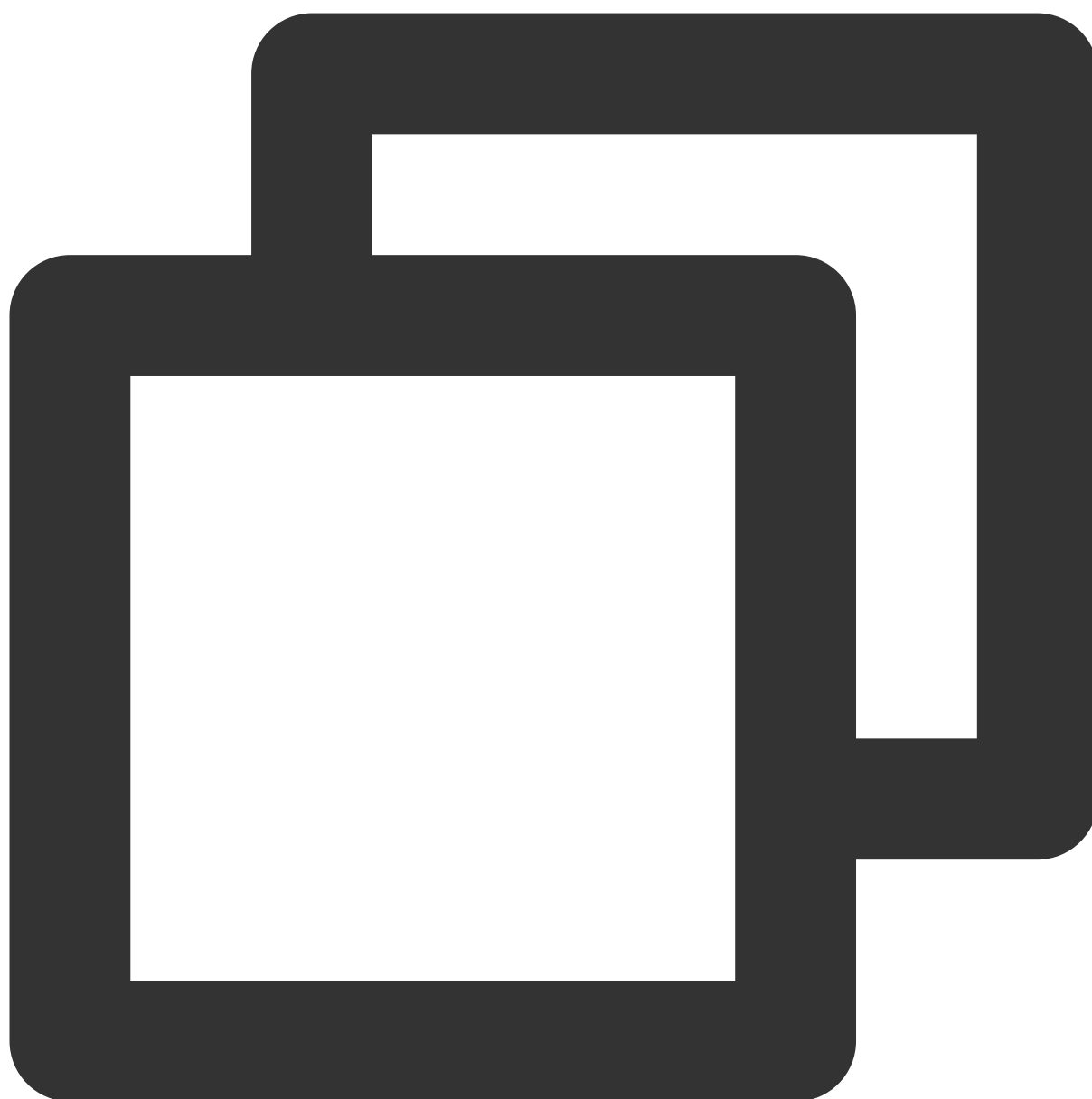
已登录节点，且该节点已安装 Node.js。

操作步骤

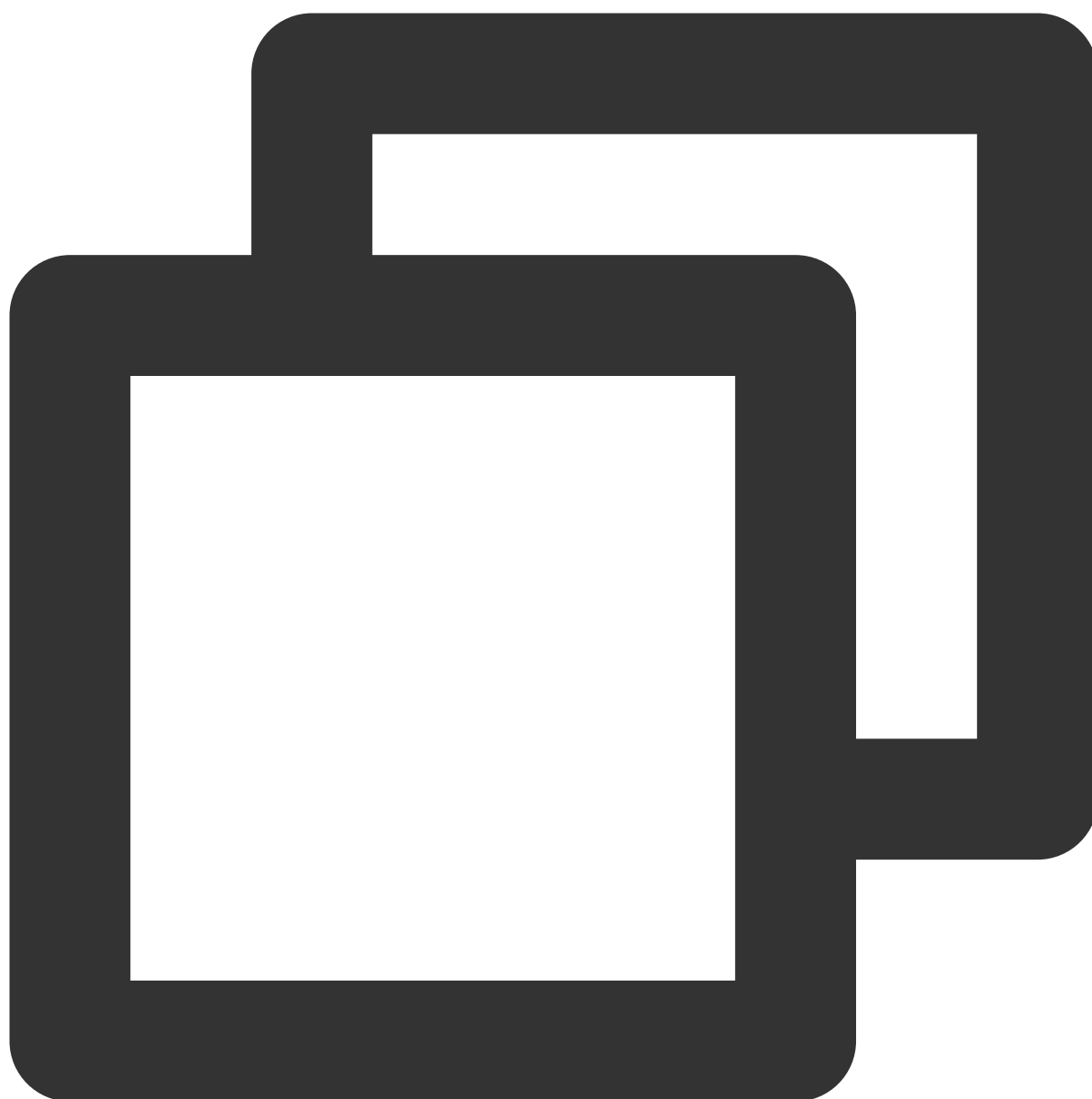
编写代码制作镜像

编写应用程序

1. 依次执行以下命令，创建并进入 hellonode 的文件夹。

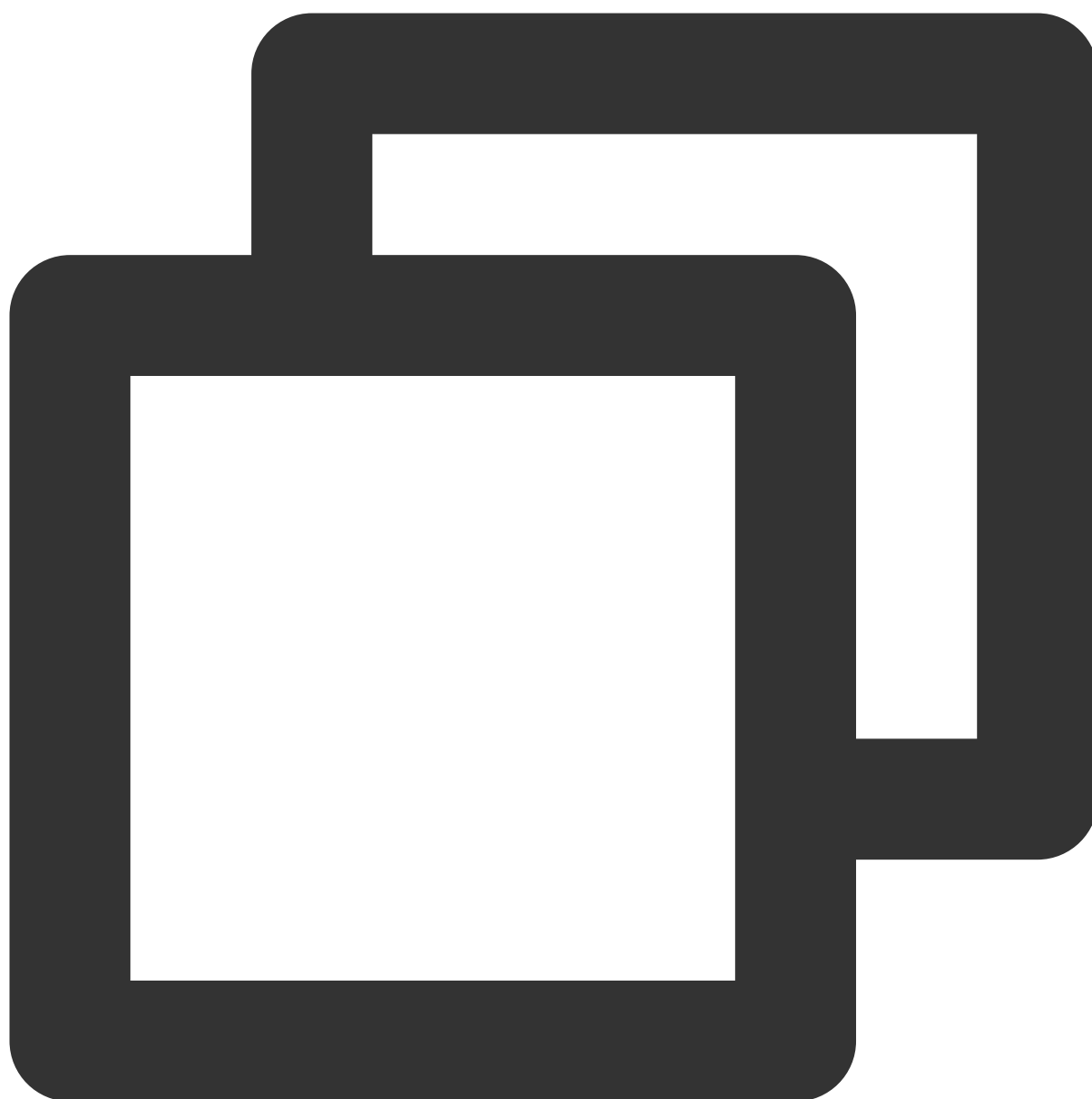


```
mkdir hellonode
```



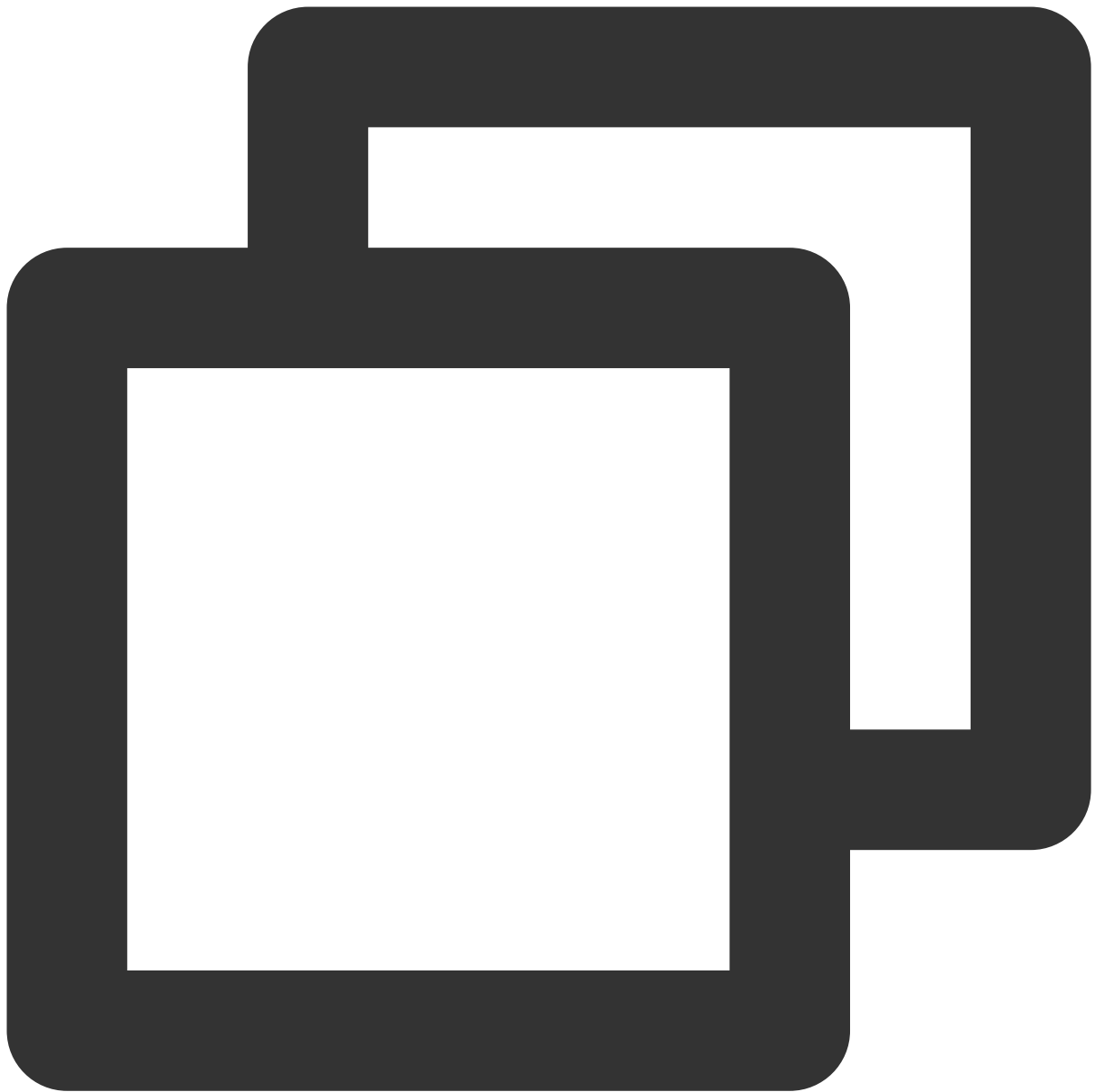
```
cd hellonode/
```

2. 执行以下命令，新建并打开 **server.js** 文件。



```
vim server.js
```

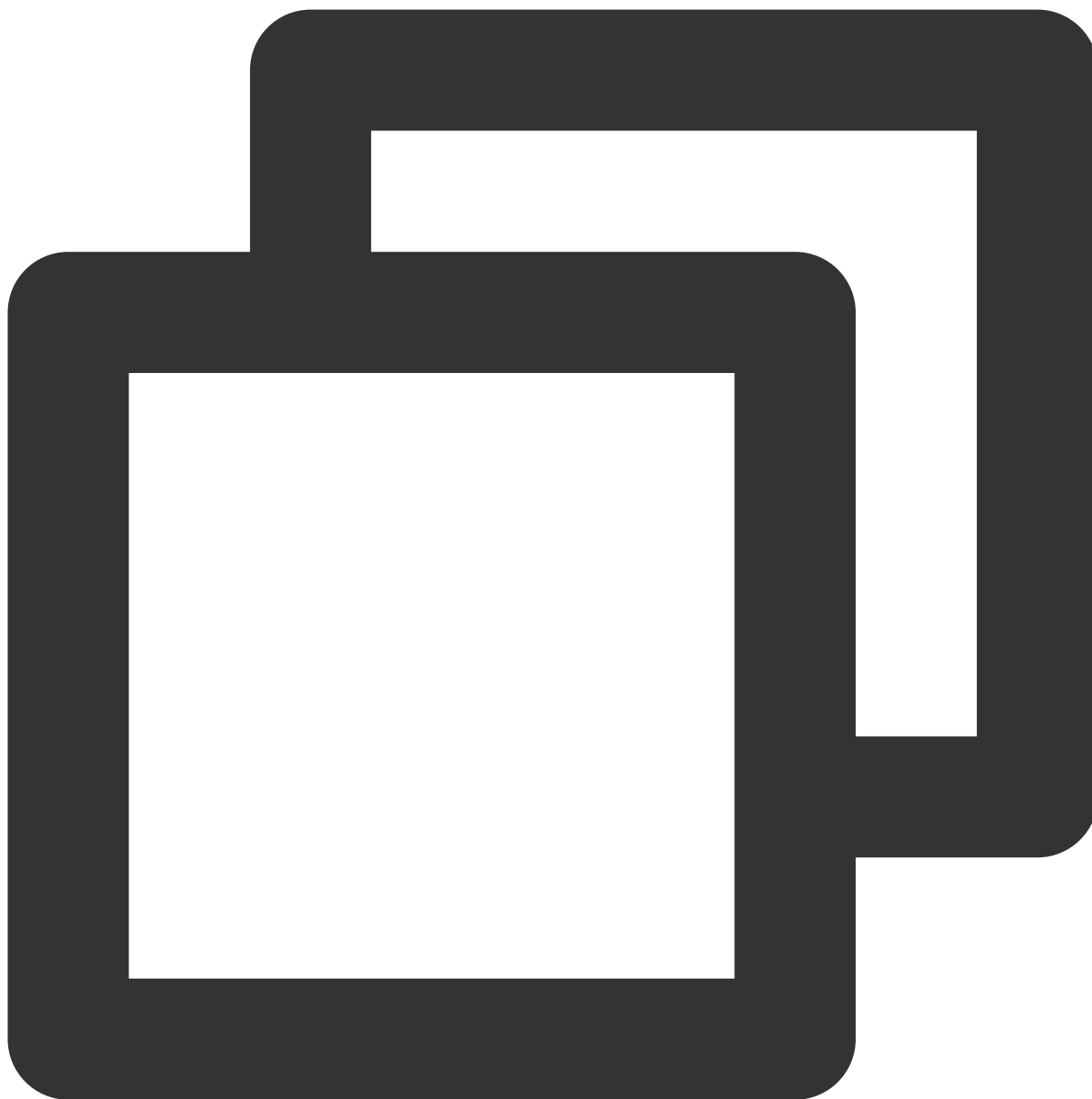
3. 按 **i** 切换至编辑模式，将以下内容输入 `server.js`。



```
var http = require('http');
var handleRequest = function(request, response) {
  console.log('Received request for URL: ' + request.url);
  response.writeHead(200);
  response.end('Hello World!');
};
var www = http.createServer(handleRequest);
www.listen(80);
```

按“Esc”，输入 `:wq`，保存文件并返回。

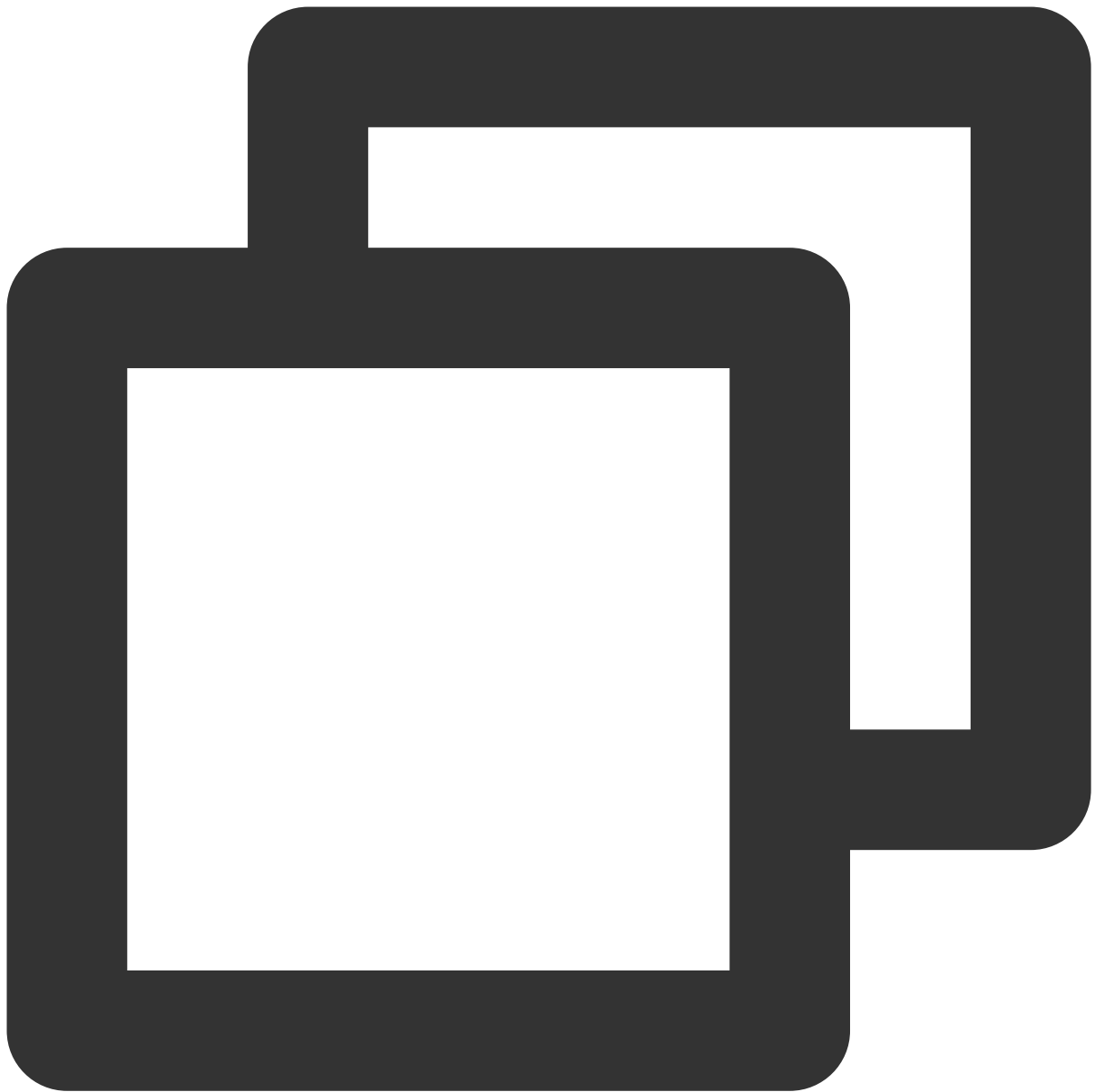
4. 执行以下命令，执行 `server.js` 文件。



```
node server.js
```

5. 测试 Hello World 程序。

方式1：再次登录节点，执行以下命令。



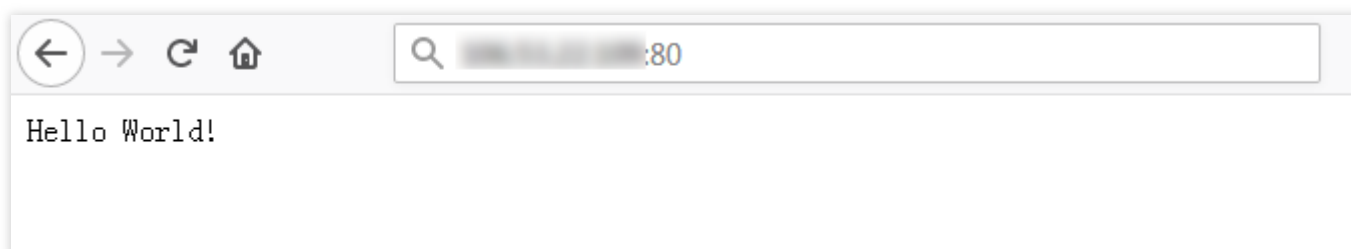
```
curl 127.0.0.1:80
```

显示如下，则说明 Hello World 程序运行成功。

```
[root@VM_2_5_centos ~]# curl 127.0.0.1:80
Hello World! [root@VM_2_5_centos ~]#
```

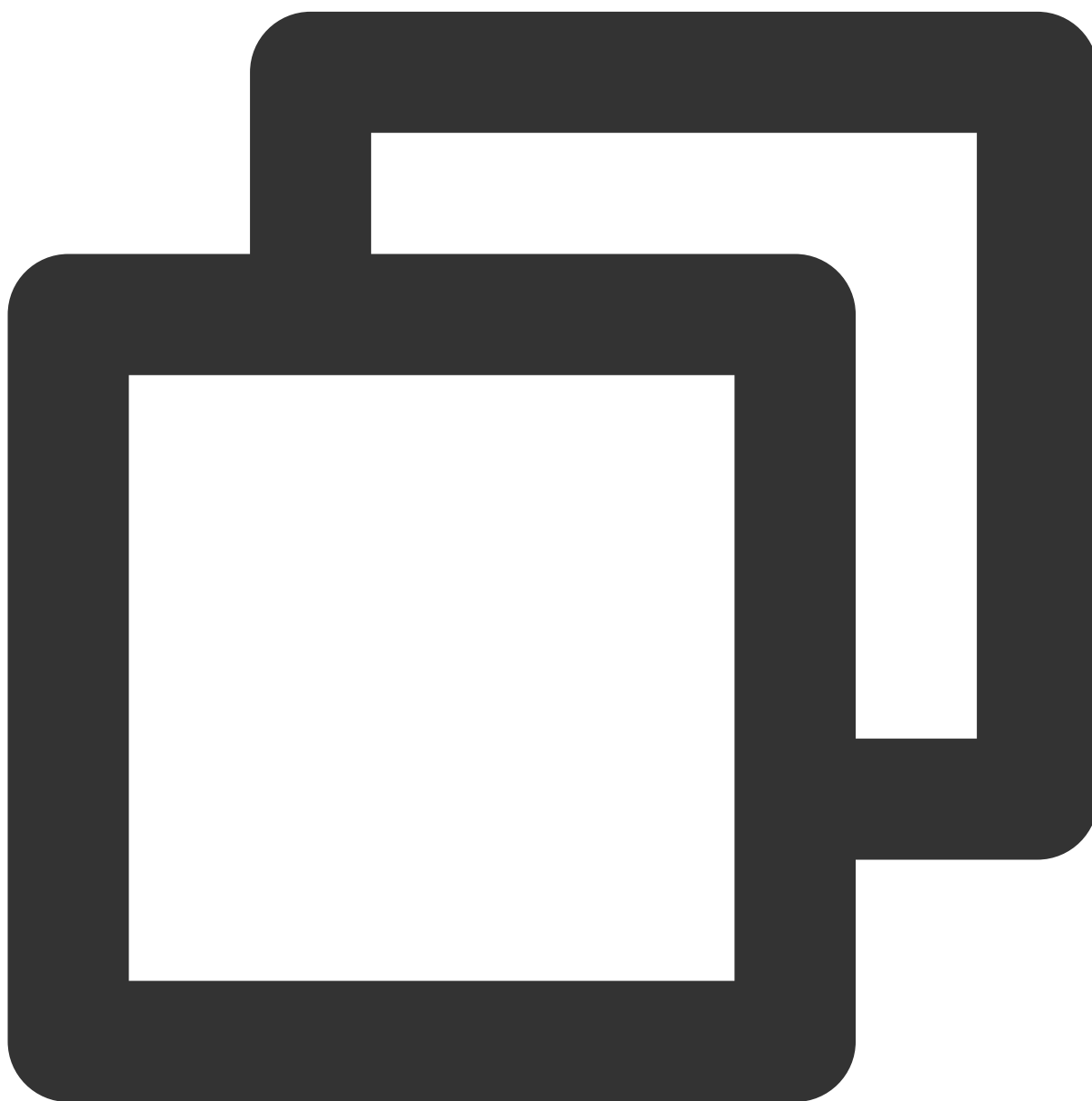
方式2：打开本地浏览器，以 云服务器实例的公网IP地址：已配置的端口号 的形式访问，端口号为80。

显示如下，则说明 Hello World 程序运行成功。

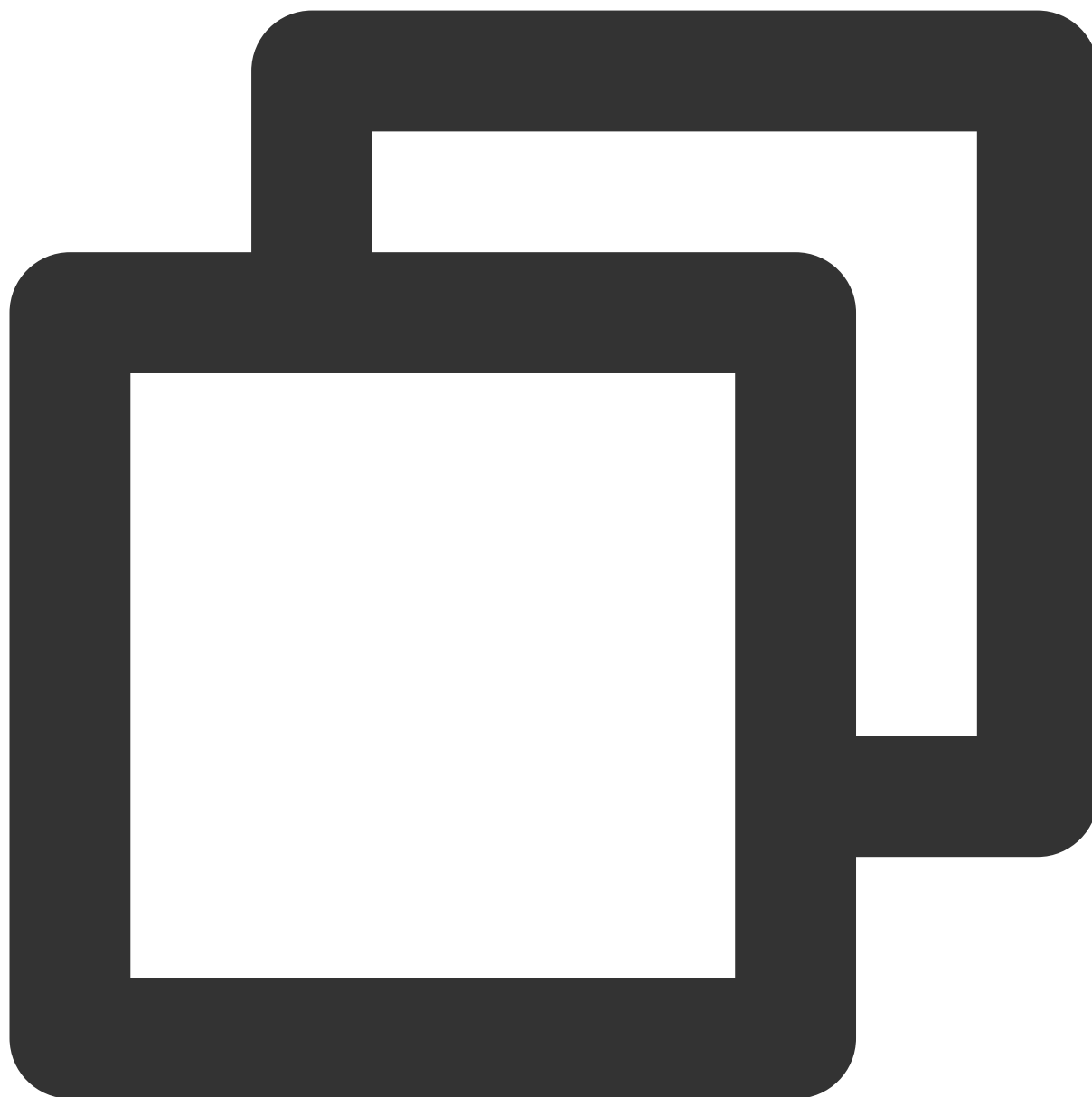


创建 Docker 镜像

1. 依次执行以下命令，在 hellonode 文件夹下，创建 Dockerfile 文件。

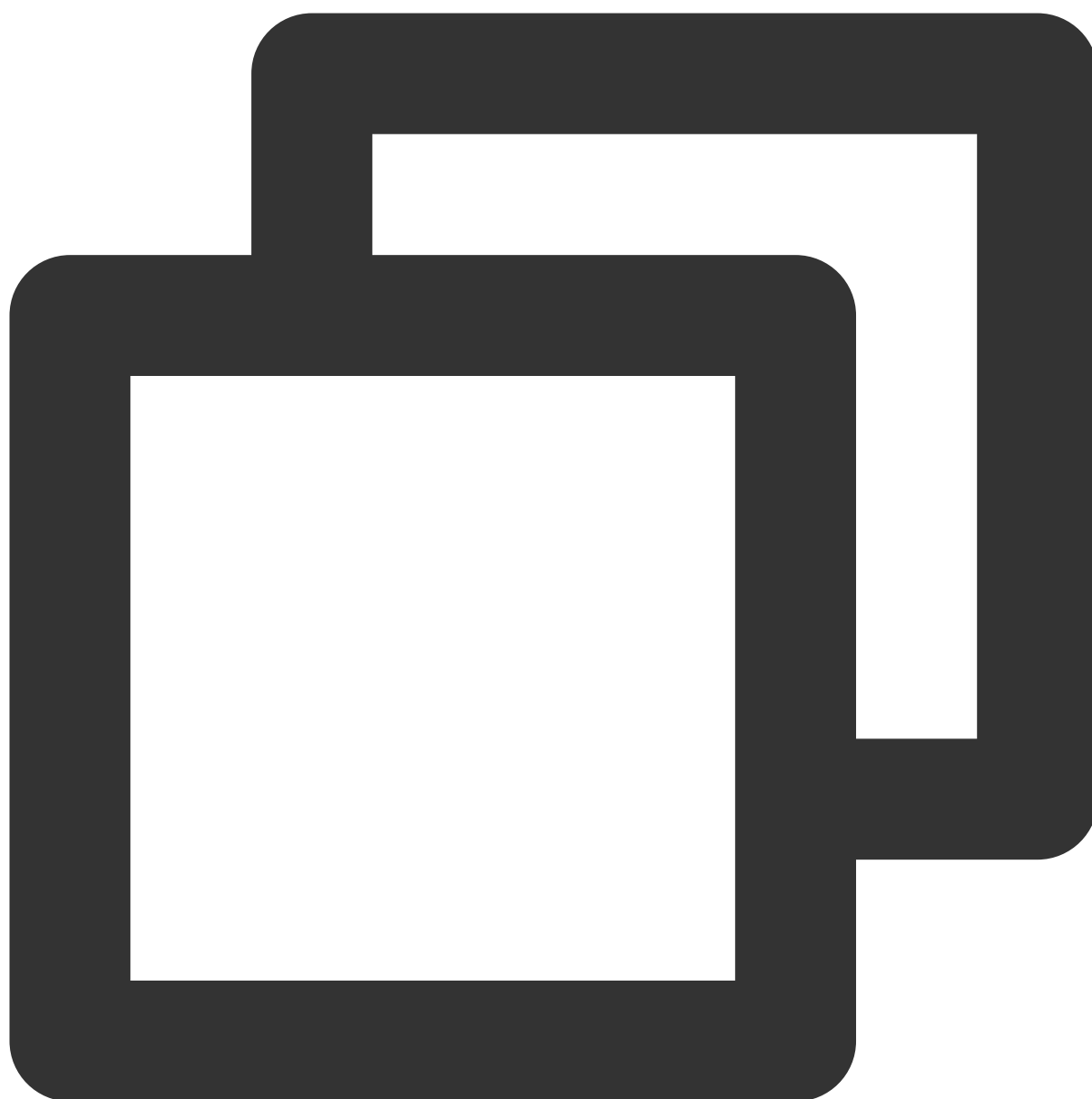


```
cd hellonode
```



```
vim Dockerfile
```

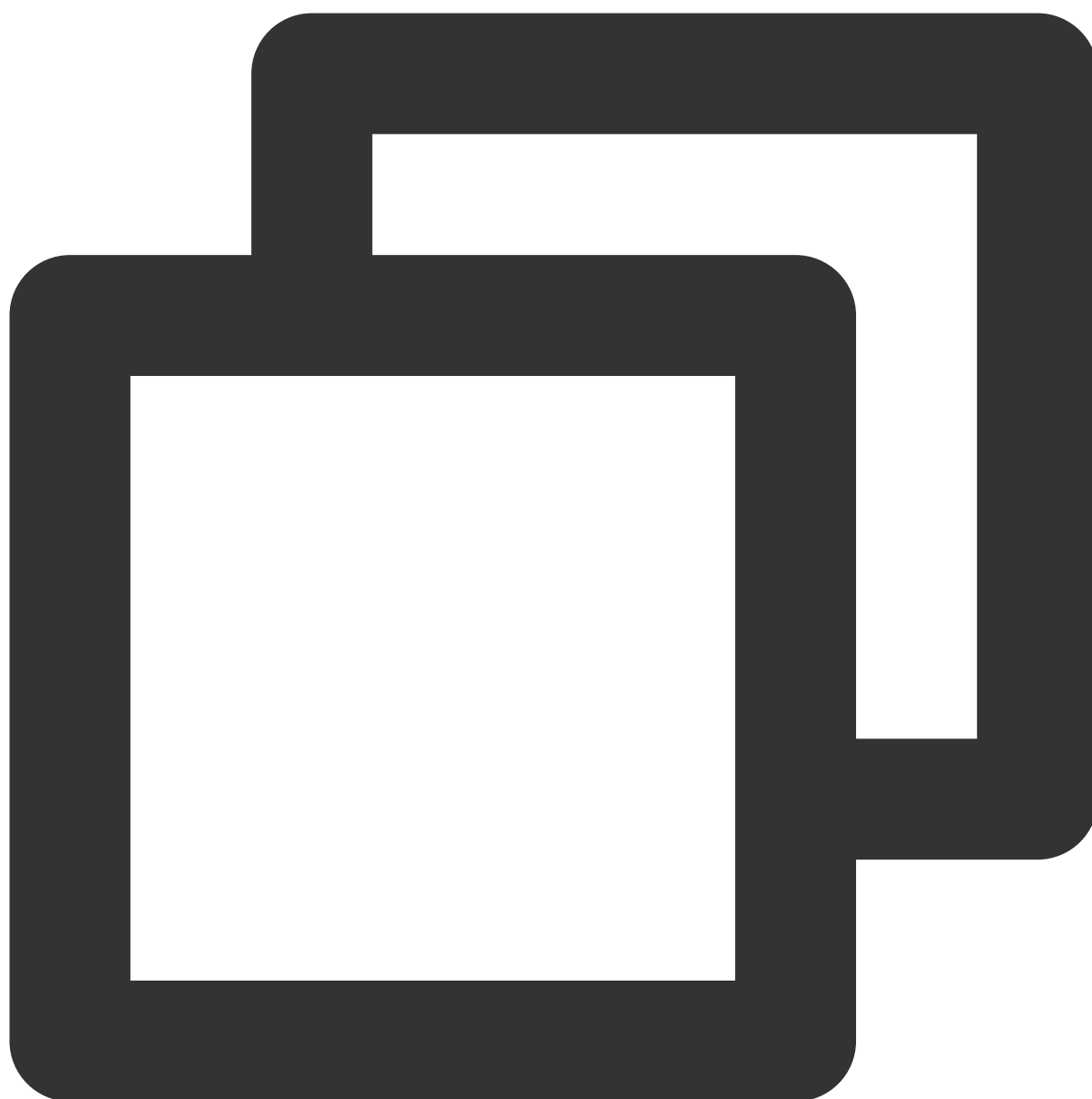
2. 按 **i** 切换至编辑模式，将以下内容输入 **Dockerfile** 文件。



```
FROM node:4.4
EXPOSE 80
COPY server.js .
CMD node server.js
```

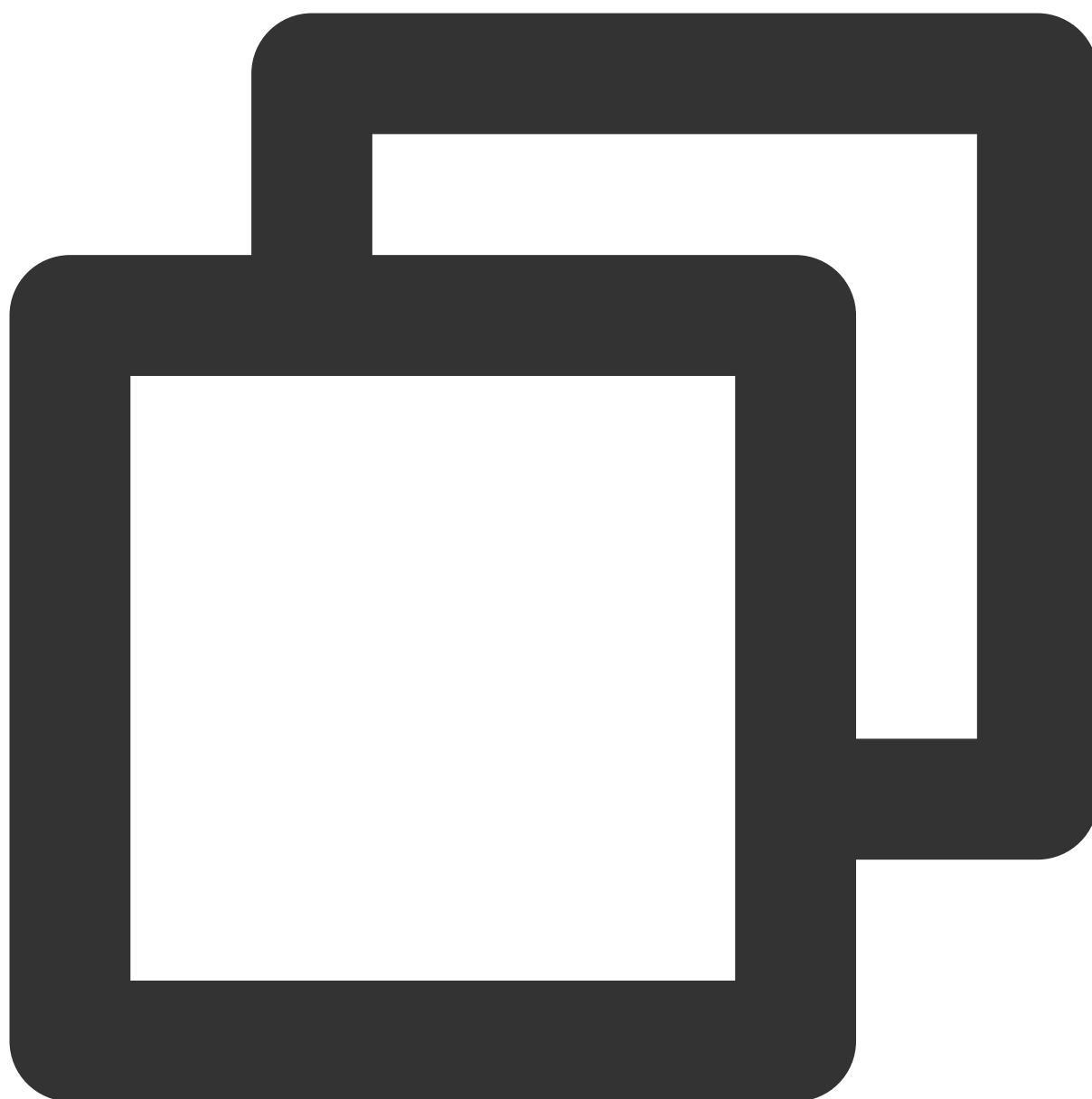
按“Esc”，输入 `:wq`，保存文件并返回。

3. 该节点安装 **docker**，并启动 **docker** 服务。



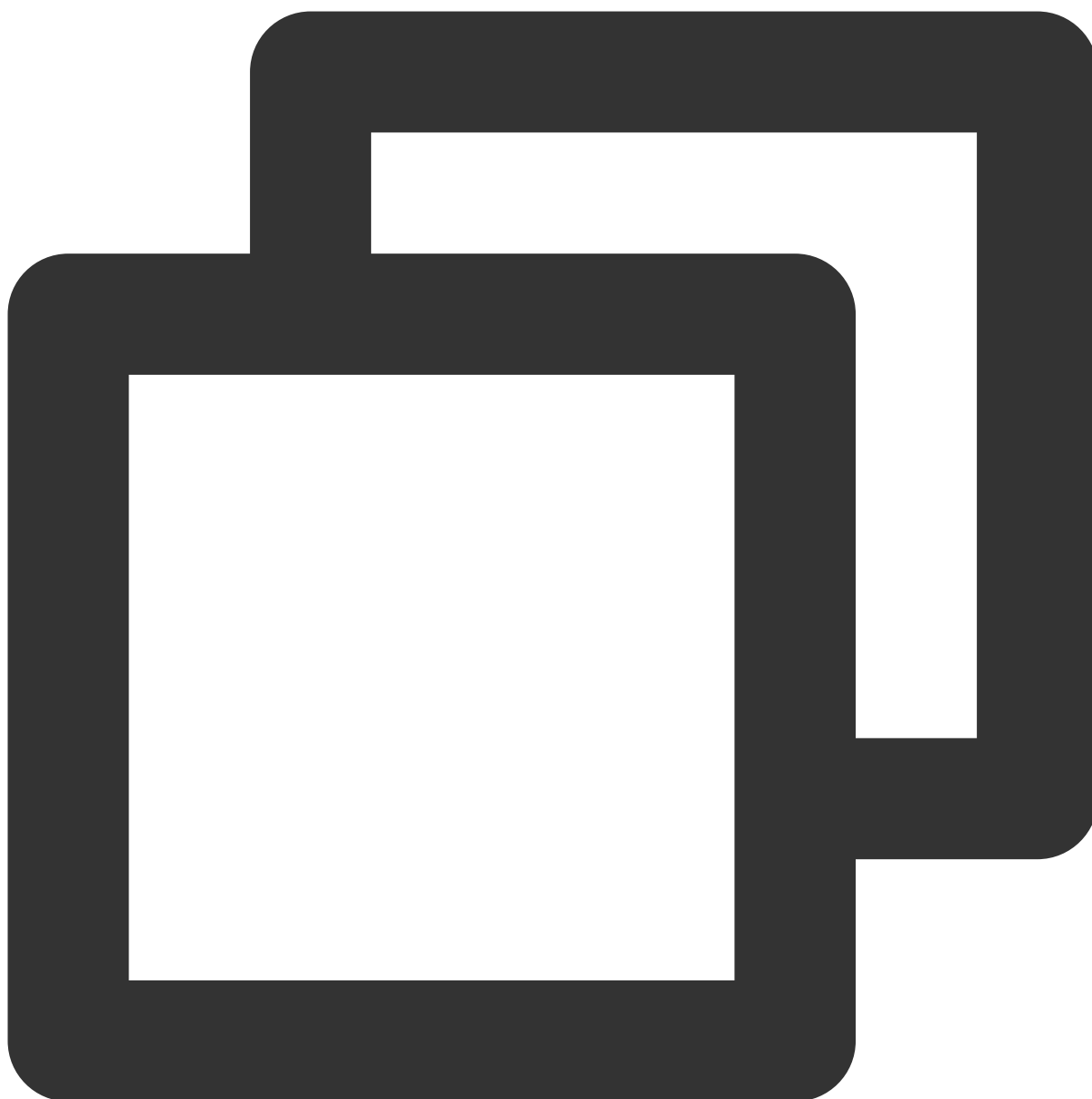
```
yum install -y docker  
systemctl start docker
```

4. 执行以下命令，构建镜像。



```
docker build -t hello-node:v1 .
```

5. 执行以下命令，查看构建好的 `hello-node` 镜像。



```
docker images
```

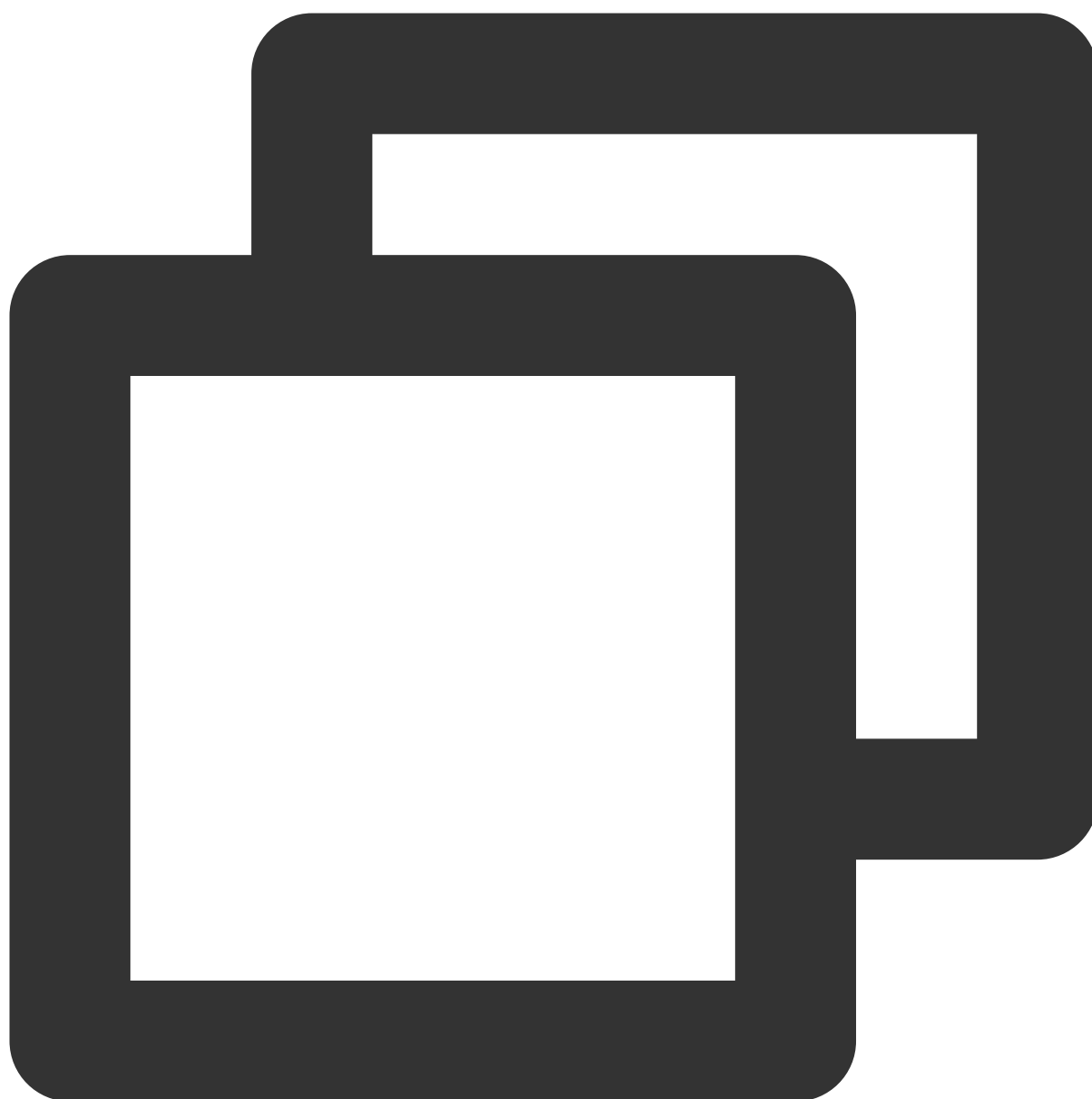
显示结果如下，则说明 `hello-node` 镜像已成功构建，记录其 `IMAGE ID`。如下图所示：

```
[root@VM_2_5_centos hellonode]# docker images
```

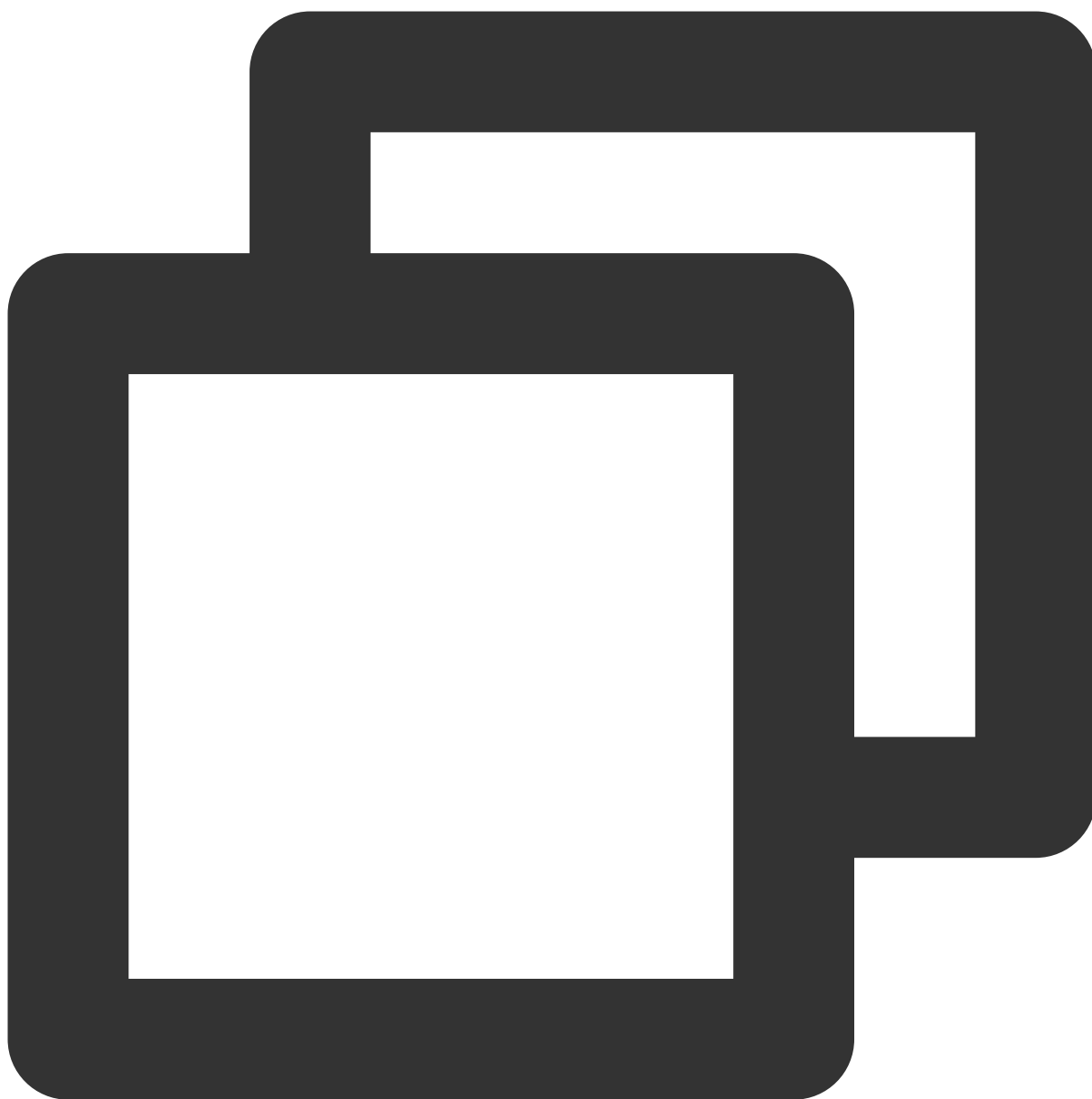
REPOSITORY	TAG	IMAGE ID	CR
hello-node	v1	5ac	2

上传该镜像到 `qcloud` 镜像仓库

依次执行以下命令，上传镜像到 `qcloud` 镜像仓库。



```
docker tag IMAGEID ccr.ccs.tencentyun.com/命名空间/hello-node:v1
```



```
docker login ccr.ccs.tencentyun.com
docker push ccr.ccs.tencentyun.com/命名空间/hello-node:v1
```

说明

请将命令中的 **IMAGEID** 替换为 [查看镜像](#) 中记录的 **IMAGEID**。

请将命令中的命名空间替换为您已创建的命名空间。若您还没有命名空间，请先创建命名空间，详情请参见 [创建命名空间](#)。

显示以下结果，则说明镜像上传成功。

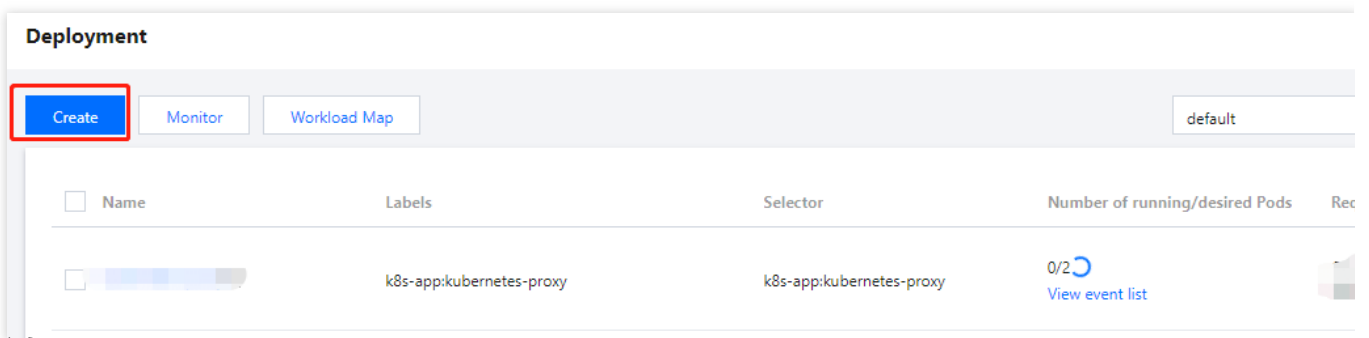
```
[root@VM_2_5_centos hellonode]# sudo docker tag ccr.ccs.tencentyun.com/
[root@VM_2_5_centos hellonode]# sudo docker push ccr.ccs.tencentyun.com/: test/hello
The push refers to repository [ccr.ccs.tencentyun.com/ test/helloworld]
7357e3a21b1f: Pushed
20a6f9d228c0: Pushed
80c332ac5101: Pushed
04dc8c446a38: Pushed
1050aff7cfff: Pushed
66d8e5ee400c: Pushed
2f71b45e4e25: Pushed
v1: digest: sha256:9c139ecbb29c49f25e02d7906b9e78c6e2e274827a75603ef48fa5547ff8a620 s:
```

通过该镜像创建 Hello World 服务

注意

在创建使用 Hello World 服务之前，您必须拥有集群。若您还没有集群，请先创建集群，详情请参见 [创建集群](#)。

1. 登录 [容器服务控制台](#)，选择左侧导航栏中的**集群**。
2. 在**集群管理**页面，选择需创建服务的集群 ID，进入**集群详情页**。
3. 选择**工作负载 > Deployment**，在 Deployment 页面单击**新建**。如下图所示：



4. 在**新建 Deployment** 页面，根据以下信息，设置工作负载基本信息。如下图所示：

The screenshot shows the configuration interface for a new container. The fields are as follows:

- Name:** helloworld. Below it, a note states: "Up to 63 characters, including lowercase letters, numbers, and hyphens ("-"). It must begin with a lowercase letter, and end with a number or hyphen."
- Description:** helloworld serve
- Namespace:** default (selected from a dropdown)
- Labels:** k8s-app = helloworld. Below this, there is an "Add" link and a note: "The key name cannot exceed 63 chars. It supports letters, numbers, "/" and "-". "/" cannot be placed at the beginning. A prefix is supported. The label key value can only include letters, numbers and separators ("-", "_", "."). It must start and end with letters and numbers."
- OS type:** Linux (selected from a dropdown). Below it, a note states: "Configurations are initialized when you change the OS type for the container."
- Volume (optional):** Add volume. Below it, a note states: "It provides storage for the container. It can be a node path, cloud disk volume, file storage NFS, config file and PVC, and must be mounted."

工作负载名：输入要创建的工作负载的名称，本文以 **helloworld** 为例。

描述：填写工作负载的相关信息。

命名空间：根据实际需求进行选择。

标签：key = value 键值对，本例中标签默认值为 **k8s-app = helloworld**。

OS 类型：根据实际需求进行选择，本文以 **Linux** 为例。

数据卷：根据实际需求设置工作负载的挂载卷，详情请参见 [Volume 管理](#)。

5. 参考以下信息设置“实例容器”。

5.1 输入实例容器名称，本文以 **helloworld** 为例。

5.2 单击**选择镜像**，在弹出框中选择**我的镜像**，通过搜索框功能查找出 **helloworld** 镜像，并单击**确定**。

主要参数信息如下：

镜像版本（Tag）：使用默认值 **latest**。

镜像拉取策略：提供 **Always**、**IfNotPresent** 和 **Never** 3种策略，请按需选择。本文以不进行设置使用默认策略为例。

6. 在“实例数量”中，根据以下信息设置服务的实例数量。如下图所示：

The screenshot shows the "Number of instances" configuration section. It has two radio buttons: "Manual adjustment" (selected) and "Auto adjustment". Below the radio buttons is the text "Set the number of Pods directly". At the bottom, there is a "Number of instances" input field with a value of "1" and minus/plus buttons.

手动调节：设定实例数量，本文实例数量设置为1。可单击“+”或“-”控制实例数量。

自动调节：满足任一设定条件，则自动调节实例（pod）数目。详情请参见 [服务自动扩缩容](#)。

7. 根据以下提示，进行工作负载的访问设置。

Service：勾选“启用”。

服务访问方式：选择“公网LB访问”。

负载均衡器：根据实际需求进行选择。

端口映射：选择 TCP 协议，将容器端口和服务端口都设置为80。

注意

服务所在集群的安全组需要放通节点网络及容器网络，同时需要放通30000 - 32768端口，否则可能会出现容器服务无法使用问题，详情请参见 [容器服务安全组设置](#)。

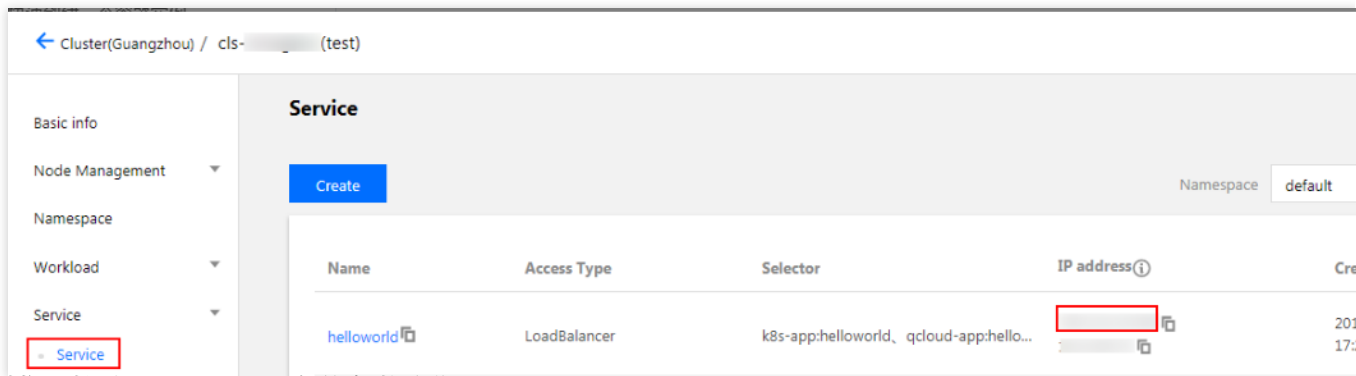
8. 单击**创建 Deployment**，完成 Hello World 服务的创建。

访问 Hello World 服务

可通过以下两种方式访问 Hello World 服务。

通过负载均衡 IP 访问 Hello World 服务

1. 登录[容器服务控制台](#)，选择左侧导航栏中的**集群**。
2. 在**集群管理**页面，单击 Hello World 服务所在的集群 ID，进入**集群详情页**。
3. 选择**服务与路由 > Service**，在 **Service** 页面
4. 在服务管理页面，复制 helloworld 服务的负载均衡 IP，如下图所示：



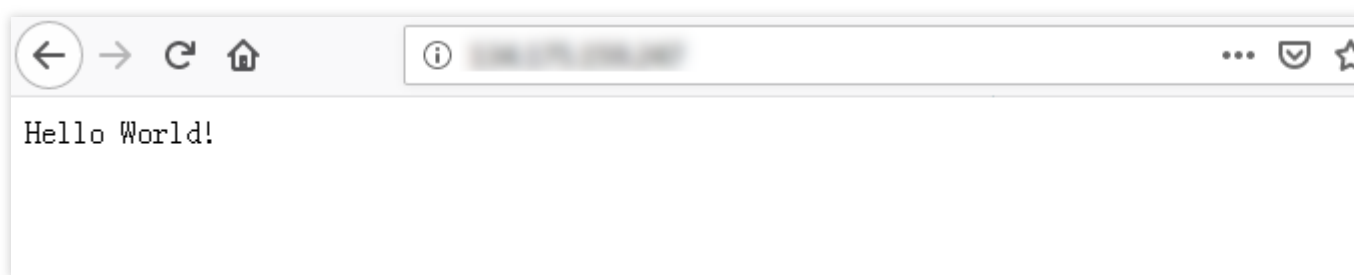
5. 在浏览器粘贴 helloworld 服务的负载均衡 IP。

通过服务名称访问 Hello World 服务

集群内的其他服务或容器可以直接通过**服务名称**访问。

验证 Hello World 服务

访问服务时显示如下，则 Hello World 服务创建成功。



若容器创建失败，可查看 [事件常见问题](#)。

单实例版 WordPress

最近更新時間：2023-05-23 14:50:16

操作场景

WordPress 是使用 PHP 语言开发的博客平台。用户可以在支持 PHP 和 MySQL 数据库的服务上架设属于自己的网站，也可以把 WordPress 当作一个内容管理系统来使用。

本文档旨在介绍如何使用 Docker Hub 官方镜像 `wordpress` 来创建一个公开访问的 WordPress 网站。

前提条件

注意：

- `wordpress` 该镜像中包含了 WordPress 所有的运行环境，直接拉取创建服务即可。
- 创建单实例版的 WordPress 仅供测试使用，不能保证数据的持久化存储。建议您使用自建的 MySQL 或使用腾讯云数据库 TencentDB 来保存您的数据，详情请见 [使用 TencentDB 的 WordPress](#)。

- 已 [注册腾讯云账户](#)。
- 已创建 TKE 标准集群。操作详情请参见 [创建集群](#)。

操作步骤

创建 WordPress 服务

- 登录容器服务控制台，选择左侧导航栏中的 [集群](#)。
- 在“集群管理”页面，选择需创建服务的集群 ID，进入集群的基本信息页面。
- 在 **工作负载 > Deployment** 页面，单击 **新建**。参数详情见 [创建 Deployment](#)。

4. 在“新建Deployment”页面，根据以下信息，设置工作负载基本信息。如下图所示：

The screenshot shows the 'New Deployment' configuration page. It includes the following fields and instructions:

- Name:** A text input field with a placeholder 'Please enter a name'. Below it, a note states: 'Up to 63 characters, including lowercase letters, numbers, and hyphens ("-"). It must begin with a lowercase letter, and end with a number or lowercase letter.'
- Description:** A large text area with a placeholder 'Up to 1000 characters'.
- Namespace:** A dropdown menu currently set to 'default'.
- Labels:** A section with an 'Add' button. Below it, a note states: 'The key name cannot exceed 63 chars. It supports letters, numbers, "/" and "-". "/" cannot be placed at the beginning. A prefix is supported. [Learn more](#)'. Another note states: 'The label key value can only include letters, numbers and separators ("-", "_", "."). It must start and end with letters and numbers.'
- Volume (optional):** A section with an 'Add volume' button. Below it, a note states: 'It provides storage for the container. It can be a node path, cloud disk volume, file storage NFS, config file and PVC, and must be mounted to the specified path of the container. [Instruction](#)'.

- **工作负载名：**输入要创建的工作负载的名称，本文以 **wordpress** 为例。
- **描述：**填写工作负载的相关信息。
- **标签：**本例中标签默认值为 `k8s-app = wordpress`。
- **命名空间：**根据实际需求进行选择。
- **数据卷：**根据实际需求设置工作负载的挂载卷，详情请参见 [Volume 管理](#)。

5. 参考以下信息设置“实例容器”。如下图所示：

The screenshot shows the 'Containers in the Pod' configuration page for a container named 'container-1'. It includes the following fields and instructions:

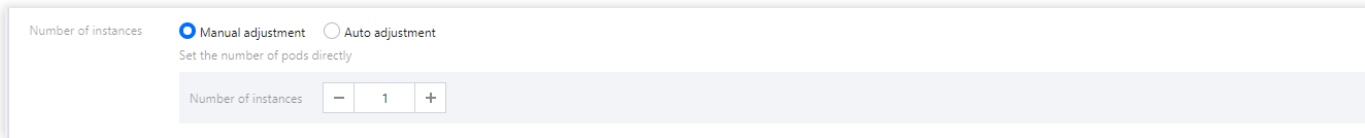
- Name:** A text input field with a placeholder 'Enter the container name.'. Below it, a note states: 'Up to 63 characters. It supports lower case letters, numbers, and hyphen ("-") and cannot start or end with "-".'
- Image:** A text input field with a placeholder and a 'Select image' button.
- Image tag:** A text input field with a placeholder '"latest" is used if it's left empty.'
- Pull image from remote registry:** Three radio buttons: 'Always', 'IfNotPresent', and 'Never'. Below them, a note states: 'If the image pull policy is not set, when the image tag is empty or "latest", the "Always" policy is used, otherwise "IfNotPresent" is used.'
- Environment variable:** A section with an 'Add variable' button. Below it, a note states: 'To enter multiple key-value pairs in a batch, you can paste multiply lines of key-value pairs (key=value or keyvalue) in the Variable Name field. They will be automatically filled accordingly.'
- CPU/memory limit:** Two sections. The 'CPU limit' section has 'request' (0.25) and 'limit' (0.5) fields, with a '-core' unit. The 'Memory limit' section has 'request' (256) and 'limit' (1024) fields, with a 'MiB' unit. Below these, a note states: 'Request is used to pre-allocate resources. When the nodes in the cluster do not have the required number of resources, the container will fail to create. Limit is used to set a upper limit for resource usage for a container, so as to avoid over usage of node resources in case of exceptions.'
- GPU resource:** Two sections. The 'Number of cards' section has a value of 0. The 'VRAM' section has a value of 0 and a 'GiB' unit. Below these, a note states: 'The number of cards can only be 0.1-1 or an integer. The value of vRAM must be an integer (number of cards and vRAM is 0 by default).'

主要参数信息如下：

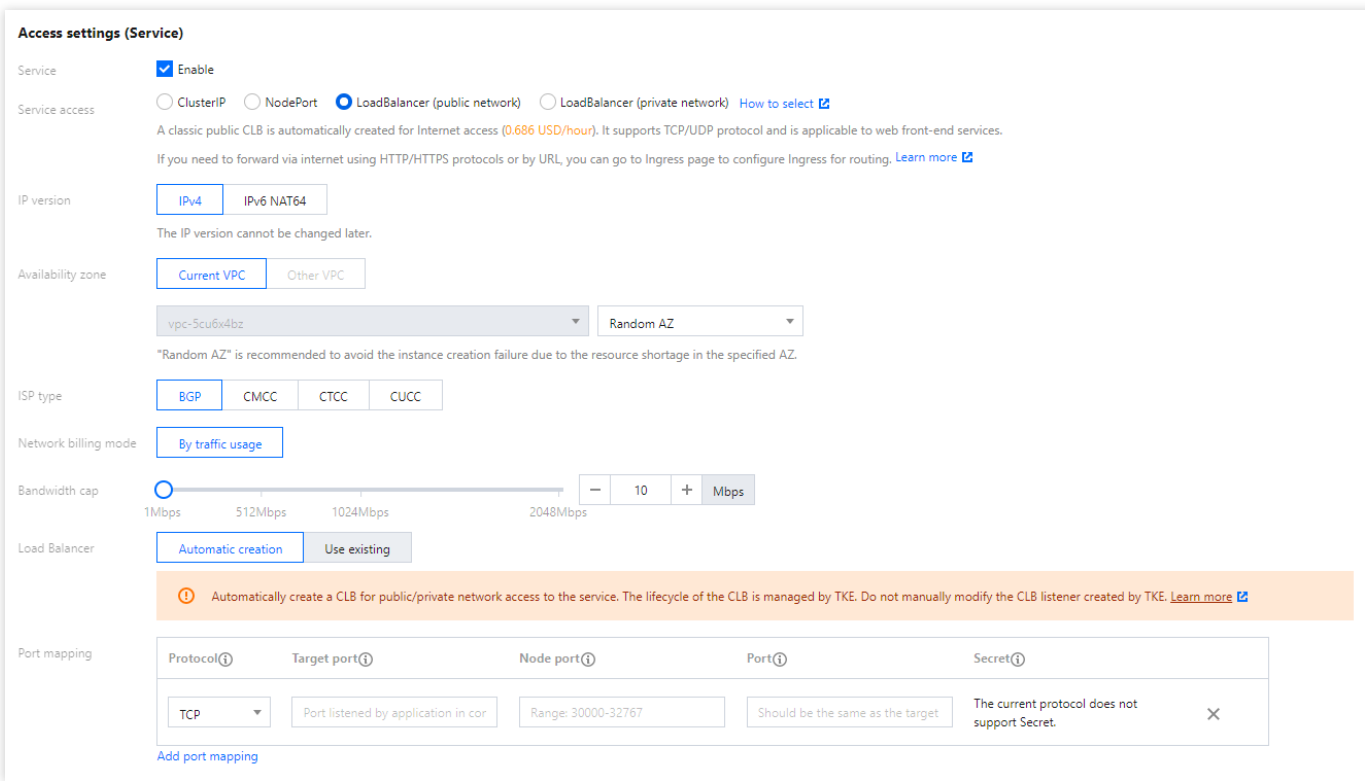
- **名称：**输入实例容器名称，本文以 **test** 为例。
- **镜像：**单击 **选择镜像**，在弹框中选择 **Docker Hub 镜像 > wordpress**，并单击 **确定**。

- **镜像版本（Tag）**：使用默认值 `latest`。
- **镜像拉取策略**：提供 Always、IfNotPresent、Never 三种策略，请按需选择，本文以**不进行设置使用默认策略**为例。

6. 在“实例数量”中，根据以下信息设置服务的实例数量。本文以**手动调节**为例，实例数量设置为1。如下图所示：



7. 根据以下提示，进行工作负载的访问设置。如下图所示：



- **Service**：勾选“启用”。
- **服务访问方式**：选择“公网LB访问”。
- **负载均衡器**：根据实际需求进行选择。
- **端口映射**：选择 TCP 协议，将容器端口和服务端口都设置为80。

注意：

服务所在集群的安全组需要放通节点网络及容器网络，同时需要放通30000 - 32768端口，否则可能会出现容器服务无法使用问题，详情请参见 [容器服务安全组设置](#)。

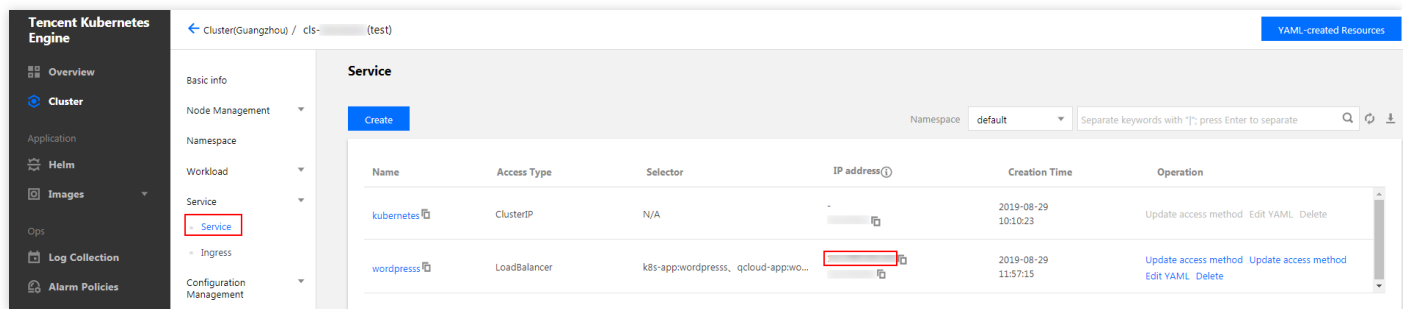
8. 单击**创建Deployment**，完成 wordpress 服务的创建。

访问 WordPress 服务

可通过以下两种方式访问 WordPress 服务。

通过负载均衡 IP 来访问 WordPress 服务

1. 单击左侧导航栏中 **集群**，进入“集群管理”页面。
2. 单击 WordPress 服务所在的集群 ID，选择**服务与路由** > **Service**。
3. 在 Service 列表页面，复制 WordPress 服务的负载均衡 IP，如下图所示：



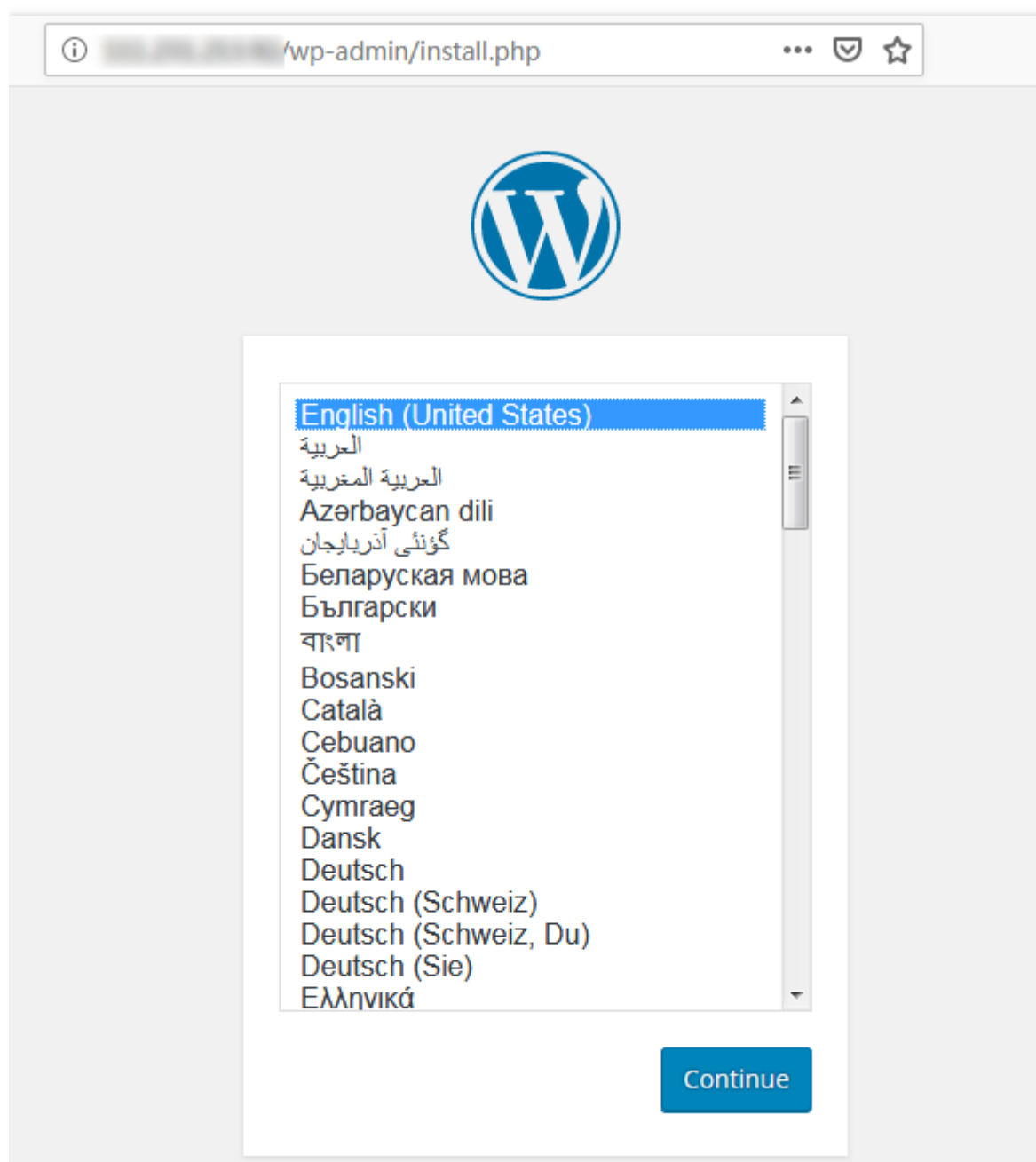
4. 在浏览器地址栏输入负载均衡 IP，按“**Enter**”即可访问服务。

通过服务名称访问服务

集群内的其他服务或容器可以直接通过服务名称访问。

验证 WordPress 服务

服务创建成功，访问服务时直接进入 WordPress 服务器的配置页。如下图所示：



更多 WordPress 设置

若容器创建失败，可查看 [事件常见问题](#)。

使用 TencentDB 的 WordPress

最近更新时间：2023-05-23 11:47:57

操作场景

[单实例版 WordPress](#) 示例中展示了如何快速创建 WordPress 服务。通过此方式创建的 WordPress 服务特点如下：

- 数据是写到同一个容器运行的 MySQL 数据库中。
- 服务可快速启动。
- 容器因某种原因停止，数据库和存储类的文件将会丢失。

而使用 MySQL 数据库可实现数据永久存储，数据库会在实例/容器重新启动后继续存在。本文档旨在介绍如何通过云数据库 TencentDB 设置 MySQL 数据库，以及如何创建使用 TencentDB 的 WordPress 服务。

前提条件

- 已 [注册腾讯云账户](#)。
- 已创建 TKE 标准集群。操作详情请参见 [创建集群](#)。

说明：

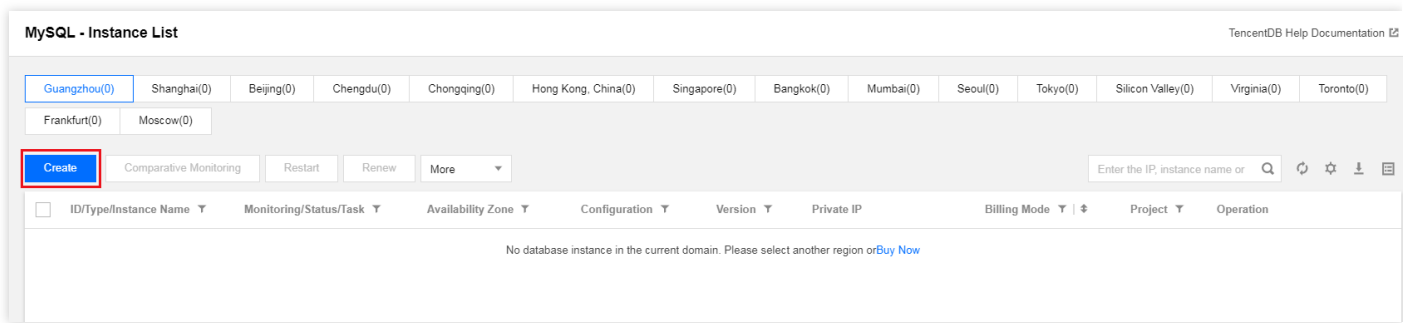
本文使用数据库为 [云数据库 MySQL](#)。

操作步骤

创建 WordPress 服务

创建云数据库 TencentDB

1. 登录 [云数据库 MySQL 控制台](#)，单击数据库实例列表上方的**新建**。如下图所示：



2. 选择购买配置，详情请见 [云数据库 MySQL](#)。

注意：

云数据库所在地域与集群相同，否则无法连接该数据库。

3. 数据库创建成功后，可在 [MySQL-实例列表](#) 中查看。
4. 对数据库进行初始化操作，详情请参见 [初始化 MySQL 数据库](#)。

创建使用 TencentDB 的 WordPress 服务

1. 登录容器服务控制台，选择左侧导航栏中的 [集群](#)。
2. 在“集群管理”页面，选择需创建服务的集群 ID，进入集群的基本信息页面。
3. 在 **工作负载 > Deployment** 页面，单击**新建**。参数详情见 [创建 Deployment](#)。
4. 在“新建Deployment”页面，根据以下信息，设置工作负载基本信息。如下图所示：

- **工作负载名**：输入要创建的工作负载的名称，本文以 `wordpress` 为例。
- **描述**：填写工作负载的相关信息。
- **标签**：本例中标签默认值为 `k8s-app = wordpress`。

- **命名空间**：根据实际需求进行选择。
- **数据卷**：根据实际需求设置工作负载的挂载卷，详情请参见 [Volume 管理](#)。

5. 参考以下信息设置“实例容器”。如下图所示：

The screenshot shows the 'Containers in the Pod' configuration page. At the top, there's a tab labeled 'container-1' and a '+ Add container' button. The main area contains several configuration sections:

- Name:** A text input field with a placeholder 'Enter the container name.' and a note: 'Up to 63 characters. It supports lower case letters, numbers, and hyphen ("-") and cannot start or end with "-".'.
- Image:** A text input field with a 'Select image' button.
- Image tag:** A text input field with a placeholder '"latest" is used if it's left empty.'
- Pull image from remote registry:** Three radio buttons: 'Always', 'IfNotPresent', and 'Never'. A note below says: 'If the image pull policy is not set, when the image tag is empty or "latest", the "Always" policy is used, otherwise "IfNotPresent" is used.'
- Environment variable:** A link 'Add variable' and a note: 'To enter multiple key-value pairs in a batch, you can paste multiply lines of key-value pairs (key=value or key:value) in the Variable Name field. They will be automatically filled accordingly.'
- CPU/memory limit:** Two sections. 'CPU limit' has 'request' (0.25) and 'limit' (0.5) fields, with a '-core' unit. 'Memory limit' has 'request' (256) and 'limit' (1024) fields, with a 'MiB' unit. A note below explains the difference between request and limit.
- GPU resource:** 'Number of cards' (0) and 'VRAM' (0) fields, with a 'GiB' unit. A note below says: 'The number of cards can only be 0.1-1 or an integer. The value of vRAM must be an integer (number of cards and vRAM is 0 by default).'

At the bottom left, there is a link 'Advanced settings'.

主要参数信息如下：

- **名称**：输入实例容器名称，本文以 **test** 为例。
- **镜像**：单击**选择镜像**，在弹框中选择**Docker Hub镜像 > wordpress**，并单击**确定**。
- **镜像版本（Tag）**：使用默认值 **latest**。
- **镜像拉取策略**：提供 **Always**、**IfNotPresent**、**Never** 三种策略，请按需选择，本文以**不进行设置使用默认策略**为例。
- **环境变量**：依次输入以下配置信息：
WORDPRESS_DB_HOST = 云数据库 MySQL 的内网 IP
WORDPRESS_DB_PASSWORD = 初始化时填写的密码

6. 在“实例数量”中，根据以下信息设置服务的实例数量。本文以**手动调节**为例，实例数量设置为1。如下图所示：

The screenshot shows the 'Number of instances' configuration section. It has two radio buttons: 'Manual adjustment' (which is selected) and 'Auto adjustment'. Below the radio buttons, there's a text label 'Set the number of pods directly'. Underneath, there's a 'Number of instances' field with a value of '1' and minus/plus buttons for adjustment.

7. 根据以下提示，进行工作负载的访问设置。如下图所示：

Access settings (Service)

Service ☒ Enable

Service access ☐ ClusterIP ☐ NodePort ☒ LoadBalancer (public network) ☐ LoadBalancer (private network) [How to select](#)

A classic public CLB is automatically created for Internet access (0.686 USD/hour). It supports TCP/UDP protocol and is applicable to web front-end services.

If you need to forward via internet using HTTP/HTTPS protocols or by URL, you can go to Ingress page to configure Ingress for routing. [Learn more](#)

IP version IPv4 IPv6 NAT64

The IP version cannot be changed later.

Availability zone Current VPC Other VPC

vpc-5cu6x4bz Random AZ

"Random AZ" is recommended to avoid the instance creation failure due to the resource shortage in the specified AZ.

ISP type BGP CMCC CTCC CUCC

Network billing mode By traffic usage

Bandwidth cap 1Mbps 512Mbps 1024Mbps 2048Mbps - 10 + Mbps

Load Balancer Automatic creation Use existing

Port mapping

Protocol	Target port	Node port	Port	Secret
TCP	Port listened by application in cor	Range: 30000-32767	Should be the same as the target	The current protocol does not support Secret.

[Add port mapping](#)

- **Service**：勾选“启用”。
- **服务访问方式**：选择“公网LB访问”。
- **负载均衡器**：根据实际需求进行选择。
- **端口映射**：选择 TCP 协议，将容器端口和服务端口都设置为80。

注意：

服务所在集群的安全组需要放通节点网络及容器网络，同时需要放通30000 - 32768端口，否则可能会出现容器服务无法使用问题，详情请参见 [容器服务安全组设置](#)。

8. 单击**创建Deployment**，完成 wordpress 服务的创建。

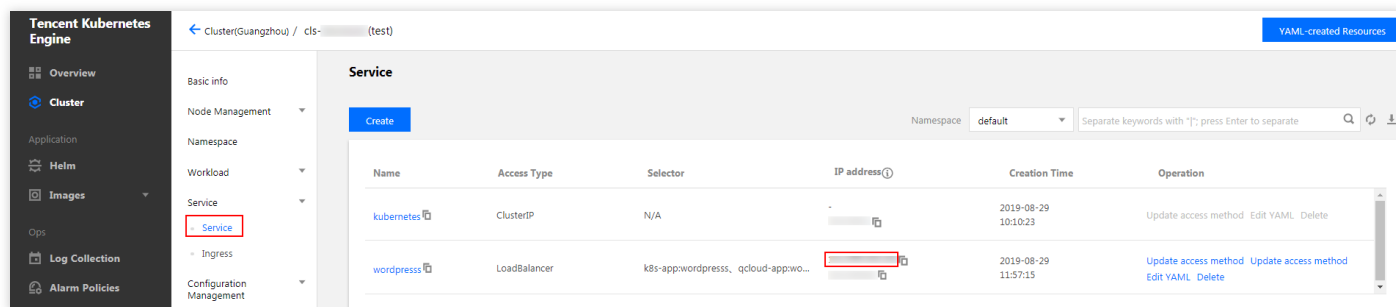
访问 WordPress 服务

可通过以下两种方式访问 WordPress 服务。

通过负载均衡 IP 访问 WordPress 服务

1. 单击左侧导航栏中 **集群**，进入“集群管理”页面。
2. 单击 WordPress 服务所在的集群 ID，选择**服务与路由** > **Service**。

3. 在 Service 列表页面，复制 WordPress 服务的负载均衡 IP，如下图所示：



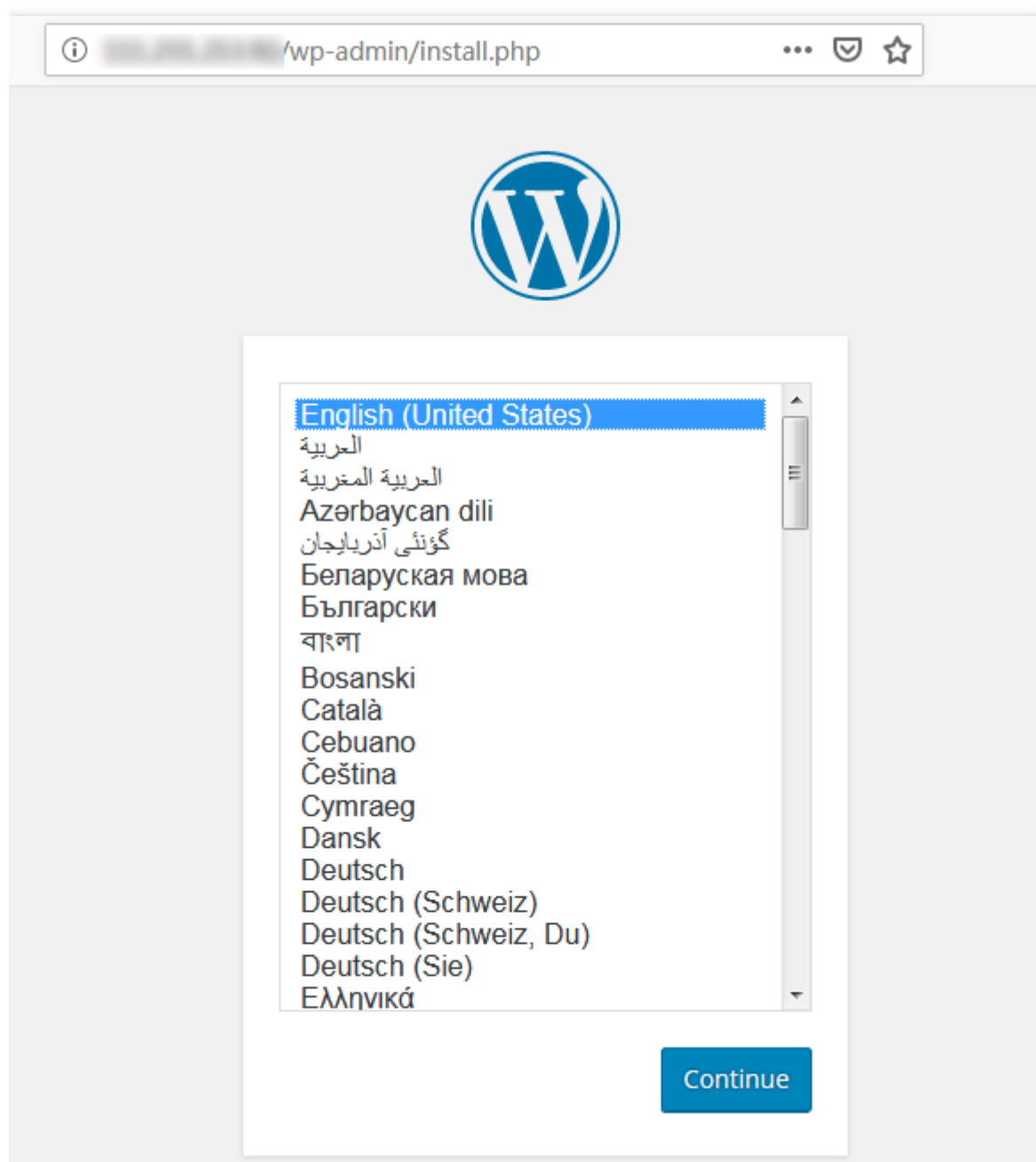
4. 在浏览器地址栏输入负载均衡 IP，按“Enter”即可访问服务。

通过服务名称访问服务

集群内的其他服务或容器可以直接通过服务名称访问。

验证 WordPress 服务

服务创建成功，访问服务时直接进入 WordPress 服务器的配置页。如下图所示：



更多 WordPress 设置

若容器创建失败，可查看 [事件常见问题](#)。

构建简单 Web 应用

最近更新时间：2023-02-02 17:05:22

操作场景

本文档指导您使用腾讯云容器服务构建一个简单的 Web 应用。

Web 应用分为以下两部分：

- 前端服务，用于处理客户端的查询和写入请求。
- 数据存储服务，使用 redis。将写入的数据存放到 redis-master，访问 redis-slave 进行读取操作。redis-master 和 redis-slave 通过主从复制来保持数据同步。

该应用是 kubernetes 项目自带的示例，详情请参见 [Guestbook App](#)。

前提条件

- 已 [注册腾讯云账户](#)。
- 已创建集群。如未创建，请参见 [创建集群](#)。

操作步骤

创建 redis-master 服务

1. 登录容器服务控制台，选择左侧导航栏中的 [集群](#)。
 2. 单击需要创建应用的集群 ID，进入集群基本信息页，在 **工作负载 > Deployment** 中单击 **新建**。
 3. 在“新建Deployment”页面，设置工作负载基本信息。主要参数信息如下，其余选项保持默认设置：
- **工作负载名**：本文以 redis-master 为例。

说明

Deployment 参数详情请参考 [创建 Deployment](#)。

4. 根据以下提示，设置**实例内容器**。主要参数信息如下，其余选项保持默认设置：

- **名称**：输入实例容器名称，本文以 **master** 为例。
- **镜像**：输入 `ccr.ccs.tencentyun.com/library/redis`。
- **镜像版本（Tag）**：输入 `latest`。
- **镜像拉取策略**：本文以不进行设置使用默认策略为例。

5. 根据以下提示，进行工作负载的访问设置。

- **Service**：勾选“启用”。
- **服务访问方式**：选择“仅在集群内访问”。
- **端口映射**：选择 TCP 协议，并将服务端口和容器端口都设置为6379。其它服务可以通过服务名称 `redis-master` 以及端口6379访问到 `master` 容器。

6. 单击**新建 Deployment**，完成 `redis-master` 服务的创建。

创建 `redis-slave` 服务

1. 登录容器服务控制台，选择左侧导航栏中的 **集群**。
2. 单击需要创建应用的集群 ID，进入集群基本信息页，在**工作负载 > Deployment** 中单击**新建**。
3. 在“新建Deployment”页面，设置工作负载基本信息。主要参数信息如下，其余选项保持默认设置：

- **工作负载名**：要创建的工作负载的名称，本文以 `redis-slave` 为例。

4. 根据以下提示，设置**实例内容器**。主要参数信息如下，其余选项保持默认设置：

- **名称**：输入实例容器名称，本文以 `slave` 为例。
- **镜像**：输入 `ccr.ccs.tencentyun.com/library/gb-redisslave`。
- **镜像版本（Tag）**：输入 `latest`。
- **镜像拉取策略**：请按需选择，本文以不进行设置使用默认策略为例。
- **环境变量**：输入 `GET_HOSTS_FROM = dns`

5. 根据以下提示，进行工作负载的访问设置。

- **Service**：勾选“启用”。
- **服务访问方式**：选择“仅在集群内访问”。
- **端口映射**：选择 TCP 协议，并将服务端口和容器端口都设置为6379。其它服务可以通过服务名称 `redis-master` 以及端口6379访问到 `master` 容器。

7. 单击**新建Deployment**，完成 `redis-slave` 服务的创建。

创建 frontend 服务

1. 登录容器服务控制台，选择左侧导航栏中的 **集群**。
2. 单击需要创建应用的集群 ID，进入集群基本信息页，在**工作负载 > Deployment** 中单击**新建**。
3. 在“新建Deployment”页面，设置工作负载基本信息。主要参数信息如下，其余选项保持默认设置：

- **工作负载名**：要创建的工作负载的名称，本文以 frontend 为例。

4. 根据以下提示，设置**实例内容器**。主要参数信息如下，其余选项保持默认设置：

- **名称**：输入实例内容器名称，本文以 frontend 为例。
- **镜像**：输入 `ccr.ccs.tencentyun.com/library/gb-frontend`。
- **镜像版本 (Tag)**：输入 latest。
- **镜像拉取策略**：请按需选择，本文以不进行设置使用默认策略为例。
- **环境变量**：输入 `GET_HOSTS_FROM = dns`

5. 根据以下提示，进行工作负载的访问设置。

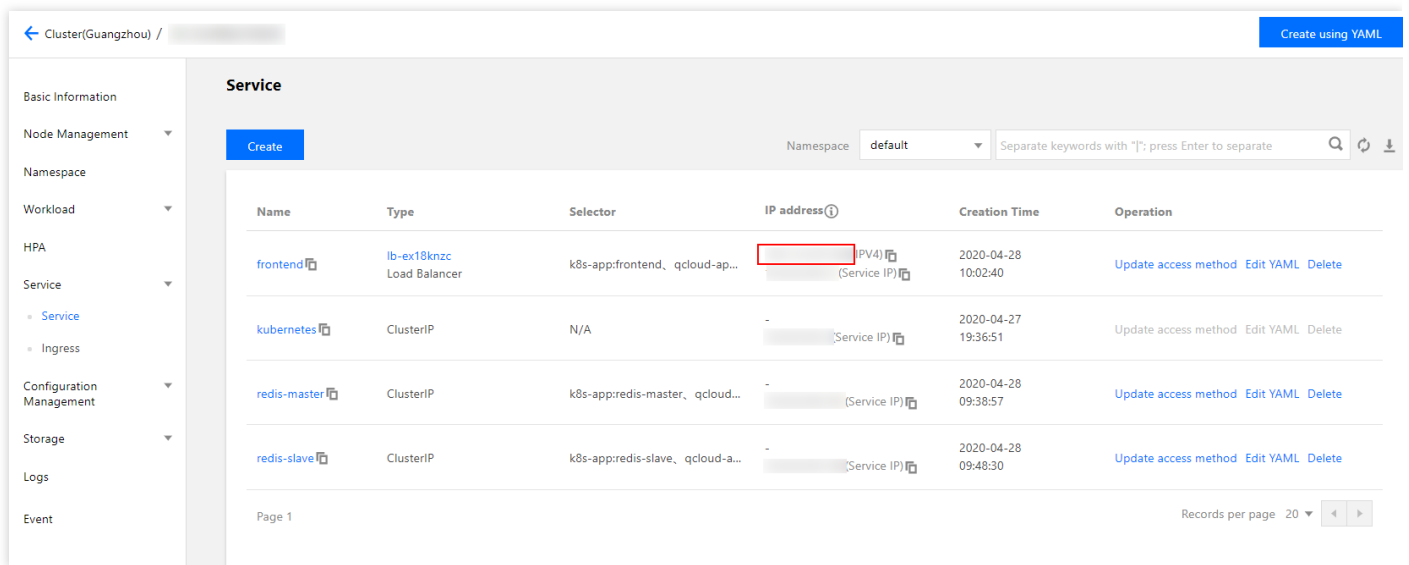
- **Service**：勾选“启用”。
- **服务访问方式**：选择“提供公网访问”。
- **端口映射**：选择 TCP 协议，并将服务端口和容器端口都设置为80。用户通过浏览器访问负载均衡 IP 即可访问到 frontend 容器。

8. 单击**新建Deployment**，完成 frontend 服务的创建。

验证 Web 应用

1. 在集群详情页，选择**服务与路由 > Service**。

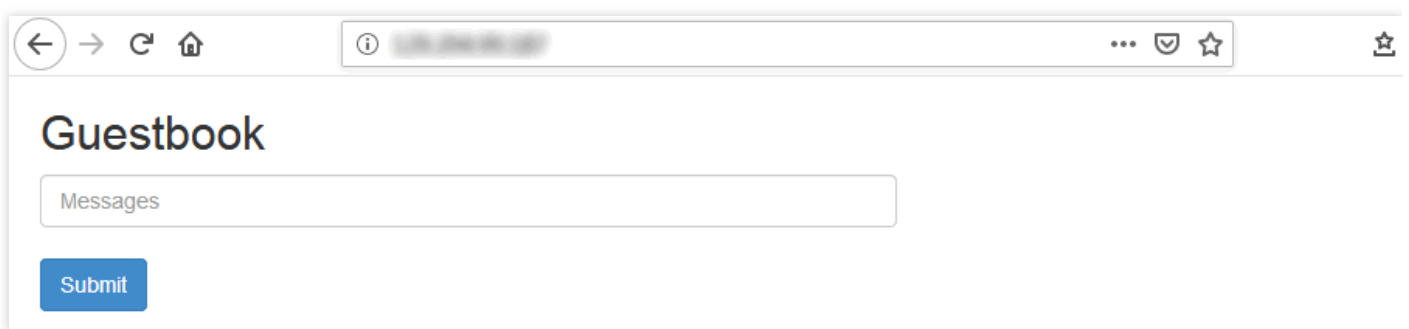
2. 在 Service 页面，复制 frontend 服务的负载均衡 IP。如下图所示：



说明：

- 在创建 redis-master 和 redis-slave 服务时，因设置了**仅在集群内访问**的访问方式，服务只具备一个内网 IP，且只能在集群内被其它服务访问。
- 在创建 frontend 服务时，因设置了**提供公网访问**的访问方式，服务具备负载均衡 IP（即公网 IP）和内网 IP，可在集群内被其它服务访问，也可通过公网访问。

3. 使用浏览器访问 frontend 服务的负载均衡 IP，返回如下图所示页面，即表示可以正常访问 frontend 服务。



4. 在输入框中输入任意的字符串后单击 **Submit**，发现输入的记录已被保存，并且展示在页面下方。

刷新浏览器页面，重新访问该服务 IP 地址。原输入的数据依然存在，即表示输入的字符串已经保存到 redis 中。

开发实践

以下示例代码是 Guestbook App 的 frontend 服务的完整代码，当 frontend 服务收到一个 HTTP 请求后，会进行判断是否为 set 命令：

- 如果是 `set` 命令，则取出参数中的 `key` 和 `value`，连接到 `redis-master` 服务。并将 `key` 和 `value` 设置到 `redis-master` 中。
- 如果不是 `set` 命令，则连接到 `redis-slave` 服务，获取参数 `key` 对应的 `value` 值，并返回到客户端进行展示。

```
<?php
error_reporting(E_ALL);
ini_set('display_errors', 1);
require 'Predis/Autoloader.php';
Predis\Autoloader::register();
if (isset($_GET['cmd']) === true) {
    $host = 'redis-master';
    if (getenv('GET_HOSTS_FROM') == 'env') {
        $host = getenv('REDIS_MASTER_SERVICE_HOST');
    }
    header('Content-Type: application/json');
    if ($_GET['cmd'] == 'set') {
        $client = new Predis\Client([
            'scheme' => 'tcp',
            'host' => $host,
            'port' => 6379,
        ]);
        $client->set($_GET['key'], $_GET['value']);
        print('{"message": "Updated"}');
    } else {
        $host = 'redis-slave';
        if (getenv('GET_HOSTS_FROM') == 'env') {
            $host = getenv('REDIS_SLAVE_SERVICE_HOST');
        }
        $client = new Predis\Client([
            'scheme' => 'tcp',
            'host' => $host,
            'port' => 6379,
        ]);
        $value = $client->get($_GET['key']);
        print('{"data": "' . $value . '"}');
    }
} else {
    phpinfo();
}
```

说明事项

- 当 `frontend` 服务访问 `redis-master` 和 `redis-slave` 服务时，连接**服务名和端口**。集群自带 `dns` 服务将服务名解析成对应的服务 IP，并根据服务 IP 进行负载均衡。

例如，redis-slave 服务有三个实例。在访问 redis-slave 服务时，直接连接 redis-slave 和6379，dns 会自动将 redis-slave 解析成 redis-slave 的服务 IP（即一个浮动 IP，类似于负载均衡的 IP），并根据 redis-slave 的服务 IP 自动进行负载均衡，将请求发往某个 redis-slave 服务的实例中。

- 容器环境变量设置：
- **使用默认设置（推荐设置）**：frontend 容器运行时，会读取已设置的 GET_HOSTS_FROM 环境变量值 dns，则直接通过服务名来连接。
- **其他设置**：需通过另一个环境变量来获取 redis-master 或者 redis-slave 的域名。