容器服务

快速入门

腾讯云

版权所有：腾讯云计算（北京）有限责任公司

【版权声明】

©2013-2025 腾讯云版权所有

本文档（含所有文字、数据、图片等内容）完整的著作权归腾讯云计算（北京）有限责任公司单独所有，未经腾讯云事先明确书面许可，任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为构成对腾讯云著作权的侵犯，腾讯云将依法采取措施追究法律责任。

【商标声明】

腾讯云

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。未经腾讯云及有关权利人书面许可，任何主体不得以任何方式对前述商标进行使用、复制、修改、传播、抄录等行为，否则将构成对腾讯云及有关权利人商标权的侵犯，腾讯云将依法采取措施追究法律责任。

【服务声明】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况，部分产品、服务的内容可能不时有所调整。
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。

【联系我们】

我们致力于为您提供个性化的售前购买咨询服务，及相应的技术售后服务，任何问题请联系 4009100100或95716。

文档目录

快速入门
新手指引
快速创建一个标准集群
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

版权所有：腾讯云计算（北京）有限责任公司

快速入门
新手指引

最近更新时间：2025-06-11 10:01:22

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**服务授权**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

腾讯云提供多种容器服务，方便您部署、管理和扩展容器，您可以按需选择：
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。

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 CVM 机器至集群管理 。
- **为什么我的服务一直在启动中?**
  服务内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：

- 如果您认为文档内容对您有帮助，您可以单击文档页面底部的**有帮助**反馈具体细节并提出更多建议；如果发现产品文档的问题，如链接、内容、API 错误等，您可以单击文档页面底部的**没帮助**或直接选中问题段落后单击**文档反馈**，在弹出的对话框中填写您的宝贵意见。
- 如果遇到产品相关问题，您可通过 在线支持 寻求帮助。
- 如果您有其他疑问，可前往 腾讯云开发者社区 进行提问。

版权所有：腾讯云计算（北京）有限责任公司

快速创建一个标准集群

最近更新时间：2024-12-12 09:57:22

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，请参见 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。具体操作请参见 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，选择左侧导航中的**集群**，在集群页面单击**新建**，开始创建集群。本文以创建一个**标准集群**为例。

集群信息

在**集群信息**页面，选择集群的基础配置和网络配置，保持其他默认选项，并单击**下一步**。
基础配置说明如下：

参数名称	使用说明
集群名称	输入要创建的集群名称，不超过50个字符。本文以 “test” 为例。
Master&Etcd 维护	选择**平台管理**，详情请参见 基础配置核心参数说明 。
集群规格	根据业务实际情况选择合适的集群规格，详情请参见 如何选择集群规格 。这里我们选择 L5。
地域	建议您根据所在地理位置选择靠近的地域，可降低访问延迟，提高下载速度，详情请参见 地域和可用区 。这里我们选择**广州**。
Kubernetes 版本	推荐您选择最新版本，版本维护规则请参见 TKE Kubernetes 版本维护机制 。
运行时组件和运行时版本	推荐您选择 Containerd 支持的最新版本。Kubernetes 1.24通过 Dockershim 对 Docker 的支持已移除，TKE 新建节点的容器运行时请使用 Containerd。
镜像提供方和操作系统	TKE 提供了包含 TencentOS、Ubuntu、CentOS 在内的多款公共镜像，请根据实际情况进行选择，详情请参见 TKE 支持的公共镜像列表 。
腾讯云标签	为集群绑定标签后可实现资源的分类管理，在集群内创建的云资源会默认继承集群标签。详情请参见 创建标签并绑定资源 。

网络配置说明如下：

配置项	说明
VPC 网络	为集群内节点选择一个私有网络。这里我们选择已有的 VPC 网络。
容器网络插件	支持选择 GlobalRouter 模式、VPC-CNI 模式和 Cilium-Overlay 模式。详情请参见 如何选择容器网络模式 。这里我们选择 Global Router。
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。

组件配置

在**组件配置**页面，确认**组件信息**，并单击**下一步**。
基础配置说明如下：

配置项	说明
存储组件	TKE 提供 CSI 组件支持多种类型的存储。这里我们选择 CBS。
监控组件	该组件负责采集容器、Pod、节点维度的监控数据，对接基础监控提供容器运维所需的基础监控大盘和告警功能。

增强组件说明如下：

配置项	说明
组件	组件包含监控、镜像、DNS 等，您可按需选择，详情请参见 扩展组件概述 。这里我们暂不选择组件。

云原生服务说明如下：

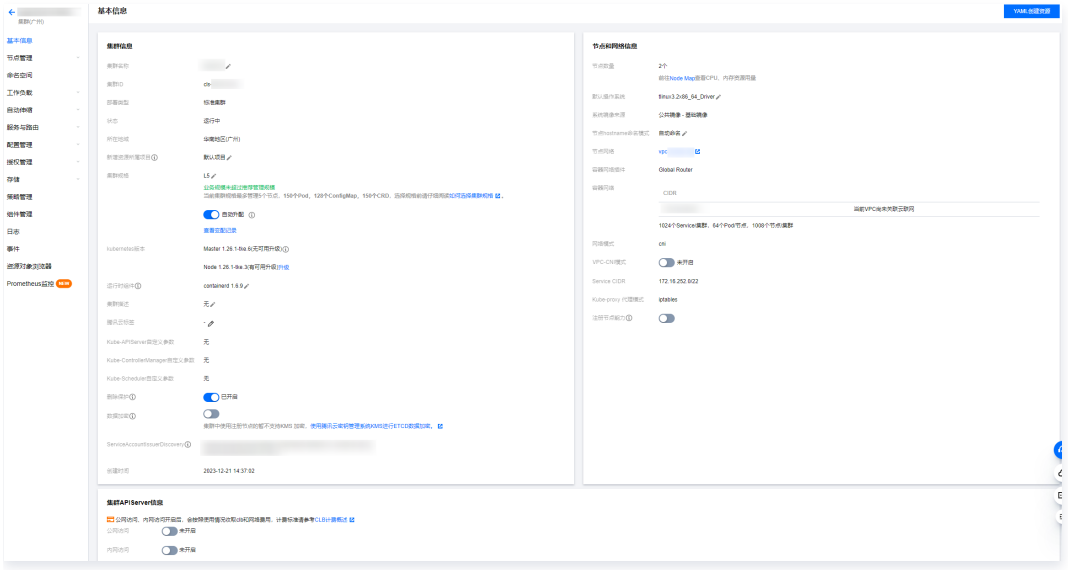
配置项	说明
Prometheus 监控服务	开启后，您可以按照实际需求灵活配置数据采集规则，并按需要配置告警规则，配置完成后即可打开 Grafana 查看监控数据。详情请参见 Prometheus 监控服务 。
日志服务	选择是否开通集群审计服务。详情请参见 集群审计 。

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，勾选**我已阅读并同意** 容器服务服务等级协议 。单击**完成**即可创建一个不包含 Worker Node 的 TKE 标准集群。
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 APIServer 信息等。如下图所示：



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**，在集群列表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：

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见 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。

入门示例
创建简单的 Nginx 服务

最近更新时间：2025-03-19 14:15:42

操作场景

本文档旨在帮助您快速在容器集群内创建一个 Nginx 服务。

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. 选择**工作负载**，在 Deployment 页单击**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。
   - **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. 设置**实例内容器**。主要参数信息如下：
   - **名称**：输入实例内容器名称，本文以 `test` 为例。
   - **镜像**：输入 `nginx` ，默认使用 DockerHub 中官方 Nginx 镜像。
   - **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，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。
   - **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 `30000~32767` 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 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**。
3. 在 Service 页面，复制 Nginx 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 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

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

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：2025-06-18 16:47:12

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。

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
```

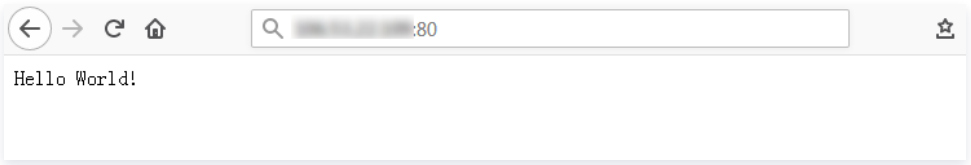
5. 测试 Hello World 程序。
   - 方式1：再次登录节点，执行以下命令。

     ```
     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
```

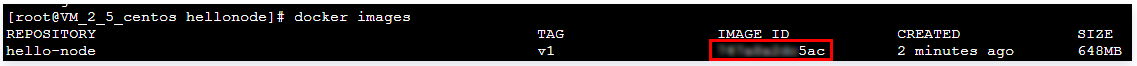
4. 执行以下命令，构建镜像。

```
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 IMAGEID。如下图所示：



上传该镜像到 qcloud 镜像仓库

创建容器镜像服务个人版实例，详情请参见 容器镜像服务个人版快速入门 。
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 [redacted] ccr.ccs.tencentyun.com/[redacted]test/helloworld:v1
[root@VM_2_5_centos hellonode]# sudo docker push ccr.ccs.tencentyun.com/[redacted]test/helloworld:v1
The push refers to repository [ccr.ccs.tencentyun.com/[redacted]test/helloworld]
7357e3a21b1f: Pushed
20a6f9d228c0: Pushed
80c332ac5101: Pushed
04dc8c446a38: Pushed
1050aff7cfff: Pushed
66d8e5ee400c: Pushed
2f71b45e4e25: Pushed
v1: digest: sha256:9c139ecbb29c49f25e02d7906b9e78c6e2e274827a75603ef48fa5547ff8a620 size: 1794
```

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**。
4. 在**新建 Deployment** 页面，根据以下信息，设置工作负载基本信息。
   - **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。
   - **Labels**：key = value 键值对，本例中标签默认值为 **k8s-app = helloworld**。
   - **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“实例内容器”。
   5.1 输入实例内容器名称，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。
   - **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，复制 helloworld 服务的负载均衡 IP。

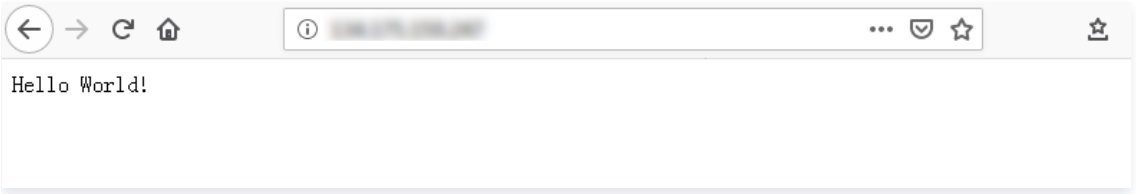
4. 在浏览器粘贴 helloworld 服务的负载均衡 IP。

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

最近更新时间：2024-09-13 10:26:51

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

1. 登录容器服务控制台，选择左侧导航栏中的 集群 。
2. 在集群列表中，单击需创建服务的集群 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。
   - **名称**：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。
   - **命名空间**：根据实际需求进行选择。
   - **Labels**：本例中标签默认值为 `k8s-app = wordpress` 。
   - **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内容器。如下图所示：

主要参数信息如下：

- **名称**：输入实例内容器名称，本文以 test 为例。
- **镜像**：输入 `wordpress` 。
- **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 – 32767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 。

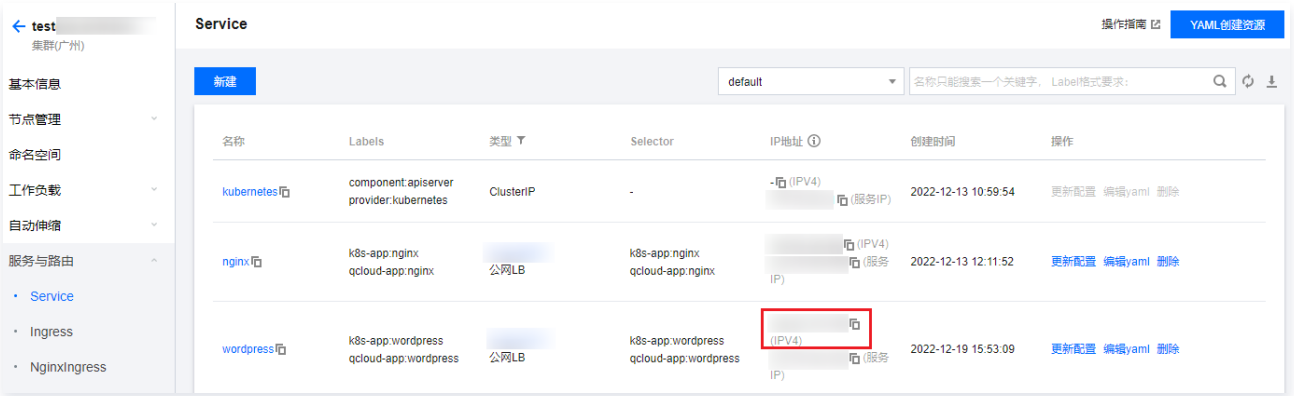
8. 单击**创建Deployment**，完成 WordPress 服务的创建。

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. 登录容器服务控制台，选择左侧导航栏中 集群 。
2. 在集群列表中，单击 WordPress 服务所在的集群 ID，进入集群详情页。
3. 在左侧导航中选择**服务与路由** > **Service**，进入 Service 列表页面。
4. 复制 WordPress 服务的负载均衡 IP，如下图所示：



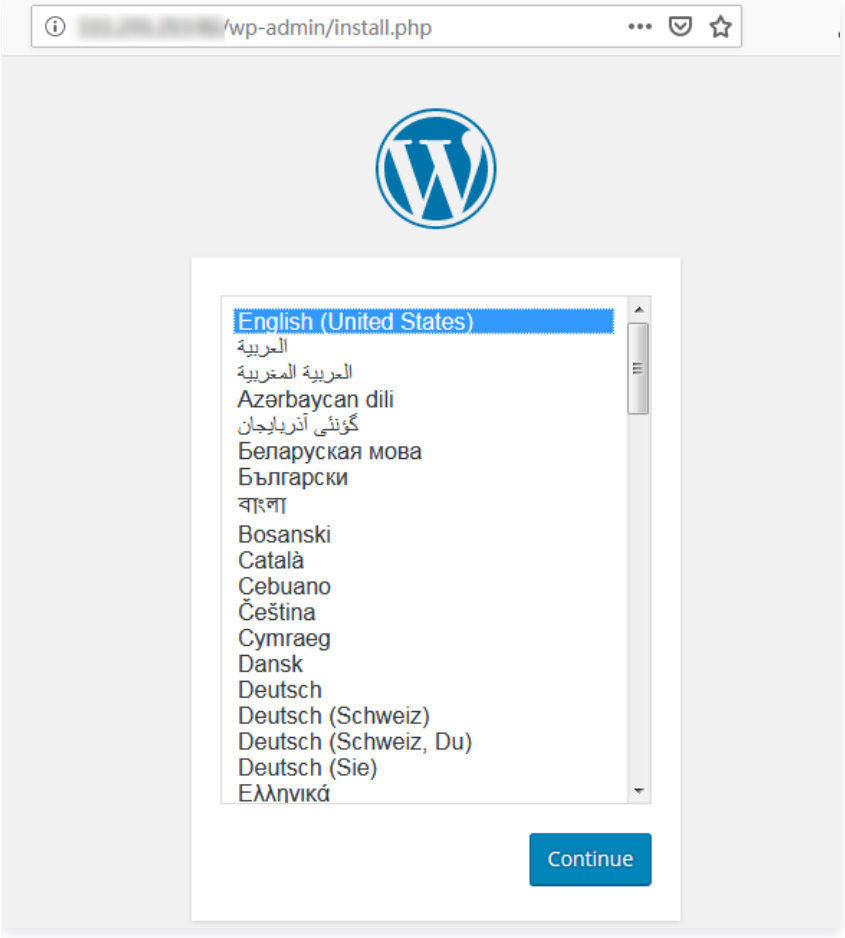
5. 在浏览器地址栏输入负载均衡 IP，按 “Enter” 即可访问服务。

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：2024-11-24 15:02:01

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“实例内容器”。如下图所示：**

主要参数信息如下：

- **名称**：输入实例内容器名称，本文以 test 为例。

- **镜像**：输入 `wordpress` 。
- **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_USER: 该云数据库的用户名称
  WORDPRESS_DB_PASSWORD: 初始化时填写的密码
  WORDPRESS_DB_NAME: 该云数据库的名称

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。

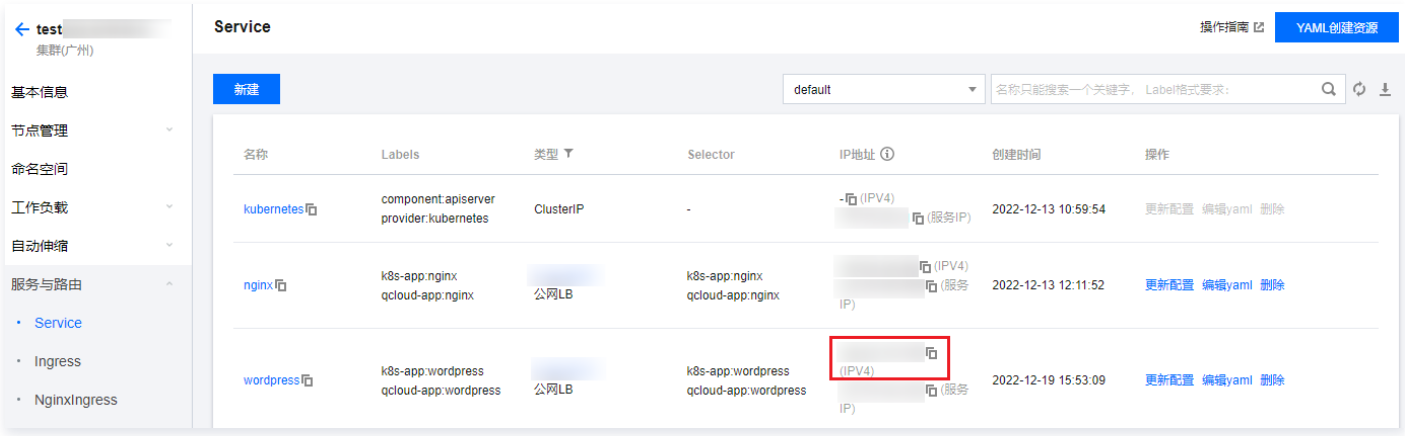
访问 WordPress 服务

可通过以下两种方式访问 WordPress 服务。

通过负载均衡 IP 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：



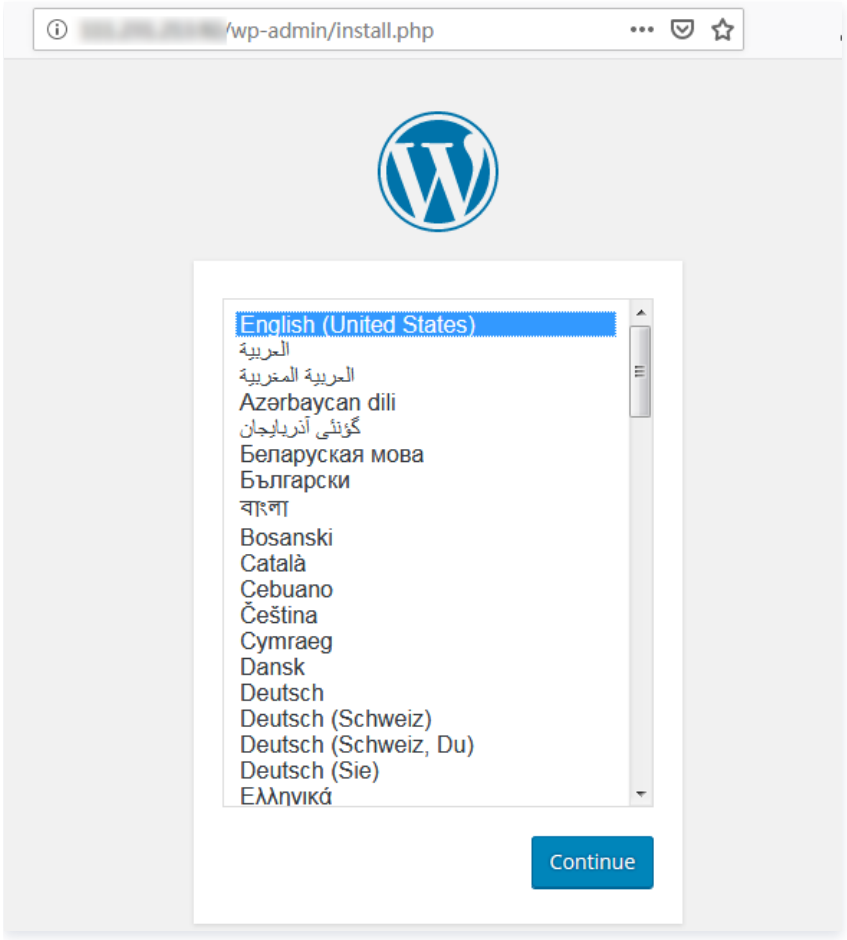
4. 在浏览器地址栏输入负载均衡 IP，按 “Enter” 即可访问服务。

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：2024-12-07 17:11:53

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。该应用由以下两部分组成：

- 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**：输入实例内容器名称，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**创建 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**：输入实例内容器名称，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-slave 以及端口6379访问到 slave 容器。

6. 单击**创建 Deployment**，完成 redis-slave 服务的创建。

创建 frontend 服务

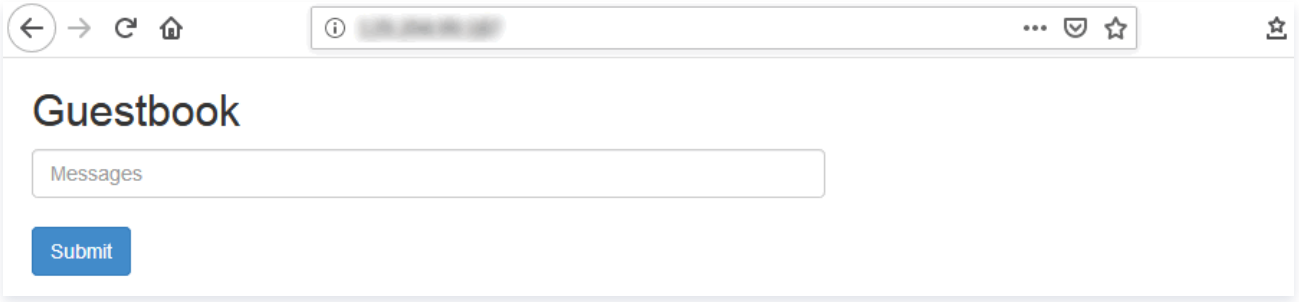
1. 登录容器服务控制台，选择左侧导航栏中的 集群 。
2. 单击需要创建应用的集群 ID，进入集群基本信息页，在**工作负载 > Deployment** 中单击**新建**。
3. 在**新建 Deployment** 页面，设置工作负载基本信息。主要参数信息如下，其余选项保持默认设置：
   - **工作负载名**：要创建的工作负载的名称，本文以 frontend 为例。
4. 根据以下提示，设置**实例内容器**。主要参数信息如下，其余选项保持默认设置：
   - **名称**：输入实例内容器名称，本文以 frontend 为例。
   - **镜像**：输入 `ccr.ccs.tencentyun.com/library/gb-frontend` 。
   - **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“公网 LB 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。
6. 单击**创建 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，找到访问入口列，复制 frontend 服务的负载均衡 IP。

> **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**。集群自带的 DNS 服务会将服务名解析为对应的服务 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，可以直接连接 redis-slave 和端口6379。DNS 会自动将 redis-slave 解析为 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，将请求发送到某个 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，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。