

# **Tencent Kubernetes Engine**

クイックスタート 製品ドキュメント





#### Copyright Notice

©2013-2024 Tencent Cloud. All rights reserved.

Copyright in this document is exclusively owned by Tencent Cloud. You must not reproduce, modify, copy or distribute in any way, in whole or in part, the contents of this document without Tencent Cloud's the prior written consent.

Trademark Notice



All trademarks associated with Tencent Cloud and its services are owned by Tencent Cloud Computing (Beijing) Company Limited and its affiliated companies. Trademarks of third parties referred to in this document are owned by their respective proprietors.

#### Service Statement

This document is intended to provide users with general information about Tencent Cloud's products and services only and does not form part of Tencent Cloud's terms and conditions. Tencent Cloud's products or services are subject to change. Specific products and services and the standards applicable to them are exclusively provided for in Tencent Cloud's applicable terms and conditions.



### カタログ:

クイックスタート

初心者ガイド

標準クラスターのクイック作成

ビギナー向け事例

簡易Nginxサービスの構築

Hello Worldサービスの手動構築

単一インスタンス版のWordPress

TencentDBを用いたWordPress

簡易Webアプリケーションの構築



# クイックスタート 初心者ガイド

最終更新日::2023-05-05 20:02:24

本文では、Tencent Kubernetes Engine(TKE)についてわかりやすく説明します。ユーザーがこのガイドに従ってTKEを簡単に開始することができます。

### 1.テンセントクバネティスエンジンとは?

Tencent Kubernetes Engine(TKE)は、ネイティブのkubernetesをベースとしてコンテナを中心に据えた、拡張性が高くハイパフォーマンスなコンテナ管理サービスを提供します。Tencent CloudのlaaS製品との密接な連携により、お客様のビジネスの速やかなコンテナ化をお手伝いします。詳細については、製品概要をご参照ください。TKEは、TKEコンソールによってクラスターやサービスの運用をサポートします。

### 2.TKE料金について

TKEは、さまざまな種類のKubernetesクラスターの作成をサポートしており、クラスターの種類によって課金項目と課金基準が異なります。課金モデルや具大的な価格については、購入ガイドをご参照ください。

### 3.TKEの使用

#### 3.1登録および認証

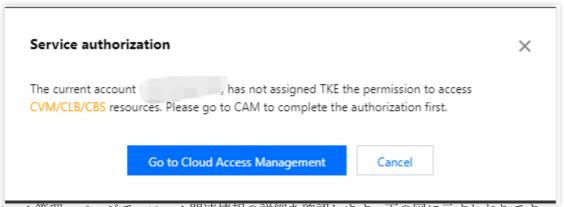
TKEを使用する前に、Tencent Cloudアカウントの登録と実名認証を完了する必要があります。

#### 3.2 ロール権限授与

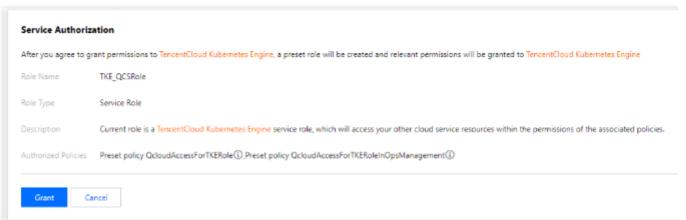
他のクラウドサービスリソースに正常にアクセスするには、現在のサービスロールに権限を授与してください。
Tencent Cloudコンソールにおいて、**クラウド製品 > TKE**を選択し、TKEコンソールに進み、インターフェースのプロンプトに従ってTKEを権限承認します。サービスの承認を完了して関連のリソース操作権限を取得した時点から、クラスターの作成を開始できます。操作手順は次のとおりです。

1. 「サービスの権限承認」というポップアップウィンドウの情報を確認し、**CAMへ**をクリックします。下の図に示すとおりです。





2. 「ロール管理」ページで、ロール関連情報の詳細を確認します。下の図に示すとおりです。



3. **承認に同意する**をクリックすると承認操作が完了し、直ちにTencent Kubernetes Engine購入ページに移動して 関連製品を購入できます。

#### 3.3クラスターの作成

標準マネージド版クラスターをクイック作成する方法については、標準クラスターのクイック作成のドキュメントをご参照ください。標準マネージド版クラスターを作成する全プロセスについては、クラスターの作成のドキュメントをご参照ください。

より多くのタイプのクラスターを使用する必要がある場合は、Serverlessクラスターの作成、コンテナインスタンスの作成およびエッジクラスターの作成のドキュメントをご参照ください。

#### 3.4 ワークロードのデプロイ

TKEは、イメージデプロイおよびYAMLファイルオーケストレーションの両方の方法でワークロードをデプロイできます。

イメージテンプレートを介してステートレスロードをデプロイ(Deployment)したい場合、操作の詳細については、シンプルなNginxサービスの作成またはシングルインスタンス版WordPressの作成をご参照ください。カスタムイメージを介してワークロードをデプロイしたい場合、操作の詳細については、Hello Worldサービスの手動ビルドをご参照ください。

#### 3.5クラスターの運用保守



コンテナサービスTKEは、クラスター、アプリケーション、ストレージ、ネットワークなどのモジュールの管理プラットフォームとして機能します。詳細な情報や実践的な作業が必要な場合は、以下を参照してさらに理解するうえで使用してください。

次のことをしたい場合	以下をご参照ください	
KubernetesのコマンドラインツールKubectlを介して、ローカルのクライアントマシンからTKEクラスターに接続します	クラスターとの接続	
実行中のKubernetesクラスターをアップグレードします	クラスターのアップグレード	
作成済みのKubernetesクラスターにインスタンスを追加し ます	ノードの追加	
Kubernetesクラスター内ノードを管理します	ノードプールの作成	
コンソールからKubernetesのネイティブオブジェクトを直 接操作します	Kubernetesオブジェクトの管理	
Serviceメソッドを使用して、コンテナ群に固定アクセス ポータルを提供します	Service基本機能	
Ingressリソースによって異なる転送ルールを設定します	Ingress管理	
TKEのストレージ機能を使用します	ストレージ管理の概要	
コンテナネットワークのアドレス範囲内のIPアドレスをク ラスター内コンテナに割り当てます	コンテナネットワークの概要	
Kubernetesクラスター内サービスログのストレージと分析 を行います	ログキャプチャ	
クラスターのモニタリングを行います	インスタンス管理のモニタリング	
Tencent Container Registry内でホストされるプライベート イメージを使用してアプリケーションのデプロイを行いま す	TCRエンタープライズ版インスタンス内のコ ンテナイメージを使用したワークロードの作 成	

### 4.入門必読

#### 基幹ネットワークでTKEを使いたいのですが、可能ですか

いいえ、TKEは現時点ではプライベートネットワークのみをサポートしています。

#### 既存のCloud Virtual Machine(CVM)を選択してクラスターに加えることは可能ですか

可能です。クラスター作成が完了した後は、既存CVMを追加できます。詳細については、既存ノードの追加をご



参照ください。

#### サービスが起動し続けた状態なのはなぜですか

サービス内コンテナに継続的に実行されているプロセスがない場合、サービスは起動し続けた状態になります。 サービスの起動に関するご質問については、イベントに関するよくあるご質問をご参照ください。

#### クラスターを作成する前、どのようにネットワークを計画すればよいですか

クラスターを作成する場合、クラスターネットワークセグメントとコンテナネットワークは重複できません。通 常、Virtual Private Cloud(VPC)内のサブネットを選択して、クラスターのノードネットワークに用います。詳細に ついては、コンテナネットワークとクラスターネットワークの説明をご参照ください。

#### 作成済みのサービスには、どのようにアクセスするのですか

提供されるアクセスポータルはアクセス方法によって異なります。詳細については、サービスアクセス方法。

#### コンテナはどのようにパブリックネットワークにアクセスするのですか

コンテナは存在するノードにブリックネットワークIP帯域幅がある場合、コンテナはパブリックネットワークに直接アクセスできます、コンテナは存在するノードにブリックネットワークIP帯域幅がいない場合、パブリックネットワークにアクセスするにはNATゲートウェイを使ってください。

#### イメージの作り方がわからないのですが、TKEを使えばよいのでしょうか

TKEに統合されたHelm 3.0関連機能により、helm chart、コンテナイメージ、ソフトウェアサービスなど、さまざまな製品やサービスの提供が可能です。作成済みのアプリケーションは、指定したクラスター内で実行され、対応する機能が提供されます。詳細については、アプリケーション管理をご参照ください。

私の業務では、たくさんのテキストや環境変数を設定する必要がありますが、どのように管理すればよいですか 設定項目から設定ファイルを管理できます。

#### サービス同士で相互アクセスしたい場合は、どうすればよいですか

クラスター下の同じNamespaceのサービス同士であれば直接相互アクセスができますが、異なるNamespaceのサービスは、 <service-name>.<namespace-name>.svc.cluster.local という形式によってアクセスする必要があります。

### 5. フィードバックおよびアドバイス

テンセントクバネティスエンジン製品およびサービスを使用する上で何らかの質問や助言がある場合は、以下の 方法でフィードバックしてください。後ほど専門スタッフがお客様のお問い合わせに回答いたします:

製品ドキュメントの問題などを発見した場合(リンク、内容、APIエラーなど)は、ドキュメントのページ右側の ドキュメントに**関するフィードバック**をクリックするか、または問題が存在するコンテンツを選択してフィード バックしてください。

製品に関する問題が発生した場合、チケットを提出からサポートを求めることができます。



# 標準クラスターのクイック作成

最終更新日::2022-04-14 16:56:29

本文では、TKEを使用してコンテナクラスターを速やかに作成する方法について説明します。

### ステップ1:Tencent Cloudアカウントの登録

Tencent Cloudのアカウント登録が済んでいる場合は、このステップを無視してかまいません。

ここをクリックしてTencent

### ステップ2:オンラインチャージ

TKEは当面、サービス料金がかかりません。ユーザーが実際に使用したクラウドリソースに応じて課金されます。ここで「ホスティングクラスター」を作成します。このモードでは、クラスターのワーカーノード、永続的なストレージおよびサービスにバインディングされたCLBなどのサービスについて料金を支払ってください。購入前に、アカウントにチャージしてください。詳細については、オンラインチャージドキュメントをご参照ください。

### ステップ3:サービス認証

Tencent Cloudコンソールで**クラウド製品>TKE**を選択し、[TKEコンソール]に移動し、画面の提示に従って、TKE 認証を行います。(TKE認証を行った場合は、このステップを省略してください。)

#### ここをクリックしてサー

## ステップ4:クラスターの新規作成

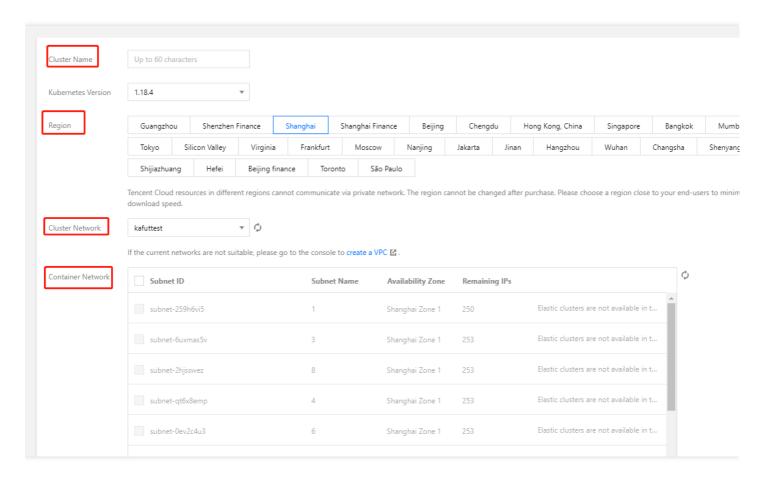
#### ここをクリックしてクラス

#### クラスターの情報

「クラスター情報」ページで、クラスター名を入力し、クラスターが存在する地域、クラスターネットワークおよびコンテナネットワークを選択します。その他のデフォルト・オプションはそのままにして、「**次へ**」をクリッ



#### クします。下図の通りです:

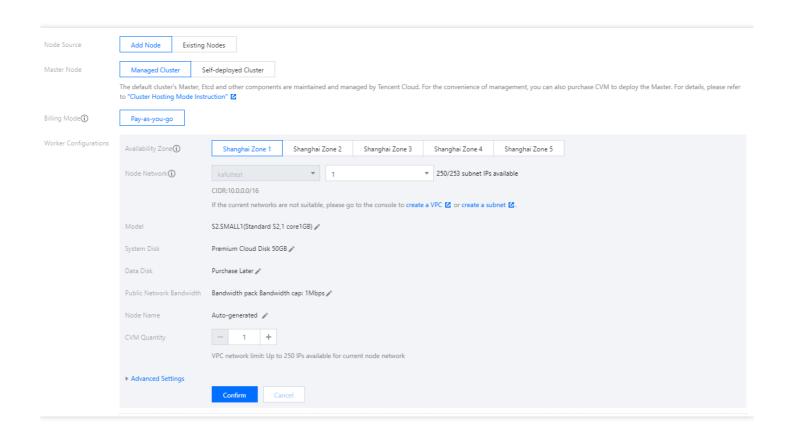


- クラスター名:作成するクラスターの名前を入力します。ここでは、「test」を例に挙げます。
- 所在地域:あなたに最も近い地域を選択してください。
- **クラスターネットワーク**: ノードのネットワークアドレス範囲内のIPアドレスをクラスター内のCVMに割り当てます。ここでは、既存のVPCネットワークを選択します。
- **コンテナネットワーク**:コンテナネットワークアドレス範囲内のIPアドレスをクラスターのコンテナに割り当てます。ここで利用可能なコンテナネットワークを選択します。

#### モデルの選択

[モデルの選択]ページで、課金モードを確認し、アベイラビリティーゾーンとそれに対応するサブネットを選択し、ノードのモデルを確認して、**[次へ]**をクリックします。下図の通りです:





- **ノードのソース**: 新規ノードと既存ノードの2つのオプションが用意されています。ここでは、「新規ノード」 を選択します。
- **Masterノード:プラットフォームホスティング**と**個別デプロイ**の2つのクラスターモードから選択できます。 ここでは、「プラットフォームホスティング」を選択します。
- 課金モード:従量課金1つの課金モードを提供します。ここでは「従量課金」を選択します。
- Worker設定: このモジュールでは、アベイラビリティーゾーンとそれに対応するサブネットを選択し、ノードのモデルを確認するだけで済みます。その他の設定項目はデフォルトのままです。
  - **。 アベイラビリティーゾーン**:最も近い地域を選択します。
  - **ノードネットワーク**:現在のVPCネットワークのサブネットを選択します。
  - **モデル**: ここでは、「S1.SMALL1 (標準型S1、1コア1GB)」を選択します。

#### CVMの設定

「CVMの設定」ページで、ログイン方法を選択し、その他の設定項目はデフォルトのままにして、「**次へ**」をクリックします。下図の通りです:



Container Directory	Set up the container and image storage directory. It's recommended to store to the data disk.						
Security Group ①	Create and bind with default secu						
	For the normal communication among nodes in the cluster, some of the ports will be open. You can check the security group and modify the rules after creating the cluster. Preview default security group rules  To configure custom security group rules, please add a security group						
Login Method	SSH Key Pair Random Password Custom Password						
Security Services	Enable for FREE						
Cloud Monitor	Free Anti-DDoS, WAF, and Cloud Workload Protection service (component installation required) Details   Enable for FREE						
Cloud Monitor	Free monitoring, analysis and alarm service, CVM monitoring metrics (component installation required) Details 🖸						
▶ Advanced Settings							

• **ログイン方式:キーの即時関連付け、パスワードの自動生成、**および**パスワードの設定**の**3**つのログイン方法があります。ここでは「パスワードの自動生成」を選択します。

#### コンポーネント構成

「コンポーネントの構成」ページでは、必要に応じてストレージ、監視、イメージなどのコンポーネントを選択できます。インストールが不要な場合は、**次へ**をクリックします。ここでは、コンポーネントをインストールせず、その他の設定項目はデフォルトのままにします。

#### 情報の確認

「情報の確認」ページで、クラスターで選択した設定情報および料金を確認し、「**完了**」をクリックします。下図の通りです:



#### **Selected Configuration**

Cluster Name

lili

Region East China(Shanghai)

Container Network GR, 192.168.0.0/16, 1024 Service/cluster, 64 Pod/node, up to 1008 nodes

Operating System Tencent Linux 2.4 64bit
Cluster Type Managed Cluster
Billing Mode Pay-as-you-go
Operating System (1) Tencent Linux 2.4 64bit

Public Image -Basic Image

Node AZ: Shanghai Zone 1

Model:S2.SMALL1(Standard S2,1 core1GB) System disk: Premium Cloud Disk 50GB

Data disk: purchase later

Public bandwidth: bandwidth pack; bandwidth cap 1Mbps

Amount:1

Add-on CBS Tencent Cloud CBS

Cluster Auditing Enabled

Fees

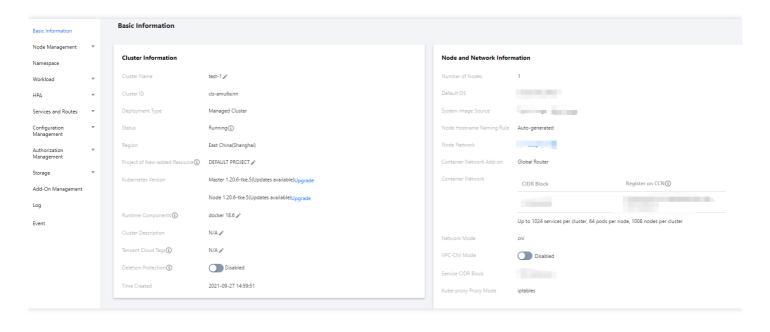
支払いが完了すると、最初のクラスターを作成できます。次に、TKEコンソールで、作成したホスティングクラスターを表示できます。

### ステップ5:クラスターの表示

作成されたクラスターがクラスターリストに表示されます。クラスターIDをクリックすると、クラスターの詳細ページに移動できます。クラスターの「基本情報」ページでは、クラスター情報、ノードおよびネットワーク情報



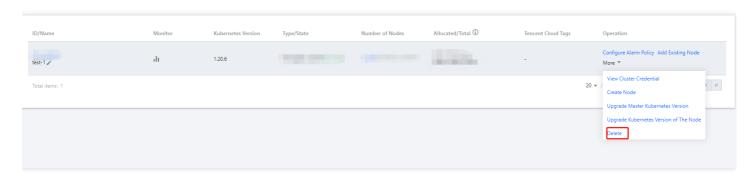
などを表示できます。下図の通りです:



### ステップ6:クラスターの削除

クラスターの起動時にリソースの消費が開始されます。このステップでは、不要なコストを回避するために、すべてのリソースをパージする方法を説明します。

1.左側のナビゲーションバーで「**クラスター**」を選択し、「クラスターの管理」ページでクラスター削除行の右側の**その他 >削除**を選択します。下図の通りです:



2.「クラスターの削除」ウィンドウで情報を確認したら、**OK**をクリックしてクラスターを削除します。

### 次の手順:クラスターの使用

本文では、TKEでクラスターを作成および削除する方法について学習しました。作成したクラスターでは、ワークロードを設定し、サービスを作成できます。一般的なタスクは次のとおりです:



- 簡易Nginxサービスの構築
- 単一インスタンス版のWordPress
- TencentDBを用いたWordPress
- Hello Worldサービスの手動構築
- 簡易Webアプリケーションの構築

## 問題が発生した場合

ご迷惑をおかけしまして、誠に申し訳ございません。[チケットを提出(https://console.tencentcloud.com/workorder) してサポートを依頼してください。



# ビギナー向け事例

# 簡易Nginxサービスの構築

最終更新日::2023-05-06 19:41:07

### 概要

このドキュメントは、コンテナクラスター内のNginxサービスを素早く理解、作成できるようにするためのものです。

### 前提条件

Tencent Cloudアカウントの登録済みであること。

すでにクラスターを作成済みであること。詳細については、クラスターの作成をご参照ください。

### 操作手順

#### Nginxサービス作成

- 1. TKEコンソールにログインし、左側ナビゲーションバーのクラスターを選択します。
- 2. **クラスター管理**ページで、作成したいサービスのクラスター**ID**を選択し、クラスター基本情報ページに進みます。
- 3. **ワークロード > Deployment**ページで、**新規作成**をクリックします。パラメータの詳しい内容は、 Deployment を作成をご参照ください。
- 4. **Deploymentの新規作成**ページで、以下の情報に基づき、ワークロードの基本情報を設定します。下図に示すとおりです。



Name	Please enter a name					
	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	Up to 1000 characters					
Namespace	default ▼					
Labels	k8s-app = nginx ×					
	Add					
	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 label key value can only include letters, numbers and separators ("-", " " "."). It must start and end with letters and numbers.					
OS type	Linux					
	Configurations are initialized when you change the OS type for the container.					
Volume (optional)	Add volume					
	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					

ワークロードの名称:ここではnginxを例にとります。

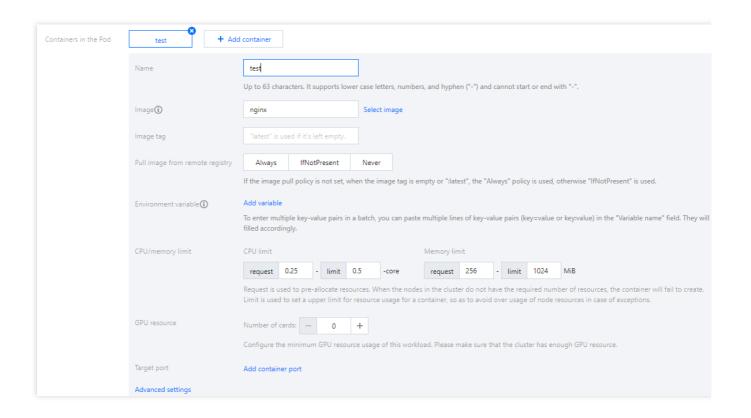
説明:ワークロードの関連情報を入力します。

タグ:この例でのタグデフォルト値は、 k8s-app = nginx です。

**ネームスペース**:実際の要求に従い選択します。デフォルトはdefaultです。

**データボリューム**:実際のニーズに応じてワークロードのマウントボリュームを設定します。詳細については、ストレージボリュームの使用説明をご参照ください。

5. 次の情報設定「インスタンス内のコンテナ」をご参照ください。下図に示すとおりです。





主なパラメータ情報は下記の通りです:

**名称**:インスタンス内のコンテナ名を入力します。ここではtestを例にとります。

**イメージ:イメージの選択**をクリックし、ポップアップボックスで**Docker Hubイメージ > nginx**を選択し、**確定** をクリックします。

イメージバージョン (Tag):デフォルト値 latest を使用します。

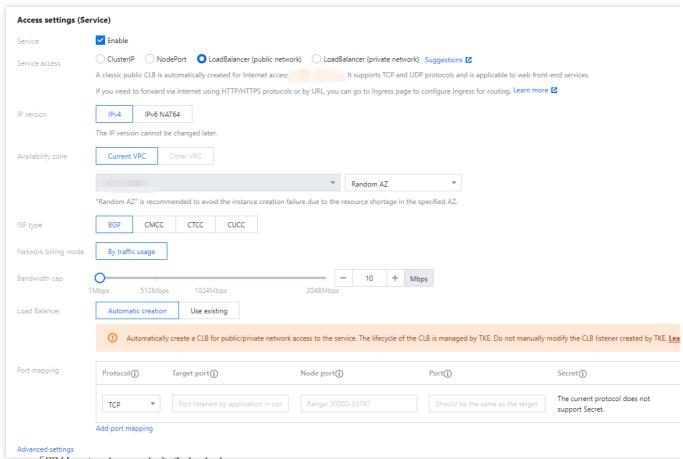
イメージプルポリシー: Always、IfNotPresent、Neverという3つのポリシーを提供しますので、必要に応じて選択してください。ここでは、**設定せずデフォルトのポリシーを使用する**を例にとります。

6. 「インスタンス数」では、次の情報に基づきサービスのインスタンス数を設定します。ここでは、**手動調節**を 例にとります。インタンス数の設定は1です。下図に示すとおりです。



7. 以下の手順に従い、ワークロードへのアクセス設定を行います。下図に示すとおりです。





**Service**:「開始」にチェックを入れます。

サービスアクセス方法:「パブリックネットワークLBアクセス」を選択します。

ロードバランサ:実際のニーズに応じて選択します。

**ポートマッピング:TCP**プロトコルを選択し、コンテナポートとサービスポートをどちらも80に設定します。

プロトコル:通信が必要なプロトコルタイプを選択します。

**コンテナポート**:コンテナ内のアプリケーションプログラムが監視するポートです。ポートの範囲は1~65535です。

**ホストポート**: Cloud Virtual Machine (CVM) IP+ホストポートを介してサービスにアクセスできます。ポートの 範囲は30000~32767で、入力しない自動割り当てです。

**サービスポート**: クラスター外ではCloud Load Balancer(CLB)ドメイン名あるいはIP+サービスポートによって サービスにアクセスでき、クラスター内ではサービス名+サービスポートによってサービスにアクセスできます。

Secret: TCP SSLプロトコルを選んだ場合のみ選択が必要です。

#### 注意

サービスが配置されているクラスターのセキュリティグループは、ノードネットワークとコンテナネットワーク を開放している必要があり、さらにポート $30000\sim32768$ も開放する必要があります。開放していないと、TKEを 使用できないといった問題が発生する場合があります。詳細については、TKEセキュリティグループの設定をご参照ください。

8. Workloadを作成をクリックし、Nginxサービスの作成を完了します。

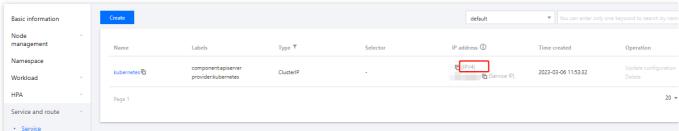
#### Nginxサービスにアクセス



次の二つの方式でNginxサービスにアクセスできます。

#### Cloud Load Balancer IPでNginxサービスにアクセス

- 1. 左側ナビゲーションバーのクラスターをクリックし、「クラスター管理」ページに進みます。
- 2. Nginxサービスが所在するクラスターIDをクリックし、サービスとルート > Serviceを選択します。
- 3. Serviceリストページで、NginxサービスのCLB IPをコピーします。下図に示すとおりです。



4. ブラウザのアドレス欄にCLB IPを入力し、「Enter」を押すとサービスにアクセスできます。

#### サービスの名前でNginxサービスにアクセス

クラスター内の他のサービスやコンテナはサービスの名前で直接アクセスできます。

#### Nginxサービス検証

サービスは作成されました。サービスにアクセスする時、直接にNginxサーバーのディフォルトウェルカムページに入ります。下図の通りです:

### 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 <u>nginx.org</u>. Commercial support is available at <u>nginx.com</u>.

Thank you for using nginx.

#### その他のNginx設定

コンテナの作成に失敗した場合は、イベントに関するよくあるご質問をご参照ください。



# Hello Worldサービスの手動構築

最終更新日::2023-05-06 19:41:07

### 概要

このドキュメントは、コンテナクラスター内のNode.js版のHello Worldのサービスをすばやく作成する方法を理解する助けとなるものです。Dockerイメージの作成方法に関する詳しいチュートリアルについては、dockerイメージの作成方法をご参照ください。

### 前提条件

クラスターを作成済みであること。詳細については、クラスターの作成をご参照ください。 ノードにログイン済みであり、そのノードにNode.jsがインストールされていること。

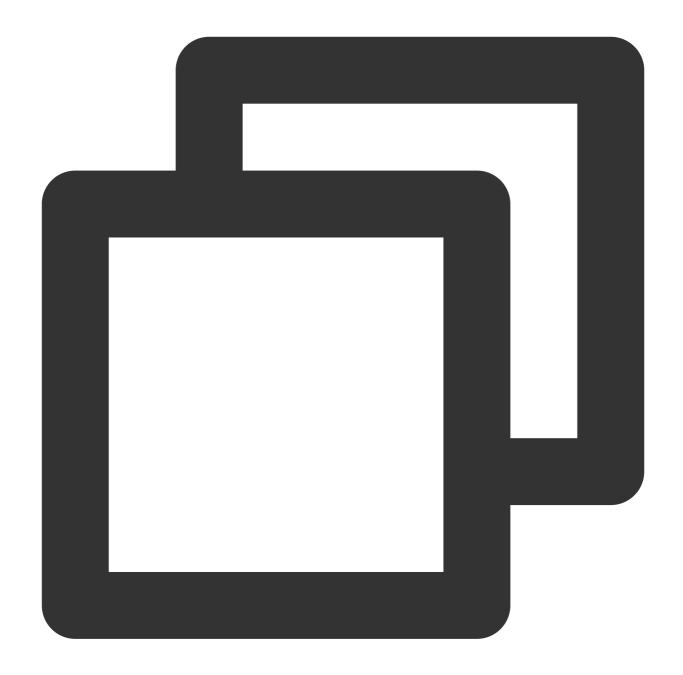
### 操作手順

コーディングとイメージ作成

#### アプリケーション作成

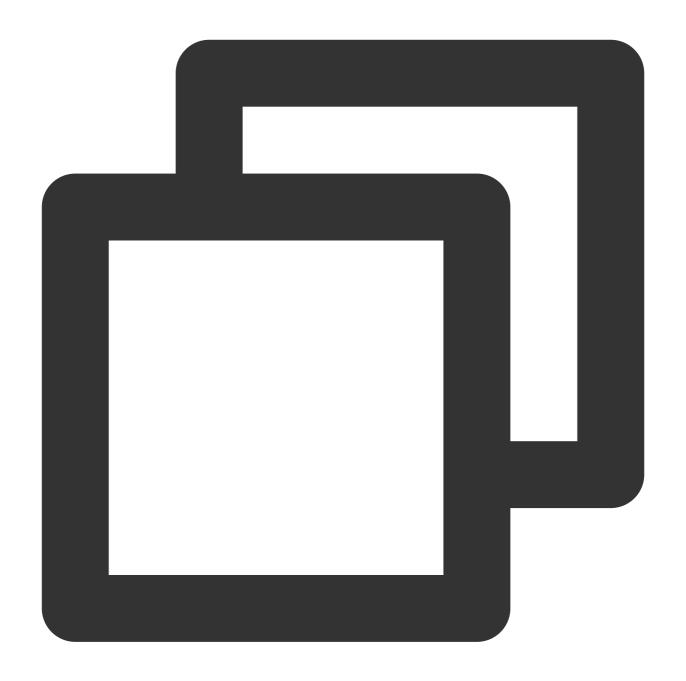
1. 以下のコマンドを順に実行し、hellonodeのフォルダを作成して入ります。





mkdir hellonode

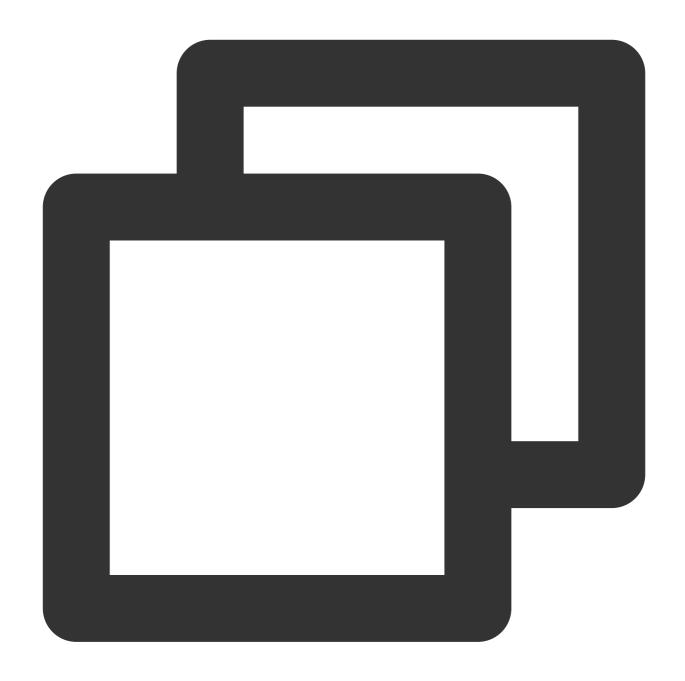




cd hellonode/

2. 以下のコマンドを順に実行し、server.jsファイルを新規作成して開きます。

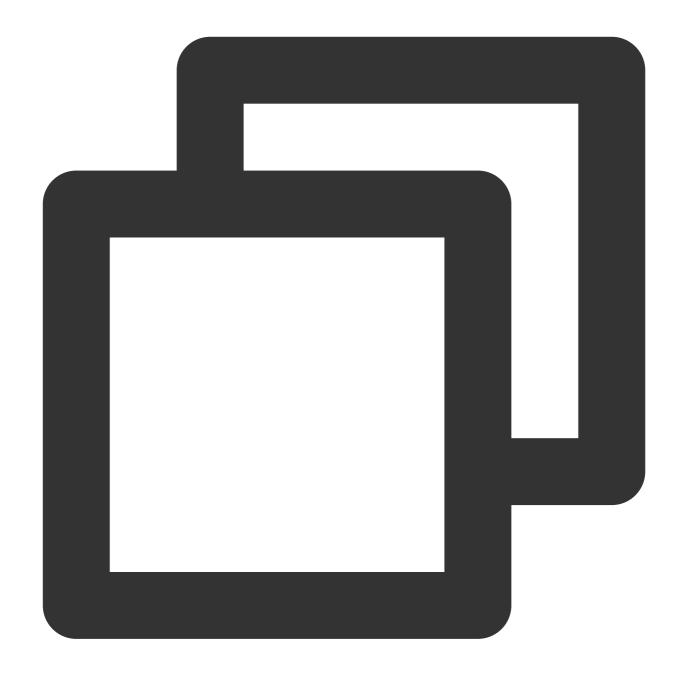




vim server.js

3. iを押して編集モードに切り替え、以下の内容をserver.jsに入力します。



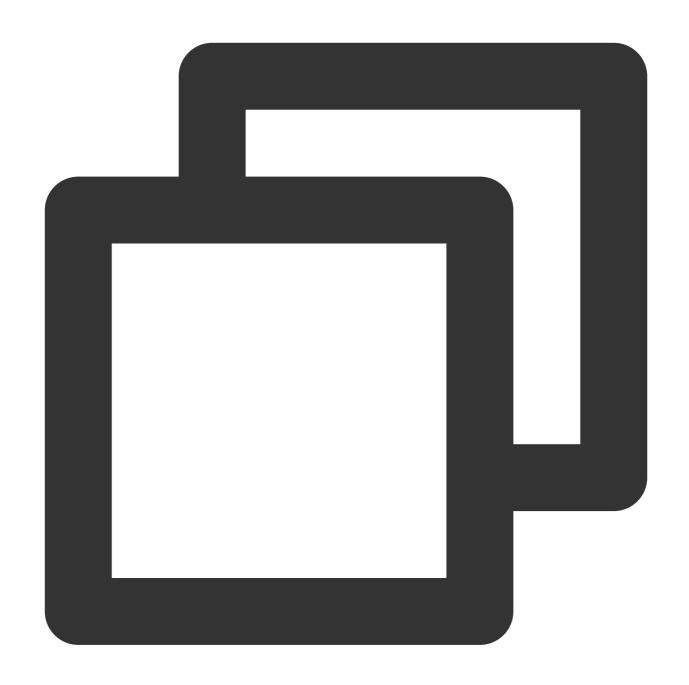


```
var http = require('http');
var handleRequest = function(request, response) {
  console.log('Received request for URL: ' + request.url);
  response.writeHead(200);
  response.end('Hello World!');
};
var www = http.createServer(handleRequest);
www.listen(80);
```

Escを押して、:wqと入力し、ファイルを保存して戻ります。



4. 以下のコマンドを順に実行し、server.jsファイルを実行します。

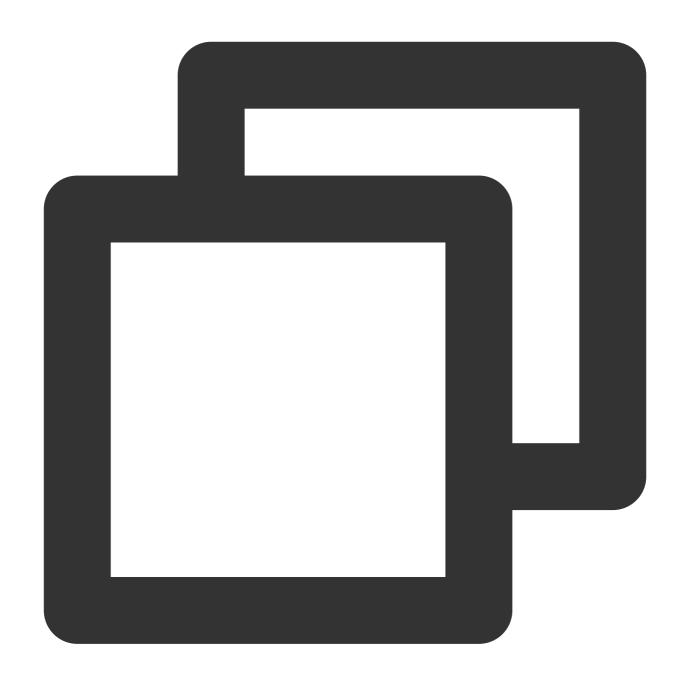


node server.js

5. Hello Worldプログラムをテストします。

方法1:ノードに再度ログインし、以下のコマンドを実行します。





curl 127.0.0.1:80

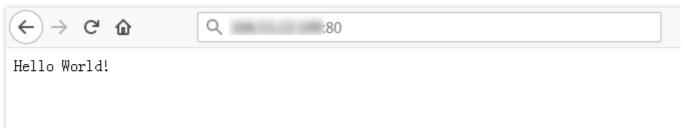
次のように表示されれば、Hello Worldプログラムが実行されたことを示します。

[root@VM\_2\_5\_centos ~] # curl 127.0.0.1:80 Hello World![root@VM\_2\_5\_centos ~]#

方法2: Cloud Virtual Machine (CVM)インスタンスのパブリックIPアドレスでローカルブラウザを開き、設定済みのポート番号の形式でアクセスします。ポート番号は80です。

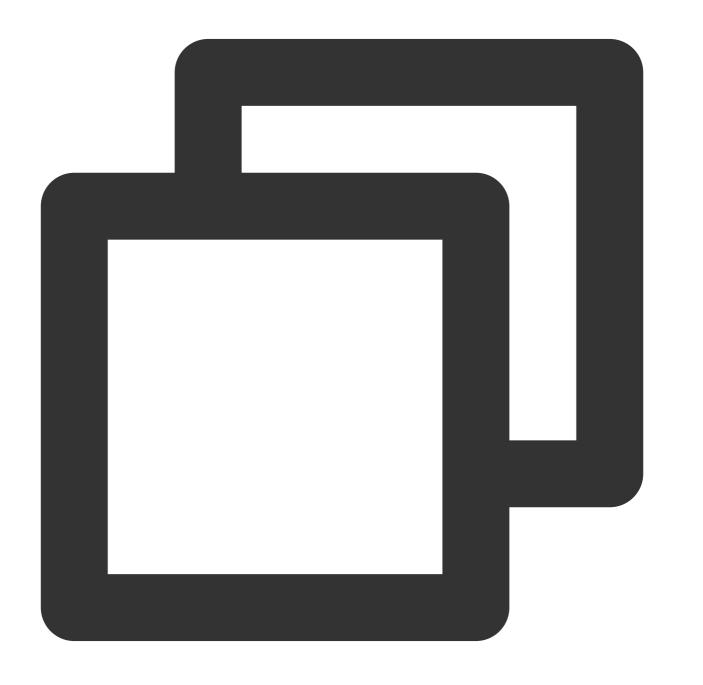
次のように表示されれば、Hello Worldプログラムが実行されたことを示します。





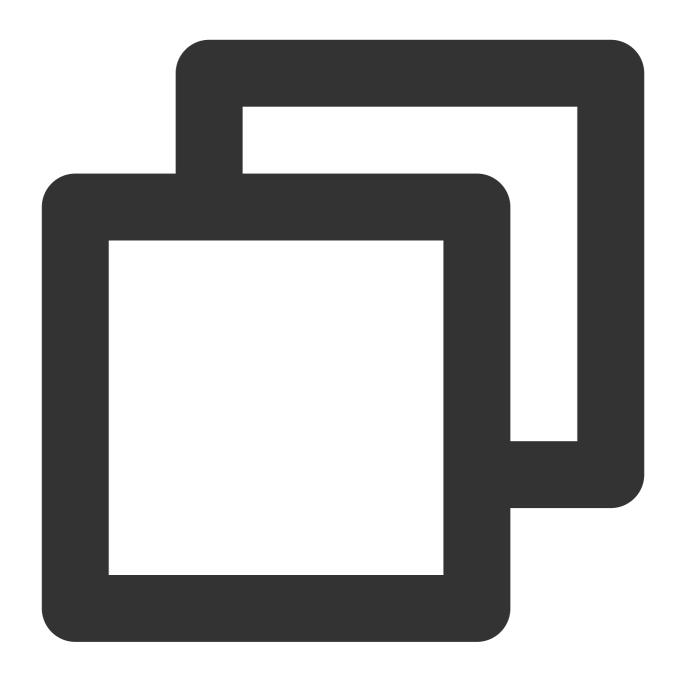
Dockerイメージ作成

1. 以下のコマンドを順に実行し、hellonodeフォルダにDockerfileファイルを作成します。



cd hellonode

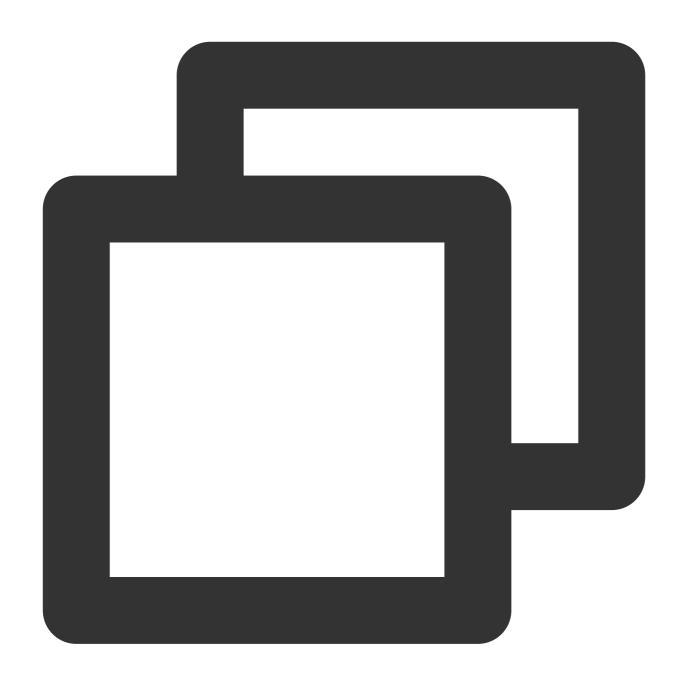




vim Dockerfile

2. iを押して編集モードに切り替え、以下の内容をDockerfileファイルに入力します。





FROM node:4.4

EXPOSE 80

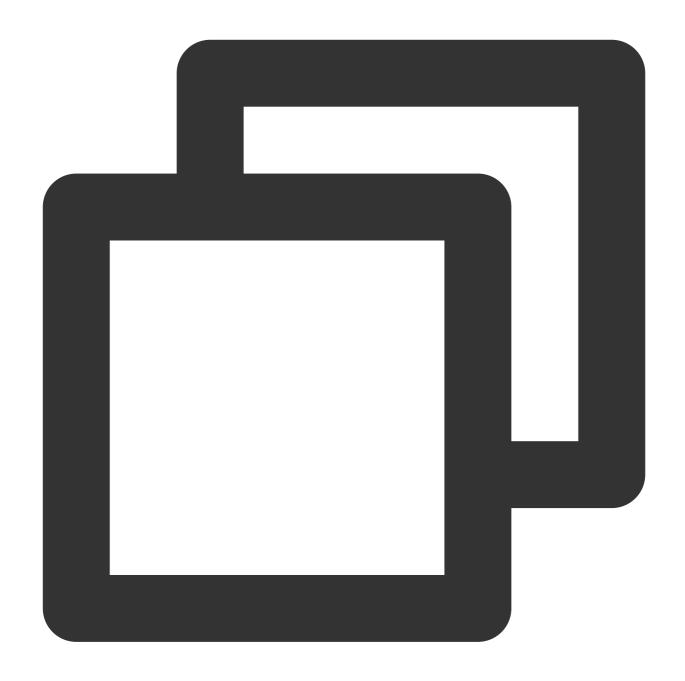
COPY server.js.

CMD node server.js

Escを押して、:wqと入力し、ファイルを保存して戻ります。

3. このノードをdockerにインストールし、dockerサービスを起動します。

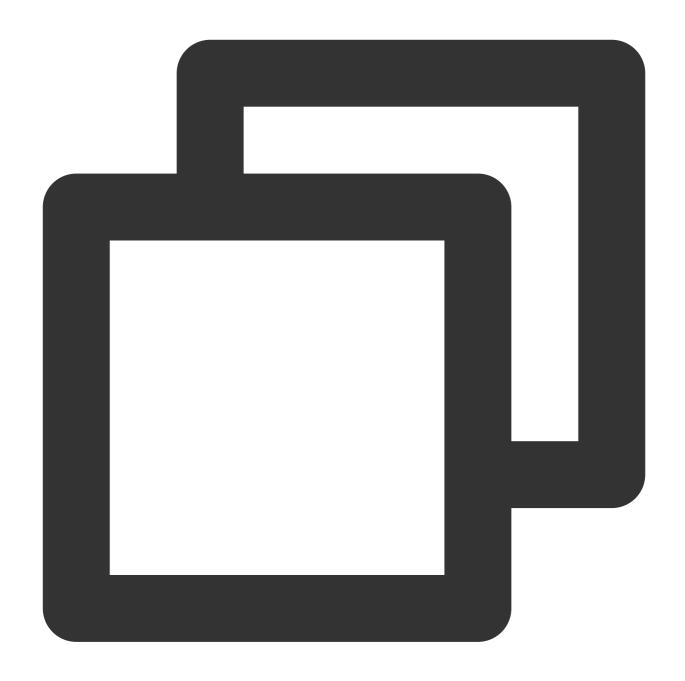




yum install -y docker
systemctl start docker

4. 次のコマンドを実行して、イメージを作成します。

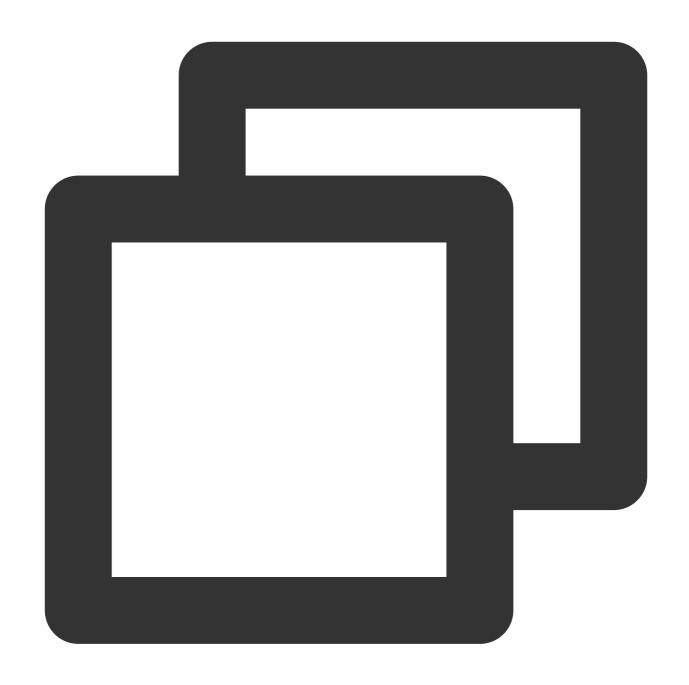




docker build -t hello-node:v1 .

5. 次のコマンドを実行して、作成したhello-nodeイメージを確認します。





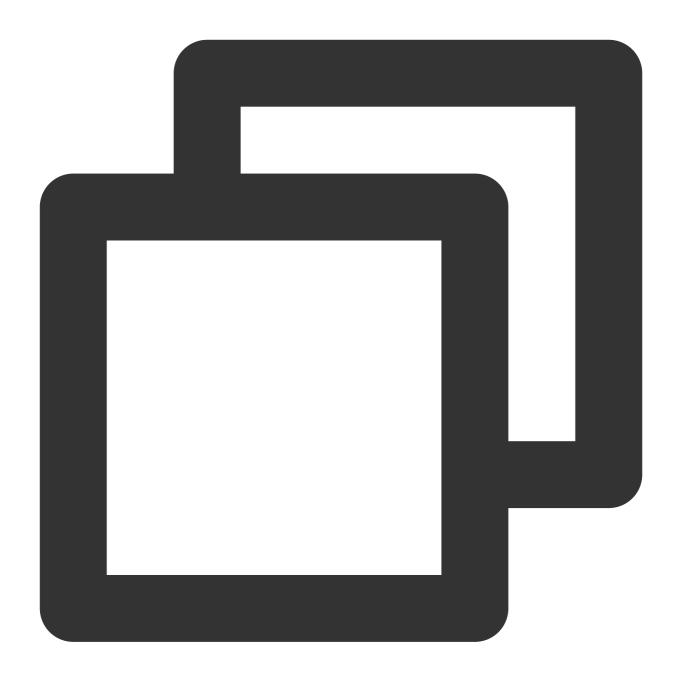
docker images

結果は次のように表示されれば、hello-nodeイメージが成功的に構築されたことを示し、このIMAGE IDを記録します。下図の通りです:



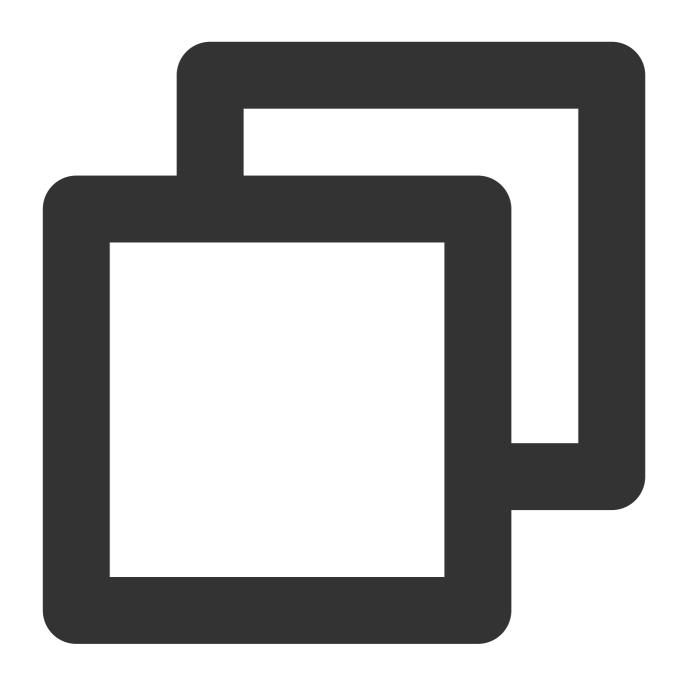
次のコマンドを順番に実行して、イメージをqcloudイメージウエアハウスにアップロードします。





docker tag IMAGEID ccr.ccs.tencentyun.com/ $\dot{\lambda}$ - $\Delta$   $Z^{-}$ -Z/hello-node:v1





docker login ccr.ccs.tencentyun.com docker push ccr.ccs.tencentyun.com/ $\lambda$ - $\Delta$ 2%- $\Delta$ 7/hello-node:v1

#### 説明

コマンドのIMAGEIDをイメージの確認に記録されたIMAGEIDに置き換えてください。

コマンド内のネームスペースを作成したネームスペースに置換してください。ネームスペースがない場合は、まずネームスペースを作成してください。詳細については、ネームスペースの作成をご参照ください。

次の結果が表示されれば、イメージが成功的にアップロードされたことを示します。

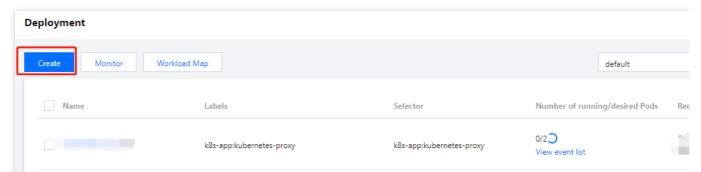


このイメージでHello Worldサービス作成

#### 注意

Hello Worldサービスを使用して作成する前に、クラスターを持っている必要があります。クラスターがない場合は、まずクラスターを作成してください。詳細については、クラスターの作成をご参照ください。

- 1. TKEコンソールにログインし、左側ナビゲーションバーから**クラスター**を選択します。
- 2. クラスター管理ページで、作成したいサービスのクラスターIDを選択し、クラスター詳細ページに進みます。
- 3. **ワークロード > Deployment**を選択し、Deploymentページで**新規作成**をクリックします。下の図に示すとおりです。



4. **Deploymentの新規作成**ページで、以下の情報に基づき、ワークロードの基本情報を設定します。下の図に示すとおりです。



Name	helloworld		letters numbers and hunbers (" ") It	must begin with a lowercase letter, and end with a nur		
	op to 03 characters, including loweres	ase i	etters, numbers, and hypnens ( - ). It	must begin with a lowercase letter, and end with a nur		
Description	helloworld serve					
			//			
Namespace	default ▼					
Labels	k8s-app	_	helloworld	×		
LUDEIS		_	nenowona			
	Add					
	The key name cannot exceed 63 chars. It supports letters, numbers, "/" and "-". "/" cannot be placed at the beginning. A prefix is supp					
	The label key value can only include letters, numbers and separators ("-", "_", "."). It must start and end with letters and numbers.					
OS type	Linux ▼	Φ				
	Configurations are initialized when yo	u ch	nange the OS type for the container.			
Volume (optional)	Add volume					

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 more

**ワークロード名**:作成するワークロードの名前を入力します。ここでは、helloworldを例として使用します。

説明:ワークロードの関連情報を入力します。

ネームスペース:実際のニーズに応じて選択してください。

**タグ**: key = valueキーバリューペア。この例では、タグのデフォルト値は k8s-app = helloworld とします。

**OSタイプ**:実際のニーズに応じて選択します。ここでは、**Linux** を例として使用します。

データボリューム:実際のニーズに応じてワークロードのマウントボリュームを設定します。詳細については、

Volume管理をご参照ください。

- 5. 以下の情報を参照して、「インスタンス内コンテナ」を設定してください。
- 5.1 インスタンス内コンテナ名を入力します。ここでは例としてhelloworld を使用します。
- **5.2 イメージを選択**をクリックし、ポップアップボックスで**マイイメージ**を選択し、検索ボックス機能で helloworldイメージを探して**OK**をクリックします。

主なパラメータ情報は下記の通りです:

イメージバージョン(Tag):デフォルト値latestを使用します。

**イメージプルポリシー: Always、IfNotPresentおよびNever**という**3**つのポリシーを提供しますので、必要に応じて選択してください。ここでは、何も設定していないデフォルトのポリシーを例として使用します。

6. 「インスタンス数」では、以下の情報に基づいてサービスのインスタンス数を設定します。下図に示すとおりです。



Number of instances	Manual adjustment Set the number of Pods di		_	nent
	Number of instances	_	1	+

**手動調節**:インスタンス数を設定します。ここではインスタンス数を1とします。「+」または「-」をクリックすると、インスタンス数を制御できます。

**自動調節**:設定した条件のいずれかが満たされると、インスタンス(pod)を自動調整します。詳細については、サービスの自動スケーリングをご参照ください。

7. 以下の手順に従い、ワークロードへのアクセス設定を行います。

**Service**:「開始」にチェックを入れます。

サービスアクセス方法:「パブリックネットワークLBアクセス」を選択します。

ロードバランサ:実際のニーズに応じて選択します。

**ポートマッピング: TCP**プロトコルを選択し、コンテナポートとサービスポートをどちらも80に設定します。

#### 注意

サービスが配置されているクラスターのセキュリティグループは、ノードネットワークとコンテナネットワーク を開放している必要があり、さらにポート30000~32768も開放する必要があります。開放していないと、TKEを 使用できないといった問題が発生する場合があります。詳細については、TKEセキュリティグループの設定をご参照ください。

8. **Deploymentの作成**をクリックし、Hello Worldサービスの作成を完了します。

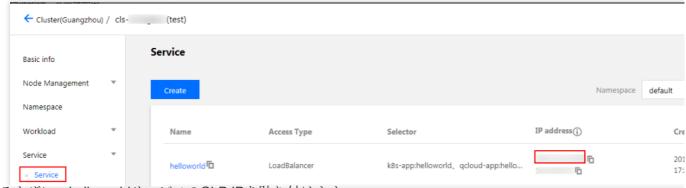
### Hello Worldサービスにアクセス

次の二つの方式でHello Worldサービスにアクセスできます。

#### Cloud Load Balancer IPでHello Worldサービスにアクセス

- 1. TKEコンソールにログインし、左側ナビゲーションバーから**クラスター**を選択します。
- 2. **クラスター管理**ページで、Hello WorldサービスのクラスターIDをクリックし、**クラスター詳細ページ**に進みます。
- 3. Service ページで、サービスとルーティング > Service を選択します。
- 4. サービス管理ページで、helloworldサービスのCloud Load Balancer IPをコピーします。下図に示すとおりです





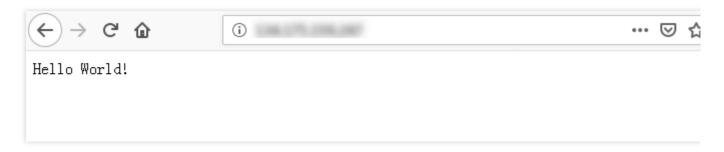
5. ブラウザに、helloworldサービスのCLB IPを貼り付けます。

#### サービスの名前でHello Worldサービスにアクセス

クラスター内のその他サービスやコンテナは、直接サービス名を介してアクセスできます。

### Hello Worldサービス検証

サービスにアクセスする時、次のように表示されれば、Hello Worldサービスが作成されたことになります。



コンテナの作成に失敗した場合は、イベントに関するよくあるご質問をご参照ください。



# 単一インスタンス版のWordPress

最終更新日::2022-04-07 11:10:35

# 操作場面

WordPressは、PHP言語を使用して開発されたブログプラットフォームです。ユーザーは、PHPとMySQLデータベースをサポートするサーバーに、自分のウェブサイトを設置することができます。また、WordPressをコンテンツ管理システム(CMS)として使用することも可能です。

このドキュメントは**Docker Hub**の公式イメージ wordpress で公的にアクセス可能な**WordPress**ウェブサイトを作成する方法を紹介します。

# 前提条件

#### 注意:

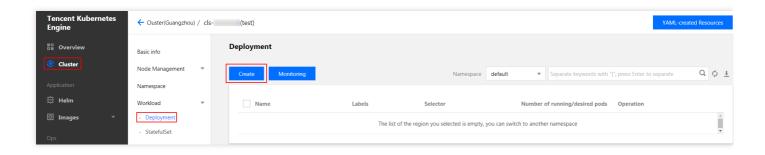
- wordpress このイメージにはWordPressのすべての運行環境が含まれているので、直接プルしてサービスを作成すればいいです。
- 単一インスタンスバージョンのWordPressを作成するのはテスト用のみです。データの持続的な保存は 保証できません。自作のMySQLやTencentDBでデータを保存することをお勧めします。詳細は TencentDBを使用しているWordPressをご参照ください。
- Tencent Cloudアカウント登録済みです。
- クラスターが作成されました。クラスターの作成については、クラスター作成をご参照ください。

# 操作手順

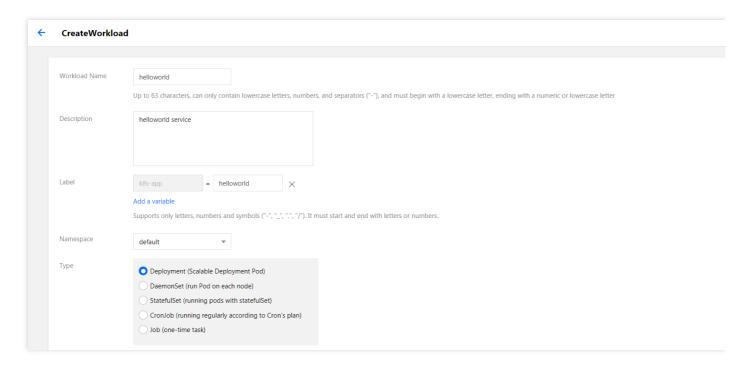
#### WordPressサービス作成

- 1. Tencent Kubernetes Engine コンソールにログインし、左側のナビゲーションバーで\*\* Cluster \*\*を選択します。
- 2. 「クラスター管理」ページでサービスを作成する必要のあるクラスターIDを選択し、クラスターのロード「Deployment」ページに入って、\*\* Create\*\*をクリックします。下図の通りです:





3. 「Create Workload」ページで次の情報によって、ロードの基本情報を設定します。下図の通りです:



- ロード名:作成するロードの名前を入力し、本文はwordpressを例として説明します。
- 説明:ロードについての情報を記入します。
- タグ: key = value、この例ではタグのデフォルト値はk8s-app = wordpressです。
- **ネームスペース**: 実際の必要に応じて選択します。
- **タイプ**:実際の必要に応じて選択します。
- データボリューム:実際の必要に応じてロードのマウントを設定します。詳細はVolume管理をご参照ください。
- 4. 次の提示によって、インスタンスコンテナを設定します。 主なパラメータ情報は下記の通りです。他のオプションはディフォルト値を保ちます:
- **名前**:カスタマイズコンテナの名前を入力し、本文はtestを例として説明します。
- イメージ: wordpress を入力します。
- イメージバージョン (Tag) : latestを入力します。



- **イメージプルポリシー**:次の三つのポリシーを提供し、必要に応じて選択してください。本文は設定しなくて デフォルトポリシーを使用するのを例として説明します。
  - イメージプルポリシーを設定しなくて、イメージバージョンはnullや latest の時、Alwaysポリシーを使用し、そうでなければIfNotPresentポリシーを使用します。
  - Always:いつもリモートでこのイメージをプルします。
  - **IfNotPresent**: ローカルイメージをデフォルトで使用し、ローカルにこのイメージがない場合、リモートでこのイメージをプルします。
  - Never:ローカルイメージだけを使用し、ローカルにこのイメージがない場合、異常を報告します。
- 6. 次の提示によって、サービスするインスタンス数量を設定します。下図の通りです:



- 手動調整:インスタンス数量を設定し、本文のインスタンス数量は1に設定します。「+」や「ー」をクリックして、インスタンス数量をコントロールします。
- **自動調整**:いずれかの設定条件を満たせば、インスタンス(**pod**)数量を自動的に調整します。詳細はサービスの自動的なスケールアウトとスケールインをご参照ください。
- 8. 次の提示によって、ロードの**アクセス設定(Service)**をします。
- **Service**: 「使用」を選択します。
- **サービスのアクセス方式**: 「パブリックネットワークLBでアクセス」を選択します。
- Cloud Load Balancer: 実際の必要に応じて選択します。
- **ポートマッピング**: TCPプロトコルを選択し、コンテナポートとサービスポートを80に設定します。

#### 注意:

サービスのあるクラスターのセキュリティグループはノードネットワークとコンテナネットワークをインターネットにオープンしてください。同時に、30000 - 32768ポートをインターネットにオープンする必要もありますが、そうでなければTencent Kubernetes Engineが使用できない問題が発生する場合があります。詳細はTencent Kubernetes Engineセキュリティグループ設定をご参照ください。

9. Create Workloadをクリックし、wordpressサービスの作成を完了します。

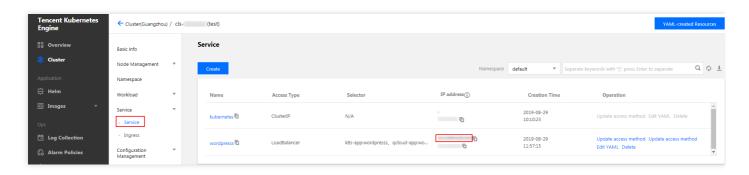


# WordPressサービスにアクセス

次の二つの方式でWordPressサービスにアクセスできます。

#### Cloud Load Balancer IPでWordPressサービスにアクセス

- 1. 左側ナビゲーションバーで\*\* Cluster\*\*をクリックして、「クラスター管理」ページに入ります。
- 2. WordPressサービスのあるクラスターIDをクリックして、Services and Routes > Serviceを選択します。
- 3. サービス管理のページに入って、WordPressサービスのCloud Load Balancer IPをコピーして、次の通りです。



**4.** ブラウザのアドレスバーにCloud Load Balancer IPを入力し、「**Enter**」を押すと、サービスにアクセスできます。

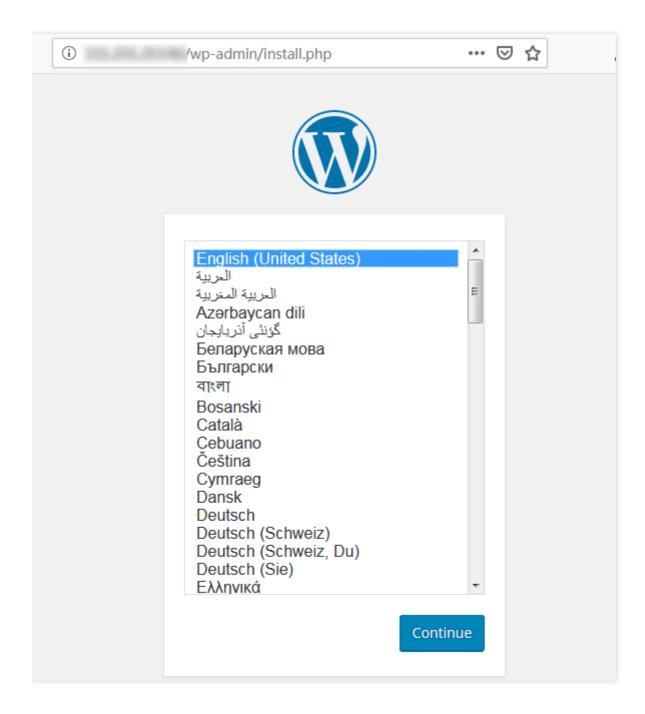
# サービスの名前でサービスにアクセス

クラスター内の他のサービスやコンテナはサービスの名前で直接にアクセスできます。

### WordPressサービス検証

サービスは成功的に作成されました。サービスにアクセスする時、直接にWordPressサーバーの構成ページに入ります。次の通りです。





# その他のWordPress設定

コンテナの作成に失敗した場合、よくある質問をご参照ください。



# TencentDBを用いたWordPress

最終更新日::2022-04-07 11:10:35

# 操作場面

単一インスタンスバージョン WordPress の例は迅速にWordPressサービスを作成する方法を示します。この方法で作成されたWordPressサービスの特徴は次の通りです:

- データは同じコンテナが運行されているMySQL DBに書き込みます。
- サービスは迅速に起動することができます。
- コンテナがある原因で停止すれば、DBとストレージファイルは失います。

MySQL DBでデータを永遠にストレージすることができます。DBはインスタンス/コンテナが再び起動した後、 存在し続けます。このドキュメントはTencentDBでMySQL DBを設定する方法とTencentDBのWordPressサービ スを作成して使用する方法を紹介します。

# 前提条件

- Tencent Cloudアカウント登録済みです。
- クラスターが作成されました。クラスターの作成については、クラスター作成をご参照ください。

#### 説明:

本文が使用されているDBはMySQLです。

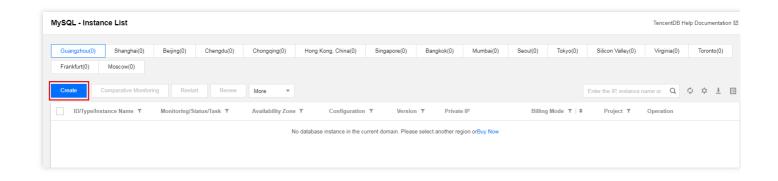
# 操作手順

### WordPressサービス作成

### TencentDB作成

1. MySQLコンソールにログインし、DBインスタンスリストの上で【 Create 】をクリックします。下図の通りです:





2. 購入する構成を選択します。詳細はMySQLをご参照ください。

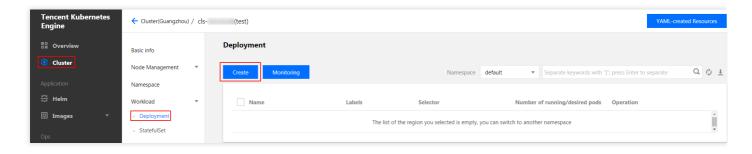
#### 注意:

TencentDBのある地域はクラスターと同じです。そうでなければこのDBに接続できません。

- 3. DBが成功的に作成された後、MySQL-インスタンスリストで確認できます。
- 4. DBについて初期化操作を行います。詳細はMySQLの初期化をご参照ください。

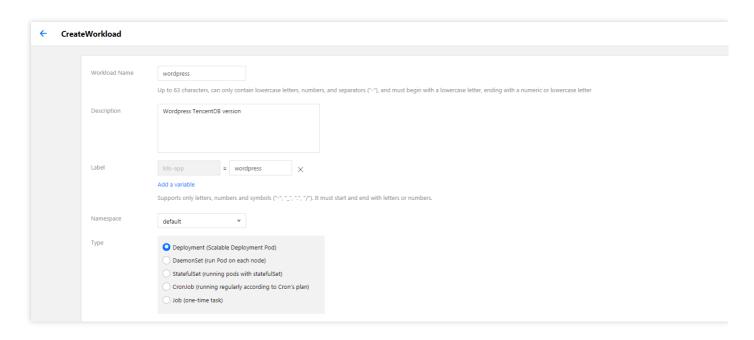
#### TencentDBを使用するWordPressサービス作成

- 1. Tencent Cloud Tencent Kubernetes Engineコンソールにログインし、左側のナビゲーションバーで【 Cluster】を選択します。
- 2. 「クラスター管理」ページでサービスを作成する必要のあるクラスターIDを選択し、クラスターのロード 「Deployment」ページに入って、【Create】をクリックします。下図の通りです:





3. 「Workloadの新規作成」ページで次の情報によって、ロードの基本情報を設定します。下図の通りです:



- ロード名:作成するロードの名前です。本文はwordpressを例として説明します。
- 説明:ロードについての情報を記入します。
- **タグ**: key = value、この例ではタグのデフォルト値がk8s-app = wordpressです。
- **ネームスペース**:必要に応じて選択します。
- **タイプ**:必要に応じて選択します。
- ボリューム:必要に応じてロードのマウントを設定します。詳細はVolume管理をご参照ください。



4. 次の情報を参照して「インスタンスコンテナ」を設定します。次の通りです。

Containers in the pod												
												✓ X
	Name	test										
		Up to 63 c	haracters. It	supports lo	wer case le	tters, num	ber, and hyp	hen ("-") a	ind ca	nnot sta	art or end	l with ("-")
	Image	wordpre	SS		Select an	image						
	Image Tag	latest										
	CPU/memory limit	CPU Limit							Mei	mory Li	mit	
		request	0.25	- limit	0.5	-core	request	256	-	limit	1024	MiB
												umber of resources, the container will fail to be created. ion of node resources.
	GPU limit	-	0 +									
	Environment Variable	WORDPI	RESS_DB_H	: =			/ ×					
	<b>(i)</b>	WORDPI	RESS_DB_PA	=			. ×					
		Add a varia	able Referer	nce ConfigN	Map/Secret							
		Supports o	only letters,	numbers an	d symbols (	("-", " <u>-</u> ", ";	"). It must sta	rt and end	l with	letters (	or numbe	rs.
	Advanced Settings											
1	Note: After Workload is crea	ted, the cont	ainer config	guration info	ormation ca	n be mod	ified by upda	iting YAML	L.			
							Add Con	tainer				

主なパラメータ情報は下記の通りです。他のオプションはディフォルト設定を保ちます:

- **名前**:カスタマイズコンテナの名前を入力し、本文はtestを例として説明します。
- イメージ: wordpress を入力します。
- イメージバージョン (Tag): latestを入力します。
- **イメージプルポリシー**:次の三つのポリシーを提供し、必要に応じて選択します。本文は設定しなくてデフォルトポリシーを使用するのを例として説明します。

イメージプルポリシーを設定しなくて、イメージバージョンはnullやlatestの時、Alwaysポリシーを使用し、そうでなければIfNotPresentポリシーを使用します。

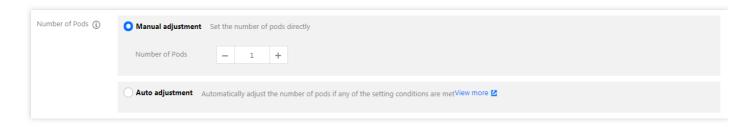
- Always:いつもリモートでこのイメージをプルします。
- **IfNotPresent**: ローカルイメージをデフォルトで使用し、ローカルにこのイメージがない場合、リモートでこのイメージをプルします。
- Never:ローカルイメージだけを使用し、ローカルにこのイメージがない場合、異常を報告します。
- 環境変数:次の構成情報を順番に入力します:

WORDPRESS\_DB\_HOST = MySQLのプライベートネットワークIP

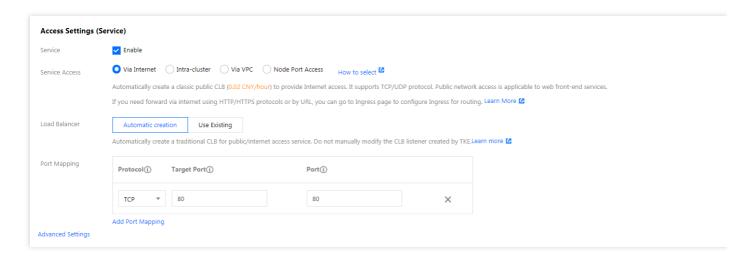
WORDPRESS\_DB\_PASSWORD = 初期化の時、記入されたパスワード



5. 次の提示によって、サービスするインスタンス数量を設定します。下図の通りです:



- 手動調整:インスタンス数量を設定し、本文のインスタンス数量は1に設定します。「+」や「ー」をクリックして、インスタンス数量をコントロールします。
- **自動調整**:いずれかの設定条件を満たせば、インスタンス(pod)数量を自動的に調整します。詳細はサービスの自動的なスケールアウトとスケールインをご参照ください。
- 6. 次の提示によって、ロードのアクセス設定をします。下図の通りです:



- **Service**:「使用」を選択します。
- **サービスのアクセス方式**: 「パブリックネットワークを提供してアクセス」を選択します。
- Cloud Load Balancer: 必要に応じて選択します。
- **ポートマッピング**: TCPプロトコルを選択し、コンテナポートとサービスポートを80に設定します。

#### 注意:

サービスのあるクラスターのセキュリティグループはノードネットワークとコンテナネットワークをインターネットにオープンしてください。同時に、30000 - 32768ポートをインターネットにオープンする必要もありますが、そうでなければTencent Kubernetes Engineが使用できない問題が発生する場合があります。詳細はTencent Kubernetes Engineセキュリティグループ設定をご参照ください。



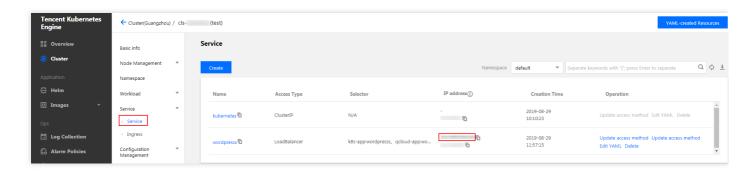
7. 【Create workload】をクリックして、WordPressサービスの作成を完了します。

### WordPressサービスにアクセス

次の二つの方式でWordPressサービスにアクセスできます。

#### Cloud Load Balancer IPでWordPressサービスにアクセス

- 1. 左側ナビゲーションバーで【Clusters】をクリックして、「クラスター管理」ページに入ります。
- 2. WordPressサービスのあるクラスターIDをクリックして、【Service】>【Service】を選択します。
- 3. サービス管理のページに入って、WordPressサービスのCloud Load Balancer IPをコピーして、下図の通りです:



4. ブラウザのアドレスバーにCloud Load Balancer IPを入力し、Enterを押すと、サービスにアクセスできます。

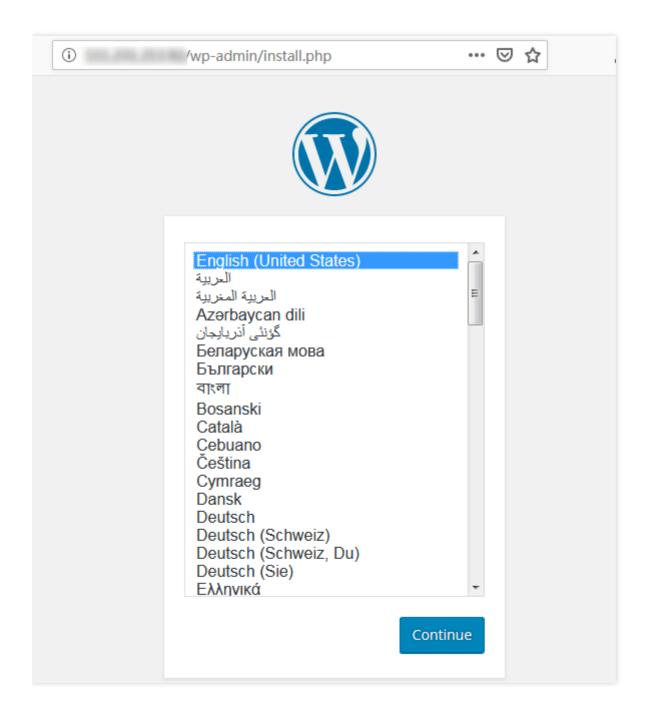
#### サービスの名前でサービスにアクセス

クラスター内の他のサービスやコンテナはサービスの名前で直接にアクセスできます。

#### WordPressサービス検証

サービスは成功的に作成されました。サービスにアクセスする時、直接にWordPressサーバーの構成ページに入ります。下図の通りです:





# その他のWordPress設定

コンテナの作成に失敗した場合、よくある質問をご参照ください。



# 簡易Webアプリケーションの構築

最終更新日::2022-12-14 16:43:12

# 運用シーン

ここでは、Tencent CloudのTKEを使用して簡単なWebアプリケーションを作成する方法をご紹介します。

Webアプリケーションは以下の2部分から構成されます:

- フロントエンドサービス。クライアント側の検索や書き込みリクエストの処理に使用されます。
- データベースストレージサービス。redisを使用します。書き込んだデータをredis-masterに保存し、redis-slave にアクセスして読み取る処理を行います。redis-master と redis-slave はマスター/スレーブレプリケーションに よりデータを同期します。

このアプリケーションはkubernetes固有のサンプルです。詳しくは、Guestbook Appをご参照ください。

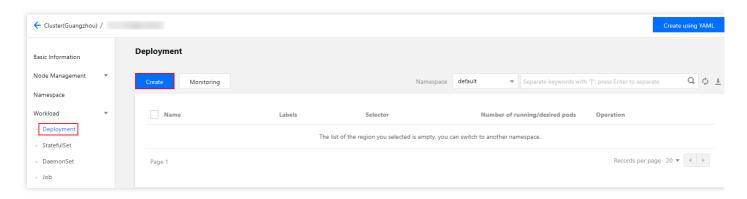
# 前提条件

- Tencent Cloudアカウントの登録が完了していること。
- クラスタが作成されていること。作成されていない場合、 クラスター作成をご参照ください。

# 操作手順

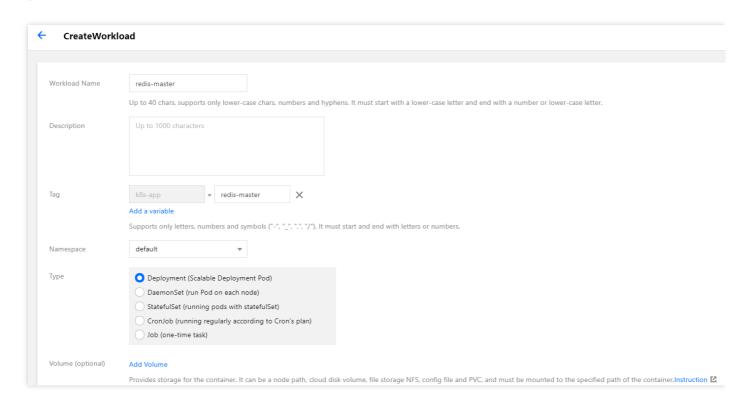
# redis-masterサービスの作成

- 1. TKEコンソールにログインし、左側のナビゲーションバーの Cluster を選択します。
- 2. アプリケーションを作成するクラスターIDをクリックし、ワークロードDeployment画面に入って、\*\* Create\*\* を選択します。下図の通りです:





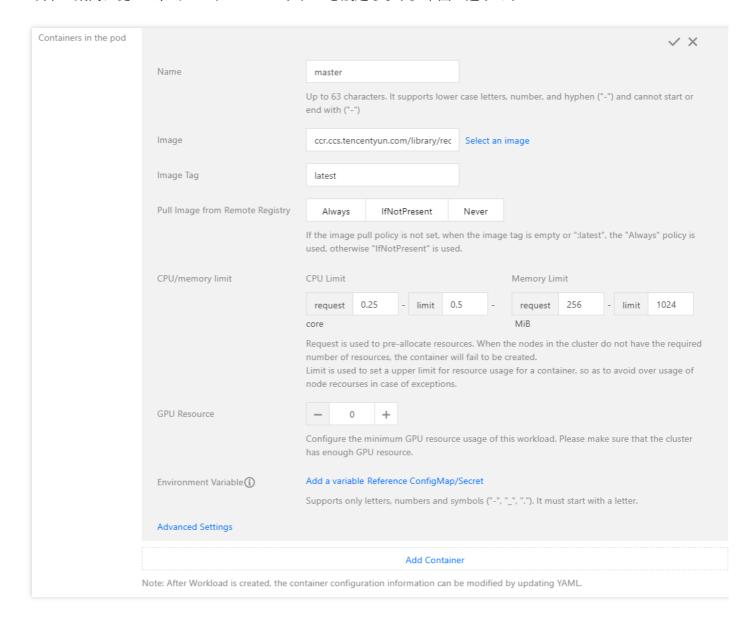
3. [Create Workload] 画面で、以下の案内に従って、ワークロードの基本情報を設定します。下図の通りです:



- ワークロード名:作成するワークロードの名前。本書では、redis-masterとします。
- 詳細:ワークロードの関連情報を記入します。
- **タグ**: key = value キーと値のペア。本例では、タグのデフォルト値は、 k8s-app = redis-masterとします。
- **ネームスペース**:必要に応じて選択してください。
- **タイプ**:必要に応じて選択してください。
- **データボリューム**:必要に応じて、ワークロードのマウントボリュームを設定してください。詳しくは、 Volume 管理をご参照ください。



4. 以下の案内に従って、インスタンスコンテナーを設定します。下図の通りです:



主なパラメータ情報は下記の通りです。ほかのオプションはデフォルトで構いません。

- 名前:インスタンスコンテナーの名前を入力します。本書では、masterとします。
- **イメージ**: 「ccr.ccs.tencentyun.com/library/redis」を入力します。
- イメージのバージョン (Tag): latestを入力します。
- イメージ取得ポリシー:以下の3ポリシーが用意されているため、必要に応じて選択してください。本書では、 デフォルトポリシーの適用を例とします。

イメージ取得ポリシーを設定しない際、イメージのバージョンが空白またはlatestの場合、Alwaysポリシーを適用します。そうでない場合、IfNotPresentポリシーを適用します。

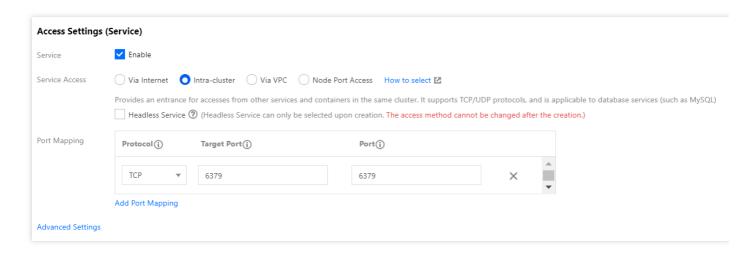
- Always:常にリモートからイメージを取得します。
- **IfNotPresent**: デフォルトではローカルイメージを使用します。ローカルイメージがない場合、リモートからイメージを取得します。



- Never:ローカルイメージだけを使用します。ローカルイメージがない場合、エラーを出力します。
- 6. インスタンス数を設定します。詳しくは下図の通りです:

Number of Pods	Manual adjustment Auto adjustment Set the number of pods directly
	Number of Pods — 1 +
Advanced Settings	

- **手動調整**:インスタンス数を設定します。本書では、インスタンス数に1を設定します。「+」または「-」をクリックしてインスタンス数を設定することができます。
- **自動調整**:いずれかの設定条件を満たした場合、自動的にインスタンス(**pod**)数を調整します。詳しくは、サービスの自動拡張/縮小をご参照ください。
- 7. 以下の案内に従って、ワークロードのアクセス設定を実施します。下図の通りです:



- Service:「有効」にチェックを入れます。
- **サービスアクセス方式**:「クラスター内部のアクセス」を選択します。
- **ロードバランサー**:必要に応じて選択してください。
- **ポートマッピング**: TCPプロトコルを選択し、サービスポートとコンテナーポート両方に**6379**を設定します。 ほかのサービスは、サービス名redis-masterまたはポート**6379**を介してmasterコンテナーにアクセスすることができます。
- 8. **Create Workload**をクリックし、redis-masterサービスの作成を完了します。

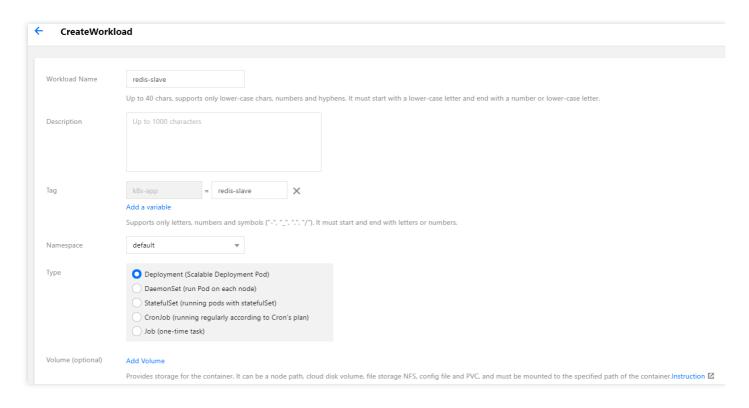
#### redis-slaveサービスの作成



- 1. TKEコンソールにログインし、左側のナビゲーションバーの \*\* Cluster \*\*を選択します。
- 2. サービスを作成するクラスターIDをクリックし、ワークロードDeployment画面に入って、**Create**を選択します。下図の通りです:



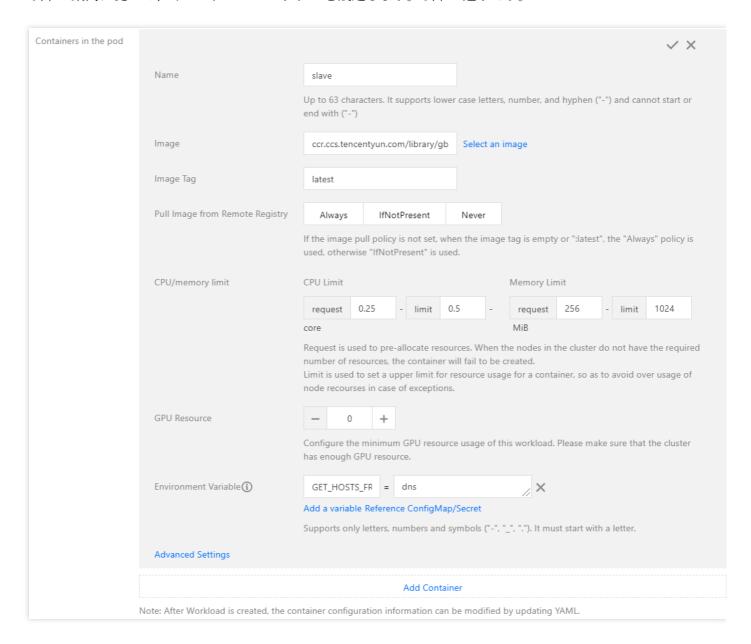
3. [New Workload] 画面で、以下の案内に従って、ワークロードの基本情報を設定します。下図の通りです:



- ワークロード名:作成するワークロードの名前。本書では、redis-slaveとします。
- 詳細:ワークロードの関連情報を記入します。
- **タグ**: key = value キーと値のペア。本例では、タグのデフォルト値は、 k8s-app = redis-slave とします。
- **ネームスペース**:必要に応じて選択してください。
- **タイプ**:必要に応じて選択してください。
- **データボリューム**:必要に応じて、ワークロードのマウントボリュームを設定してください。詳しくは、 Volume 管理をご参照ください。



4. 以下の案内に従って、インスタンスコンテナーを設定します。以下の通りです。

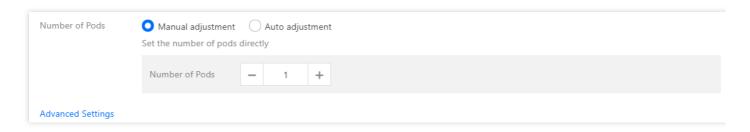


主なパラメータ情報は下記の通りです。ほかのオプションはデフォルトでよいです。

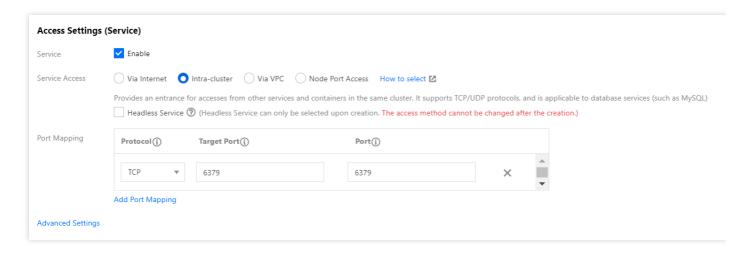
- 名前:インスタンスコンテナーの名前を入力します。本書では、slaveとします。
- イメージ: 「ccr.ccs.tencentyun.com/library/gb-redisslave」を入力します。
- イメージのバージョン (Tag): latestを入力します。
- イメージ取得ポリシー:必要に応じて選択してください。本書では、デフォルトポリシーの適用を例とします。
- 環境変数:以下の設定情報を入力します: GET HOSTS FROM = dns



5. インスタンス数を設定します。下図の通りです:



- **手動調整**:インスタンス数を設定します。本書では、インスタンス数に1を設定します。「+」または「-」をクリックしてインスタンス数を設定することができます。
- **自動調整**:いずれかの設定条件を満たした場合、自動的にインスタンス(**pod**)数を調整します。詳しくは、サービスの自動拡張/縮小をご参照ください。
- 6. 以下の案内に従って、ワークロードのアクセス設定を実施します。下図の通りです:



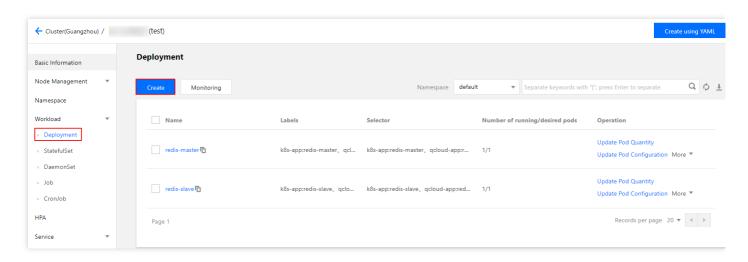
- Service:「有効」にチェックを入れます。
- **サービスアクセス方式**:「クラスター内部のアクセス」を選択します。
- **ロードバランサー**:必要に応じて選択してください。
- **ポートマッピング**: TCPプロトコルを選択し、サービスポートとコンテナーポート両方に**6379**を設定します。 ほかのサービスは、サービス名redis-masterまたはポート**6379**を介してmasterコンテナーにアクセスすることができます。
- 7. **Create Workload**をクリックし、redis-slaveサービスの作成を完了します。

### frontendサービスの作成

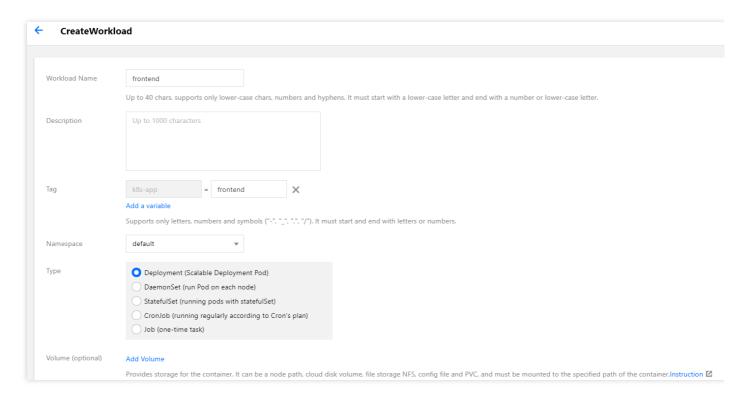
1. TKEコンソールにログインし、左側のナビゲーションバーの Cluster を選択します。



2. アプリケーションを作成するクラスターIDをクリックし、ワークロードDeployment画面に入って、**Create**を選択します。下図の通りです:



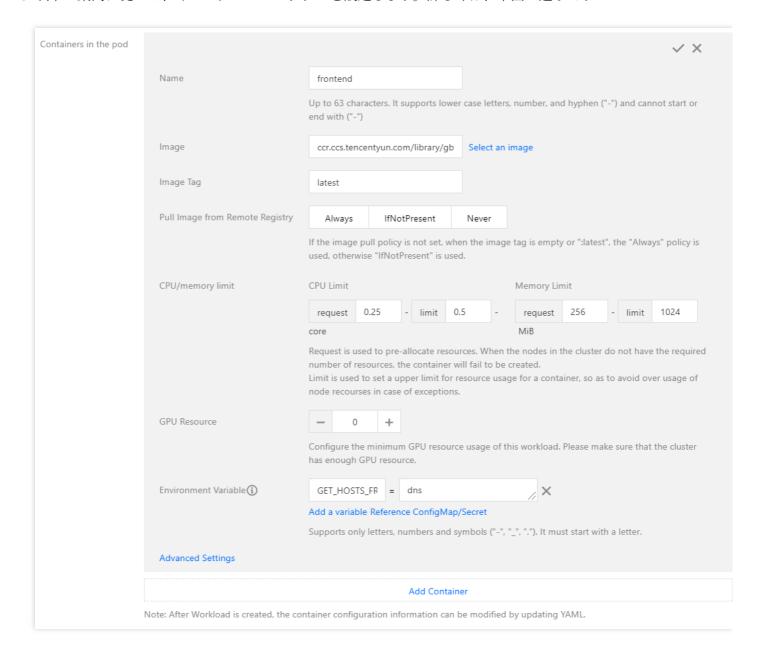
3. [New Workloa] 画面で、以下の案内に従って、ワークロードの基本情報を設定します。下図の通りです:



- ワークロード名:作成するワークロードの名前。本書では、frontendとします。
- 詳細:ワークロードの関連情報を記入します。
- **タグ**: key = value キーと値のペア。本例では、タグのデフォルト値は、 k8s-app = frontendとします。
- **ネームスペース**:必要に応じて選択してください。
- **タイプ**:必要に応じて選択してください。
- **データボリューム**:必要に応じて、ワークロードのマウントボリュームを設定してください。詳しくは、 Volume 管理をご参照ください。



### 5. 以下の案内に従って、インスタンスコンテナーを設定します。詳しくは、下図の通りです:

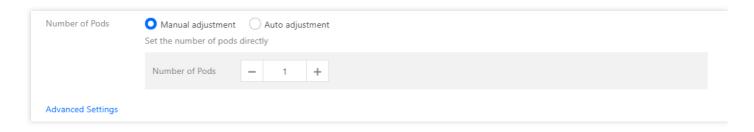


主なパラメータ情報は下記の通りです。ほかのオプションはデフォルトで構いません:

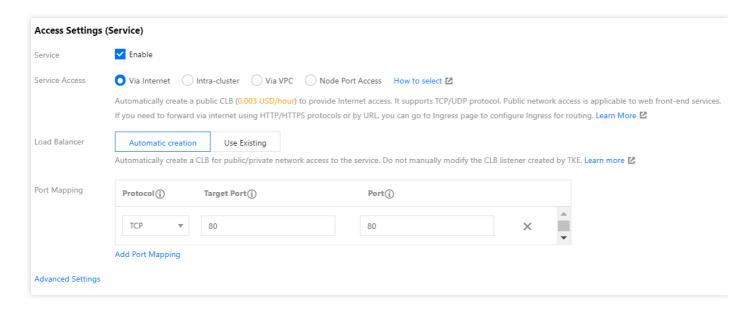
- **名前**:インスタンスコンテナーの名前を入力します。本書では、frontendとします。
- **イメージ**:「ccr.ccs.tencentyun.com/library/gb-frontend」を入力します。
- イメージのバージョン (Tag): latestを入力します。
- **イメージ取得ポリシー**:必要に応じて選択してください。本書では、デフォルトポリシーの適用を例とします。
- 環境変数:以下の設定情報を入力します: GET\_HOSTS\_FROM = dns



6. インスタンス数を設定します。詳しくは下図の通りです:



- **手動調整**:インスタンス数を設定します。本書では、インスタンス数に1を設定します。「+」または「-」をクリックしてインスタンス数を設定することができます。
- **自動調整**:いずれかの設定条件を満たした場合、自動的にインスタンス(**pod**)数を調整します。詳しくは、サービスの自動拡張/縮小をご参照ください。
- 7. 以下の案内に従って、ワークロードのアクセス設定を実施します。下図の通りです:



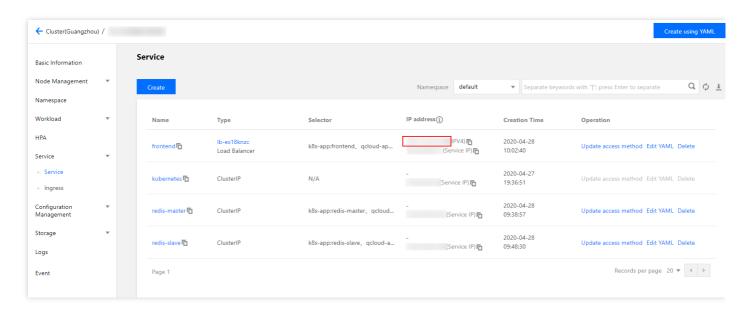
- Service:「有効」にチェックを入れます。
- サービスアクセス方式:「パブリックネットワークからのアクセスを許可」を選択します。
- **ロードバランサー**:必要に応じて選択してください。
- **ポートマッピング**: TCPプロトコルを選択し、サービスポートとコンテナーポート両方に80を設定します。 ユーザーはブラウザ経由でロードバランスIPにアクセスすることで、frontendコンテナにアクセスできます。
- 8. **Create Workload** をクリックし、frontendサービスの作成を完了します。

#### Webアプリケーションの確認

1. 左側ナビゲーションバーで、Cluster をクリックし、「クラスター管理」画面に入ります。

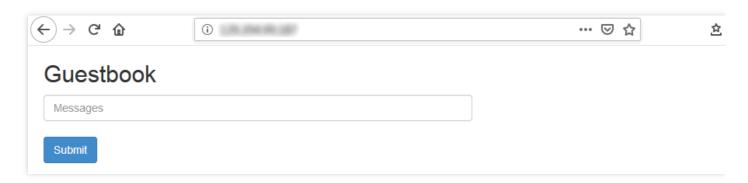


- 2. 作成したサービスが所在するクラスターIDをクリックして、Services>Serviceの順に選択します。下図の通りです
- 3. サービス管理画面に入って、frontendサービスのロードバランスIPをコピーします。下図の通りです:



#### 説明:

- redis-master と redis-slave サービスを作成する時、クラスタ内部でのアクセスのアクセス方式を設定 したため、サービスは1つの内部ネットワークIPだけがあり、クラスター内部のサービスだけがアクセスできます。
- frontendサービスを作成する時、パブリックネットワークからのアクセスを許可のアクセス方式を設定したため、サービスはロードバランスIP(すなわち、パブリックネットワークIP)と内部ネットワークIPがあるため、クラスター内部のサービスからのアクセスも可能で、パブリックネットワーク経由でもアクセスできます。
- 4. ブラウザ経由でfrontendサービスのロードバランスIPにアクセスし、下図に示している画面が返されれば、正常にfrontendサービスにアクセスできます。





5. 入力欄に任意の文字列を入力して **Submit** をクリックすると、入力された内容が保存され、画面の下側に表示されます。

ブラウザの画面をリフレッシュして、改めてこのサービスのIPアドレスにアクセスします。入力された内容が 依然として存在すれば、入力された文字列がすでにredisに保存されています。

### 開発例

以下のサンプルコードは、Guestbook Appのfrontendサービスの完全なコードです。frontendサービスがHTTPリクエストを受信すると、setコマンドかを判断します:

- setコマンドと判断した場合、パラメータ中のkeyとvalueを取得して、redis-masterサービスに接続し、keyと valueをredis-masterに設定します。
- setコマンドではないと判断した場合、redis-slaveサービスに接続し、パラメータkeyに対応するvalueの値を取得し、クライアントに返して表示します。

```
<?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: 443,
]);
$client->set($_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: 443,
]);
```



```
$value = $client->get($_GET['key']);
print('{"data": "' . $value . '"}');
}
}else{
phpinfo();
} ?>
```

#### 説明事項

• frontendサービスがredis-masterとredis-slaveサービスにアクセスする時、**サービス名とポート**に接続します。 クラスター固有のdnsサービスはサービス名を対応するサービスIPに解析し、サービスIPでロードバランシング を実行します。

例えば、redis-slaveサービスに3つのインスタンスがあるとします。redis-slaveサービスにアクセスする時、そのままredis-slaveと6379に接続します。dnsは自動的にredis-slaveをredis-slaveのサービスIP(すなわち、ロードバランス IPに類似するフローティングIP)に解析し、redis-slaveのサービスIPで自動的にロードバランシングを実行し、リクエストをあるredis-slaveサービスのインスタンスに送信します。

- コンテナー環境変数の設定:
- **デフォルト設定の適用(推奨)**: frontendコンテナーが動作時に、取得した**GET\_HOSTS\_FROM**環境変数の値がdnsの場合、サービス名で接続します。
- その他の設定:ほかの環境変数でredis-masterまたはredis-slaveのドメイン名を取得してください。