

# 容器服务

## 集群

### 产品文档



腾讯云

**【版权声明】**

©2013-2018 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模式的承诺或保证。

## 文档目录

### 集群

集群概述

集群的基本操作

添加已有云主机

集群的自动扩缩容

容器服务支持 GPU 调度

集群的生命周期

Namespace使用指引

节点的使用指引

节点的驱逐和封锁

升级集群

集群启用 IPVS

设置同地域集群间互通

设置跨地域集群间互通

设置容器集群与 IDC 互通

设置云主机集群与黑石集群互通

使用 Network Policy 进行网络访问控制

# 集群

## 集群概述

最近更新时间：2018-06-21 14:41:28

集群是指容器运行所需云资源的集合，包含了若干台云服务器、负载均衡器等腾讯云资源。

## 集群信息

**集群类型**：目前支持VPC容器集群。

**集群配置**：可在创建集群时自行设置，包括云服务器的机型、操作系统、系统盘和数据盘大小、登录密码等。

**集群组成**：当前支持包年包月云服务器、按量计费云服务器。

## 管理集群

集群支持创建集群、扩缩节点、删除集群、通过kubernetes API 直接操作集群等。

## 集群预留资源说明

集群中的每个节点都会保留部分资源以支持kubernetes运行，具体预留规则如下：

### CPU

节点总量/单位：核	1	2	4	8	16	32
节点预留量/单位：核	0.06	0.07	0.08	0.09	0.11	0.14

### Memory

节点总量/单位：Gib	1	2	4	8	12	16	24	32	48
节点预留量/单位：Mib	160	320	420	830	1200	1300	1660	1830	2420

另外每个节点还会额外保留100Mib内存以防系统OOM。

注：如果集群开启了日志收集功能，每个节点还会占用约0.3核和250Mib的资源来运行日志收集插件，该消耗会直接算入用户的已分配资源。

## 使用帮助

- [集群的基本操作](#)
- [集群的生命周期](#)
- [集群配额限制](#)
- [集群节点及容器网络设置](#)
- [集群节点硬盘设置](#)
- [集群节点公网IP设置](#)
- [集群节点安全组设置](#)

# 集群的基本操作

最近更新时间：2018-10-09 18:02:03

## 创建集群

1. 登录 [腾讯云容器服务控制台](#)。
2. 单击左侧导航栏中的 **集群**，单击集群列表页的【新建】。



3. 设置集群的基本信息。

- **集群名称**：要创建的集群的名称。不超过60个字符。
- **计费模式**：提供包年包月和按量计费两种计费模式，详细对比请查看 [计费模式说明](#)。
- **所在地域**：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。
- **可用区**：同地域内，内网互通；不同地域，内网不通。需要多个内网通信的用户须选择相同的地域。
- **节点网络**：为集群内主机分配在节点网络地址范围内的 IP 地址。参阅 [容器及节点网络设置](#)。
- **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。参阅 [容器及节点网络设置](#)。

- **集群描述**：创建集群的相关信息。该信息将显示在 **集群信息** 页面。

### 创建集群

1 集群信息 > 2 选择机型 > 3 云主机配置 > 4 信息确认

当您使用容器服务时，需要先创建集群，容器服务运行在集群中。一个集群由若干节点（云服务器）构成，可运行多个容器服务。集群的更多说明参考 [集群概述](#)

集群名称

新增资源所属项目

集群内新增的云主机、负载均衡等资源将会自动分配到该项目下。 [使用指引](#)

节点设置  新增节点  空集群

Kubernetes版本

所在地域  广州  上海  北京  成都  香港  新加坡  孟买  硅谷  弗吉尼亚  莫斯科  法兰克福

处在不同地域的云产品内网不通，购买后不能更换。建议选择靠近您客户的地域，以降低访问延时、提高下载速度。

集群网络  CIDR: 10.0.0.0/16

如现有的网络不合适，您可以去控制台 [新建私有网络](#)

#### 4. 选择机型 (支持系统盘为云盘的所有机型)。

- **系列**：提供 **系列 1** 和 **系列 2**。详细对比参看 [实例类型概述](#)。

- **机型**：机型选择方案参看 [确定云服务器配置方案](#)。

### 创建集群

集群信息 > **2 选择机型** > 3 云主机配置 > 4 信息确认

**已选配置**

集群名	实例集群
Kubernetes版本	1.10.5
所在地域	华南地区(广州)
容器网络	172.16.0.0/16

---

计费模式 ⓘ  按量计费  包年包月 [详细对比](#)

所在地域 华南地区(广州)

可用区 ⓘ  广州一区  广州二区  广州三区  广州四区

节点网络 ⓘ TomVPC 此私有网络在该可用区无有效子网, 现在新建

CIDR: 10.0.0.0/16

如现有的网络不合适, 您可以去控制台[新建私有网络](#) 或 [新建子网](#)

实例族  全部实例族  标准型  高IO型  内存型  计算型

实例类型  全部实例类型  标准型S1  高IO型I1  标准型S2  高IO型I2  内存型M2  计算型C2

## 5. 填写云主机配置。

- **系统盘**：固定为 50G。
- **数据盘**：步长 10G，最高为 4000G。
- **公网宽带**：提供两种计费模式，详细对比参看 [购买网络带宽](#)。
- **带宽**：勾选 **免费分配公网 IP**，系统将免费分配公网 IP，若不需要，请选择带宽值为 0。
- **登录方式**：提供三种对应登录方式。
  - 设置密码**：请根据提示设置对应密码。
  - 立即关联密钥**：密钥对是通过一种算法生成的一对参数，是一种比常规密码更安全的登录云服务器的方式。详细参阅 [SSH 密钥](#)。
  - 自动生成密码**：自动生成的密码将通过站内信发送给您。
- **安全组**：安全组具有防火墙的功能，用于设置云主机 CVM 的网络访问控制。参阅 [容器服务安全组设置](#)。

- 云主机数量：选择服务器数量。

←
创建集群

✓ 集群信息
>
✓ 选择机型
>
3 云主机配置
>
4 信息确认

**已选配置**

集群名	实例集群
Kubernetes版本	1.10.5
所在地域	华南地区(广州)
容器网络	172.16.0.0/16
计费模式	按量计费
机型	标准型S2
规格	S2.SMALL1 (1核1GB)

---

操作系统 Ubuntu Server 16.04.1 LTS 64位

系统盘 本地硬盘 普通云硬盘 SSD云硬盘

本地硬盘固定为50GB，系统盘不支持更换介质，使用本地硬盘的服务器暂不支持升级CPU/内存/硬盘

数据盘 暂不购买 立即购买

数据盘类型 本地硬盘

0GB
|
100GB
|
250GB
|
500GB
-
0
+
GB

公网宽带 i 按带宽计费 按使用流量 [详细对比](#)

6. 创建完成的集群将出现在集群列表中。

集群 广州(2) 上海(0) 北京(0) 成都(0) 香港(0) 新加坡(0) 孟买(0) 硅谷(0) 弗吉尼亚(0) 莫斯科(0) 法兰克福(0)

[新建](#)

ID/名称	监控	集群状态	kubernetes版本	节点状态	节点数量	已分配/总CPU
cls-r3gaw0zg 实例集群		创建中	1.10.5	-	0台	-/0
cls-h9bgbkzi vxjun-custer		运行中	1.7.8	全部正常	1台	0.56/0.94

## 添加云主机

1. 在集群列表页中，单击右侧 **新建节点**。

[新建](#)

ID/名称	监控	集群状态	kubernetes版本	节点状态	节点数量	操作
cls-r3gaw0zg 实例集群		运行中	1.10.5	-	1台	<a href="#">新建节点</a> <a href="#">添加已有节点</a> <a href="#">更多</a>
cls-h9bgbkzi vxjun-custer		运行中	1.7.8	全部正常	1台	<a href="#">新建节点</a> <a href="#">添加已有节点</a> <a href="#">更多</a>

2. 设置添加云主机的所属 **网络**、**机型** 和 **配置信息**。

允许将主机创建在同一地域下不同可用区下的不同子网中。

## ← 新建节点

1 集群信息 > 2 选择机型 > 3 云主机配置 > 4 信息确认

当您使用容器服务时，需要先创建集群，容器服务运行在集群中。一个集群由若干节点（云服务器）构成，可运行多个容器服务。集群的更多说明参考 [集群概述](#)

集群名称 实例集群

新增资源所属项目 ⓘ 默认项目

Kubernetes版本 1.10.5

所在地域 华南地区(广州)

集群网络

如现有的网络不合适，您可以去控制台 [新建私有网络](#)

集群描述

3. 新添加的云主机将出现在 **ID/节点名** 列表中。

← cls-r3gaw0zg (实例集群)

节点列表    Namespace列表    伸缩组列表    集群信息

新建节点    添加已有节点    移出    封锁    取消封锁

<input type="checkbox"/>	ID/节点名 ↕	状态	kubernetes版本	主机类型	配置
<input type="checkbox"/>	ins-5y60rmas ccs_cls-r3gaw0zg_...	健康	1.10.5	标准型S2	1核, 1GB, 1Mbps 系统盘: 50GB 本地硬盘
<input checked="" type="checkbox"/>	ins-d5vhvia2	创建中 ⓘ	1.10.5	标准型S2	1核, 1GB, 1Mbps 系统盘: 50GB 本地硬盘

## 销毁云主机

1. 在集群列表页中单击某集群的 ID/名称，进入如下界面，选择需销毁的云主机，单击右侧 **移出**。



2. 弹出提示页面，显示要移出的节点信息，单击【确定】删除节点。



## 查看节点信息

1. 在集群列表中集群的 **ID/名称**。
2. 单击【节点列表】来查看集群节点列表信息。

← cls-r3gaw0zg (实例集群)

节点列表    Namespace列表    伸缩组列表    集群信息

新建节点    添加已有节点    移出    封锁    取消封锁

<input type="checkbox"/>	ID/节点名 ↕	状态	kubernetes版本	主机类型	配置
<input type="checkbox"/>	ins-5y60rmas ccs_cls-r3gaw0zg_...	健康	1.10.5	标准型S2	1核, 1GB, 1Mbps 系统盘: 50GB 本地硬盘

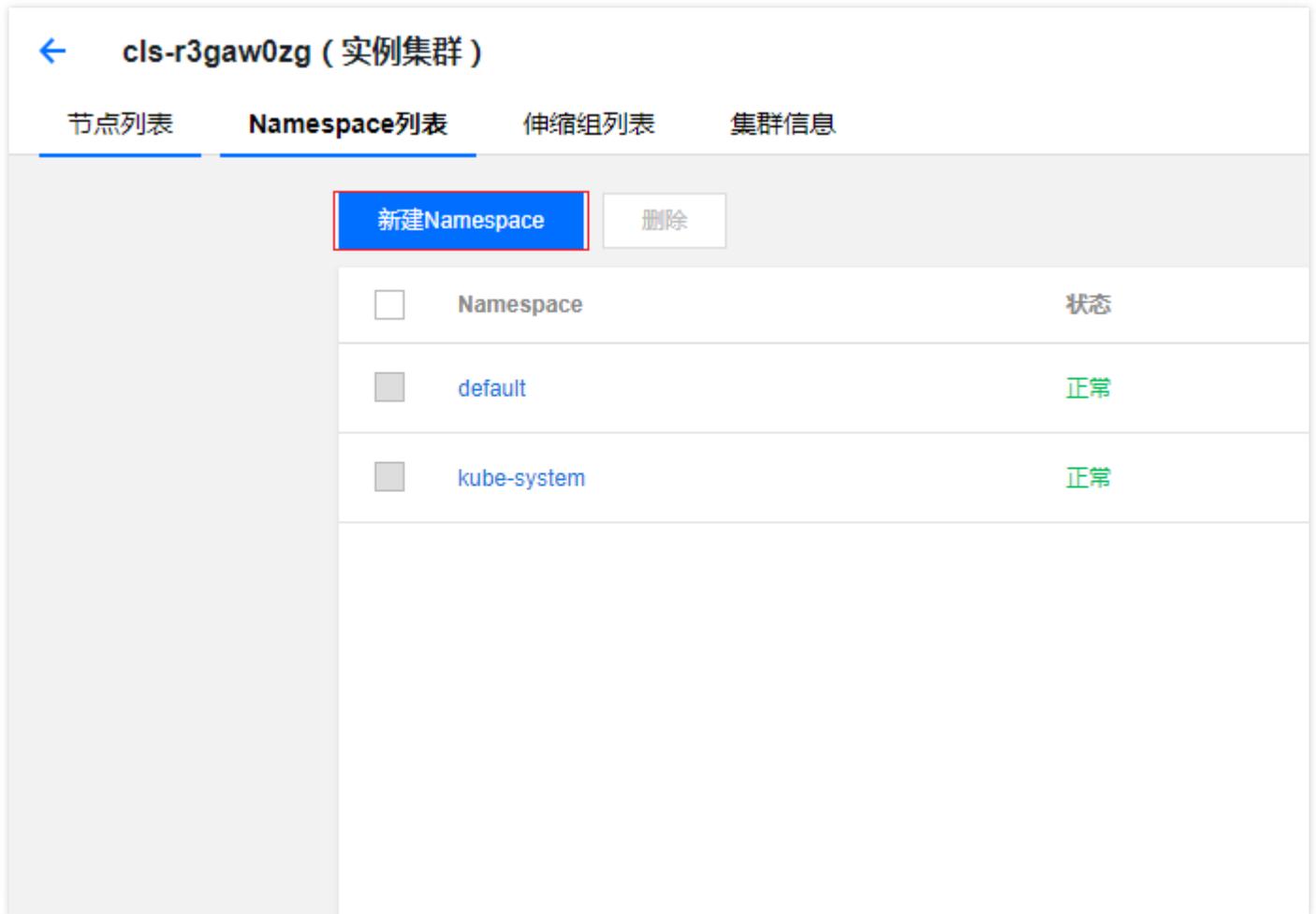
## 登录到节点

当前节点支持腾讯云云主机，如何登录请查看 [登录到云主机](#)。

## 创建集群 Namespace

1. 在集群列表页中选择某集群的 **ID/名称**。

2. 单击 **Namespace 列表**，单击【新建 Namespace】。



The screenshot shows the 'Namespace 列表' (Namespace List) page for a cluster named 'cls-r3gaw0zg (实例集群)'. The page has four tabs: '节点列表' (Node List), 'Namespace列表' (Namespace List), '伸缩组列表' (Scaling Group List), and '集群信息' (Cluster Information). The 'Namespace列表' tab is active. At the top of the list area, there are two buttons: '新建Namespace' (Create Namespace) and '删除' (Delete). The '新建Namespace' button is highlighted with a red border. Below the buttons is a table with the following data:

<input type="checkbox"/>	Namespace	状态
<input type="checkbox"/>	default	正常
<input type="checkbox"/>	kube-system	正常

3. 填写信息并单击【提交】。

### 新建Namespace ✕

名称  [Namespace使用指引](#) 

最长63个字符，只能包含小写字母、数字及分隔符("-")，且不能以分隔符开头或结尾

描述

## 删除集群 Namespace

1. 在集群列表页中选择某集群的 **ID/名称**。

2. 单击 **Namespace 列表**，选择需删除的 Namespace，单击右侧【删除】。

新建Namespace		删除		请输入Namespace名称 <input type="text"/> <input type="button" value="Q"/> <input type="button" value="↓"/>		
<input type="checkbox"/>	Namespace	状态	描述	创建时间	操作	
<input type="checkbox"/>	default	正常		2018-10-09 11:58:20	删除	
<input type="checkbox"/>	kube-system	正常		2018-10-09 11:58:20	删除	
<input type="checkbox"/>	xxxx	正常		2018-10-09 12:09:54	<b>删除</b>	

共3项 每页显示行 20   1/1

3. 弹出提示页面，显示要删除的 Namespace 信息，单击【确定】删除 Namespace。

集群 <span style="border: 1px solid blue; padding: 2px;">广州(1)</span> 上海(0) 北京(0) 成都(0) 香港(0) 新加坡(0) 孟买(0) 硅谷(0) 弗吉尼亚(0) 莫斯科(0) 法兰克福(0)						
<b>新建</b>						
ID/名称	监控	集群状态	kubernetes版本	节点状态	节点数量	
cls-h9bgbkzi vxjun-custer		运行中	1.7.8	全部正常	1台	

**注意：**

删除 Namespace 将销毁 Namespace 下的所有资源，销毁后所有数据将被清除且不可恢复，清除前将请提前备份数据。

# 添加已有云主机

最近更新时间：2018-10-09 15:19:00

## 添加已有云主机

### 概述

腾讯云容器服务支持新增节点到容器集群，同时也支持添加已有的云主机到集群内。

当前添加已有云主机到集群功能仅支持与集群在同一VPC内的主机，敬请期待基础网络 and 不同VPC内的云主机资源复用。

### 前置说明

1. 当前仅支持添加同一VPC下的云主机。
2. 添加已有节点到集群，将重装改云主机的操作系统。
3. 添加已有节点到集群，将迁移主机所属项目到集群所设置的项目
4. 有且仅有一块数据盘的节点加入到集群，可以选择是否设置容器目录，设置容器目录会格式化数据盘。无数据盘或多块数据盘的云主机设置容器目录不生效。

### 操作方法

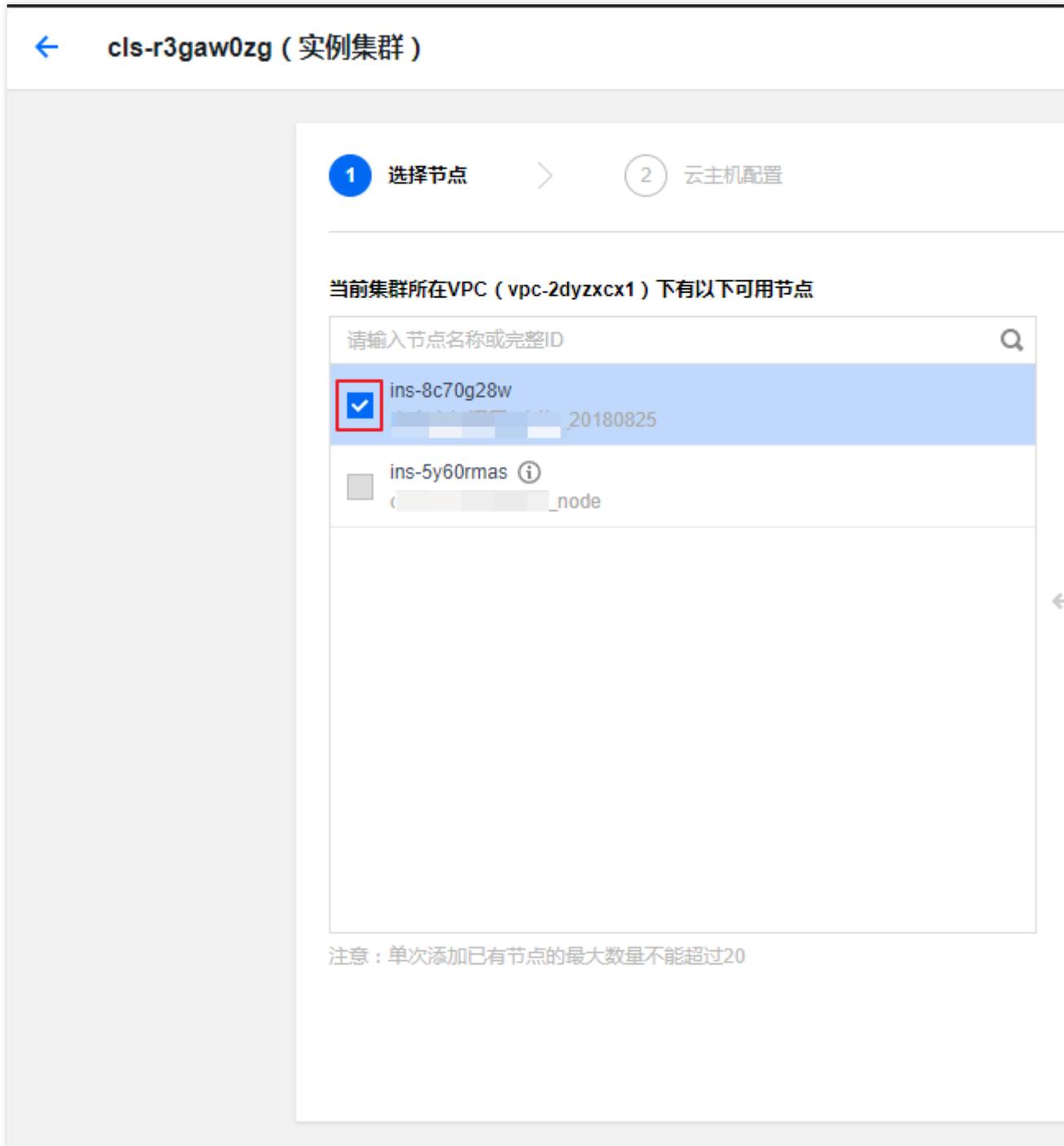
1. 在集群列表页中选择某集群的ID/节点名，单击【节点列表】，选择添加已有节点。



The screenshot displays the 'Node List' page for a cluster named 'cls-r3gaw0zg (实例集群)'. The page has four tabs: '节点列表' (Node List), 'Namespace列表' (Namespace List), '伸缩组列表' (Scaling Group List), and '集群信息' (Cluster Information). The '节点列表' tab is active. At the top, there are five buttons: '新建节点' (New Node), '添加已有节点' (Add Existing Node), '移出' (Remove), '封锁' (Lock), and '取消封锁' (Unlock). The '添加已有节点' button is highlighted with a red border. Below the buttons is a table with the following columns: 'ID/节点名' (ID/Node Name), '状态' (Status), 'kubernetes版本' (Kubernetes Version), and '主机类型' (Host Type). The table contains one row with the following data:

ID/节点名	状态	kubernetes版本	主机类型
<input type="checkbox"/> ins-5y60rmas ccs_cls-r3gaw0zg_...	健康	1.10.5	标准型S2

2. 选择需要添加到集群的云主机。



3. 设置容器目录

- 暂不设置：不设置容器和镜像的存储路径，系统盘容量足够大或无数据盘的云主机，或者数据盘需要后续手动挂载并使用的情况，建议选择不设置容器目录。
- 将容器与镜像存储在数据盘：系统盘容量较小，或有数据盘的主机，需接受格式化数据盘的情况，可选择设置容器与镜像的存储目录。

数据盘设置  暂不设置  将容器和镜像存储在数据盘

云主机的数据盘将自动格式化为ext4，且docker、docker的镜像的系统程序自动存放到您指定的容器目录。  
注意：无数据盘或有多块数据盘的云主机的容器目录设置不生效。

容器目录

设置容器目录将自动格式化数据盘，请注意备份数据，以免造成数据丢失。

4. 设置节点的登录方式。

登录方式

注：请牢记您所设置的密码，如遗忘可登录CVM控制台重置密码。

用户名

密码

linux机器密码需8到16位，至少包括两项（[a-z,A-Z]，[0-9]和[() `~!@#\$%^&\*+=[\];',./?]的特殊符号）

确认密码

5. 设置安全组（迁移项目后云主机需要选择安全组，建议选择跟集群内其他云主机相同的安全组）。

安全组 ⓘ   [使用指引](#)

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

如您有业务需要放通其他端口，您可以 [新建安全组](#)

# 集群的自动扩缩容

最近更新时间：2018-10-09 18:02:55

## 集群的自动扩缩容

### 1.简介

集群自动扩缩容，又称Cluster Autoscaler ( CA )，是一个独立的程序，它可以动态地调整集群的节点数量来满足需求。当集群中出现由于资源不足而无法调度的pod时自动触发扩容，从而减少人力成本。当满足节点空闲等缩容条件时自动触发缩容，为您节约资源成本。

### 2.使用方法

#### 2.1 开启集群自动伸缩

## 1.创建集群开启自动扩缩容，自动创建伸缩组

[< 返回](#) | [创建集群](#)

密码   
linux机器密码需8到16位，至少包括两项（[a-z,A-Z]，[0-9]和[() `~!@#\$%^&\*+=|[]:;,./?]的特殊符号）

确认密码

安全组 ①  [使用指引](#)  
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  
如您有业务需要放通其他端口，您可以 [新建安全组](#)

云主机数量

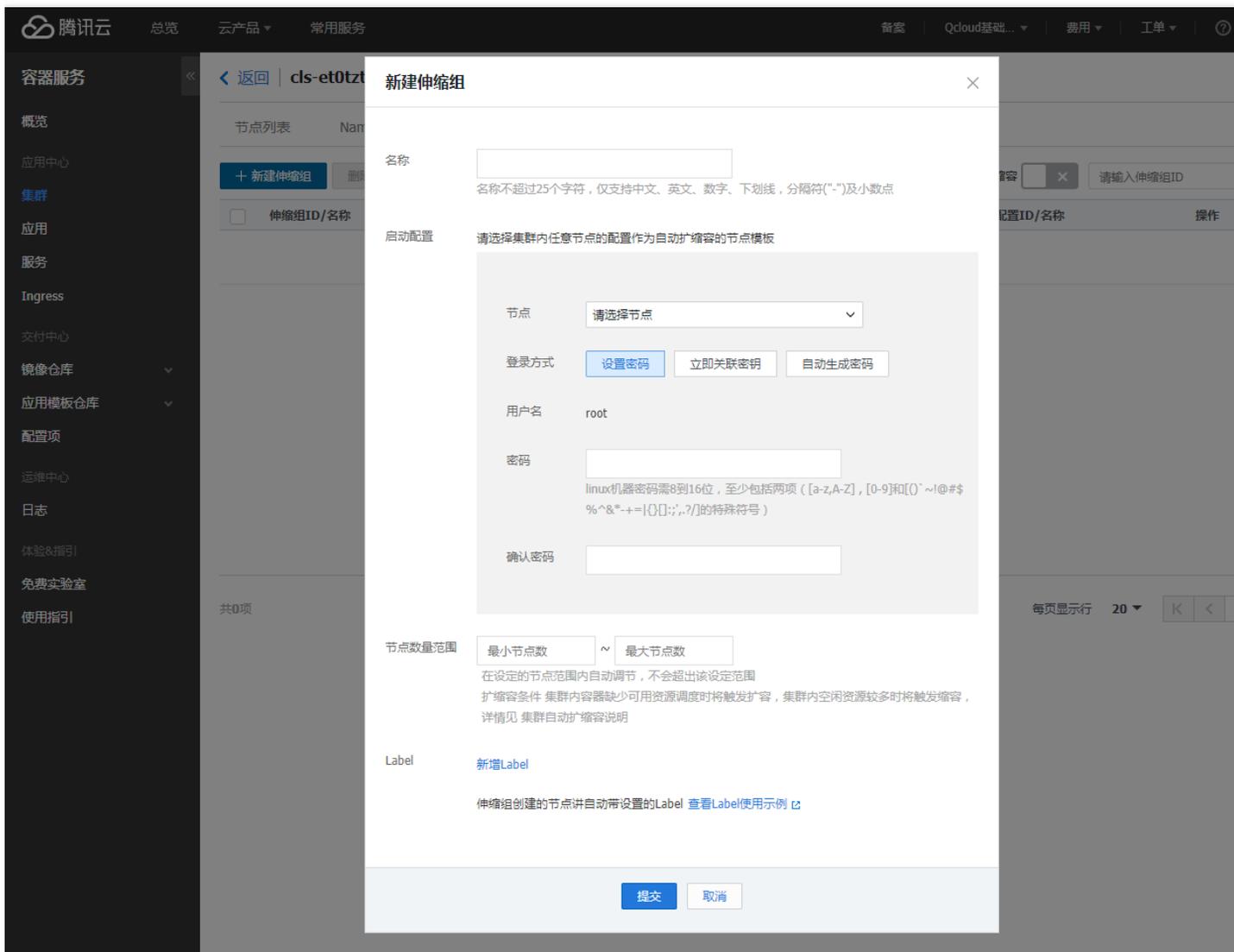
自动调节  开启  
自动创建最大节点数为2的伸缩组，当集群内节点出现异常导致集群资源不足时，将创建同等配置的按量计费的云主机来避免集群故障

总计费用 ¥ 0.24/小时(配置费用) | ¥ 0.06/小时(网络费用)

[上一步](#)

[完成](#)

## 2.为集群添加多个伸缩组



使用自动扩缩容需要创建弹性伸缩组，可以指定最小最大值，以及label。

- 最小最大值---限制了伸缩组内节点的数量范围。
- label---为伸缩组设置label，会在自动扩容出的节点上设置label，从而实现服务的灵活调度策略。

### 注意事项：

1. 需要配置服务下容器的request值：自动扩容的触发条件是集群中存在由于资源不足而无法调度的pod，而判断资源是否充足正是基于pod的request来进行的。
2. 不要直接修改属于伸缩组内的节点。
3. 同一伸缩组内的所有节点应该具有相同的配置（机型和label等）。
4. 可以使用PodDisruptionBudget(敬请期待UI支持)来防止Pod在缩容时被删除。
5. 在指定伸缩组的最小/最大值节点数量设置之前，检查所在可用区的配额是否足够大。

6. 不建议启用基于监控指标的节点弹性伸缩
7. 删除伸缩组会同时销毁伸缩组内的CVM，请谨慎操作

## 2.2 扩容缩容触发条件

### 扩容条件

集群中出现因为缺少可用资源而无法调度的容器实例时，将触发自动扩容策略，尝试扩容节点来运行这些实例。每当kubernetes调度程序找不到一个运行pod的地方时，它会将pod的PodCondition设置为false，并将原因设置为“不可调度的”。集群自动扩缩容程序正是每隔一段时间扫描一次是否有不可调度的pod来进行扩容的，如果有就尝试扩容节点来运行这些pod。

### 缩容条件

当节点上所有pod（实例）的CPU和内存request占比同时小于50%时，作为备选缩容节点尝试缩容，如果满足如下描述的所有缩容条件，此节点上的所有POD都可调度到其他节点上，才会进行缩容。

节点上有以下类型的pod（实例）时不会被缩容：

- 设置了严格的PodDisruptionBudget的pod，不满足PDB则不会缩容
- Kube-system下的pod
- 节点上有非deployment, replica set, job, stateful set等等控制器创建的pod
- pod有本地存储
- 不能被调度到其他节点上的pod

## 3. 扩容缩容常见问题

### 3.1. Cluster Autoscaler与基于监控指标的弹性伸缩的节点扩缩容有什么不同？

Cluster Autoscaler确保集群中的所有POD都可调度，而不管具体的负载。而且它试图确保集群中没有不需要的节点。

基于监控指标的节点弹性伸缩在自动扩缩时不关心POD。因此可能会添加一个没有任何POD的节点，或者删除一个有一些系统关键POD的节点，比如kube-dns。这种自动缩容机制是Kubernetes不鼓励的。因此他们是冲突的，请不要同时启用。

### 3.2. CA和伸缩组的对应关系

启用CA的集群，会根据选择的节点配置创建一个启动配置和绑定此启动配置的伸缩组。绑定后会在此伸缩组内进行扩缩容，扩容后的cvm自动加入集群。自动扩缩容的节点都是按量计费的。伸缩组的相关文档请参见[弹性伸缩文档](#)。

### 3.3. CA会不会缩容我在容器服务控制台手动添加的节点

不会，CA缩容的节点只限于伸缩组内的节点。在[容器服务控制台](#)添加的节点不会加入伸缩组，只有在伸缩组内的节点才可能缩容。

### 3.4.可以在弹性伸缩控制台添加或者移出云主机吗？

不可以，不建议您在[弹性伸缩控制台](#)进行任何修改操作。

### 3.5.会继承所选节点的哪些配置

创建伸缩组时，需要选择集群内的一个节点作为参考来创建[启动配置](#)，参考的节点配置包括：

- vCPU
- 内存
- 系统盘大小
- 数据盘大小
- 磁盘类型
- 带宽
- 带宽计费模式
- 是否分配公网IP
- 安全组
- 私有网络
- 子网

### 3.6.如何使用多个伸缩组？

根据服务的重要级别、类型等特点，可以通过创建多个伸缩组，为伸缩组设置不同的label，从而指定伸缩组扩容出节点的label，来对服务进行分类。

### 3.7.最大值可以设置为多大？

目前腾讯云用户每个可用区均有30个按量计费类型 CVM 配额，如果希望伸缩组有超过 30 台按量计费的 CVM，请提交工单申请。

请参见您当前可用区的云服务器[实例数及配额](#)。另外弹性伸缩也有最大值的限制, 最大是200, 如果超过此值请提交工单申请。

### 3.8.我的集群启用缩容安全吗？

由于在缩容节点时会发生 Pod 重新调度的情况，所以服务必须可以容忍重新调度和短时的中断时再启用缩容。建议为您的服务设置PDB。PDB能指定一个Pod集合在任何时候处于运行状态的副本的最小数量或者最小百分比。有了PodDisruptionBudget，应用部署者可以保证那些会主动移除Pod的集群操作永远不会同一时间销毁太多Pod，从而导致数据丢失，服务中断或者无法接受的服务降级等后果。

### 3.9.节点上有哪些类型的pod时不会被缩容

- 设置了严格的PodDisruptionBudget的pod，不满足PDB则不会缩容

- Kube-system下的pod
- 节点上有非deployment, replica set, job, stateful set等等控制器创建的pod
- pod有本地存储
- pod不能被调度到其他节点上

### 3.10.节点满足缩容条件后多长时间会触发缩容

10分钟

### 3.11.节点Not Ready后多长时间会触发缩容

20分钟

### 3.12.多长时间扫描一次是否需要扩缩容

10秒

### 3.13.需要扩容时多长时间可以扩容出cvm ?

一般在10分钟内，相关弹性伸缩的说明文档请参见[弹性伸缩](#)

### 3.14.为什么有Unschedulable的pod，却未进行扩容？

请确认pod的请求资源是否过大，是否设置了node selector，伸缩组的最大值是否已经达到，帐号余额是否充足（帐号余额不足，弹性伸缩无法扩容），以及配额不足等[其他原因](#)

### 3.15.如何防止Cluster Autoscaler缩容特定节点？

# 可以在节点的annotations中设置如下信息

```
kubectl annotate node <nodename> cluster-autoscaler.kubernetes.io/scale-down-disabled=true
```

### 3.16.扩缩容事件如何反馈给用户

用户可在弹性伸缩控制台查询伸缩组的伸缩活动，也可查看k8s的事件。在下面三种资源上都会有对应的事件

1.kube-system/cluster-autoscaler-status

2.pod

3.node

- kube-system/cluster-autoscaler-status config map:
  - **ScaledUpGroup** - CA 触发扩容.
  - **ScaleDownEmpty** - CA删除了一个没有运行pod的节点.
  - **ScaleDown** - CA缩容.
- node:
  - **ScaleDown** - CA缩容.
  - **ScaleDownFailed** - CA 缩容失败.

- 
- pod:
    - **TriggeredScaleUp** - CA 由于此pod触发扩容.
    - **NotTriggerScaleUp** - CA 无法找到可扩容的伸缩组使得此pod可调度.
    - **ScaleDown** - CA 尝试驱逐此pod来缩容节点.

# 容器服务支持 GPU 调度

最近更新时间：2018-06-21 15:49:33

## 简介

如果您的业务需要进行深度学习、高性能计算等场景，您可以使用腾讯云容器服务支持 GPU 功能，通过该功能可以帮助您快速使用 GPU 容器。如需要开通 GPU 功能，您可以 [提交工单](#) 进行申请。

## 使用指南

### 一. 在集群中添加 GPU 节点

添加 GPU 节点有以下两种方法：

#### 方法 1：新建 GPU 云服务器

1. 在 [容器服务控制台](#) 新建集群时选择新建 GPU 云服务器，选择 GPU 机型。

计费模式  按量计费  包年包月 [详细对比](#)

所在地域 西南地区(成都)

可用区  成都一区  成都二区

节点网络   共4093个子网IP，剩4090个可用

实例族

实例类型

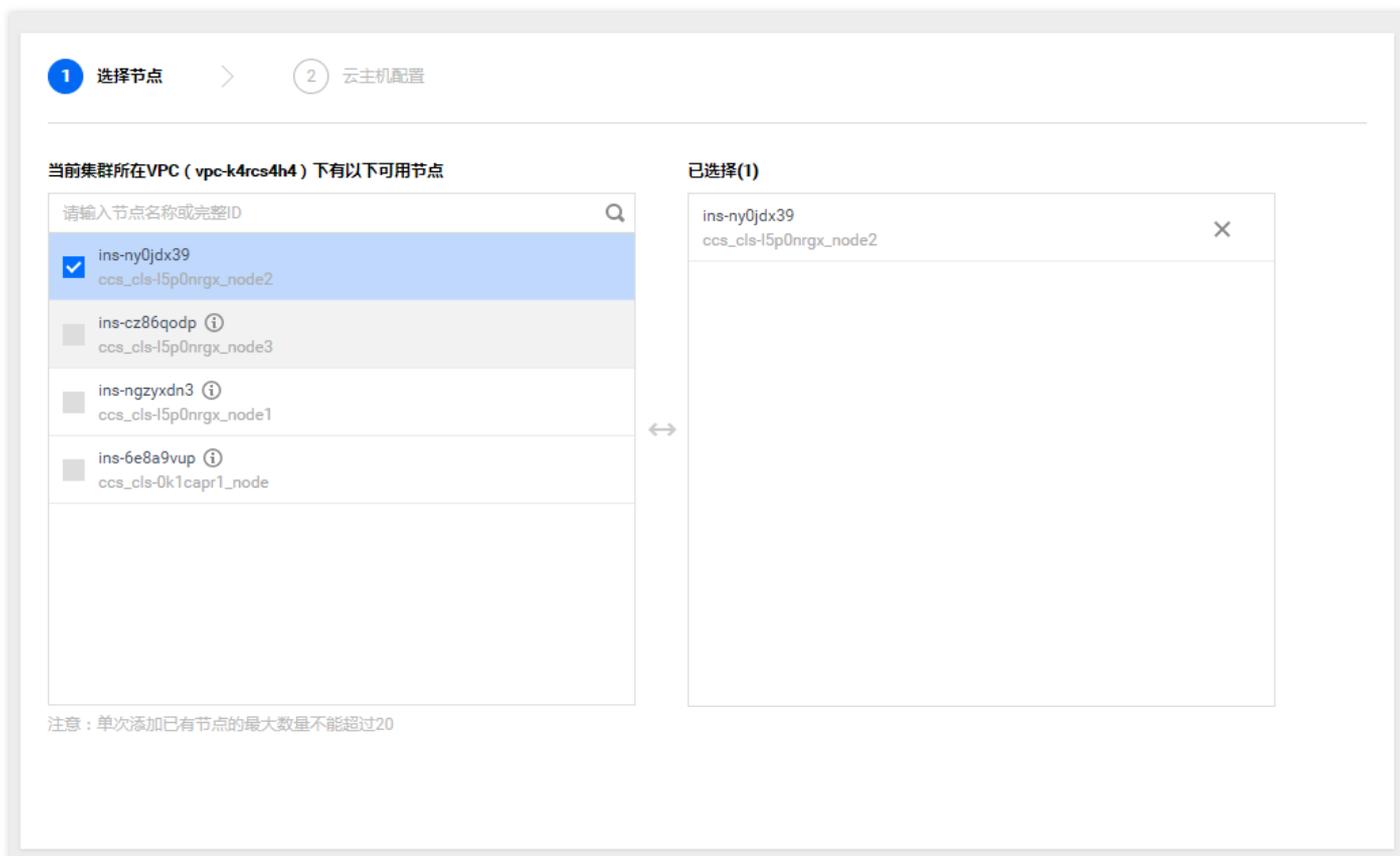
机型	规格	CPU	内存	配置费用
<input checked="" type="radio"/> GPU计算型GN8	GN8.XLARGE56	6核	56GB	¥ 11.62元/小时起
<input type="radio"/> GPU计算型GN8	GN8.3XLARGE112	14核	112GB	¥ 23.94元/小时起
<input type="radio"/> GPU计算型GN8	GN8.7XLARGE224	28核	224GB	¥ 47.88元/小时起
<input type="radio"/> GPU计算型GN8	GN8.14XLARGE448	56核	448GB	¥ 95.76元/小时起

2. 选择 GPU 的操作系统，并完成创建。



方法 2：添加已有 GPU 云服务器

1. 在 [容器服务控制台](#) 新建集群时选择已有的 GPU 节点



2. 选择 GPU 的操作系统，并完成添加。

新增资源所属项目 默认项目

集群内新增的云主机、负载均衡器等资源将会自动分配到该项目下。[使用指引](#)

操作系统 ① CentOS 7.2 64位 GPU

请确认所有节点都是GPU机型，否则有可能安装失败

登录方式 设置密码 立即关联密钥 自动生成密码

注：请牢记您所设置的密码，如遗忘可登录CVM控制台重置密码。

用户名 root

密码

linux机器密码需8到16位，至少包括两项（[a-z,A-Z],[0-9]和[()~!@#\$%^&\*+=[]\;:~/?]的特殊符号）

确认密码

安全组 ① 请选择安全组 使用指引

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  
如您有业务需要放通其他端口，您可以 [新建安全组](#)

二. 创建 GPU 服务的容器

您可以通过控制台创建服务时选择容器占用的 GPU 的个数。

数据卷(选填) ⓘ

[添加数据卷](#)

为容器提供存储，目前支持临时路径、主机路径、云硬盘数据卷、文件存储NFS、配置项，还需挂载到容器的指定路径中。 [使用指引](#)

运行容器

✓ ×

名称

最长63个字符，只能包含小写字母、数字及分隔符("-)，且不能以分隔符开头或结尾

镜像  [选择镜像](#)

镜像版本 ( Tag )

CPU/内存限制

CPU限制	内存限制
<input type="text" value="request"/> <input type="text" value="0.25"/> - <input type="text" value="limit"/> <input type="text" value="0.5"/> 核	<input type="text" value="request"/> <input type="text" value="256"/> - <input type="text" value="limit"/> <input type="text" value="1024"/> MiB

Request用于预分配资源,当集群中的节点没有request所要求的资源数量时,容器会创建失败。  
Limit用于设置容器使用资源的最大上限,避免异常情况下节点资源消耗过多。

GPU限制    个

环境变量 ⓘ [新增变量](#) [从配置项导入](#)

变量名只能包含大小写字母、数字及下划线，并且不能以数字开头

---

[显示高级设置](#)

注意：服务创建完成后，容器的配置信息可以通过更新服务的方式进行修改

[添加容器](#)

或通过应用或 kubectl 命令创建，在 YAML 文件中添加 GPU 字段。

模板内容

模板可以通过从UI导入服务或新增空服务并手动编写来创建多个服务的YAML描述，详情可查看 [应用模板操作指引](#)

服务名	操作	内容
[REDACTED]	删除	<pre> 19 metadata: 20   creationTimestamp: null 21 spec: 22   containers: 23   - image: nginx 24     imagePullPolicy: Always 25     name: test 26   resources: 27     limits: 28       cpu: 500m 29       memory: 1Gi 30       nvidia.com/gpu: "1" 31   requests: 32     cpu: 250m 33     memory: 256Mi 34   securityContext: 35     privileged: false 36   serviceAccountName: "" 37   volumes: null           </pre>

[新增空服务](#) [从UI导入服务](#)

## 注意事项

1. 仅在集群 kubernetes 版本大于 **1.8.\*** 时支持使用 GPU 调度。
2. 当前 GPU 仅支持 CentOS 系统，请留意您的集群的操作系统。
3. 容器之间不共享 GPU，每个容器可以请求一个或多个 GPU。无法请求 GPU 的一小部分。
4. 建议搭配亲和性调度来使用 GPU 功能。

# 集群的生命周期

最近更新时间：2017-11-03 15:40:13

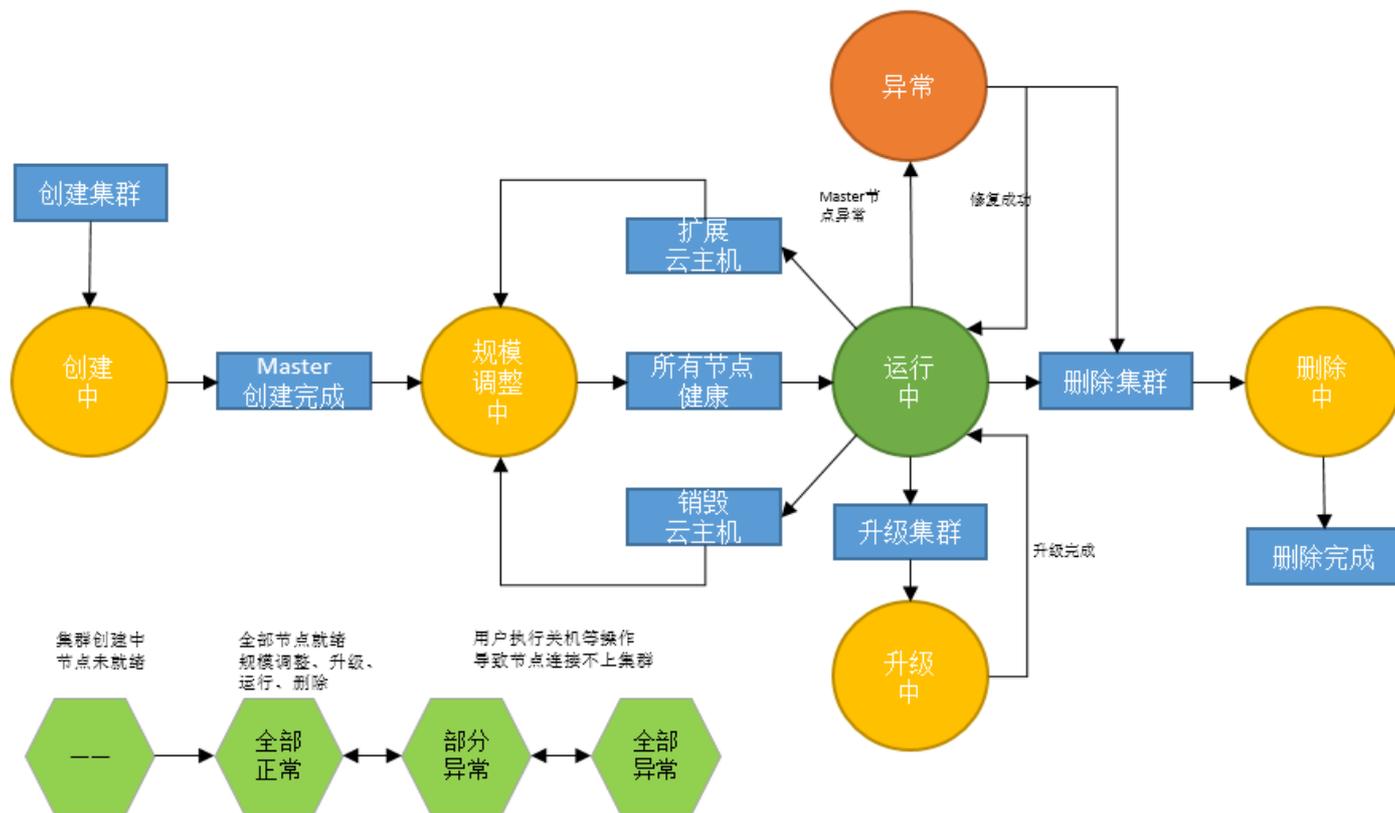
## 集群的生命周期

### 状态说明

状态	说明
创建中	集群正在创建，正在申请云资源
规模调整中	集群的节点数量变更，添加节点或销毁节点中
运行中	集群正常运行
升级中	升级集群中，敬请期待升级功能
删除中	集群在删除中
异常	集群中存在异常，如节点网络不可达等

### 状态流转图示

**集群生命周期**：集群状态之间转换如下图(六边形为节点状态)。



## 节点的生命周期

### 状态说明

状态	说明
健康	节点正常运行，并连接上集群
异常	节点运行异常，未连接上集群
其他状态	参考 <a href="#">云主机生命周期</a>

# Namespace使用指引

最近更新时间：2018-05-30 15:03:04

命名空间 ( Namespace ) 是对一组资源和对象的抽象集合。例如可以将开发环境，联调环境，测试环境的服务分别放到不同的 Namespace 中。

## Namespace 类别

Namespace 按创建类型分为两大类：集群默认创建的 Namespace 的和用户创建的 Namespace 。

### 集群默认创建的 Namespace

Kubernetes 集群在启动时会默认创建 `default` 和 `kube-system` 这两个命名空间，这两个命名空间不可以删除。

- 在不指定命名空间时，默认使用 `default namespace` 。
- 系统服务一般建议创建在 `kube-system namespace` 。

### 用户创建的 Namespace

用户可以在集群中按照需要创建 Namespace 。可以按照不同的环境创建对应的 Namespace ，例如开发环境，联调环境和测试环境分别创建对应的 Namespace 。或者按照不同的应用创建对应的 Namespace ，例如应用 App1 和应用 App2 分别创建对应的 Namespace 。

#### 注意：

用户创建的 Namespace 可以进行删除，但删除 Namespace 操作会依次删除 Namespace 下的所有服务。

## Namespace 操作指引

### 创建 Namespace

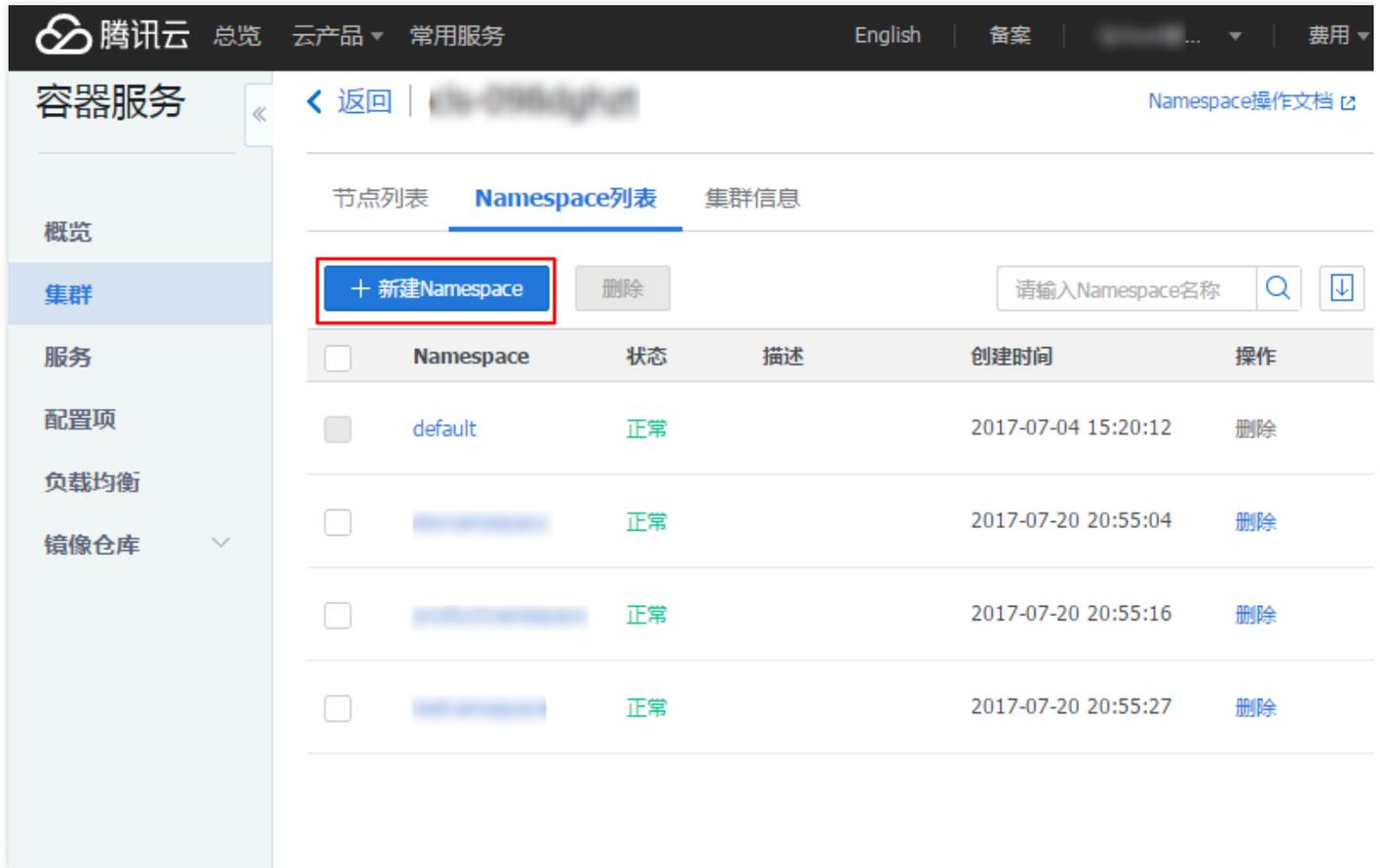
1. 登录 [容器服务控制台](#) 。
2. 单击左侧导航栏中的 **集群** 。

3. 在集群列表中单击集群的 ID/名称。

The screenshot shows the Tencent Cloud Container Service console. On the left is a navigation menu with options like '容器服务', '概览', '集群', '服务', '配置项', '负载均衡', and '镜像仓库'. The main area is titled '集群' and includes filters for regions: '广州(4)', '上海(1)', '北京(3)', and '新加坡(2)'. Below the filters is a '+ 新建' button and a table of clusters.

ID/名称	监控	集群状态	节点状态	节点数量	已分配/总
cls-7kaky7w		运行中	全部正常	2台	0.46/2
[blurred]		运行中	全部正常	1台	0.26/1
[blurred]		运行中	全部正常	4台	1.46/7
[blurred]		运行中	全部正常	8台	1.06/64

4. 单击 **Namespace 列表**，单击【新建 Namespace】。



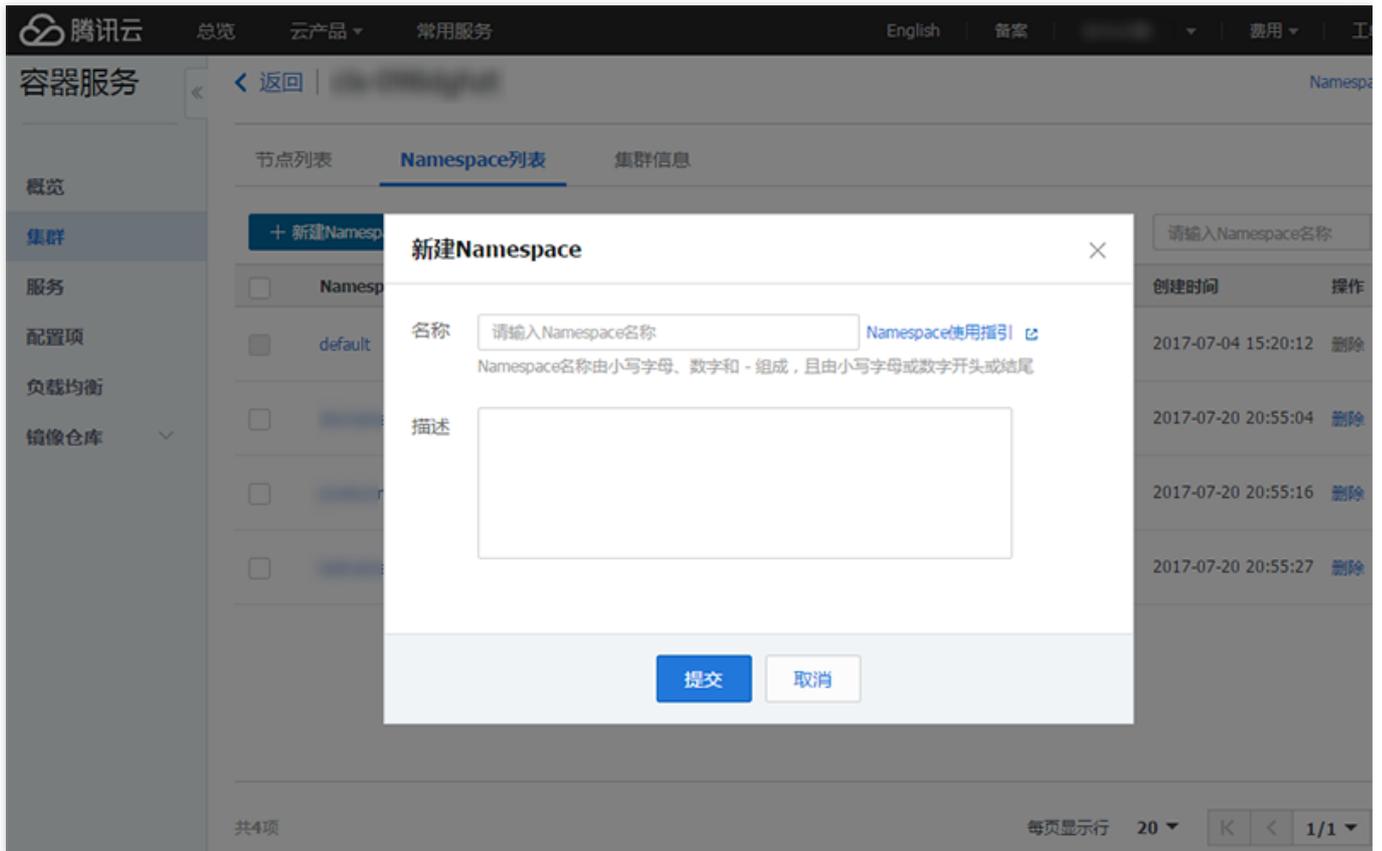
The screenshot shows the Tencent Cloud Container Service console. The left sidebar contains navigation options: 容器服务 (Container Service), 概览 (Overview), 集群 (Cluster), 服务 (Service), 配置项 (Configuration), 负载均衡 (Load Balancing), and 镜像仓库 (Image Registry). The main content area is titled 'Namespace列表' (Namespace List) and includes a search bar and a '+ 新建Namespace' button. Below the search bar is a table with columns for Namespace, Status, Description, Creation Time, and Action.

<input type="checkbox"/>	Namespace	状态	描述	创建时间	操作
<input type="checkbox"/>	default	正常		2017-07-04 15:20:12	删除
<input type="checkbox"/>		正常		2017-07-20 20:55:04	删除
<input type="checkbox"/>		正常		2017-07-20 20:55:16	删除
<input type="checkbox"/>		正常		2017-07-20 20:55:27	删除

5. 填写信息并单击【提交】。

- 名称：输入 Namespace 的名称。

- **描述**：创建 Namespace 的相关信息。该信息将显示在 **Namespace 列表** 页面。



## 查看 Namespace 列表

1. 登录 [容器服务控制台](#)。
2. 单击左侧导航栏中的 **集群**。

3. 在集群列表中单击集群的 ID/名称。



The screenshot shows the Tencent Cloud Container Service console. The left sidebar contains navigation options: 容器服务 (Container Service), 概览 (Overview), 集群 (Clusters), 服务 (Services), 配置项 (Configurations), 负载均衡 (Load Balancing), and 镜像仓库 (Image Registry). The main content area is titled '集群' (Clusters) and includes filters for regions: 广州(4) (Guangzhou 4), 上海(1) (Shanghai 1), 北京(3) (Beijing 3), and 新加坡(2) (Singapore 2). A '+ 新建' (New) button is visible. Below is a table of clusters:

ID/名称	监控	集群状态	节点状态	节点数量	已分配/总
cls-7kaky7w		运行中	全部正常	2台	0.46/2
[blurred]		运行中	全部正常	1台	0.26/1
[blurred]		运行中	全部正常	4台	1.46/7
[blurred]		运行中	全部正常	8台	1.06/64

4. 单击要查看集群的 **Namespace 列表**。

The screenshot displays the Tencent Cloud console interface for managing namespaces in a cluster. The top navigation bar includes the Tencent Cloud logo, '总览', '云产品', '常用服务', 'English', '备案', and '费用'. The left sidebar shows the '容器服务' (Container Service) menu with sub-items: '概览', '集群' (selected), '服务', '配置项', '负载均衡', and '镜像仓库'. The main content area is titled 'Namespace列表' and includes a search bar with the placeholder '请输入Namespace名称' and a search icon. Below the search bar is a table with the following data:

Namespace	状态	描述	创建时间	操作
default	正常		2017-07-04 15:20:12	删除
[blurred]	正常		2017-07-20 20:55:04	删除
[blurred]	正常		2017-07-20 20:55:16	删除
[blurred]	正常		2017-07-20 20:55:27	删除

使用 Namespace

1. 创建服务时，选择对应的 Namespace。

The screenshot shows the 'New Service' (新建服务) page in the Tencent Cloud console. The left sidebar contains navigation options: 概览 (Overview), 集群 (Cluster), 服务 (Service), 配置项 (Configuration), 负载均衡 (Load Balancing), and 镜像仓库 (Image Repository). The main content area is for creating a new service. The 'Run Cluster' (运行集群) dropdown menu is open, showing 'default' as the selected namespace. A red box highlights the dropdown menu. The 'Service Name' (服务名称) field is empty, with a placeholder '请输入服务名称, 不超过63个字符'. The 'Location' (所在地域) is set to '北京' (Beijing). The 'Service Description' (服务描述) field is empty, with a placeholder '请输入描述信息, 不超过1024个字符'. The 'Data Volume' (数据卷) section is optional, with a link to '添加数据卷' (Add Data Volume). The 'Run Container' (运行容器) section has fields for 'Name' (名称) and 'Image' (镜像), with a 'Select Image' (选择镜像) button. At the bottom, there are 'Create Service' (创建服务) and 'Cancel' (取消) buttons.

2. 查询服务时，选择对应的 Namespace ，查看对应 Namespace 下的所有服务。



### 删除集群 Namespace

1. 登录 [容器服务控制台](#)。
2. 单击左侧导航栏中的 **集群**。
3. 在集群列表中单击集群的 **ID/名称**。



4. 单击 **Namespace 列表**，选择需删除的 Namespace，单击右侧【删除】。

The screenshot shows the Tencent Cloud console interface for a cluster. The left sidebar contains navigation options: 容器服务 (Container Service), 概览 (Overview), 集群 (Cluster), 服务 (Services), 配置项 (Configurations), 负载均衡 (Load Balancing), and 镜像仓库 (Image Registry). The main content area is titled '容器服务' and shows the cluster ID 'cls-098dghzt'. Below the title, there are tabs for '节点列表' (Node List), 'Namespace列表' (Namespace List), and '集群信息' (Cluster Information). The 'Namespace列表' tab is active, displaying a table of namespaces. At the top of the table, there are buttons for '+ 新建Namespace' (New Namespace) and '删除' (Delete), along with a search input field labeled '请输入Namespace名称' (Please enter Namespace name). The table has columns for 'Namespace', '状态' (Status), '描述' (Description), '创建时间' (Creation Time), and '操作' (Action). The first row is 'default' with status '正常' (Normal). The second row has a blurred namespace name, status '正常', and a '删除' button highlighted with a red box. The third and fourth rows also have blurred namespace names, status '正常', and '删除' buttons.

Namespace	状态	描述	创建时间	操作
default	正常		2017-07-04 15:20:12	删除
[Blurred]	正常		2017-07-20 20:55:04	删除
[Blurred]	正常		2017-07-20 20:55:16	删除
[Blurred]	正常		2017-07-20 20:55:27	删除

5. 弹出提示页面，显示要删除的 Namespace 信息，单击【确定】删除。



#### 注意：

删除 Namespace 将销毁 Namespace 下的所有资源，销毁后所有数据将被清除且不可恢复，清除前将请提前备份数据。

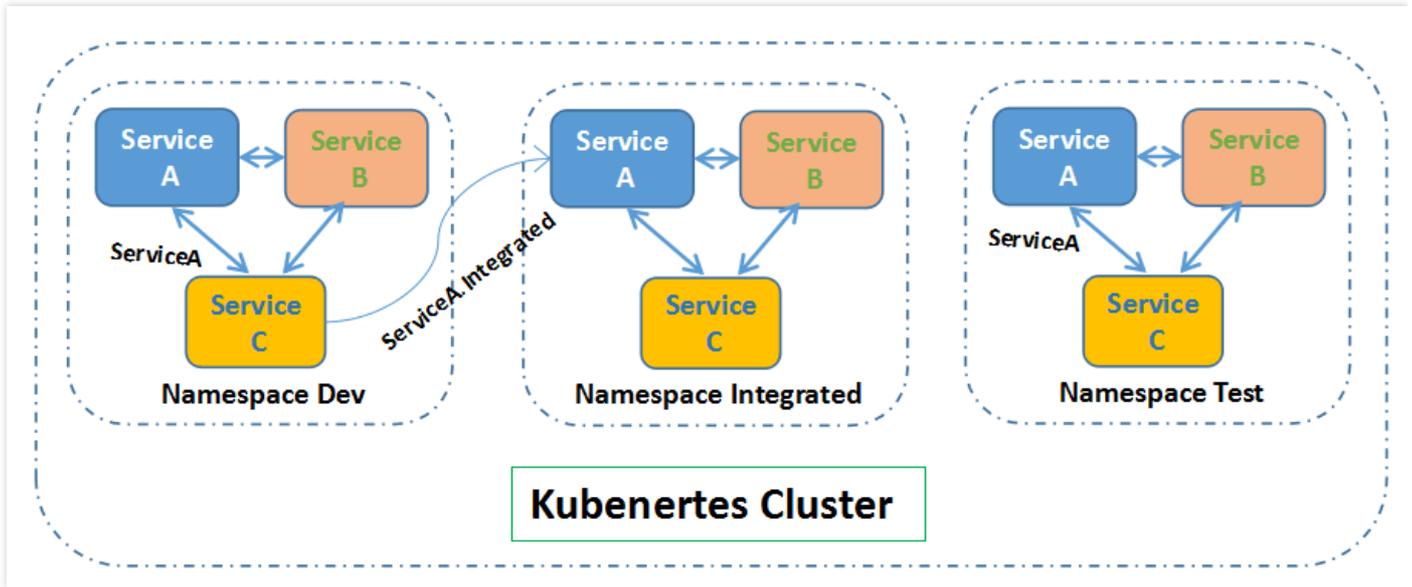
## Namespace 使用实践

### 按照不同环境划分 Namespace

一般情况下，服务的发布过程中会经过开发环境、联调环境、测试环境到生产环境的过程。这个过程中不同环境部署的服务相同，只是在逻辑上进行了定义。分为两种做法：

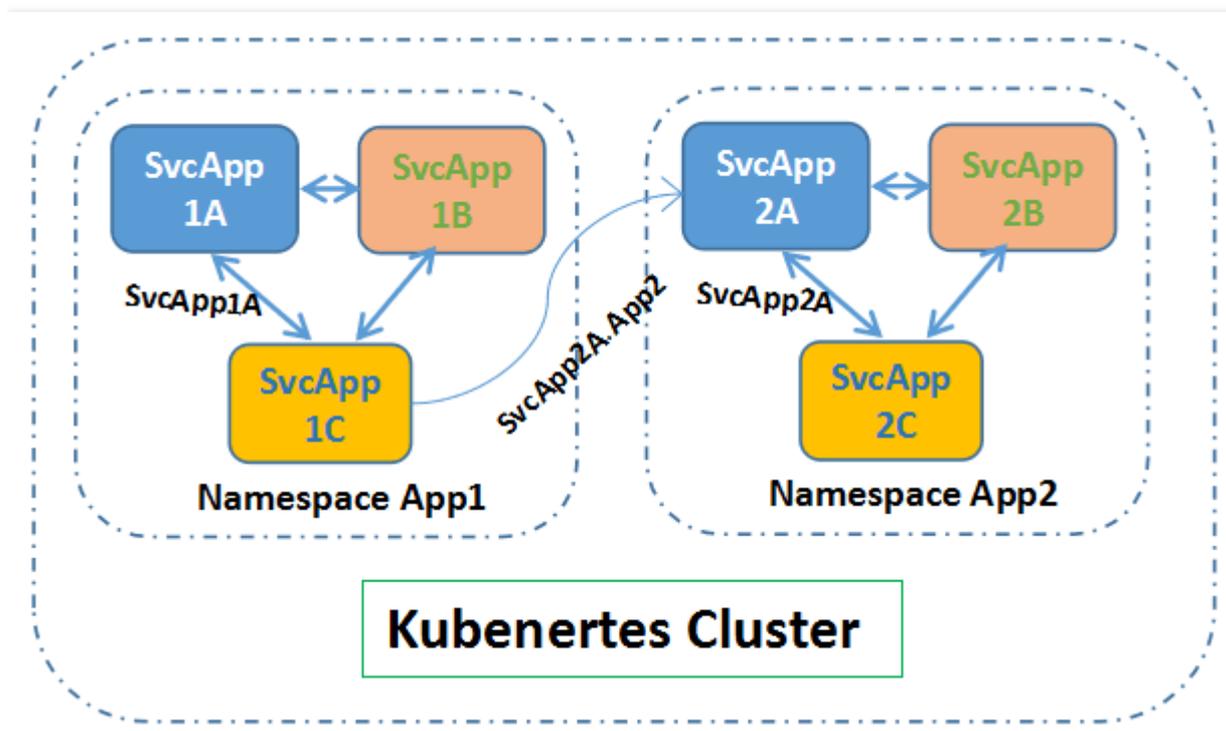
1. 分别创建不同的集群。但这样在不同环境中资源不能进行共享。同时，不同环境中的服务互访也需要通过服务配置的 Load Balance(负载均衡) 才能够实现。
2. 对于不同环境创建对应的 Namespace。同一 Namespace 下可以通过服务名称（service-name）直接访问，跨 Namespace 可以通过 service-name.namespace-name 访问。

例如下图，开发环境、联调环境和测试环境分别创建 Namespace Dev、Namespace Intergrated 和 Namespace Test。



### 按照应用划分 Namespace

对于同一个环境中，服务数量比较多的情况，建议进一步按照应用划分 Namespace。例如下图中，按照 App1 和 App2 划分了不同的 Namespace，将不同应用的服务在逻辑上当做一个服务组进行管理。



---

同样的，在同一个应用（同一个 Namespace）内的服务通过服务名称（service-name）直接访问，不同的应用（不同的 Namespace）通过 service-name.namespace-name 访问。

# 节点的使用指引

最近更新时间：2018-08-08 20:47:15

节点是指一台已注册到集群内的云服务器，一个集群由 n 个节点组成。腾讯云容器服务支持新增节点到容器集群，同时也支持添加已有的节点到集群内。

## 注意：

当前添加已有节点功能仅支持与集群在同一 VPC (私有网络) 内的主机，敬请期待基础网络 and 不同 VPC 内的云服务器资源复用。

## 前提条件

如果之前没有创建过集群，您需要先创建集群。有关如何创建集群的详细信息，参见 [新建集群](#)。

## 扩展节点

1. 登录 [容器服务控制台](#)。
2. 单击左侧导航栏中的【集群】，在集群列表中单击右侧【新增节点】。



The screenshot shows the Tencent Cloud Container Service console interface. On the left is a navigation menu with options like '概览', '应用中心', '集群', '服务', 'Ingress', '交付中心', '镜像仓库', '配置项', '运维中心', and '日志'. The main area displays a list of clusters. The first cluster is 'cls-kza53cvt' in the '广州(1)' region, with a status of '运行中' and '全部正常'. The '操作' column for this cluster has a red box around the '新建节点' button, with a tooltip that says '添加已有节点 更多'. At the bottom, there are pagination controls showing '共1项' and '每页显示行 20'.

ID/名称	监控	集群状态	节点状态	节点数量	已分配/总CPU	已分配/总内存	操作
cls-kza53cvt		运行中	全部正常	2台	0.72/2	0.21/2	新建节点 添加已有节点 更多

3. 设置新建节点所属网络、机型和配置信息。

<
返回
|
新建节点

✔
集群信息
>

2
选择机型
>

计费模式 ⓘ 按量计费 包年包月 [详细对比](#)

所在地域 华南地区 (广州)

可用区 ⓘ 广州二区 广州三区

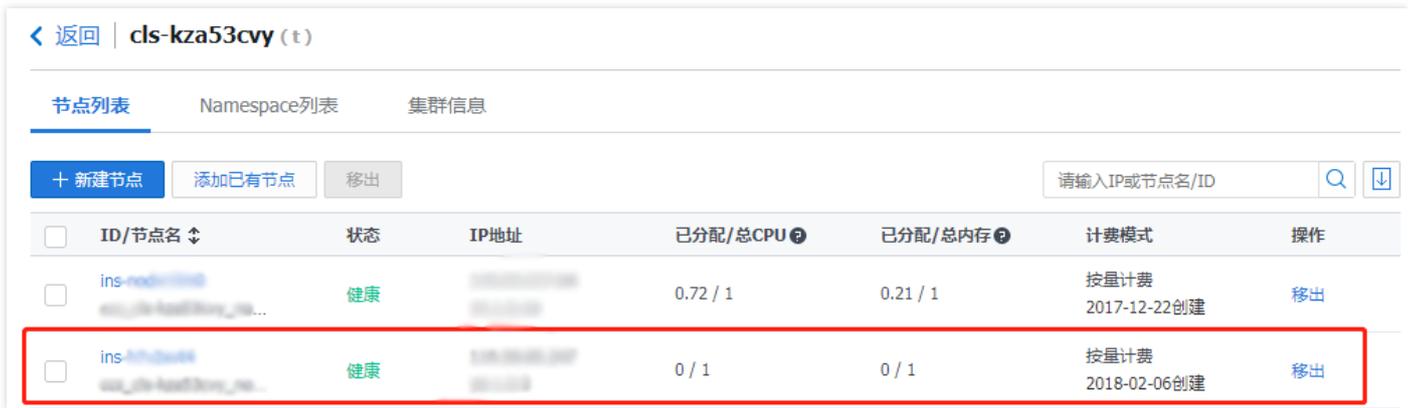
节点网络 ⓘ cr-test2 yunyuxiao\_test 共253个子网IP, 剩248个可用

系列 ⓘ 系列1 系列2 [详细对比](#)

机型 标准型S1

	机型	CPU ▾	内存 ▾
<input checked="" type="radio"/>	标准型S1	1核	1GB
<input type="radio"/>	标准型S1	1核	2GB
<input type="radio"/>	标准型S1	1核	4GB
<input type="radio"/>	标准型S1	2核	2GB

4. 新添加的节点将出现在节点列表中。



## 添加已有节点

1. 登录 [容器服务控制台](#)。
2. 单击左侧导航栏中的【集群】，在集群列表中单击右侧【添加已有节点】。



3. 在左侧可用节点列表栏选择要添加的节点，选择的节点 ID 将显示在右侧已选择栏。



4. 填写云服务器配置。提供三种对应登录方式。

- **设置密码**：请根据提示设置对应密码。
- **立即关联密钥**：密钥对是通过一种算法生成的一对参数，是一种比常规密码更安全的登录云服务器的方式。详细参阅 [SSH 密钥](#)。

- **自动生成密码**：自动生成的密码将通过站内信发送给您。

< 返回 | cls-098dghzt

① 选择节点
② 云主机配置

已选节点 ins- ,

提示：以上节点需要重装系统

注意：重装后，节点系统盘内的所有数据将被清除，恢复到初始状态；节点数据盘的数据不会丢失，但需要手动挂载才能使用。

操作系统 ⓘ Ubuntu 16.04 64位

登录方式 设置密码 立即关联密钥 自动生成密码

注：请牢记您所设置的密码，如遗忘可登录CVM控制台重置密码。

用户名 ubuntu

密码 请输入主机密码

linux机器密码需8到16位，至少包括两项（[a-z,A-Z]，[0-9]和[() `~!@#\$%^&\*~+=|[]:;,./?]的特殊符号）

确认密码 请输入主机密码

5. 单击【完成】，新添加的节点将出现在节点列表中。

< 返回 | cls-098dghzt

节点列表
Namespace列表
集群信息

+ 新建节点
添加已有节点
移出

🔍
⌵

ID/节点名 ↕	状态	IP地址	已分配/总CPU <sup>?</sup>	已分配/总内存 <sup>?</sup>	计费模式	操作
<input type="checkbox"/> <span style="color: blue;">ins-<span style="background-color: #ccc; padding: 0 5px;"> </span></span> ccs_cls-098dghzt	健康		0.86 / 1	0.36 / 1	按量计费 2017-07-27创建	移出
<input type="checkbox"/> <span style="color: blue;">ins-<span style="background-color: #ccc; padding: 0 5px;"> </span></span> ccs_cls-098dghzt	健康		0.6 / 1	0.38 / 1	包年包月 2017-10-04到期	移出
<input checked="" type="checkbox"/> <span style="color: blue;">ins-<span style="background-color: #ccc; padding: 0 5px;"> </span></span> 未命名	创建中		- / 1	- / 1		移出

**注意：**

- i. 当前仅支持添加同一 VPC 下的云服务器。
- ii. 添加存量的云服务器到集群，将重装该云服务器的操作系统。

## 查看节点信息

1. 在集群列表中，单击集群的 **ID/名称**（如 cls-098dghzt）。

The screenshot shows the Tencent Cloud console interface for cluster management. On the left is a navigation sidebar with options like '概览', '应用中心', '集群', '服务', 'Ingress', '交付中心', '镜像仓库', '配置项', '运维中心', and '日志'. The main area displays a table of clusters. The first cluster, 'cls-kza53cvyt', is highlighted with a red box. The table columns include 'ID/名称', '监控', '集群状态', '节点状态', '节点数量', '已分配/总CPU', '已分配/总内存', and '操作'. The cluster 'cls-kza53cvyt' has a status of '运行中' and '全部正常'. Below the table, there is a pagination control showing '共1项' and '每页显示行 20'.

ID/名称	监控	集群状态	节点状态	节点数量	已分配/总CPU	已分配/总内存	操作
cls-kza53cvyt		运行中	全部正常	3台	0.72/3	0.21/3	新建节点 添加已有节点 更多

2. 进入**节点列表**查看集群节点信息。

< 返回
cls-kza53cvy (t)

节点列表
Namespace列表
集群信息

+ 新建节点
添加已有节点
移出

请输入IP或节点名/ID
 🔍
📄

<input type="checkbox"/>	ID/节点名 ↓	状态	IP地址	已分配/总CPU	已分配/总内存	计费模式	操作
<input type="checkbox"/>	ins- <span style="color: #00a0e3;">xxxxxx</span>	健康	192.168.1.1	0.72 / 1	0.21 / 1	按量计费 2017-12-22创建	移出
<input type="checkbox"/>	ins- <span style="color: #00a0e3;">xxxxxx</span>	健康	192.168.1.2	0 / 1	0 / 1	按量计费 2018-02-06创建	移出
<input type="checkbox"/>	ins- <span style="color: #00a0e3;">xxxxxx</span>	健康	192.168.1.3	0 / 1	0 / 1	按量计费 2018-02-06创建	移出

共3项
每页显示行 20

⏪ ⏩ 1/1 ⏪ ⏩

## 移出节点

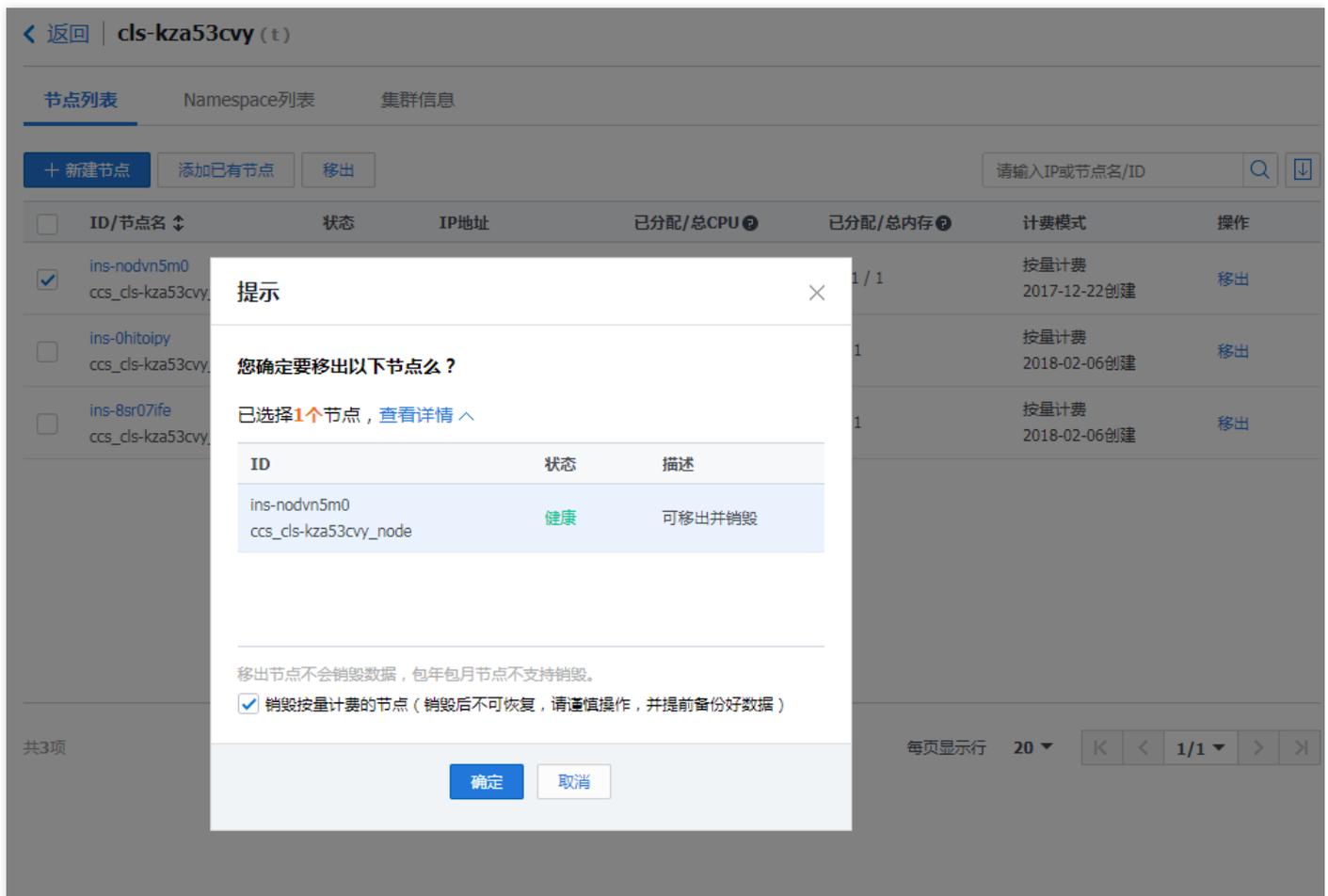
1. 在集群列表中，单击集群的 **ID/名称**（如 cls-098dghzt）。

The screenshot shows the 'Container Service' (容器服务) console. On the left is a navigation menu with options like 'Overview' (概览), 'Application Center' (应用中心), 'Cluster' (集群), 'Service' (服务), 'Ingress', 'Delivery Center' (交付中心), 'Image Repository' (镜像仓库), 'Configuration' (配置项), 'Operations Center' (运维中心), and 'Logs' (日志). The main area displays a list of clusters under the 'Cluster' (集群) tab, with filters for regions: 'Guangzhou (1)' (广州(1)), 'Shanghai (0)' (上海(0)), 'Beijing (0)' (北京(0)), 'Singapore (0)' (新加坡(0)), and 'Hong Kong (0)' (香港(0)). A search bar is present with the placeholder '请输入集群名称'. The cluster list table has columns: 'ID/名称', 'Monitoring' (监控), 'Cluster Status' (集群状态), 'Node Status' (节点状态), 'Node Count' (节点数量), 'Allocated CPU' (已分配/总CPU), 'Allocated Memory' (已分配/总内存), and 'Operations' (操作). The cluster 'cls-kza53cvy' is highlighted with a red box. Below the table, it shows '共1项' (Total 1 item) and pagination controls for '20' items per page.

2. 进入 **节点列表** 页面，单击右侧【移出】。

The screenshot shows the 'Node List' (节点列表) page for the cluster 'cls-kza53cvy'. The breadcrumb is '< 返回 | cls-kza53cvy (t)'. There are three tabs: 'Node List' (节点列表), 'Namespace List' (Namespace列表), and 'Cluster Information' (集群信息). The 'Node List' tab is active. At the top, there are buttons for '+ New Node' (新建节点), 'Add Existing Node' (添加已有节点), and 'Remove' (移出). A search bar is present with the placeholder '请输入IP或节点名/ID'. The table has columns: 'ID/Node Name', 'Status' (状态), 'IP Address' (IP地址), 'Allocated CPU' (已分配/总CPU), 'Allocated Memory' (已分配/总内存), 'Billing Mode' (计费模式), and 'Operations' (操作). Three nodes are listed, all with a status of 'Healthy' (健康). The 'Remove' button for the first node is highlighted with a red box.

3. 弹出提示页面，显示要移出的节点信息，单击【确定】删除节点。



## 封锁 ( cordon ) 节点

封锁节点后，将不接受新的 Pod 调度到该节点，需要手动取消封锁的节点。

### 方法一

在新增节点时在高级设置中勾选封锁节点，用于先进行业务所需的初始化操作。

高级设置

自定义数据 ⓘ

可选，用于启动时配置实例，支持 Shell 格式，原始数据不能超过 16 KB

封锁  开启封锁

封锁节点后，将不接受新的Pod调度到该节点，需要手动取消封锁的节点，或在自定义数据中执行 [取消封锁命令](#)

## 方法二

在节点列表页对选中节点进行封锁。

节点列表    Namespace列表    伸缩组列表    集群信息

新建节点    添加已有节点    移出    **封锁**    取消封锁

请输入IP或节点名/ID

ID/节点名	状态	主机类型	配置	IP地址	已分配/总资源	所属伸缩组	计费模式	操作
<input checked="" type="checkbox"/> ins-dj4sz5gk ccs_cls-q0...	健康	标准网络...	1核, 2GB, 1... 系统盘: 50GB ...	150.109... 10.0.0.11	CPU: 0.26 / 0.94 内存: 0.11 / 1.42	-	按量计费 2018-05-2...	移出 更多

## 取消封锁 (uncordon) 节点

取消封锁节点后，将允许新的 Pod 调度到该节点。

### 方法一

在新增节点时的脚本中添加取消封锁的命令。

如下，执行完成您的自定义命令后，再执行 `kubectl uncordon` 的命令，即可取消封锁节点。

```
#!/bin/sh
# your initialization script
echo "hello world!"

# If you set unschedulable when you create a node,
```

```
# after executing your initialization script,  
# use the following command to make the node schedulable.  
node=`ifconfig eth0 | grep inet | awk '{print $2}' | tr -d "addr:"`  
#echo ${node}  
kubectl uncordon ${node} --kubeconfig=/root/.kube/config
```

## 方法二

在节点列表页对已封锁的节点进行取消封锁。



## 驱逐 ( drain ) 节点

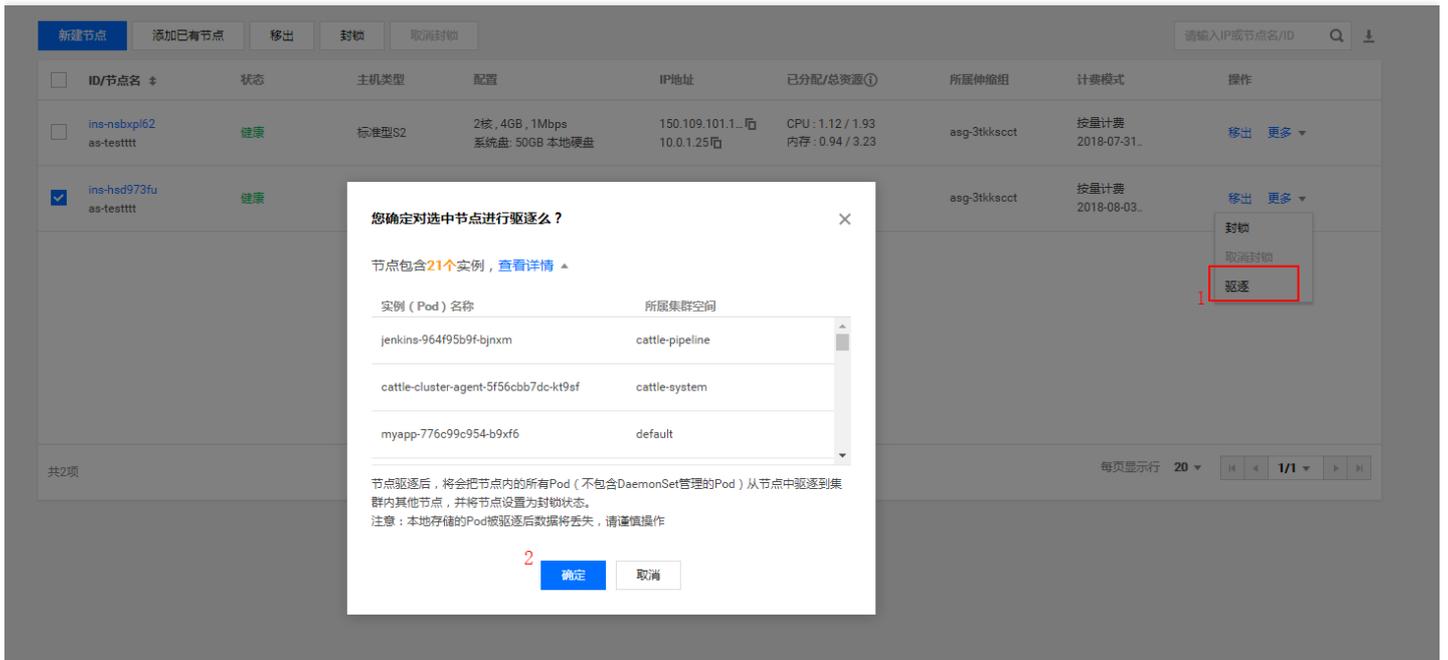
驱逐可以用于在节点上执行维护之前安全地从节点中逐出 Pod，节点驱逐后，将会把节点内的所有 Pod（不包含 DaemonSet 管理的 Pod）从节点中驱逐到集群内其他节点，并将节点设置为封锁状态。对应 kubectl 的 drain 命令。

### 注意：

本地存储的 Pod 被驱逐后数据将丢失，请谨慎操作。

## 操作方法

在节点列表页对需要维护的节点进行驱逐操作。



# 节点的驱逐和封锁

最近更新时间：2018-08-08 20:51:39

## 封锁 ( cordon ) 节点

封锁节点后，将不接受新的 Pod 调度到该节点，需要手动取消封锁的节点。

### 方法一

在新增节点时在高级设置中勾选封锁节点，用于先进行业务所需的初始化操作。

高级设置

自定义数据 ⓘ  
可选，用于启动时配置实例，支持 Shell 格式，原始数据不能超过 16 KB

封锁  开启封锁

封锁节点后，将不接受新的Pod调度到该节点，需要手动取消封锁的节点，或在自定义数据中执行 [取消封锁命令](#)

### 方法二

在节点列表页对选中节点进行封锁

节点列表 Namespace列表 伸缩组列表 集群信息

新建节点 添加已有节点 移出 **封锁** 取消封锁

请输入IP或节点名/ID 🔍 ⬇

ID/节点名	状态	主机类型	配置	IP地址	已分配/总资源	所属伸缩组	计费模式	操作
<input checked="" type="checkbox"/> ins-dj4sz5gk ccs_cls-q0...	健康	标准网络...	1核, 2GB, 1... 系统盘: 50GB ...	150.109... 10.0.0.11	CPU: 0.26 / 0.94 内存: 0.11 / 1.42	-	按量计费 2018-05-2...	移出 更多

## 取消封锁 ( uncordon ) 节点

取消封锁节点后，将允许新的 Pod 调度到该节点。

## 方法一

在新增节点时的脚本中添加取消封锁的命令。

如下，执行完成您的自定义命令后，再执行 `kubectl uncordon` 的命令即可取消封锁节点。

```
#!/bin/sh
# your initialization script
echo "hello world!"

# If you set unschedulable when you create a node,
# after executing your initialization script,
# use the following command to make the node schedulable.
node=`ifconfig eth0 | grep inet | awk '{print $2}' | tr -d "addr:"`
#echo ${node}
kubectl uncordon ${node} --kubeconfig=/root/.kube/config
```

## 方法二

在节点列表页对已封锁的节点进行取消封锁



节点列表    Namespace列表    伸缩组列表    集群信息

新建节点    添加已有节点    移出    封锁    **取消封锁**    请输入IP或节点名/ID    🔍    ⬇

<input type="checkbox"/>	ID/节点名	状态	主机类型	配置	IP地址	已分配/总资源	所属伸缩组	计费模式	操作
<input type="checkbox"/>	ins-dj4sz5gk ccs_cls-q0...	健康	标准网络...	1核, 2GB, 1... 系统盘: 50GB ...	150.109... 10.0.0.11	CPU: 0.26 / 0.94 内存: 0.11 / 1.42	-	按量计费 2018-05-2...	移出 更多

## 驱逐 ( drain ) 节点

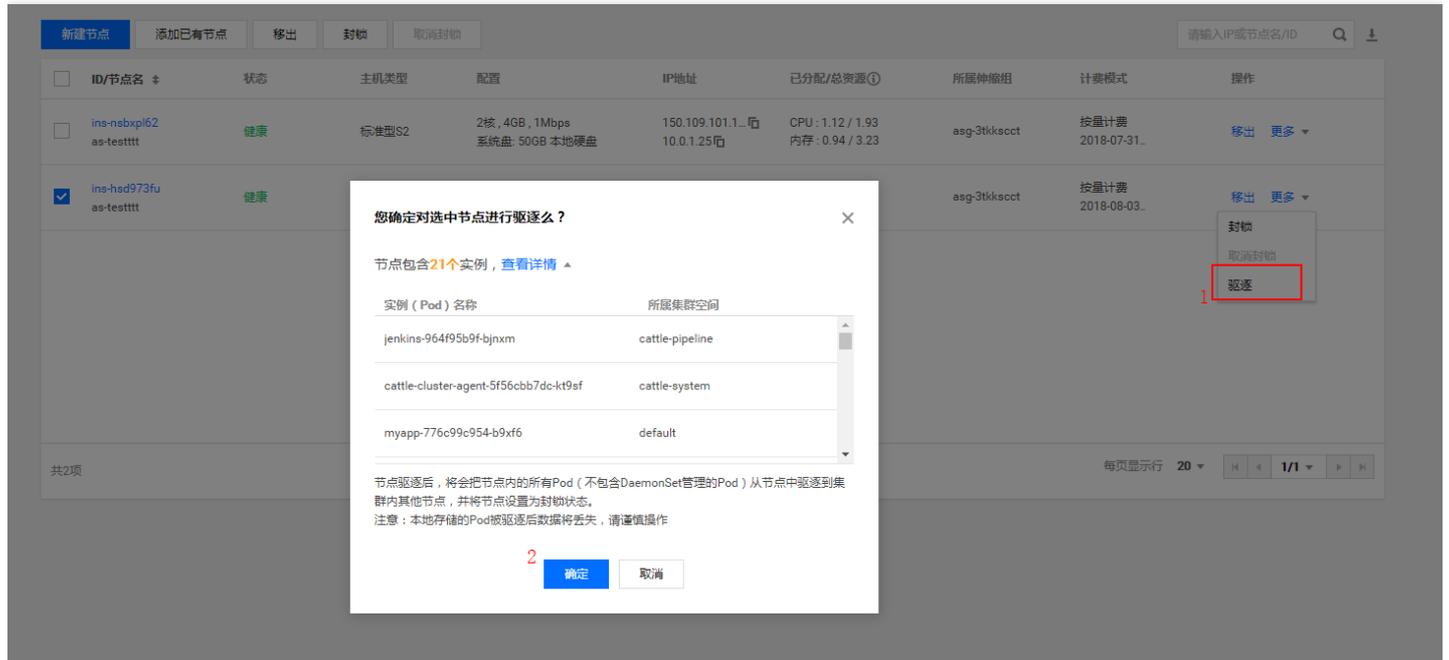
驱逐可以用于在节点上执行维护之前安全地从节点中逐出 Pod，节点驱逐后，将会把节点内的所有 Pod（不包含 DaemonSet 管理的 Pod）从节点中驱逐到集群内其他节点，并将节点设置为封锁状态。对应 `kubectl drain` 命令。

### 注意：

本地存储的 Pod 被驱逐后数据将丢失，请谨慎操作。

## 操作方法

在节点列表页对需要维护的节点进行驱逐操作。



# 升级集群

最近更新时间：2018-09-20 14:50:19

腾讯云容器服务 TKE 提供升级 Kubernetes 版本的功能，通过该功能可以为您运行中的 Kubernetes 集群进行升级。

## 注意事项

1. 升级集群需要先升级Master再升级Node, 您可以通过[提交工单](#)联系我们升级Master版本,当前仅提供Node升级.
2. 升级节点将会驱逐节点上运行的Pod, 请保证集群有足够的资源用于存放被驱逐的Pod, 建议预先新建合适配置的节点。
3. 节点升级将进行重装系统, 请注意提前备份数据。
4. 请在升级集群前查看集群下节点是否均是健康状态, 若节点不正常可以自行修复, 也可以通过提交工单, 腾讯云工程师将协助您进行修复。

## 操作步骤

1. [提交工单](#) 联系我们升级 Master 版本。
2. 确认 Master 版本已升级, 进入集群详情页, 单击版本处 Node 版本升级。
3. 选择需升级的节点(可分批升级, 但最终需要保证集群内所以节点均升级完成), 并填写节点相关配置。
4. 单击完成, 可在节点列表处查看节点升级的情况。

注：

节点会进行滚动升级, 升级完成上一个节点后才会升级下一个节点。

# 集群启用 IPVS

最近更新时间：2018-09-12 18:14:37

默认情况下，Kube-proxy 使用 iptables 来实现 Service 到 Pod 之间的负载均衡。

TKE 支持快速开启基于 IPVS 来承接流量并实现负载均衡的操作。

开启 IPVS 更适用于大规模集群，可提供更好的可扩展性和性能。

## 注意事项

1. 本功能仅在创建集群时开启，暂不支持对存量集群的修改。
2. IPVS 开启针对全集群生效，强烈不建议手动修改集群内 IPVS 和 iptables 混用。
3. 集群开启 IPVS 后不可关闭。
4. IPVS 仅针对 Kubernetes 版本 1.10 及以上的 TKE 集群生效。

## 操作指引

1. 创建集群选择高于 1.10 的 kubernetes 版本。
2. 高级设置开启 IPVS。

所在地域 广州 上海 上海金融 北京 成都 香港 新加坡 孟买 硅谷 重庆

处在不同地域的云产品内网不通，购买后不能更换。建议选择靠近您客户的地域，以降低访问延时、提高下载速度。

集群网络 Default-VPC CIDR: 172.16.0.0/16

如现有的网络不合适，您可以去控制台 [新建私有网络](#)

容器网络 ⓘ

CIDR 172 . 22 . 0 . 0 / 16 [使用指引](#)

Pod数量上限/节点 256

Service数量上限/集群 256

当前容器网络配置下，集群最多255个节点

集群描述

高级设置

ipvs 支持



开启Kube-proxy ipvs支持，注意开启后将不支持关闭，适用于大规模场景下提供更优的转发性能。



# 设置同地域集群间互通

最近更新时间：2018-09-13 17:15:44

您可以通过对等连接实现同地域，不同 VPC 下的间的集群互通。

对等连接（Peering Connection）是一种大带宽、高质量的云上资源互通服务，可以打通腾讯云上的资源通信链路，关于建立对等连接可以参考 [详情](#)。

## 注意：

1. 本文的假设：已经 [创建集群](#) 并已添加节点（单击了解 [创建集群](#)）。
2. 请先确保对等连接间成功建立，子机间能互通，若对等连接建立有问题，请着重排查 [控制台路由表项](#)、[CVM 安全组](#)、[子网 ACL](#) 是否设置有问题。

## 步骤 1

1. 登录腾讯云 [容器服务控制台](#)，单击左侧导航栏中的【集群】。



2. 在集群列表页中单击某集群的【ID/名称】，单击后界面单击红框中【集群信息】。



3. 记录下 A 集群容器网络的网段和掩码。

基本信息	
集群名称	
新增资源所属项目 ⓘ	默认项目 ✎
集群ID	
状态	运行中
k8s版本	1.8.13
节点数量	0个
配置	0核 0GB
所在地域	华北地区(北京)
节点网络	
容器网络	172.31.0.0/16, 256个Service/集群, 256个Pod/节点, 上限255个节点
集群凭证	<a href="#">显示凭证</a>
创建时间	2018-09-03 16:21:07
更新时间	2018-09-03 16:28:04
描述	无 ✎

4. 重复操作上述的操作，记录 B 集群容器网络的网段和掩码。

## 步骤 2

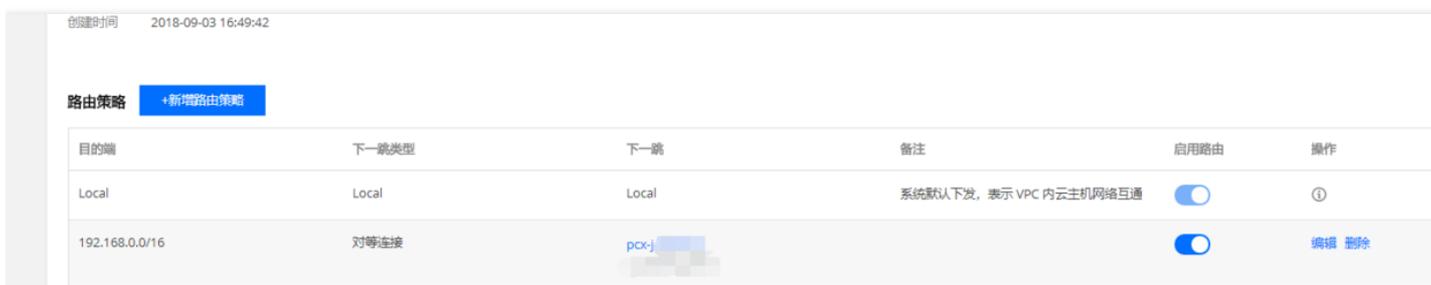
1. 登录腾讯云 [私有网络控制台](#)。
2. 单击左侧目录中的【子网】，进入管理页面。
3. 单击对等连接本端指定子网（子网 A）的关联路由表 ID（路由表 A），进入路由表的详情页。



4. 单击【+ 新增路由策略】。



5. 目的端中填入 B 集群容器的网段 CIDR，下一跳类型选择【对等连接】，下一跳选择已建立的对等连接。



6. 对端路由表配置方法与本端相同。

### 步骤 3

测试容器间是否能互通，可以分别登录集群 A、B 的两个容器（登录方法请单击 [详情](#)）互相访问验证。

容器 A 访问容器 B

```
[root@centos-ssh-8456f58d49-hv9k2 /]# ping 172.31.0.6
PING 172.31.0.6 (172.31.0.6) 56(84) bytes of data.
64 bytes from 172.31.0.6: icmp_seq=1 ttl=62 time=1.47 ms
64 bytes from 172.31.0.6: icmp_seq=2 ttl=62 time=1.29 ms
64 bytes from 172.31.0.6: icmp_seq=3 ttl=62 time=1.44 ms
^^
```

容器 B 访问容器 A

```
[root@centos-d999ccdd6-z42t4 /]# ping 192.168.0.11
PING 192.168.0.11 (192.168.0.11) 56(84) bytes of data.
64 bytes from 192.168.0.11: icmp_seq=1 ttl=62 time=1.40 ms
64 bytes from 192.168.0.11: icmp_seq=2 ttl=62 time=1.36 ms
64 bytes from 192.168.0.11: icmp_seq=3 ttl=62 time=1.41 ms
^^
```

# 设置跨地域集群间互通

最近更新时间：2018-09-13 17:15:49

您可以通过对等连接实现跨地域不同集群互通。

对等连接（Peering Connection）是一种大带宽、高质量的云上资源互通服务，可以打通腾讯云上的资源通信链路，关于建立对等连接可以参考[详情](#)。

## 注意：

1. 本文的假设：已经 **创建集群** 并已添加节点（单击了解 [创建集群](#)）。
2. 请先确保对等连接成功建立，子机间能互通，若对等连接建立有问题，请着重排查 **控制台路由表项、CVM安全组、子网 ACL** 是否设置有问题。

## 步骤 1

1. 登录腾讯云 [容器服务控制台](#)，单击左侧导航栏中的【**集群**】。



2. 在集群列表页中单击某集群的【ID/名称】，单击后界面单击红框中【**集群信息**】。



3. 记录下 A 集群容器网络的地域、VPCID、容器网段和掩码。

The screenshot shows the '集群信息' (Cluster Information) tab in the Tencent Cloud console. The '基本信息' (Basic Information) section contains the following details:

集群名称	middleware_1.8.3
新增资源所属项目	默认项目
集群ID	cls-ldlu58ut
状态	运行中
k8s版本	1.8.13
节点数量	1个
配置	0.94核 0.71GB
所在地域	华北地区(北京)
节点网络	vpc-38ms
容器网络	172.31.0.0/16, 256个Service/集群, 256个Pod/节点, 上限255个节点
集群凭证	<a href="#">显示凭证</a>
创建时间	2018-09-03 16:21:07

4. 重复操作上边的操作，记录 B 集群容器网络的地域、VPCID、容器网段和掩码。

The screenshot shows the '账号信息' (Account Information) section in the Tencent Cloud console. The following details are visible:

账号ID	3206
APPID	1251

5. 登录腾讯云 [私有网络控制台](#)，选择左侧对等连接，记录 **对等连接 ID**。

## 步骤 2

1. 登录腾讯云 [私有网络控制台](#)。
2. 单击左侧目录中的【子网】，进入管理页面。
3. 单击对等连接本端指定子网（子网 A）的关联路由表 ID（路由表 A），进入路由表的详情页。



4. 单击【+ 新增路由策略】。



5. 目的端中填入 B 集群容器的网段 CIDR，下一跳类型选择【对等连接】，下一跳选择已建立的对等连接。



6. 对端路由表配置方法与本端相同。

### 步骤 3

测试容器间是否能互通，可以分别登录集群 A、B 的两个容器（登录方法请单击 [详情](#)）互相访问验证。

上海 pod 访问北京 pod

选中文字进行复制，按下Shift+Insert进行粘贴

```
[root@centos-sh-65d4dc775-csjd5 /]# ping 172.31.2.7
PING 172.31.2.7 (172.31.2.7) 56(84) bytes of data.
64 bytes from 172.31.2.7: icmp_seq=1 ttl=60 time=28.9 ms
64 bytes from 172.31.2.7: icmp_seq=2 ttl=60 time=28.7 ms
64 bytes from 172.31.2.7: icmp_seq=3 ttl=60 time=28.7 ms
64 bytes from 172.31.2.7: icmp_seq=4 ttl=60 time=28.8 ms
64 bytes from 172.31.2.7: icmp_seq=5 ttl=60 time=28.7 ms
^C
--- 172.31.2.7 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 28.706/28.810/28.953/0.202 ms
[root@centos-sh-65d4dc775-csjd5 /]#
```

北京 pod 访问上海 pod

```
[root@centos-bj-bdcd88f45-w9tgz /]# ping 10.110.1.4
PING 10.110.1.4 (10.110.1.4) 56(84) bytes of data.
64 bytes from 10.110.1.4: icmp_seq=1 ttl=60 time=35.0 ms
64 bytes from 10.110.1.4: icmp_seq=2 ttl=60 time=35.0 ms
64 bytes from 10.110.1.4: icmp_seq=3 ttl=60 time=35.0 ms
64 bytes from 10.110.1.4: icmp_seq=4 ttl=60 time=35.0 ms
^C
--- 10.110.1.4 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 35.010/35.045/35.082/0.033 ms
[root@centos-bj-bdcd88f45-w9tgz /]#
```

# 设置容器集群与 IDC 互通

最近更新时间：2018-09-13 17:15:54

目前容器集群与用户 IDC 互通主要通过两种方式：**专线** 和 **IPsec VPN**。

## 注意：

1. 前提：已经 **创建集群** 并已添加节点（单击了解 [创建集群](#)）。
2. 请先确保成功容器服务所在的 VPC 和您 IDC 机房通过专线或 VPN 连接，（单击了解 [VPN 连接](#)）子机间能互通，若通道未连接，可以参考 [详情](#) 排查。

## 专线打通

1. [申请物理专线](#)
2. [申请通道](#)
3. [创建专线网关](#)
4. 验证容器节点与 IDC 互通。（**请保证本步骤验证通过**）
5. 准备地域，applianceId，集群Id，vpcId，专线网关Id 等信息，[提交工单](#) 打通容器网络。
6. 若 IDC 使用的是 BGP 协议，容器网段路由将自动同步。若是其他协议，需在 IDC 内配置访问容器网段下一跳路由到专线网关。
7. 验证容器与 IDC 互通。

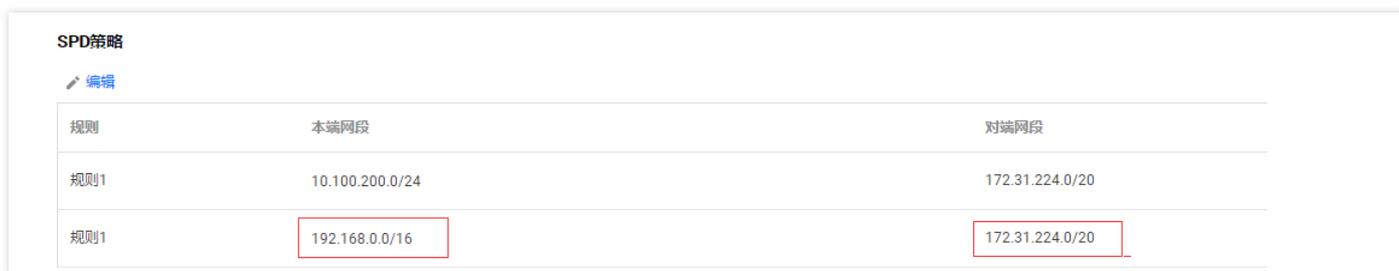
## VPN 打通

### 步骤 1

1. 登录腾讯云 [私有网络控制台](#)，单击左侧导航栏中的【VPN 连接】>【VPN 通道】。



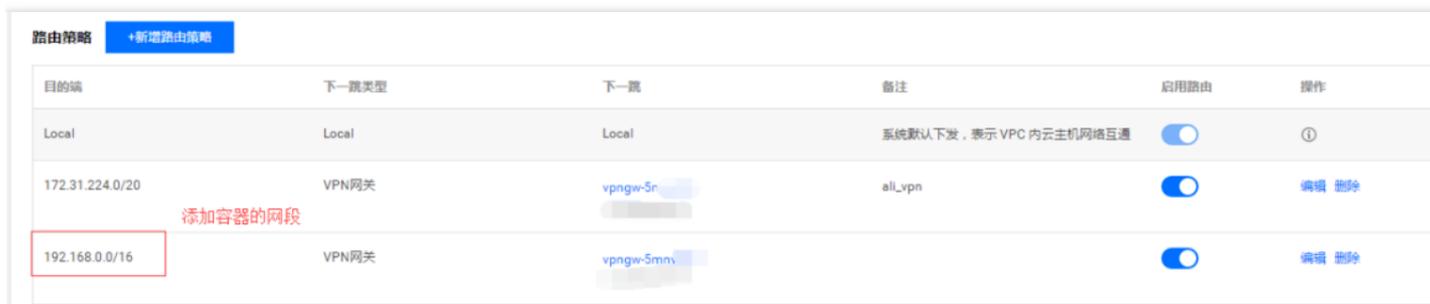
2. 单击【VPN 通道 ID】进入详情页，在【SPD 策略】下方单击编辑，添加容器网段。



同理您对端 VPN 通道也需要 SPD 策略，添加腾讯云容器所在网段。

## 步骤 2

单击左侧导航栏中【路由表】找到之前的 VPN 添加对端子机路由的那张路由表，追加容器网段，并关联子机所在的子网。

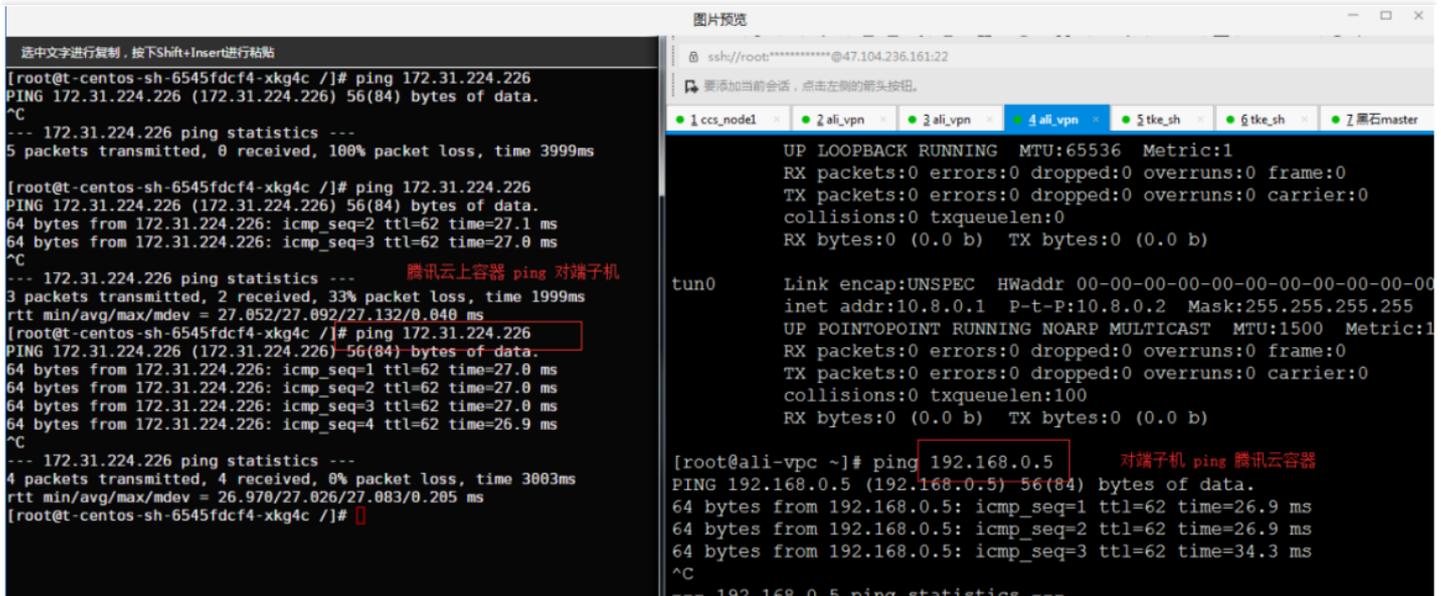


同理您对端的路由设备上，添加腾讯云容器所在网段。

注意：一个子网只能绑定一个路由表，若关联多个路由表，将被替换成最后一个绑定的路由表。

### 步骤3

测试腾讯云一个容器和对端子机是否互通。



可以看到容器间与 VPN 对端子机已经实现互通。

总结：云上容器与 IDC 机房通过 ipsec VPN 互通，设置主要就是 **SPD 策略** 和 **路由表** 两部分配置。

# 设置云主机集群与黑石集群互通

最近更新时间：2018-09-13 17:15:59

您可以通过对等链接实现云服务器集群与黑石集群互通。

对等连接（Peering Connection）是一种大带宽、高质量的云上资源互通服务，可以打通腾讯云上的资源通信链路，本文主要介绍黑石私有网络与公有云私有网络之间建立对等连接，黑石物理机与容器间的互通。

## 注意:

1. 本文的前提：在公有云已经 **创建集群** 并已添加节点（单击了解 [创建集群](#)）。
2. 请先确保对黑石公有云对等连接成功建立，子机与黑石间能互通（单击了解 [对等连接](#)）若对等连接建立有问题，请着重排查 **控制台路由表项、安全组、子网 ACL** 是否设置有误。

## 步骤 1

1. 登录腾讯云 [黑石私有网络控制台](#)，单击左侧导航栏中的【私有网络】记录需要建立黑石 VPC 的 **CIDR**。

ID/名称	状态	CIDR	子网	物理服务器	弹性公网IP	负载均衡
vpc-e2hj6htd cary	运行中	10.0.0.0/16	1	4	17	1

2. 登录腾讯云 [容器服务控制台](#)，单击左侧导航栏中的【集群】。

容器服务	集群	广州(1)	上海(2)	上海金融(0)	北京(5)	成都(1)	香港(0)	新加坡(0)	孟买(0)	硅谷(1)	弗吉尼亚(0)	莫斯科(0)	深圳金融(0)
应用中心	新建												
集群	ID/名称	监控	集群状态	节点状态	节点数量								
应用	cls-ldlu		运行中	-	0台								

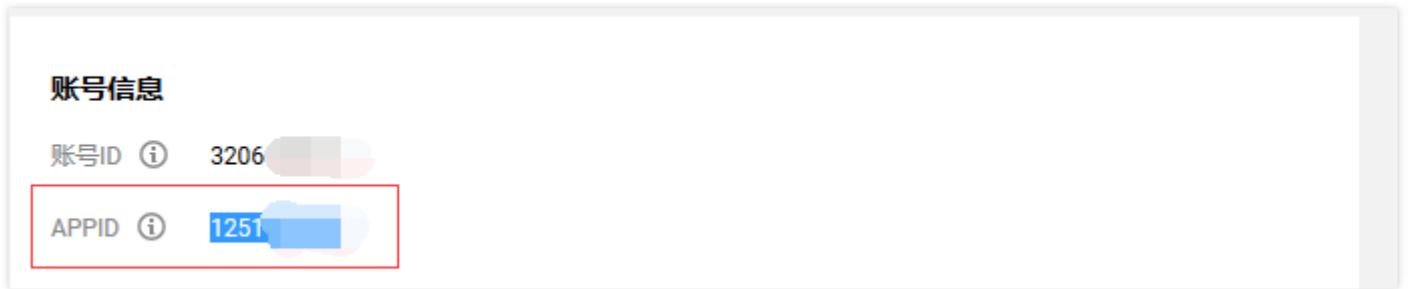
3. 在集群列表页中单击某集群的【ID/名称】，单击后界面单击红框中【集群信息】。



4. 记录下公有云容器集群的 地域、VPCID、容器网段和掩码。



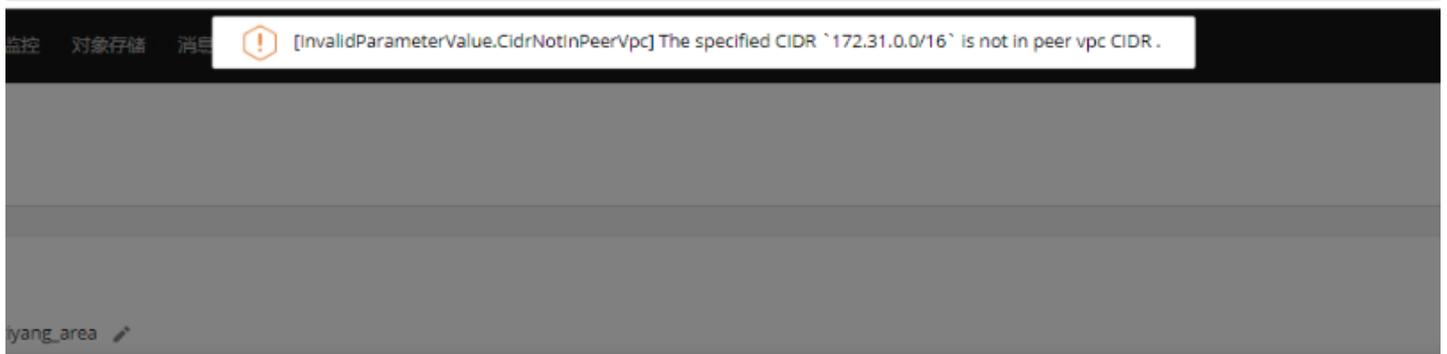
5. 查询当前账户 APPID (单击 [查看](#))。



6. 登录腾讯云 [私有网络控制台](#)，选择左侧对等连接如下图，记录 **对等连接 ID**。

## 步骤 2

目前控制台在对等连接添加路由时，当路由网段非 VPC 的子网网段，会报下图错误：



当前云服务器集群与黑石集群互通需 [提交工单](#),并提供以下信息：

- 需要打通的两个集群的地域、VPCid、容器网段和掩码
- 当前账号 ID

## 步骤 3

测试容器与黑石机器间是否能互通，可以登录公有云集群容器（登录方法请单击 [详情](#)）互相访问验证。  
容器访问黑石物理机

```
0 packets transmitted, 0 received, 100% packet loss, time 0.000ms
[root@t-centos-sh-6545fdcf4-xkg4c /]# ping 10.0.0.2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data:
64 bytes from 10.0.0.2: icmp_seq=1 ttl=59 time=1.14 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=59 time=1.12 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=59 time=1.16 ms
^C
```

黑石物理机访问容器

```
root@10:~# ping 192.168.0.5
PING 192.168.0.5 (192.168.0.5) 56(84) bytes of data:
64 bytes from 192.168.0.5: icmp_seq=1 ttl=251 time=1.23 ms
64 bytes from 192.168.0.5: icmp_seq=2 ttl=251 time=1.22 ms
64 bytes from 192.168.0.5: icmp_seq=3 ttl=251 time=1.11 ms
64 bytes from 192.168.0.5: icmp_seq=4 ttl=251 time=1.14 ms
^C
```

# 使用 Network Policy 进行网络访问控制

最近更新时间：2018-09-13 17:16:03

## Network Policy

**Network Policy** 是 k8s 提供的一种资源，用于定义基于 pod 的网络隔离策略。它描述了一组 pod 是否可以与其它组 pod，以及其它 network endpoints 进行通信。

## Kube-router

- 官网: <https://www.kube-router.io>
- 项目地址: <https://github.com/cloudnativelabs/kube-router>

目前 kube-router 最新版本为 0.2.0

kube-router 的三大功能：

- Pod Networking
- IPVS/LVS based service proxy
- Network Policy Controller

在腾讯云 TKE 上，Pod Networking 的功能由基于 IAAS 层 VPC 的高性能容器网络实现，service proxy 功能由 kube-proxy 所支持的 ipvs/iptables 两种模式来提供。建议在 TKE 上，只使用 kube-router 的 Network Policy 功能。

## 在 TKE 上部署 kube-router

### 腾讯云提供的 kube-router 版本

下面提供的 yaml 文件中使用的 kube-router 镜像 `ccr.ccs.tencentyun.com/library/kube-router:v1` 是由腾讯云 PAAS 团队提供的。这个镜像是基于社区 2018-07-29 的 commit "e2ee6a76"，加上腾讯云 PAAS 团队的两个 bugfix（均已被社区合并）：

- <https://github.com/cloudnativelabs/kube-router/pull/488>
- <https://github.com/cloudnativelabs/kube-router/pull/498>

我们会跟踪社区进展，提供版本升级。

## 部署 kube-router

Daemonset yaml 文件：[#kube-router-firewall-daemonset.yaml.zip#](#)

在 **能访问公网**，也能访问 TKE 集群 apiserver 的机器上，执行以下命令即可完成 kube-router 部署。

如果集群节点开通了公网 IP，则可以直接在集群节点上执行以下命令。

如果集群节点没有开通公网 IP，则可以手动下载和粘贴 yaml 文件内容到节点，保存为 kube-router-firewall-daemonset.yaml，再执行最后的 kubectl create 命令。

```
wget https://ask.qcloudimg.com/draft/982360/90i1a7pucf.zip
unzip 90i1a7pucf.zip
kuebectl create -f kube-router-firewall-daemonset.yaml
```

## yaml 文件内容和参数说明

kube-router-firewall-daemonset.yaml 文件内容：

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: kube-router-cfg
  namespace: kube-system
labels:
  tier: node
  k8s-app: kube-router
data:
  cni-conf.json: |
  {
    "name": "kubernetes",
    "type": "bridge",
    "bridge": "kube-bridge",
    "isDefaultGateway": true,
    "ipam": {
      "type": "host-local"
    }
  }
  ---
apiVersion: extensions/v1beta1
kind: DaemonSet
metadata:
  name: kube-router
  namespace: kube-system
labels:
  k8s-app: kube-router
```

```
spec:
template:
metadata:
labels:
k8s-app: kube-router
annotations:
scheduler.alpha.kubernetes.io/critical-pod: ""
spec:
containers:
- name: kube-router
image: ccr.ccs.tencentyun.com/library/kube-router:v1
args: ["--run-router=false", "--run-firewall=true", "--run-service-proxy=false", "--kubeconfig=/var/lib/kube-router/kubeconfig", "--iptables-sync-period=1s", "--cache-sync-timeout=5m"]
securityContext:
privileged: true
imagePullPolicy: Always
env:
- name: NODE_NAME
valueFrom:
fieldRef:
fieldPath: spec.nodeName
livenessProbe:
httpGet:
path: /healthz
port: 20244
initialDelaySeconds: 10
periodSeconds: 3
volumeMounts:
- name: lib-modules
mountPath: /lib/modules
readOnly: true
- name: cni-conf-dir
mountPath: /etc/cni/net.d
- name: kubeconfig
mountPath: /var/lib/kube-router/kubeconfig
readOnly: true
initContainers:
- name: install-cni
image: busybox
imagePullPolicy: Always
command:
- /bin/sh
- -c
- set -e -x;
if [ ! -f /etc/cni/net.d/10-kuberouter.conf ]; then
```

```
TMP=/etc/cni/net.d/tmp-kuberouter-cfg;
cp /etc/kube-router/cni-conf.json ${TMP};
mv ${TMP} /etc/cni/net.d/10-kuberouter.conf;
fi
volumeMounts:
- name: cni-conf-dir
mountPath: /etc/cni/net.d
- name: kube-router-cfg
mountPath: /etc/kube-router
hostNetwork: true
tolerations:
- key: CriticalAddonsOnly
operator: Exists
- effect: NoSchedule
key: node-role.kubernetes.io/master
operator: Exists
volumes:
- name: lib-modules
hostPath:
path: /lib/modules
- name: cni-conf-dir
hostPath:
path: /etc/cni/net.d
- name: kube-router-cfg
configMap:
name: kube-router-cfg
- name: kubeconfig
hostPath:
path: /root/.kube/config
```

args 说明：

1. "--run-router=false", "--run-firewall=true", "--run-service-proxy=false"：只加载 firewall 模块；
2. kubeconfig：用于指定 master 信息，映射到主机上的 kubectl 配置目录 /root/.kube/config；
3. --iptables-sync-period=1s：指定同步 iptables 规则的间隔时间，根据实时性的要求设置，默认 5 m；
4. --cache-sync-timeout=5m：指定启动时将 k8s 资源做缓存的超时时间，默认 5 m；

## NetworkPolicy 配置示例

1. nsa namespace 下的 pod 可互相访问，而不能被其它任何 pod 访问。

```
apiVersion: extensions/v1beta1
kind: NetworkPolicy
```

```
metadata:  
name: npa  
namespace: nsa  
spec:  
ingress:  
- from:  
- podSelector: {}  
podSelector: {}  
policyTypes:  
- Ingress
```

2. nsa namespace 下的 pod 不能被任何 pod 访问。

```
apiVersion: extensions/v1beta1  
kind: NetworkPolicy  
metadata:  
name: npa  
namespace: nsa  
spec:  
podSelector: {}  
policyTypes:  
- Ingress
```

3. nsa namespace 下的 pod 只在 6379/TCP 端口可以被带有标签 app: nsb 的 namespace 下的 pod 访问，而不能被其它任何 pod 访问。

```
apiVersion: extensions/v1beta1  
kind: NetworkPolicy  
metadata:  
name: npa  
namespace: nsa  
spec:  
ingress:  
- from:  
- namespaceSelector:  
matchLabels:  
app: nsb  
ports:  
- protocol: TCP  
port: 6379  
podSelector: {}
```

**policyTypes:**

- Ingress

4. nsa namespace 下的 pod 可以访问 CIDR 为 14.215.0.0/16 的 network endpoint 的5978/TCP 端口，而不能访问其它任何 network endpoints ( 此方式可以用来为集群内的服务开访问外部 network endpoints 的白名单 )。

**apiVersion: extensions/v1beta1****kind: NetworkPolicy****metadata:****name: npa****namespace: nsa****spec:****egress:**

- to:

- ipBlock:

cidr: 14.215.0.0/16

**ports:**

- protocol: TCP

port: 5978

**podSelector: {}****policyTypes:**

- Egress

5. default namespace 下的 pod 只在 80/TCP 端口可以被 CIDR 为 14.215.0.0/16 的 network endpoint 访问，而不能被其它任何 network endpoints 访问。

**apiVersion: extensions/v1beta1****kind: NetworkPolicy****metadata:****name: npd****namespace: default****spec:****ingress:**

- from:

- ipBlock:

cidr: 14.215.0.0/16

**ports:**

- protocol: TCP

port: 80

**podSelector: {}**

**policyTypes:**

- Ingress

**附: 测试情况**

用例名称	测试结果
不同 namespace 的 pod 互相隔离，同一 namespace 的 pod 互通	通过
不同 namespace 的 pod 互相隔离，同一 namespace 的 pod 隔离	通过
不同 namespace 的 pod 互相隔离，白名单指定 B 可以访问 A	通过
允许某个 namespace 访问集群外某个 CIDR，其他外部 IP 全部隔离	通过
不同 namespace 的 pod 互相隔离，白名单指定 B 可以访问 A 中对应的 pod 以及端口	通过
以上用例，当 source pod 和 destination pod 在一个 node 上时，隔离是否生效	不通过

功能测试用例：

[#kube-router 测试用例.xlsx.zip#](#)

## 性能测试方案

在 k8s 集群中部署大量的 Nginx 服务，通过 ApacheBench 工具压测固定的一个服务，对比开启和不开启 kube-router 场景下的 QPS，衡量 kube-router 带来的性能损耗。

### 测试环境

VM 数量: 100

VM 配置: 2 核 4 G

VM OS: ubuntu

k8s: 1.10.5

kube-router version: 0.2.0

### 测试流程

1. 部署 1 个 service，对应两个 pod（Nginx），作为测试组；
2. 部署 1000 个 service，每个分别对应 2/6/8 个 pod（Nginx），作为干扰组；
3. 部署 NetworkPolicy 规则，使得所有 pod 都被选中，以便产生足够数量的 iptables 规则：

```
apiVersion: extensions/v1beta1
```

```
kind: NetworkPolicy
```

```
metadata:
```

```
name: npd
```

```
namespace: default
```

```
spec:
```

```
ingress:
```

```
- from:
```

```
- ipBlock:
```

```
cidr: 14.215.0.0/16
```

```
ports:
```

```
- protocol: TCP
```

```
port: 9090
```

```
- from:
```

```
- ipBlock:
```

```
cidr: 14.215.0.0/16
```

```
ports:
```

```
- protocol: TCP
```

```
port: 8080
```

```
- from:
```

```
- ipBlock:
```

```
cidr: 14.215.0.0/16
```

```
ports:
```

```
- protocol: TCP
```

```
port: 80
```

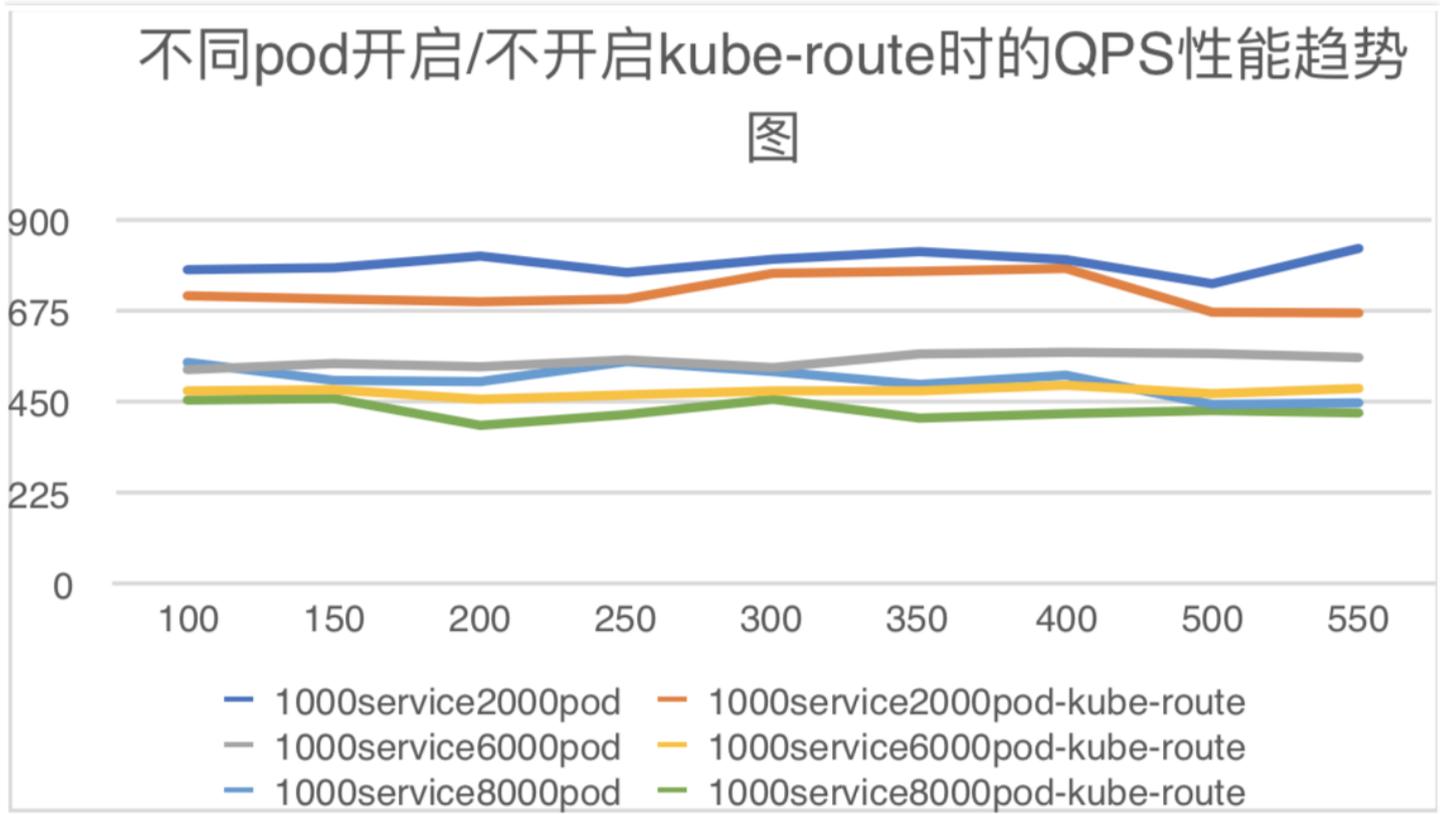
```
podSelector: {}
```

```
policyTypes:
```

```
- Ingress
```

4. 使用 ab 压测测试组的服务，记录 QPS.

## 性能曲线



X轴：ab 并发数

Y轴：QPS

### 测试结论

pod 数量从 2000 到 8000，开启 kube-router 时的性能比不开启时要下降 10%-20%。