阿里云 容器服务Kubernetes版

Serverless Kubernetes集群用户指南

文档版本: 20190828

为了无法计算的价值 | [-] 阿里云

<u>法律声明</u>

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读 或使用本文档,您的阅读或使用行为将被视为对本声明全部内容的认可。

- 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档,且仅能用于自身的合法 合规的业务活动。本文档的内容视为阿里云的保密信息,您应当严格遵守保密义务;未经阿里云 事先书面同意,您不得向任何第三方披露本手册内容或提供给任何第三方使用。
- 未经阿里云事先书面许可,任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分 或全部,不得以任何方式或途径进行传播和宣传。
- 3. 由于产品版本升级、调整或其他原因,本文档内容有可能变更。阿里云保留在没有任何通知或者 提示下对本文档的内容进行修改的权利,并在阿里云授权通道中不时发布更新后的用户文档。您 应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
- 4. 本文档仅作为用户使用阿里云产品及服务的参考性指引,阿里云以产品及服务的"现状"、"有缺陷"和"当前功能"的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引,但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的,阿里云不承担任何法律责任。在任何情况下,阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害,包括用户使用或信赖本文档而遭受的利润损失,承担责任(即使阿里云已被告知该等损失的可能性)。
- 5. 阿里云网站上所有内容,包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计,均由阿里云和/或其关联公司依法拥有其知识产权,包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意,任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外,未经阿里云事先书面同意,任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称(包括但不限于单独为或以组合形式包含"阿里云"、Aliyun"、"万网"等阿里云和/或其关联公司品牌,上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司)。
- 6. 如若发现本文档存在任何错误,请与阿里云取得直接联系。

通用约定

格式	说明	样例
•	该类警示信息将导致系统重大变更甚至 故障,或者导致人身伤害等结果。	禁止: 重置操作将丢失用户配置数据。
A	该类警示信息可能导致系统重大变更甚 至故障,或者导致人身伤害等结果。	▲ 警告: 重启操作将导致业务中断,恢复业务所需 时间约10分钟。
	用于补充说明、最佳实践、窍门等,不 是用户必须了解的内容。	道 说明: 您也可以通过按Ctrl + A选中全部文件。
>	多级菜单递进。	设置 > 网络 > 设置网络类型
粗体	表示按键、菜单、页面名称等UI元素。	单击 确定。
courier 字体	命令。	执行 cd /d C:/windows 命令,进 入Windows系统文件夹。
##	表示参数、变量。	bae log listinstanceid Instance_ID
[]或者[a b]	表示可选项,至多选择一个。	ipconfig[-all -t]
{}或者{a b }	表示必选项,至多选择一个。	<pre>swich {stand slave}</pre>

目录

法律声明	I
通用约定	I
1 概述	1
1 Miles	ייייי ביייי ר
2 Kubernetes 功能又行	3
3 集群管理	5
3.1 创建 Serverless Kubernetes 集群	5
3.2 通过 kubectl 连接 Kubernetes 集群	
3.3 删除集群	9
4 应用管理	11
4.1 通过命令管理应用	11
4.2 使用镜像创建应用	11
4.3 创建服务	15
4.4 删除服务	18
4.5 查看容器	19
4.6 查看服务	20
4.7 基于云解析 PrivateZone 的服务发现	20
5 配置项	23
5.1 创建配置项	23
5.2 删除配置项	24
5.3 修改配置项	25
6 负载均衡管理	27
6.1 通过负载均衡访问服务	27
6.2 通过Ingress提供7层服务访问	30
7 日志管理	37
7.1 概述	
7.2 查看集群日志	
	38
8 更多问题和交流 请扫码加入钉钉交流群	41
○ 人フ [4/☎1 P 人 //ll /P14 = P 4/4H / ★/4 /4 人 //ll H] · · · · · · · · · · · · · · · · · ·	17

1概述

使用 Serverless Kubernetes,您可以专注于设计和构建应用程序,而不是管理运行应用程序的基础设施。

阿里云 Serverless Kubernetes 让您无需管理和维护集群与服务器,即可快速创建 Kuberentes 容器应用,并且根据应用实际使用的 CPU 和内存资源量进行按需付费。使用 Serverless Kubernetes,您可以专注于设计和构建应用程序,而不是管理运行应用程序的基础设施。它基于 阿里云弹性计算基础架构,并且完全兼容 Kuberentes API 的解决方案,充分结合了虚拟化资源带 来的安全性、弹性和 Kubernetes 生态。

它具有以下优势:

・ 简单易用

秒级部署应用。无需管理 Kubernetes 集群基础设施,内建高可用和安全性。

・兼容并包

您可使用熟悉的 Kubernetes 命令行和 API 来部署容器应用,无需修改就支持 Kubernetes 典型应用场景。也可轻松的迁移您的应用到支持 Kubernetes 的运行平台,如阿里云 Kubernetes 集群。

・安全隔离

基于阿里云弹性计算基础架构,应用容器之间相互隔离防止互相干扰。

・按需扩容

无需再担心集群节点的的水平扩展,根据应用负载,轻松灵活扩容应用所需资源。

・互联互通

支持容器应用与阿里云基础服务无缝整合。支持与用户 VPC 中现有应用、数据库直接交互;支持容器与虚拟机应用的互联互通。

公测开放地域

目前,阿里云容器服务 Serverless Kubernetes 集群处于公测阶段。暂只开放部分地域,其他地域将相继开放。

使用限制

集群中pod基于ECI创建,关于pod的规格和使用限制请参考弹性容器实例ECI > 产品简介 > 使用 限制页面中的使用限制节点。

产品定价

关于ECI的定价请参考弹性容器实例ECI > 产品定价下的计费说明

集群中SLB、PrivateZone等资源价格请参考相应云产品收费标准。

Serverless Kubernetes不会收费,将持续为用户提供免费服务。

与容器服务的对比

应用场景

serverless kubernetes 应用场景非常广泛,基于容器的高移植性高灵活性,和 IaaS 的高弹性高 隔离性,为 serverless 类应用提供强大的基础运行设施。适合的 serverless 应用包括多媒体处 理、捕获数据修改、物联网传感器数据处理、流式计算、聊天机器人、批量计算、web 应用、移动 应用后端服务、商业逻辑处理、持续集成。

2 Kubernetes 功能支持

API 版本

支持 Kubernetes 1.9 API。

应用负载

- · 支持Deployment、StatefulSet、Job/CronJob、Bare Pod。
- · 不支持 DaemonSet。

Pod 定义

支持启动多个容器,设置环境变量,设置 RestartPolicy,设置健康检查命令,挂载volumes 等。

负载均衡

- · 支持创建 LoadBalancer 类型应用。
- ・支持Ingress。
- · 不支持 NodePort 类型。

配置

支持 Secret 和 Configmap。

存储

- ・ 支持 emptyDir 和 nfs volume 类型。
- · 不支持 PersistentVolume 和 PersistentVolumeClaim。

命名空间

用户只可看见 default 命名空间,不可添加命名空间。

节点

用户不可查看 kubernetes 的节点信息。

事件

用户可查看 default 命名空间下的事件。

容器logs

用户可通过kubectl logs实时查看容器的日志。

容器exec/attach

用户可通过kubectl exec进入容器执行命令。

3 集群管理

3.1 创建 Serverless Kubernetes 集群

您可以通过容器服务管理控制台非常方便地快速创建 Serverless Kubernetes 集群。

前提条件

登录 容器服务管理控制台 和 RAM 管理控制台 开通相应的服务。

操作步骤

- 1. 登录容器服务管理控制台。
- 2. 在Kubernetes 菜单下,单击左侧导航栏的集群 > 集群,进入集群列表页面。
- 3. 单击页面右上角的创建 Kubernetes 集群, 在弹出的选择集群模板中, 选择标准 Serverless 集群页面, 单击创建。

= (-) 阿里云	账号全部资源 - (3) 全球	Q 搜测	5		第月 工単 留室 企业 支持与援(☆ 前休中文 (
容器服务 - Kubernetes *	集群列表	达掉集研模板				^	多配额遗提交工单 用新	创建 Kubernetes 編訂
極原		托管集群						
▼ 供計	名称 *	(意) 标准托管集群	🍥 GPU 托營集群	(神龙) 集群	(會) 基因计算托管集群	(公園) Windows 集群 (公園)		
91.27	集群翁称/ID	创建一个标准的 Kubernetes 托普版集 群。忽只器创建工作节点即可开展业 冬。管理节点完全由 ACK 托管、为促	创建一个以 GPU 型实例为工作节点的 集群,适合AI、深度学习、图形渲染 每计算型业务	创建一个以弹性裸金属(神龙)实例 为工作节点的集群,神龙服务器为用 户提供胡豪性能容器本例,活合豪岛	创建一个以高性能计算型实例为工作 节点和具备大型工作流引擎的基因计 算加速集群,适合BCL_FASTO。	提供一个可以部署 Windows 容器应用 的集群,甚至可以加入 Linux 节点和 Windows 节点协同工作		操作
节点	k8s-duster	节音资源和运输的成本		動巡带宽震求的业务场景	BAM/SAM/VCF数据分析,装配,变异 检出等工作流计算型业务		管理	1 査習日志 控制台 度現状な 更多。
存储卷		≢ 创建	≓ €182	🛒 काश्च	🛫 estate	🛒 कारहो	-	1000 W 1 000
命名空间	managed-cluster c03ffu1	其他举型					管理	査習日志 控制台 集群扩容 更多▼
授权管理	k8s-test	1 Million					首理	査署日志 控制台
▼ 应用	c65 c	(章)标准专有集群	🎯 GPU 专有集群	(章) 基因计算专有集群	👔 FPGA 专有集群	(意) 标准 Serverless 集群		集群扩容 更多。
无状态	k8s-cluster-serverless c7712	创建一个标准的 Kubernetes 专有版集 群。您可以完整她创建集群管理节点 和工作节点,对整个集群享有完全控	创建一个以 GPU 型实例为工作节点的 集群,适合AI、深度学习、图形渲染 每计算型业务	创建一个以 离性能计算型实例为工作 节点和具备大型工作流引擎的基因计 算加速集群, 适合BCL, FASTQ,	创建一个以 FPGA 型实例为工作节点 的视频图像加速集群,适合视频编码 H265、图片转码JPEG-to-HEIF和	创建一个标准的 Serverless Kubernetes 集群,您无案创建任何节 点即可开始部署容置实例。容置实例	管理	査吾日本 删除 通过 CoudShell 管理集群
有状态	managed-k8s	制能力		BAM/SAM/VCF数据分拆,装配,变异 检出等工作流计算型业务	JPEG-to-webp瞭计算型业务	只在运行时按量收费,适合开发测 试、单次和批量任务、突发流量等场	管理	· 音雅日志 · 神制台
守护进程集	cd82	ゴ 创建	ઝ 6138	ゴ 句録	2 608	(2) (1) (1) (1) (1) (1) (1) (1) (1) (1) (1		東部扩容 更多。
任务								

默认进入 Serverless Kubernetes 集群配置页面。

创建 Kubernetes 集群							
Kubernetes	Kubernetes 托管版	多可用区 Kubernetes	Serverless Kubernetes (公测)				
* 集群名称	serverless 名称为1-63个字符,	可包含数字、汉字、英文字	符,或"-"				

4. 填写集群的名称。

集群名称应包含 1-63 个字符,可包含数字、汉字、英文字符或连字符(-)。

5. 选择集群所在的资源组。

账号全部资源 🔺 🚱 全部	求	
账号全部资源 默认资源组	2回集群列表	
测试环境	版 多可用区 Kubernetes	Serverless Kubernetes (公测)
管理资源组		
名称为1-6	3个字符,可包含数字、汉字、英文字	符,或"-"
资源组 未选择		

6. 选择集群所在的地域和可用区。

地域	华北 2 (北京)	华东 1 (杭州)	华东 2 (上海)	华南 1 (深圳)	美国西部1(硅谷)
可用区	华北 2 可用区 G		•		

7. 设置集群的网络。

Kubernetes 集群仅支持专有网络。支持自动创建和使用已有的 VPC。

专有网络	自动创建	使用已有		
	vpc-k8s-for-cs-		 (vsw-bp1	

说明: ・ 自动创建:集群会自动新建一个VPC,并在VPC中自动创建NAT网关以及配置SNAT规则。 ・ 使用已有:您可以在已有 VPC 列表中选择所需的 VPC 和交换机。如需访问公网,比如下载

- 容器镜像,要配置NAT网关,建议将容器镜像上传到集群所在区域的阿里云镜像服务,并通 过内网VPC地址拉取镜像。
- 8. 设置是否为专有网络创建 NAT网关 并配置 SNAT 规则。



若您选择自动创建 VPC 时必须配置 SNAT;若您选择使用已有VPC,可选择是否自动配置SNAT网关。若选择不自动配置 SNAT,您可自行配置NAT 网关实现 VPC 安全访问公网环境;或者手动配置 SNAT,否则 VPC 内实例将不能正常访问公网,会导致集群创建失败。

9. 勾选是否开启基于PrivateZone的服务发现功能,该功能允许您在集群VPC内通过服务域名访问服务。

PrivateZone	✓ 开启基于DNS的服务发现机制,可以通过域名直接访问服务;需要开启Private Zone服务(收费服务)
	请确认您已开通 PrivateZone服务
~	
∐ ÿⅢ·	

在使用勾选该功能前,请确认您已开通PrivateZone服务,参见#unique_7。

10.为集群绑定标签。

输入键和对应的值,单击添加。

标签	: 添加
	标签由区分大小写的键值对组成,您最多可以设置20个标签。 标签键不可以重复,最长为64位:标签值可以为空,最长为128位。标签键和标签值都不能以"alivun"、"acs;"、"https://"或"http://"开头。
ren "	4 m
U II	2明:
・键是	必需的,而 值 是可选的,可以不填写。
・键不	能是 aliyun、http:// 、https:// 开头的字符串,不区分大小写,最多 64 个字符。
・値不	能是 http:// 或 https://,可以为空,不区分大小写,最多 128 个字符。
・同一	个资源,标签键不能重复,相同标签键(Key)的标签会被覆盖。
・如果-	一个资源已经绑定了 20 个标签,已有标签和新建标签会失效,您需要解绑部分标签后
才能再	再绑定新的标签。
勾选无服	务器 Kubernetes 服务协议。

12.在页面右侧,单击创建集群,启动部署。

后续步骤

集群创建成功后,您可以在容器服务管理控制台的 Kubernetes 集群列表页面查看所创建的 Serverless 集群。

集群列表						您最多可以创建 100) 个集群,每个集群。	修可以添加 1000	个节点,申请更多配额请提交 <u>工</u> 单	周新	创建 Kubernetes 集群
♂如何创建集群 ♂ 创建 GPU 集群 ♂ 扩育	容和缩容集群	Ø 通过 kubectl 连接	ē Kubernetes 集群 🔗 通过命令管理)	应用 🔗 VPC 下 Kubernet	aes 的网络地址段规划 🥝 集群)	为什么创建失败 🔗 授权管理	🖉 收集 Kubernete:	;诊断信息 8 提	交工单		
名称 ▼	标签										
集群名称/ID		标签	集群类型	地域 (全部) 👻	网络类型	集群状态	节点个数	创建时间	版本		摄作
serverless cbf920af		۲	Serverless Kubernetes	华北2		●运行中		2019-06-21 11:10:28		管理	查看日志 删除 通过 CloudShell 管理集群

您可以单击右侧的管理,查看集群的基本信息和连接信息。

集群:serverless			
容器服务正在逐步优化安全策略,禁止未授权的子账号访问集群资源, 请立即升级当前集群授权信息			
基本信息			
集群ID: cbf92(虚拟专有网络	●运行中	地域: 华北2
集群信息			
API Server 公网连接端点	-a.cn-be	ijing.aliyuncs.com:6443	
集群资源			
虚拟专有网络 VPC	vpc-		
虚拟专有网络 VSwitch	VSW-		
安全组	sg-		

3.2 通过 kubectl 连接 Kubernetes 集群

如果您需要从客户端计算机连接到 Kubernetes 集群,请使用 Kubernetes 命令行客户端 kubectl。

操作步骤

- 1. 从Kubernetes 版本页面下载最新的 kubectl 客户端。
- 2. 安装和设置 kubectl 客户端。

有关详细信息,参见安装和设置 kubectl。

3. 配置集群凭据。

您可以在集群信息页面查看集群凭据。

- a) 登录容器服务管理控制台。
- b) 在 Kubernetes菜单下,单击集群进入 Kubernetes 集群列表页面。
- c) 选择所需的集群并单击右侧的管理。
- d) 您可以在连接信息处查看集群的连接地址。

集群 : serverless				
基本信息				
集群ID:	虚拟专有网络	●运行中	地域: 华东2	
连接信息				
API Server连接端点	(the second s	:6443	
集群资源				
虚拟专有网络 VPC	vpc			
虚拟专有网络 VSwitch	VSN			
安全组	SG-I			
通过 kubectl 连接 Kubernetes 集群				
1. 从 Kubernetes 版本页面 下载最新的 kubectl 客户端。				
2. 安装和设置 kubectl 客户端。有关详细信息,参见 安装和设置 k	ubectl			
2 配票使联任据,终于面内应有制到大机和UOME/ Jurba/config				
3. RUELSCRTPUNE, 17 IVALISTER SERIES 44/(QTIOMC). KUDE/CONIG				
api∛ersion: v1 clusters:				复制

e) 拷贝集群凭据到本地文件中,您可创建并将集群凭据保存到 \$HOME/.kube/config

(kubectl 预期凭据所在的位置)。或者命名一个新的文件,如/tmp/kubeconfig,并执行命令 export KUBECONFIG=/tmp/kubeconfig。

f)执行上述操作后,您可执行以下命令,确认集群连接情况。

#kubectl get pod
No resources found.

后续步骤

配置完成后,您即可使用 kubectl 从本地计算机访问 Kubernetes 集群。

3.3 删除集群

您可以通过容器服务管理控制台删除不再使用的集群。

操作步骤

- 1. 登录容器服务管理控制台。
- 2. 在 Kubernetes 菜单下,单击左侧导航栏中的集群,进入 Kubernetes 集群列表页面。

3. 选择所需的集群并单击右侧的删除。

集群列表				您最多	多可以创建 5 个集群,每个	集群最多可以添加 40 个节点	刷新	创建 Kubernetes 集群
常见问题: & 如何创建集群 & 扩容和缩	容集群 🛛 ⑧ 通过 kubectl 连	接 Kubernetes 集群	🔗 通过命令管理应用					
名称 ▼								
集群名称/ID	集群类型	地域 (全部) 👻	网络类型	集群状态	创建时间	Kubernetes 版本		操作
serverless-test	Serverless Kubernetes	华东1	虚拟专有网络 vpc-bp1jijc9lic	●运行中	2018-09-06 09:51:48	1.9.7	管理	查看日志 删除

4. 在弹出的确认对话框中,单击确定。

4 应用管理

4.1 通过命令管理应用

您可以通过命令创建应用或者查看应用的容器。

前提条件

在本地使用命令前,您需要先设置#unique_12。

通过命令创建应用

可以通过运行以下语句来运行简单的容器(本示例中为 Nginx Web 服务器):

root@master # kubectl run nginx --image=registry.cn-hangzhou.aliyuncs. com/spacexnice/netdia:latest

此命令将为该容器创建一个服务入口,指定 --type=LoadBalancer 将会为您创建一个阿里云负

载均衡路由到该 Nginx 容器。

root@master # kubectl expose deployment nginx --port=80 --target-port= 80 --type=LoadBalancer

通过命令查看容器

运行如下命令列出所有 default 命名空间里正在运行的容器。

root@master # kubectl get pods NAME ACE	READY	STATUS	RESTARTS
nginx-2721357637-dvwq3 9h	1/1	Running	1

4.2 使用镜像创建应用

前提条件

创建一个 Serverless Kubernetes 集群。详情参见#unique_14。

操作步骤

- 1. 登录容器服务管理控制台。
- 2. 在 Kubernetes菜单下,单击左侧导航栏中的应用 > 无状态,进入无状态(Deployment)页面。

3. 单击页面右上角的使用镜像创建。

容器服务 - Kubernetes •	无状态(Deployment)					刷新	使用镜像创建	使用模板创建
▲ 授权管理	常见问题: 🖉 如何支持私有镜像 🔗 🕯	別建应用 🥝 指定节:	点调度 🔗 创建4层路日	由服务 🔗 创建7层路	由服务 🔗 设置 Pod 自动伸缩	◎ 蓝绿发布	2	
▼ 应 用	集群 serverless v 命名S	间 default	Ŧ				输入名称查询	Q
无状态	容器服务正在逐步优化安全策率如无法访问相关资源,请联系部	8,禁止未授权的子账 E账号使用"授权管理"	号访问集群资源。 功能完成集群资源授权。	请参考:帮助文档				
有状态	名称	标签	容器组数量	镜像	创建时间			操作
守护进程集								
任务								
定时任务								

4. 设置应用名称、部署集群、命名空间副本数量类型标签和注解,并单击下一步进入应用配置页面。

如果您不设置命名空间,系统会默认使用 default 命名空间。

创建应用						
	应用基本信息	容器配置	\rangle	高级配置	>	创建完成
应用名称	serviceless-app 名称为1-64个字符,可包含数字、小写英文字	·符,或"-",且不能以-开头				
部署集群	serverless	Y				
命名空间	default	¥				
副本数量	2					
类型	无状态	Ŧ				
标签	◎ 添加					
注解	● 添加					
						返回 下—步

- 5. 选择所要使用的镜像和镜像的版本。
 - · 镜像名称:您可以单击选择镜像,在弹出的对话框中选择所需的镜像并单击确定。您还可以 填写私有 registry。填写的格式为 domainname/namespace/imagename:tag。本例中 为 nginx。
 - · 镜像版本:您可以单击选择镜像版本选择镜像的版本。若不指定,默认为 latest。
- 6. 设置容器数量。

本示例是单容器 Pod,若指定多个容器,会启动相同数量的 Pod。

7. 设置容器的资源限制和所需资源。

Serverless Kubernetes 目前处于公测阶段, 仅支持 2C4G 规格。

- ・资源限制:可指定该应用所能使用的资源上限,包括 CPU 和内存两种资源,防止占用过多资源。
- ·所需资源:即为该应用预留资源额度,包括 CPU 和内存两种资源,即容器独占该资源,防止 因资源不足而被其他服务或进程争抢资源,导致应用不可用。
- 其中,CPU 资源的单位为 millicores,即一个核的千分之一;内存的单位为 Bytes,可以为 Gi 、Mi 或 Ki。
- 8. 配置环境变量。

支持通过键值对的形式为 Pod 配置环境变量。用于给 Pod 添加环境标志或传递配置等,具体请 参见 Pod variable。

9. 配置容器信息。

您可以为运行在 Pod 中的容器配置 Command、Arguments。

Command 和 Args:若不配置,会使用镜像默认的设置;若配置,则会覆盖镜像里的默认设置,并且如果只配置 Arguments,容器启动时,默认的 Command 会执行新的 Arguments。

Command 和 Arguments 在 Pod 创建之后不可更改。

10.完成应用配置后,单击下一步,进入访问设置页面,即设置一个绑定后端 pod 的服务。

您可以选择不创建服务,或者选择服务类型,目前只支持负载均衡类型。

- ・ 负载均衡:即 LoadBalancer,是阿里云提供的负载均衡服务,可选择公网访问或内网访问。
- · 名称:默认会生成一个应用名后缀 svc 的服务名称,本例中为 serverless-app-svc,您也可更改服务的名称。
- ・端口映射:您需要添加服务端口和容器端口,支持 TCP/UDP 协议。

创建服务		\times
名称:	serverless-app-svc	
类型:	负载均衡 ▼ 公网访问 ▼	
端囗映射:	 ◆添加 服务端口 容器端口 协议 80 TCP ▼ ● 	
注解:	○ 添加 负载均衡配置参数	
标签:	⊙ 添加	
	创建 耳	刘治

11.完成访问配置后,单击创建。

后续步骤

创建成功后,进入创建完成页面,会列出应用包含的对象,您可以前往部署列表查看。

您可以看到新建的 serverless-app-svc 出现在部署列表下。

部署列表				使用镜像创建	刷新
集群 serverless 💌 命名空间 default 💌					
名称	标签	容器组数量	创建时间		操作
serverless-app-deployment	app:serverless-app	0/1	2018-05-01 23:49:45	详情 更新	斤 删除

单击左侧导航栏的路由与负载均衡 > 服务 ,可以看到新建的服务 serverless-app-svc 出现在服务 列表下。

服务 (Service)						创建 刷新
集群 serverless 💌	命名空间 default 💌					
名称	类型	创建时间	集群IP	内部端点	外部端点	操作
serverless-app-svc	LoadBalancer	2018-05-01 23:49:45	172.21.3.78	serverless-aj serverless-aj	139.	更新 丨 删除

在浏览器中访问外部端点,您可访问 nginx 欢迎页。

4.3 创建服务

Kubernetes Service 定义了这样一种抽象: 一个 Pod 的逻辑分组,一种可以访问它们的策略,通 常称为微服务。这一组 Pod 能够被 Service 访问到,通常是通过 Label Selector 来实现。

在 Kubernetes 中, pod 虽然拥有独立的 IP, 但 pod 会快速地创建和删除,因此,通过 pod 直接对外界提供服务不符合高可用的设计准则。通过 service 这个抽象,Service 能够解耦 frontend(前端)和 backend(后端)的关联,frontend 不用关心 backend 的具体实现,从 而实现松耦合的微服务设计。

更多详细的原理,请参见Kubernetes service。

前提条件

您已经成功创建一个 Serverless Kubernetes 集群,参见#unique_14。

步骤1创建 deployment

使用镜像创建一个 deployment,本例中创建 serverless-app-deployment。具体操作参见#unique_16。

步骤2创建服务

- 1. 登录 容器服务管理控制台。
- 2. 在 Kubernetes 菜单下,单击左侧导航栏中的应用 > 服务,进入服务列表页面。

3. 选择所需的集群和命名空间,单击页面右上角的创建。

容器服务 - Kubernetes ▼	服务列表				刷新创建
概览	集群 serverless v 命名空间	default 🔻			4
▼ 集群	名称 类型	创建时间	集群IP 内部端点	外部端点	操作
集群					
节点					
存储卷					
命名空间					
▼应用 1					
部署					
容器组					
服务 2 프					

4. 在弹出的创建服务对话框中,进行配置。

创建服务		\times
名称:	serverless-service	
类型:	负载均衡 ▼ 公网访问 ▼	
关联部署:	serverless-app-deploymen 🔻	
端口映射:	●添加	
	服务端口 容器端口 协议	
	80 8080 TCP • •	
注解:	还加 奇裁均衡配置条数	
	名称 值	
	service.beta.kubernetes.ic 20	
标签:	⊙ 添加	
	名称 值	
	app nginx 🗢	
	创建 取	消

- · 名称: 输入服务的名称, 本例中为 serverless-service。
- ・ 类型:选择服务类型,即服务访问的方式。目前仅支持负载均衡,即 LoadBalancer,指阿 里云提供的负载均衡服务 (SLB),可选择公网访问或内网访问。
- · 关联部署:选择服务要绑定的后端对象,本例中使用 serverless-app-deployment。若不 进行关联部署,则不会创建相关的 Endpoints 对象,您可自己进行绑定,参见 serviceswithout-selectors。

- ・端口映射:添加服务端口和容器端口,容器端口需要与后端的 pod 中暴露的容器端口一致。
- 注解:为该服务添加一个注解(annotation),配置负载均衡的参数,例如设置service.
 beta.kubernetes.io/alicloud-loadbalancer-bandwidth:20表示将该服务的带宽
 峰值设置为20Mbit/s,从而控制服务的流量。更多参数请参见#unique_17。
- ·标签:您可为该服务添加一个标签,标识该服务。
- 5. 单击创建, serverless-service 服务出现在服务列表中。

容器服务 - Kubernetes ▼	服务列表						周	新创建
概选	集群 serverless ▼	命名空间 defa	ult 🔻 3					
▼ 集群	名称	美型	创建时间	集群IP	内部端点	外部端点		操作
集群	serverless-service	LoadBalancer	2018-07-17 16:56:23		serverless-service:80 TCP serverless-service:31306 TCP	1:80	详情 更新 查看	YAML 删除
节点								
存储卷								
命名空间								
<a>应用								
部署								
容器组								
服务 2 =								

6. 您可查看服务的基本信息,在浏览器中访问 serverless-service 的外部端点。

C 1 2/?spm=5176.2020520152.0.0.49fd16ddLtgTUa	
	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.

至此,您完成如何创建一个关联到后端的 deployment 的服务,最后成功访问 Nginx 的欢迎页面。

4.4 删除服务

您可以通过容器服务控制台快速对服务进行删除。

前提条件

- · 您已经成功创建一个 Serverless Kubernetes 集群,参见#unique_14。
- ・您已经成功创建一个服务,参见#unique_19。

操作步骤

- 1. 登录容器服务管理控制台。
- 2. 在 Kubernetes 菜单下,单击左侧导航栏中的应用 > 服务,进入服务列表页面。

3. 选择集群和命名空间,选择所需的服务(本示例中选择 serverless-service),单击右侧的删

除。

服务列表						刷新 创建
集群 serverless ▼	命名空间 default ▼					
名称	类型	创建时间	集群IP	内部端点	外部端点	撮作
serverless-service	LoadBalancer	2018-08-13 14:09:03		serverless-service:80 TCP serverless-service:32685 TCP	:80	详情 更新 查看YAML 删除

4. 在弹出的窗口中,单击确定,确认删除,该服务在服务列表中消失。

4.5 查看容器

您可以通过容器服务管理控制台的容器组页面查看 Serverless Kubernetes 集群的容器组页面。

操作步骤

- 1. 登录容器服务管理控制台。
- 2. 在Kubernetes 菜单下,单击左侧导航栏中的应用 > 容器组,进入容器组页面。
- 3. 选择所需的集群和命名空间,选择所需的容器组,单击右侧的详情。

▋ 说明:

您可对容器组进行更新和删除操作,对于通过部署(deployment)创建的容器组,建议您通过 deployment 进行管理。

容器组					刷新
集群 serverless 🔻 命名空间 default 💌					
名称	状态	Pod IP	节点	创建时间	
serverless-app-deployment-6d4;	●等待中	172	192.	2018-05-01 23:49:45	详情删除

4. 进入容器组的详情页,您可查看该容器组的详情信息。

容器组 - serverless-a			刷新
基本信息			
名称: serverless-a	an an 10 m a sha	命名空间: default	
状态: Pending		创建时间: 2018-05-02 00:44:34	
节点: viking-cb;	and a second	Pod IP: 17	
标签: app: serverless-app	pod-template-hash: 2803040087		
容器 事件 创建者	初始化容器 存储 日志		
名称	镜像		端口
serverless-app	nginx:latest		TCP 80

4.6 查看服务

您在创建应用时,如果配置了外部服务,除了运行容器,会创建外部 Service,用于预配负载均衡器,以便将流量引入到集群中的容器。

操作步骤

- 1. 登录容器服务管理控制台。
- 2. 在Kubernetes 菜单下,单击左侧导航栏中的应用 > 服务,进入服务列表页面。

3. 您可以选择所需的集群和命名空间, 查看部署的服务。

您可查看服务的名称、类型、创建时间、集群 IP 以及外部端点等信息。本例中您可看到分配给 服务的外部端点(IP 地址)。

服务列表						刷新 创建
集群 serverless v a	命名空间 default 🔻					
名称	类型	创建时间	集群IP	内部端点	外部端点	操作
serverless-service	LoadBalancer	2018-08-13 14:09:03		serverless-service:80 TCP serverless-service:32685 TCP		详情 更新 查看YAML 删除

4.7 基于云解析 PrivateZone 的服务发现

阿里云Serverless Kubernetes已经支持服务发现功能,目前支持内网SLB和Headless Service的服务发现。

关于云解析PrivateZone

云解析 PrivateZone,是基于阿里云专有网络VPC(Virtual Private Cloud)环境的私有域名解 析和管理服务。您能够在自定义的一个或多个专有网络中将私有域名映射到IP资源地址,同时在其 他网络环境无法访问您的私有域名。



说明:

PrivateZone的收费规则参见收费标准。

前提条件

- 1. 需要先开通云解析 PrivateZone,在云解析DNS控制台中开通。
- 2. 您已成功创建一个 Serverless Kubernetes 集群,参见#unique_23。
- 3. 您已成功连接到 Kubernetes 集群,参见#unique_24。

操作步骤

1. 通过 kubectl 连接到 Kubernetes 集群,执行如下命令,确认连接的集群。

kubectl cluster-info

Kubernetes master is running at https://xxxxxx.serverless-1. kubernetes.cn-shanghai.aliyuncs.com:6443

2. 部署 deployment 和创建 Service。目前仅支持 Intranet Service 和 Headless Service。

以Intranet Service 为例。创建示例nginx-deployment-basic.yaml文件。

```
vim nginx-deployment-basic.yaml
```

样例模板如下所示,在yaml文件中复制如下yaml代码,然后执行 kubectl create -f

nginx-deployment-basic.yaml命令进行创建。

```
apiVersion: apps/v1beta2 # for versions before 1.8.0 use apps/
v1beta1
kind: Deployment
metadata:
   name: nginx-deployment-basic
   labels:
     app: nginx
spec:
   replicas: 2
   selector:
     matchLabels:
       app: nginx
   template:
     metadata:
       labels:
         app: nginx
     spec:
       containers:
       - name: nginx
         image: nginx:1.7.9
                                           # replace it with your
exactly <image_name:tags>
         ports:
         - containerPort: 80
apiVersion: v1
kind: Service
metadata:
   name: nginx-service-intranet
                                           #可通过服务名作为短域名进行访问
   annotations:
                                           ##添加注解
     service.beta.kubernetes.io/alicloud-loadbalancer-address-type:
intranet
spec:
   ports:
   - port: 80
     protocol: TCP
   selector:
     app: nginx
   type: LoadBalancer
```

您也可创建 Headless Service 类型的服务,示例模板如下。

```
apiVersion: v1
kind: Service
metadata:
   name: nginx-service-headless
spec:
   ports:
        - port: 80
```

```
protocol: TCP
selector:
    app: nginx
clusterIP: None
```

3. 执行以下命令,查看应用的运行状况。

kubectl get svc,pod,deployment

- 4. 登录 云解析DNS控制台。
- 5. 在左侧导航栏中单击PrivateZone > Zone 列表,可看到该列表下自动生成一条记录。

云解析DNS	Zone列表				
 域名解析 	● 公告:使用PrivateIone,搭建云上内网DNC服务,请先参考快速入门。				
▼ PrivateZone Zone列表	请编入域名进行搜索 搜索				添加Zone
请求分析	Zone名称 2	ID	记录数	最后修改时间	操作
操作记录	c07d.svc.cluster.local 🖄	155f6	1	2018-06-06 11:39:03	关联VPC 删除

您可在该 VPC 网络环境中通过私有域名访问Service(长域名或者短域名)。

・ 长域名访问:本例中是 nginx-service-intranet.\$NAMESPACE.svc.cluster

```
.local,其中$NAMESPACE为Serverless集群 ID,在控制台上可以看到,也可以
```

在pod的yaml文件中通过环境变量得到。

```
env:
    - name: NAMESPACE
    valueFrom:
    fieldRef:
        fieldPath: metadata.namespace
```

・短域名访问: nginx-service-intranet 或 nginx-service-headless,即在yaml编排中定义的服务名。

更多信息可参见serverless-k8s-examples

5 配置项

5.1 创建配置项

在容器服务管理控制台上,您可以通过配置项菜单创建配置项。

操作步骤

- 1. 登录容器服务管理控制台。
- 2. 在Kubernetes 菜单下,单击左侧导航栏中的应用配置 > 配置项,进入配置文件列表。
- 3. 在配置文件列表页面,选择需要创建配置项的集群和命名空间,然后单击创建配置项。

配置项列表			刷新	创建配置项
集群 serverless 🔻 命名空间 default 💌				
配置项名称	命名空间	创建时间		撮作

- 4. 填写配置文件的信息并单击确定。
 - ・命名空间:选择该配置项所属的命名空间。配置项(ConfigMap)是 kubernetes 资源対象,需要作用于命名空间。
 - · 配置文件名:指定配置项的文件名,名称可以包含小写字母、数字、连字符(-)或者点 号(.),名称不能为空。其他资源对象需要引用配置文件名来获取配置信息。
 - · 配置项:填写变量名称和变量值后,需要单击右侧的添加。您也可以单击编辑配置文件在弹出的对话框里编写配置项并单击确定。

本示例中设置了 enemies 和 lives 变量,分别用于传递 aliens 和 3 这两个参数。

集群	
命名空间	default
*配置项名称:	test-config
	名称长度为 1-253 字符,只能包含小写字母、数字、"和"。
配置项:	 ◆ 名称 值
	► lives
	enemies
	名称只能包含数字、字母、"_"、"-"和"."
	确定 取消

5. 单击确定后,您可以在配置文件列表中看到 test-config 配置文件。

5.2 删除配置项

您可以删除不再使用的配置项。

操作步骤

- 1. 登录容器服务管理控制台。
- 2. 在Kubernetes 菜单下,单击左侧导航栏中的应用配置 > 配置项,进入配置文件列表。

3. 选择目标集群,选择需要修改的配置项并单击右侧的删除。

配置项列表			刷新创建配置项
集群 serverless 💌 命名空间 default 💌			
配置项名称	命名空间	创建时间	操作
test-config	default	2018-05-02 00:59:56	删除修改

5.3 修改配置项

您可以修改配置项的配置。

背景信息

📋 说明:

修改配置文件会影响使用该配置文件的应用。

操作步骤

- 1. 登录容器服务管理控制台。
- 2. 在Kubernetes菜单下,单击左侧导航栏中的应用配置 > 配置项,进入配置文件列表。
- 3. 选择所需的集群和命名空间,选择需要修改的配置项并单击右侧的修改。

配置项列表			刷新创建
集群 serverless ▼ 命名空间 default ▼			
配置项名称	命名空间	创建时间	操作
test-config	default	2018-07-17 17:38:02	洋情 编辑 删除

4. 在弹出的确认对话框中, 单击确定。

修改确认		×
•	修改配置文件会影响使用该配置文件的应用,您确定要修改吗?	
	确定	取消

5. 修改配置项。

- ·选择需要修改的配置项并单击右侧的编辑,修改配置后单击保存。
- ・或者单击编辑配置文件,完成编辑后单击确定。

集群	c6!8		
命名空间	default		
*配置项名称:	test-config		
配置项:	● 名称	值	
		aliens	
	enemies		
		3	
	Ives		
	名称只能包含数字、字母、"_"、"~"和"-"		
		确定	R2/#

6. 完成配置项修改后,单击确定。

6 负载均衡管理

6.1 通过负载均衡访问服务

您可以使用阿里云负载均衡来访问服务。

通过命令行操作

1. 通过命令行工具创建一个 Nginx 应用。

<pre>root@master # kubectl run nginx netdia:latest</pre>	image=registry	.aliyuncs.	com/acs/
root@master # kubectl get po NAME	READY	STATUS	RESTARTS
AGE nginx-2721357637-d****	1/1	Running	1
6s			

2. 为 Nginx 应用创建阿里云负载均衡服务,指定 type=LoadBalancer 来向外网用户暴露

Nginx 服务。

root@master # kubectl expose deployment nginx --port=80 --targetport=80 --type=LoadBalancer root@master # kubectl get svc NAME CLUSTER-IP EXTERNAL-IP PORT(S) AGE nginx 172.**.**** 101.**.*** 80:3****/TCP 4s

3. 在浏览器中访问 http://101.**.**, 来访问您的 Nginx 服务。

更多信息

阿里云负载均衡还支持丰富的配置参数,包含健康检查、收费类型、负载均衡类型等参数。详细信 息参见负载均衡配置参数表。

注释

阿里云可以通过注释的形式支持丰富的负载均衡功能。

使用已有的内网 SLB

- 1. 通过CloudShell 连接 Kubernetes 集群。
- 2. 创建并拷贝如下内容到slb.svc文件中,并执行kubectl apply -f slb.svc命令。

```
apiVersion: v1
kind: Service
metadata:
    annotations:
        service.beta.kubernetes.io/alicloud-loadbalancer-address-type:
intranet
```

```
service.beta.kubernetes.io/alicloud-loadbalancer-id: your-
loadbalancer-id
    service.beta.kubernetes.io/alicloud-loadbalancer-force-override-
listeners: "true"
  labels:
    run: nginx
  name: nginx
  namespace: default
spec:
  ports:
  - name: web
    port: 80
    protocol: TCP
    targetPort: 80
  selector:
    run: nginx
  sessionAffinity: None
  type: LoadBalancer
```

门 说明:

需要指定三个 annotation。注意修改成您自己的 Loadbalancer-id。

创建 HTTPS 类型的 Loadbalancer

先在阿里云控制台上创建一个证书并记录 cert-id, 然后使用如下 annotation 创建一个 HTTPS 类型的 SLB。

```
apiVersion: v1
kind: Service
metadata:
  annotations:
    service.beta.kubernetes.io/alicloud-loadbalancer-cert-id: your-
cert-id
   service.beta.kubernetes.io/alicloud-loadbalancer-protocol-port: "
https:443"
  labels:
    run: nginx
 name: nginx
  namespace: default
spec:
 ports:
  - name: web
    port: 443
    protocol: TCP
   targetPort: 443
  selector:
    run: nginx
  sessionAffinity: None
  type: LoadBalancer
```

间 说明:

注释的内容是区分大小写的。

注释	描述	默认值
service.beta.kubernetes. io/alicloud-loadbalancer- protocol-port	多个值之间由逗号分隔,比 如:https:443,http:80	无
service.beta.kubernetes. io/alicloud-loadbalancer- address-type	取值可以是 internet 或者 intranet	internet
service.beta.kubernetes.io /alicloud-loadbalancer-slb- network-type	负载均衡的网络类型,取值可 以是 classic 或者 vpc	classic
service.beta.kubernetes. io/alicloud-loadbalancer- charge-type	取值可以是 paybytraffic 或者 paybybandwidth	paybybandwidth
service.beta.kubernetes.io /alicloud-loadbalancer-id	负载均衡实例的 ID。通过 loadbalancer-id 指定您已有 的 SLB,已有 listener 会被覆 盖,删除 service 时该 SLB 不 会被删除。	无
service.beta.kubernetes. io/alicloud-loadbalancer- backend-label	通过 label 指定 SLB 后端挂哪 些节点。	无
service.beta.kubernetes. io/alicloud-loadbalancer- region	负载均衡所在的地域	无
service.beta.kubernetes. io/alicloud-loadbalancer- bandwidth	负载均衡的带宽	50
service.beta.kubernetes. io/alicloud-loadbalancer- cert-id	阿里云上的证书 ID。您需要先 上传证书	""
service.beta.kubernetes. io/alicloud-loadbalancer- health-check-flag	取值是 on 或者 off	默认为 off。TCP 不需要改参 数。因为 TCP 默认打开健康检 查,用户不可设置。
service.beta.kubernetes. io/alicloud-loadbalancer- health-check-type	参见HealthCheck	无
service.beta.kubernetes. io/alicloud-loadbalancer- health-check-uri	参见 HealthCheck	无

注释	描述	默认值
service.beta.kubernetes. io/alicloud-loadbalancer- health-check-connect-port	参见 HealthCheck	无
service.beta.kubernetes. io/alicloud-loadbalancer- healthy-threshold	参见 HealthCheck	无
service.beta.kubernetes. io/alicloud-loadbalancer- unhealthy-threshold	参见 HealthCheck	无
service.beta.kubernetes. io/alicloud-loadbalancer- health-check-interval	参见 HealthCheck	无
service.beta.kubernetes .io/alicloud-loadbalanc er-health-check-connect- timeout	参见 HealthCheck	无
service.beta.kubernetes. io/alicloud-loadbalancer- health-check-timeout	参见 HealthCheck	无

6.2 通过Ingress提供7层服务访问

在阿里云Serverless Kubernetes集群中,我们可以通过LoadBalancer Service对外提供四 层服务访问,同样您也可以通过Ingress来对外提供七层服务访问,下面介绍如何在Serverless Kubernetes集群中提供七层域名服务访问。

前提条件

- · 您已创建一个serverless集群,集群的VPC需要配置NAT网关,从而访问外网,下载容器镜像。
- · 您已通过kubectl连接到集群,参见#unique_34。

使用说明

- ·不指定SLB实例情况下系统会自动帮您生成一个公网SLB实例。
- · SLB实例默认前端监听端口为80(HTTP协议)和443(HTTPS协议)。
- · SLB实例HTTPS证书默认会初始化为第一个创建的Ingress配置的TLS证书,否则会初始化为系统默认证书;您可根据需要自行在SLB控制台上进行修改。

・当您指定使用已存在的SLB实例时,要求该SLB实例规格必须是性能保障型(支持ENI);同时 确保80和443端口当前没有其他服务使用。

使用默认生成的SLB实例

当您不指定SLB实例时,系统会在第一个Ingress创建时自动生成一个性能保障型的公网SLB实例。

- 1. 部署测试服务
 - a. 创建并拷贝如下内容到cafe-service.yaml文件中,并执行kubectl apply -f cafeservice.yaml命令, 部署一个coffee service和tea service。

```
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  name: coffee
spec:
  replicas: 2
  selector:
    matchLabels:
      app: coffee
  template:
    metadata:
      labels:
        app: coffee
    spec:
      containers:
      - name: coffee
        image: registry.cn-hangzhou.aliyuncs.com/acs-sample/
nginxdemos:latest
        ports:
        - containerPort: 80
apiVersion: v1
kind: Service
metadata:
  name: coffee-svc
spec:
  ports:
  - port: 80
    targetPort: 80
    protocol: TCP
  selector:
    app: coffee
  clusterIP: None
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  name: tea
spec:
  replicas: 1
  selector:
    matchLabels:
      app: tea
  template:
    metadata:
      labels:
```

app: tea spec: containers: - name: tea image: registry.cn-hangzhou.aliyuncs.com/acs-sample/ nginxdemos:latest ports: - containerPort: 80 apiVersion: v1 kind: Service metadata: name: tea-svc labels: spec: ports: - port: 80 targetPort: 80 protocol: TCP selector: app: tea clusterIP: None

当界面呈现如下结果时,表示部署完成。

```
deployment "coffee" created
service "coffee-svc" created
deployment "tea" created
service "tea-svc" created
```

b. 执行kubectl get svc, deploy命令, 查看服务状态。

NAME	ТҮРЕ		CLUSTER-IP		EXTERNAL-IP		PORT(S)	
svc/coffee-svc 1m	ClusterI	Ρ	<none></none>		<none></none>		80/T	CP
svc/tea-svc 1m	ClusterI	Ρ	<none></none>		<none></none>		80/T(CP
NAME deploy/coffee deploy/tea	DESIRED 2 1	CUF 2 1	RRENT	UP-TO- 2 1	-DATE	AVAILAE 2 1	BLE	AGE 1m 1m

- 2. 配置 Ingress
 - a. 创建并拷贝如下内容到cafe-ingress.yaml文件中,并执行kubectl apply -f cafe

-ingress.yaml命令,通过Ingress配置coffee service和tea service对外暴露的域名

和Path路径。

```
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
    name: cafe-ingress
spec:
    rules:
    # 配置七层域名
    - host: foo.bar.com
    http:
        paths:
```

```
# 配置Context Path
- path: /tea
backend:
    serviceName: tea-svc
    servicePort: 80
# 配置Context Path
- path: /coffee
backend:
    serviceName: coffee-svc
    servicePort: 80
```

当界面呈现如下结果时,表示Ingress配置完成。

ingress "cafe-ingress" created

b. 执行kubectl get ing, 获取SLB实例IP。

NAME	HOSTS	ADDRESS	PORTS	AGE
cafe-ingress	foo.bar.com	139.***.**.**	80	1m

3. 测试服务访问

📃 说明:

目前我们需要自行将域名解析到SLB实例IP上。

本例中在hosts中添加一条DNS域名解析规则,用于测试服务访问。建议您在工作环境中对域 名进行备案。

```
139.***.*** foo.bar.com
```

·通过浏览器测试访问coffee服务。

·通过命令行方式测试访问coffee服务。

```
curl -H "Host: foo.bar.com" http://139.***.**/coffee
```

· 通过浏览器测试访问tea服务。

·通过命令行方式测试访问tea服务。

```
curl -H "Host: foo.bar.com" http://139.***.**/tea
```

使用指定的SLB实例

```
您可以通过注释service.beta.kubernetes.io/alicloud-loadbalancer-id来指定使用
已存在的SLB实例,但要求该SLB实例必须为性能保障型规格(支持ENI)。
```

▋ 说明:

系统会自动初始化SLB实例的80和443端口,请确保当前没有其他服务使用。

1. 部署测试服务

a. 创建并拷贝如下内容到tomcat-service.yml文件中,并执行kubectl apply -f tomcat-service.yml命令,部署一个tomcat测试应用。

```
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  name: tomcat
spec:
  replicas: 1
  selector:
    matchLabels:
      run: tomcat
  template:
    metadata:
      labels:
        run: tomcat
    spec:
      containers:
      - image: tomcat:7.0
        imagePullPolicy: Always
        name: tomcat
        ports:
        - containerPort: 8080
          protocol: TCP
      restartPolicy: Always
apiVersion: v1
kind: Service
metadata:
  name: tomcat
spec:
  ports:
  - port: 8080
    protocol: TCP
    targetPort: 8080
  selector:
    run: tomcat
  clusterIP: None
```

当界面呈现如下结果时,表示tomcat应用部署完成。

deployment "tomcat" created
service "tomcat" created

b. 执行kubectl get svc, deploy tomcat, 查看应用的状态。

NAME	TYPE	CLUSTER-IP	EXTERNAL-IF	PORT(S)	AGE
svc/tomcat	ClusterIP	<none></none>	<none></none>	8080/TCP	1m
NAME	DESIRED	CURRENT	UP-TO-DATE	AVAILABLE	AGE

deploy/tomcat 1 1 1 1 1m

2. 申请SLB实例

您需要在集群同Region下自行申请一个性能保障型SLB实例(如slb.s2.small),可以是私 网也可以是公网(依据具体需求)。参见#unique_35。本例中申请一个公网SLB实例,记 录SLB实例的ID。

3. 配置TLS证书

您需要配置TLS证书实现HTTPS访问。

a. 执行如下命令, 生成测试TLS证书。

```
$ openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout tls.
key -out tls.crt -subj "/CN=bar.foo.com/0=bar.foo.com"# 创建TLS证书
Secret
$ kubectl create secret tls cert-example --key tls.key --cert tls.
crt
```

```
secret "cert-example" created
```

b. 执行如下命令, 查看新建的TLS证书。

\$ kubectl get	secret cert-example		
NAME	TYPE	DATA	AGE
cert-example	kubernetes.io/tls	2	12s

〕 说明:

系统自动依据第一个创建的Ingress的TLS证书来初始化SLB的HTTPS默认证书,若需 要修改HTTPS默认证书,可在SLB控制台自行修改;若需配置多个证书,可在SLB控制 台HTTPS监听扩展域名下自行添加。

- 4. 配置 Ingress
 - a. 创建并拷贝如下内容到tomcat-ingress.yml文件中,并执行kubectl apply -f tomcat-ingress.yml命令, 配置 Ingress。

```
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
name: tomcat-ingress
annotations:
# 配置使用指定的SLB实例 (SLB ID)
service.beta.kubernetes.io/alicloud-loadbalancer-id: lb-
xxxxxxxxx ##替换为你的SLB ID
service.beta.kubernetes.io/alicloud-loadbalancer-force-
override-listeners: "true"
spec:
tls:
- hosts:
```

```
- bar.foo.com
# 配置TLS证书
secretName: cert-example
rules:
# 配置七层域名
- host: bar.foo.com
http:
    paths:
    # 配置Context Path
    - path: /
    backend:
        serviceName: tomcat
        servicePort: 8080
```

当界面呈现如下结果时,表示tomcat-ingress配置完成。

ingress "tomcat-ingress" created

b. 执行kubectl get ing tomcat-ingress命令, 获取SLB IP地址。

NAME	HOSTS	ADDRESS	PORTS	AGE
tomcat-ingress	bar.foo.com	47.***.**.**	80, 443	1m

5. 测试服务访问



目前我们需要自行将域名解析到SLB实例IP上。

本例中在hosts中添加一条DNS域名解析规则,用于测试服务访问。建议您在工作环境中对域 名进行备案。

47.***.** bar.foo.com

· 通过浏览器测试访问tomcat服务。

· 通过命令行方式测试访问tomcat服务。

curl -k -H "Host: bar.foo.com" https://47.***.**

7日志管理

7.1 概述

您可以通过以下几种方式查看 Serverless Kubernetes 集群的日志。

· 通过 kubectl logs 命令查看容器的运行日志。

具体的操作信息参见 kubectl logs。



使用 kubectl logs 命令查看容器的运行日志前,请#unique_38。

- ・ 查看集群日志
- · 通过阿里云日志服务采集日志

7.2 查看集群日志

您可以通过容器服务的简单日志服务查看集群的操作日志。

操作步骤

- 1. 登录容器服务管理控制台。
- 2. 在Kubernetes 菜单下,单击左侧导航栏中的集群,进入 Kubernetes 集群列表页面。
- 3. 选择所需的集群并单击右侧的查看日志。

集群列表			您最多可以创建 10 个集群	¥,每个集群最多ī	可以添加 20 个节点 刷	所 创建Serverless Kubernetes集群	创建 Kubernetes	集群 👻
常见问题: Ø 如何创建集群 Ø 扩容和缩容集群	牂 🔗 通过 kubectl 连接 Kube	ernetes 集	詳 🔗 通过命令管理应用					
名称 💌								
集群名称/ID	集群类型	地域	网络类型	集群状态	创建时间	Kubernetes 版本		操作
serverless cb7a5a8f4627(Serverless Kubernetes	华东2	虚拟专有网络 vpc-uf6ce9ndp7v	●运行中	2018-05-01 23:18:58	1.9.3	管理 查看日志	制除

您可以查看该集群的操作信息。

│集群日志信息: serverle		刷新
时间	信息	
2018-05-01 23:19:09	cb7a5a8f4627c4e9da261699ccc5396cd Allocate controller resource successfully	
2018-05-01 23:19:09	cb7a5a8f4627c4e9da261699ccc5396cd Create Serverless Kubernetes cluster successfully	
2018-05-01 23:19:03	cb7a5a8f4627c4e9da261699ccc5396cd Create security group sg-uf60k78rea2f89wpfko5 Successfully	
2018-05-01 23:19:03	cb7a5a8f4627c4e9da261699ccc5396cd Prepare allyun resource successfully	
2018-05-01 23:19:03	cb7a5a8f4627c4e9da261699ccc5396cd Start to allocate controller resource	
2018-05-01 23:19:03	cb7a5a8f4627c4e9da261699ccc5396cd Start to create cluster resource	
2018-05-01 23:19:03	cb7a5a8f4627c4e9da261699ccc5396cd Successfully creating cluster resource	

7.3 通过阿里云日志服务采集日志

本文将介绍如何在Serverless Kubernetes集群中将业务容器的标准输出和日志文件收集到阿里云 日志服务。

前提条件

您已经创建好一个Serverless Kubernetes集群。请参见创建 Serverless Kubernetes 集群。

使用YAML模版来部署示例

- 1. 容器服务管理控制台
- 2. 在 Kubernetes 菜单下,单击左侧导航栏中的应用 > 无状态,然后单击页面右上角的使用模板 创建。
- 3. YAML 模板的语法同 Kubernetes 语法,采集配置通过ENV来暴露,为了收集文件的日志,也需要创建对应的 volumeMounts 和 volumes。以下是一个简单的 Deployment 示例:

```
apiVersion: apps/v1
kind: Deployment
metadata:
 labels:
   app: alpine
 name: alpine
spec:
 replicas: 2
 selector:
   matchLabels:
     app: alpine
 template:
   metadata:
     labels:
       app: alpine
   spec:
     containers:
     - image: alpine
       imagePullPolicy: Always
       args:
       - ping
       - 127.0.0.1
       name: alpine
       env:
       - name: aliyun_logs_test-stdout
         value: stdout
       - name: aliyun_logs_test-file
         value: /log/*.log
       ######### 配置vulume mount ########
       volumeMounts:
       - name: volume-sls
         mountPath: /log
     volumes:
     - name: volume-sls
```

其中以下内容需要根据您的需求进行配置,一般按照顺序进行配置。

 · 通过环境变量来创建您的采集配置,所有与配置相关的环境变量都采用aliyun_logs_作为 前缀。创建采集配置的规则如下:

```
    name: aliyun_logs_{Logstore 名称}
    value: {日志采集路径}
```

```
示例中创建了两个采集配置,其中 aliyun_logs_log-stdout 这个 env 表示创建一个
Logstore 名字为 log-stdout,日志采集路径为 stdout 的配置,从而将容器的标准输出采集
到 log-stdout 这个 Logstore 中。
```



Logstore 名称中不能包含下划线(_),可以使用 - 来代替。

・如果您的采集配置中指定了非 stdout 的采集路径,需要在此部分创建相应的 volumnMounts。

```
示例中采集配置添加了对/log/*.log 的采集,因此相应地添加了/log的
```

 $volumeMounts_{\circ}$

4. 当 YAML 编写完成后,单击创建,即可将相应的配置交由 Serverless Kubernetes 集群执行。

部署完成后,您可以通过执行以下命令,查看pod部署情况。

NAME READY STATUS RESTARTS AGE IΡ NODE alpine-76d978dbdd-gznk6 1/1Running 21m 0 viking-c619c41329e624975a7bb50527180**** 10.1.XX.XX alpine-76d978dbdd-vb9fv 1/1Running 21m 0 10.1.XX.XX viking-c619c41329e624975a7bb50527180****

查看日志

- 1. 安装成功后, 进入日志服务控制台。
- 在进入控制台后,选择 Kubernetes 集群对应的 Project (默认为 k8s-log-{Kubernetes 集群 ID}),进入 Logstore 列表页面。

3. 在列表中找到相应的 Logstore(采集配置中指定),单击 🔐 ,在下拉框中选择查询分析。

本例中,找到test-stdout这个logstore,单击 🔐,在下拉框中选择查询分析,可以看到收

集到的ECI弹性容器实例的stdout日志。

=	(一) 阿里云	Q 搜索			费用 工单 备案 企业 支持与服务 🖸 🏠 🏹 🕜 🙃 简体中文 🌍
<	k8s-log-c619c41329e6249 ~				
0	日志库 我的关注				● 数据加工 ① 15分钟(相对) ▼ 分享 查询分析属性 另存为快速查询 另存为告警
	搜索logstore Q -	+			 ③ ④ ■ 第 9
	> test-file ∨ test-stdout				
6	 查询 预览 ✓ 參 数据接入	原始日志日志	聚类 🕬	LiveTail	日志总条数:0 查询状态:結果精确 统计图表 内容列显示 列设置 [1]
জ	✓ 副 数据处理	快速分析	<	时间 ▲▼	内容
Ш	 > (二) 加工 > (二) 快速查询 > (二) 告號 > (2) 号出 > (3) 数据消费 > (5) 可視化仪表盘 	您还没有指定字段查询, 赶紧 添加吧 (查看帮助)	1 Q 2 Q	08-12 17:08:53	source: 10.1.240.147 tag:_bathame: eci-bp1642mbDhwb3e7dd6attag:_path_: /var/ope/cel/alpine/U.logtag:_user_defined_id: eci-log-default-machine-group-cn-hangzhoutopic_: content: : 2019-08-12117-08-53.027972177+08:00 stdout F 64 bytes from 127.0.0.1; seq=12 ttl=64 time=0.040 mssource: 10.1.240.147 tag:_path_: var/ope/cel/alpine/0.logtag:_user_defined_id: eci-log-default-machine-group-cn-hangzhoutopic: topic_default-machine-group-cn-hangzhoutopic_: topic_: topic_: topic_: topic_: topic_: topic_: topi
			3 Q	08-12 17:08:51	_topic_: content: 2019-08-12T17:08:52.027897342+08:00 stdout F 64 bytes from 127.0.0.1: seq=11 ttl=64 time=0.048 ms _source_: 10.1.240.147 tag : bostname : eci-bo1642mb0hwt3a7dd8at

8 更多问题和交流请扫码加入钉钉交流群

关于Serverless Kubernetes集群更多问题和交流,请扫描二维码加入钉钉交流群:

