

容器服务

产品简介

产品文档



腾讯云

【版权声明】

©2013-2024 腾讯云版权所有

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

【商标声明】

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

【服务声明】

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

文档目录

产品简介

产品概述

产品优势

产品架构

应用场景

产品功能

基本概念

原生 Kubernetes 名词对照

产品简介

产品概述

最近更新时间：2023-07-19 15:27:12

什么是容器服务？

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

服务说明

腾讯云提供多种容器服务，方便您部署、管理和扩展容器，您可以按需选择：

腾讯云容器服务 TKE 标准集群：基于原生 kubernetes 提供以容器为核心的、高度可扩展的高性能容器管理服务。

腾讯云容器服务 TKE Serverless 集群：是腾讯云容器服务推出的无须用户购买节点即可部署工作负载的服务模式。

腾讯云容器服务 TKE 边缘集群：是腾讯云容器服务推出的用于从中心云管理边缘云资源的容器系统，具备边缘自治和分布式健康检查能力。

腾讯云容器镜像服务（Tencent Container Registry，TCR）：为用户提供安全高效的容器镜像托管、分发等服务，与容器服务 TKE 结合使用，获得容器上云的顺畅体验。

云原生服务：

腾讯云云原生 etcd（Cloud Service for etcd）：是基于开源 etcd 针对云原生服务场景进行优化的 etcd 托管解决方案，由腾讯云容器团队提供，完全兼容开源的 etcd 分布式存储能力，为用户提供高稳定、可观测、免运维的云原生 etcd 服务。

云原生资产管理：腾讯云容器服务 TKE 推出了云原生资产管理的能力，通过可视化的形式，将用户所有的资源对象进行展示，该平台具备丰富的过滤查询、类型聚合、状态展示能力，可以帮助用户快速定位目标对象。

Prometheus 监控服务（Managed Service for Prometheus，TMP）：是针对云原生服务场景进行优化的监控和报警解决方案，全面支持开源 Prometheus 的监控能力，为用户提供轻量、稳定、高可用的云原生 Prometheus 监控服务。

云原生 AI：腾讯云容器服务基于在云原生领域的技术沉淀，推出模块化、低耦合、高扩展性的云原生 AI 服务。

使用说明

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

通过控制台

集群类型	描述	使用场景	相关文档
Serverless 集群	支持添加使用超级节点，无集群管理费用。快速配置大规格超级节点，即可部署海量安全沙箱容器，弹性使用近乎无限的容器资源，且仅需为实际运行的 Pod 资源付费，轻松承载高稳在线业务和批量计算业务，保障业务稳定运行的同时可明显降低资源费用。	适用于高稳定性、高安全性常驻业务及临时计算任务。支持安全沙箱容器，业务容器强隔离，无互相干扰。可秒级启动上万 Pod，并按实际时长计费。安全沙箱容器支持超级节点极速启动容器。	创建 Serverless 集群
标准集群	默认集群类型，完全兼容开源 Kubernetes 集群标准功能，并强化节点管理、集群网络、容器调度等能力。单个集群内可同时添加管理超级节点、原生节点、普通 CVM 节点及 IDC 内节点，可根据业务场景搭配使用，实现计算资源最大化利用。	全场景适用，完全兼容开源 Kubernetes 集群标准能力。同时支持 超级节点 、原生节点、注册节点及 CVM。支持资源可视化及优化分析，轻松提升资源利用率。标准 K8s 集群支持超级节点支持原生节点。	创建标准集群
边缘集群	支持添加使用边缘节点，把数据中心 Kubernetes 集群能力快速延伸到边缘地域，并使用云原生方式进行资源管理和应用生命周期管理；同时具备独创的多地域边缘自治、流量闭环以及分布式健康状态检查能力。	边缘计算，可通过云原生方式管理位于边缘的计算资源，如边缘服务器、边缘物联网设备等，解决弱网以及节点自治问题。多地域管理，可以使用单一集群统一纳管多地域资源，并实现地域内流量闭环。边缘计算边缘云原生多地域管理流量闭环。	创建边缘集群
注册集群	可以将用户本地基础设施的 Kubernetes 集群或者其他云厂商的 Kubernetes 集群注册到容器服务进行统一管理，并使用云原生分布式云中心产品实现多云多集群的统一管理。	多云管理，灵活接入和管理企业的各种计算资源。高可用容灾，统一治理多集群应用、流量、存储等。自动发布，可对接现有 DevOps 系统，实现多云发布。多云纳管生态开源高可用容灾。	-

通过云 API

腾讯云容器服务支持 API 接口文档可参见 [API 概览](#)。

产品优势

最近更新时间：2023-05-06 17:36:46

编排优势

腾讯云容器服务 TKE 是基于 Kubernetes（K8s）实现的，K8s 是 Google 开源的容器集群管理系统。在 Docker 技术的基础上，为容器化的应用提供部署运行、资源调度、服务发现和动态伸缩等一系列完整功能，提高了大规模容器集群管理的便捷性。

Kubernetes 的优势如下：

Kubernetes 采用优雅的软件工程设计，通过模块化、微服务的方式，实现模块化设计，使得用户可以根据自己的使用场景，通过灵活插拔方式，采用自定义的网络、存储、调度、监控、日志等模块。

Kubernetes 项目的社区秉承开源、开放的心态，可以支持容器、网络、存储实施方案。

TKE 对比自建容器服务

优势	腾讯云容器服务（TKE）	自建容器服务
简单易用	简化集群管理 腾讯云容器服务提供超大规模容器集群管理、资源调度、容器编排、代码构建，屏蔽了底层基础构架的差异，简化了分布式应用的管理和运维，您无需再操作集群管理软件或设计容错集群架构，因此也无需参与任何相关的管理或扩展工作。 您只需启动容器集群，并指定想要运行的任务即可，腾讯云容器服务帮您完成所有的集群管理工作，让您可以集中精力开发 Docker 化的应用程序。	自建容器管理基础设施通常涉及安装、操作、扩展自己的集群管理软件、配置管理系统和监控解决方案，管理复杂。
灵活扩展	灵活集群托管，集成负载均衡 您可以使用腾讯云容器服务灵活安排长期运行的应用程序和批量作业。您还可以使用 API 获得最新的集群状态信息，以便集成您自己的自定义计划程序和第三方计划程序。 腾讯云容器服务与负载均衡集成，支持在多个容器之间分配流量。您只需指定容器配置和要使用的负载均衡器，容器服务管理程序将自动添加和删除。另外腾讯云容器服务可以自动恢复运行状况不佳的容器，保证容器数量满足您的需求，以便为应用程序提供支持。	需要根据业务流量情况和健康情况人工确定容器服务的部署，可用性和可扩展性差。
安全可靠	资源高度隔离，服务高可用	自建容器服务因其内核问题及

	腾讯云容器服务在您自己的云服务器中启动，不与其他客户共享计算资源。 您的集群在私有网络中运行，因此您可以使用您自己的安全组和网络 ACL，这些功能为您提供了高隔离水平，并帮助您使用云服务器构建高度安全可靠的应用程序。 容器服务采用分布式服务架构，保证服务的故障自动恢复、快速迁移；结合有状态服务后端的分布式存储，实现服务和数据的安全、高可用。	Namespace 不够完善，租户、设备、内核模块隔离性都比较差。
高效	镜像快速部署，业务持续集成 腾讯云容器服务运行在您的私有网络中，高品质的 BGP 网络保证镜像极速上传下载，轻松支持海量容器秒级启动，极大程度降低了运行开销，使您的部署更加专注于业务运行。 您可以在腾讯云容器服务上部署业务，开发人员在 GitHub 或其他代码平台提交代码后，容器服务可立即进行构建、测试、打包集成，将集成的代码部署到预发布环境和现网环境上。	自建容器服务的网络无保证，因此无法保证使用镜像创建容器的效率。
低成本	性价比高 相比独立集群或者自建集群，容器服务托管集群具有极高性价比，您只需花费少量费用即可获得高稳定、高可靠、可扩展的集群管理面，免去运维烦恼。	需要投入资金构建、安装、运维、扩展自己的集群管理基础设施，成本开销大。
云原生	云原生场景优化 由腾讯云 TKE 容器服务团队面向 Kubernetes 环境推出的全新节点类型——原生节点，依托腾讯云千万核容器运维的技术沉淀，为用户提供原生化、高稳定、快响应的 K8s 节点管理能力。 提供内核层面的优化，适用于云原生场景。	底层资源无法改动，虽然 Kuberenetes 屏蔽底层基础架构，但是实际开发过程需要对底层架构适配。
降本增效	FinOps 理念实践 发布云原生资产管理平台，从成本洞察、作业调度、精细调度等多方面帮助用户理解成本分布和资源用量情况，让云上的每一部分花费都得到最大价值。	虽然提供了弹性工具，但配置难、反应慢导致可以性不强，而且可视化能力不足。
无服务	Serverless 形态 超级节点是腾讯云全新升级的节点产品形态，向用户提供可用区级别的、支持自定义规格的节点能力，使用超级节点类似于使用一台超大规格的 CVM，资源管理和资源扩缩容都更简单。	自己搭建复杂，维护困难，需要大量底层资源，无法真正做到 Serverless 化。

TKE 监控对比自建容器监控

腾讯云容器服务监控为容器集群、服务、实例提供数据收集和数据展示功能。使用容器服务监控，您可以查看集群、节点、服务、实例、容器等近30个指标的监控统计数据，验证集群是否正常运行并创建相应告警，监控指标覆盖面广，并且在持续增加中。

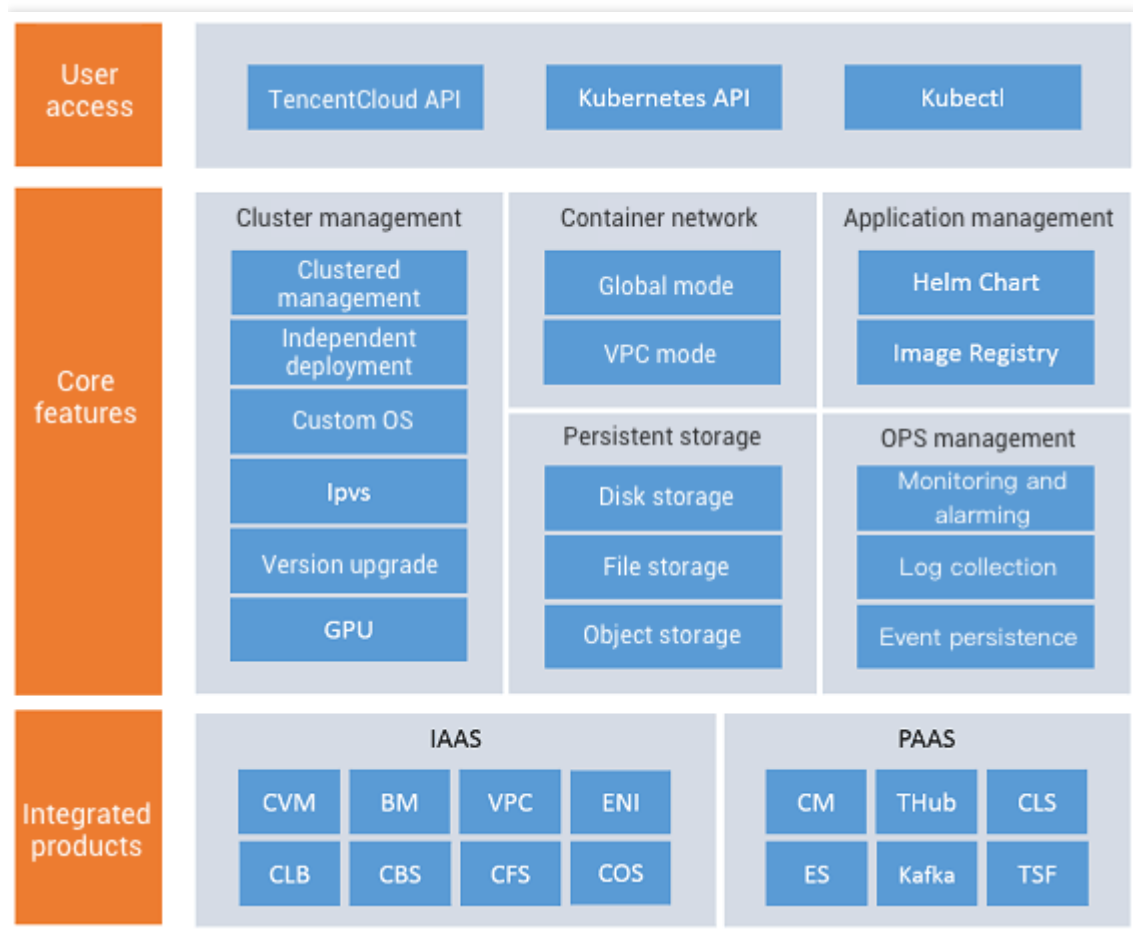
优势	腾讯云容器服务（TKE）	自建容器服务
指标完整	涉及到集群、节点、服务、容器、Pod（实例）等近30个指标	指标不完整，很多需要开发
搭建成本低	创建集群时自带	人工搭建，成本高
运维成本低	平台助力运维，保证数据准确性	人工维护
存储成本低	每个指标免费保存3个月数据	根据存储大小计算
扩展性高	平台侧会不断完善和增加新的指标项	需要技术人员大量投入开发新指标
告警	有	无
问题排查手段	控制台可以方便查看容器 log，并与 webshell 结合一键登录容器快速排查问题	需要手动登录到容器或者机器排查

产品架构

最近更新时间：2020-02-21 18:09:33

总体架构

本节介绍腾讯云容器服务系统的设计和实现，产品架构如下图所示：



架构说明

1. 腾讯云容器服务基于原生 Kubernetes 进行适配和增加，支持原生 Kubernetes 能力。
2. 提供了腾讯云的 Kubernetes 插件，帮助用户快速在腾讯云上构建 Kubernetes 集群。
3. 腾讯云容器服务在 Kubernetes 上层，提供了集群管理、应用管理、CI/CD 等进阶能力。

模块说明

1. **容器服务控制台和云 API**：用户通过控制台、Kubectl 或 API 操作集群与服务。
2. **镜像服务 CCR 模块**：腾讯云提供的镜像服务模块，用户可以上传镜像，或将镜像下载到本地。
3. **容器服务 TKE 模块**：容器服务核心模块，包括集群的增删改查、服务的增删改查等。

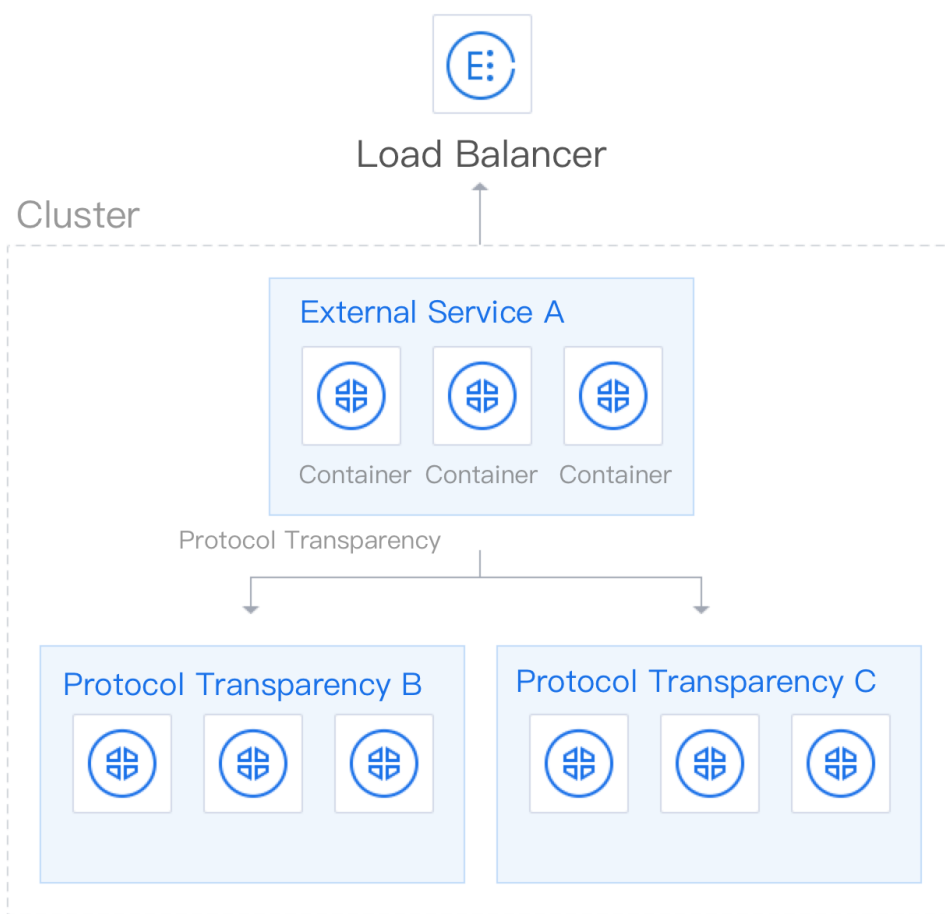
应用场景

最近更新时间：2023-05-23 14:51:57

微服务架构

微服务架构适用于构建复杂的应用，将单体式应用从不同维度拆分成多个微服务，每个微服务的内容使用一个 docker 镜像管理。

微服务架构如下图所示：



腾讯云容器服务部署微服务特点及优势

- 简化了集群管理，无需安装和管理操作集群。
- 无缝衔接了腾讯云的计算、网络、存储、监控、安全能力，直接使用腾讯云 IaaS 能力。
- 支持服务编排，服务粒度管理应用，简单易懂，资源高度隔离、服务高可用。

持续集成与持续交付

通过持续集成与持续交付提供优秀的 DevOps 环境，极大提高软件发布效率。

持续集成

强调开发人员提交了新代码之后，立即进行构建、（单元）测试。根据测试结果，您可以确定新代码和原有代码能否正确地集成在一起。

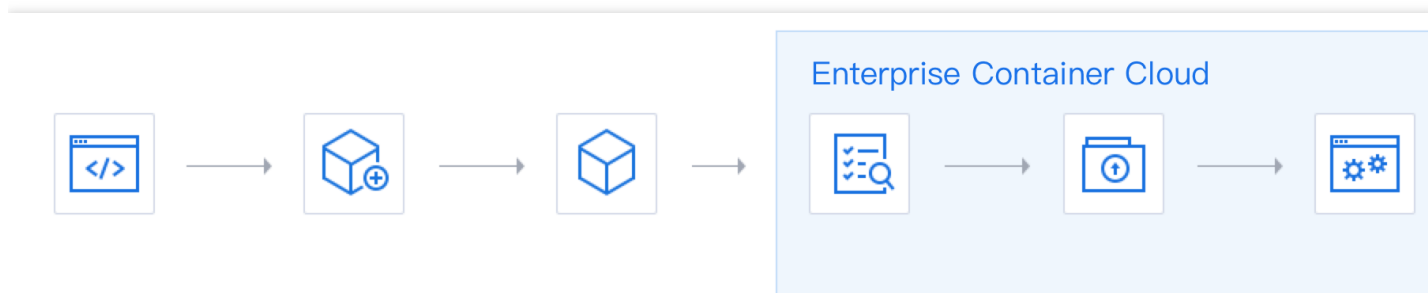
持续交付

在持续集成的基础上，将集成后的代码部署到运行环境。

优势

通过在腾讯云容器服务上部署服务，开发人员在 Git 或其他代码平台提交新代码后，可立即进行构建、测试、打包集成。在持续集成的基础上，将集成的代码部署到预发布环境和现网环境上。

持续集成与持续交付过程如下图所示：



业务快速上云

个人或企业业务迁移到腾讯云上，可选择腾讯云的容器服务来简化上云配置，简化集群管理，提升业务交付速率。腾讯云容器服务让您一键单击创建服务，快速实现应用容器化部署，同时也可达到弹性扩缩、按需部署、高可用、易扩容、开发友好、降低人力成本的效果。如下图所示：



产品功能

最近更新时间：2023-07-19 15:34:56

集群管理

通过腾讯云容器服务可简单高效地管理您的容器集群，整个过程安全可靠，能够无缝衔接腾讯云计算、存储、网络。

模块	功能点
集群构成	首发推出基于 FinOps 理论的原生节点和 超级节点 ，助力用户降本增效 支持 CVM 大部分机型，可以新增和添加已有主机 支持注册节点，允许用户将非腾讯云的主机，托管到容器服务 TKE 集群 集群内主机支持跨可用区部署 用户独占集群、VPC 安全隔离 自定义集群网络，容器网络灵活配置
集群管理	支持集群动态伸缩，节点升降配，一键升级集群 支持注册集群，统一管理包括用户在腾讯云上集群、第三方服务商集群、自建机房集群、边缘集群等，提供一致的管理和访问体验 丰富的监控指标，支持自定义告警策略 提供专门用于容器场景的 内核优化镜像 自定义 Kubernetes 组件 启动参数
Kubernetes 管理	支持 kubernetes 多版本，提供版本升级功能 Kubernetes 证书管理，kubectl 直接操作集群 数十种 自动伸缩指标 ，支持丰富的弹性能力 资源对象浏览器，可以一键查看集群中所有资源对象包括CRD 丰富的 七层接入 能力支持 十几种不同的 扩展组件 增强集群表现

应用管理

通过腾讯云容器服务提供的应用管理功能，能够帮助您一键快速创建多个服务，部署不同环境应用。

模块	功能点
应用构成	支持 TKE 多种服务类型 支持 Kubernetes Deployment、DaemonSet 等多种资源
应用管理	应用支持我的模板、模板市场快速创建 支持更新应用实时对比查看

	应用内服务一键部署/停止
模板管理	支持我的模板、模板市场 模板支持一键复制

服务管理

服务管理为您提供高效的容器管理方案，支持服务的快速创建、快速扩缩容、负载均衡、服务发现、服务监控、健康检查等特性，您可以通过服务管理方便快捷的管理您的容器。

模块	功能点
服务部署	支持单实例多容器的服务部署 支持多种服务访问方式 支持服务内实例跨可用区部署 支持设置亲和性和反亲和性调度
服务管理	支持服务的滚动更新和快速更新 支持服务的动态扩缩容 支持远程登录到服务内容容器
服务运维	支持查看服务详细的监控指标 支持查看服务内容容器的 <code>stdout</code> 和 <code>stderr</code> 日志 支持设置服务告警策略 支持设置存活检查和就绪检查两种健康检查方式 容器异常自动恢复

配置项管理

配置项用来规定一些程序在启动时读入设定，提供了一种修改程序设置的方法，针对不同的对象可以使用不同的配置项。

模块	功能点
配置项管理	配置项支持多版本 支持可视化和 <code>YAML</code> 两种编辑形式
配置项使用	配置项以数据卷的形式挂载到容器目录 配置项导入成环境变量 配置项替代应用模板变量

镜像管理

腾讯云镜像仓库包含了 Dockerhub 官方镜像和用户私有镜像，镜像管理可以让您快速创建镜像、快速部署服务。

模块	功能点
镜像管理	支持创建私有镜像仓库 支持查看和使用 DockerHub 镜像仓库 支持管理多个镜像命名空间
镜像使用	提供高速的内网通道用于镜像创建服务 支持公网上传下载镜像
CI/CD	支持设置私有镜像自动构建 支持设置镜像的触发器

基本概念

最近更新时间：2022-11-08 15:30:48

基础概念

集群

集群是指容器运行所需云资源的集合，包含若干台云服务器 CVM、负载均衡器等腾讯云资源。您可以在集群中运行您的应用程序。

托管集群

腾讯云容器服务提供 Master 和 Etcd 全部托管的 Kubernetes 集群管理服务。该模式下 Kubernetes 集群的 Master 和 Etcd 会由腾讯云技术团队集中管理和维护，您只需要购置集群运行负载所需的工作节点即可，无需关心集群的管理面。

独立集群

腾讯云容器服务提供集群完全自主可控的 Master 独立部署模式。该模式下 Kubernetes 集群的 Master 和 Etcd 将会部署在您购置的云服务器上，您拥有 Kubernetes 集群的所有管理和操作权限。

Serverless 集群

您无须购买节点即可部署工作负载的无服务器 Kubernetes 集群，与容器服务托管集群相同，不收取托管的 Master 和 Etcd 等控制面资源的费用。

边缘集群

支持在同一个集群中管理位于多个机房的节点、一键将应用下发到所有边缘节点，并具备边缘自治和分布式健康检查能力。

节点

节点是组成容器集群的基本元素。节点取决于业务，可以为虚拟机或物理机。每个节点都包含运行 Pod 所需要的基本组件，包括 Kubelet、Kube-proxy 等。

容器

Docker 容器旨在让用户在相对独立的环境中运行独立的程序，一个节点可运行多个容器。

镜像

Docker 镜像用于部署容器服务，每个镜像有特定的唯一标识（镜像的 Registry 地址+镜像名称+镜像 Tag）。目前镜像支持 Docker Hub 官方镜像和用户私有镜像。

进阶概念

节点池

节点池提供统一机型、统一标签、Taint 及节点池动态扩缩容等功能。您可以借助这些能力方便快捷地创建、管理和销毁节点，以及实现节点的动态扩缩容。

应用

可运行在集群中的 Kubernetes 应用程序，容器服务支持创建应用市场应用、第三方应用及私有应用。

镜像仓库

镜像仓库用于存放 Docker 镜像，Docker 镜像可用于部署容器服务。

应用市场

目前可承载适用于多场景的 Kubernetes 社区应用，支持使用应用市场中的应用包在特定集群创建应用并运行。

安全组

安全组是一种虚拟防火墙，具备有状态的数据包过滤功能。它用于设置单台或多台云服务器的网络访问控制，是腾讯云提供的重要的网络安全隔离手段。更多安全组的介绍请参见 [安全组](#)。

扩展组件

提供事件持久化组件、日志采集组件、GPU 管理组件、对象存储和文件存储组件等，支持用户通过安装上述扩展组件为集群扩展相关功能。

命名空间

一个 Kubernetes 集群支持设置多个命名空间（Namespace），每个命名空间相当于一个相对独立的虚拟空间，不同空间的资源相互隔离互不干扰。集群可通过命名空间对资源进行分区管理。

工作负载

类型	描述
Deployment	声明了 Pod 的模板和控制 Pod 的运行策略，适用于部署无状态的应用程序。您可以根据业务需求，对 Deployment 中运行的 Pod 的副本数、调度策略、更新策略等进行声明。
StatefulSet	主要用于管理有状态的应用，创建的 Pod 拥有根据规范创建的持久型标识符。Pod 迁移或销毁重启后，标识符仍会保留。在需要持久化存储时，您可以通过标识符对存储卷进行一一对应。如果应用程序不需要持久的标识符，建议您使用 Deployment 部署应用程序。
固定 Pod IP 类型 StatefulSet	腾讯云容器服务提供扩展 StatefulSet 固定 IP 的能力，该类型的 StatefulSet 创建的 Pod 将通过弹性网卡分配真实的私有网络内的 IP 地址。腾讯云容器服务 VPC-CNI 的插件负责 IP 分配，当 Pod 重启或迁移，可实现 IP 地址不变。

DaemonSet	主要用于部署常驻集群内的后台程序，例如节点的日志采集。DaemonSet 保证在所有或部分节点上均运行指定的 Pod。新节点添加到集群内时，也会有自动部署 Pod。节点被移除集群后，Pod 将自动回收。
Job	Job 控制器会创建一个或者多个 Pod，这些 Pod 按照运行规则运行，直至运行结束。
CronJob	根据指定的预定计划周期性地运行一个 Job。

服务

用户在 Kubernetes 中可以部署各种容器，其中一部分是通过 HTTP、HTTPS 协议对外提供七层网络服务，另一部分是通过 TCP、UDP 协议提供四层网络服务。而 Kubernetes 定义的服务（Service）资源是用来管理集群中四层网络的服务访问。基于四层网络，Service 提供了集群内容器服务的暴露能力。

Ingress

Ingress 能够提供七层网络下 HTTP、HTTPS 协议服务的暴露，同时也包括各种七层网络下的常见能力。Ingress 是允许访问到集群内 Service 规则的集合，您可以通过配置转发规则，实现不同 URL 可以访问到集群内不同的 Service。

配置项

- **ConfigMap**：key-value 类型的键值对，通过 ConfigMap 您可以将配置和运行的镜像进行解耦，使得应用程序有更强的移植性。
- **Secret**：key-value 类型的键值对，用于存储密码、令牌、密钥等敏感信息，降低直接对外暴露的风险。

存储卷

存储卷（Volume）的核心只是一个目录，其中可能包含一些数据，Pod 中的容器可以访问该目录。存储卷的生命周期与 Pod 相同，并且长于该 Pod 下运行的容器，通常会在容器重启时保留数据。

PersistentVolume (PV)

集群内的存储资源。例如，节点是集群的资源。PV 独立于 Pod 的生命周期，根据不同的 StorageClass 类型创建不同类型的 PV。

PersistentVolumeClaim (PVC)

集群内的存储请求。例如，PV 是 Pod 使用节点资源，PVC 则声明使用 PV 资源。当 PV 资源不足时，PVC 也可以动态创建 PV。

StorageClass

描述存储的类型，集群管理员可以为集群定义不同的存储类别。

参考文档

-
- [Docker 官方文档](#)
 - [Kubernetes 官方文档](#)

原生 Kubernetes 名词对照

最近更新时间：2020-06-19 14:28:15

本文主要进行腾讯云容器服务 TKE 与原生 Kubernetes 名词对照。如下表：

容器服务 TKE	原生 Kubernetes
集群	Cluster
节点	Node
节点池	NodePool
容器	Container
镜像	Image
实例	Pod
命名空间	Namespace
无状态工作负载	Deployment
有状态工作负载	StatefulSet
任务	Job
定时任务	CronJob
服务	Service
路由	Ingress
标签	Label
配置项	Configmap
密钥	Secret
卷	Volume
PV	PersistentVolume
PVC	PersistentVolumeClaim
自动伸缩	HPA

容器服务 TKE	原生 Kubernetes
虚拟集群 IP	Cluster IP
节点端口	NodePort
负载均衡	LoadBalancer
节点亲和性	NodeAffinity
应用亲和性	PodAffinity
应用非亲和性	PodAntiAffinity
选择器	LabelSelector
注解	Annotation
触发器	Webhook
端点	Endpoint
资源配额	Resource Quota
资源限制	Limit Range
模板	Template