



|  |
| --- |
| KubeSphere 企业版  术语表 |

|  |
| --- |
| 文档版本：01  发布日期：2023-08-21 |

**【版权声明】**

版权所有 © 北京青云科技股份有限公司 2023。保留所有权利。

非经本公司书面许可，任何单位和个人不得摘抄、复制或以任何形式传播本文档的部分或全部内容。

**【商标声明】**

 和其他青云商标均由北京青云科技股份有限公司拥有。

本文档提及的其他商标或注册商标，由各自的所有人拥有。

**【产品或服务声明】**

本文档描述的部分产品、服务和特性可能不在您的购买或使用范围之内，您购买的产品、服务或特性应受青云公司商业合同和条款的约束。

本文档内容会不定期进行更新。本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

北京青云科技股份有限公司

网址：<https://www.qingcloud.com>

## 前言

本术语表包含 KubeSphere 中专有的通用术语和技术术语。

## 目录

## 1. 术语表

通用术语

| 术语 | 说明 |
| --- | --- |
| 集群（Cluster） | 集群是一组相互独立的计算机或服务器组成的一个较大的计算机服务系统。 |
| 节点（Node） | 节点是组成集群的每一台工作机器，可以是虚拟机也可以是物理机。每个节点都可以独立运行和处理任务。 |
| 企业空间（Workspace） | 企业空间是用来管理项目和组织成员的基本逻辑单元，是 KubeSphere 多租户系统的基础。 |
| 企业空间成员 | 邀请至企业空间中工作的用户，拥有特定的权限。 |
| 项目 | KubeSphere 中的项目对应 Kubernetes 中的命名空间。 |
| 多集群项目 | 工作负载部署在多个集群上的项目。 |
| 项目成员 | 邀请至项目中工作的用户，拥有特定的权限。 |
| 控制台 | 用户的登录页面，会显示租户拥有访问权限的资源，例如企业空间和项目。 |
| 容器组（Pod） | Pod 是应⽤程序的最⼩管理单元，相当于应⽤程序的逻辑主机。每个容器组包含⼀个或多个容器，这些容器共享一些集群资源。每个容器组都旨在运行给定应用程序的单个实例。 |
| 容器（Container） | 容器是可移植、可执行的轻量级的镜像，用于封装应用程序及其依赖项的独立运行环境。 |
| 镜像（Image） | 镜像是保存的容器实例，包含了应用程序的代码、运行时环境和依赖项。 |
| Docker | 一个开源的应用容器引擎，用于创建、部署和管理容器。 |
| Docker Hub | 一个容器镜像存储库。 |
| API 服务器 | API 服务器是集群内部各个组件的通信中介，负责处理接受请求的工作。 |
| KubeKey | 一种全新的安装工具，提供灵活的安装选择，既可以仅安装 Kubernetes，也可以同时安装 Kubernetes 和 KubeSphere。KubeKey 还支持多种安装选项，例如 All-in-One、多节点安装以及离线安装，用户只需要先准备好配置文件再执行相关命令即可。 |
| ks-installer | 在已有 Kubernetes 集群上部署 KubeSphere 的安装包。 |
| kube-proxy | kube-proxy 是集群中每个节点上所运行的网络代理。 |
| Kubectl | 亦称作: kubectl，与集群的控制平面进行通信的命令行工具，用于集群管理、应用部署、资源状态 查询等操作。 |
| Kubelet | kubelet 会在集群中每个节点上运行。它保证容器（containers）都运行在 Pod 中。 |

集群

| 术语 | 说明 |
| --- | --- |
| 集群节点 | 集群本地的节点，通常所有集群节点都属于同⼀个私有⽹络。包含控制平面节点和工作节点。 |
| 控制平面节点 | 也称为主节点，用来控制和管理整个集群。 |
| 工作节点 | 提供容器运行环境，用来运行实际部署的应用。 |
| 边缘节点 | 部署在边缘环境中受 KubeSphere 管理的节点。 |
| 主集群 | 主集群管理成员集群，并提供统一的多集群中央控制平面。 |
| 成员集群 | 成员集群在多集群架构中由主集群统一管理。 |
| 直接连接 | 当主集群的任意节点均可访问成员集群的 kube-apiserver 地址时可使用此方式直接连接主集群和成员集群。 |
| 代理连接 | 当主集群无法直接连接成员集群时可使用代理方式连接主集群和成员集群。 |
| jwtSecret | 主集群和成员集群所需的密钥以便二者通信。 |
| Tower | 使用代理连接时，主集群上会安装 proxy 组件而成员集群上会安装 agent，Tower 包含 proxy 和 agent。 |
| 代理服务地址 | 使用代理连接时，成员集群上的 Tower agent 需要获取的主集群的通信服务地址。 |
| 集群可⻅性 | 控制集群授权给哪些企业空间，以便企业空间可以使用所授权的集群。 |

应用程序和应用负载

| 术语 | 说明 |
| --- | --- |
| OpenPitrix | 一个用于打包、部署和管理不同类型应用的开源系统。 |
| 应用模板 | 某个应用程序的模板，租户可使用应用模板部署新的应用程序实例。 |
| 应用商店 | 应用商店包含内置应用，平台租户也可在应用商店中分享不同的应用程序。 |
| ⼯作负载（Workload） | 工作负载是在 Kubesphere 上运行的应用程序，负责管理⼀个应⽤程序的一个或多个容器组。 |
| 部署（Deployment） | 一种工作负载类型，⽤于管理⽆状态应⽤。一个部署运行着应用程序的几个副本，它会自动替换宕机或故障的实例。有关更多信息，请参阅[部署](https://kubernetes.io/zh/docs/concepts/workloads/controllers/deployment/)。 |
| 有状态副本集（StatefulSet） | 有状态副本集是用于管理有状态应用程序的工作负载对象，例如 MySQL。有关更多信息，请参阅[有状态副本集](https://kubernetes.io/zh/docs/concepts/workloads/controllers/statefulset/)。 |
| 守护进程集（DaemonSet） | 守护进程集管理多组容器组副本，确保 Pod 的副本在集群中的一组节点上运行。有关更多信息，请参阅[守护进程集](https://kubernetes.io/zh/docs/concepts/workloads/controllers/daemonset/)。 |
| 任务（Job） | Job 是需要运行完成的确定性的或批量的任务，⽤于管理仅运⾏⼀次或周期性运⾏的容器组。有关更多信息，请参阅[任务](https://kubernetes.io/zh/docs/concepts/workloads/controllers/job/)。 |
| 定时任务（CronJob） | 定时任务是需要在特定的时间运行，或在指定的时间间隔内重复运行的批处理任务。有关更多信息，请参阅[定时任务](https://kubernetes.io/zh/docs/concepts/workloads/controllers/cron-jobs/)。 |
| 服务（Service） | 将运行在容器组上的应用程序公开为网络服务，提供了固定的地址（域名或 IP 地址）供客⼾端访问。有关更多信息，请参阅[服务](https://kubernetes.io/zh/docs/concepts/services-networking/service/)。 |
| NodePort | 通过每个节点上的 IP 和静态端口（NodePort）暴露服务，可通过<节点 IP>:<NodePort>方式来访问服务。 |
| LoadBalancer | 使用云服务商提供的负载均衡器向外部暴露服务。 |
| 应⽤路由（Ingress） | 应⽤路由⽤于对服务进⾏聚合并提供给集群外部访问。每个应⽤路由包含域名及其⼦路径到不同服务的映射规则。KubeSphere 应用路由对应 Kubernetes 中的 Ingress。 |

存储

| 术语 | 说明 |
| --- | --- |
| 存储卷 | 一个基础资源对象，用于向容器提供存储。 |
| 存储类（Storage Class） | 定义可供容器使⽤的存储卷类型。 |
| 持久卷声明（Persistent Volume Claim, PVC） | 持久卷声明是用户对于存储需求的一种声明，它是命名空间里的资源，声明信息中可以指定存储大小、访问模式等。系统根据持久卷声明创建持久卷。 |
| 持久卷（Persistent Volume, PV） | 根据持久卷声明中的参数，在后端存储系统中创建的可供容器使⽤的存储区域。它是通用的、可插拔的、并且不受单个 Pod 生命周期约束的持久化资源。 |
| 卷快照类 | 定义可保存快照数据的⼀类卷快照。 |
| 卷快照 | 卷的数据在某一个时刻的完整拷贝或镜像。可通过快照将数据完整地恢复到快照时间点。 |
| 卷快照内容 | 根据卷快照中的参数，在后端存储系统中保存的快照数据。 |

DevOps

| 术语 | 说明 |
| --- | --- |
| DevOps 项目 | DevOps 项目用于创建和管理流水线和凭证。 |
| SCM (Source Control Management) | 源控制管理，例如 GitHub 和 Gitlab。 |
| In-SCM | 通过 SCM 工具构建基于 Jenkinsfile 的流水线。 |
| Out-of-SCM | 通过图形编辑面板构建流水线，无需编写 Jenkinsfile。 |
| CI 节点 | 流水线、S2I 和 B2I 任务的专用节点。一般来说，应用程序往往需要在构建过程中拉取多个依赖项，这可能会导致如拉取时间过长、网络不稳定等问题，从而使得构建失败。为了确保流水线正常运行并加快构建速度（通过缓存），您可以配置一个或一组 CI 节点以供 CI/CD 流水线和 S2I/B2I 任务专用。 |
| B2I (Binary-to-Image) | B2I 是一套从二进制可执行文件（例如 Jar 和 War 等）构建可再现容器镜像的工具和工作流。开发者和运维团队在项目打包成 War 和 Jar 这一类的制品后，可快速将制品或二进制的 Package 打包成 Docker 镜像，并发布到 DockerHub 或 Harbor 等镜像仓库中。 |
| S2I (Source-to-Image) | S2I 是一套从源代码构建可再现容器镜像的工具和工作流。通过将源代码注入容器镜像，自动将编译后的代码打包成镜像。在 KubeSphere 中支持 S2I 构建镜像，也支持以创建服务的形式，一键将源代码生成镜像推送到仓库，并创建其部署和服务最终自动发布到 Kubernetes 中。 |

日志、事件和审计

| 术语 | 说明 |
| --- | --- |
| 日志 | 日志是集群或应用程序记录的事件列表。 |
| ⽇志接收器 | 收集系统的各类⽇志，包括：容器⽇志、资源事件、审计⽇志。 |
| 审计策略 | 审计策略定义事件记录和所含数据的一系列规则。 |
| 审计规则 | 审计规则定义如何处理审计日志。 |
| 审计 Webhook | Kubernetes 审计日志会发送至审计 Webhook。 |

网络

| 术语 | 说明 |
| --- | --- |
| 网关（Gateway） | 为服务提供反向代理。⽹关根据应⽤路由中定义的规则将业务流量转发给不同的服务。 |
| ⽹络策略 | ⽤于控制集群中容器组的访问和被访问权限。可以只允许容器组访问特定的其他容器组或⽹段；只允许容器组被特定的其他容器组或⽹段访问。 |
| 容器组 IP 池 | 包含多个虚拟 IP 地址，⽤于为容器组分配虚拟 IP 地址。每个容器组 IP 池包含⼀个可在集群内部访问的私⽹ IP ⽹段。 |

监控、告警和通知

| 术语 | 说明 |
| --- | --- |
| 告警规则组 | 用于在特定监控指标满⾜预设条件和持续时间时⽣成告警。 |
| Prometheus | 负责监控存储系统的各项数据，根据告警规则向告警管理器发送告警信息。 |

其他

| 术语 | 说明 |
| --- | --- |
| 污点（Taint） | ⽤⼾在节点上创建的标记，由键、值和效果三部分组成。与容器组上创建的容忍度配合使⽤，以确保不会将 Pod 调度到不适合的节点上。 |
| 容忍度（Toleration） | 容忍度表示允许将 Pod 调度到具有对应污点的节点或节点组上。由键、值和效果三部分组成。容忍度和污点共同作用可以确保不会将 Pod 调度在不适合的节点上。 |
| 标签（Label） | 标签是为对象设置的可标识的键值对，通常用来管理和选择对象子集。 |
| 注解（Annotation） | 注解是以键值对的形式给资源对象附加随机的无法标识的元数据。 |
| 会话保持 | 将同⼀个会话中来⾃同⼀个客⼾端的请求全部转发给同⼀个容器组。 |
| 保密字典（Secret） | 包含 Base64 编码的键值对，⽤于存储密码、令牌、密钥等保密数据。 |
| 配置字典（ConfigMap） | 以键值对的形式存储环境变量、命令⾏参数和配置⽂件等⾮保密数据。 |
| 服务账户（ServiceAccount） | 存储当前集群的访问信息，⽤于向集群内外的应⽤程序提供集群的访问权限。 |
| 定制资源定义（CustomResourceDefinition） | 使⽤定制资源定义创建定制资源。通过定制化的代码给 API 服务器增加资源对象，而无需编译完整的定制 API 服务器。 |