

kubernetes

NSD

DAY01

kubernetes概述

k8s概述

- 它是一个全新的基于容器技术的分式架构领先方案
- 谷歌10几年以来大规模应用容器技术的重要成果
- Borg的一个开源版本
- 不限于任何语言，可以通过任何语言编写服务



k8s组成

- master节点
 - 集群控制节点，基本上所有的控制命令都是发给它，它来负责具体的执行过程
- node节点
 - 早期版本称作Minion
 - 集群中的工作负载节点



master的进程

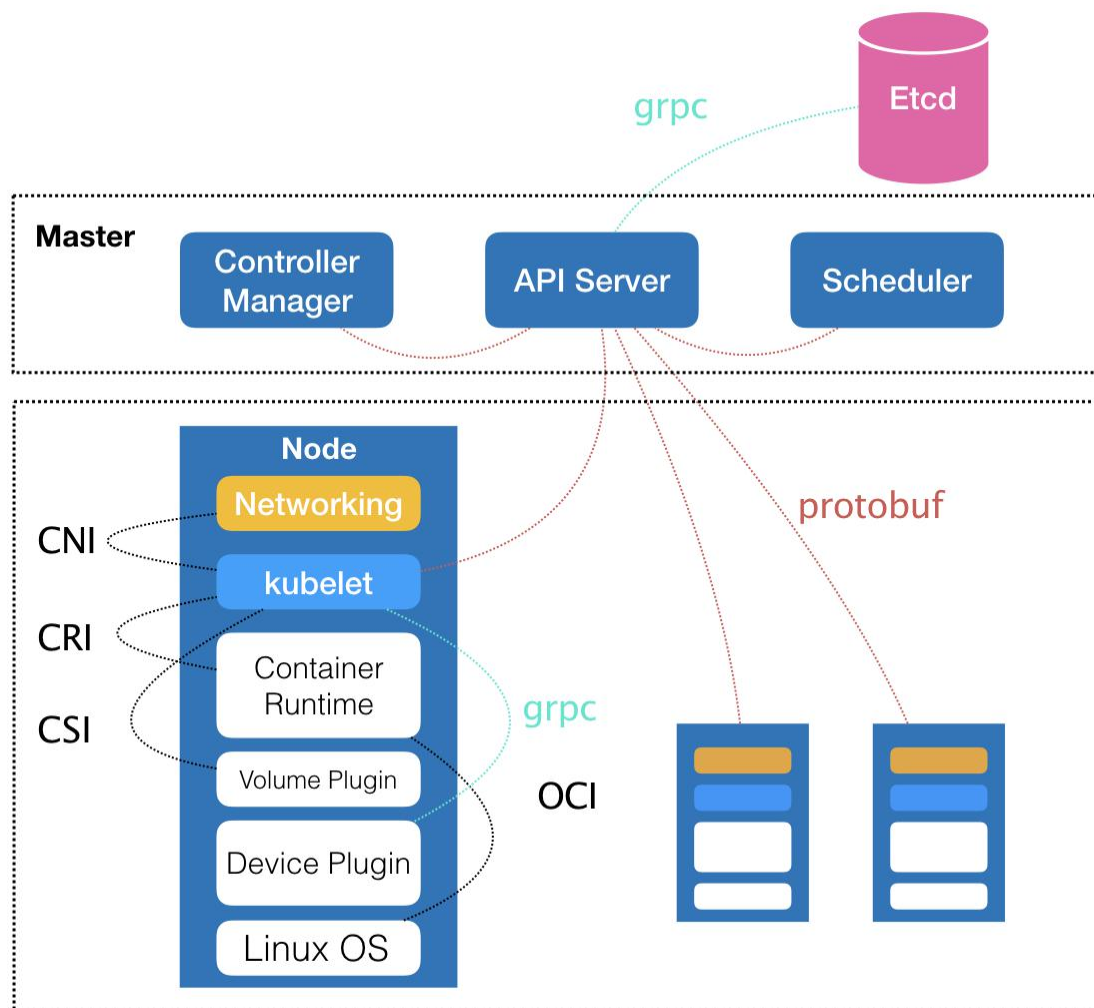
- Kubernetes API server
 - 所有资源增删改查的唯一入口
- Kubernetes Controller Manager
 - 所有资源对象的自动化控制中心
- Kubernetes Scheduler
 - 负责资源调度



node的进程

- Kubelet
 - 负责pod对应容器的创建、启停等
- Kube-proxy
 - 实现Kubernetes service的通信与负载均衡机制的重要组件
- Docker Engine
 - Docker进程，负责本机容器的创建与管理





pod

- 由若干容器组成的一个容器组
- 同个组内的容器共享一个存储卷(volume)
- 每个Pod被分配到节点上运行直至运行结束或被删除
- 同个Pod下的容器使用相同的网络命名空间、IP地址和端口区间，相互之间能通过localhost来发现和通信



service

- Kubernetes Service 定义了这样一种抽象：逻辑上的一组 Pod，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
- 对非 Kubernetes 集群中的应用，Kubernetes 提供了基于 VIP 的网桥的方式访问 Service，再由 Service 重定向到 backend Pod
- 在 Kubernetes 集群中，每个 Node 运行一个 kube-proxy 进程。kube-proxy 负责为 Service 实现了一种 VIP（虚拟 IP）的形式



标签和选择器

- 为了建立service和pod间的关联，k8s先给每个pod打上一个标签（Label）。如给mysql的pod打上name=mysql标签
- 然后再给相应的service定义标签选择器（Label Selector），如mysql service标签选择器的选择条件为name=mysql



RC/RS/Deployment

- RC: Replication Controller，它定义了一个期望的场景，即声明某种pod的副本数量在任意时刻都符合某个预期值，包括以下几个部分：
 - 期待的pod副本数
 - 用于筛选目标pod的Label Selector
 - 副本数不足时，用于创建新pod的pod模板
- RS和Deployment逐步替换了rc的作用，在功能上做了进一步的升级



总结和答疑
