**K8s学习**

# K8s发展历史

## 资源管理器对比

1. Mesos

采用apache开源协议，开源的分布式的资源管理框架，在2019.5不再使用，转向k8s

1. Docker swarm

分布式资源管理框架，针对docker容器化，在2019.7阿里云不再使用docker swarm

1. K8s

是Google旗下采用go语言基于borg系统开发的一款资源管理器kubernets

## K8s的特点及优势

1. 轻量级：消耗资源小
2. 开源
3. 弹性伸缩：服务器动态增减而不需要重启服务
4. 负载均衡：采用IPVS

# K8s组件说明

## borg系统架构

|  |
| --- |
| 1575710933(1) |

Borg系统架构组件说明

1. BorgMaster：负责请求分发，为了防止BorgMaster挂了，borgMaster集群，节点数量 为单数
2. Borglet：真正执行请求
3. 请求方式：web browsers(浏览器)、command-line tools(命令行)、borgcfg(文件的读取)

当通过这些方式去发起请求的时候，会先经过BorgMaster进行请求分发，然后到Borglet 去执行请求

1. Scheduler(调度器)：scheduler讲数据写入paxos数据库(google的一个键值对数据库)， borglet会实时的去paxos数据库里面进行监听，如果发现有对应的请求，就会获取并进 行执行

## K8s架构

|  |
| --- |
| 1575711693(1) |

k8s架构组件说明

1. scheduler(调度器)：负责接受任务，选择合适的节点进行分配任务
2. replication controller：维护副本的期望数目
3. api server：所有服务访问的统一入口
4. etcd：一个可信赖的(不会存在单点故障，天生支持集群化)分布式键值(kv)存储服务， 它能够为整个分布式集群存储一些关键数据，协助分布式集群的正常运转。

k8s使用etcd作为持久化方案(v2,v3两个版本)，v2版本已在k8s版本v1.11中弃用

|  |
| --- |
| 1575712254(1) |

etcd内部架构图

1. 采用http协议，k8s也是采用http协议，因为http协议支持多种操作方式
2. Ralt：存放所有读写信息，并且为了防止信息出现损坏，会将日志信息写入本地磁 盘中(store)
3. node节点：node节点需要安装三个组件(kubelet：和docker进行交互，操作docker 创建对应的容器即实现容器的生命周期管理、kube proxy：负责写入规则至iptables或者ipvs来实现服务映射访问、docker)

## K8s其它插件说明

1. CoreDNS：可以为集群中的svc创建一个域名ip的对应关系解析
2. Dashboard：给k8s提供一个B/S结构的访问体系
3. Ingress Controller：官方只能实现4层代理，Ingress可以实现7层代理
4. Federation：提供一个可以跨集群中心多k8s统一管理功能
5. Prometheus(普罗米修斯)：提供一个k8s集群的监控能力
6. Elk：提供k8s集群日志统一分析接入平台

# k8s基础概念