



链滴

# Kubernetes 基础概念

作者: [etscript](#)

原文链接: <https://ld246.com/article/1563593065997>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

# 本文主要参考了以下几篇文章，将一些基础的概念整理了一下

# Kubernetes架构

# Kubernetes指南

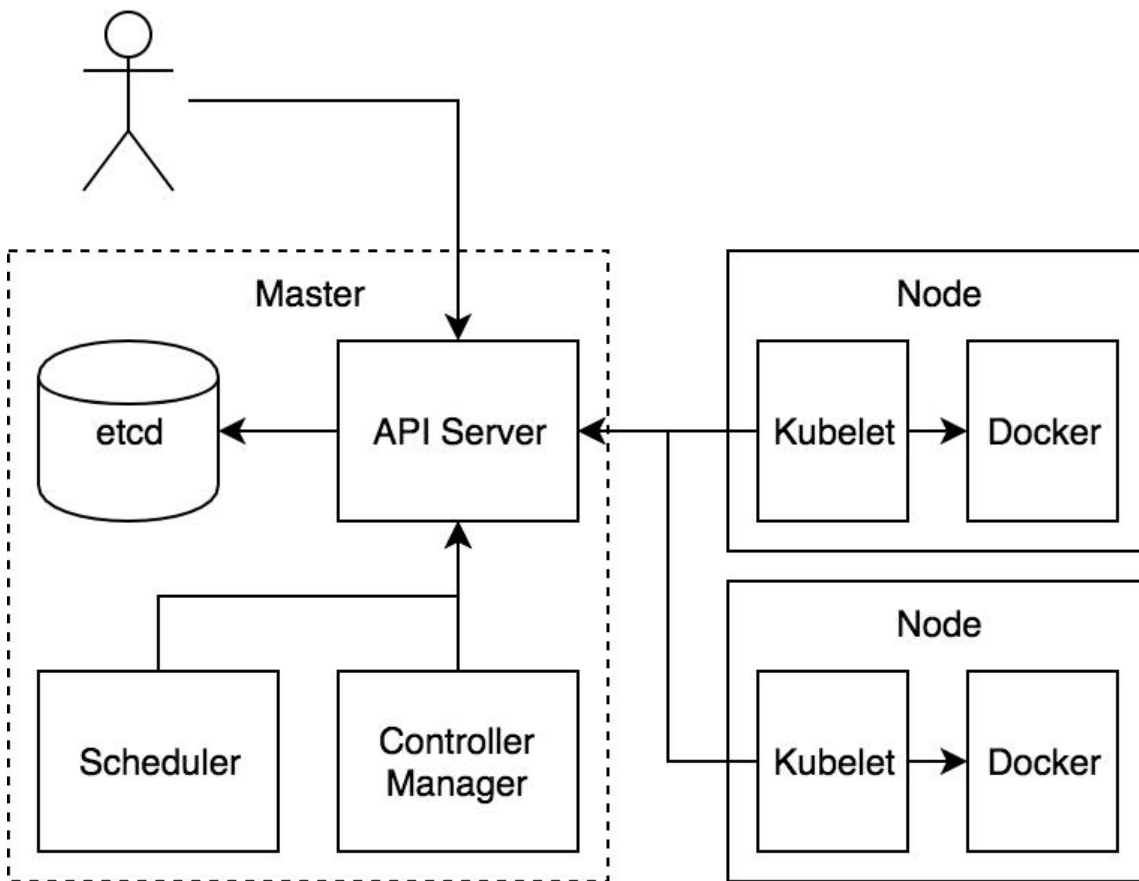
# 不完美的 K8S 与阿里的解决之道

之前我们快速部署了Kubernetes，对于Kubernetes架构没有涉及，这里对基础架构整理一下：

1. 架构是个啥样子
2. 一般都有点啥、有啥用
3. 组件间大概的流程

## 1. 架构是个啥样子：

Kubernetes 大概的架构如下图所示，Master+Slave（两个Node）：



## 2. 一般都有点啥、有啥用：

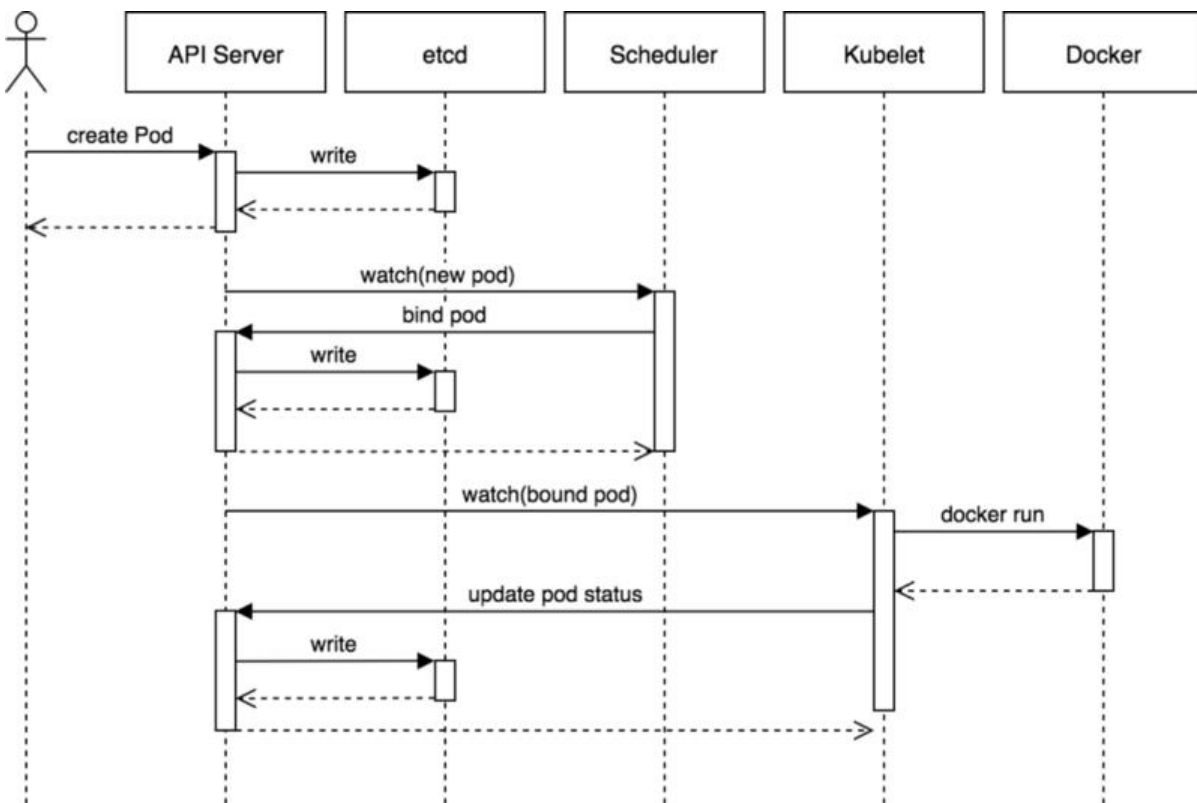
- etcd 保存了整个集群的状态；
- kube-apiserver 提供了资源操作的唯一入口，并提供认证、授权、访问控制、API 注册和发现等机；
- kube-controller-manager 负责维护集群的状态，比如故障检测、自动扩展、滚动更新等；
- kube-scheduler 负责资源的调度，按照预定的调度策略将 Pod 调度到相应的机器上；

- kubelet 负责维持容器的生命周期，同时也负责 Volume (CVI) 和网络 (CNI) 的管理；
- Container runtime 负责镜像管理以及 Pod 和容器的真正运行 (CRI) ， 默认的容器运行时为 Docker；
- kube-proxy 负责为 Service 提供 cluster 内部的服务发现和负载均衡；
- kube-dns 负责为整个集群提供 DNS 服务

### 3. 组件间大概的流程

比如创建一个 Pod 的流程为：

1. 用户通过 REST API 创建一个 Pod
2. apiserver 将其写入 etcd
3. scheduler 检测到未绑定 Node 的 Pod，开始调度并更新 Pod 的 Node 绑定
4. kubelet 检测到有新的 Pod 调度过来，通过 container runtime 运行该 Pod
5. kubelet 通过 container runtime 取到 Pod 状态，并更新到 apiserver 中



点到为止，不过度深入，想再了解可以看下开头提到的三篇文章

看到文章的最好进[博客](#)看文章哦，体验应该是最最好的