# K8s

## Tips

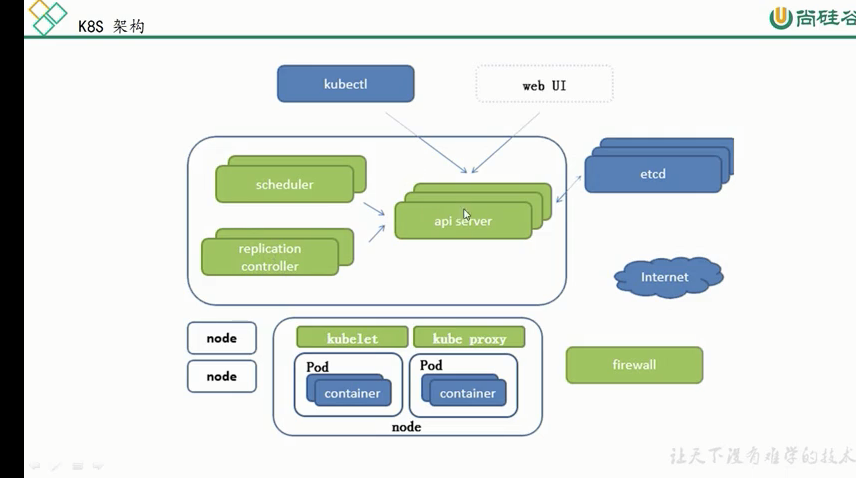
软路由是指利用台式机或服务器配合软件形成路由解决方案，主要靠软件的设置，达成[路由器](https://baike.baidu.com/item/%E8%B7%AF%E7%94%B1%E5%99%A8/108294" \t "/Users/yangwenshuo/Documents\\x/_blank)的功能；而[硬路由](https://baike.baidu.com/item/%E7%A1%AC%E8%B7%AF%E7%94%B1/6801739" \t "/Users/yangwenshuo/Documents\\x/_blank)则是以特有的硬设备，包括处理器、电源供应、[嵌入式软件](https://baike.baidu.com/item/%E5%B5%8C%E5%85%A5%E5%BC%8F%E8%BD%AF%E4%BB%B6/5345503" \t "/Users/yangwenshuo/Documents\\x/_blank)，提供设定的路由器功能。比如koolshare

路由器bai上的WAN口是用来连接外网du（公网），或者说是连接宽带运营商的设备的zhi。例如电话线dao上网时WAN口用来连接Moden（猫）；光钎上网时，WAN口用来连接光猫；网线入户上网时，WAN口用来连接入户网线。

而路由器上的LAN口（1、2、3、4），是用来连接内网（局域网）中的设备的，主要是用来连接电脑、交换机、打印机等设备的。

docker部署的主机是不让有虚拟内存的,不想让docker运行在虚拟内存里.

## 架构

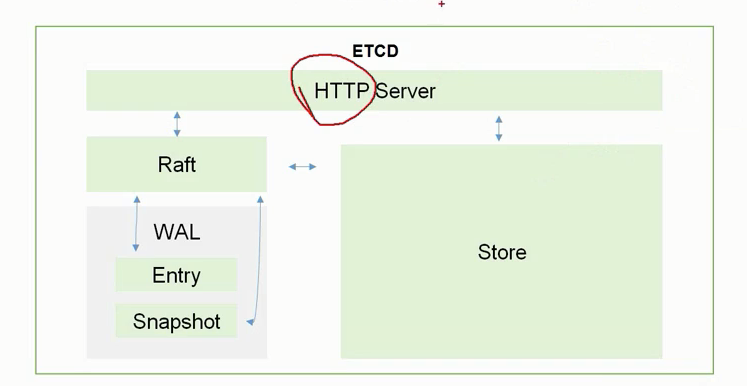


apiserver所有服务访问的入口.

Replication controller manager就是控制副本数的, 配了5个reply,死了一个这个要负责拉一个新的.

Schedule 调度器,负责接收任务,选择合适的节点进行任务分配.

Etcd可信赖的分布式键值存储服务,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.要用v3版本v2已经不行了



kubelet是维持pod的生命周期,叉棍将啥的.

Kube proxy 负责写入规则至iptables(防火墙)IPVS实现pod的映射.

Coredns可以为集群中的svc创建一个域名ip的对应关系解析

Dashboard 给k8s集群提供一个bs结构体访问体系.

Ingress controlier 七层代理.

Fedetation 提供一个跨集群中心的多k8s统一管理功能.

Prometheus 提供k8s集群的监控能力

Elk 集群日志.

## Pod控制器

pod共享网络栈,所以同一个pod端口不能冲突而且同一个pod中的一个容器想访问另一个容器直接写localhost:port就可以了.

还共享了硬盘卷.

### replicaSet

新版本中replicationcontroller被替换成了replicaSet(rs),这个更好一些,rs也可以维护你设置的容器数.

### Deployment

要想实现滚动更新和回滚,还需要一个东西叫做deployment,这个东西是控制rs的,

比如你想更新一个pod,他就会创建一个新的rs.

### Horizontal pod auto(hpa)

配合deployment和rs可以执行自动水平自动扩缩容,会根据cpu的使用率啥的扩缩容,但是再怎么缩也不会小于你设置的哪个.

### Statefulset

因为deployment和rs都是无状态的,那么需要存储的一些东西就需要一个有状态的东西存储.这个就是用来存储数据的.

### Demonset

确保全部或某些node上会想守护线程一样的运行一个pod,当有node加入集群时,也会为他们新增这些pod.

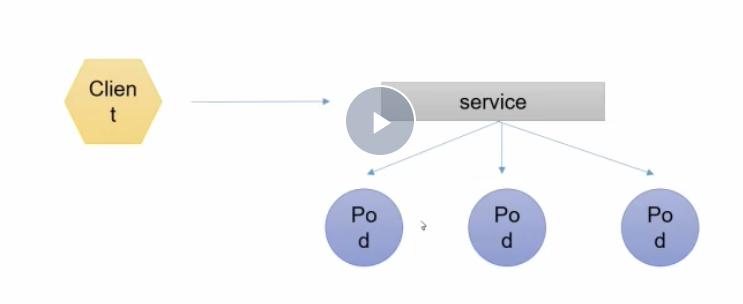
如果可以为每个node增加logstash,fluentd这种日志收集的pod.

还有什么监控的prometheus啥的.

### Cronjob

负责批处理任务.

## Service



同一个服务标签的pod可以组成一个service,service会自动的把相同服务标签的pod划在一起.service还可以做负载均衡.

## 网络通讯

K8s的网络模型假定了所有的pod都在一个可以直接联通的扁平网络空间中.

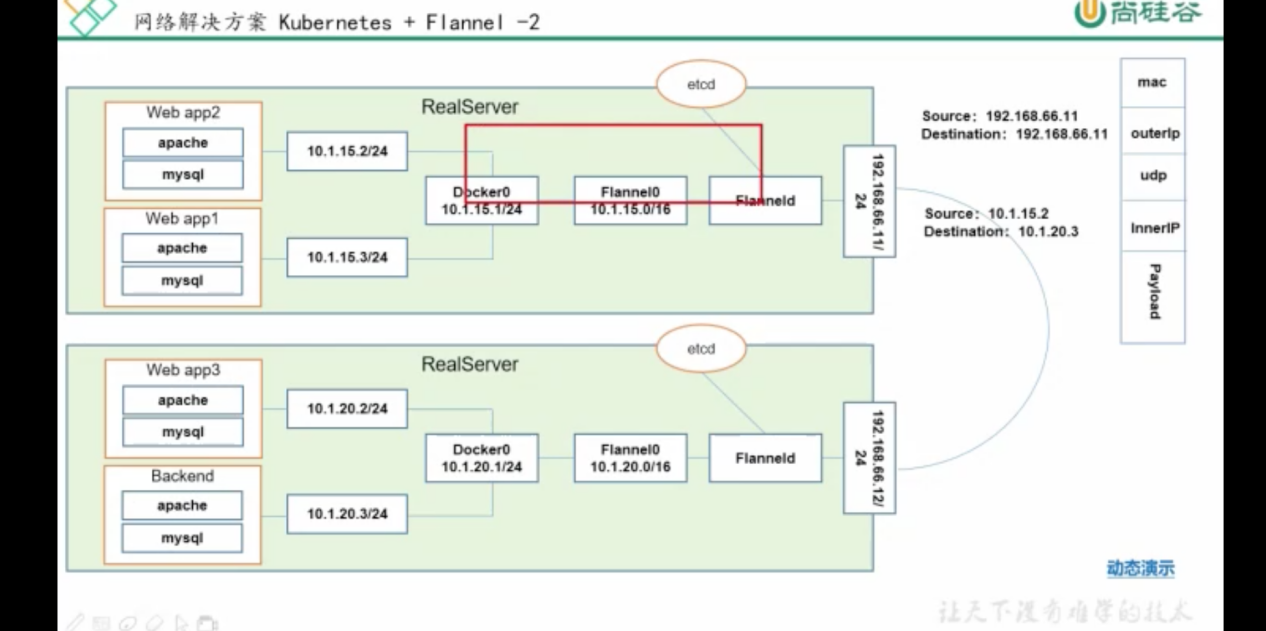
在私有云里面我们要自己实现这个网络假设,将不同节点上的docker容器之间的互相访问先打通.

是的examapi的docker和examinatiohn的docker要打通的.

各个pod之间的通信, overlay network

pod和service之间的通信, 个节点的iptables规则.

是针对k8s设计的一个网络规划服务,他的功能是让集群中的不同节点主机创建的docker容器都具有全集群唯一的虚拟ip地址.而且它还能在这个ip地址上创建一个网络,数据包原封不动的传递到目标容器.



首先docker0就类似于一个网关,pod1想要发消息到外面想发到docker0,然后flannel0类似一个网桥,监听docker0,然后flanneld监听flannel0,并且从etcd获取各个pod的路由信息,确定该发到哪里,然后把数据打包,发到对应的物理机器,接收物理机层层解包然后一路发送到对应的容器中.

etcd和flannel

etcd存储flanned可分配的ip地址段资源

flannel监控etcd各个pod的实际地址,维护pod节点的路由表.

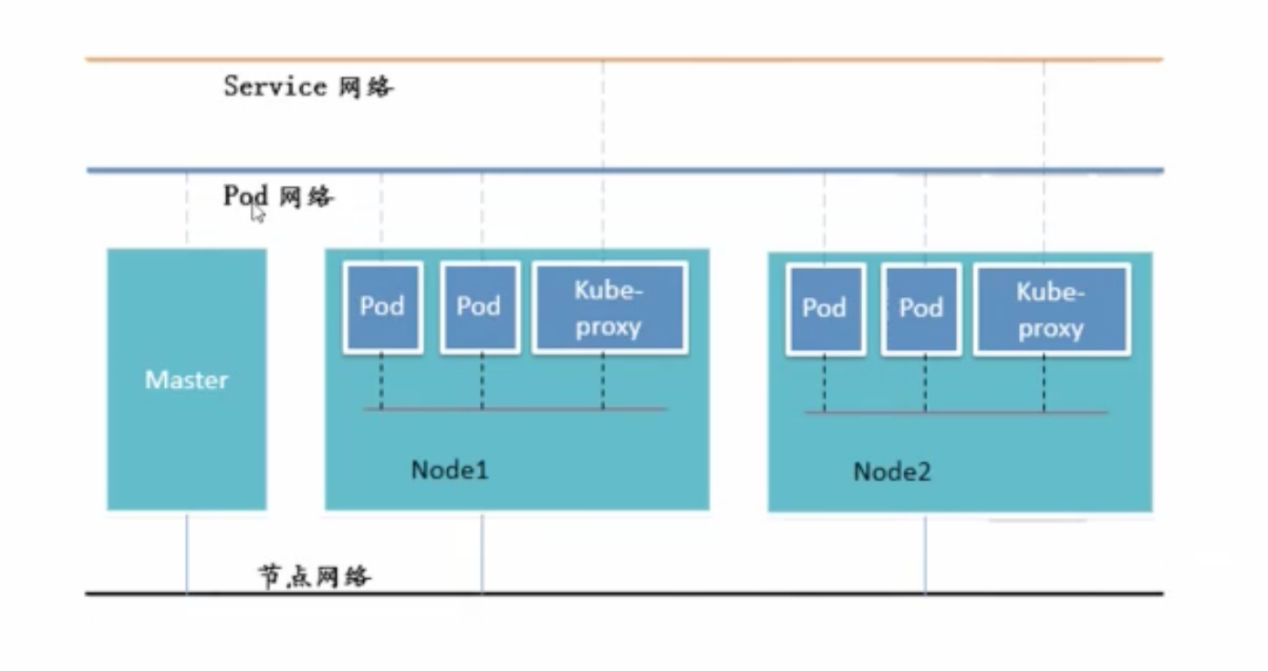
Docker0是可以维护同一个机器中通信的

想要跨机器就要加上flannel,这就是这两个的意义.

如果pod1和pod2在同一个物理机器,通过docker0网桥直接就可以通信.

pod到service的网络,用的iptables最新的用的lvc.

外面访问pod,通过service.



真实的物理ip的网络就是node网络,pod网络和service网络都是虚拟网络.

他们之间通过docker0 flannel和iptables lvs链接.