技术交流

微信：

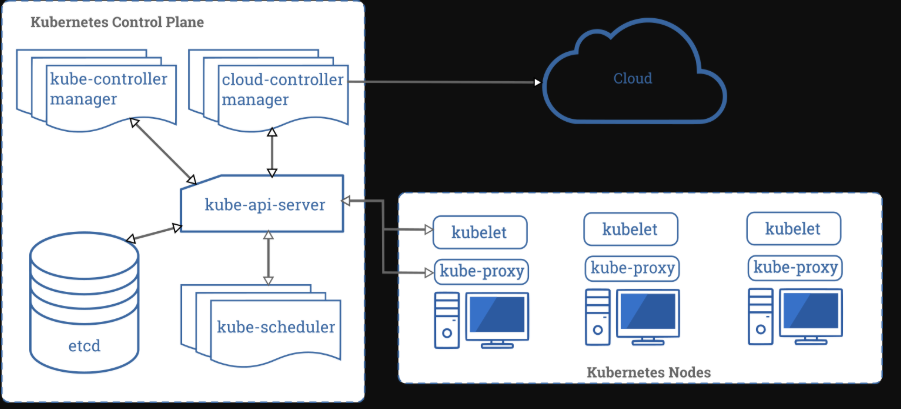
luckylucky421302

加微信备注k8s即可进群交流学习



一个Kubernetes集群是由一组工作机器组成，这些机器称为节点，在这些节点上运行容器部署的应用程序。 每个k8s集群至少有一个工作节点。工作节点主要用来运行部署应用程序的pod。 control plane（控制平面也可成为master节点）用来管理工作节点和整个kubernetes集群的pod。 在生产环境中，master节点通常跨越多台计算机，集群通常运行多个节点，提供容错和高可用性。下面内容会给大家介绍kubernetes集群中涉及到的所有组件。

下图把kubernetes所关联的一些组件都列了出来，可先看下图：



**kubernetes组件介绍**

**1.控制平面组件（也被称为master节点组件）**

控制平面的组件我们会找一台单独的机器来部署，我们习惯上把部署控制平面组件的机器称为master节点，以下都会用master节点来代替控制平面这个概念，master节点的组件能够对k8s的集群做出全局决策（例如，调度），以及检测和响应集群事件(例如，当部署的副本字段不满足时启动一个新的POD)。Master节点组件可以在k8s集群中的任何机器上运行。 然而，为了简单起见，通常会在同一台机器上启动所有控制平面组件，这台机器上最好不运行其他的容器化程序，所以我们就把专门部署控制平面组件的集群称为master节点。

**1）kube-apiserver**

kube-apiserver是Kubernetes master节点的组件，它公开了Kubernetes API。 API服务是Kubernetes master节点的前端。Kubernetes API服务是通过kube-apiserver组件实现的，kube-apiserver被设计成可以进行自动扩缩容，你可以运行多个kube-apiserver组件，通过keepalive+lvs或者其他负载均衡策略在这些组件之间平衡流量。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注册和发现等机制，负责接收、解析、处理请求。

**2）kube-scheduler**

kube-scheduler是kubernetes master节点的组件，用来监视已经被创建但是没有调度到node节点的pod，然后选择一个node节点用来运行它，kube-scheduler主要是负责pod的调度，按照预定的调度策略（如亲和性，反亲和性等）将Pod调度到相应的机器上。

**3）kube-controller-manager**

控制器管理器，用来检测控制器健康状态的，控制器是负责维护集群的状态，检查pod的健康状态，比如故障检测、自动扩展、滚动更新等一些操作。

**4）etcd**

etcd是一个key/value形式的键值存储，保存了整个kubernetes集群的状态，在kubernetes中使用etcd时，需要对etcd做备份，保证高可用。整个kubernetes系统中一共有两个服务需要用到etcd，用etcd来协同和存储配置，分别是：

（1）网络插件calico、对于其它网络插件也需要用到etcd存储网络的配置信息

（2）kubernetes本身，包括各种对象的状态和元信息配置

**注意：**网络插件操作etcd使用的是v2的API，而kubernetes操作etcd使用的v3的API，所以在下面我们执行etcdctl的时候需要设置ETCDCTL\_API环境变量，该变量默认值为2，表示使用v2版本的etcd api，v3表示使用v3版本的etcd api

**5）docker**

是一个容器引擎，用于运行容器

**6）kube-proxy**

k8s代理，是在集群中的每个节点上运行的网络代理，kube-proxy负责请求转发，一旦发现了某一个Service关联的Pod信息发生了改变（如IP、Port等），由Kube-Proxy就会把变化后的service转换成IPVS或IPtables规则中，完成对后端pod的负载均衡

**7）calico**

Calico是一个纯三层的网络插件，calico的bgp模式类似于flannel的host-gw，calico在kubernetes中可提供网络功能和网络策略

**8）cordns**

k8s1.11之前使用的是kube dns，1.11之后才有coredns，coredns是一个DNS服务器，能够为Kubernetes services提供DNS记录

**9）kubelet**

负责与master节点的apiserver进行通信的，接收到客户的请求，进行创建Pod，管理Pod，启动pod等相关操作

**2.工作节点组件（也被称为node节点组件）**

节点组件在k8s集群的每一个节点上都需要运行，工作节点用来运行业务pod，

**1）kubelet**

kubelet在k8s集群的每一个节点上都需要运行，属于节点组件，负责与master节点的apiserver进行通信的，接收到客户的请求，进行创建Pod，管理Pod，启动pod等相关操作

**2）kube-proxy**

k8s代理，是在群集中的每个节点上运行的网络代理，kube-proxy负责请求转发，一旦发现了某一个Service关联的Pod信息发生了改变（如IP、Port等），由Kube-Proxy就会把变化后的service转换成IPVS或IPtables规则中，完成对后端pod的负载均衡

**3）docker**

是一个容器引擎，用于运行容器

**4）calico**

Calico是一个纯三层的网络插件，calico的bgp模式类似于flannel的host-gw，calico在kubernetes中可提供网络功能和网络策略

**3.附加组件**

附件组件使用kubernetes的资源（如daemonset，deployment，statefulset等）实现集群功能，因为它们提供的是集群级别的属性，所以这些附件组件都需要部署到kube-system这个名称空间下。

**1）coredns**

k8s1.11之前使用的是kubedns，1.11之后才有coredns，coredns是一个DNS服务器，能够为 Kubernetes services提供 DNS记录。

**2）Web UI（Dashboard）**

Dashboard是k8s集群的一个web ui界面，通过这个界面可以对k8s资源进行操作，如创建pod，创建存储，创建网络等，也可以监控pod和节点资源使用情况。

**3）Ingress Controller**

七层负载均衡控制器，可以通过创建nginx或者traefik这种七层负载组件，实现域名和https访问。

**4）prometheus+alertmanager+Grafana**

监控系统，可以对kubernetes集群本身的组件监控，也可对物理节点，容器做监控，对监控到的超过报警阀值的数据进行报警，这个报警会发送到指定的目标，如钉钉，微信，qq，slack等。

**5）efk-（全称elasticsearch、fluentd、kibana）**

日志管理系统，可以对物理节点和容器的日志进行统一收集，把收集到的数据在kibana界面展示，kibana提供按指定条件搜索和过滤日志。

**6）metrics**

用于收集资源指标，hpa需要基于metrics实现自动扩缩容