**Kubernetes组件**

**Master组件**

提供集群的管理面板。用于做全局集群决策（计划表），并检测和回复集群事件（当部署的replicas属性未满足时启动新的pod）

可以允许在集群的任意机器上。但为了简化，通常在同一个机器上启动所有master组件，且不会在该机器上运行用户容器。

**Kube-apiserver**

该组件会暴露Kubernetes API，是一个Kubernetes控制面板的前端。设计用来水平扩展，即部署更多的实例。

**Etcd**

一致性和高可用的key-value存储，为集群数据提供存储支持。如果Kubernetes集群使用Etcd作为存储支持，要保证有备份。、

**Kube-scheduler**

该组件会观察新创建且未分配节点的pod，并为其选择一个节点。

调度决策所考虑的因素包括个人和集体资源需求，硬件/软件/策略约束，亲和力和反亲和性规范，数据位置，工作负载间干扰和最后期限。

**Kube-controller-manager**

运行controller的组件。

逻辑上，每个controller是分开的进程，但为了减少复杂度，都会在一个进程中编译和运行。

包含有：

* Node Controller，负责通知和node宕机。
* Replication Controller，负责为系统中的每个复制控制器对象维护正确数量的pod。
* Endpoints Controller，填充端点对象。
* Service Account & Token Controller，为新的命名空间创建账户和API令牌。

**Cloud-controller-manager**

负责与底层云支持交互，该组件的二进制是一个Alpha特性，在Kubernetes 1.6中介绍。

该组件仅运行cloud-provider-specific controller循环。你必须在Kube-controller-manager中禁用这些循环。也可以通过设置--cloud-provider=external。

该组件允许云供应商代码与Kubernetes代码独立开发。早期版本中，Kubernetes核心代码依赖于云指定的功能性代码。在未来版本中，云供应商的指定代码应有他们自己维护，在运行Kubernetes时连接到该组件上。

如下组件具有云提供商依赖：

* Node Controller，用于检查云提供程序，以确定节点在停止响应后是否已在云中删除.
* Route Controller，为底层云基础设备设置路径。
* Service Controller，用于创建，更新和删除云提供程序负载平衡器
* Volume Controller，用于创建，附加和装载卷，以及与云提供商交互以协调卷。

**Node组件**

Node组件会运行在每一个节点上，维护运行中的pod，为Kubernetes提供运行时环境。

**kubelet**

集群中运行在每个节点的代理，确保容器在pod中运行。

kubelet采用一系列通过各种机制提供的Pod Specs，并确保这些Pod Specs中描述的容器正在运行且健康，kubelet不会管理不由Kubernetes创建的容器。

**Kube-proxy**

Kube-proxy是一个网络代理，运行在每个节点上，实现了部分Kubernetes服务的概念。

Kube-proxy在节点上维护网络规则，允许在集群中通过会话在Pods间通信。

Kube-proxy会使用操作系统的包过滤层，若不存在则直接转发。

**Container Runtime**

该组件是负责响应运行容器的软件。

Kubernetes支持多个容器运行：Docker，containerd，cri-o，rktlet和任何基于Kubernetes CRI（Container Runtime Interface）的实现。

**Addons（插件）**

插件使用Kubernetes的资源来实现集群特性。因为提供了集群级别的特性，插件的命名空间资源属于kube-system命名空间。

已选插件如下，其他可用插件可参考列表

**DNS**

尽管其他插件不是一定需要的，所有Kubernetes的集群必须有cluster DNS。

Cluster DNS是一个DNS服务，除了在环境中的其他DNS服务外，它为Kubernetes提供DNS记录。

容器通过Kubernetes启动时自动加入该DNS服务进入DNS搜索列表。

**Web UI（Dashboard）**用于Kubernetes集群的web界面，允许用户管理和查错集群中的应用

**Container Resource Monitoring（容器资源监视器）**

记录了容器在中心数据库的通用时间序列的指标，并提供给UI浏览。

**Cluster-level Logging**

集群级日志的机制会保存容器日志到中心日志存储，该日志可以被搜索/浏览接口获取。