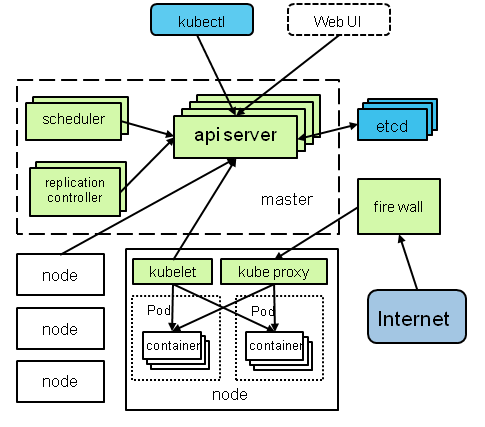
**kubernetes学习手册**

1. **服务**



**1.master：**

**api server:** 集群的核心组件，主要负责集群各功能模块之间的通信，集群内的功能模块通过Api server将信息存入到分布式文件系统etcd中与其他节点进行信息交互。

**scheduler:** scheduler是k8s的调度模块，做的事情就是拿到pod之后在node中寻找合适的进行适配这么一个单纯的功能。scheduler作为一个客户端，从apiserver中读取到需要分配的pod，和拥有的node，然后进行过滤和算分，最后把这个匹配信息通过apiserver写入到etcd里面，供下一步的kubelet去拉起pod使用。

**Controller Manager:**主要负责集群的故障检测和恢复的自动化，它内部的组件如下:   
 a.endpointController:定期关联service和pod，关联信息由endpoint负责创建和更新。   
 b.ReplicationController:完成pod的复制或移除，以确保ReplicationController定义的复本数量与实际运行pod的数量一致性

**2.node:**

**kube proxy:**负责为pod提供代理。它会定期从etcd获取所有的service，并根据service信息创建代理。当某个客户pod要访问其他pod时，访问请求会经过本机proxy做转发。

**kubelet:** kubelet的主要功能就是定时从某个地方获取节点上 pod/container 的期望状态（运行什么容器、运行的副本数量、网络或者存储如何配置等等），并调用对应的容器平台接口达到这个状态。

<https://www.2cto.com/net/201702/598560.html（kubelet详解）>

<http://blog.csdn.net/qq_36349607/article/details/53055208>（k8s各组件分析）

**二、源码分析**

**1.Makefile分析**

**1.1基础知识：**

Makefile 是和 make 命令一起配合使用的.很多大型项目的编译都是通过 Makefile 来组织的, 如果没有 Makefile, 那很多项目中各种库和代码之间的依赖关系不知会多复杂.

Makefile基本格式如下:

target ... : prerequisites ...

command

...

...

* target        - 目标文件, 可以是 Object File, 也可以是可执行文件
* prerequisites - 生成 target 所需要的文件或者目标
* command       - make需要执行的命令 (任意的shell命令), Makefile中的命令必须以 [tab] 开头

**1.2 GNU make工作方式**

* 读入主Makefile (主Makefile中可以引用其他Makefile)
* 读入被include的其他Makefile
* 初始化文件中的变量
* 推导隐晦规则, 并分析所有规则
* 为所有的目标文件创建依赖关系链
* 根据依赖关系, 决定哪些目标要重新生成
* 执行生成命令

DBG\_MAKEFILE：