**Master-Node通信**

该部分用于描述master与集群的通信路径。目的是允许用户自定义其安装以强化网络配置，以便群集可以在不受信任的网络（或云提供商上的完全公共IP）上运行。

**Cluster to Master（集群到Master）**

所有从集群到master的通信路径都在apiserver终止（其余master组件的设计不会暴露远程服务）。在经典部署方式下，apiserver启用一种或多种形式的客户端验证，配置在HTTPS端口（443）用于监听远程连接。一种或多种的授权应该激活，特别是允许匿名请求或服务令牌的情况下。

Nodes应该提供好集群的公共根证书，这样才能以合法客户端证书连接apiserver。

Pods若想要连接apiserver是十分安全的，初始化完成后，可以借助一个服务账号使Kubernetes会自动将公共根证书和合法bearer令牌注入到pod中。Kubernetes服务配置有一个虚拟IP地址，会通过kube-proxy重定向到apiserver的HTTPS端点。

Master组件同样通过安全端口与apiserver通信。所以，默认连接操作是默认安全的，并可以在不受信的公网上运行。

**Master to Cluster（Master到集群）**

从Master到集群的通信路径主要有两个。第一个是从apiserver到kubelet进程（每个节点上运行）。第二个是从apiserver通过apiserver的代理功能到任意node，pod或service。

**Apiserver to kubelet**

从apiserver到kubelet的连接主要用于：

* 为pods获取日志记录
* 通过kubectl附加到运行中的pods。
* 提供kubelet的转发端口功能

这些连接都会在kubelet的HTTPS端点终止。默认情况下，apiserver不会验证kubelet的服务证书，这会使连接受到中间人攻击，且在公网或不受信网络上运行也不安全。

为了验证连接，使用--kubelet-certificate-authority标识来提供apiserver的根证书绑定，用于验证kubelet的服务证书。

如果这不能实现，在apiserver和kubelet间搭建SSH隧道来避免连接处于不受信网络。

最后，kubelet的验证与授权功能应该激活来保证kubelet API的安全性。

**Apiserver to nodes, pods, and services**

该路径默认是纯粹的HTTP连接，所以没有验证也没有加密。可以在连接前加上https:前缀来使用HTTPS连接，但不会验证证书，也不会提供客户端证书，不提供任何完整性保证。在公网或不受信网络运行时不安全的。

**SSH Tunnels**

Kubernetes支持SSH隧道来保护Master到Cluster的通信路径。在此配置下，apiserver会为每个node初始化SSH隧道，并通过每个目的是kubelet，node，pod或service的请求。该隧道会保证请求不会暴露在外。

SSH隧道当前已经弃用，除非你明确了解在做什么否则不要使用。替代方案正在设计中。