##### 资料

好书：<https://www.servicemesher.com/istio-handbook/>

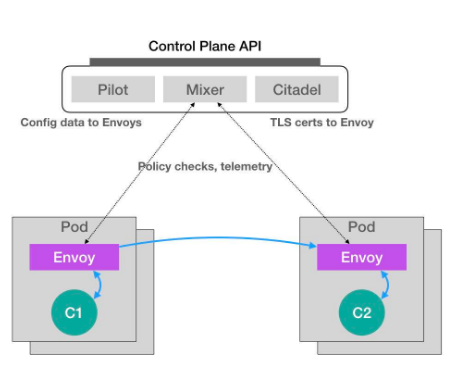
### 服务网格

几个特点：

* 位于应用程序间通讯的中间层
* 轻量级网络代理
* **应用程序无感知**
* 解耦应用程序的重试/超时、监控、追踪和服务发现

### 什么是Istio

架构：



简单的介绍：

逻辑上分为控制平面(Control Plane )和数据平面(Data Plane)。

官方的定义如下：

控制平面由Pliot、Mixer、Istio-Auth组成，负责管理和配置代理的路由流量以及运行时服务治理策略的执行。

数据平面由一组作为sidecar智能代理（Envoy）和它掌控网络通信的的微服务实例组成。

核心组件：

* Mixer：访问控制、遥测
* Pilot：对Envoy的生命周期进行管理，服务发现、流量管理（超时、重试、熔断）、高级路由
* Citadel：终端用户认证、流量加密
* Envoy：高性能轻量级代理，掌控了service的入口流量和出口流量，它提供了很多内置功能，如动态负载服务发现、负载均衡、熔断、健康检查等

最根本的组件，是运行在每一个应用Pod当中的Envoy容器。

Istio使用sidecar的方式在每一个应用Pod中都运行了这个代理服务。而Pod里的所有容器都共享同一个Network Namespace，所以Envoy可以配置Pod里的iptables规则，接管整个Pod的进出流量。

接下来，Istio的Control Plane中的Pilot组件，调用每个Envoy容器的API，对Envoy进行配置，实现微服务治理。

Envoy项目是Lyft公司推出的一个高性能C++代理