**1.Envoy能做什么**

Envoy 是一个高性能的边缘和服务代理，可以提供多种功能，包括：

**负载均衡**：Envoy 可以将流量分发给多个后端服务实例，从而提高服务可用性和吞吐量。

**流量控制**：Envoy 可以根据规则对流量进行路由、限速、熔断等控制，从而保证服务的可靠性和稳定性。

**服务发现**：Envoy 可以与多种服务发现机制集成，如 Consul、ETCD 等，自动发现服务实例并进行负载均衡。

**2.Istio能做什么**

虽然 Envoy 可以提供很多功能，但是它不像 Istio 那样提供完整的服务网格控制平面，因此缺少一些 Istio 提供的高级功能，如：

**统一的流量管理**：Istio 提供了一组流量管理规则和策略，可以对服务流量进行全局控制，例如路由、限速、熔断等。

如果仅使用 Envoy，**需要自己构建服务网格控制平面**，并**编写相应的规则和策略**来实现流量管理、安全性控制和监控等功能。

**3.Istio中哪个组件实现了统一流量管理**

Pilot 是 Istio 的控制平面组件之一，它负责管理和配置服务代理（例如 Envoy），以实现服务的流量管理、负载均衡和故障恢复等功能。

具体来说，当一个服务实例启动时，它会**向 Istio 注册自己**的元数据，例如服务名、服务版本、IP 地址等信息。**Pilot 会通过服务发现机制自动发现这些服务实例**，并将它们组织成一个服务拓扑结构。然后，Pilot 会**将服务的流量管理规则和策略转换成 Envoy 配置**，并**推送给 Envoy 代理**。

Envoy 代理会根据配置进行流量管理，例如根据服务名和版本进行负载均衡、路由、限速、熔断等操作。同时，**Envoy 代理会定期向 Pilot 报告自己的状态**，例如连接数、延迟、错误率等指标，以便 Pilot 进行实时的监控和故障恢复。

通过 Pilot 的统一配置管理，Istio 可以实现多种流量管理功能，例如：

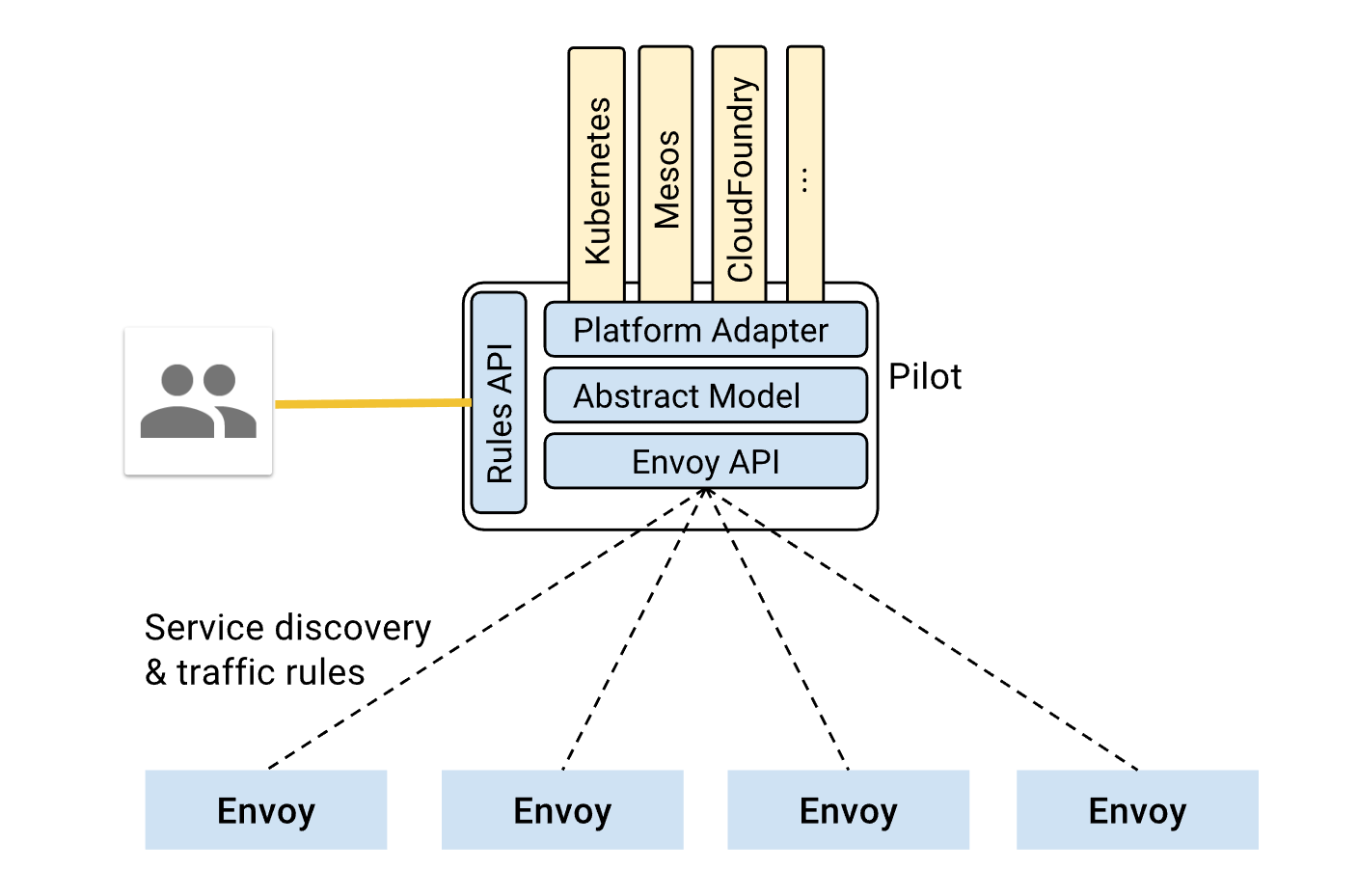
负载均衡：Istio 可以根据服务实例的健康状态和负载情况，动态地调整流量的分发策略，从而实现负载均衡的效果。

服务发现：Istio 可以自动发现服务实例，并将它们组织成一个服务拓扑结构，以方便进行流量管理和监控。

流量控制：Istio 可以根据规则对流量进行路由、限速、熔断等控制，从而保证服务的可靠性和稳定性。

安全性：Istio 支持 TLS 加密、证书认证、RBAC 等安全功能，可以保障服务通信的安全性。

可观测性：Istio 支持多种监控和日志系统集成，如 Prometheus、Zipkin、Jaeger 等，以便对服务的性能和健康状况进行监控和分析。



Envoy是小兵 Pilot是将军

将军知道都有哪些小兵（envoy注册，pilot发现）

元首有命令将军能听懂（用户通过 Istio 的 API 或命令行工具修改流量规则时，这些规则会被传递到 Pilot 组件中）

将军能把命令转化成士兵听的懂的命令告诉士兵（pilot将规则解析为envoy的配置文件）

士兵能够正确执行命令（解析后推送给envoy进行**动态**加载）

**4.Istio如何动态加载新的envoy配置文件**

在 Istio 中，Pilot 组件通过 Envoy 的 **xDS 协议**将配置信息推送给 Envoy 的 Sidecar 代理，而不需要重启 Envoy，因此不会导致服务中断。

**5.更笨的办法**

需要修改 Envoy 的配置文件，也可以使用 Envoy 的热重载功能，而不需要重启 Envoy。Envoy的热重载功能可以实现在不中断服务的情况下，重新加载新的配置文件。

Envoy 的热重载功能是通过启动两个独立的 Envoy 进程来实现的。一个进程负责接收和处理来自客户端的请求，另一个进程负责加载和应用新的配置文件。当新的配置文件加载完毕后，两个进程会进行一次无缝切换，新的进程成为接收和处理请求的主进程，旧的进程退出。在这个过程中，客户端不会察觉到任何中断或故障。

Envoy 的热重载功能具有以下优点：

零停机升级：Envoy 的热重载功能可以在不中断服务的情况下，重新加载新的配置文件，从而实现零停机升级。这种方式可以避免因服务中断而导致的不必要的损失。

动态规则修改：Envoy 的热重载功能可以实现动态修改服务的流量规则、负载均衡策略、安全控制等功能。例如，当您需要添加新的路由规则或修改负载均衡策略时，可以直接修改配置文件，并重新加载配置文件，从而使新的规则立即生效。

高可用性：Envoy 的热重载功能可以保证服务的高可用性和可靠性。在重新加载配置文件的过程中，Envoy 会启动两个独立的进程，一个负责接收和处理请求，另一个负责加载和应用新的配置文件。因此，在切换过程中，服务不会因为 Envoy 进程的中断或故障而出现服务不可用的情况。

总之，Envoy 的热重载功能是一种实现零停机升级、动态规则修改、高可用性的重要机制。通过使用热重载功能，可以更加灵活和可靠地管理和控制服务的流量。

**6.开发目标**

完成iptables配置，将入口流量转发给envoy

正确配置envoy，实现envoy流量代理功能（至少两版本）

实现envoy热重启功能（缺乏对envoy的探索）

基于xDS协议进行配置修改，取代热重启

规则->配置文件的对应方式（辅助前端页面，输入规则，修改）

模拟pilot统一进行规则修改

（哪个组件与所有envoy进行交互？它如何正确发现所有envoy并建立通信[监听k8s对象，比如pod，当含envoy新pod创建时建立通信]）

（这个组件如何与k8s进行交互？在k8s中以什么样的对象存在）

**参考资料**

<https://cloudnative.to/author/christian-posta/>

作者主页

https://cloudnative.to/blog/guidance-for-building-a-control-plane-to-manage-envoy-proxy-based-infrastructure/

如何为 Envoy 构建一个控制面来管理集群网络流量系列文章（go/java版本控制面板）

http://www.zhaowenyu.com/envoy-doc/intro/arch\_overview/hot\_restart.html

envoy热重启（还有很多其他资料）

<https://istio.io/v0.7/docs/concepts/traffic-management/pilot.html>

Istio v0.7版本的官方文档

<https://zhuanlan.zhihu.com/p/234428285>

Istio v1.7 部分源码分析（含xDS生成及下发梳理）

<https://blog.csdn.net/m0_67778103/article/details/123384980>

https://www.infoq.cn/article/Hpu9ifGK71ElKaptHwUN

网易基于 Envoy 的云原生网关实践