打造更好用的 Kubernetes Ingress Controller

apisix.apache.org

张晋涛

个人介绍

- 张晋涛
- Apache APISIX committer
- Kubernetes Ingress Nginx reviewer
- containerd/Docker/Helm/Kubernetes contributor
- 『K8s 生态周报』维护者
- https://github.com/tao12345666333

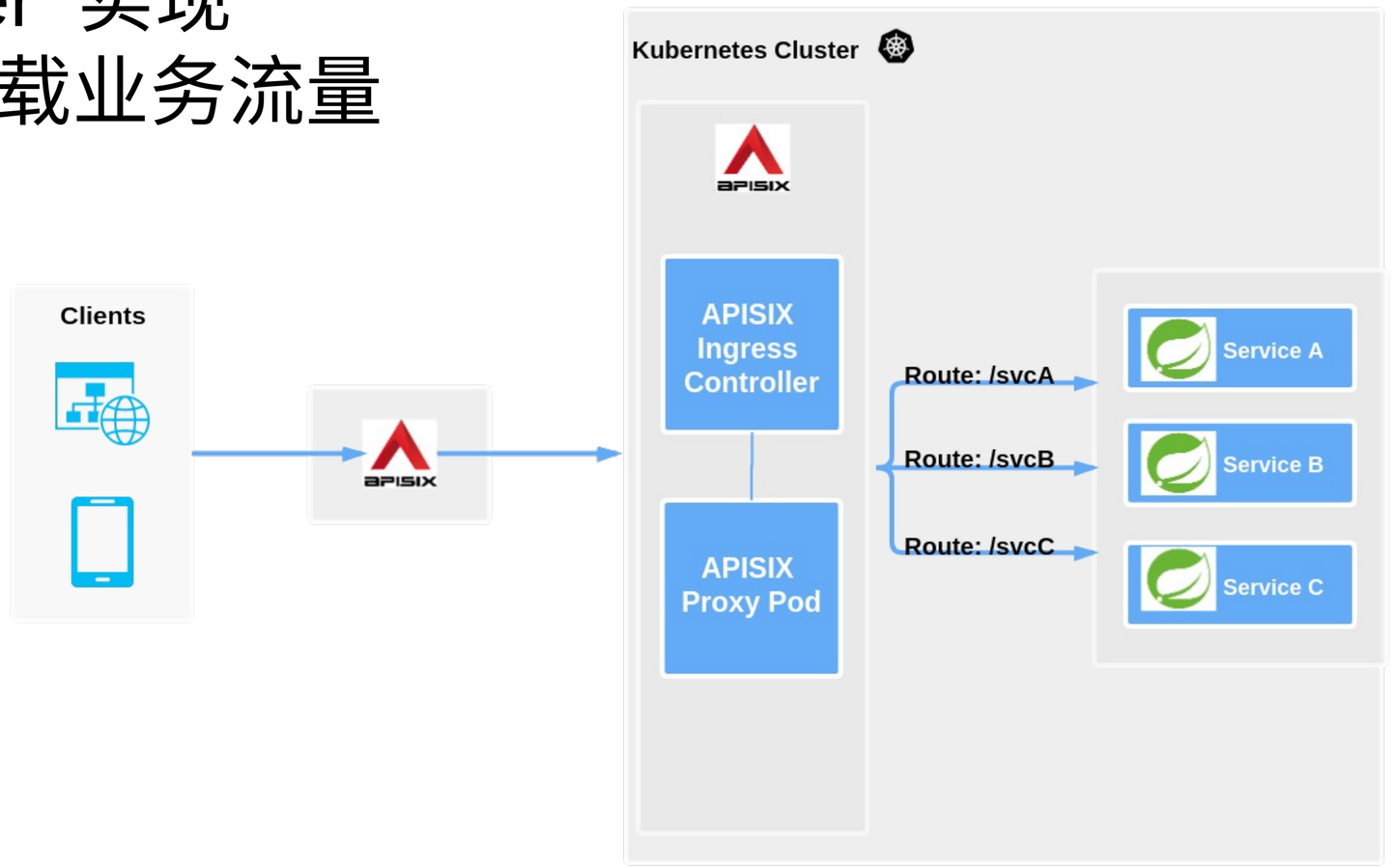
Agenda

- Apache APISIX Ingress 概览
- Apache APISIX Ingress 的设计
- Apache APISIX Ingress 典型应用场景
- Apache APISIX Ingress 的生态及后续规划

一、Apache APISIX Ingress 概览

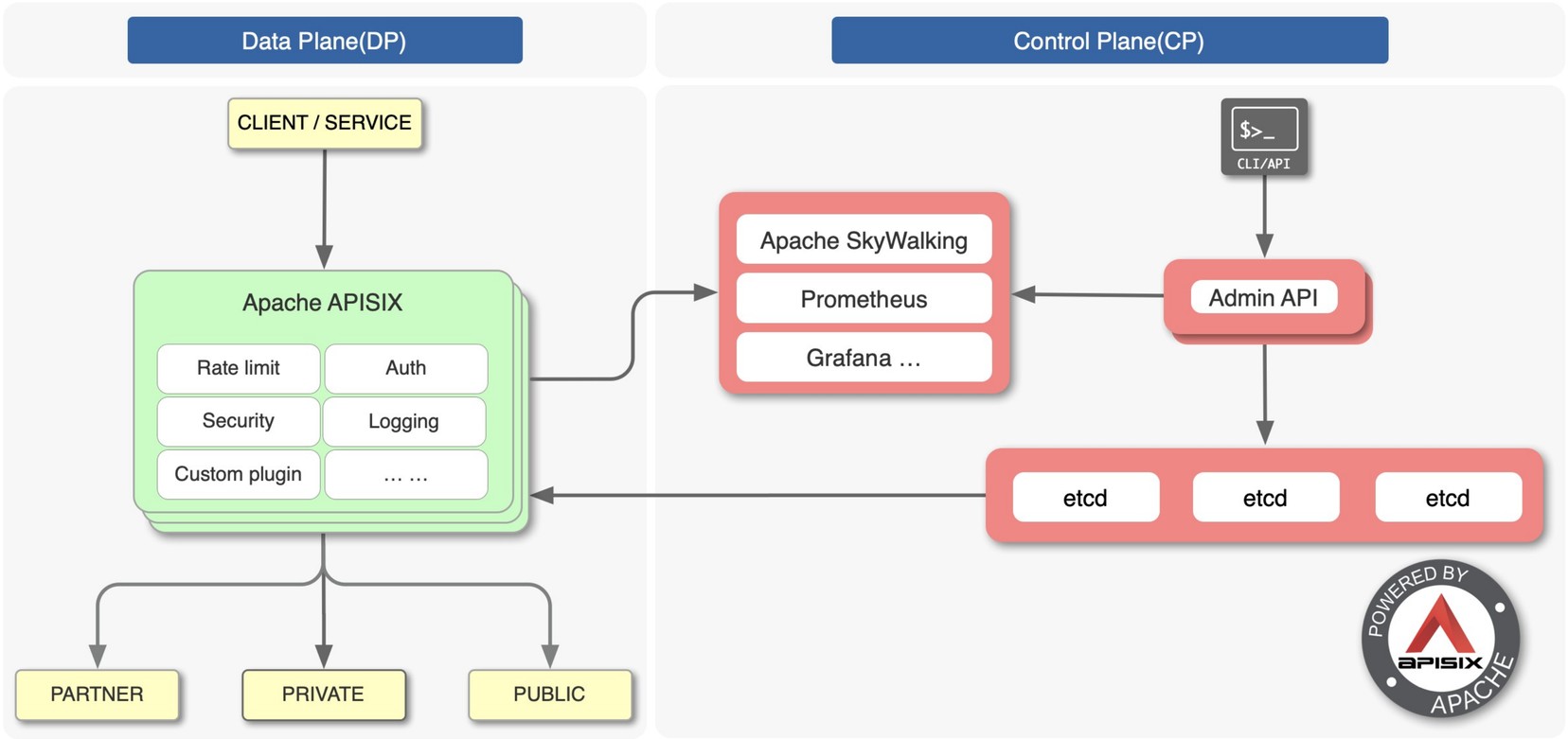
APISIX Ingress 是什么

- 另一种 Ingress Controller 实现
- 使用 Apache APISIX 承载业务流量
- 通过 CRD 进行实现
- 支持原生 Ingress 资源



Apache APISIX 是什么

- Apache 基金会顶级项目
- 最活跃的开源网关项目
- 全动态高性能网关
- 云原生网关



Apache APISIX Ingress 特性

- 全动态：路由、 SSL 证书、上游、插件...
- 支持 Custom Resource Definitions，更容易理解的声明式配置
- 支持 K8S 原生 Ingress 配置 （ v1 / v1beta1 ）
- 通过 annotation 的方式对 Ingress 能力进行扩展
- 服务自动注册发现，无惧扩缩容
- 更灵活的负载均衡策略
- 健康检查开箱即用

Apache APISIX Ingress 特性

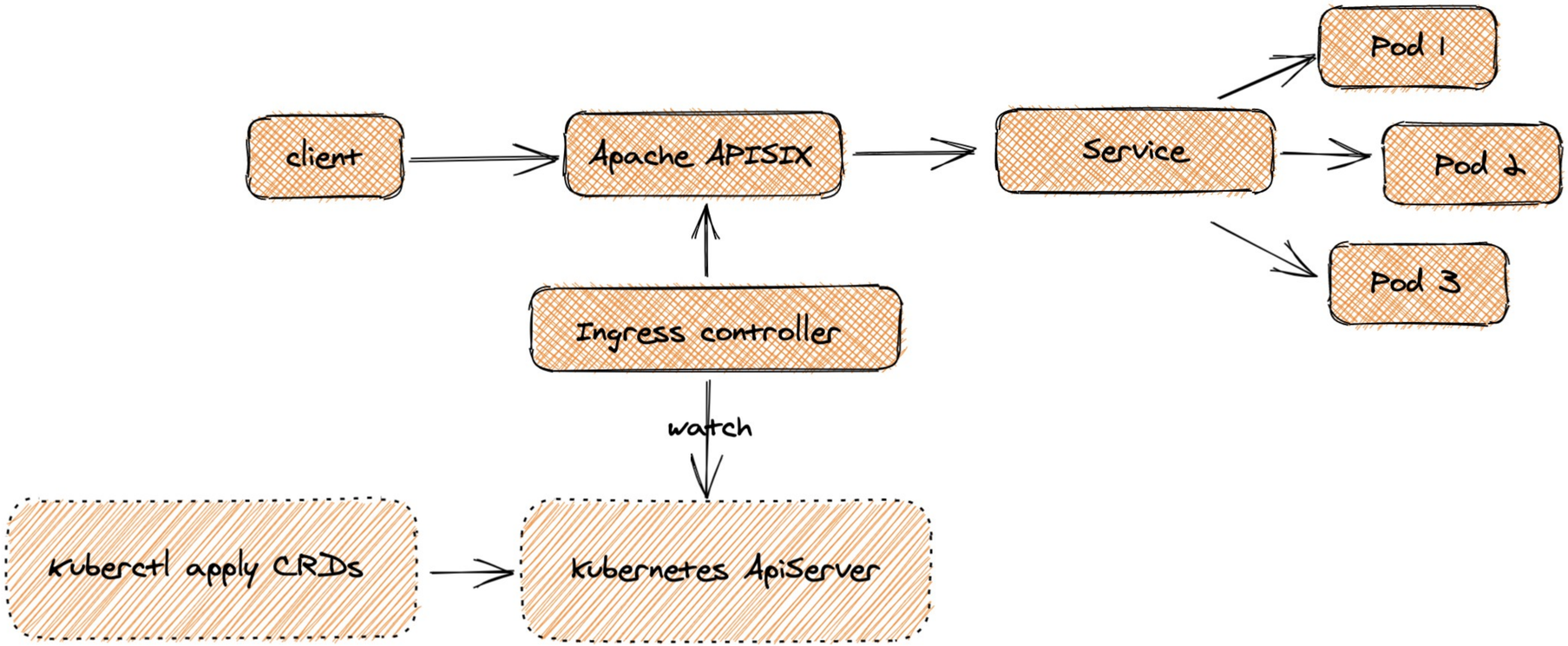
- 支持高级路由匹配规则
- 支持流量切分
- 支持 Apache APISIX 官方 50 多个插件 & 客户自定义插件
- gRPC plaintext 支持
- TCP 4 层代理
- 状态检查：快速掌握声明配置的同步状态
- ...

APISIX Ingress vs K8s ingress-nginx

- 单个配置实现灰度部署
- 丰富的插件和功能
- 数据面和控制面架构分离

概览

- 通过 CRD 进行扩展

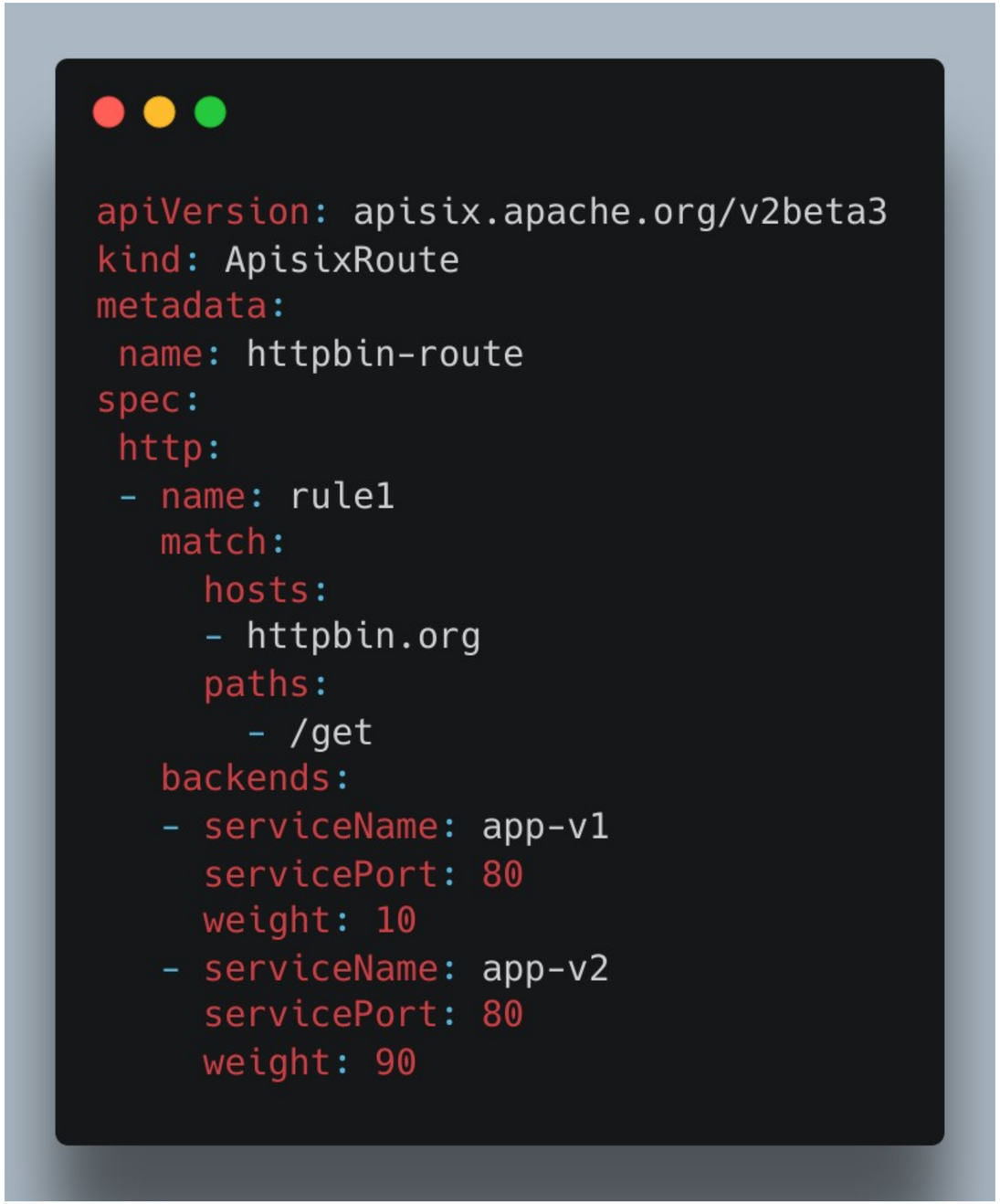


自定义资源

- ApisixRoute: 路由
- ApisixUpstream： 上游
- ApisixConsumer： 消费者
- ApisixTls： 证书相关
- ApisixClusterConfig： 集群公共配置
- ApisixPluginConfig： 插件配置

ApisixRoute

- Spec.http → 7 层代理
- Spec.stream → 4 层代理



```yaml
apiVersion: apisix.apache.org/v2beta3
kind: ApisixRoute
metadata:
 name: httpbin-route
spec:
 http:
 - name: rule1
   match:
     hosts:
     - httpbin.org
     paths:
       - /get
   backends:
   - serviceName: app-v1
     servicePort: 80
     weight: 10
   - serviceName: app-v2
     servicePort: 80
     weight: 90
```

ApisixUpstream

- name 与 backend （svc）一致
- Spec.loadbalancer： 负载均衡策略
- Spec.scheme： 协议，支持 http/https 和 gRPC/gRPCs
- Spec.healthCheck： 自定义健康检查

```yaml
apiVersion: apisix.apache.org/v2beta3
kind: ApisixUpstream
metadata:
  name: httpbin
spec:
  loadbalancer:
    type: ewma
  schema: http
  retries: 3
  healthCheck:
    active:
      type: http
      httpPath: /healthz
      healthy:
        httpCodes: [200]
        interval: 1s
      unhealthy:
        httpFailures: 2
        interval: 1s
```

ApisixConsumer

- Spec.authParameter： 支持 basicAuth 和 keyAuth
- 支持直接配置或使用 secret

```yaml
apiVersion: apisix.apache.org/v2beta3
kind: ApisixConsumer
metadata:
  name: basicvalue
spec:
  authParameter:
    basicAuth:
      secretRef:
        name: basic
```

ApisixTls

- Spec.hosts： 域名
- Spec.secret： 证书
- Spec.client： 用于 mTLS

```yaml
apiVersion: apisix.apache.org/v2beta3
kind: ApisixTls
metadata:
  name: sample-mtls
spec:
  hosts:
  - mtls.foo.bar
  secret:
    name: foo-secret
    namespace: apisix
  client:
    caSecret:
      name: foo-client-secret
      namespace: apisix
    depth: 10
```

ApisixClusterConfig

- 用于通用公共配置

```yaml
apiVersion: apisix.apache.org/v2beta3
kind: ApisixClusterConfig
metadata:
  name: default
spec:
  monitoring:
    prometheus:
      enable: true
```

ApisixPluginConfig

- 作为通用 Plugin 配置的集合
- 在 v1.4 中发布

```yaml
apiVersion: apisix.apache.org/v2beta3
kind: ApisixPluginConfig
metadata:
  name: cors-and-force-https
spec:
  plugins:
  - name: cors
    enable: true
  - name: redirect
    enable: true
    config:
      http_to_https: true
```

三、Apache APISIX Ingress 典型应用场景

流量切分

- 按比例进行流量切割

```yaml
apiVersion: apisix.apache.org/v2beta3
kind: ApisixUpstream
metadata:
  name: httpbin
spec:
  subsets:
  - name: v1
    labels:
      version: v1
  - name: v2
    labels:
      version: v2
```

```yaml
apiVersion: apisix.apache.org/v2beta3
kind: ApisixRoute
metadata:
  name: httpbin-route
spec:
  http:
  - name: rule1
    match:
      hosts:
      - httpbin.com
      paths:
      - /ip
    backends:
    - serviceName: httpbin
      servicePort: 80
      subset: v1
      weight: 90
    - serviceName: httpbin
      servicePort: 80
      subset: v2
      weight: 10
```

Basic Auth

- 对携带指定 header 访问路由的请求增加 basic auth

```yaml
apiVersion: apisix.apache.org/v2beta3
kind: ApisixConsumer
metadata:
  name: basicvalue
spec:
  authParameter:
    basicAuth:
      secretRef:
        name: basic
```

```yaml
apiVersion: apisix.apache.org/v2beta3
kind: ApisixRoute
metadata:
 name: httpbin-route
spec:
 http:
 - name: rule1
   match:
     hosts:
     - httpbin.org
     paths:
       - /ip
     exprs:
     - subject:
         scope: Header
         name: X-Foo
       op: Equal
       value: bar
   backends:
   - serviceName: httpbin
     servicePort: 80
   authentication:
     enable: true
     type: basicAuth
```

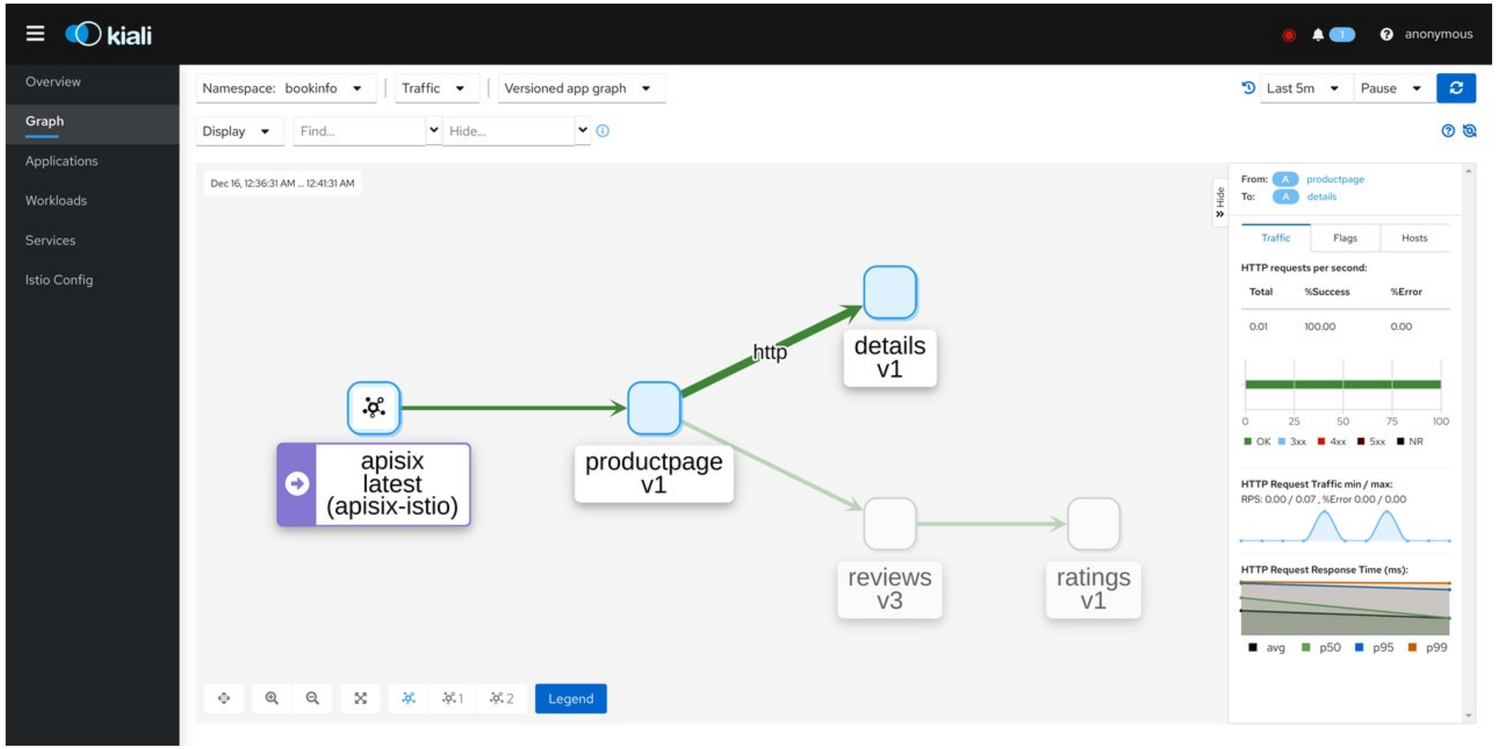
对 Ingress 的扩展

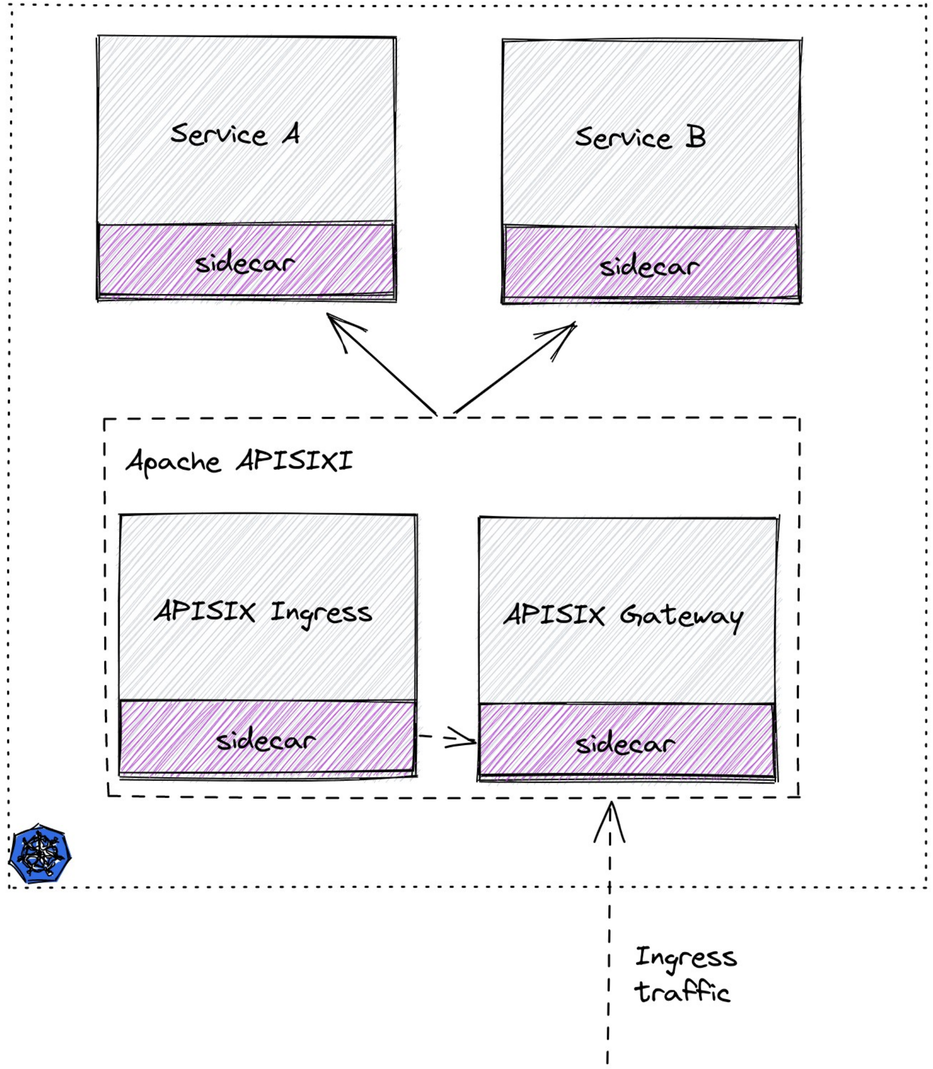
```yaml
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  annotations:
    nginx.ingress.kubernetes.io/rewrite-target: /ip
  name: ingress-rewrite
spec:
  rules:
  - host: httpbin.org
    http:
      paths:
      - backend:
          serviceName: httpbin
          servicePort: 80
        path: /sample
```

```yaml
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  annotations:
    kubernetes.io/ingress.class: apisix
    k8s.apisix.apache.org/rewrite-target: "/ip"
  name: ingress-v1
spec:
  rules:
  - host: httpbin.org
    http:
      paths:
      - path: /sample
        pathType: Exact
        backend:
          service:
            name: httpbin
            port:
              number: 80
```

四、Apache APISIX Ingress 生态及后续规划

作为 Istio Ingress Gateway

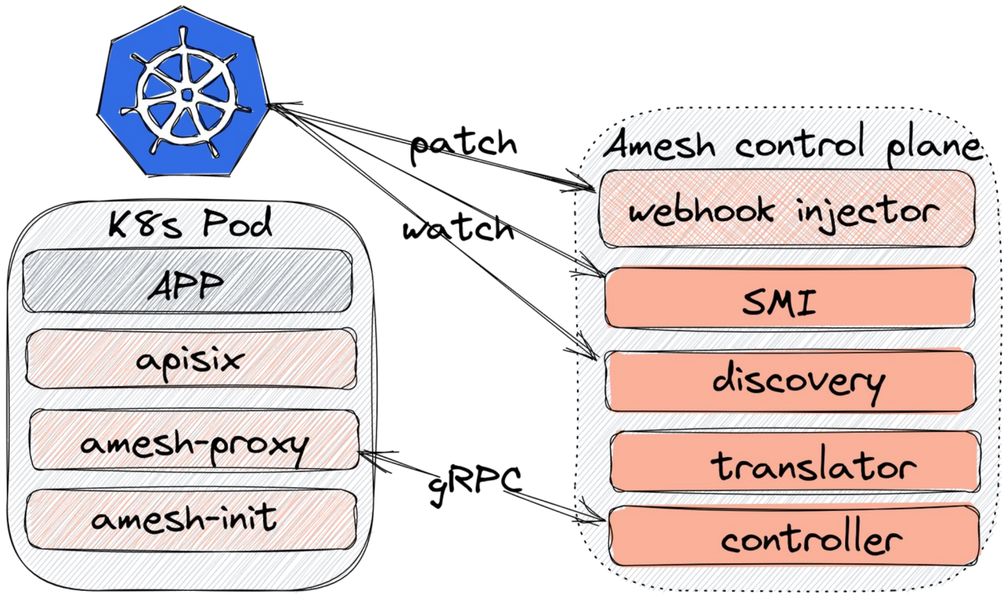




文章已经发布至 Apache APISIX Blog: https://sourl.cn/z2T5MA

基于 Apache APISIX 的服务网格方案 - Amesh

- 自研控制面
- 去除 APISIX 对 etcd 的依赖
- 使用 SMI 开放协议
- 即将开源
  https://github.com/amesh-io



后续规划

- Kubernetes v1.22+ CRD v1 支持（已实现）
- cert-manager 集成（已实现）
- Gateway API 支持（已合并初步支持，持续迭代）
- 扩展新架构
- 丰富生态，与 ArgoCD 等 GitOps 工具结合
- 扩展社区

谢谢

zhangjintao@apache.org