Istio: The Packet’s-Eye View

Matt Turner
@mt165
mt165.co.uk

Tetrate

Kubecon Seattle
December 2018

Objectives

Learn how a packet traverses an Istio/Envoy/Kubernetes system

See how the control plane is involved in that process

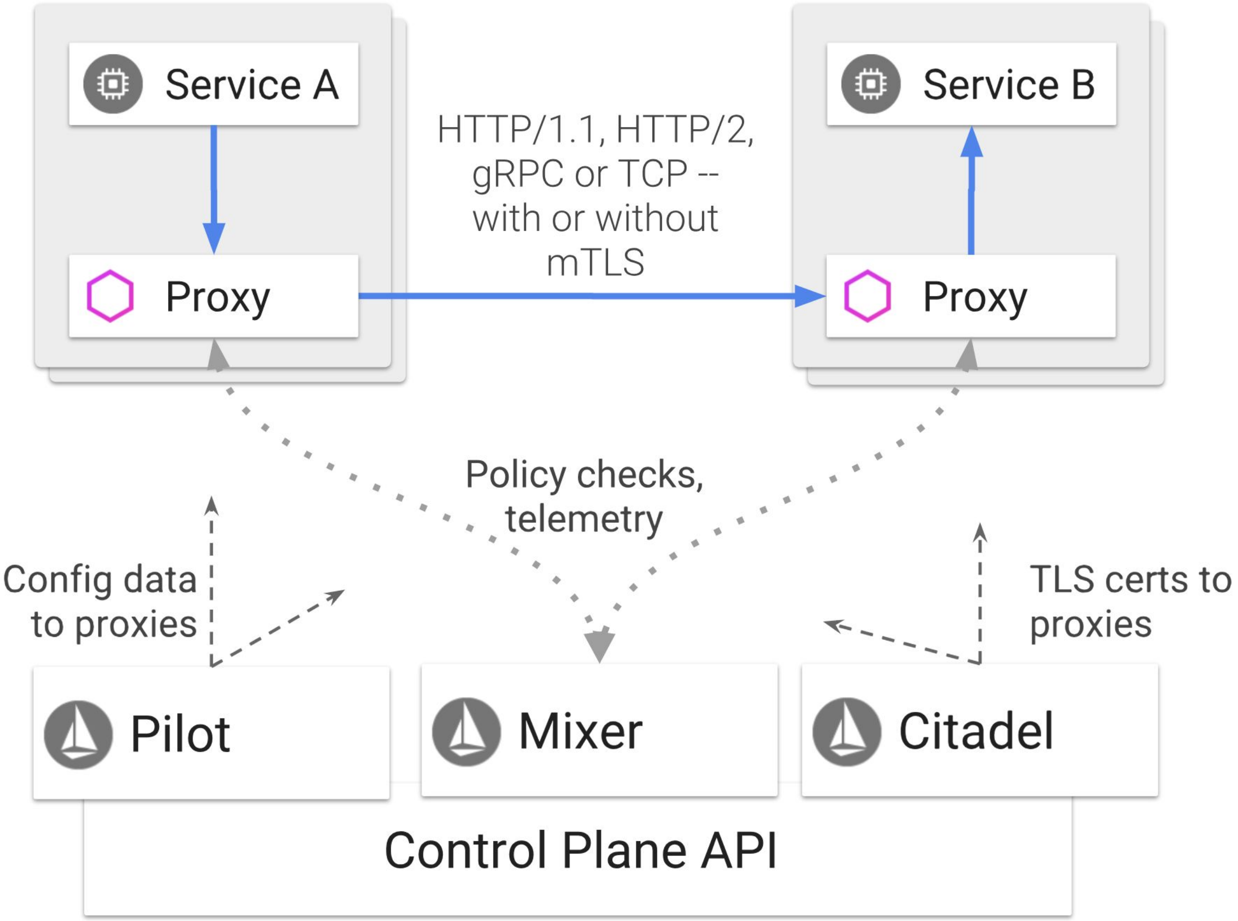
Build a useful mental model for reasoning about, and debugging Istio

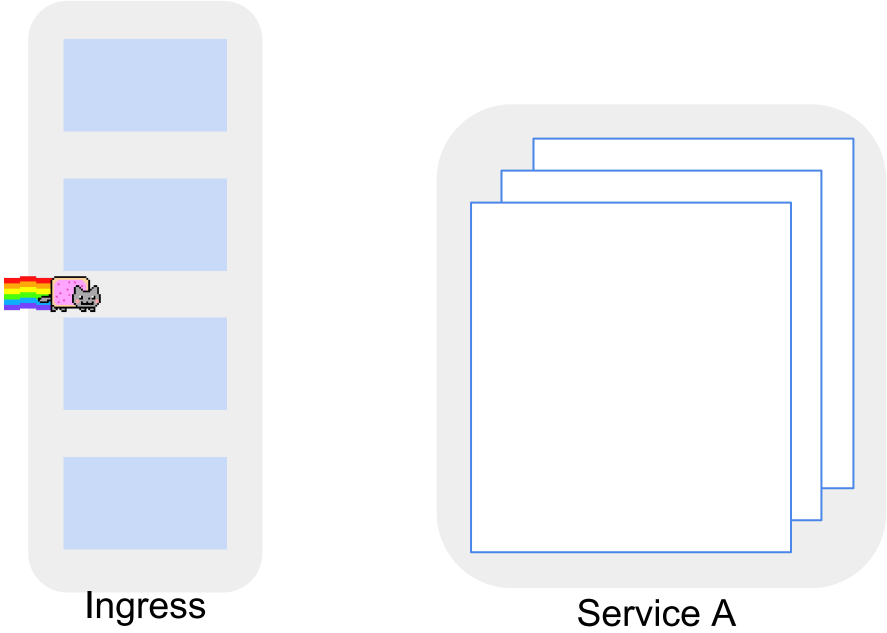
Prerequisites

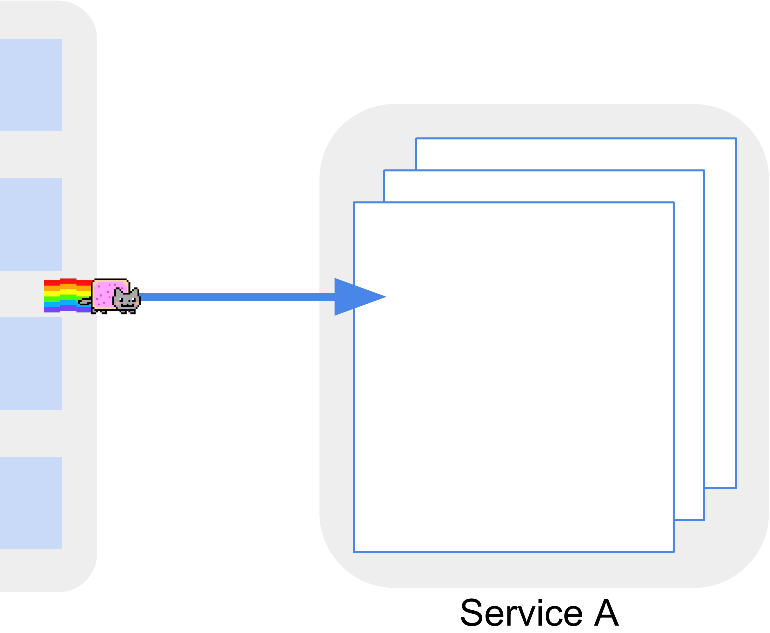
Basic networking knowledge

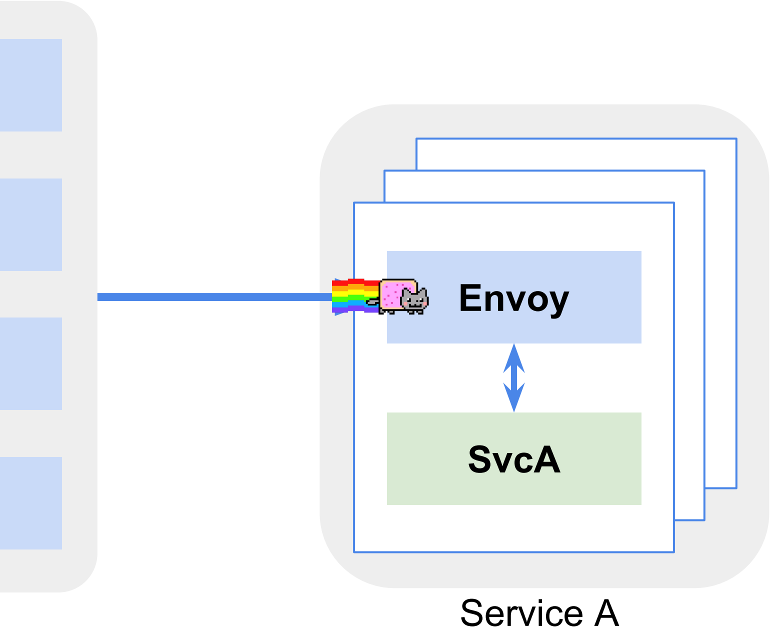
Intermediate Kubernetes knowledge

An understanding of what Istio is and does

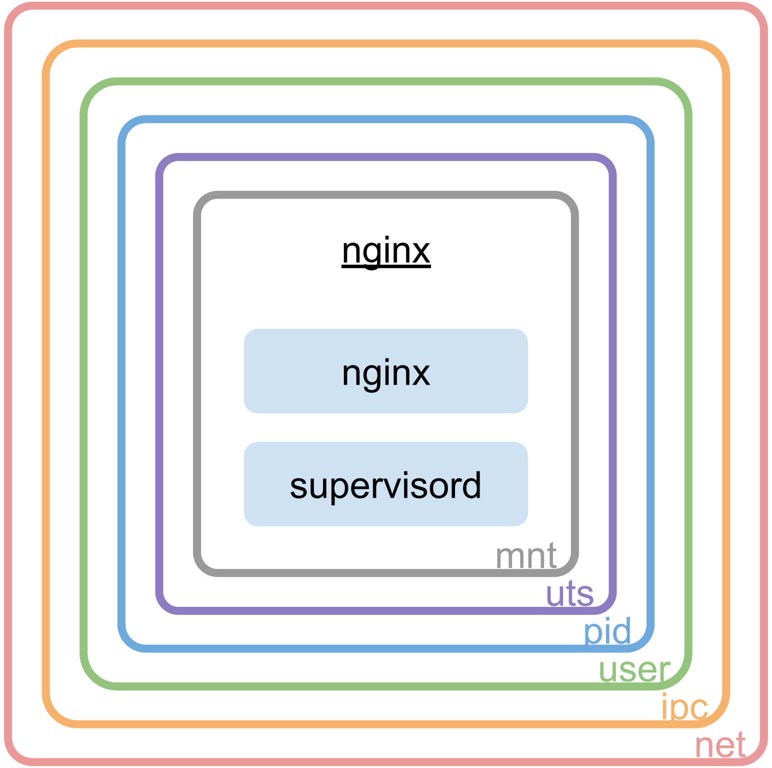

Service A
Proxy
HTTP/1.1, HTTP/2, gRPC or TCP -- with or without mTLS
Service B
Proxy
Policy checks, telemetry
Config data to proxies
TLS certs to proxies
Pilot
Mixer
Citadel
Control Plane API


Ingress
Service A

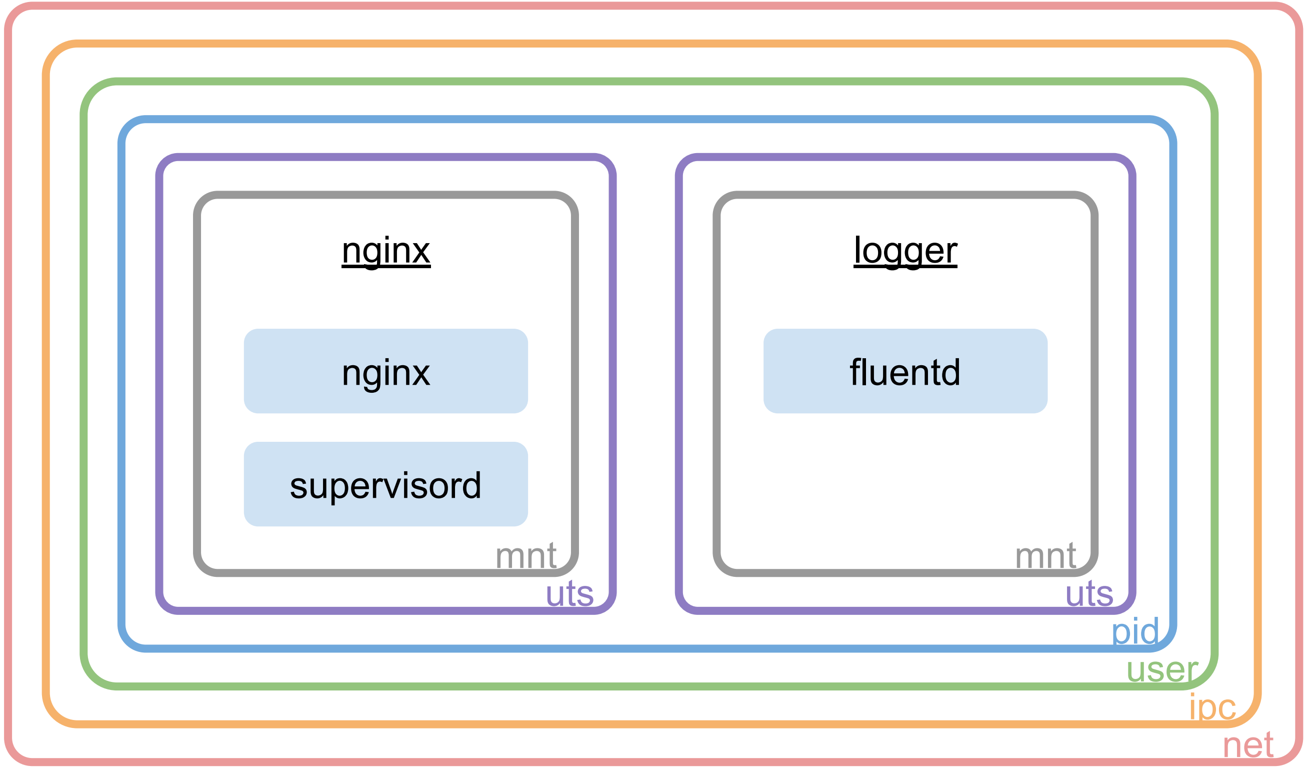

Service A


Envoy
SvcA
Service A

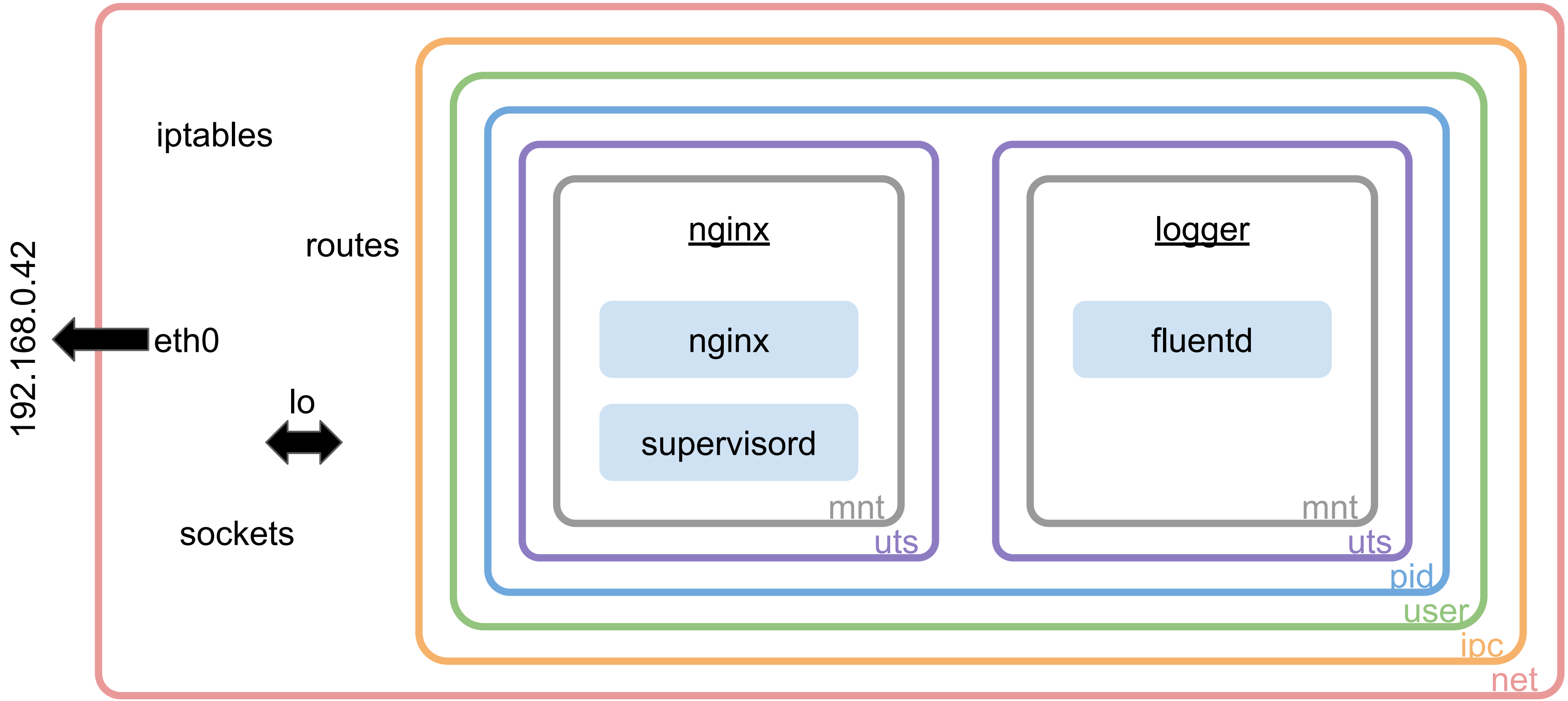
“Containers”



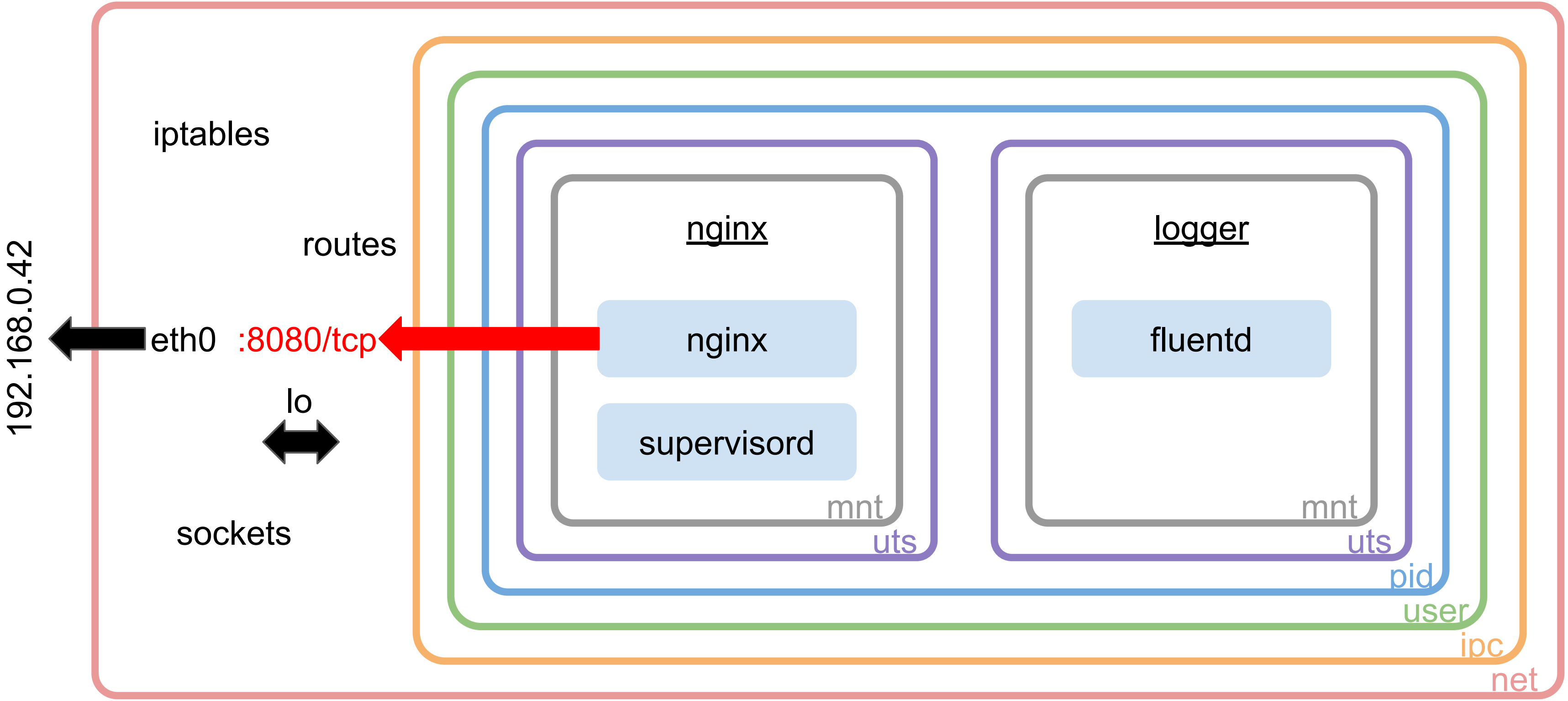
Kubernetes Pods



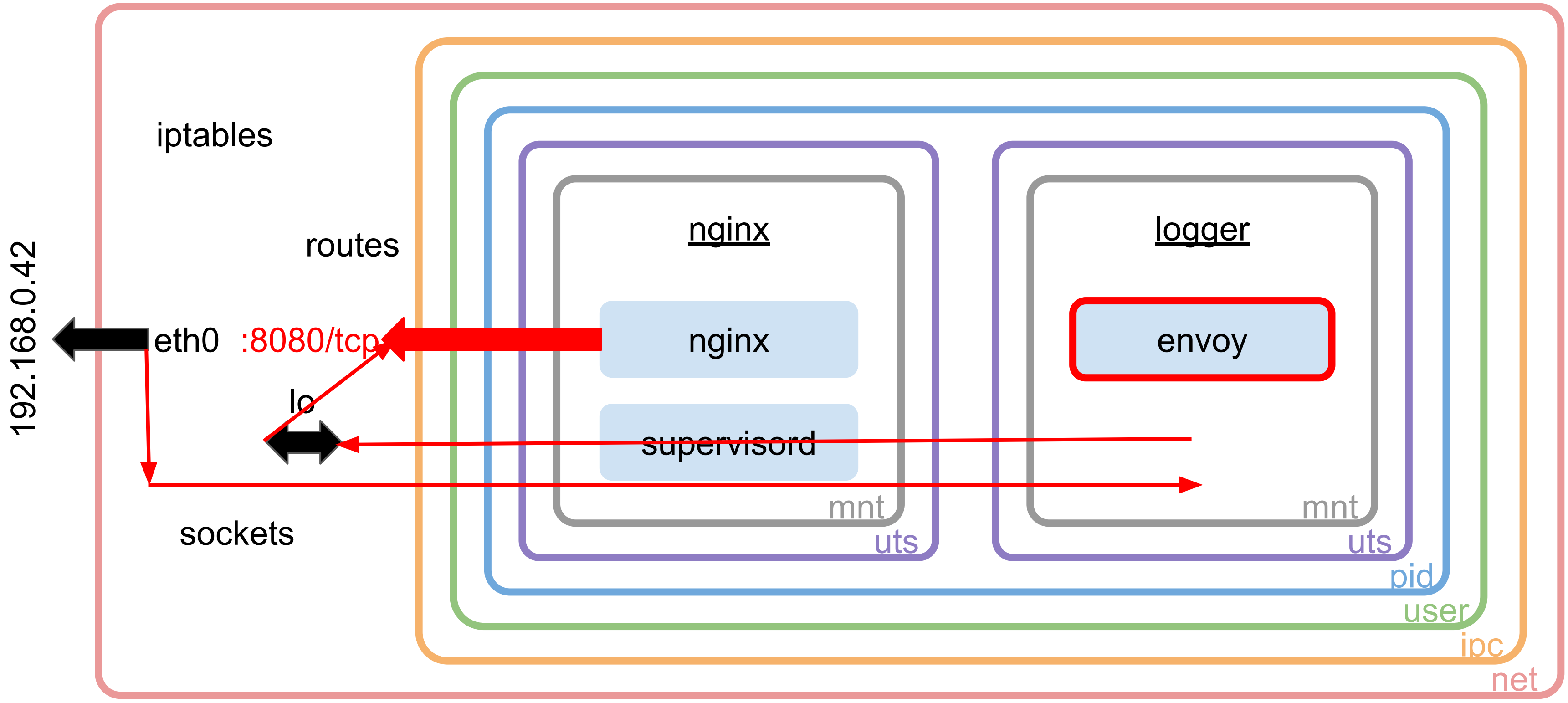
Kubernetes Pods



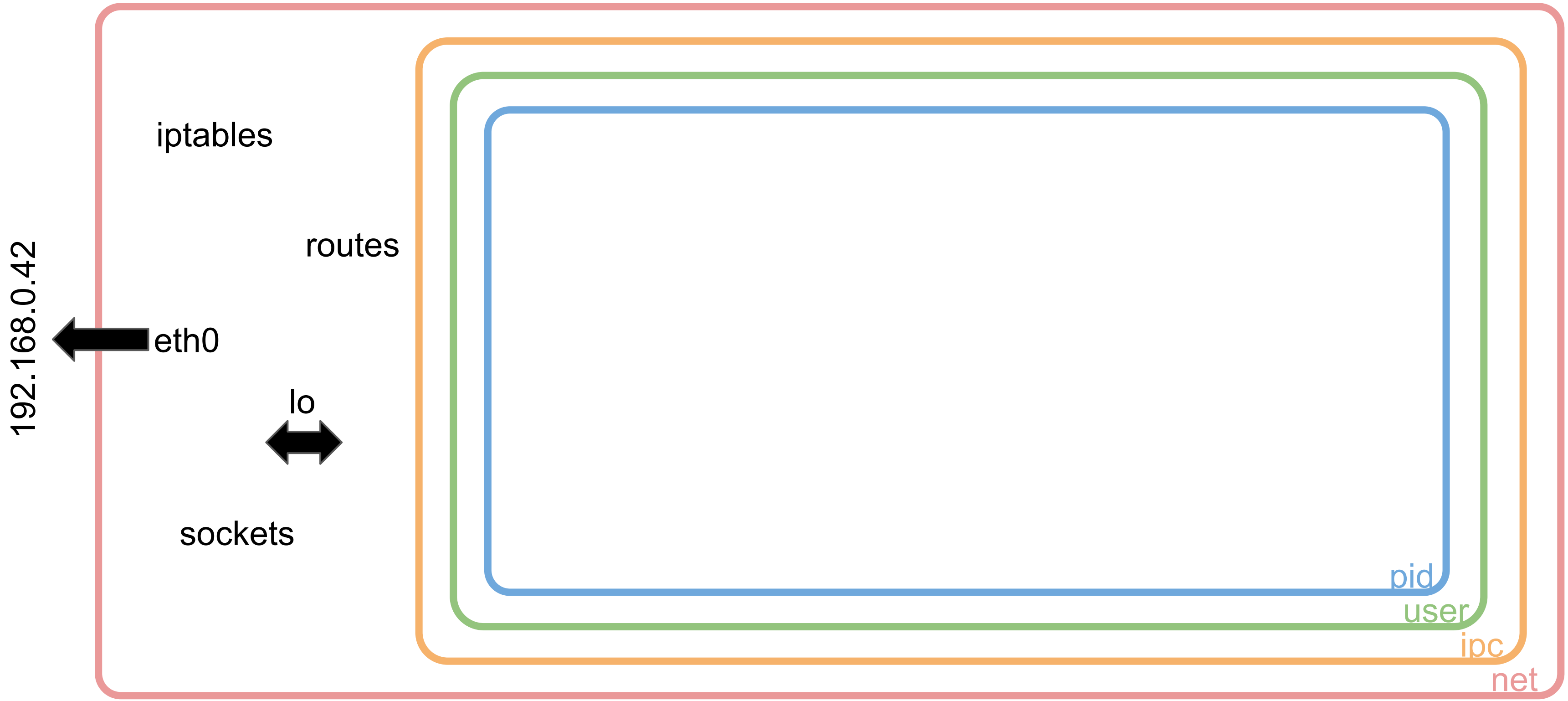
Kubernetes Pods



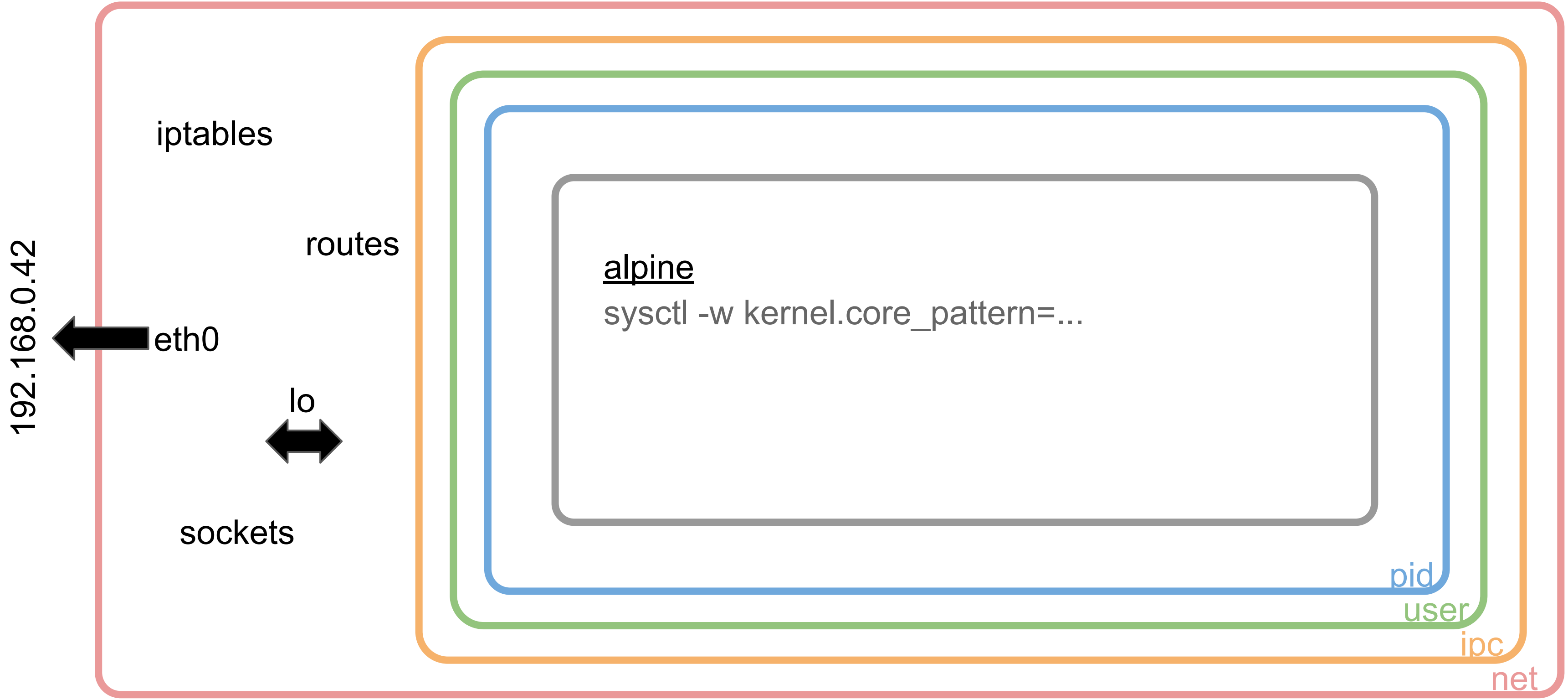
Kubernetes Pods



Sidecar Injection

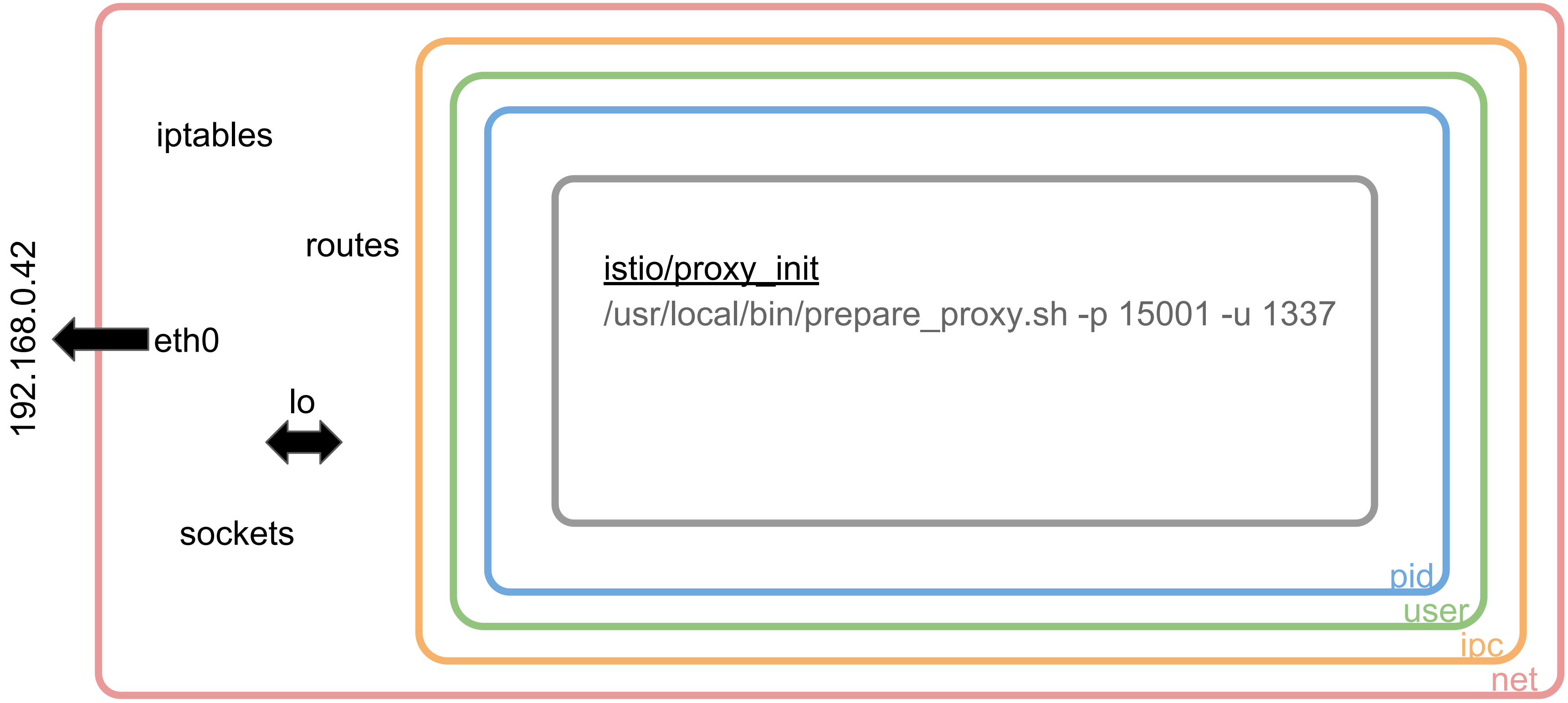


Sidecar Injection

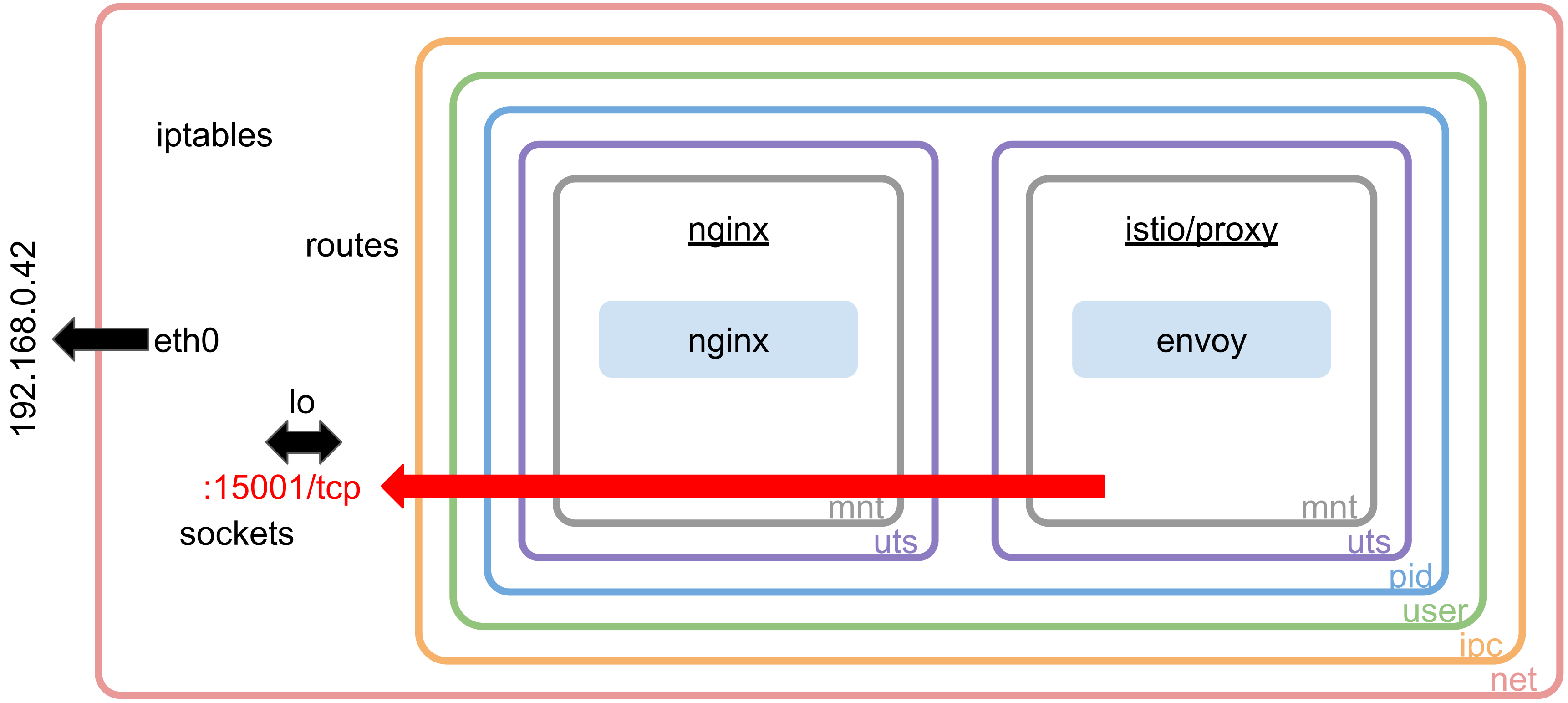


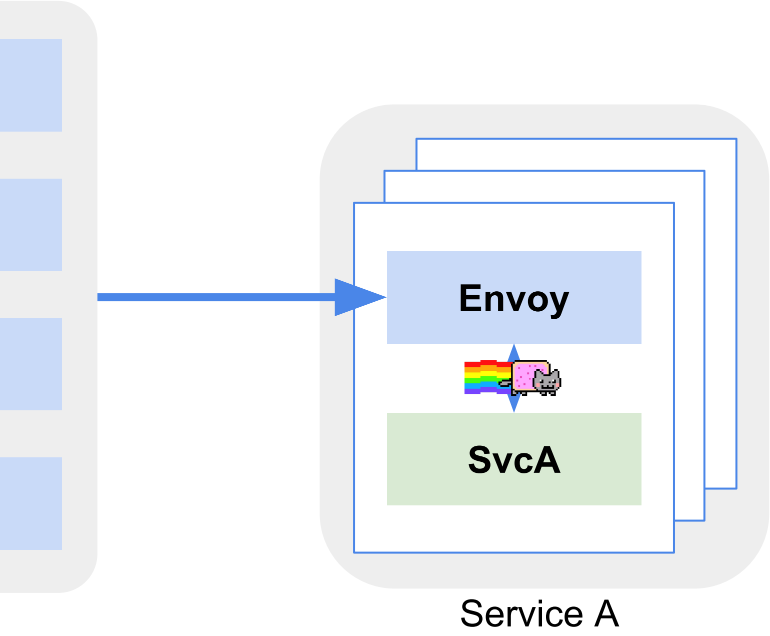
sysctl -w kernel.core_pattern=...

Sidecar Injection

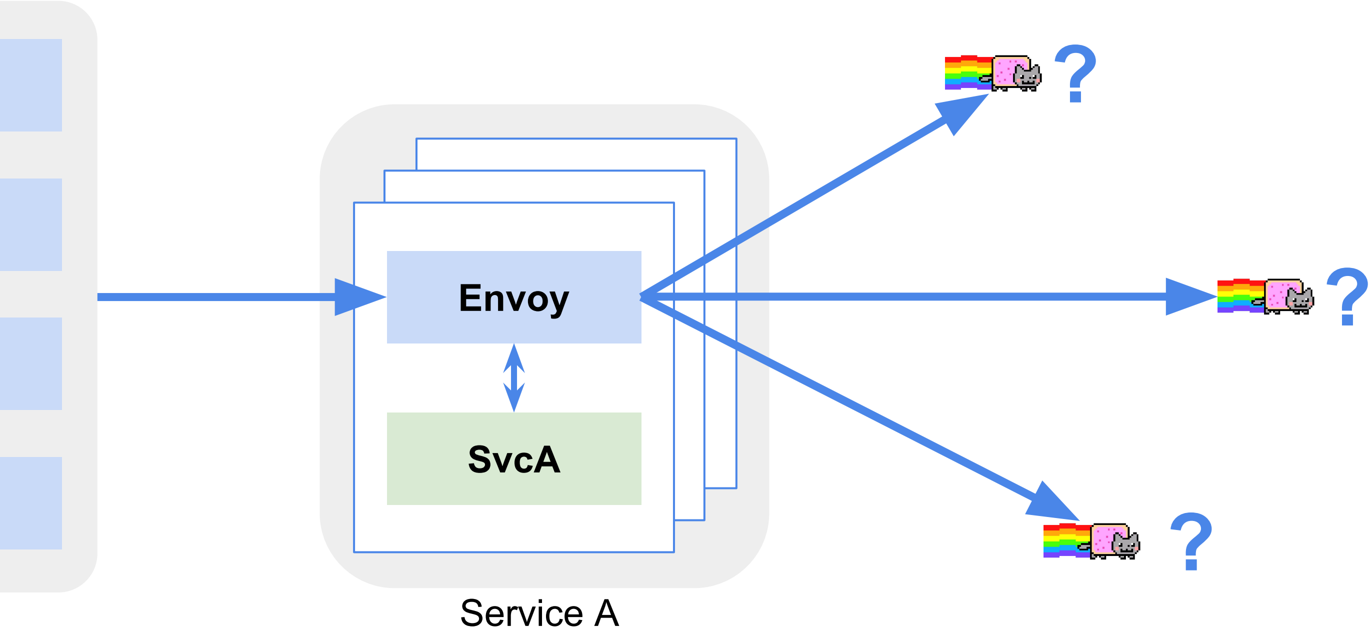


Sidecar Injection




Envoy
SvcA
Service A

Pilot and Routing


Envoy
SvcA
Service A
?
?
?

Services

```
$ kubectl get services  -o wide | grep httpbin

httpbin     NodePort    10.0.0.244   <none>        80:30082/TCP   16m       app=httpbin
```

Service DNS exposure

```
# dig httpbin.default.svc.cluster.local

        httpbin.default.svc.cluster.local. 23 IN A  10.0.0.244
```

Pods

```
$ kubectl get pods -o wide | grep httpbin

httpbin-76ddd74666-2m6ds   1/1     Running   0          16m     172.17.0.13   minikube

httpbin-76ddd74666-ls66n   1/1     Running   0          16m     172.17.0.12   minikube

httpbin-76ddd74666-x5ql2   1/1     Running   0          16m     172.17.0.5    minikube
```

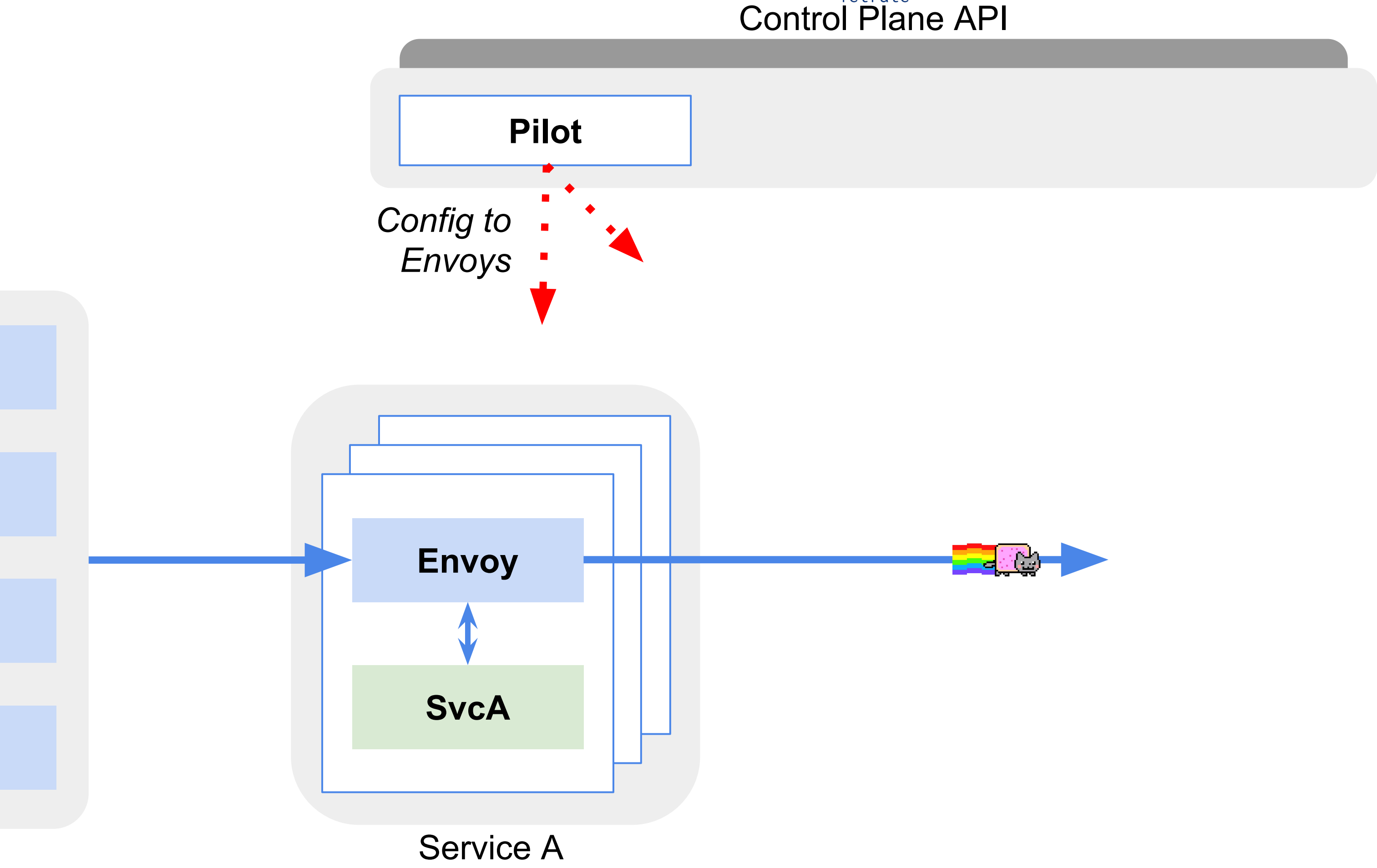
Endpoints

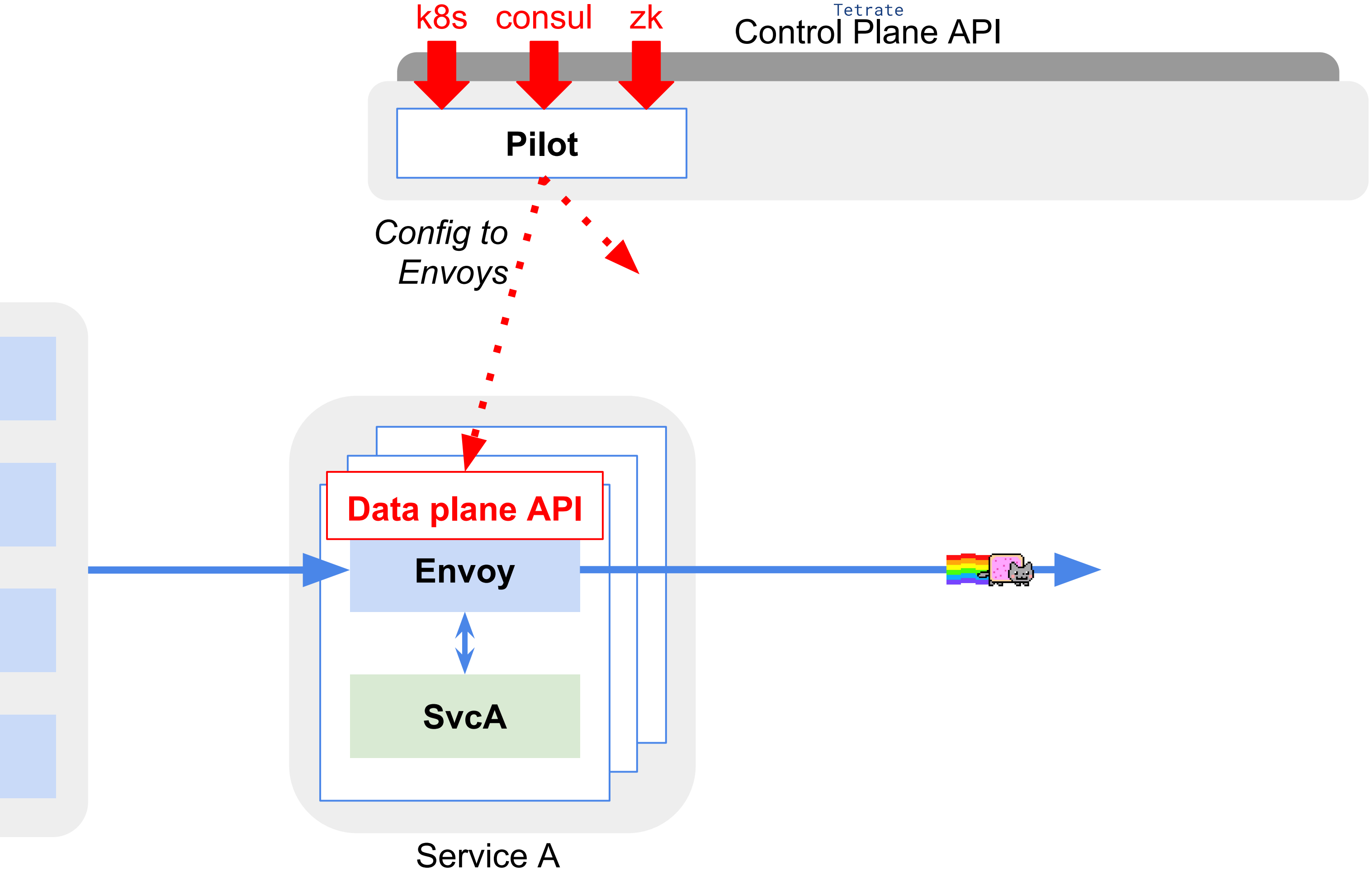
```
$ kubectl get endpoints | grep httpbin

httpbin      172.17.0.12:8000,172.17.0.13:8000,172.17.0.5:8000   21m
```

Endpoints

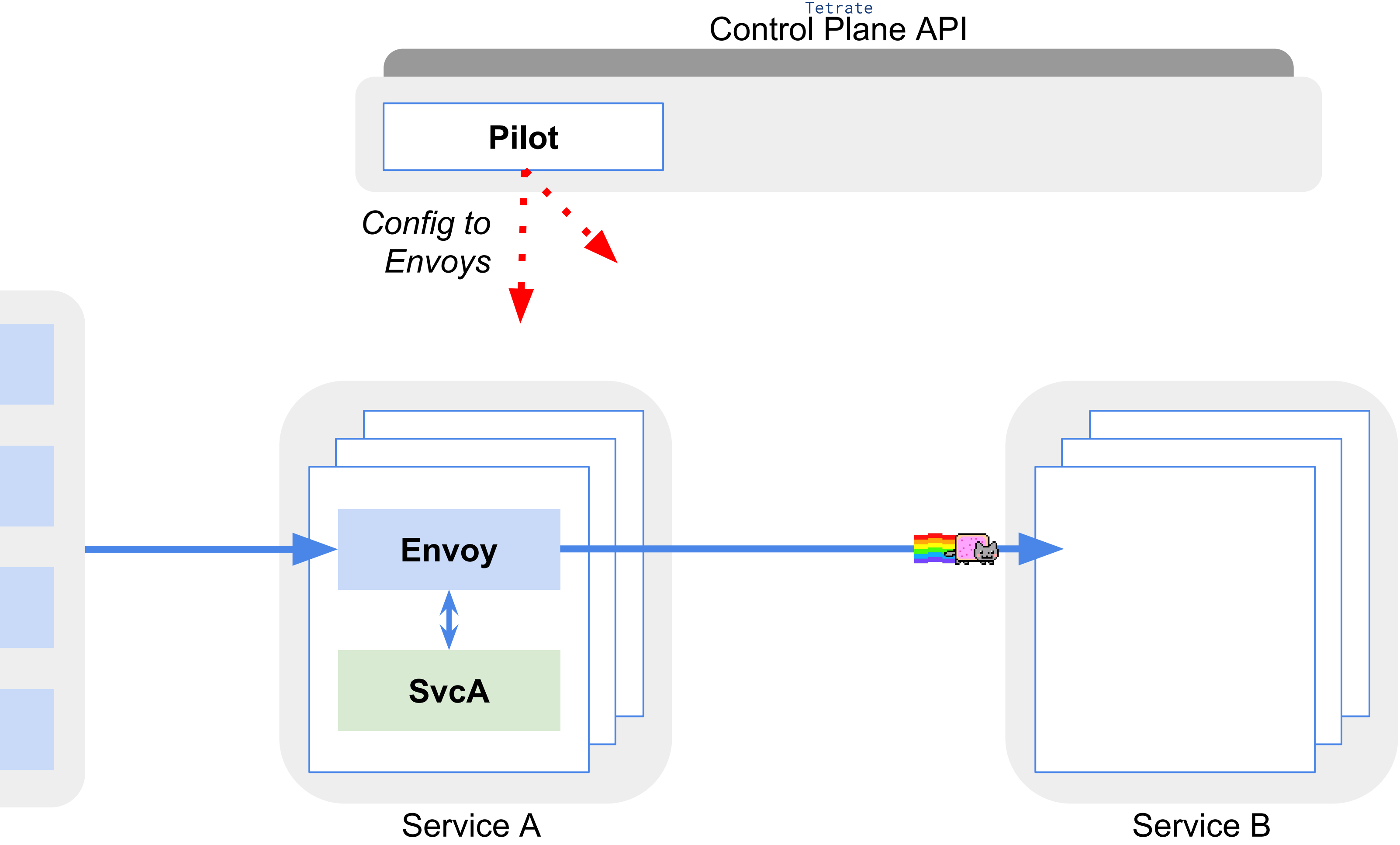
```
$ kubectl get endpoints httpbin -o yaml
apiVersion: v1
kind: Endpoints
…
subsets:
- addresses:
  - ip: 172.17.0.12
    nodeName: minikube
    targetRef:
      kind: Pod
…
  ports:
  - name: http
    port: 8000
    protocol: TCP
```


Control Plane API
Pilot
Config to Envoys
Envoy
SvcA
Service A


k8s
consul
zk
Control Plane API
Pilot
Config to Envoys
Data plane API
Envoy
SvcA
Service A

Pilot

- Ingress Routing
- Traffic Mirroring
- Traffic Shifting
- Canary Deployments
- Circuit Breaking
- Fault Injection



Control Plane API

Pilot

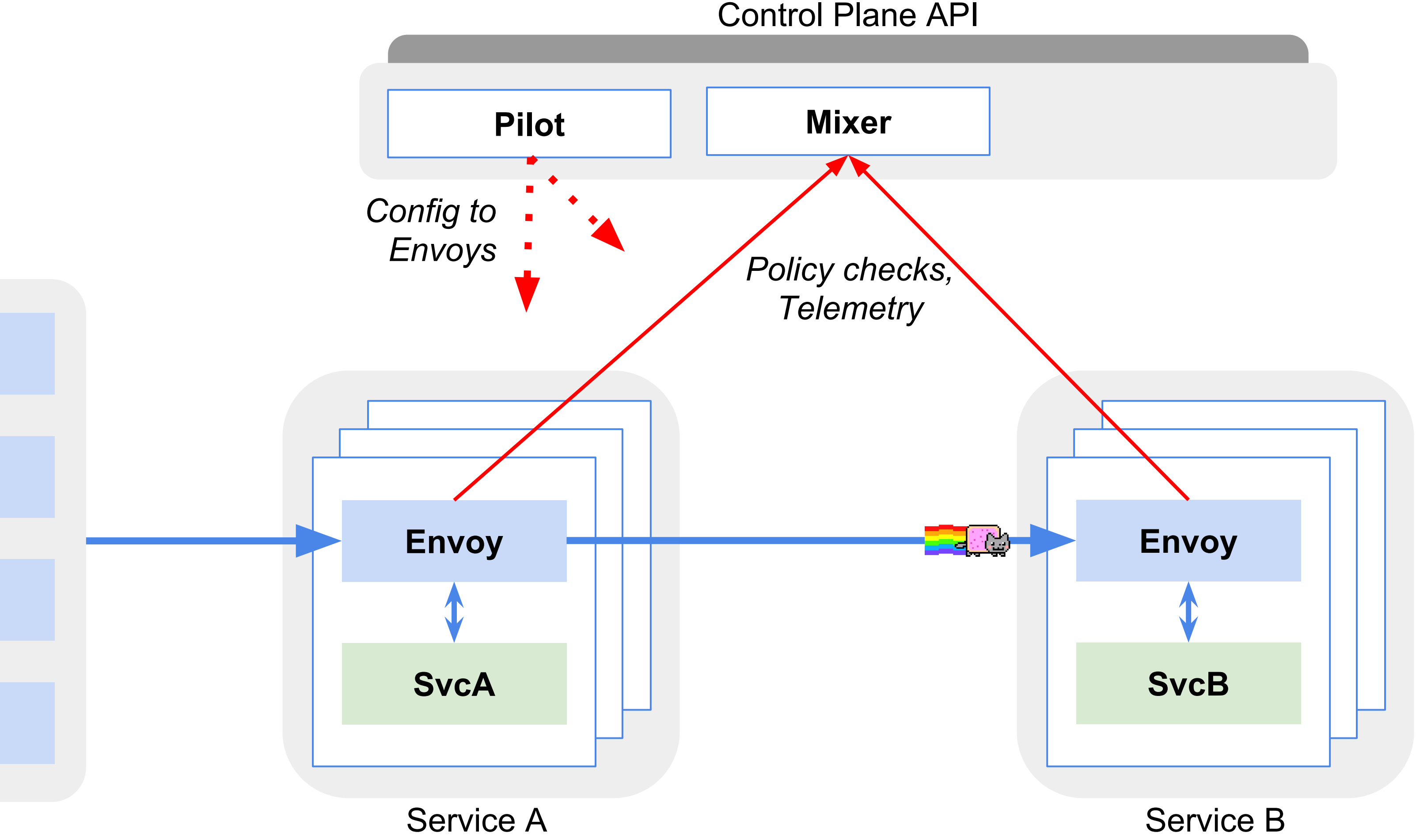
Config to Envoys

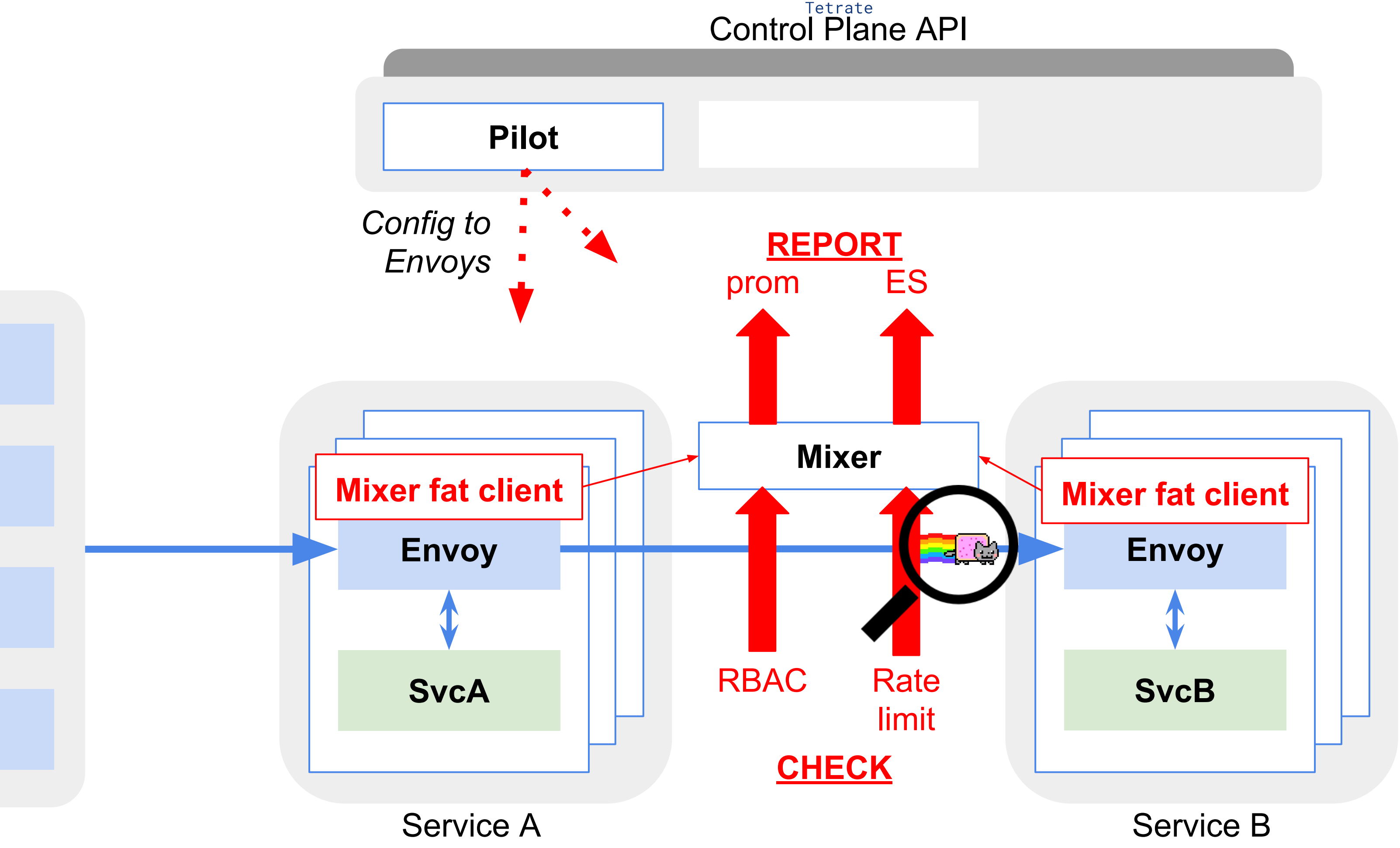
Envoy

SvcA

Service A

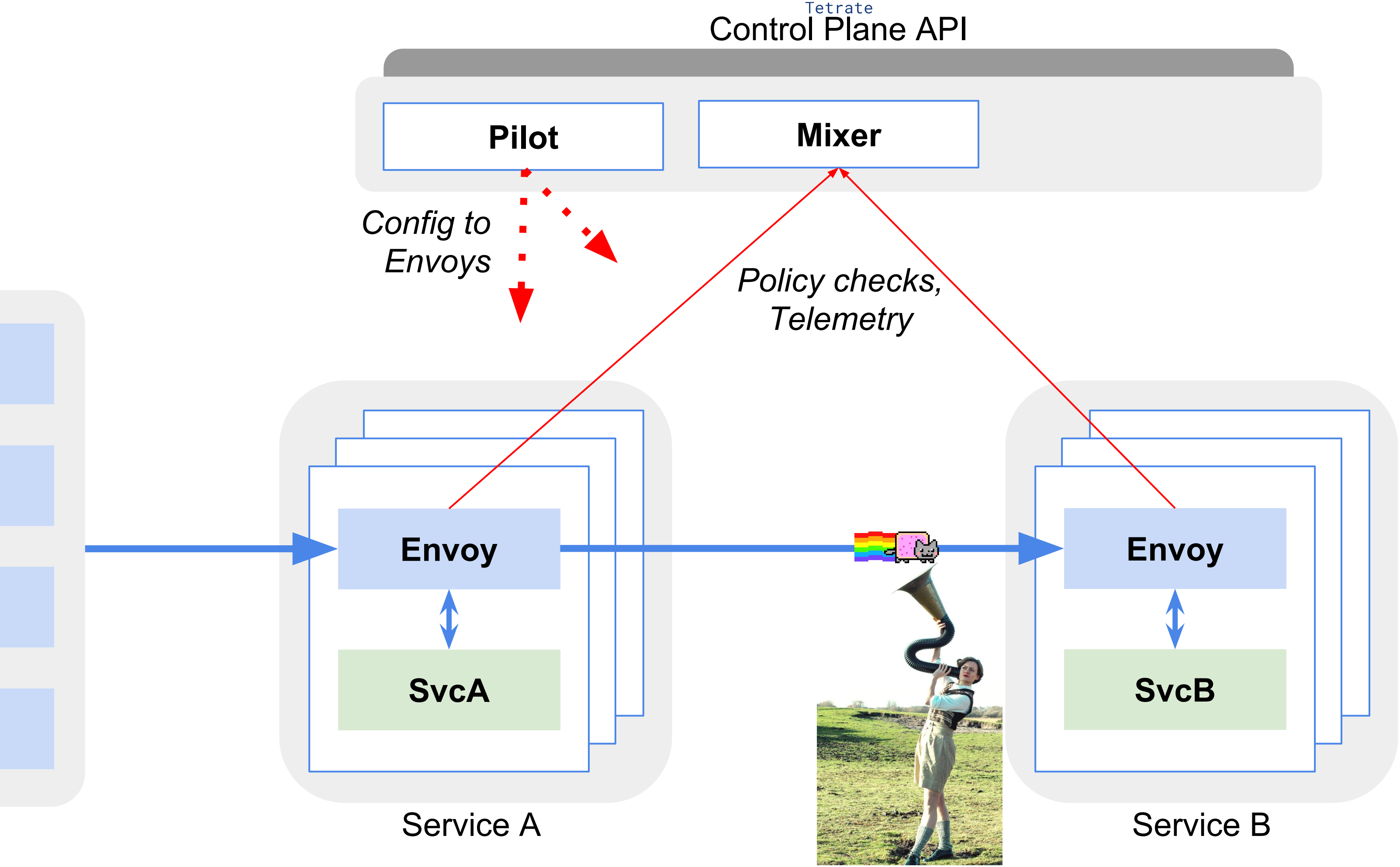
Service B

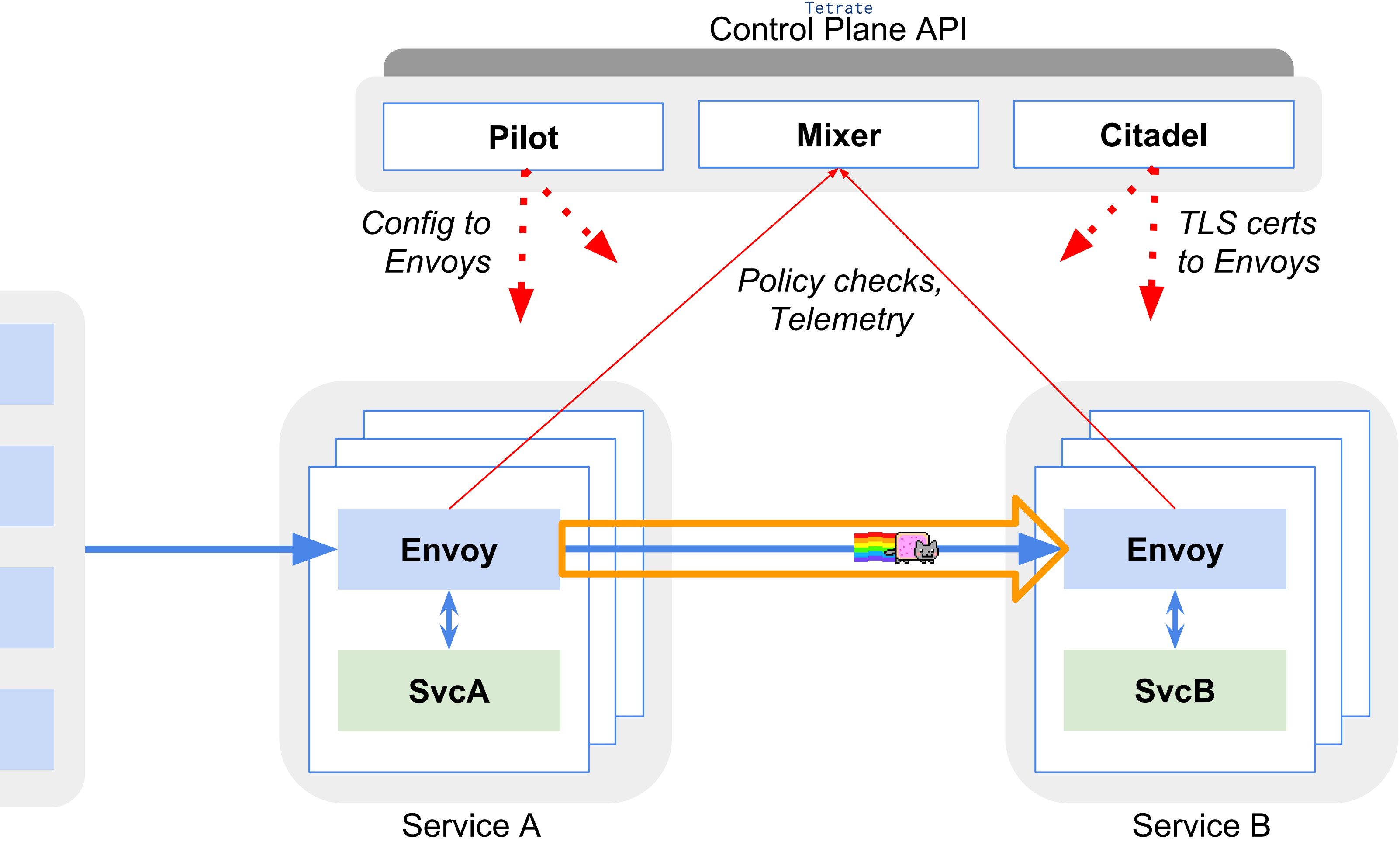

Control Plane API
Pilot
Mixer
Config to Envoys
Policy checks, Telemetry
Envoy
SvcA
Envoy
SvcB
Service A
Service B

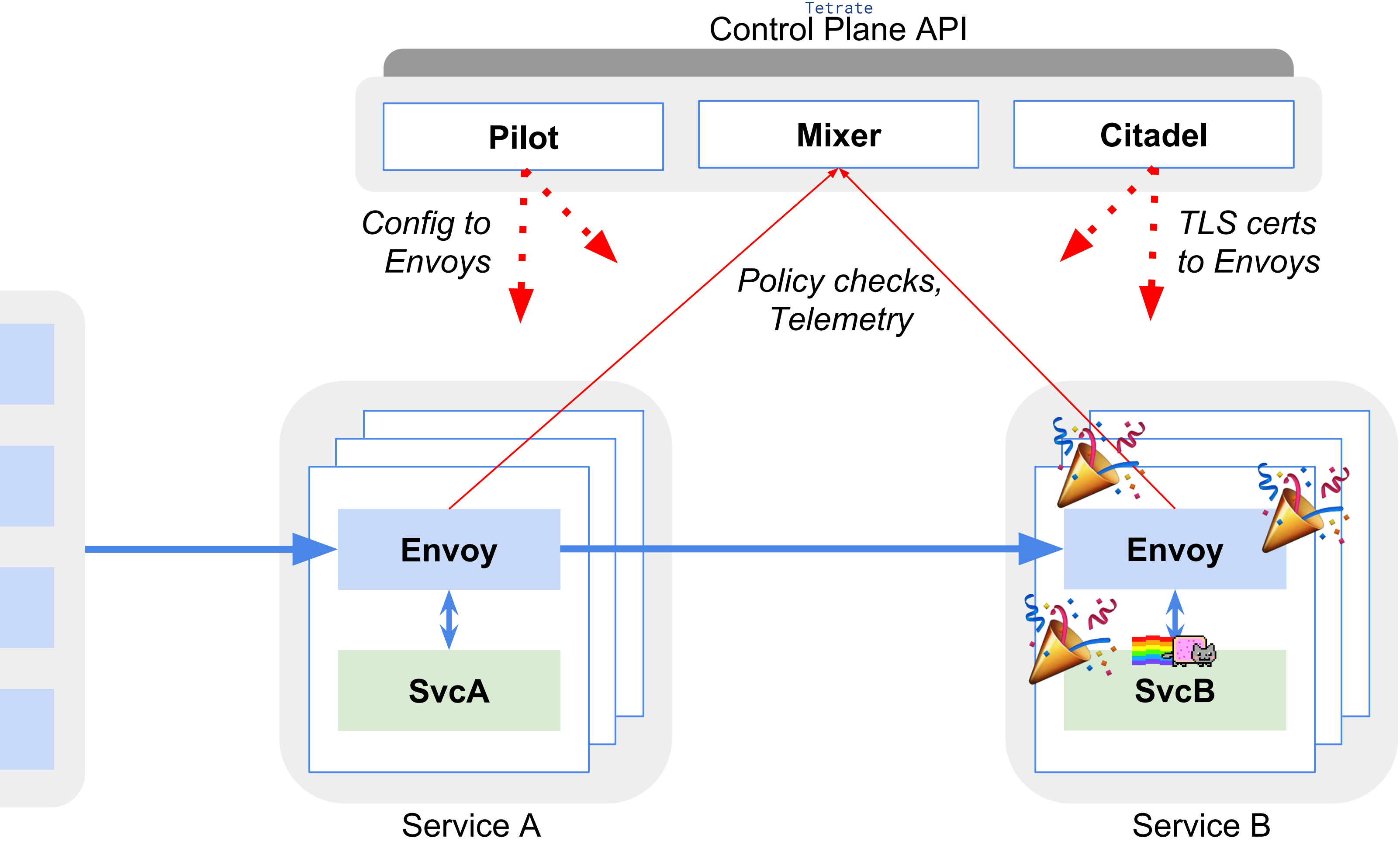

Control Plane API
Pilot
Config to Envoys
REPORT
prom
ES
Mixer
Mixer fat client
Envoy
SvcA
Mixer fat client
Envoy
SvcB
RBAC
Rate limit
CHECK
Service A
Service B

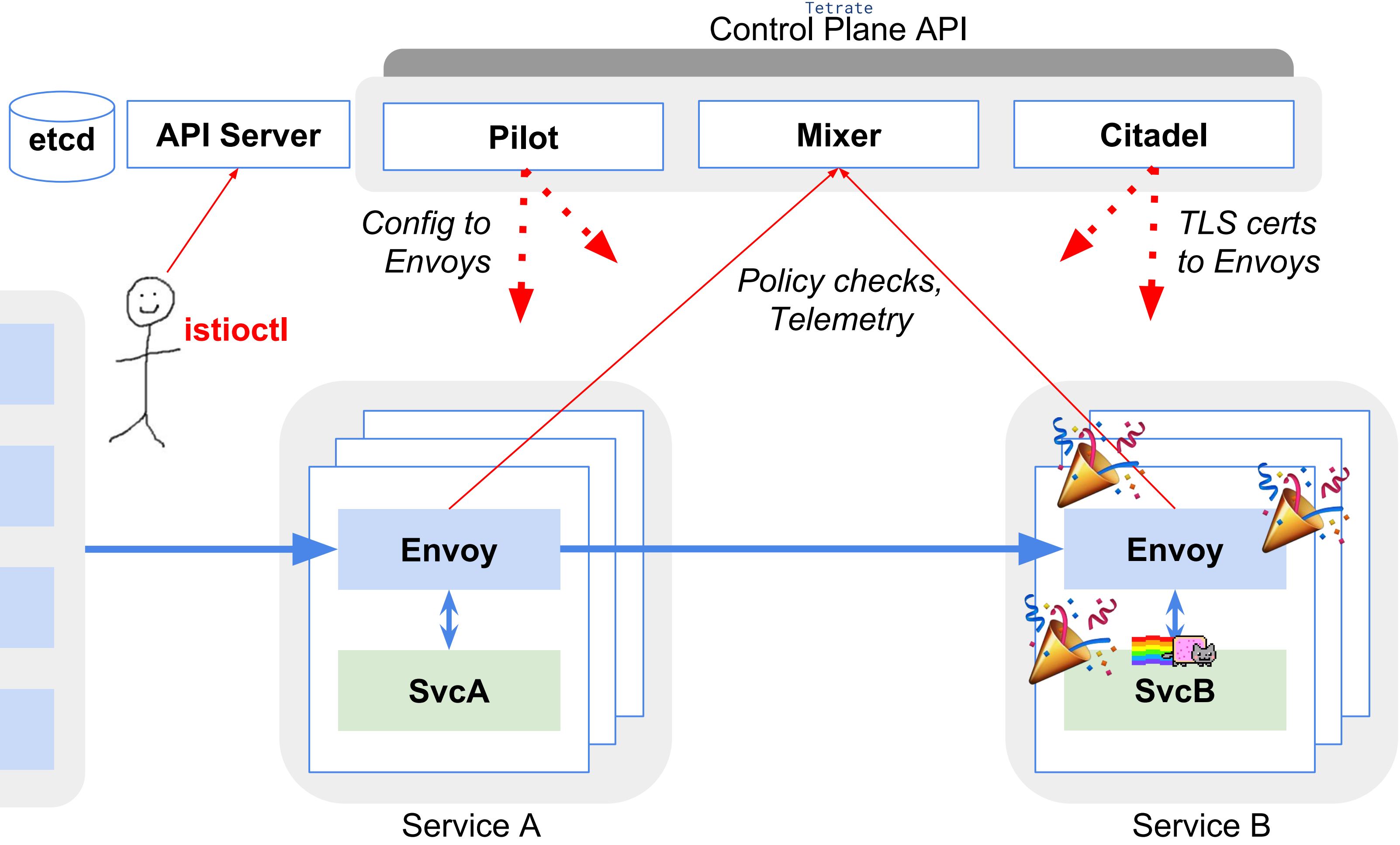
Mixer

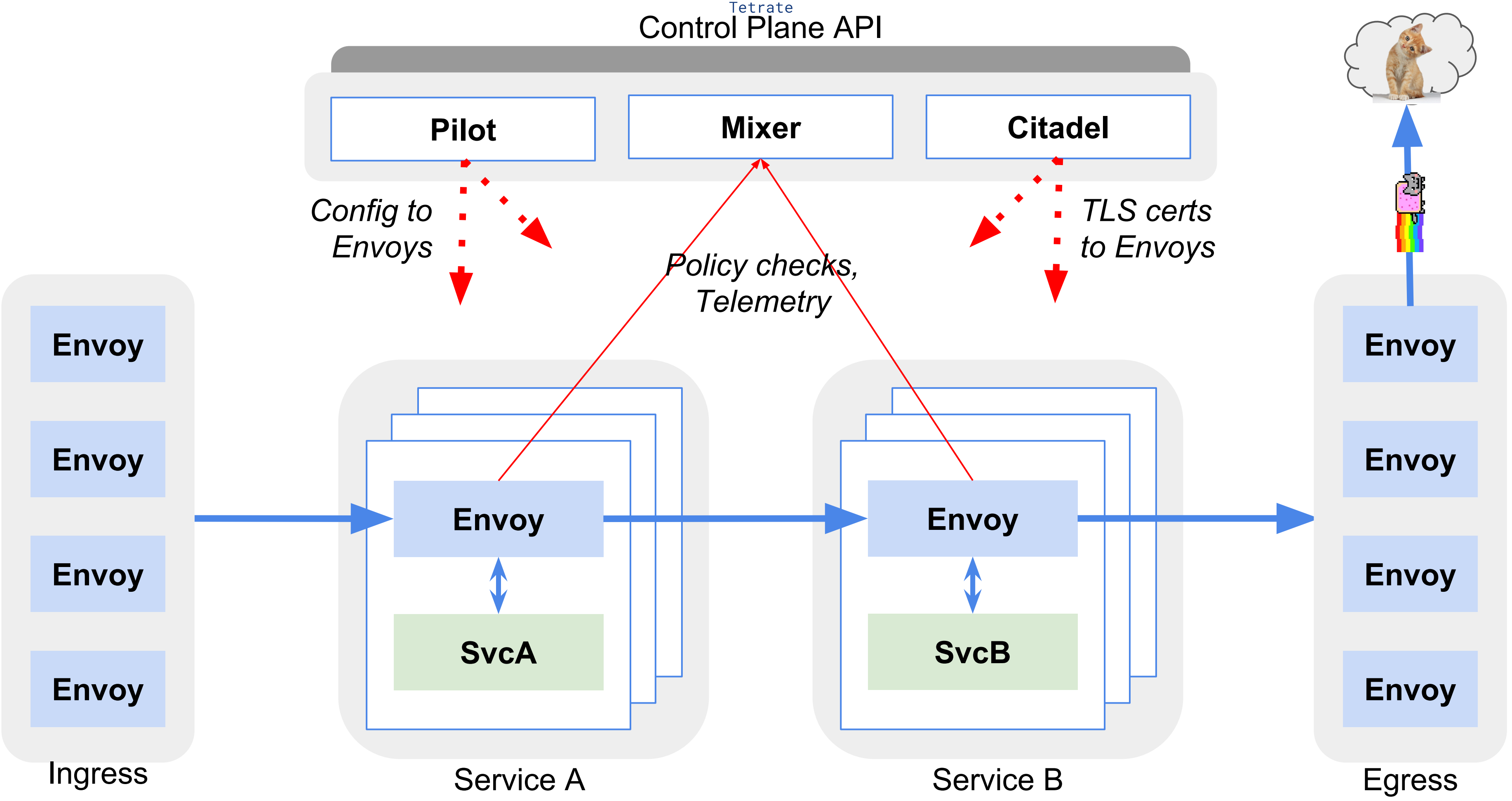
- Check
  - ACLs / Authorization
  - Rate Limiting
- Report
  - Logs
  - Metrics
  - Tracing


Control Plane API
Pilot
Mixer
Config to Envoys
Policy checks, Telemetry
Envoy
SvcA
Service A
Envoy
SvcB
Service B


Control Plane API
Pilot
Mixer
Citadel
Config to Envoys
Policy checks, Telemetry
TLS certs to Envoys
Envoy
SvcA
Service A
Envoy
SvcB
Service B


Control Plane API
Pilot
Mixer
Citadel
Config to Envoys
Policy checks, Telemetry
TLS certs to Envoys
Envoy
SvcA
Envoy
SvcB
Service A
Service B


Control Plane API
etcd
API Server
Pilot
Mixer
Citadel
Config to Envoys
Policy checks, Telemetry
TLS certs to Envoys
istioctl
Envoy
SvcA
Service A
Envoy
SvcB
Service B


Control Plane API
Pilot
Mixer
Citadel
Config to Envoys
Policy checks, Telemetry
TLS certs to Envoys
Envoy
Envoy
Envoy
Envoy
Ingress
Envoy
SvcA
Service A
Envoy
SvcB
Service B
Envoy
Envoy
Envoy
Envoy
Egress

Recap

We learned:

- How a packet traverses an Istio/Envoy/Kubernetes system
- What control plane calls are made in that process
- A useful mental model for reasoning about, and debugging Istio

Thanks!

@mt165

Tetrate