Security in Applications

API Management and Service Mesh (ISTIO)

Alfred Bach

abach@redhat.com
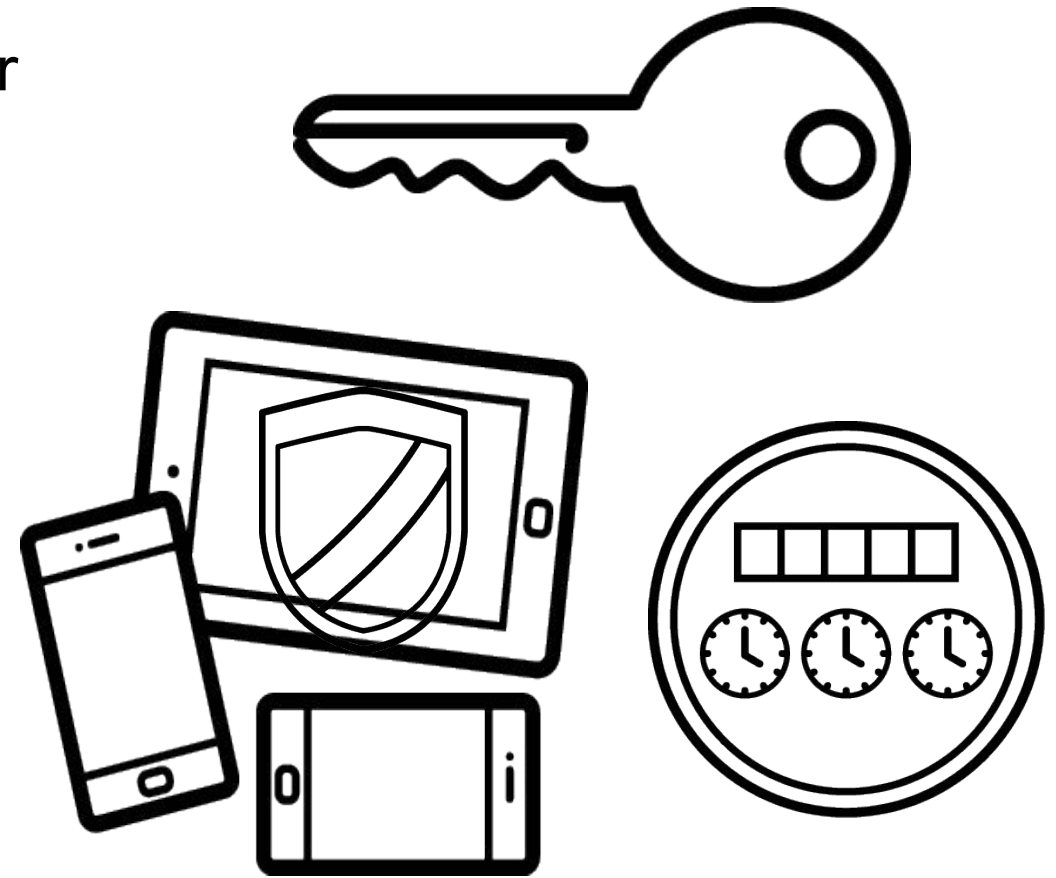
Red Hat

API Management (3 Scale)

V0000000
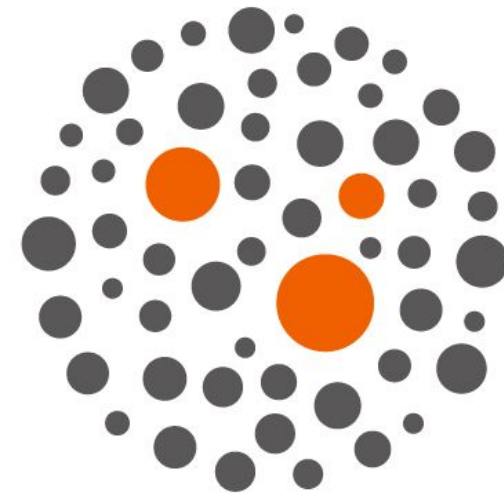
APPLICATION API MANAGEMENT

Consider configuring an API gateway for container platform & application APIs

- ▶ Authentication and authorization

- ▶ LDAP integration

- ▶ End-point access controls

- ▶ Rate limiting

Red Hat

RED HAT 3SCALE API MANAGEMENT

Key capabilities

Control

- Security
- Key management
- Rate limiting
- Policy enforcement
- App and user management
- Provisioning

Visibility

- Analytics
- App tracking
- User tracking
- Traffic alerts
- Engagement
- Developer support

Flexibility

- Distributed
- Multi-department
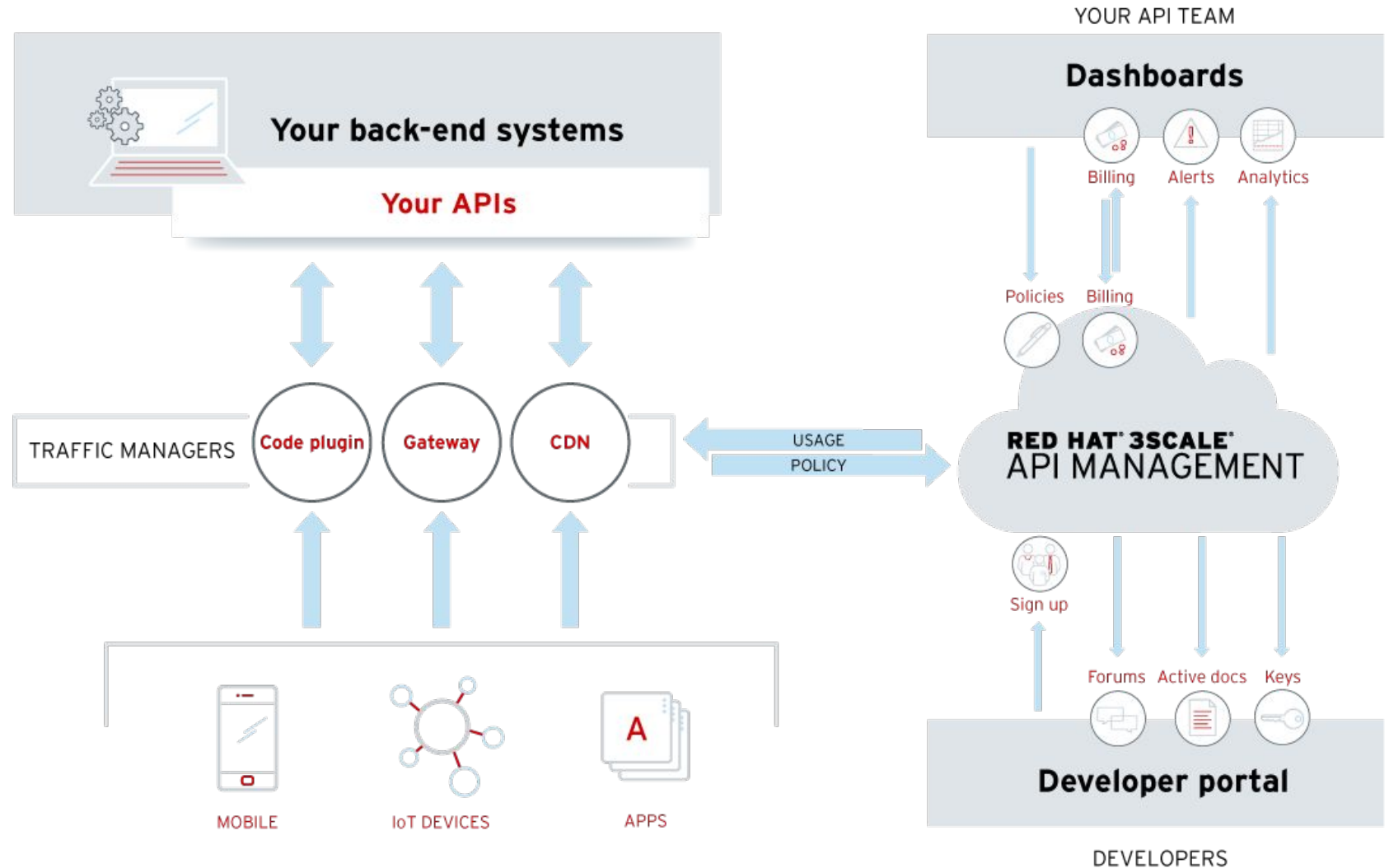- Multi-environment
- Highly scalable
- Powerful APIs
- Webhooks

Flexible distributed control

Modular
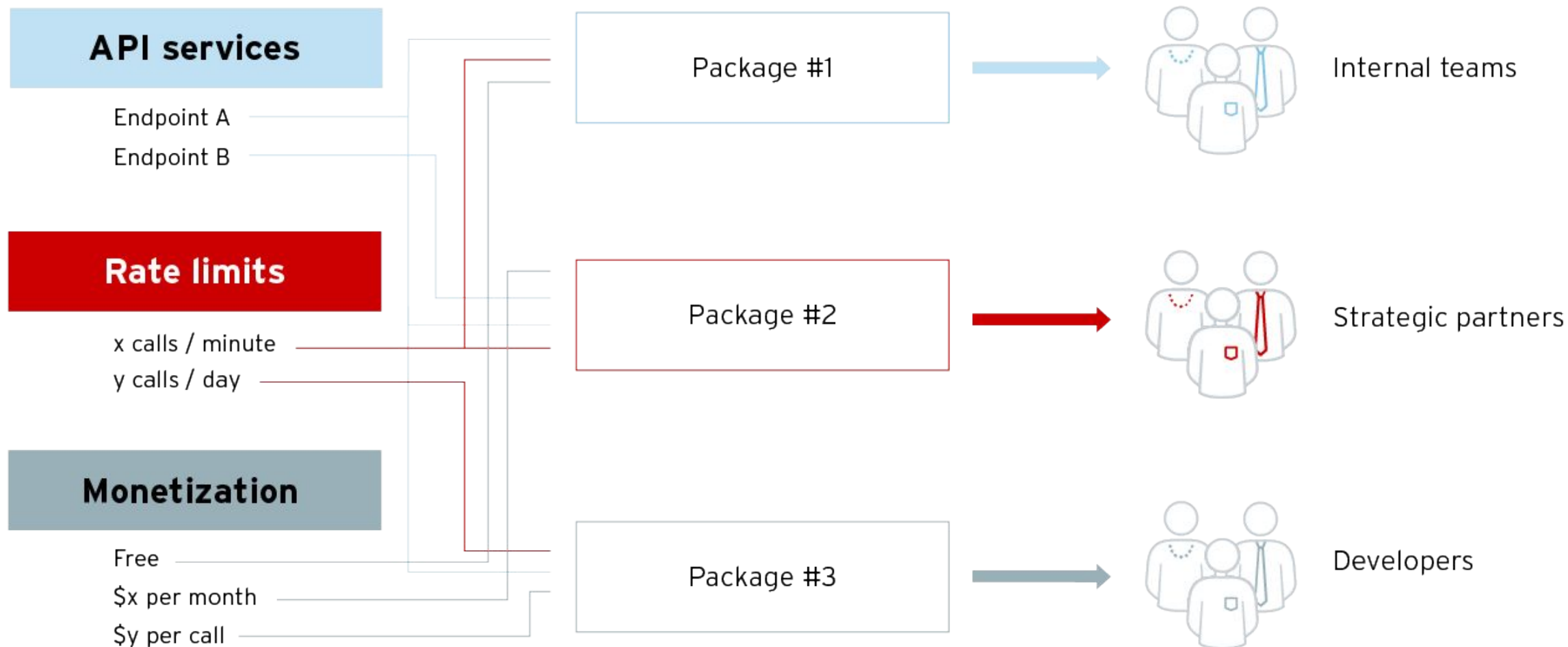
No single point of failure

Hybrid cloud access

Highly scalable

API contracts and rate limits
Package your APIs. Create access tiers. Set rate limits.

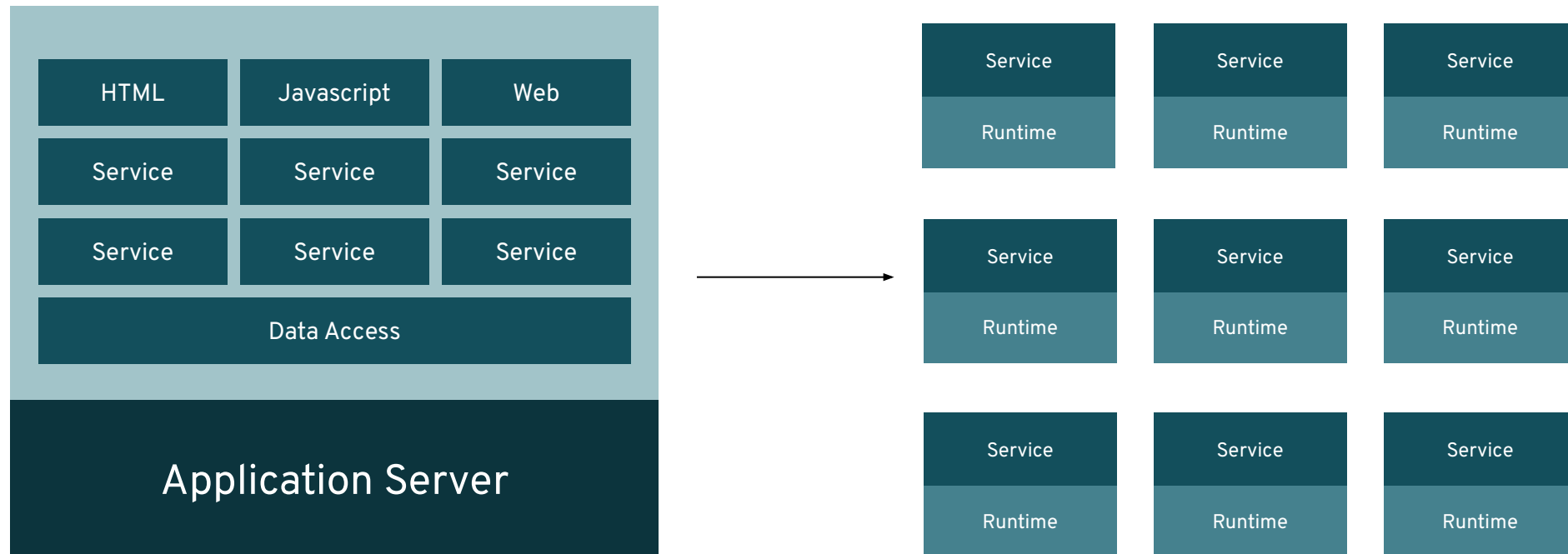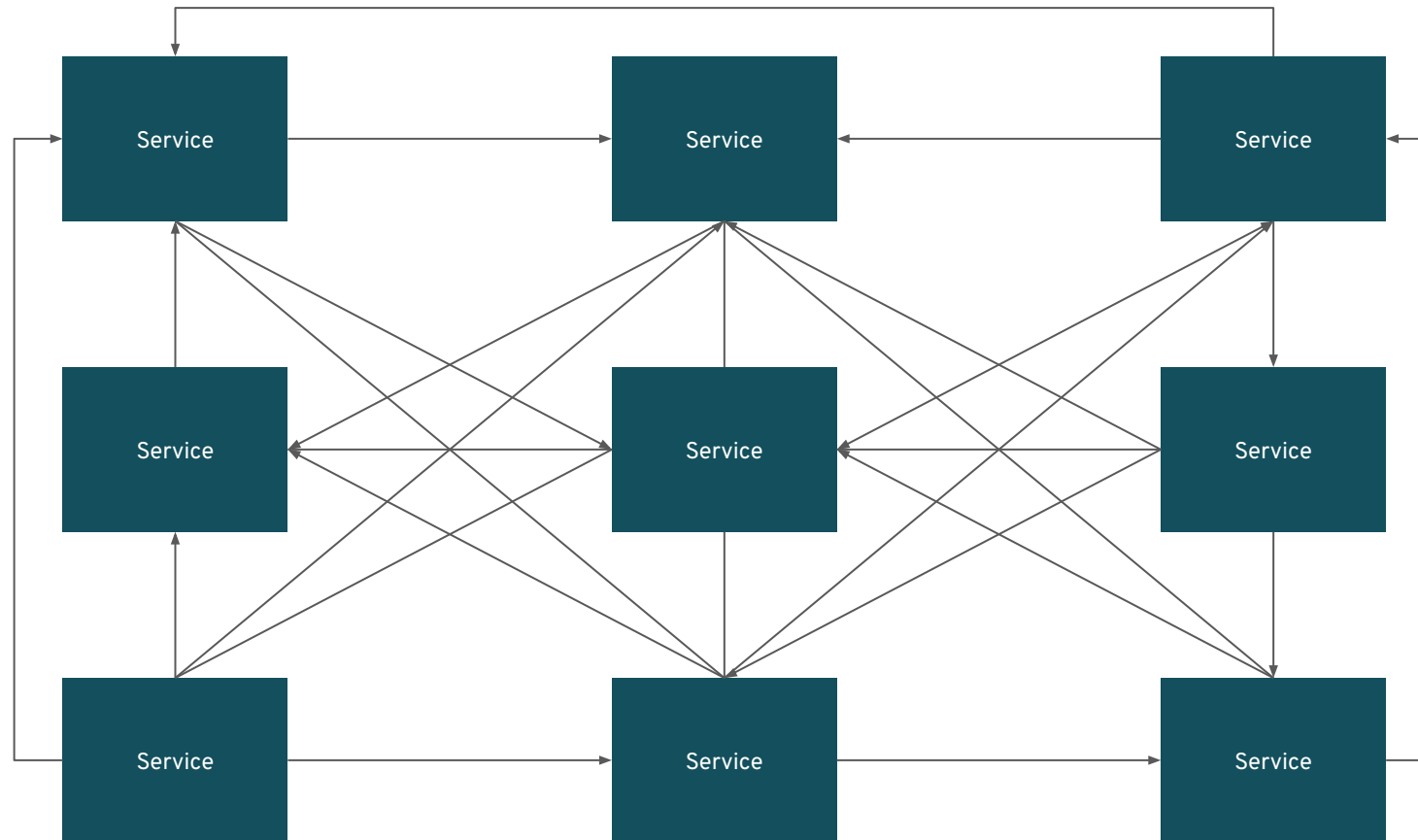Allow/restrict access to your API endpoints along with rate limits.

API services

Endpoint A

Endpoint B

Rate limits

x calls / minute

y calls / day

Monetization

Free

$x per month

$y per call

Package #1 → Internal teams

Package #2 → Strategic partners

Package #3 → Developers

V0000000

Red Hat Service Mesh (Istio)

~~MICROSERVICES~~ ARCHITECTURE

DISTRIBUTED

DISTRIBUTED ARCHITECTURE

A better way with a service mesh

Service
Tracing
Circuit Breaker
Routing
Svc Discovery
Config
Platform

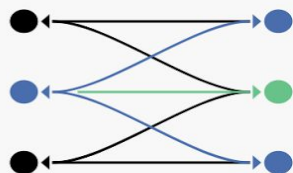| Service |
|---|
| Container Platform (+ Service Mesh) |

...2014   2018
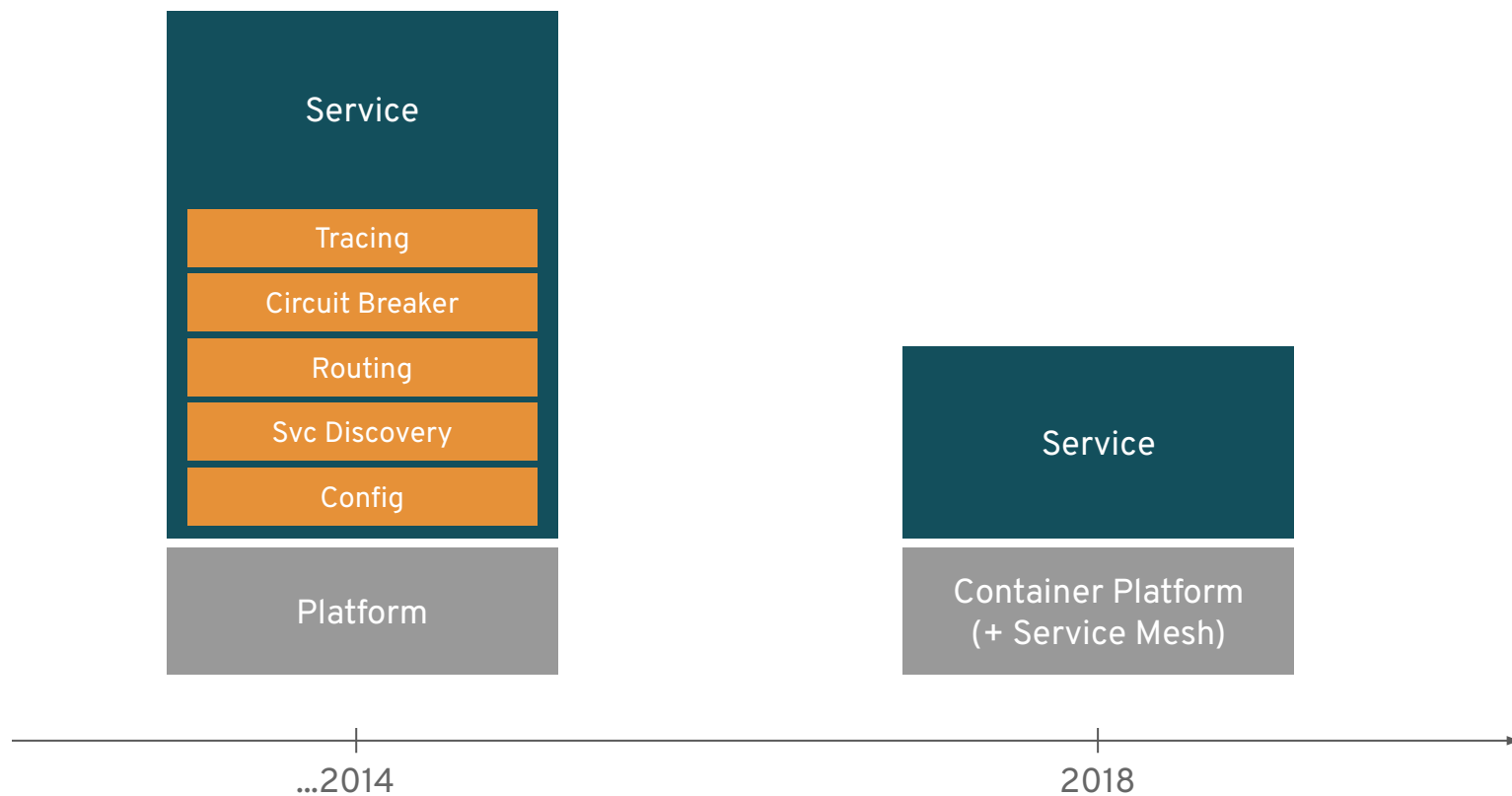
A service mesh provides a **transparent** and **language-independent** network for connecting, observing, securing and controlling the connectivity between services.

Red Hat

Istio

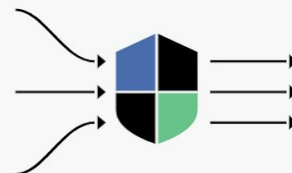Connect, secure, control, and observe services.

Connect

Intelligently control the flow of traffic and API calls between services, conduct a range of tests, and upgrade gradually with red/black deployments.
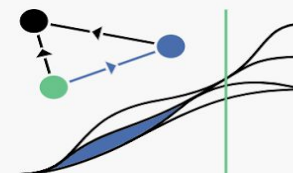
Secure

Automatically secure your services through managed authentication, authorization, and encryption of communication between services.

Control

Apply policies and ensure that they're enforced, and that resources are fairly distributed among consumers.
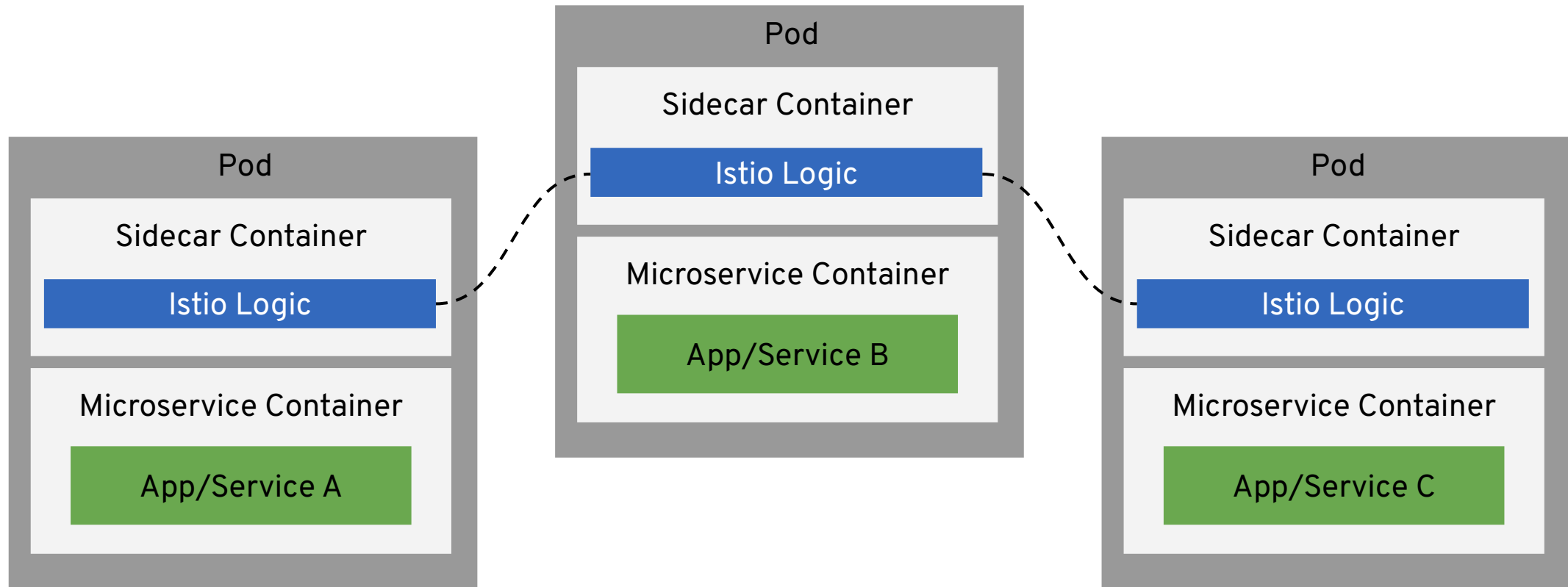
Observe

See what's happening with rich automatic tracing, monitoring, and logging of all your services.

MICROSERVICES WITH ISTIO

connect, manage, and secure microservices _transparently_

WHAT IS A SIDECAR?

A proxy instance that abstracts common logic away from individual services

SIDECAR PATTERN

- A utility container in the same pod to enhance the main container's functionality
- Share the same network and lifecycle
- Istio uses an Istio Proxy (L7 Proxy) sidecar to proxy all network traffic between apps

POD

SIDECAR

APP

Red Hat
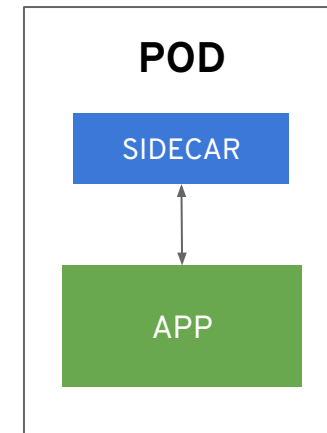
ISTIO PROVIDES BOTH CONTROL AND DATA PLANES

The **data plane** is composed of a set of intelligent proxies (Envoy) deployed as sidecars that mediate and control all network communication between microservices.

The **control plane** is responsible for managing and configuring proxies to route traffic, as well as enforcing policies at runtime.
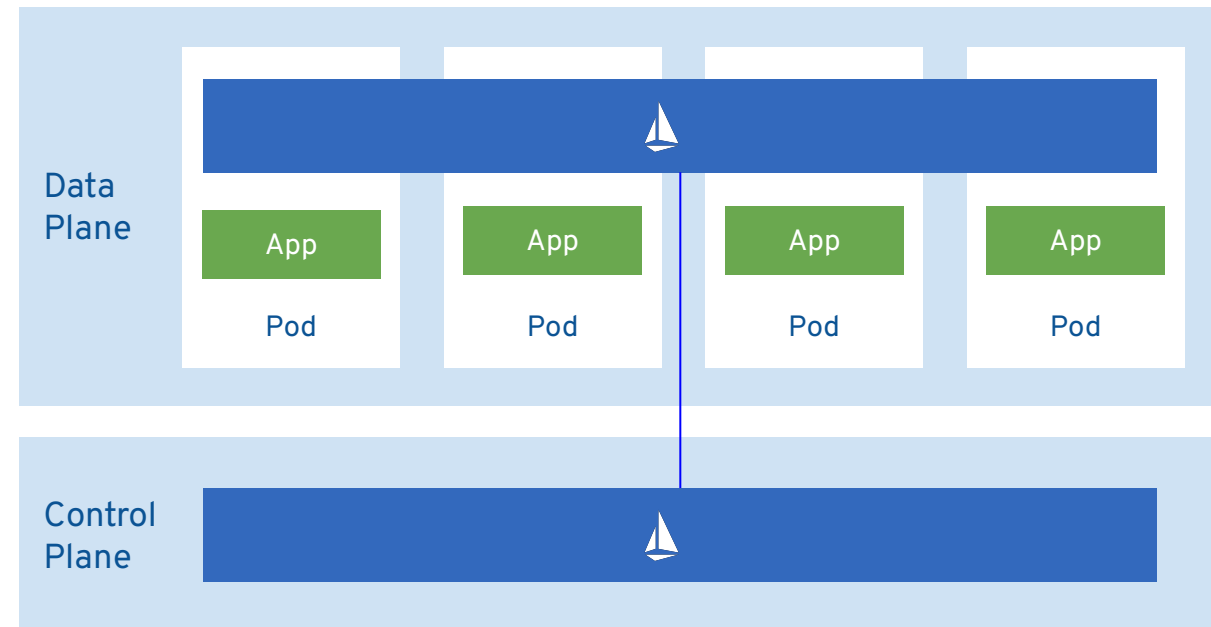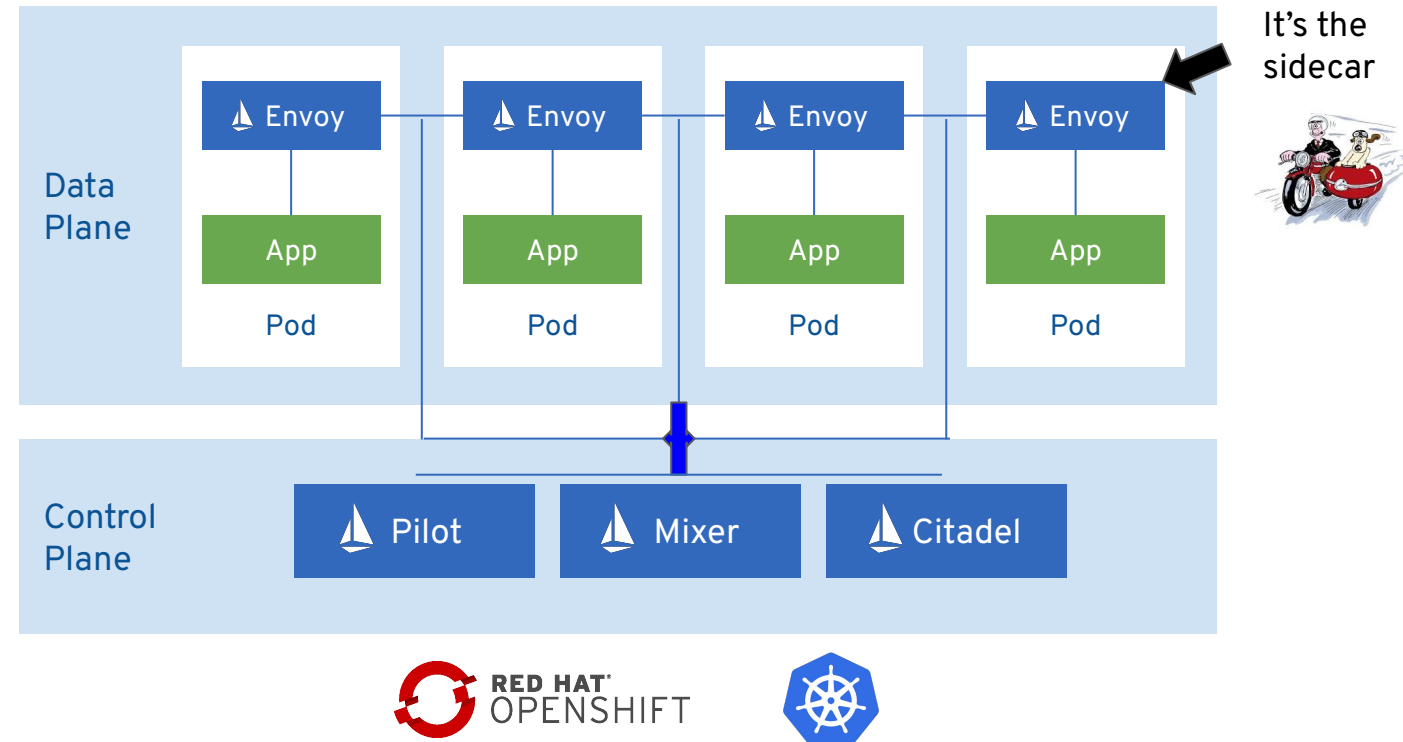
COMPONENTS OF ISTIO

Envoy, originally from Lyft - it's an intelligent proxy. Highly parallel non-blocking, network filtering, service discovery, health checking, dynamically configurable.
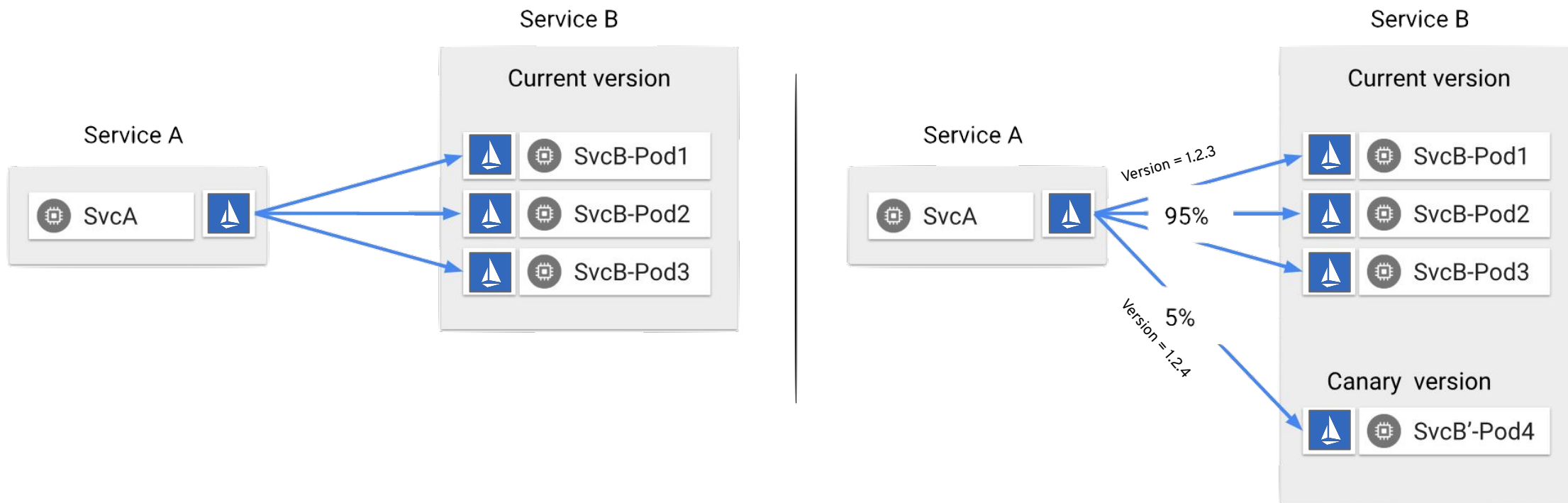
Pilot, the component responsible for managing a distributed deployment of Envoy proxies in the service mesh. Intelligent routing, traffic mgmt, resiliency

Mixer, which provides the policy and access control mechanisms within the service mesh. Monitoring, reporting, quotas - plugin-based.

Citadel, control service-service traffic based on origin and user. Key mgmt certificate authority.

It's the sidecar

Data Plane

Envoy	Envoy	Envoy	Envoy
App	App	App	App
Pod	Pod	Pod	Pod

Control Plane

Pilot	Mixer	Citadel

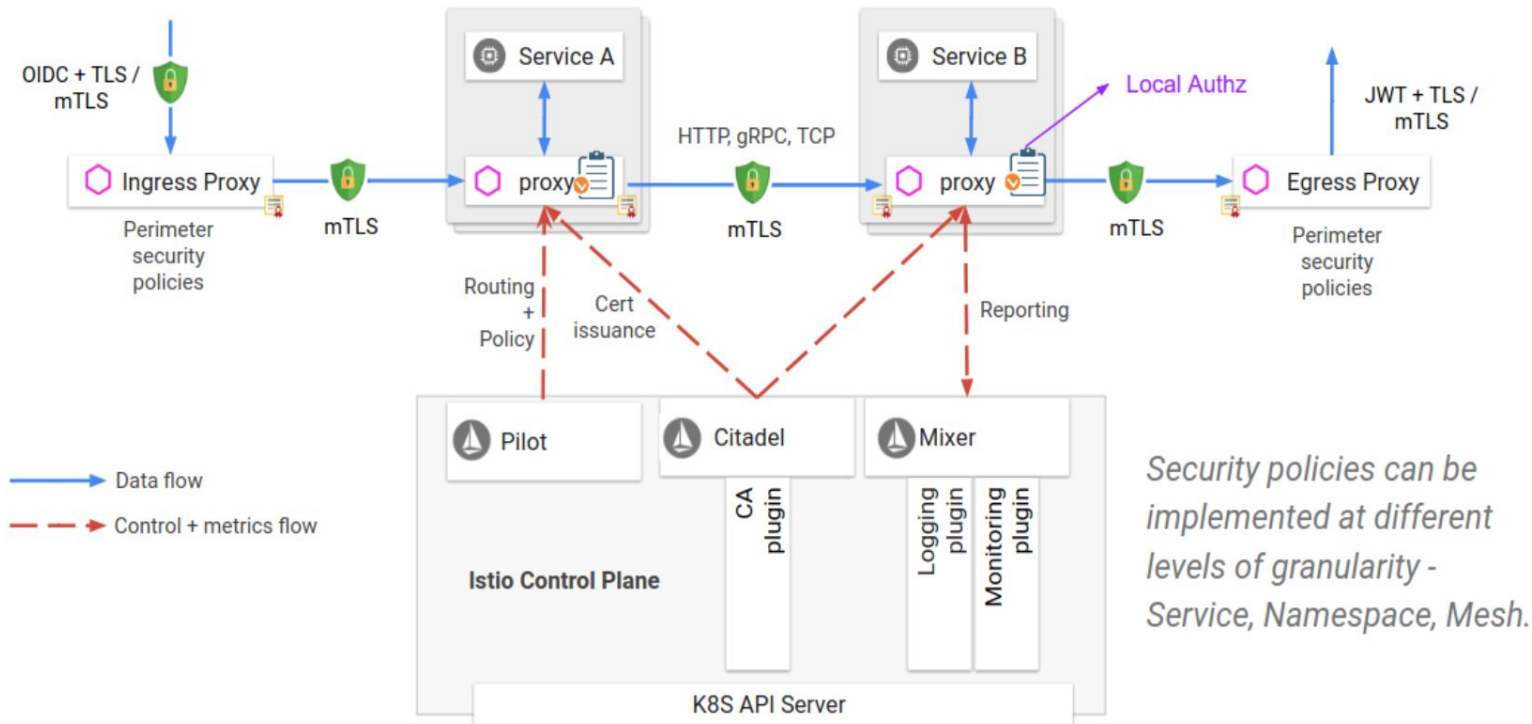RED HAT® OPENSHIFT

Red Hat

WHAT DOES <u>CONNECT</u> MEAN?

Discovery and Routing: Decoupled from infrastructure, load balancing modes, dynamic routing...
Advanced Deployments: A/B testing, gradual rollouts, canary releases, mirroring...
Failure, Health, and Testing: timeouts, retries, circuit breakers, fault injection, active health checks...
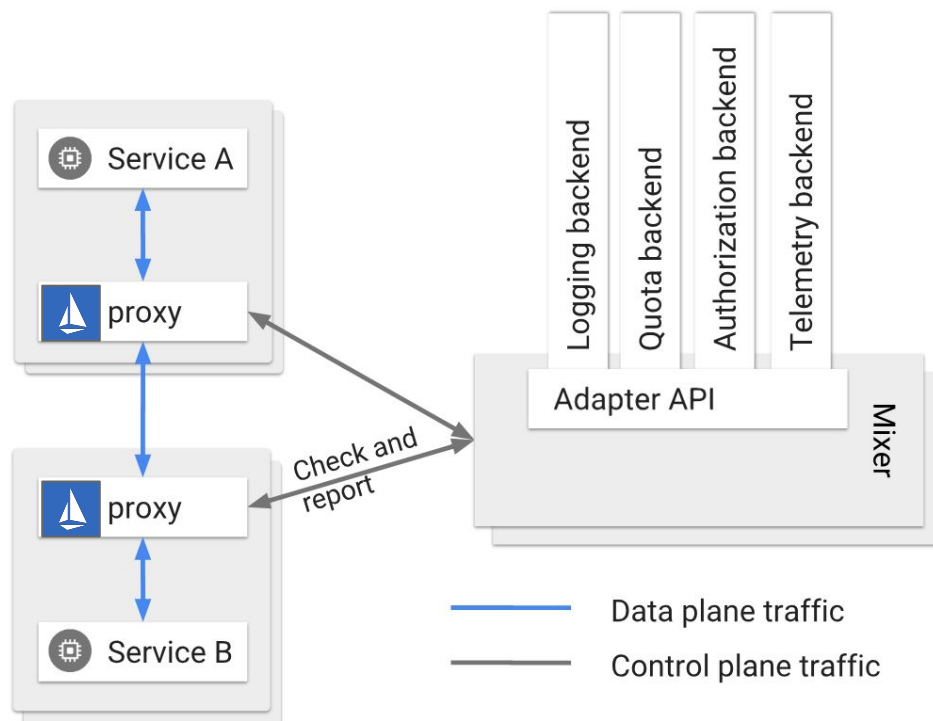
HOW DO YOU <u>SECURE</u> SERVICES?

Security by default
no changes needed for application code and infrastructure

Defense in depth
integrate with existing security systems to provide multiple layers of defense

Zero-trust network
build security solutions on untrusted networks

WHAT CAN YOU <u>CONTROL</u>?

Restrict to 2 requests per second per IP :

```
quotas:
  - name: requestcount.quota.istio-system
      overrides:
      - dimensions:
              destination: someservice
          maxAmount: 2
```
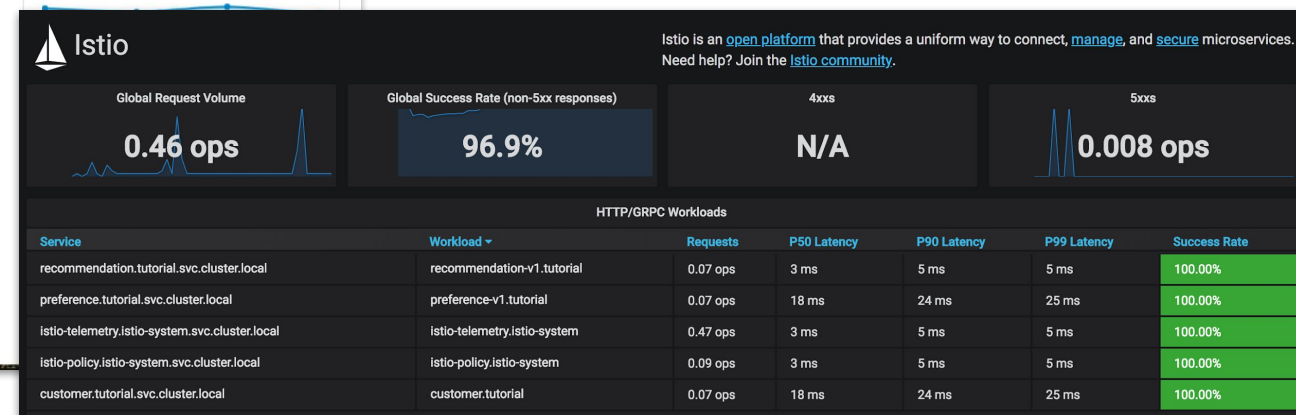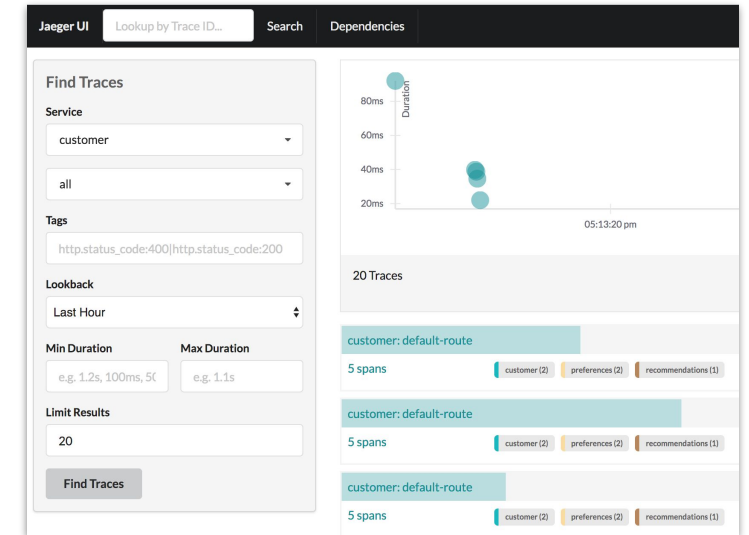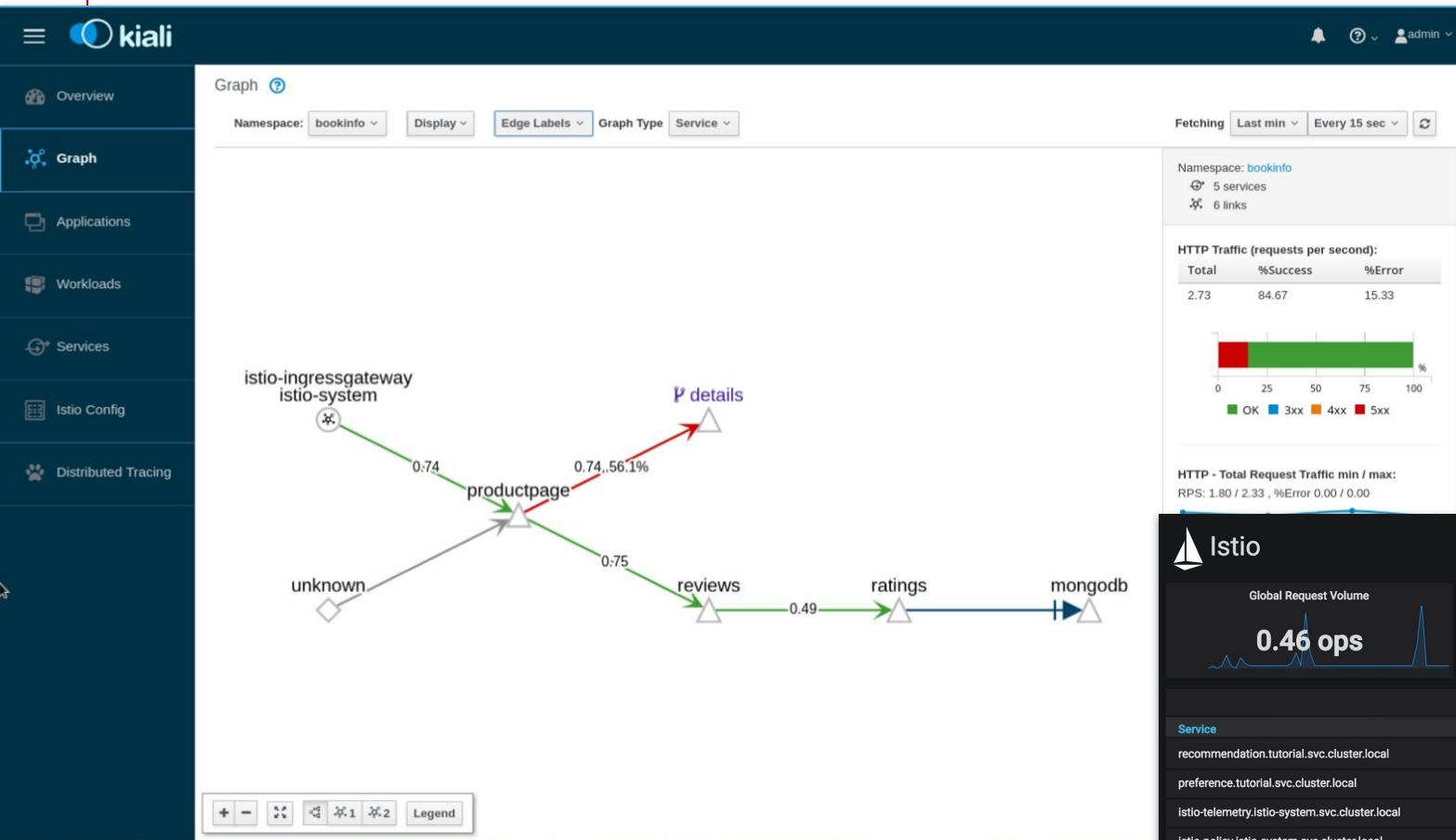
Exempt if:
```
match(request.headers["cookie"], "user=*") == false
```

Set and Check Policy: Open-ended, connection limits, rate limits, simple denials, lists

HOW CAN YOU <u>OBSERVE?</u>

Understand how your services are operating: Metrics, tracing, network visibility

Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.

linkedin.com/company/red-hat

youtube.com/user/RedHatVideos

facebook.com/redhatinc

twitter.com/RedHat

Red Hat