Networking for Hybrid Cloud

API Management and Service Mesh (ISTIO)

Alfred Bach
abach@redhat.com

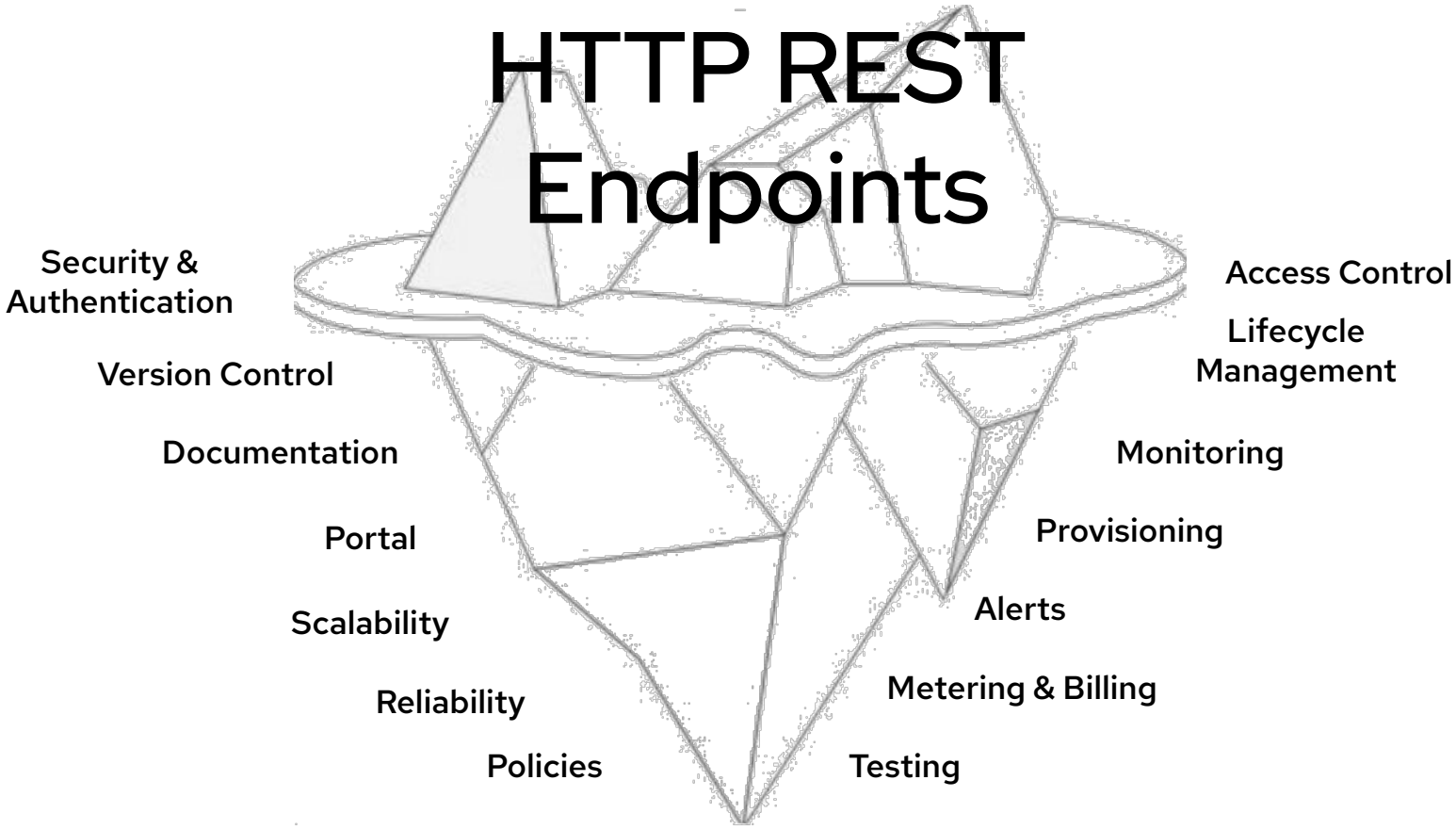
Red Hat

Red Hat 3scale API Management

For more in-depth information on Red Hat 3scale API Management please check the Architectural Overview

Take Control of Your APIs

Creating & Exposing APIs is just the start



Red Hat 3scale API Management

Flexible Distributed Control

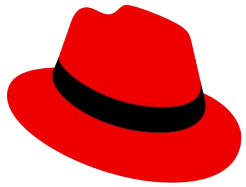
100% Open Source

Modular

No single point of failure

Container Native

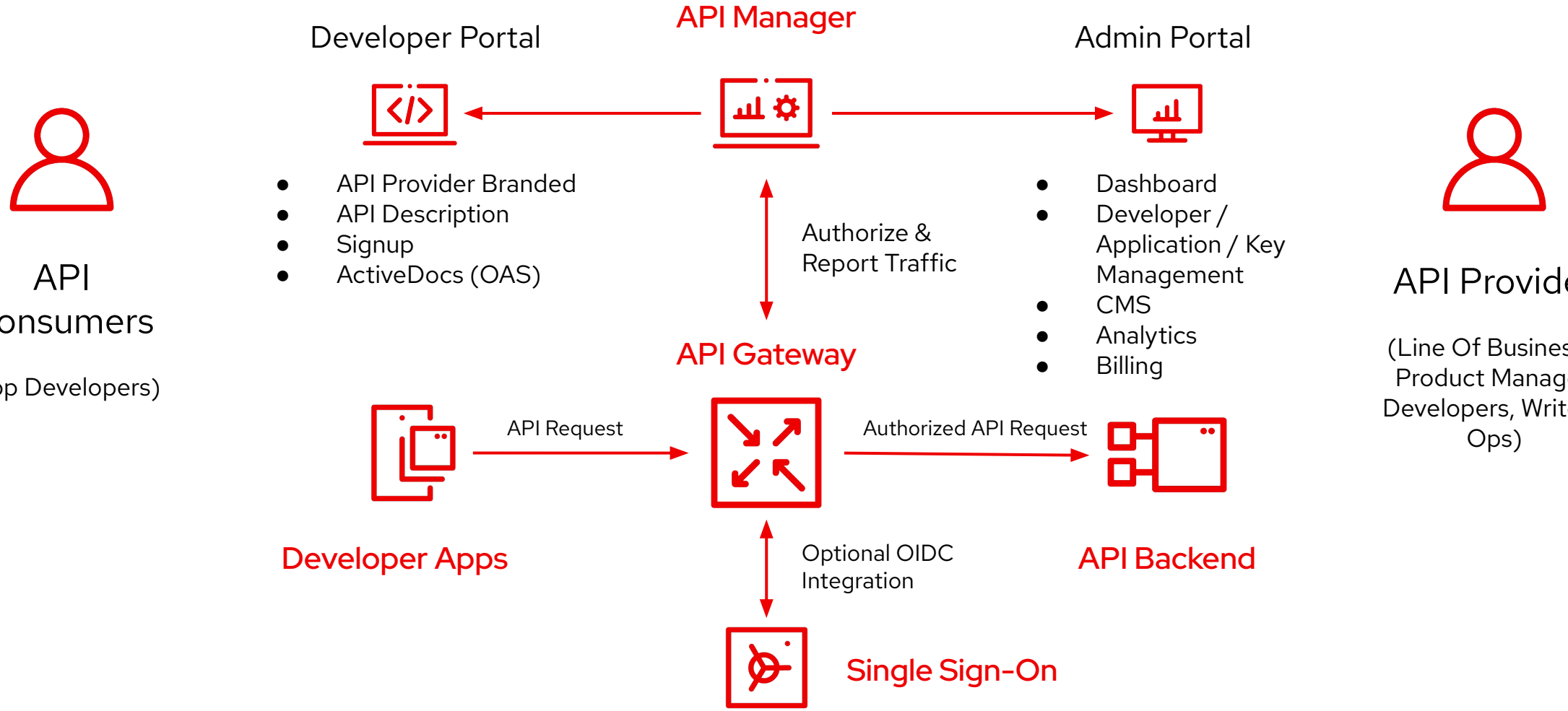
Highly scalable



Red Hat 3scale API Management

Source:

Red Hat 3scale API Management



Authentication

Ensure every client application is uniquely identified and can prove its identity

API Key: a shared secret used to authenticate a client application. Cannot easily be renewed.

API Key Pair: an identifier + a shared secret used to authenticate a client application. The identifier remains the same during the whole lifetime of the application, the secret can easily be renewed to ensure higher security.

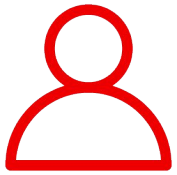
OpenID Connect: a standard protocol to authenticate the client application and the end-user connected on this application. Currently the highest level of security.

Access Control

Access Control APIs

Application Access Control strives to answer the following question:

"Is this client application allowed to call this API or subset of this API ?"

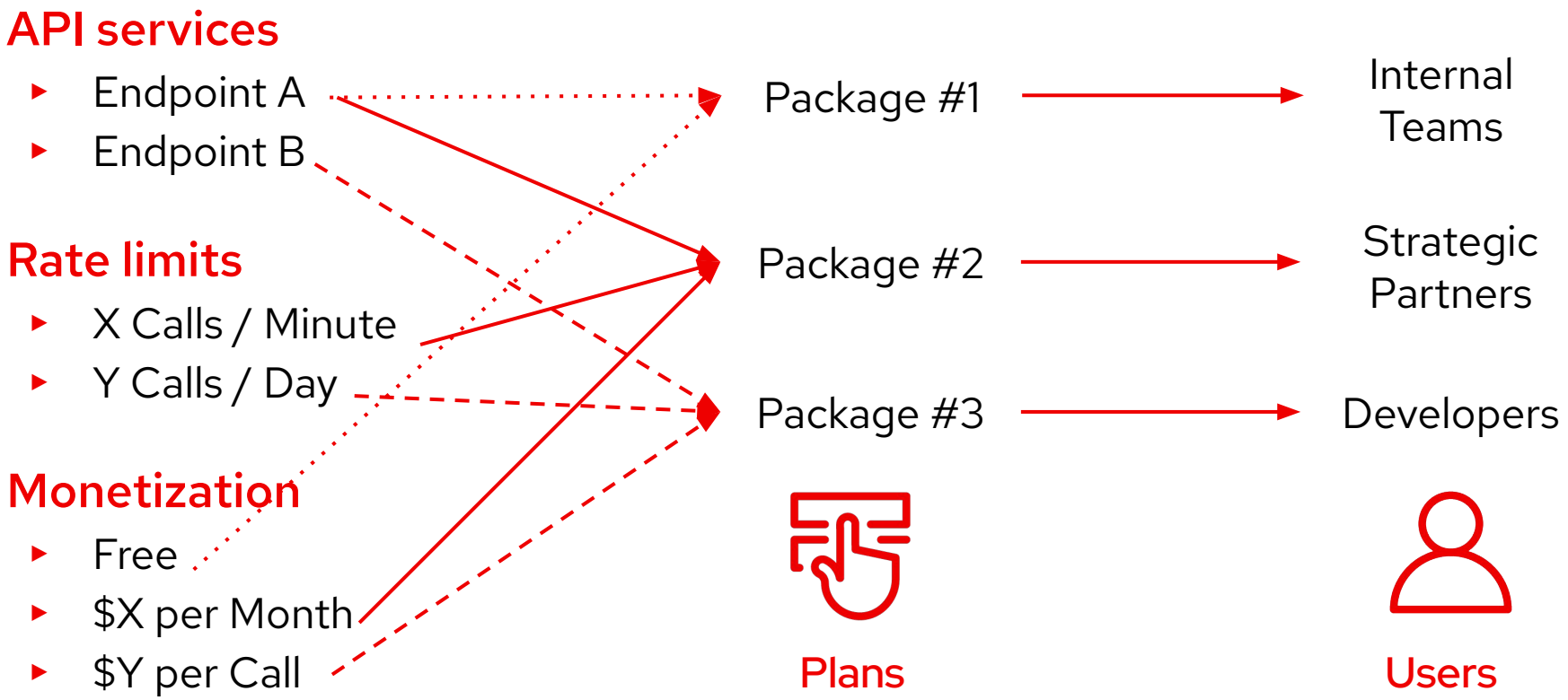


User Access Control strives to answer the following question:

"Is this end-user allowed to call this API or subset of this API ?"

API Contracts, Throttling and Limits

Package your APIs. Create access tiers. Set rate limits.



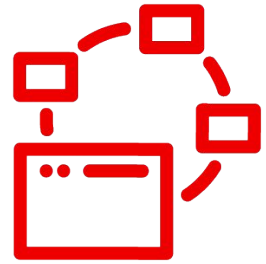
Source

API Gateway

Deployment Options

Multi Cloud

No constraints on adopting two or more deployment modes if needs vary for different APIs or different API consumers



Multi Gateways

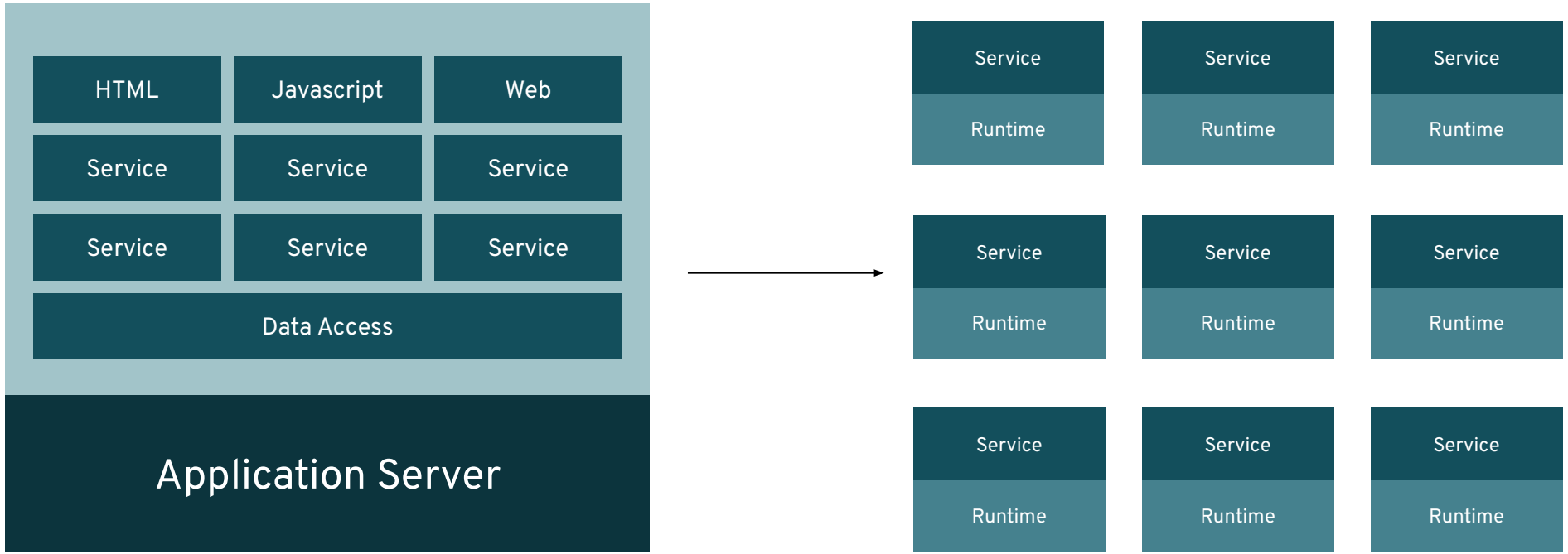
No restrictions on the number of traffic gateways used or their physical locations. Access keys issued to developers are equally valid at all locations.

Source

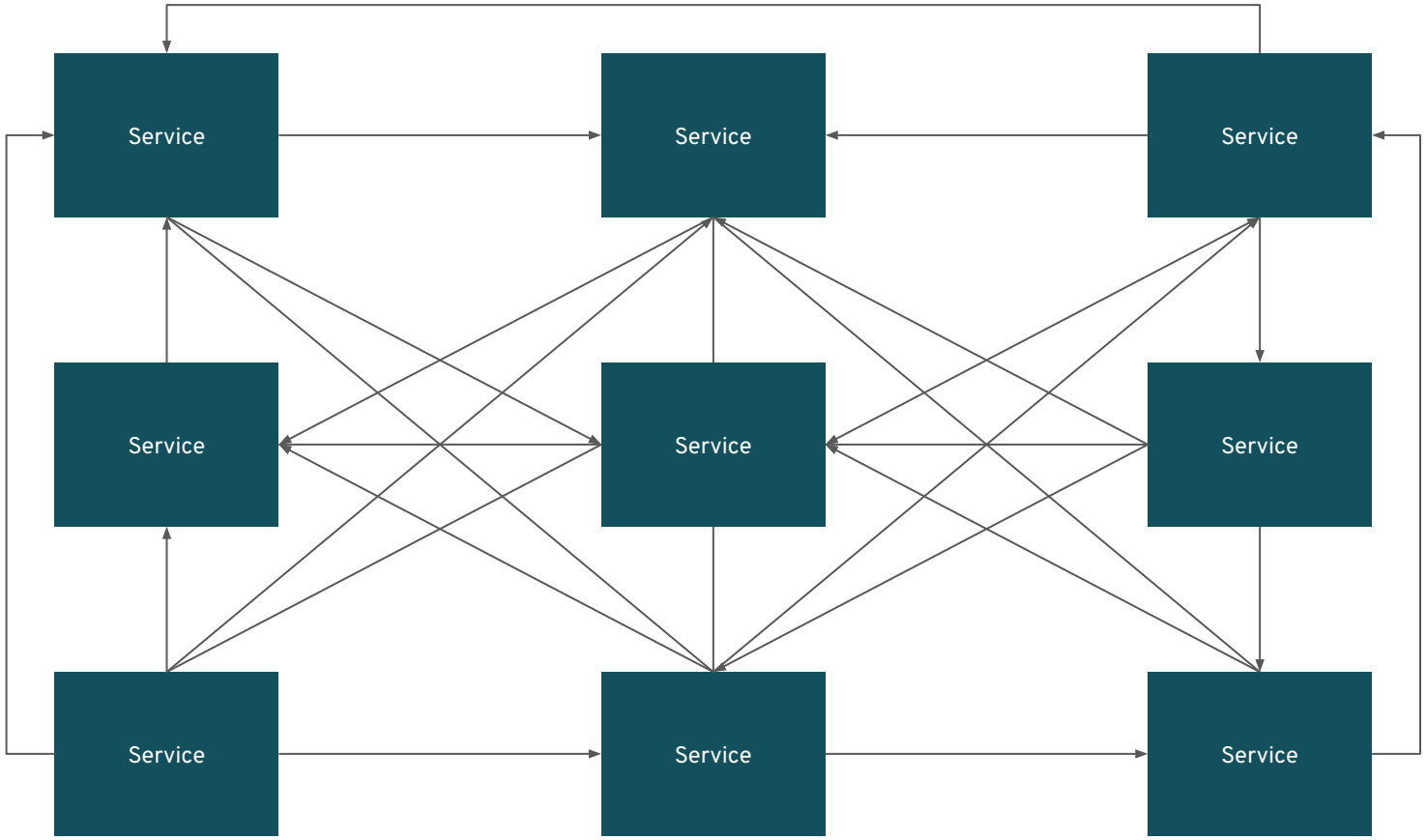
Red Hat Service Mesh (Istio)

~~MICROSERVICES~~ ARCHITECTURE

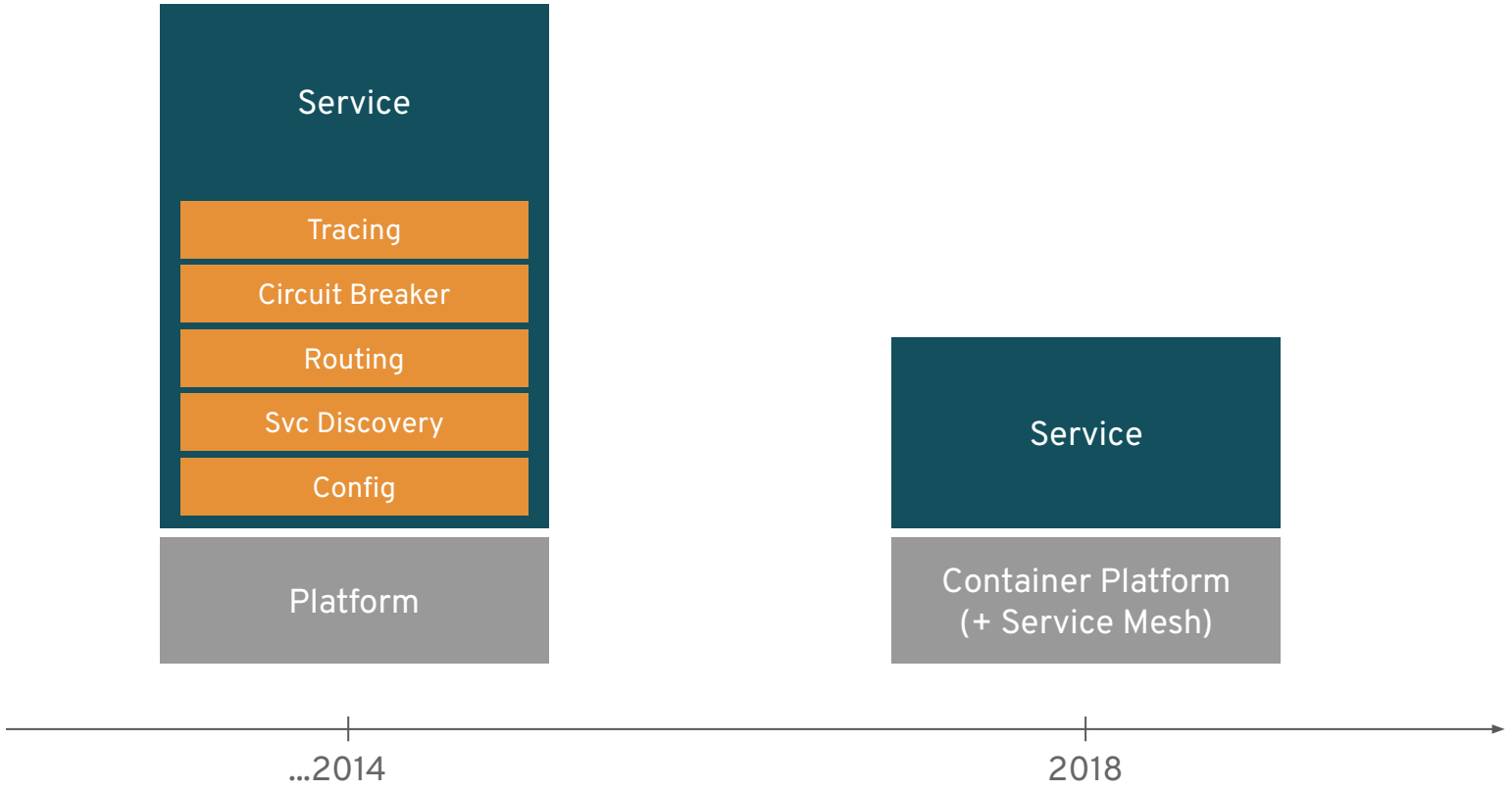
DISTRIBUTED



DISTRIBUTED ARCHITECTURE



A better way with a service mesh

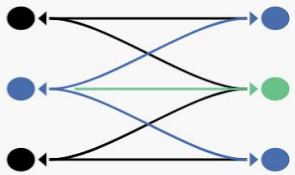


A service mesh provides a **transparent** and **language-independent** network for connecting, observing, securing and controlling the connectivity between services.

Red Hat

Istio

Connect, secure, control, and observe services.

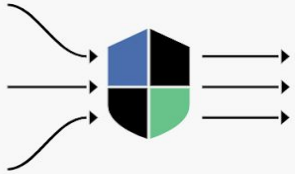


Connect

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

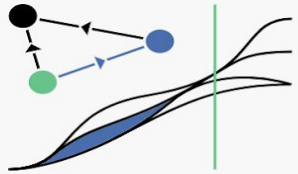
Secure

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.

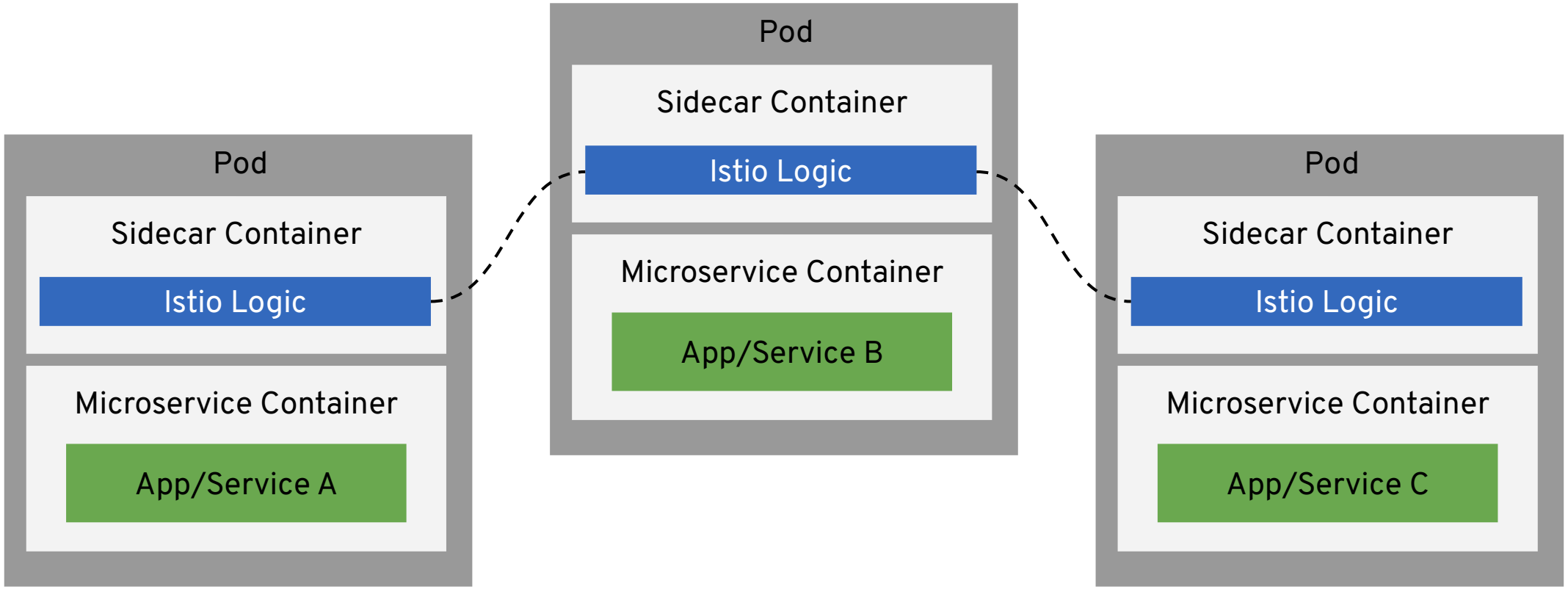


Observe

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

MICROSERVICES WITH ISTIO

connect, manage, and secure microservices *transparently*



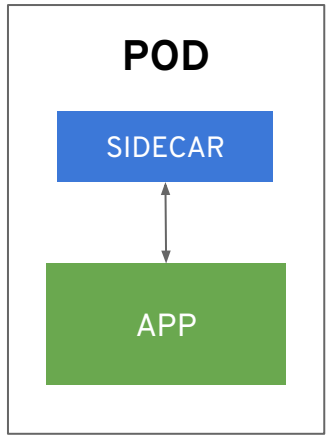
Red Hat

WHAT IS A SIDECAR?

A proxy instance that abstracts common logic away from individual services

SIDECAR PATTERN

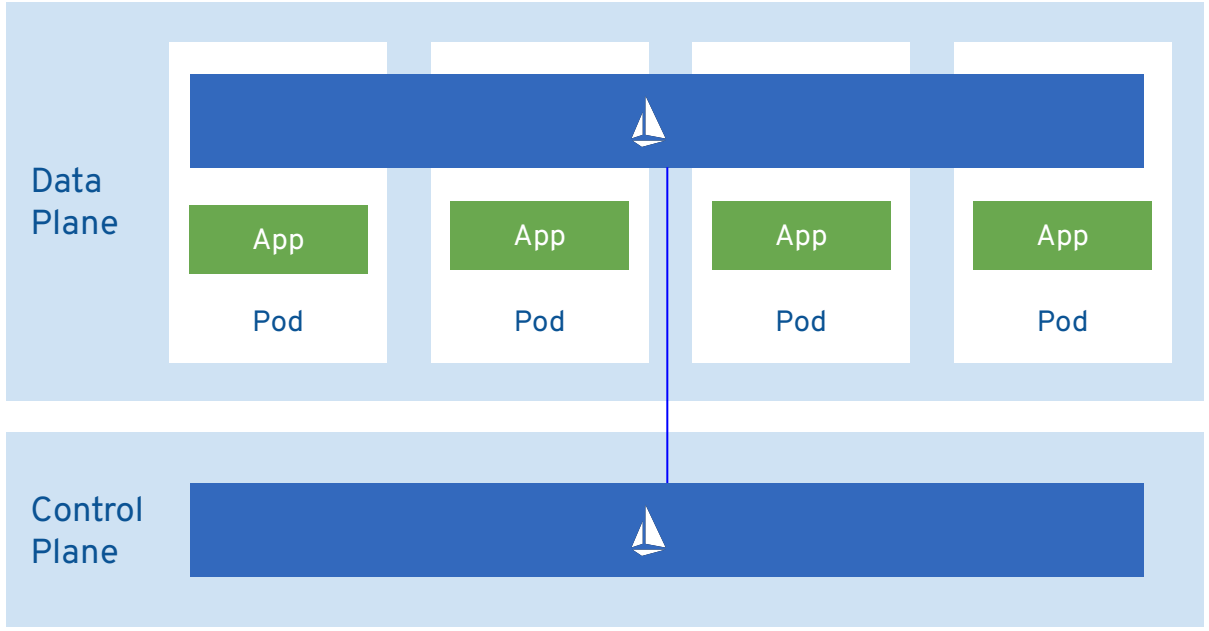
- A utility container in the same pod to enhance the main container's functionality
- Share the same network and lifecycle
- Istio uses an Istio Proxy (L7 Proxy) sidecar to proxy all network traffic between apps



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.



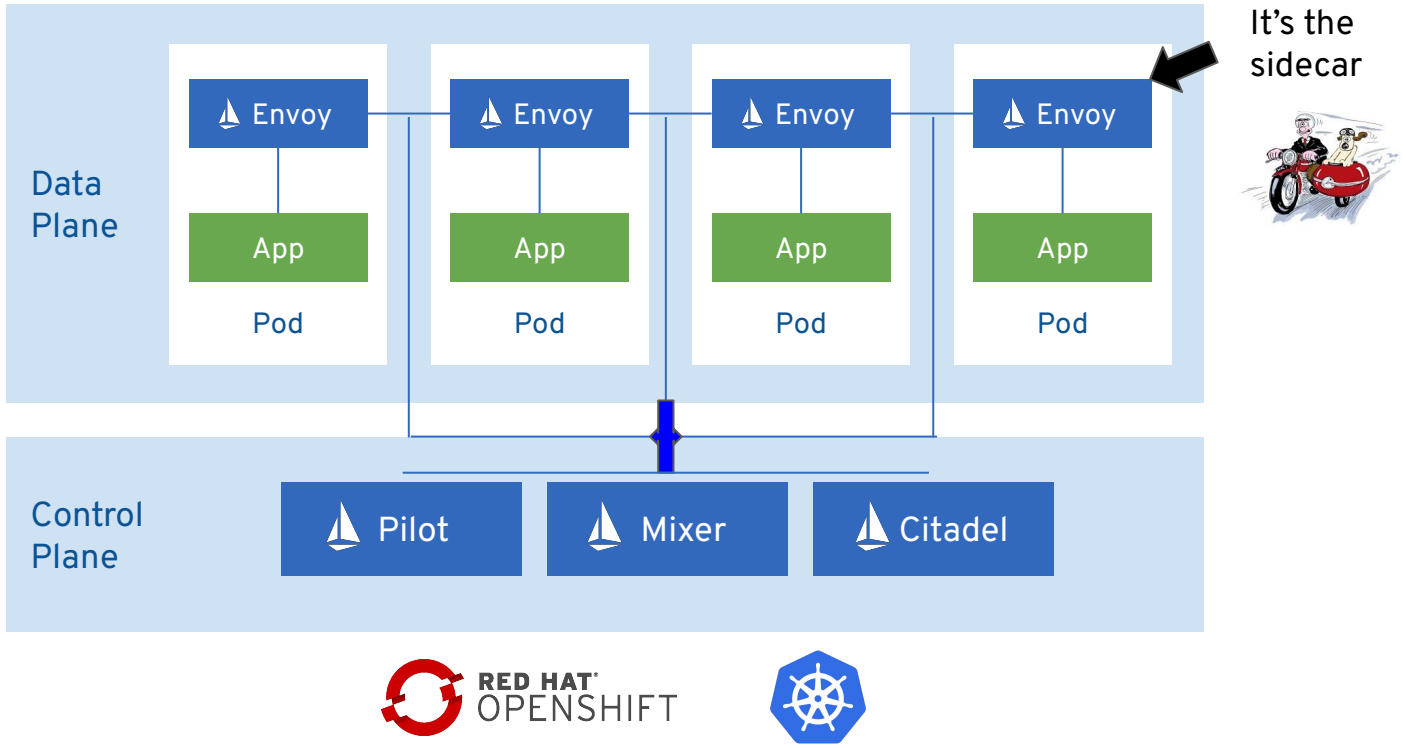
COMPONENTS OF ISTIO

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

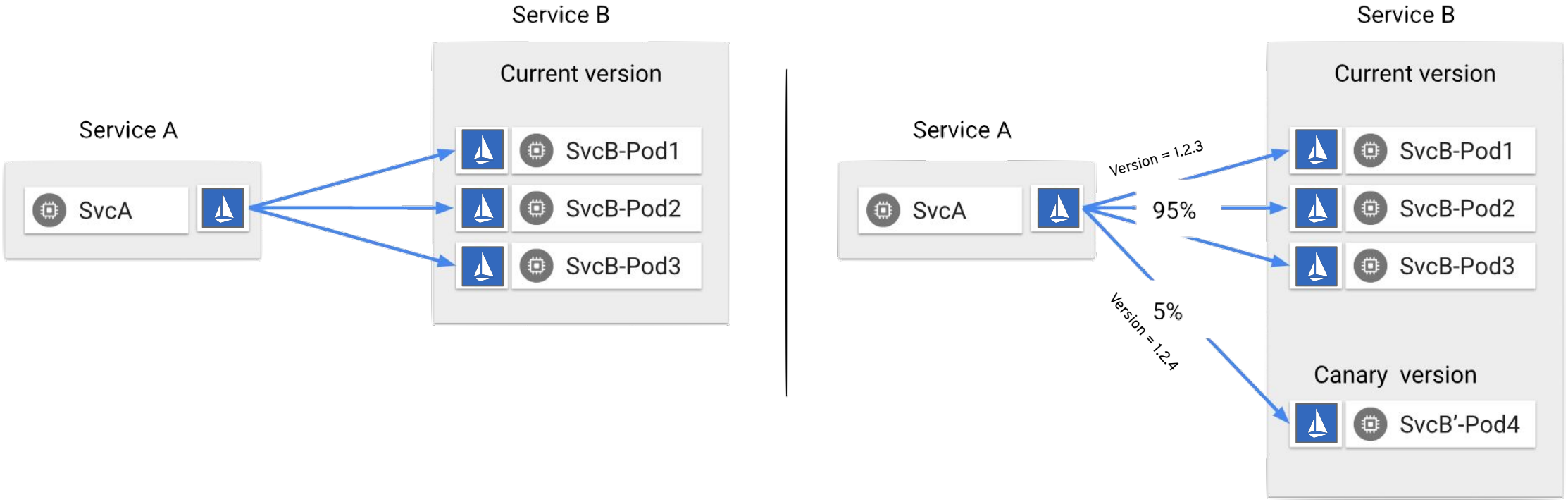
Mixer, which provides the policy and access control mechanisms within the service mesh. Monitoring, reporting, quotas - plugin-based.

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



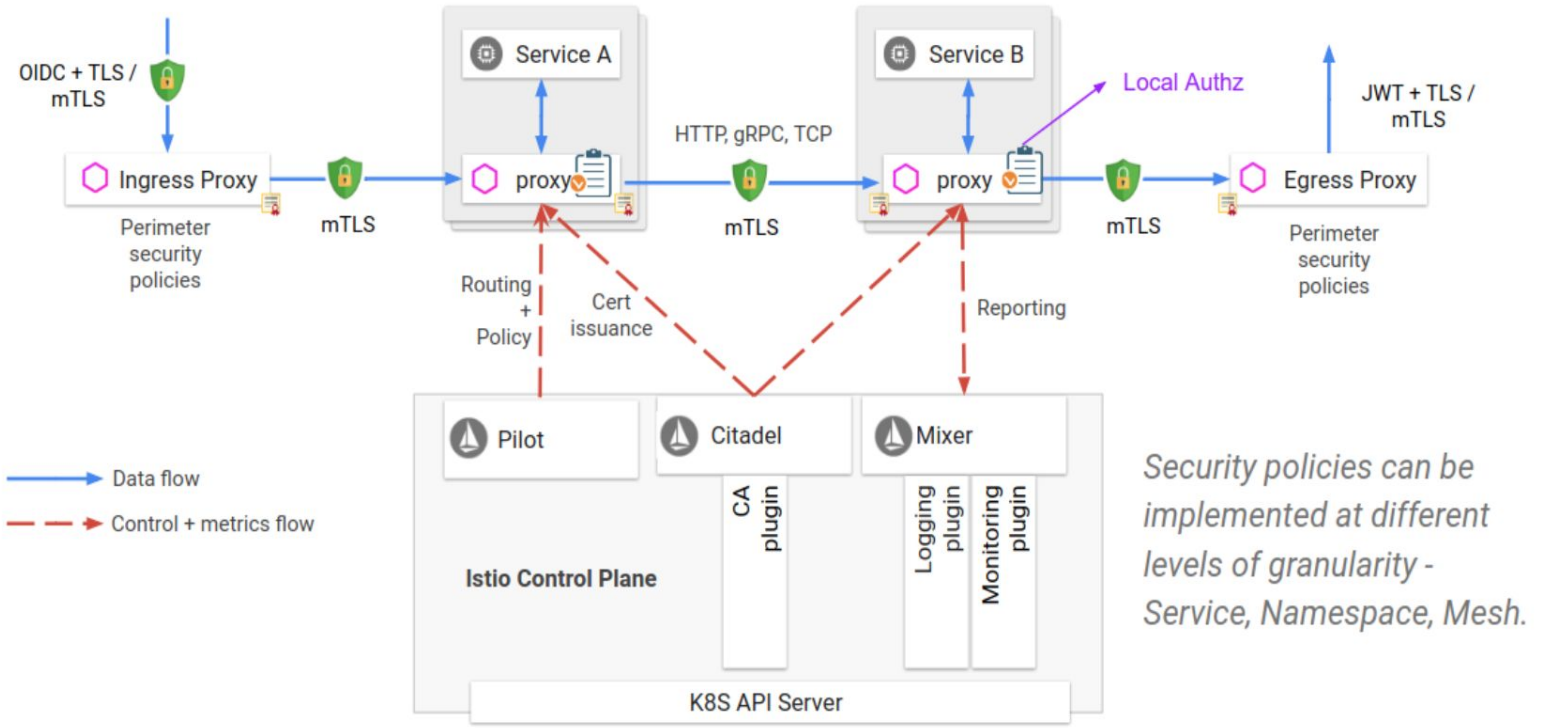
Red Hat

WHAT DOES CONNECT 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 SECURE SERVICES?

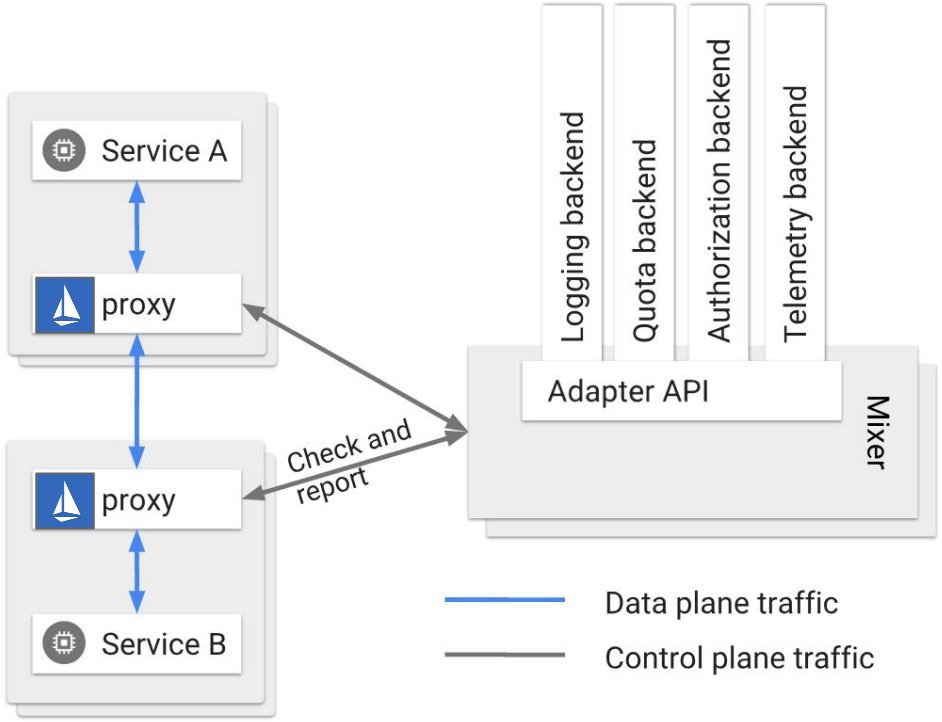


Security by default
no changes needed for application code and infrastructure

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 CONTROL?



Restrict to 2 requests per second per IP :

```
quotas:
  - name: requestcount.quota.istio-system
      overrides:
      - dimensions:
              destination: someservice
                maxAmount: 2
```

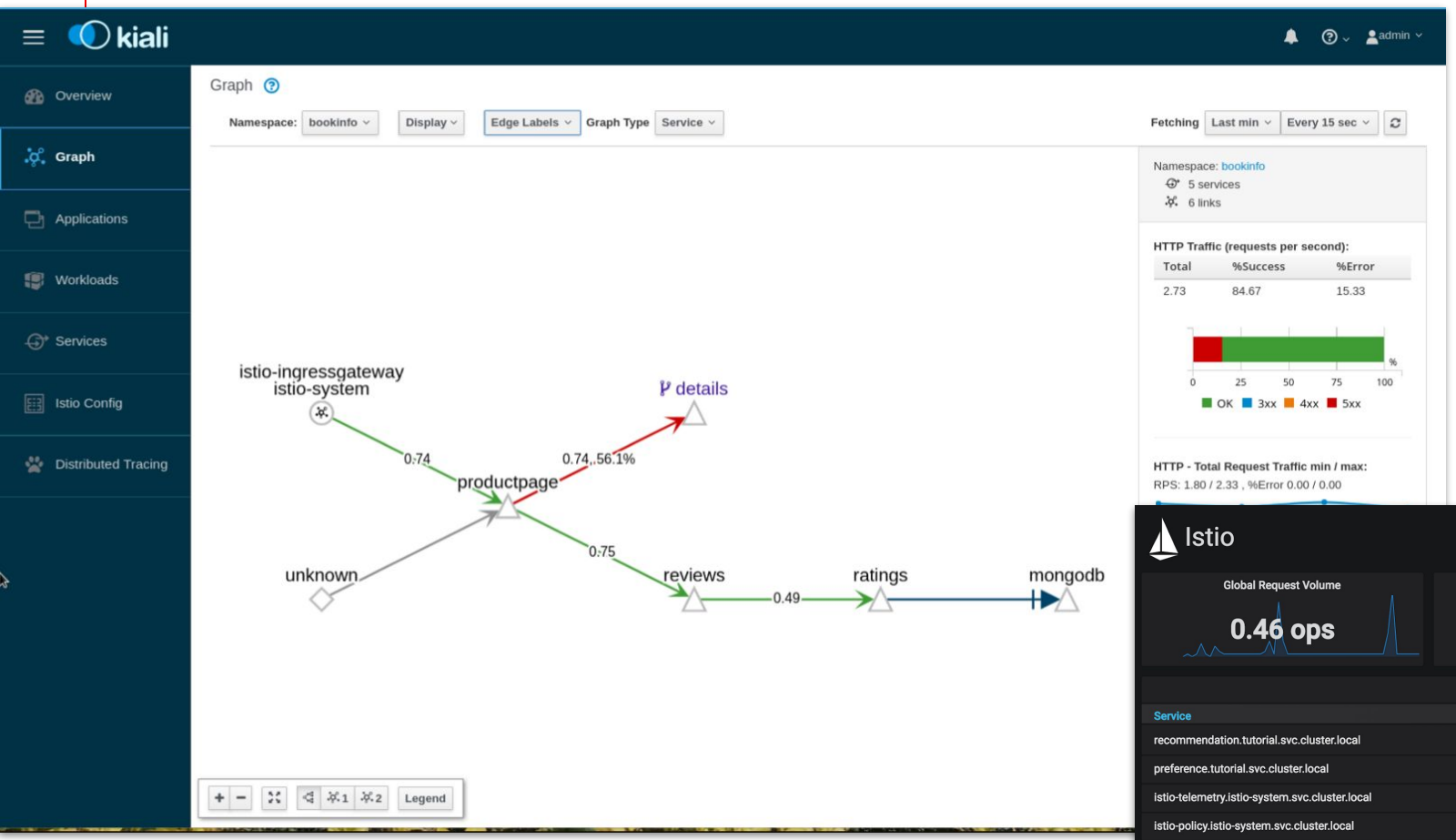
Exempt if:

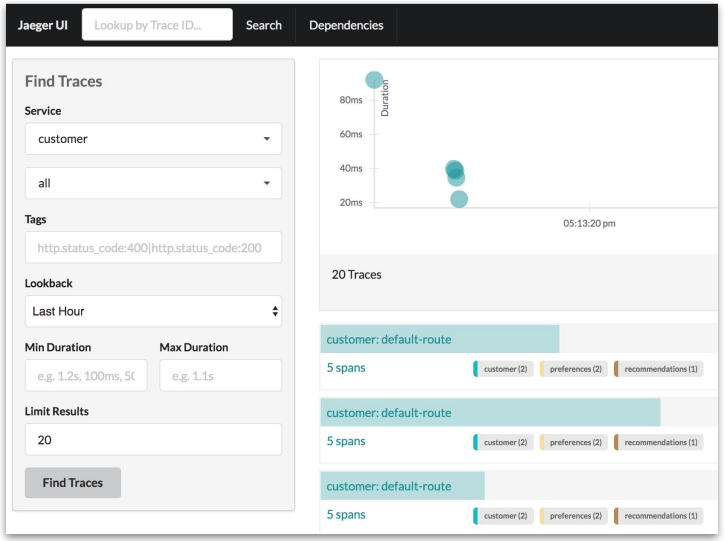
```
match(request.headers["cookie"], "user=*") == false
```

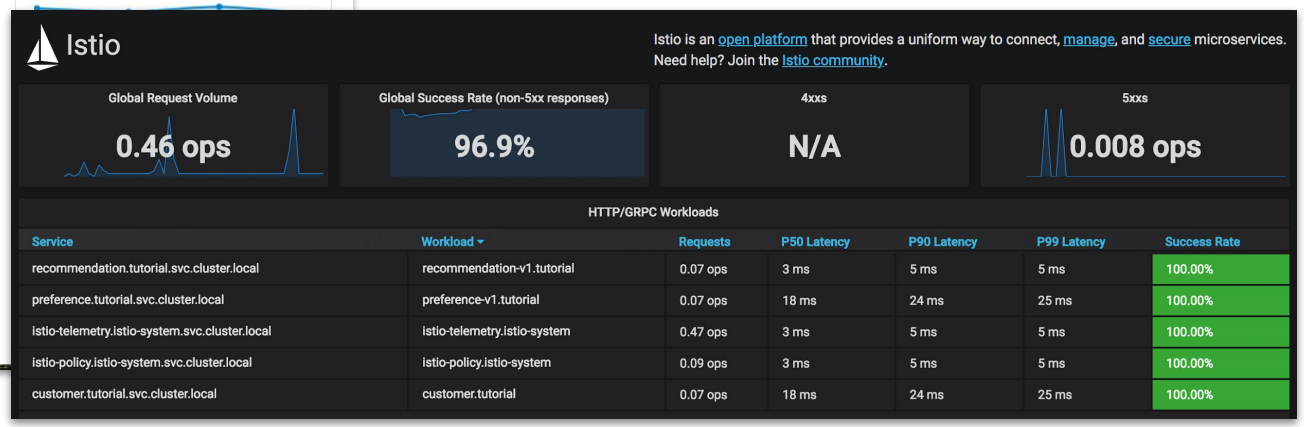
Set and Check Policy: Open-ended, connection limits, rate limits, simple denials, lists

Red Hat

HOW CAN YOU OBSERVE?

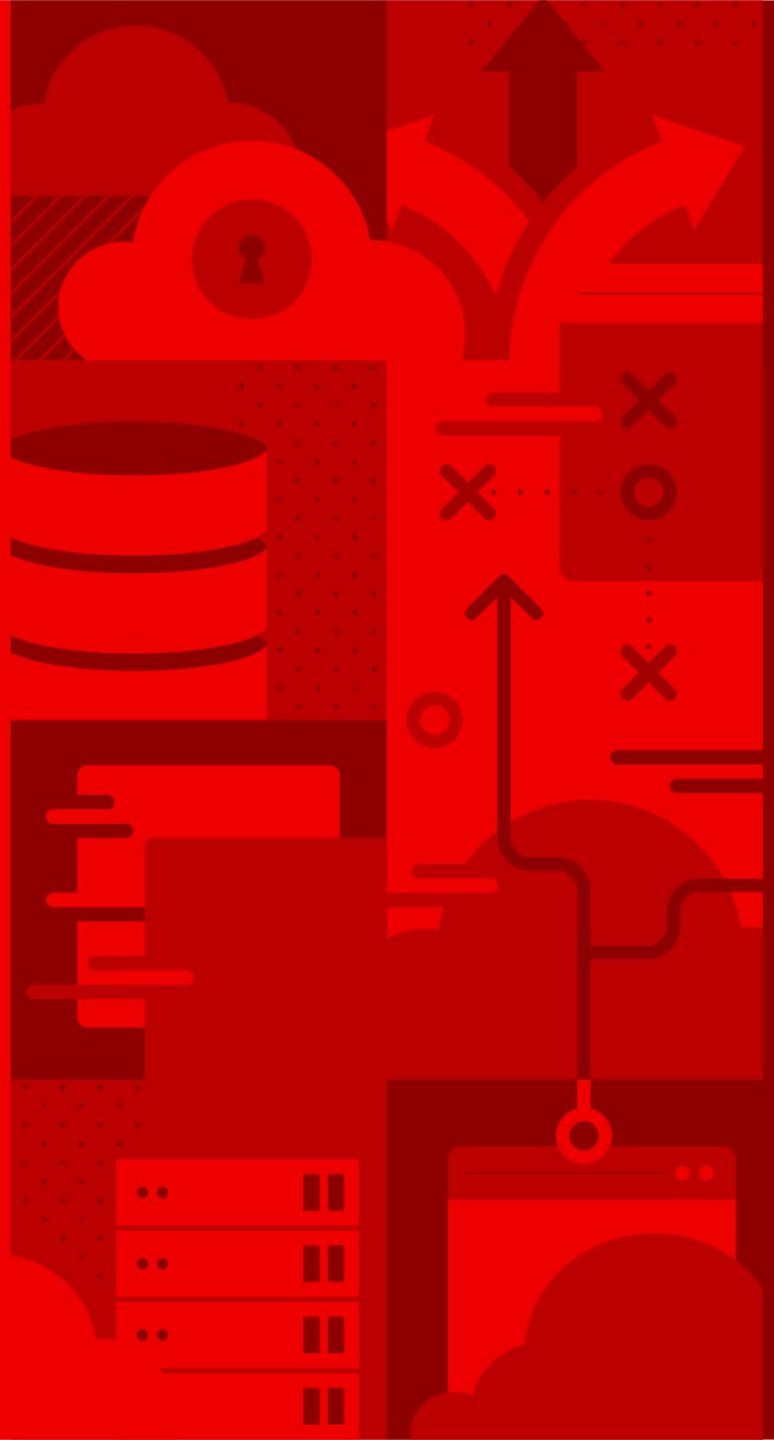






Understand how your services are operating: Metrics, tracing, network visibility

Red Hat



Application Connectivity

Drivers for Hybrid Multi-Cloud

Security & Compliance

Regional regulations, internal company wide policy enforcement. Industry specific rules. National supervisory requirements.

GeoLocation

Closer to business. Closer to Help-center establishment. Map workload. Expand geographical coverage.

IT Agility

Choose right cloud for your workload. Keep options open. Better when cross-cloud resilience applied.

Data Gravity

Data close to where it's heavily used. Less ingress/egress traffic. Data Lake access offering choices.

Flexibility

Avoid vendor lock-in, deploy close to development center. Backup and contingency plan. Exit strategy. Optimize limited budgets.

Better Solution Offerings

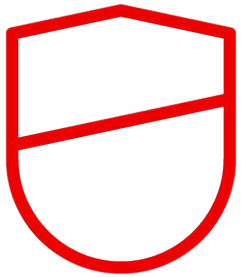
Cloud vendors offer better service on certain areas.

Hybrid Multi-Cloud Challenges

Workloads continue to exist on-premise, SaaS and in multiple clouds

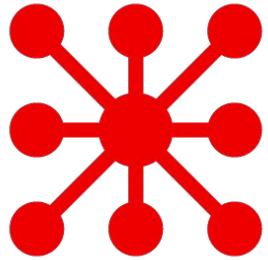
Service Silos

Isolated services because of geo regulation, company structured teams, deployment policies, or vendor specific features.



Data Integrity

Data needs to be sync across multi-cloud while real-time event stream processing still requires availability, correctness, and consistency.



Integration Mess

Point-to-point lead to a disarray of connections. Changes, upgrades or migrations take more time, resources, and money.

App Connectivity In The Real World

Has a lot of brownfield applications

A diverse mix of environments:

- Multiple versions of OpenShift
- Kubernetes from other providers
- Bare metal and VMs
- Legacy systems (old unixes, mainframes)

Complex network topologies:

- NAT and firewalls
- A mix of IPv4 and IPv6
- VPNs and VPCs (and CIDR conflicts)
- Multiple administrative domains

Skupper and Interconnect Router

Interconnect Router is the *data plane* (mature):

- 20 GA releases since 2017
- Lightweight
  - Written in C, runs purely in memory
  - A message *router*, not a broker
  - Scales down to small devices
- Routers form a fault-tolerant backbone
  - Multipath routing
  - Application-layer addressing

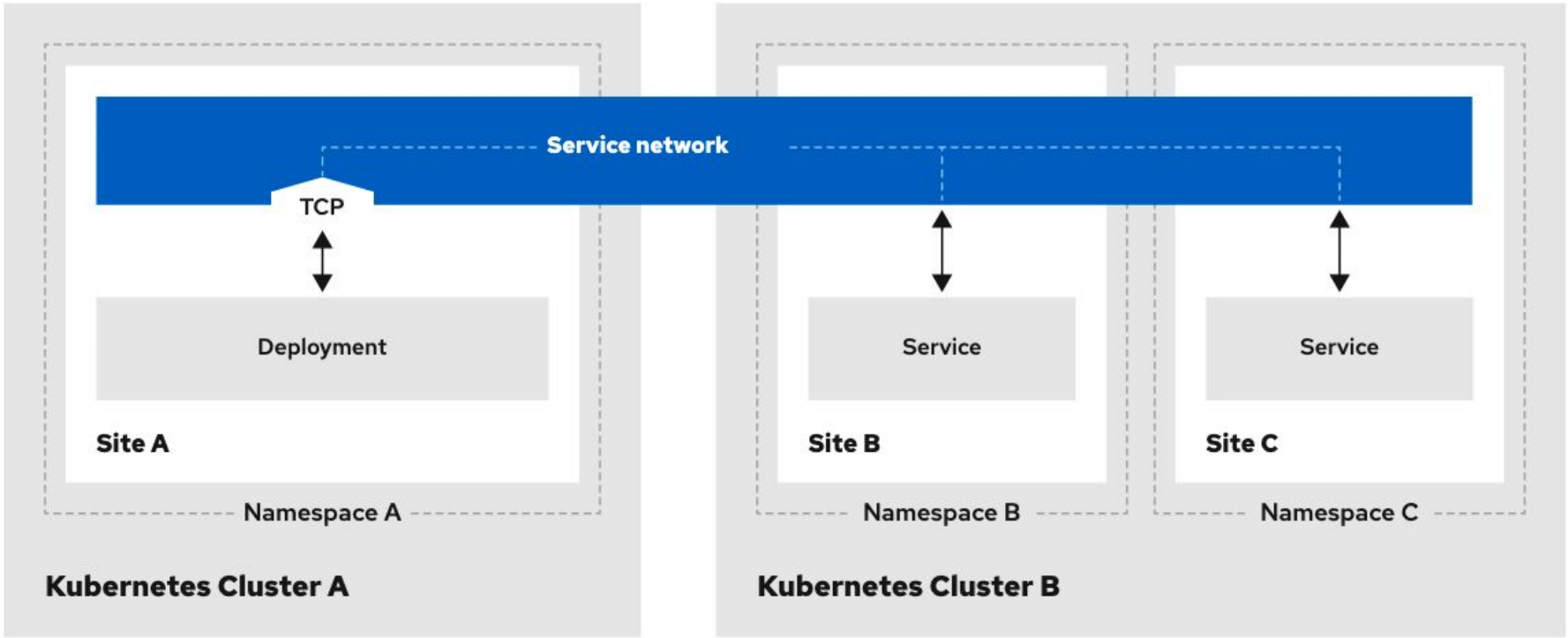
Skupper is the *control plane* (new):

- Automation for Interconnect Router
- Integration for all environments:
  - OpenShift, any Kubernetes flavor
  - Docker and Podman
  - VMs and bare metal
- Gives the *app owner* control and agility
  - You don't need cluster admin
  - You don't need new firewall rules
  - You don't need to change your app

https://qpid.apache.org/components/dispatch-router/index.html
https://skupper.io/

Application Connectivity

Using Red Hat Interconnect to create a service network



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