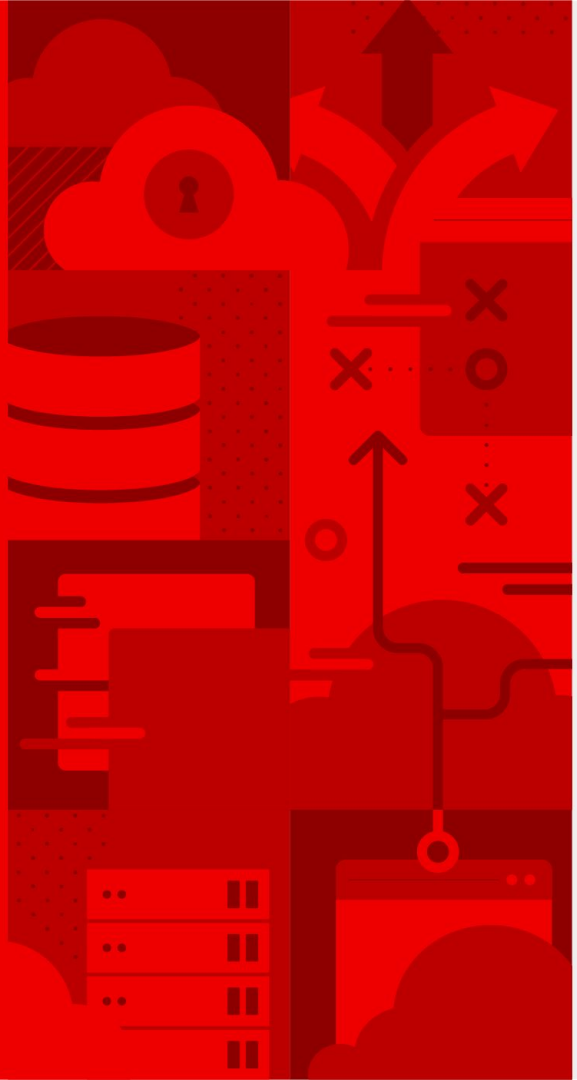


GENERAL DISTRIBUTION

OpenShift Developer

Architecture Workshop

ServiceMesh

linkedin.com/company/red-hat

facebook.com/redhatinc

youtube.com/user/RedHatVideos

twitter.com/RedHat

Red Hat

Self introduction

Name: Wanja Pernath

Email: wpernath@redhat.com

Base: Germany (very close to the Alps)

Role: EMEA Technical Partner Development Manager - OpenShift and MW

Experience: Years of Consulting, Training, PreSales at Red Hat and before

Twitter: https://twitter.com/wpernath

LinkedIn:https://www.linkedin.com/in/wanjapernath/

GitHub: https://github.com/wpernath

First book just published

Getting GitOps

A technical blueprint for developing with Kubernetes and OpenShift based on a REST microservice example written with Quarkus

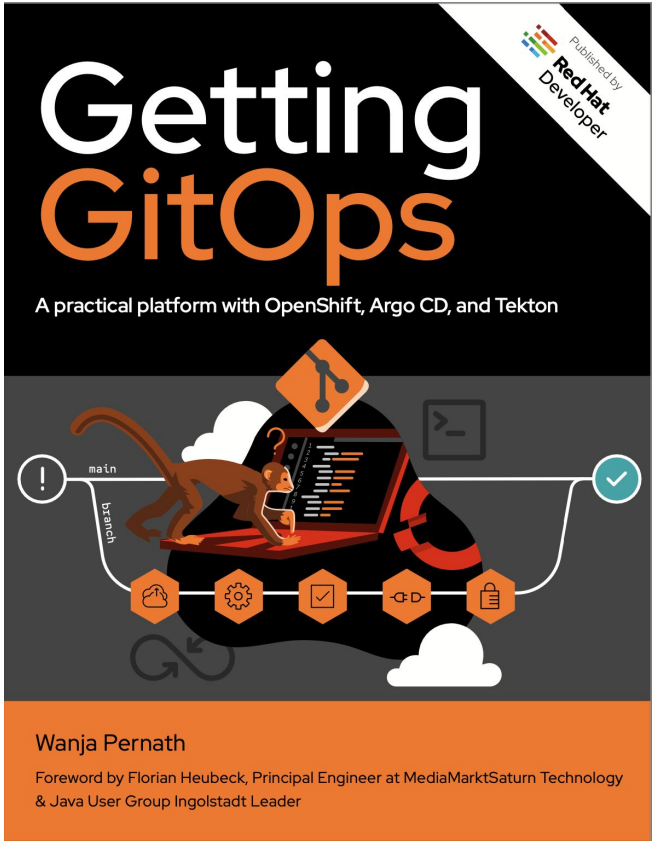
Technologies discussed:
Quarkus, Helm Charts, Kustomize, Tekton Pipelines, Kubernetes Operators, OpenShift Templates, ArgoCD, CI/CD, GitOps….

Download for free at:

https://developers.redhat.com/e-books/getting-gitops-practical-platform-openshift-argo-cd-and-tekton

Interview with full GitOps Demo:

https://www.youtube.com/watch?v=znMfVqAIRzY&ab_channel=OpenShift



Agenda

Agenda

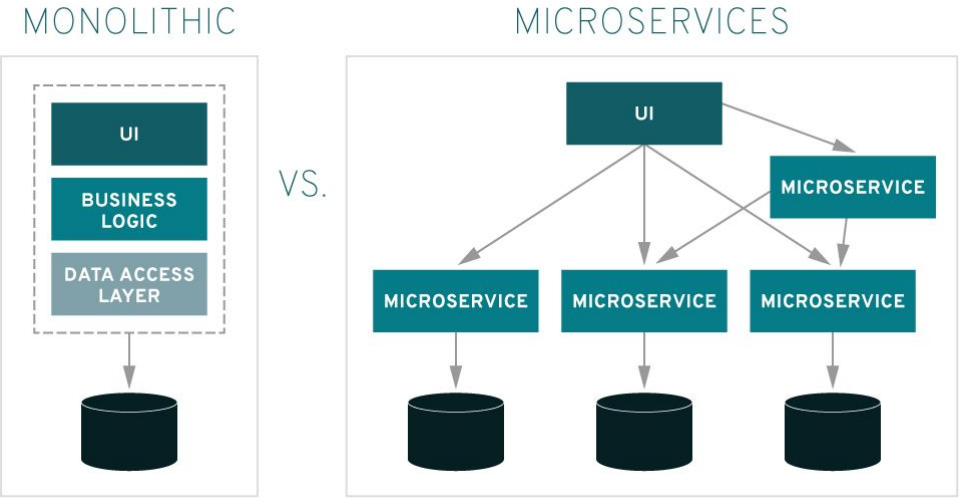
- ServiceMesh
- Summary & Thank you

ServiceMesh / Istio

What are Microservices?

an architectural style that structures an application as a collection of services

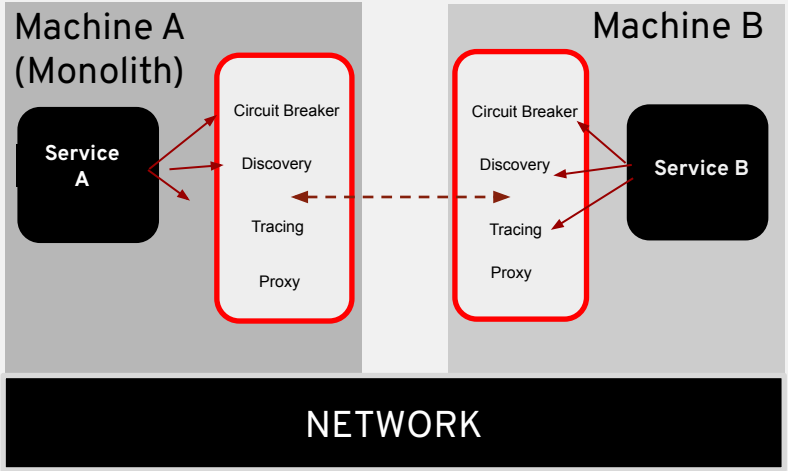
- Single purpose
- Independently deployable
- Have their context bound to a biz domain
- Owned by a small team
- Often stateless



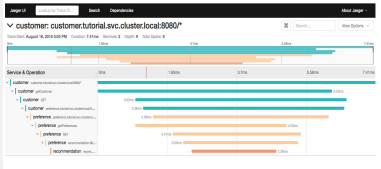
Red Hat

OVERVIEW

WHAT IS A SERVICEMESH ?



SERVICE MESH ECOSYSTEM



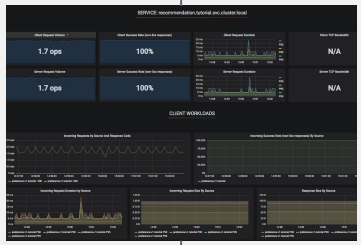
Jaeger

Secure

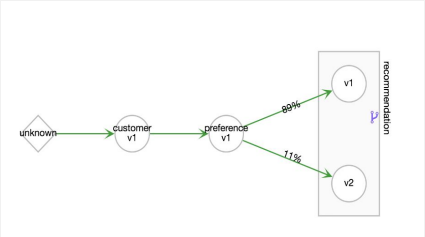
Istio

Observe

Prometheus



Observe



Connect

Control

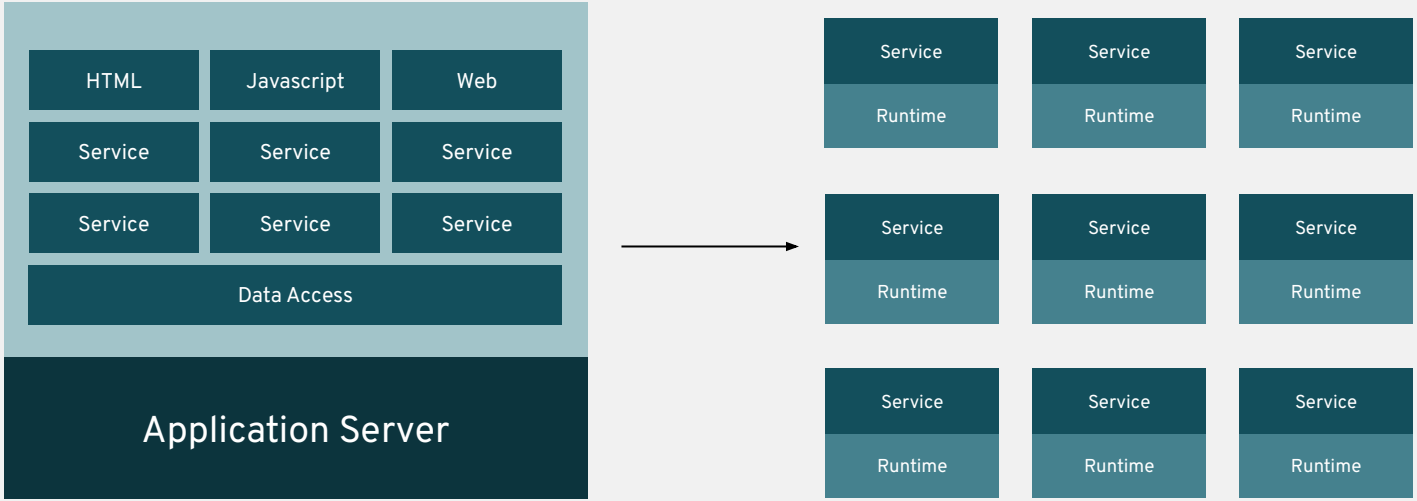
kiali

Kiali

Grafana

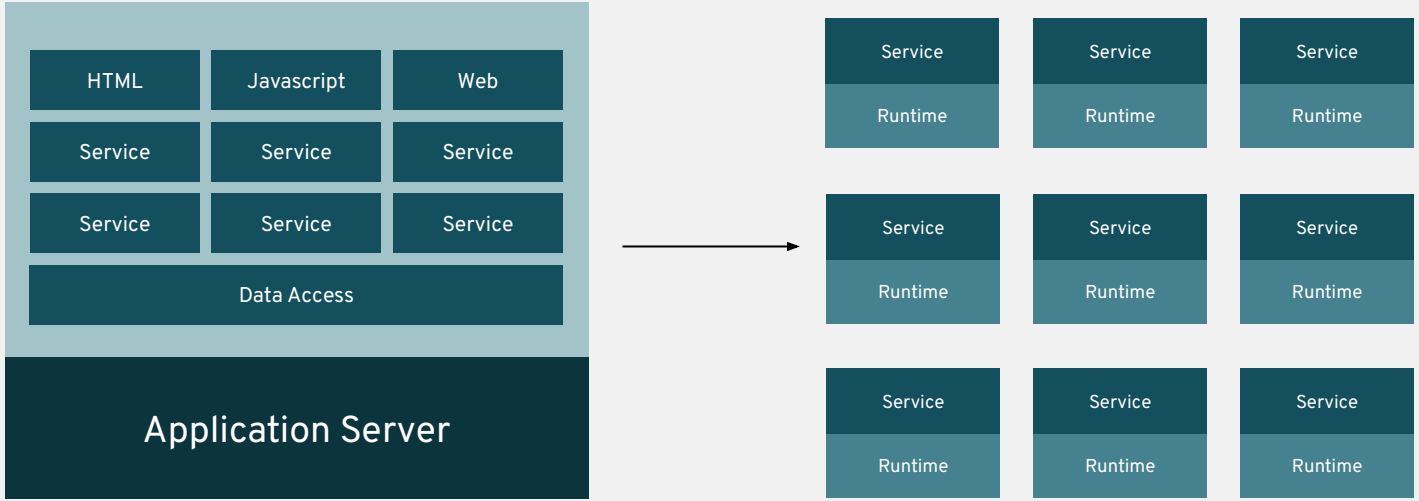
UNDER THE HOOD

MICROSERVICES ARCHITECTURE

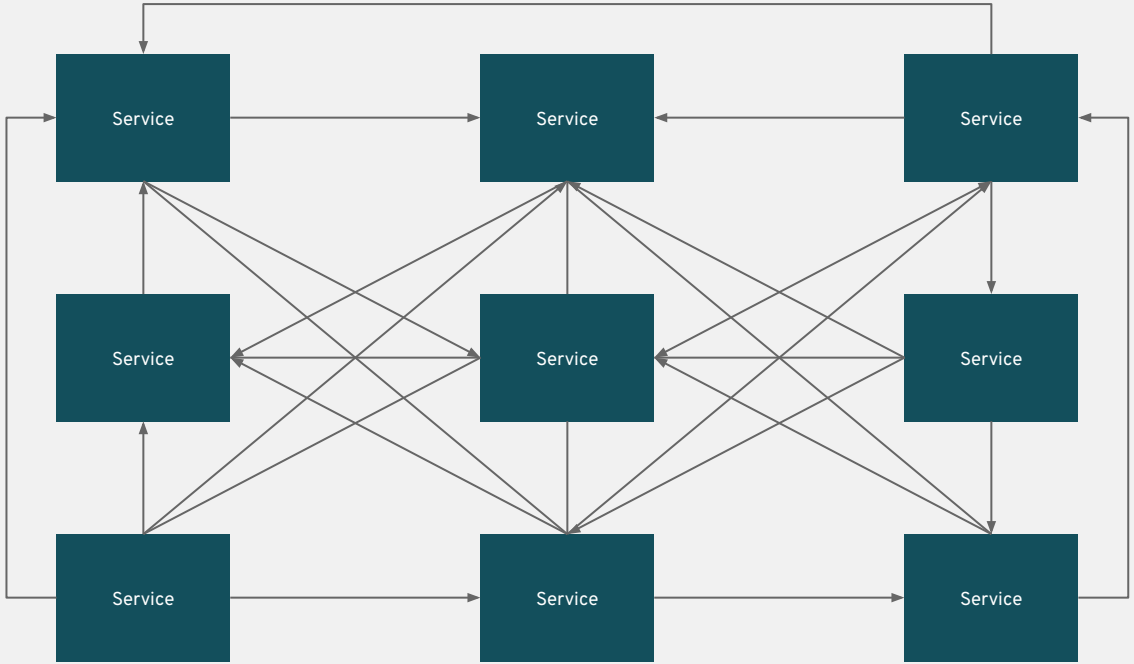


~~MICROSERVICES~~ ARCHITECTURE

DISTRIBUTED



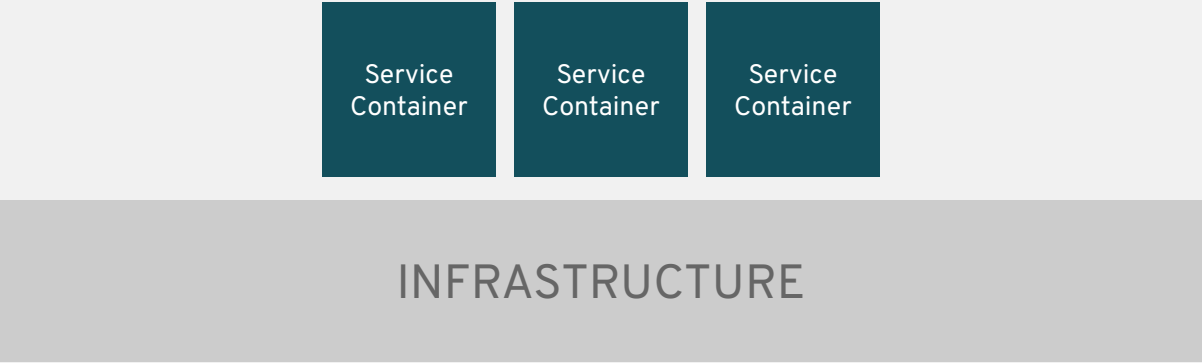
DISTRIBUTED ARCHITECTURE



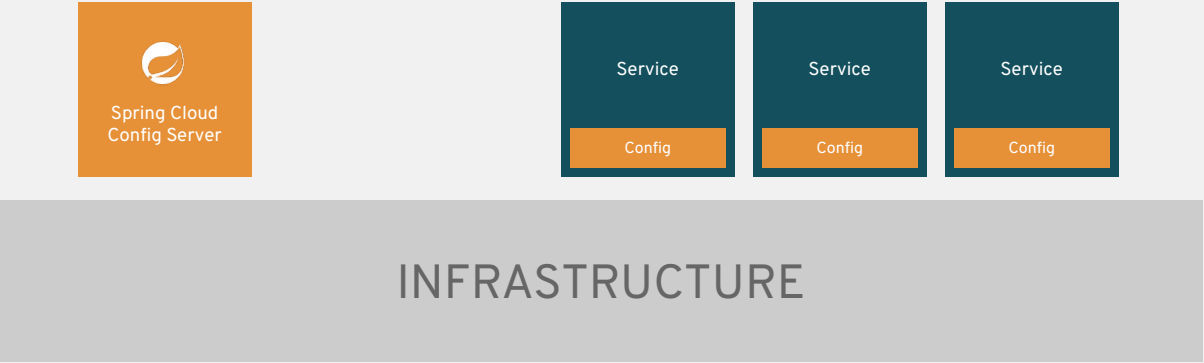
HOW TO DEAL WITH THE COMPLEXITY?

Photo by Clint Adair on Unsplash

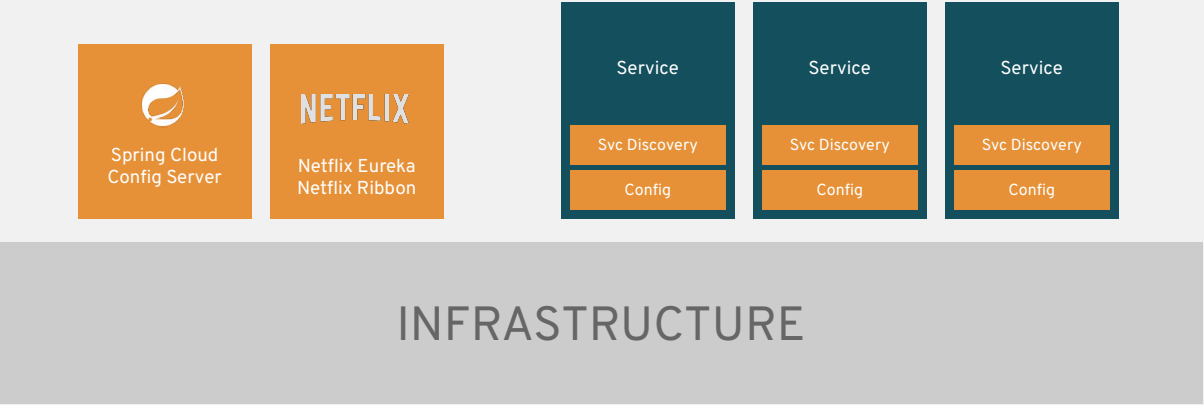
DEPLOYMENT


Service Container
Service Container
Service Container
INFRASTRUCTURE

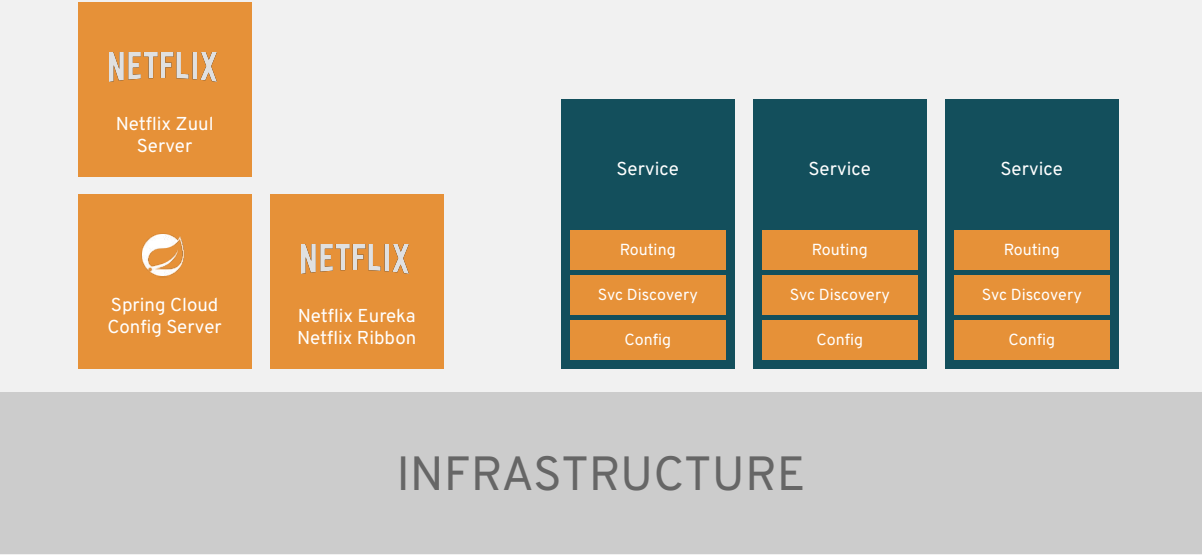
CONFIGURATION



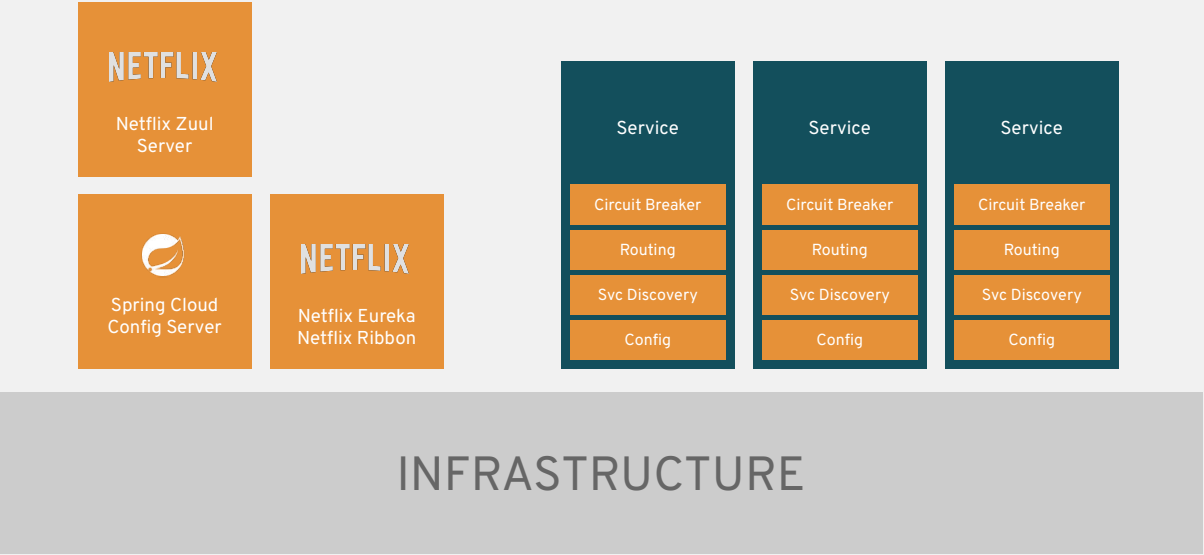
SERVICE DISCOVERY



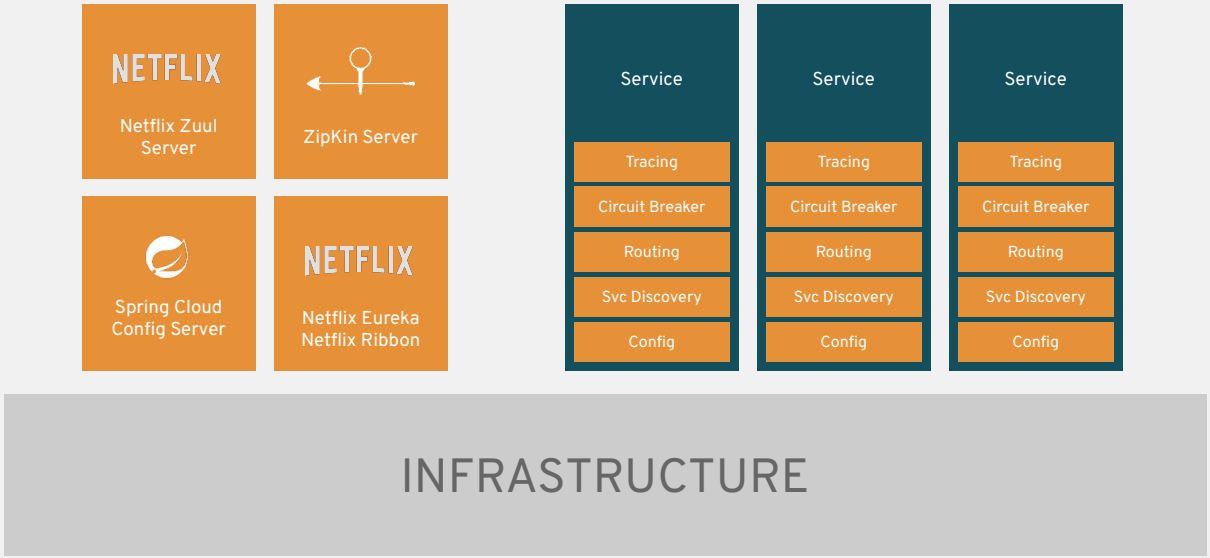
DYNAMIC ROUTING



FAULT TOLERANCE



TRACING AND VISIBILITY



WHAT ABOUT…?

POLYGLOT APPS

EXISTING APPS

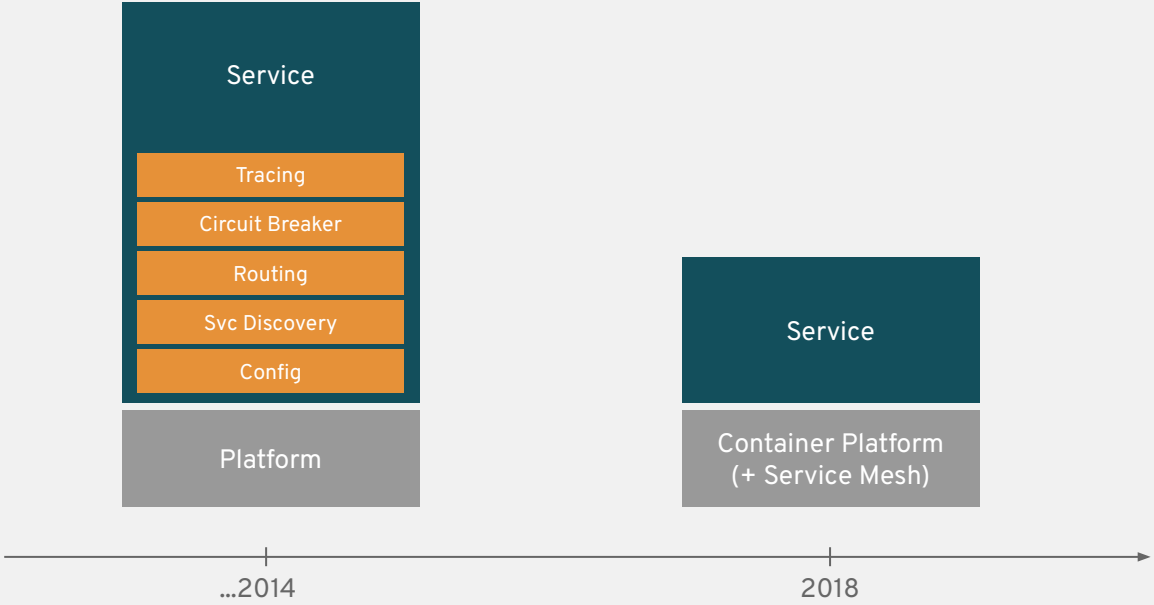
THERE SHOULD BE A BETTER WAY

ADDRESS THE COMPLEXITY IN THE INFRASTRUCTURE

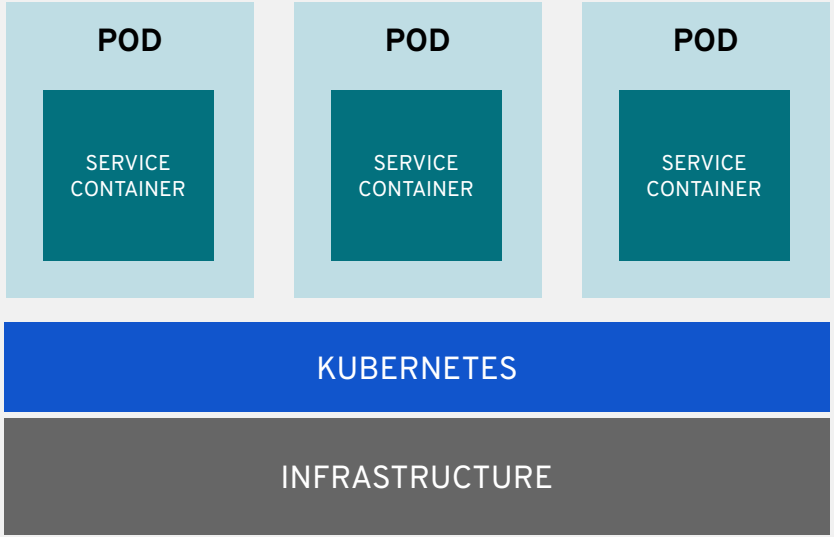
SERVICE MESH

A dedicated infrastructure layer for service-to-service communications

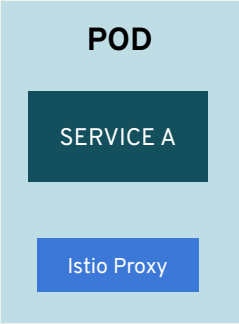
MICROSERVICES EVOLUTION



AUTOMATING CONTAINER DEPLOYMENT



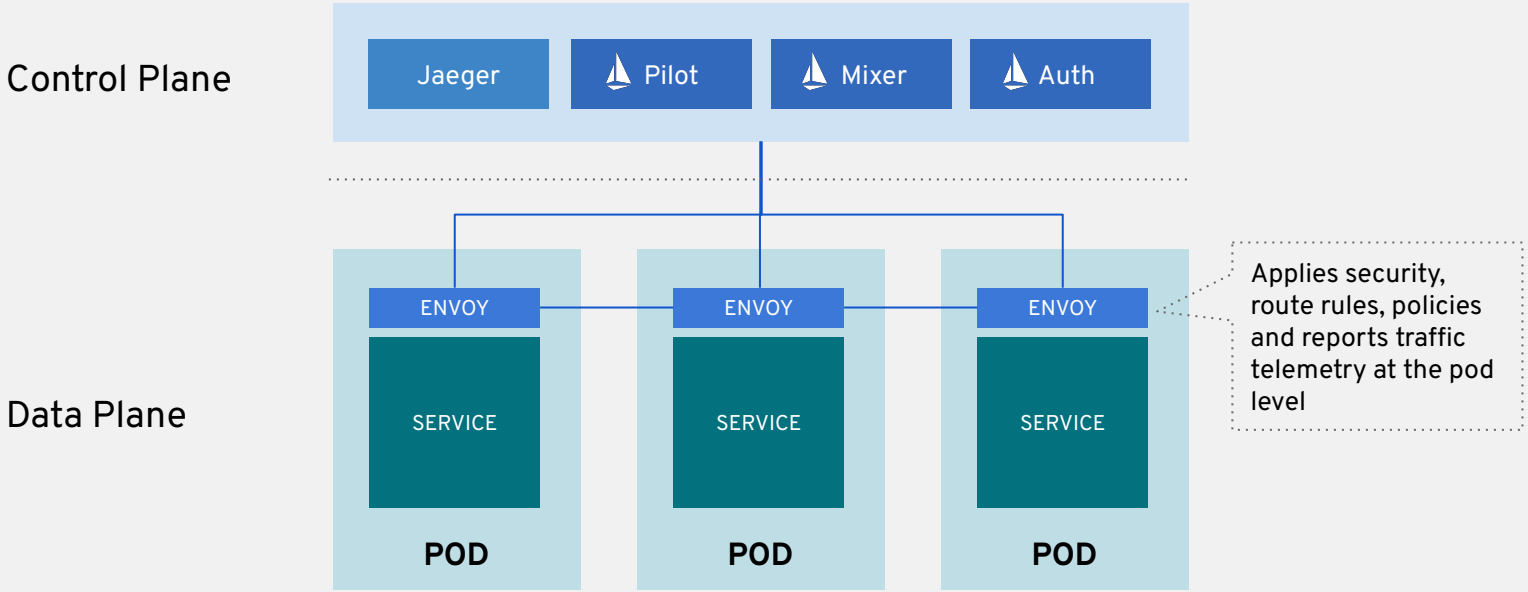
SIDECARS



- Two or more containers deployed to same pod
- Share
  - Same
    - Namespace
    - Pod IP
  - Shared lifecycle
- Used to enhance the co-located containers
- Istio Proxy (L7 Proxy)
  - Proxy all network traffic in and out of the app container

Source: http://blog.kubernetes.io/2015/06/the-distributed-system-toolkit-patterns.html

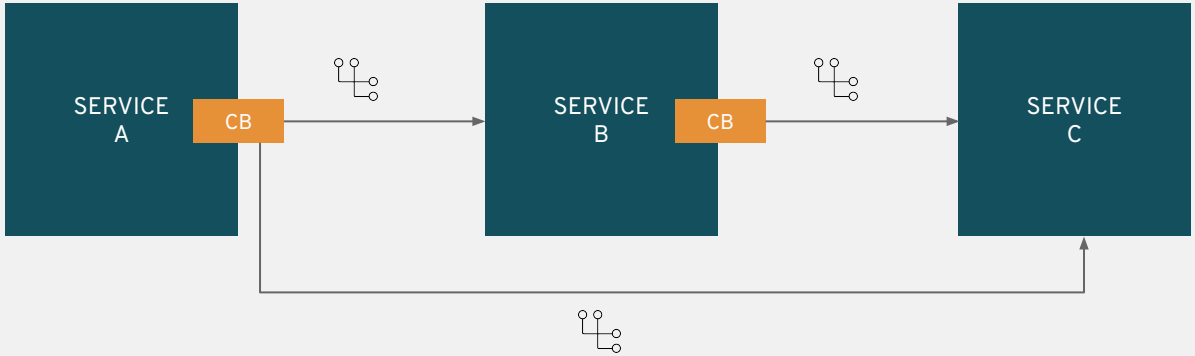
SERVICE MESH ARCHITECTURE



MAJOR FUNCTIONALITY

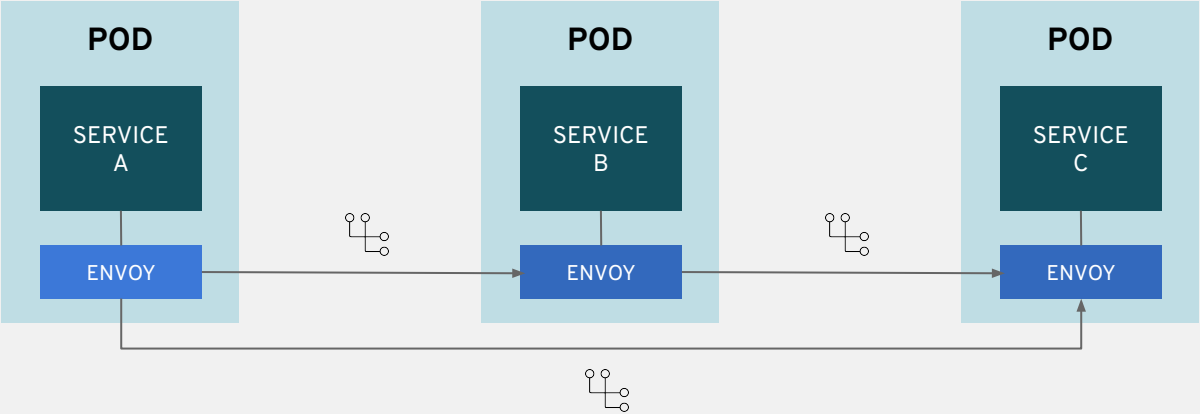
FAULT TOLERANCE

CIRCUIT BREAKERS WITHOUT ISTIO



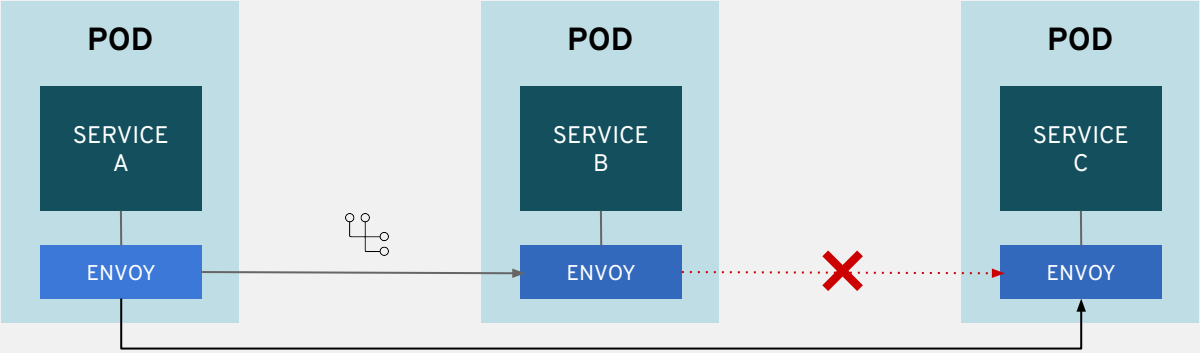
coupled to the service code

CIRCUIT BREAKERS WITH ISTIO



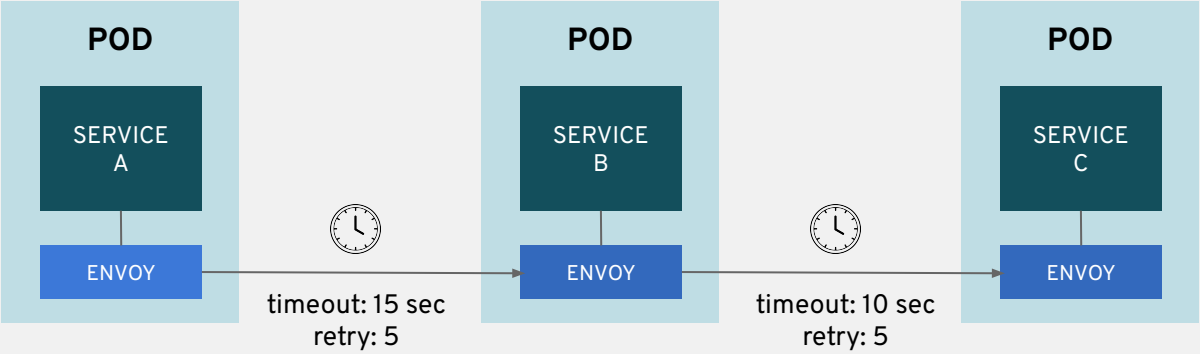
transparent to the services

CIRCUIT BREAKERS WITH ISTIO



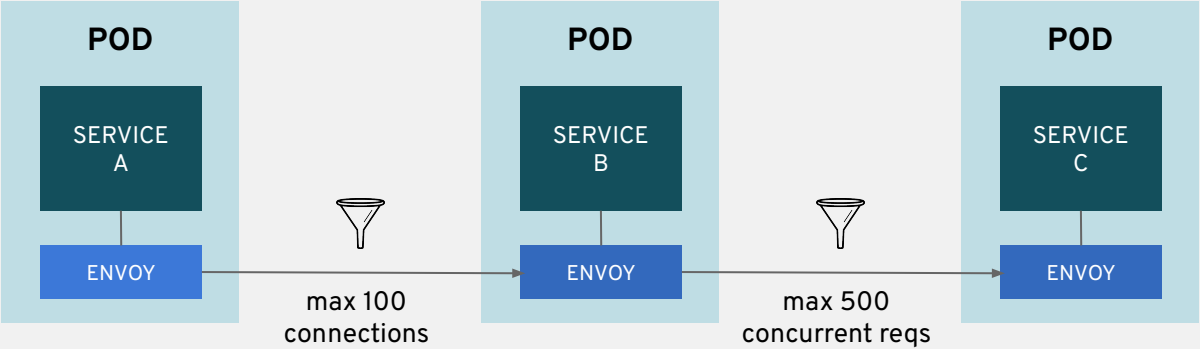
improved response time with global circuit status

TIMEOUTS AND RETRIES WITH ISTIO



configure timeouts and retries, transparent to the services

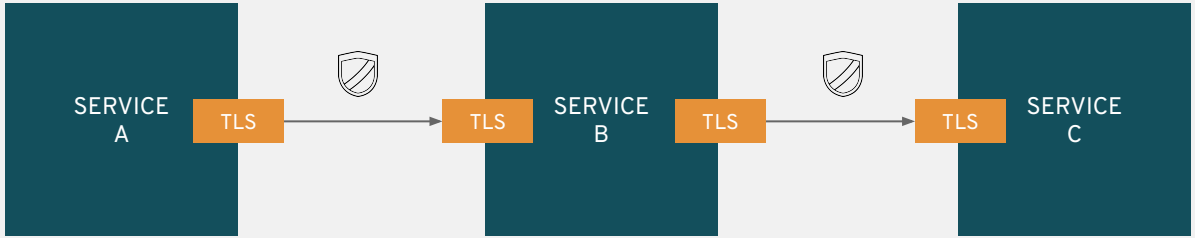
RATE LIMITING WITH ISTIO



limit invocation rates, transparent to the services

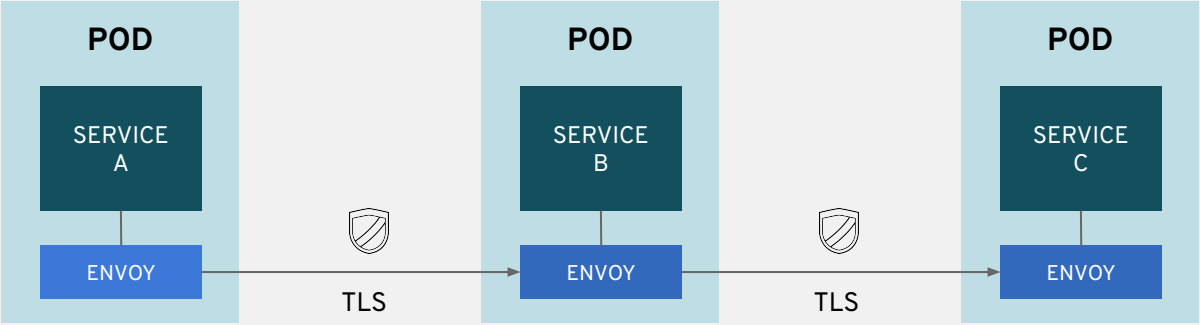
SERVICE SECURITY

SECURE COMMUNICATION WITHOUT ISTIO



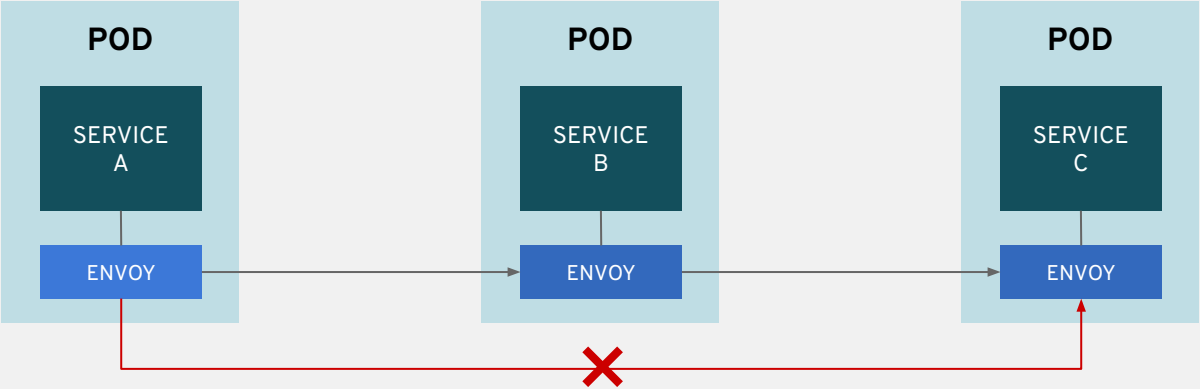
coupled to the service code

SECURE COMMUNICATION WITH ISTIO



mutual TLS authentication, transparent to the services

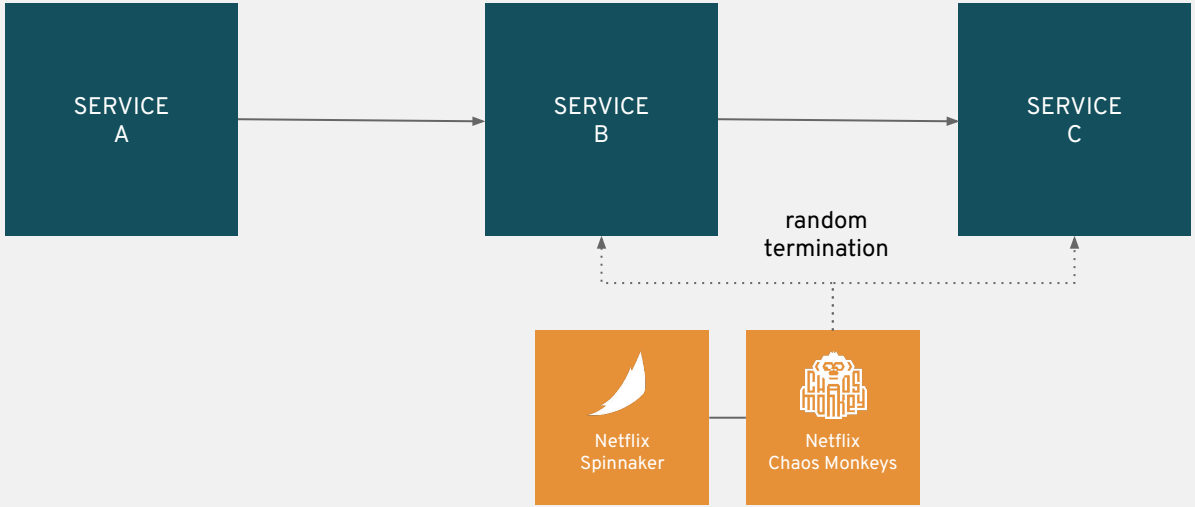
CONTROL SERVICE ACCESS WITH ISTIO



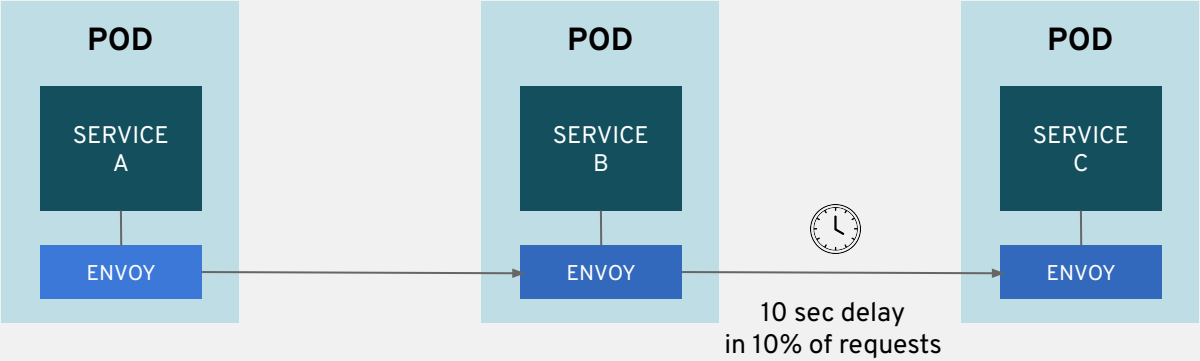
control the service access flow, transparent to the services

CHAOS ENGINEERING

CHAOS ENGINEERING WITHOUT ISTIO

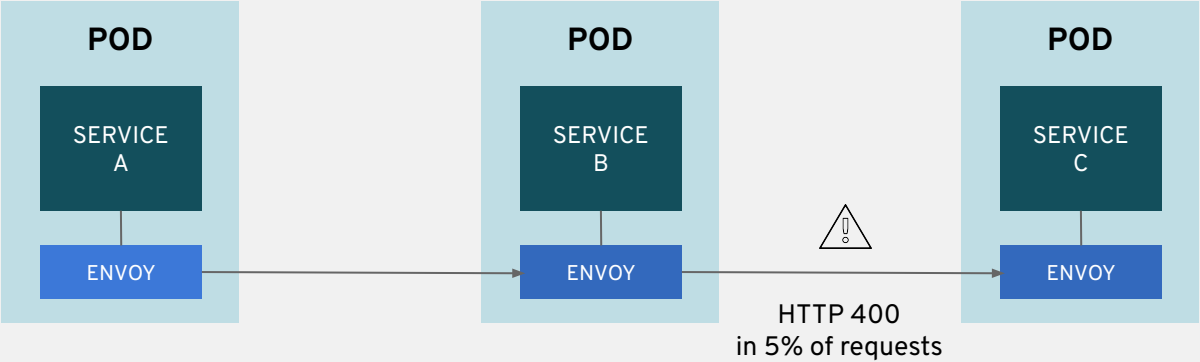


CHAOS ENGINEERING WITH ISTIO



inject delays, transparent to the services

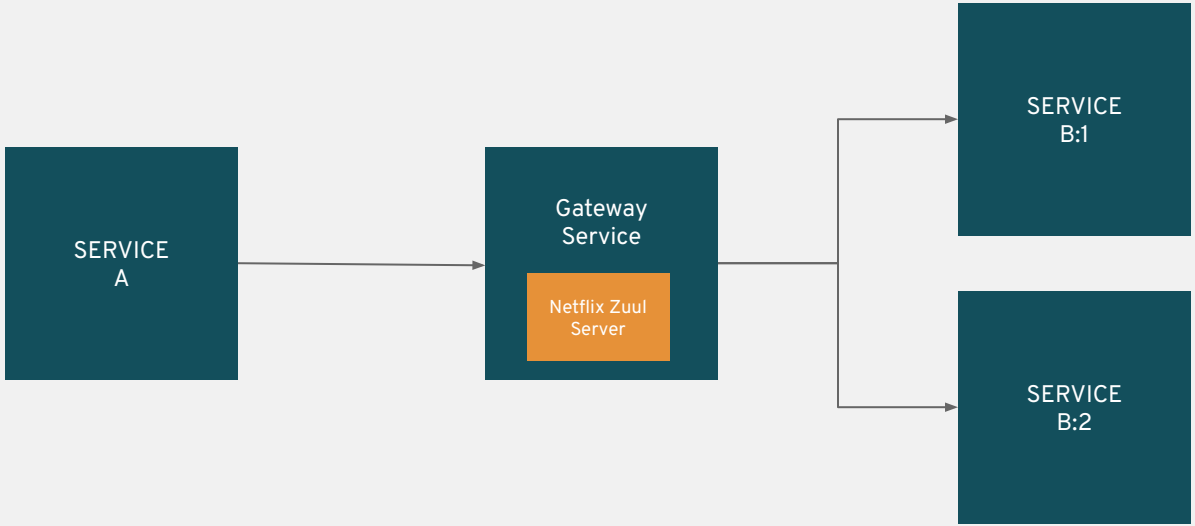
CHAOS ENGINEERING WITH ISTIO



inject protocol-specific errors, transparent to the services

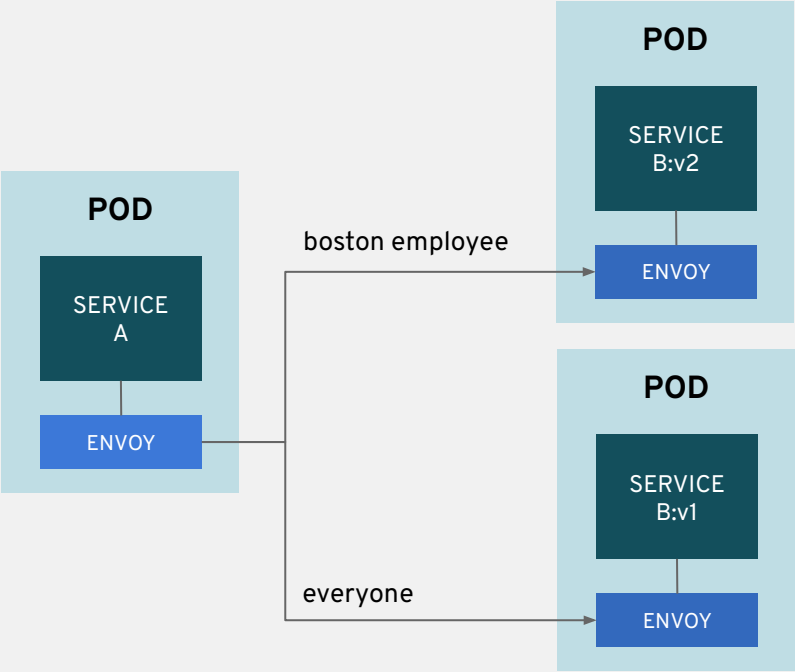
DYNAMIC ROUTING

DYNAMIC ROUTING WITHOUT ISTIO

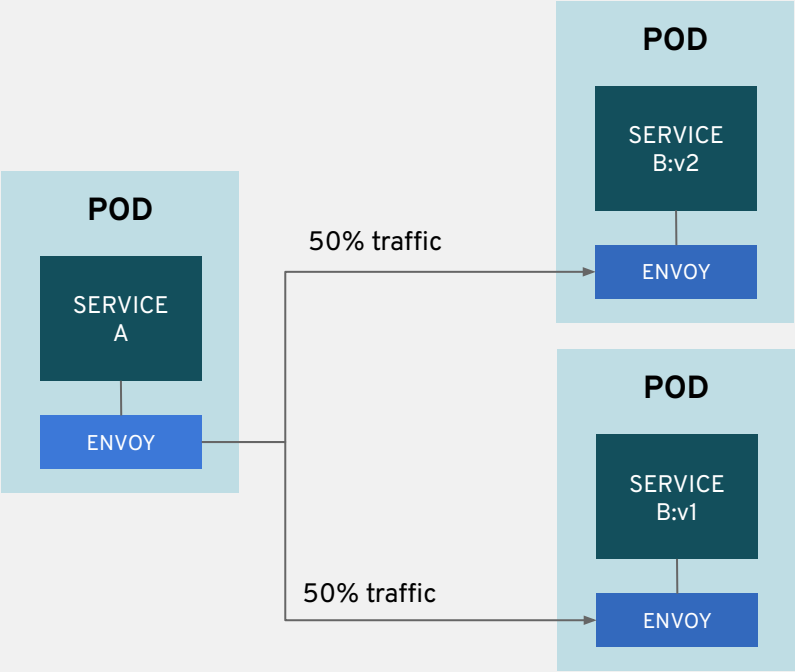


custom code to enable dynamic routing

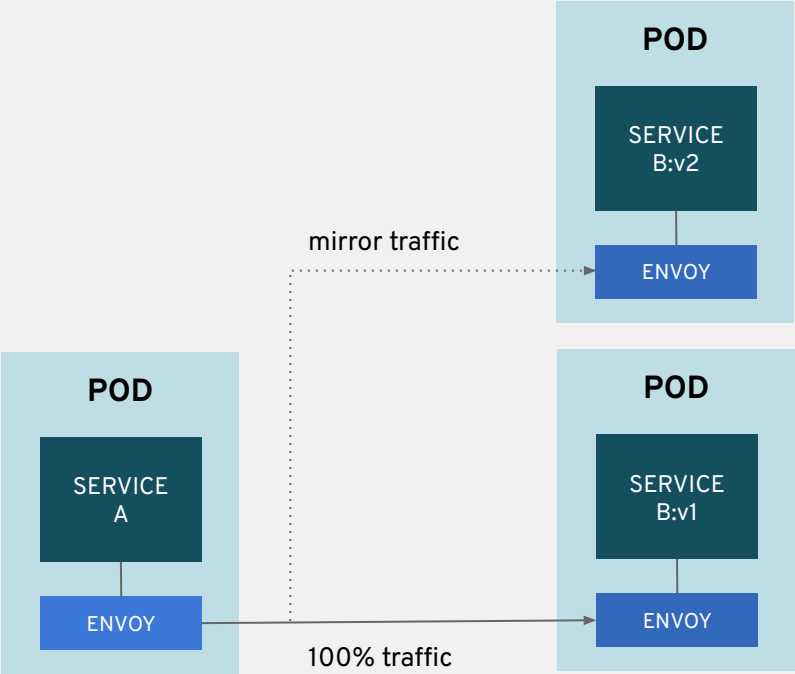
CANARY DEPLOYMENT WITH ISTIO



A/B DEPLOYMENT WITH ISTIO

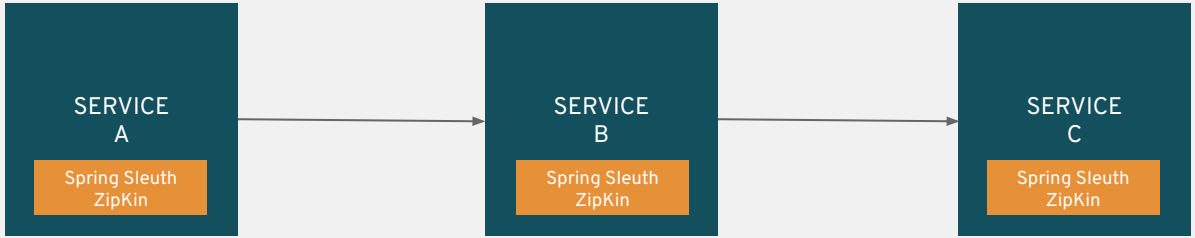


DARK LAUNCHES WITH ISTIO



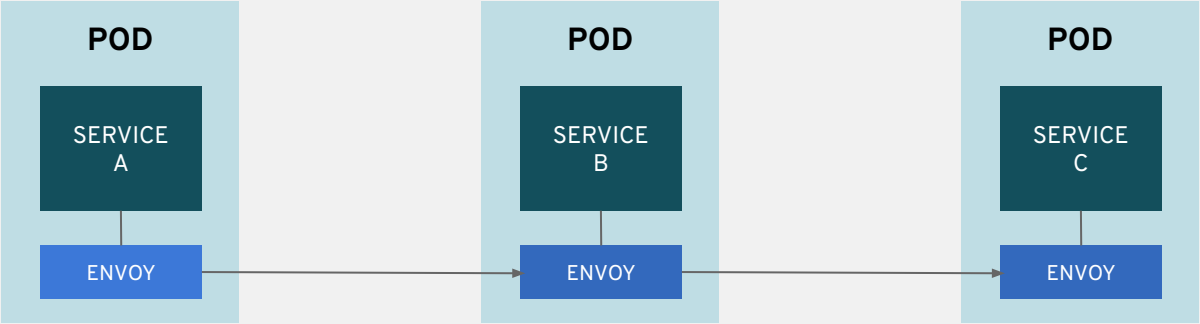
DISTRIBUTED TRACING (JAEGER)

DISTRIBUTED TRACING WITHOUT ISTIO



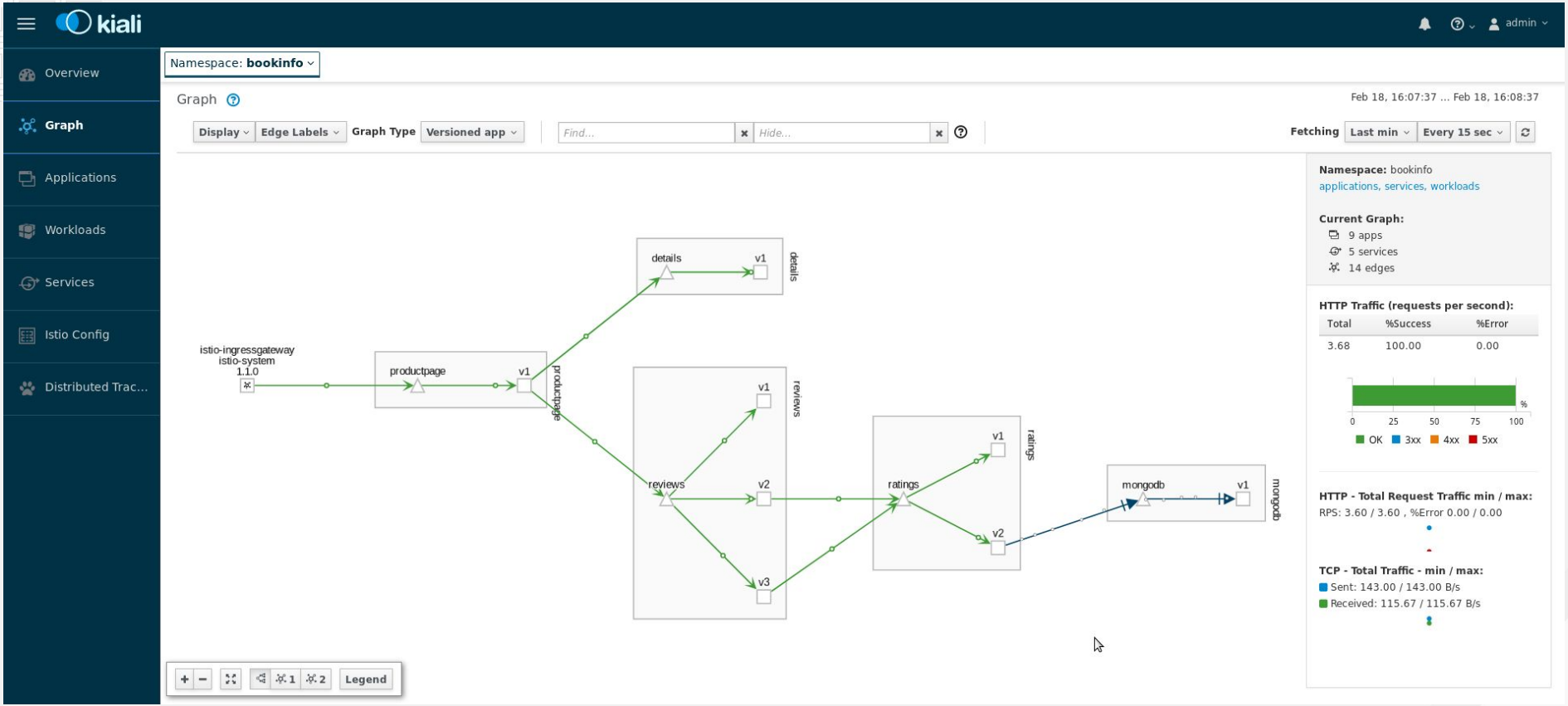
code to enable dynamic tracing

DISTRIBUTED TRACING WITH ISTIO & JAEGER

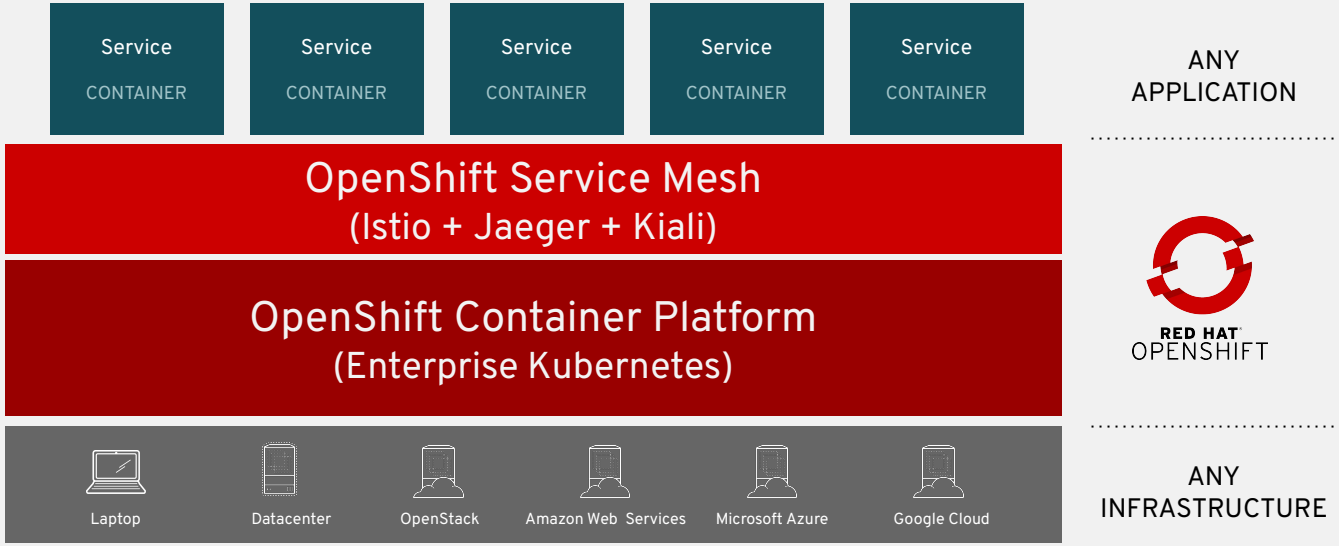


discovers service relationships and process times, transparent to the services

SERVICE MESH OBSERVABILITY (KIALI)


kiali
admin
Overview
Graph
Applications
Workloads
Services
Istio Config
Distributed Trac...
Namespace: bookinfo
Graph
Feb 18, 16:07:37 ... Feb 18, 16:08:37
Display
Edge Labels
Graph Type
Versioned app
Find...
Hide...
Fetching
Last min
Every 15 sec
istio-ingressgateway
istio-system
1.1.0
productpage
v1
details
v1
reviews
v1
v2
v3
ratings
v1
v2
mongodb
v1
Namespace: bookinfo
applications, services, workloads
Current Graph:
9 apps
5 services
14 edges
HTTP Traffic (requests per second):
Total
%Success
%Error
3.68
100.00
0.00
0
25
50
75
100
%
OK
3xx
4xx
5xx
HTTP - Total Request Traffic min / max:
RPS: 3.60 / 3.60 , %Error 0.00 / 0.00
TCP - Total Traffic - min / max:
Sent: 143.00 / 143.00 B/s
Received: 115.67 / 115.67 B/s
Legend

DISTRIBUTED SERVICES PLATFORM



Istio - Caveats

- Caveats
  - Be careful, istio is NOT a ServiceBus
  - Be careful, istio is NOT an API Manager
  - Be careful, istio is NOT meant to be a router
  - Be careful, istio is NOT a BPM tool

- Istio is meant to help solving some challenges
  - Centrally encryption / management of internal service communication
  - Distributed networking / communication of services
  - Increase Fault tolerance
  - Fault simulation

Thank you

linkedin.com/company/red-hat

facebook.com/redhatinc

youtube.com/user/RedHatVideos

twitter.com/RedHat

Red Hat