Kubernetes Workshop Series

JTC10
Kubernetes
ISTIO and knative

Niklaus Hirt
DevOps Architect / Cloud Architect
nikh@ch.ibm.com

Welcome to the

Kubernetes Workshop Series

Housekeeping

Meeting is being recorded to be shared on Social Media

Meeting Mute All: Unmute to speak

Breaks: every 60mins (interrupt me if I forget ;-)

Questions:

- In Slack # (not in Webex!)
- Adressed at the end of the Module
- Additional questions: unmute to speak

We will monitor the Slack channel during the Labs
→ Feel free to answer other participants questions

Who am I?

Niklaus Hirt

Passionate about tech for over 35 years

- High-school in Berne
- Degree in Computer Science at EPFL
- ELCA
- CAST
- IBM

✉ nikh@ch.ibm.com
@nhirt

Agenda – ISTIO

Module 0: Prepare the Labs

Module 1: Mesh Networking with ISTIO

Module 2: Mesh Networking Hands-On

Videos, sources and documentation will be available here:

All Workshop Recordings
https://www.youtube.com/channel/UCIS0jmGOQrG2AKKPkTJYj9w/videos

https://github.com/niklaushirt/k8s_training_public

https://github.com/niklaushirt/training

Session Quiz & Feedback

We will collect some feedback.

Please make sure you can access https://kahoot.it/
either on your PC of Phone.

You will get the Game PIN
later in the training.

Kubernetes Workshop Series

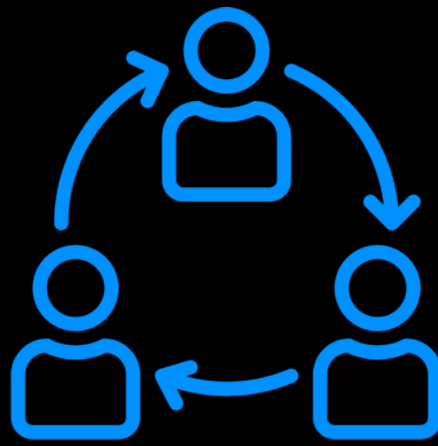
Prepare the Labs

00

Session Objectives

Attendees will run their own ***Personal Training Environment (PTE)*** in the VM.

Following some lectures will be ***hands-on*** work that each participant can to complete.



JTC90 Lab Setup

Task 1: Download Training VM

Task 2: Setup VMWare / VirtualBox

Task 3: Start Training VM

Task 4: Login / Check

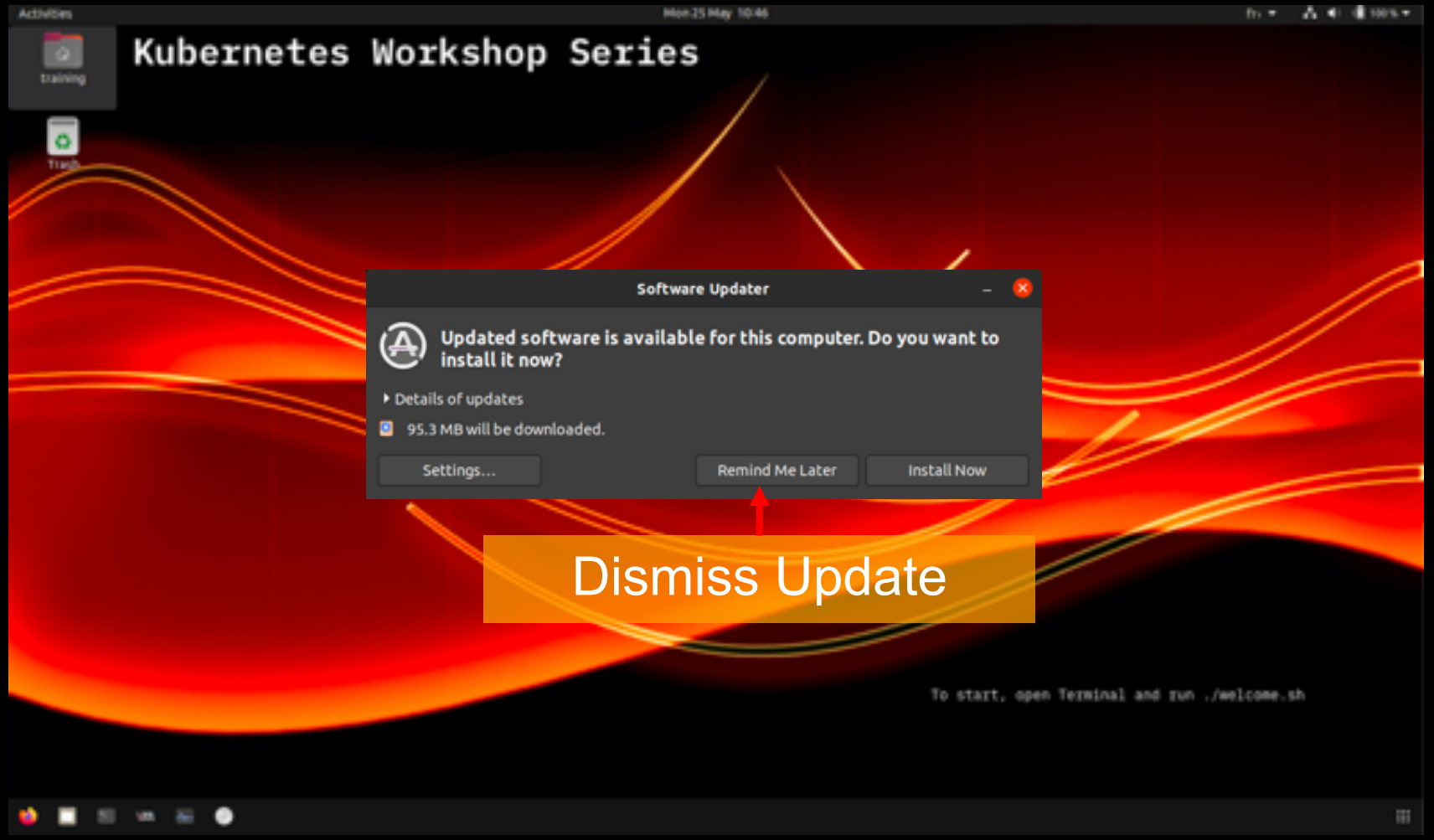
Accessing your Personal Training Environment

Temporary Personal Training Environment

http://pte-default.cp4mcp-demo-006-a376efc1170b9b8ace6422196c51e491-0000.eu-de.containers.appdomain.cloud/

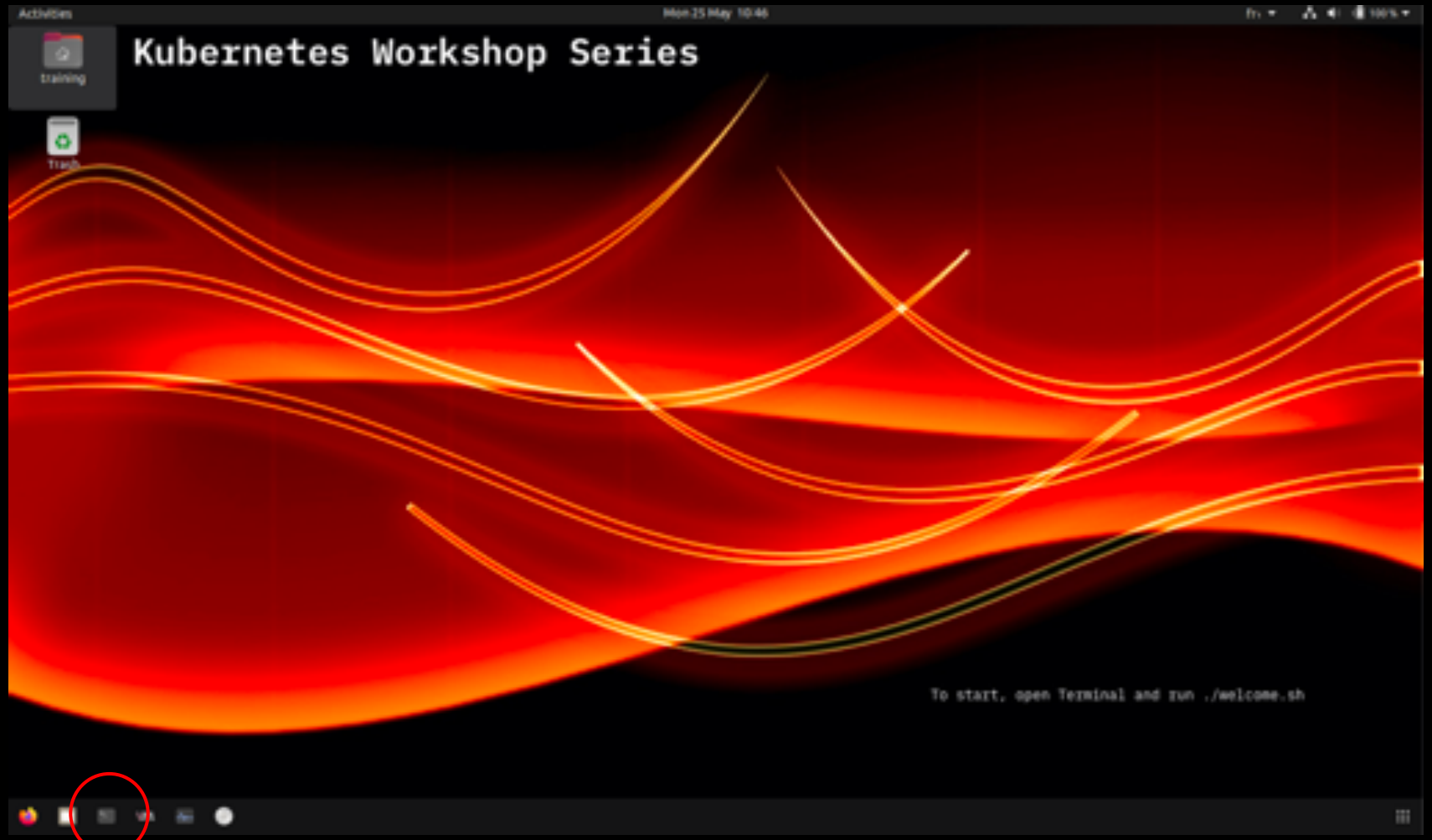
Pinned to the Slack channel

Accessing your Personal Training Environment



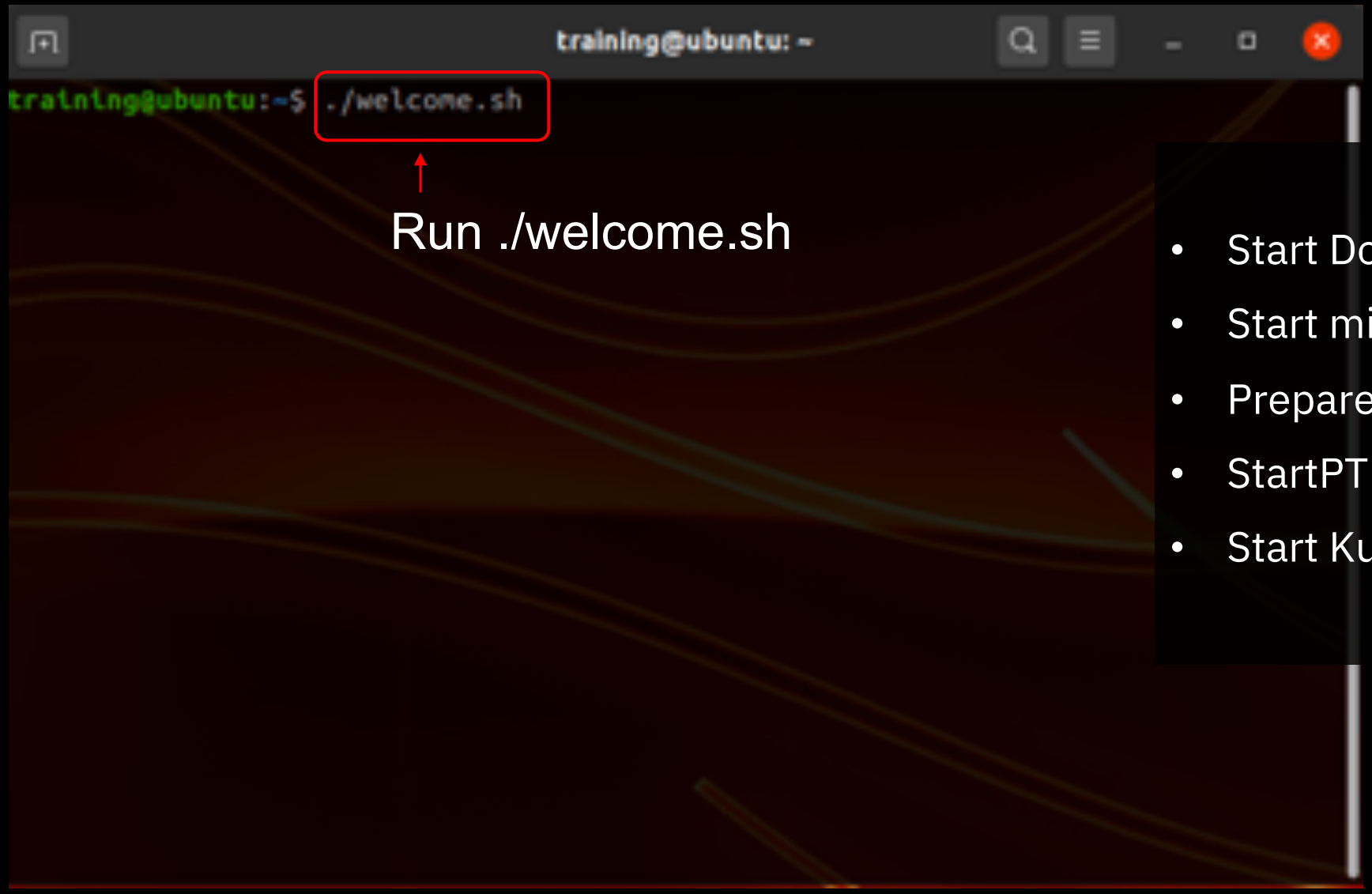
Accessing your Personal Training Environment

Accessing your Personal Training Environment



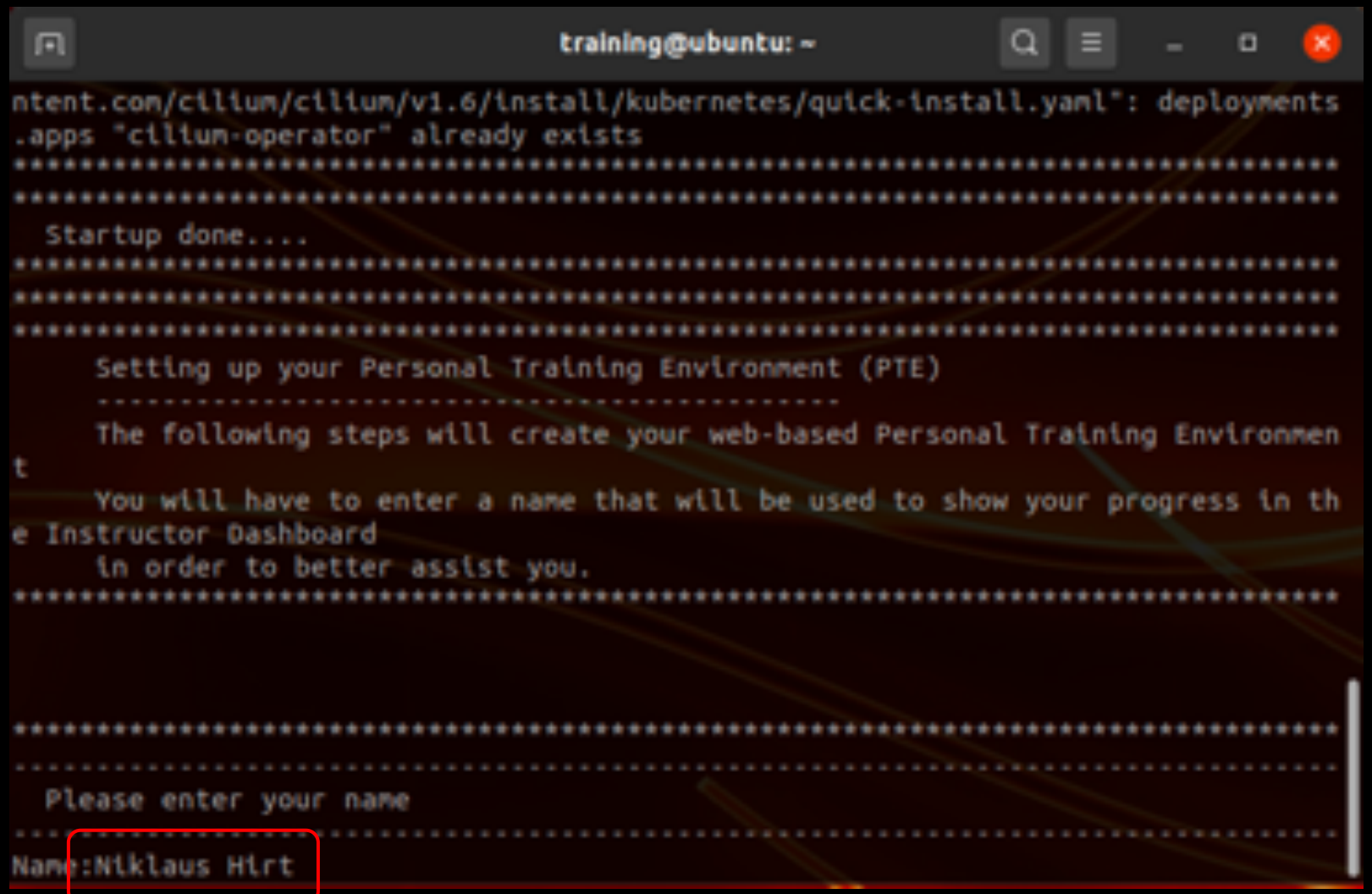
Start Terminal

Accessing your Personal Training Environment



- Start Docker
- Start minikube
- Prepares networking
- StartPTE
- Start Kubernetes Dashboard

Accessing your Personal Training Environment



Enter your name

Name will be used to show your progress in the Instructor Dashboard in order to better assist you

Accessing your Personal Training Environment

Troubleshooting

- If the startup script doesn't work you can run ./resetEnvironment.sh
  (this can take up to 30 minutes as it has to redownload all Docker images)

- If you lose your PTE Webpage just run `minikube service student-ui`

- Windows 10 problems can mostly be fixed by turning off Hyper-V by running (as admin)
  `bcdedit /set hypervisorlaunchtype off`
  and rebooting.
  This disables Hyper-V and allows Virtualbox to support nested virtualisation.

- You can turn it back on again with
  `bcdedit /set hypervisorlaunchtype auto`

Accessing your Personal Training Environment

Troubleshooting

I have added a standalone version to the Git repository for participants wishing to run the Labs directly on their PC.

This is **untested** and I cannot guarantee that all the Labs will be working 100%.

You must have the following setup on your PC:

- Minikube
- Docker
- Git

1. Clone the repository to your home directory
   ```
   git clone https://github.com/niklaushirt/training.git
   ```
2. Go to the installation directory
   ```
   cd ~/training/standalone
   ```
3. Run the preparation script
   ```
   ./welcome.sh
   ```

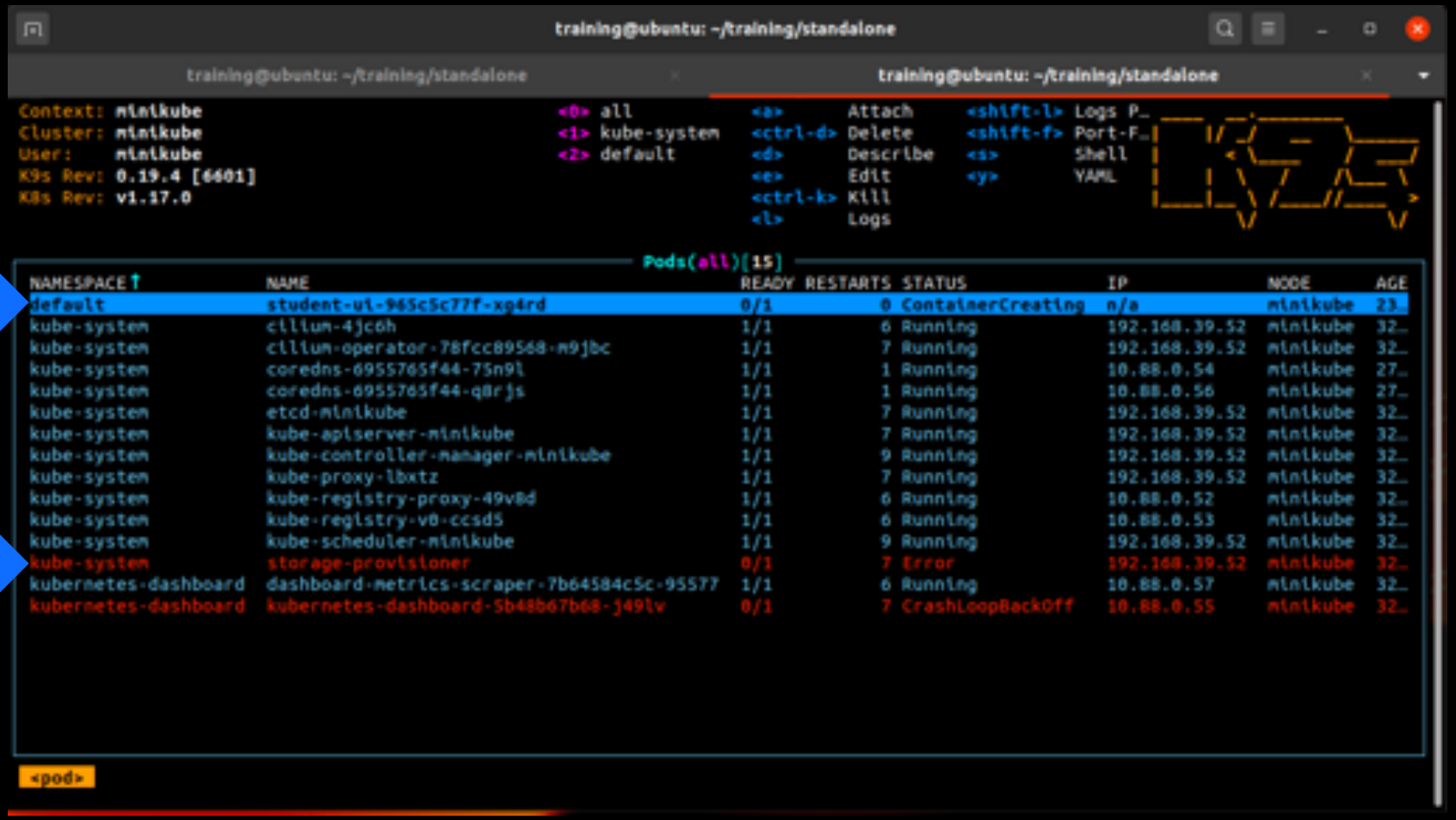
Accessing your Personal Training Environment

Troubleshooting

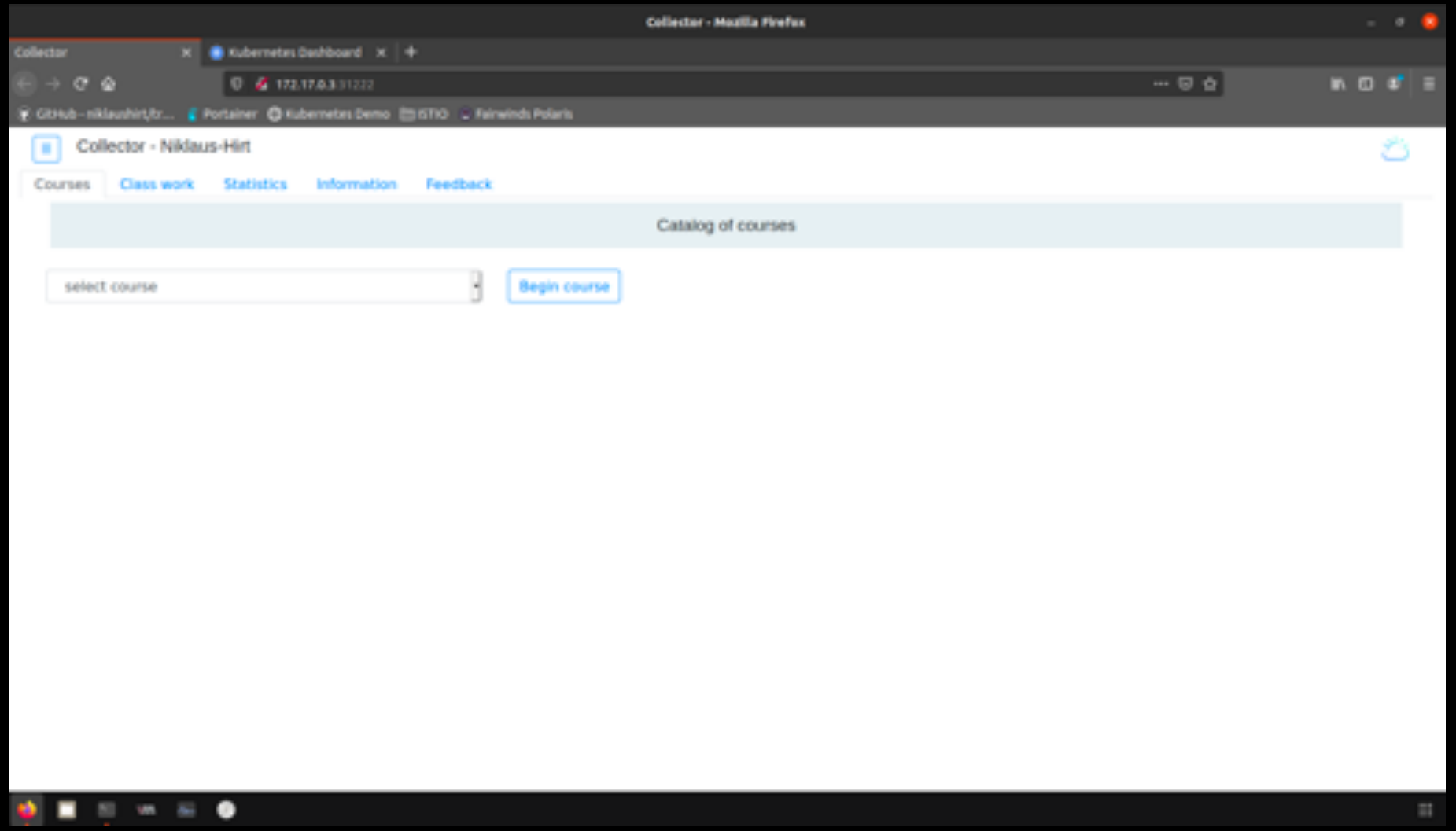
- Run ***k9s*** in the Terminal – wait for all the pods to be Running (blue – 1/1)

Image pulling - wait

Dependencies - wait



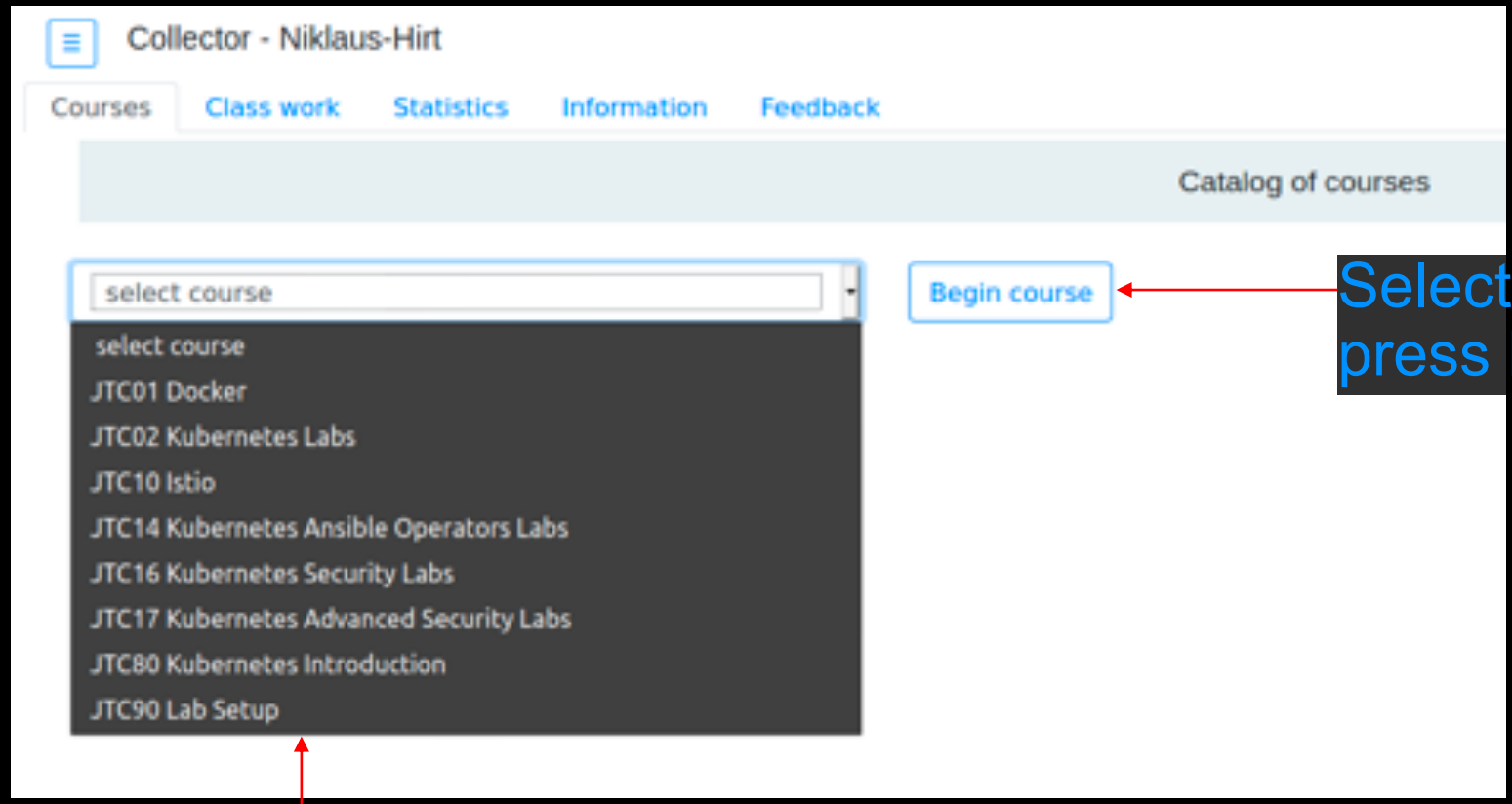
Accessing your Personal Training Environment



When completed, your PTE and Kubernetes Dashboard will open automatically

Accessing your Personal Training Environment

Name will be shown

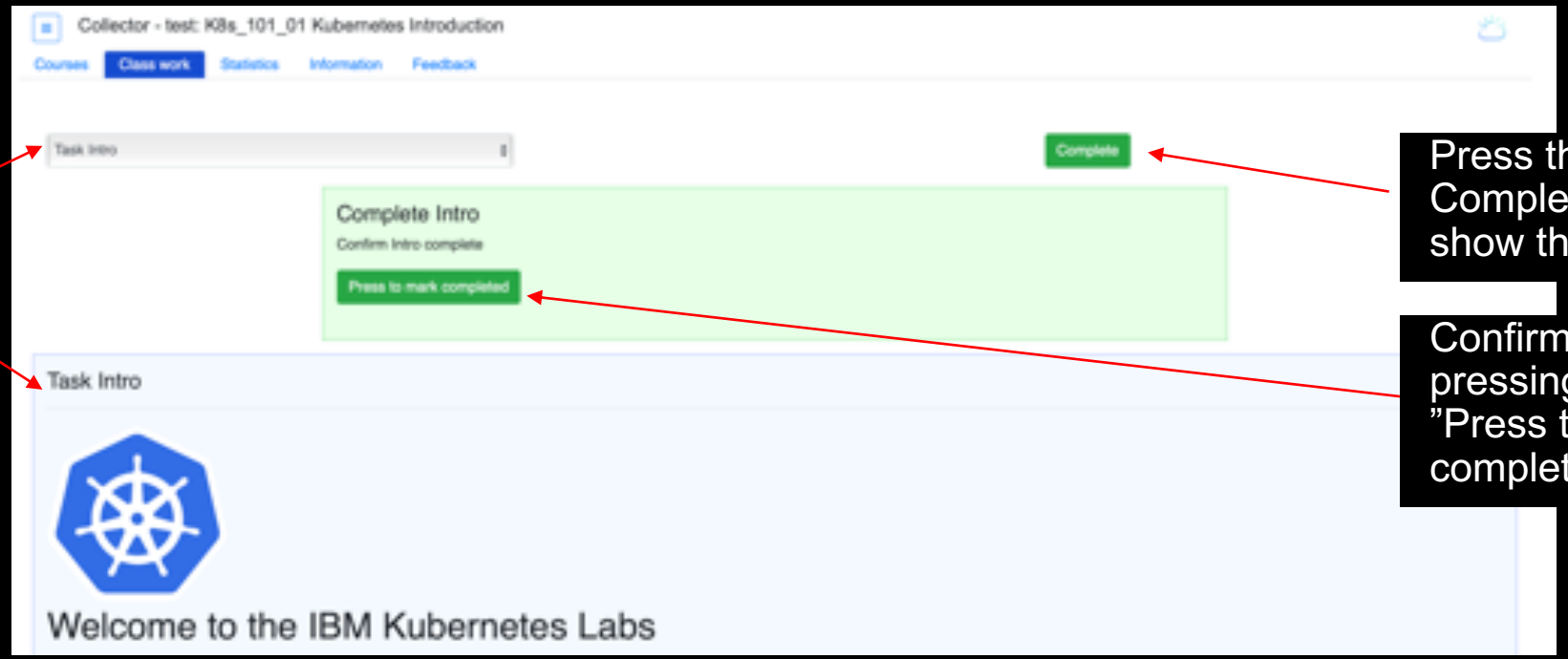


Select course and press button to begin

Current course catalog

Class Work

Select class work and the blue portion of the screen is shown



Press the green Complete button to show the green portion.

Confirm completion by pressing the green ”Press to mark completed” button.

! The Complete Button might not show instantly depending on the course settings

Following your progress

Course title

The number of items tracked will change based on the current course selected.

Green checkmark - item is completed
Red circle - item is waiting to be completed

QUESTIONS?

Kubernetes Workshop Series

Mesh Networking

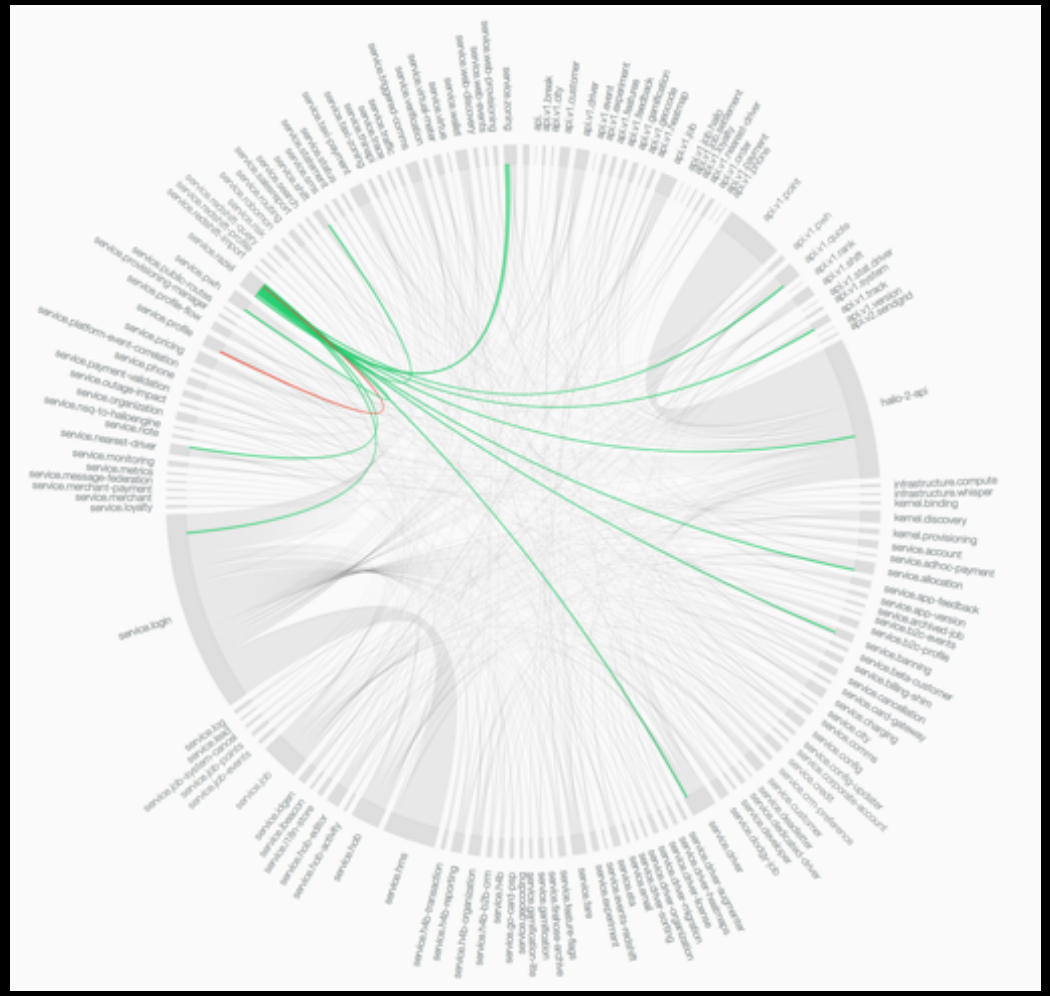
01

The trade off

Improved delivery velocity

in exchange for

increased operational complexity



Common DevOps Challenge 1

How do I **roll out** a newer version of my microservice **without down time**?

How do I **ensure traffic** continue to go to the current version before the newer version is tested and ready?

Common DevOps Challenge 2

How do I do **canary testing**?

How do I proceed to a **full rollout** after satisfactory testing of the new version?

Common DevOps Challenge 3

How do I do **A/B testing**?

- Release a new version to a subset of users in a precise way

I want to leverage crowdsourced testing. How do I **test** the new version **with a subset of users**?

I have **launched B in the dark**, but how can I keep B to myself or a small testing group?

Other common DevOps Challenges

4. Things don’t always go correctly in production… How do I **inject fault** to my microservices to prepare myself?

5. My services can only **handle certain rate**, how can I limit rate for some of my services?

Other common DevOps Challenges

6. I need to **view and monitor** what is going on with each of my services when crisis arises.

7. How can I **secure my services**.

Service Mesh

Dedicated infrastructure layer
to make
service-to-service communication
fast, safe and reliable

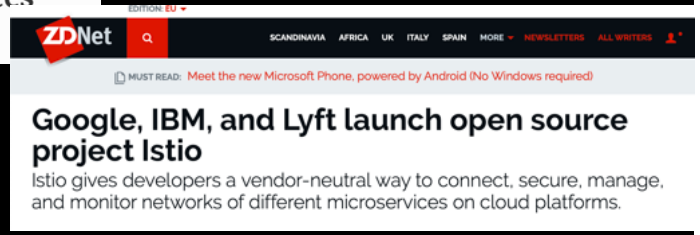
Istio

A service mesh designed to connect, manage and secure micro services


Forbes

3,859 views | May 25, 2017, 01:39am

Google, IBM And Lyft Want To Simplify Microservices Management With Istio

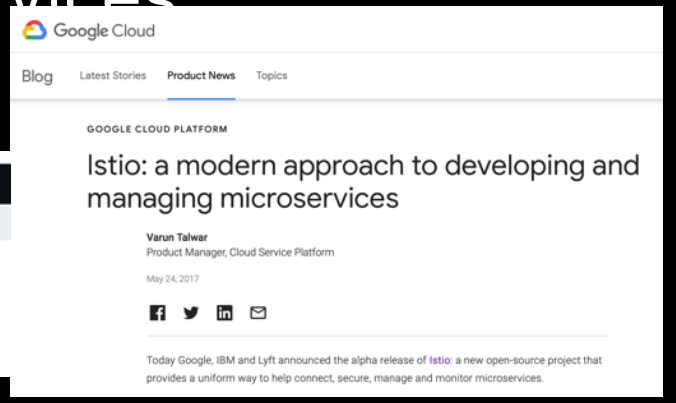

ZDNet

SCANDINAVIA AFRICA UK ITALY SPAIN MORE NEWSLETTERS ALL WRITERS

MUST READ: Meet the new Microsoft Phone, powered by Android (No Windows required)

Google, IBM, and Lyft launch open source project Istio

Istio gives developers a vendor-neutral way to connect, secure, manage, and monitor networks of different microservices on cloud platforms.


Google Cloud

Blog Latest Stories Product News Topics

GOOGLE CLOUD PLATFORM

Istio: a modern approach to developing and managing microservices

Varun Talwar
Product Manager, Cloud Service Platform

May 24, 2017

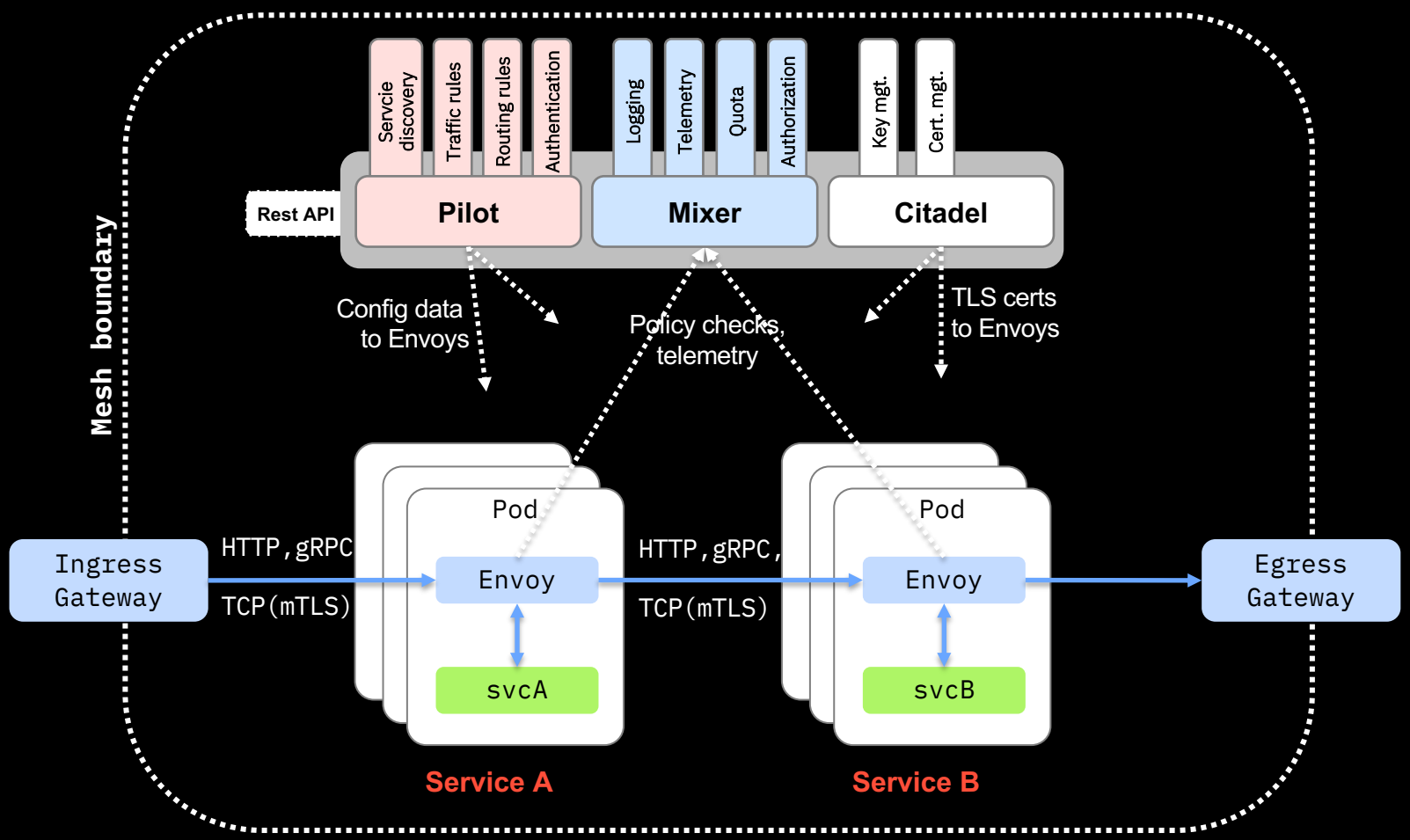
Today Google, IBM and Lyft announced the alpha release of Istio: a new open-source project that provides a uniform way to help connect, secure, manage and monitor microservices.

Launched in May 2017 by Google, Lyft and IBM

Open Source

Zero Code Changes

ISTIO - Architecture



Istio control plane traffic. Request routing rules, resilience configuration (circuit breakers, timeouts, retries), policies (ACLs, rate limits, auth), and metrics/reports from proxies.

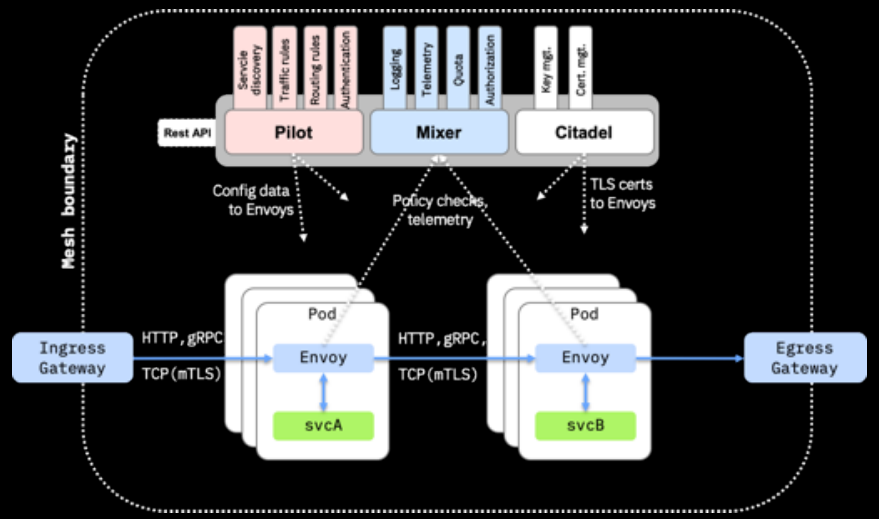
User/application traffic. HTTP/1.1, HTTP/2, gRPC, TCP with or without TLS

ISTIO - Architecture

Istio

Operates at **Layer 7**

- policies can be applied based on virtual host, URL, or other HTTP headers.
- Flexibility in processing.
- Allows it to be distributed



ISTIO
Gateway

Load balancer operating at the edge of the mesh receiving incoming HTTP/TCP connections

- Configures ports to expose externally
- Maps each exposed port to a request destination
- Each gateway can have one or more Virtual Services that defines these request destinations

Gateway is attached to Istio Ingress Controller (which can be the Kubernetes Ingress Controller)

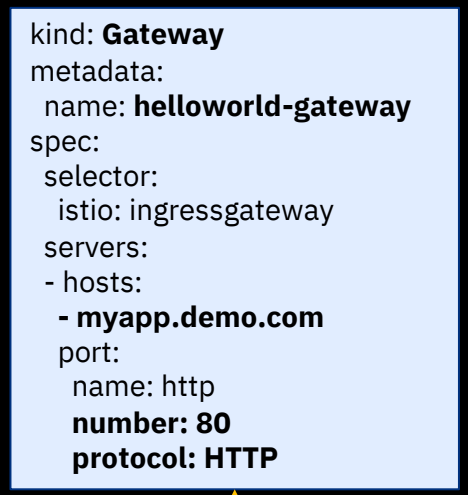
Request destinations

- Ports for the gateway to expose and hosts for the corresponding services
- Attributes: **servers**

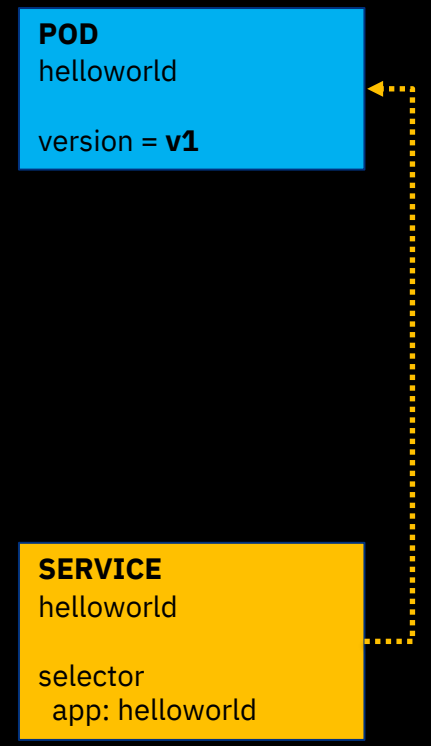
```yaml
kind: Gateway
metadata:
  name: helloworld-gateway
spec:
  selector:
    istio: ingressgateway
  servers:
  - hosts:
    - myapp.demo.com
    port:
      name: http
      number: 80
      protocol: HTTP
```

ISTIO

Custom resource definitions



http://myapp.demo.com



ISTIO
Virtual Service

Istio

Request sources
- Hosts that sources can invoke
- Attributes: **hosts** and **gateways**

Route destinations
- Subset of the destination
- Attributes: **route** and **destination**

Protocol selection
- How to connect to the destination subset
- Attributes: **http**, **tcp**, **tls**

Routing rules
- Additional routing attributes, applied for the route destinations
- Attributes: **weight** and **match**

HTTP traffic policy
- Protocol-specific connection quality of service
- Attributes: **timeout**, **retries**, **fault**, **rewrite**, and **redirect**

```
kind: VirtualService
metadata:
  name: helloworld
spec:
  hosts:
  - myapp.demo.com
  gateways:
  - helloworld-gateway
  http:
  - match:
    - uri:
        exact: /demo
    route:
    - destination:
        host: helloworld
```

ISTIO

Custom resource definitions

Ingress Configuration

```
kind: Gateway
metadata:
  name: helloworld-gateway
spec:
  selector:
    istio: ingressgateway
  servers:
  - hosts:
    - myapp.demo.com
    port:
      name: http
      number: 80
      protocol: HTTP
```

URL Routing

```
kind: VirtualService
metadata:
  name: helloworld
spec:
  hosts:
  - myapp.demo.com
  gateways:
  helloworld-gateway
  http:
  - match:
    - uri:
        exact: /demo
    route:
    - destination:
        host: helloworld
```

POD
helloworld

version = **v1**

SERVICE
helloworld

selector
app: helloworld

`helloworld.namespace.svc.cluster.local`

`http://myapp.demo.com/demo`

ISTIO
Destination Rule

Istio

Destination host
- Host that route destinations can select
- Attribute: **host**

Host subset
- Identify a subset of service endpoints
- Attribute: **labels**

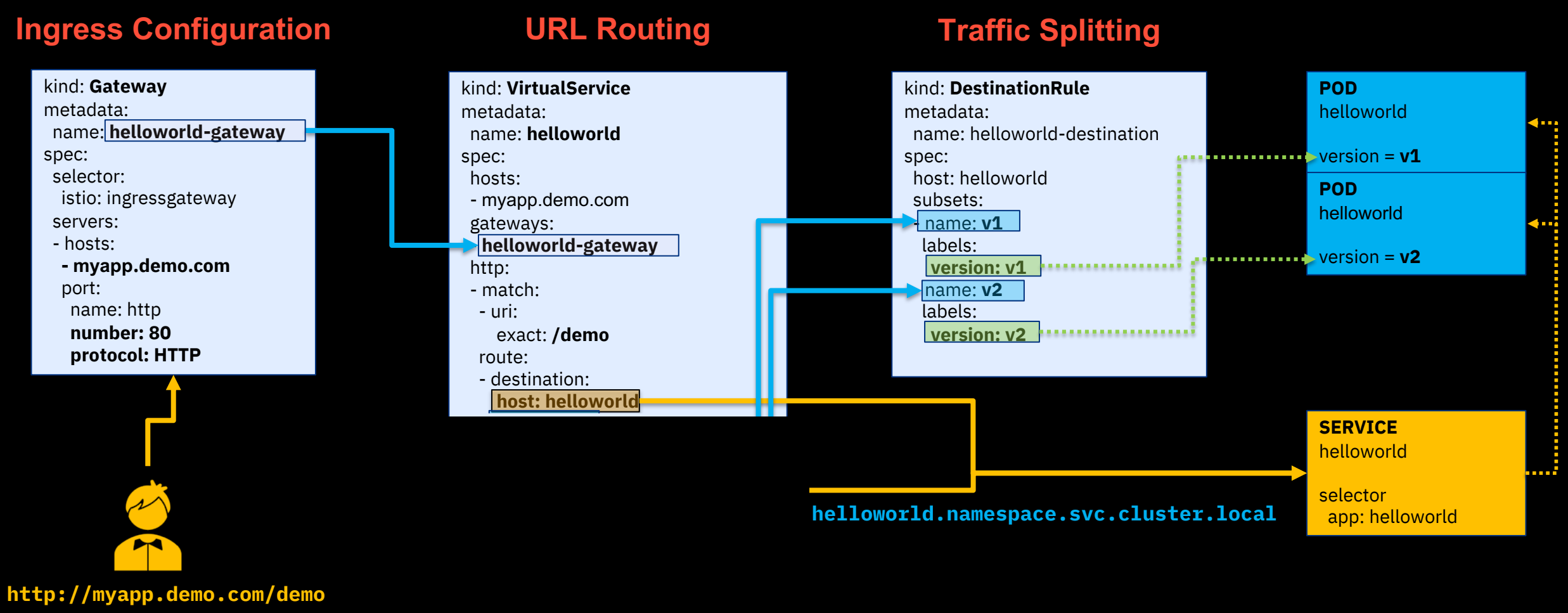
Traffic policy
- Influence expected quality of service for destinations
- Attributes: **trafficPolicy**
  loadBalancer,connectionPool, outlierDetection, and tls

Traffic policy can be applied to a port
- Attribute: **portLevelSettings**

```
kind: DestinationRule
metadata:
  name: helloworld-destination
spec:
  host: helloworld
  subsets:
  - name: v1
    labels:
      version: v1
  - name: v2
    labels:
      version: v2
  trafficPolicy:
    tls:
      mode: SIMPLE
    connectionPool:
      tcp:
        maxConnections: 100
```

ISTIO

Custom resource definitions



ISTIO
Virtual Service

Request sources
- Hosts that sources can invoke
- Attributes: **hosts** and **gateways**

Route destinations
- Subset of the destination
- Attributes: **route** and **destination**

Protocol selection
- How to connect to the destination subset
- Attributes: **http**, **tcp**, **tls**

Routing rules
- Additional routing attributes, applied for the route destinations
- Attributes: **weight** and **match**

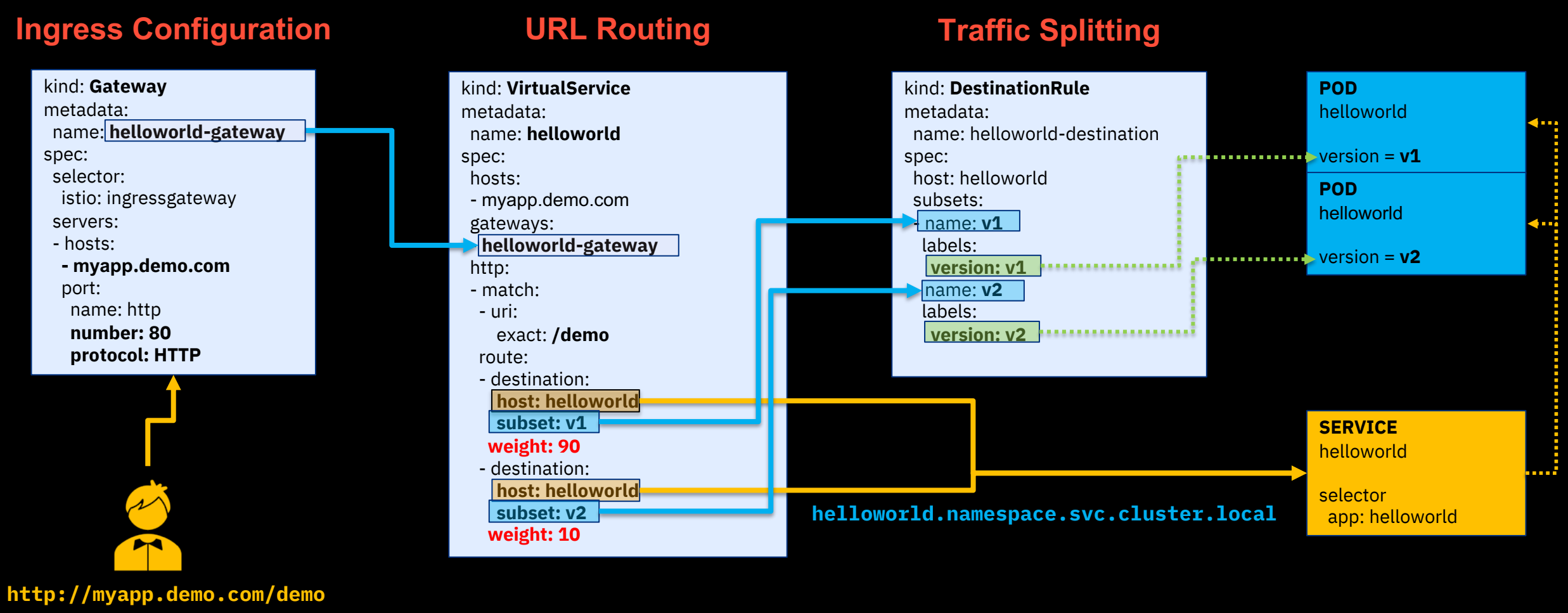
HTTP traffic policy
- Protocol-specific connection quality of service
- Attributes: **timeout**, **retries**, **fault**, **rewrite**, and **redirect**

Istio

```
kind: VirtualService
metadata:
  name: helloworld
spec:
  hosts:
  - myapp.demo.com
  gateways:
  - helloworld-gateway
  http:
  - match:
    - uri:
        exact: /demo
    route:
    - destination:
        host: helloworld
        subset: v1
      weight: 90
    - destination:
        host: helloworld
        subset: v2
      weight: 10
```

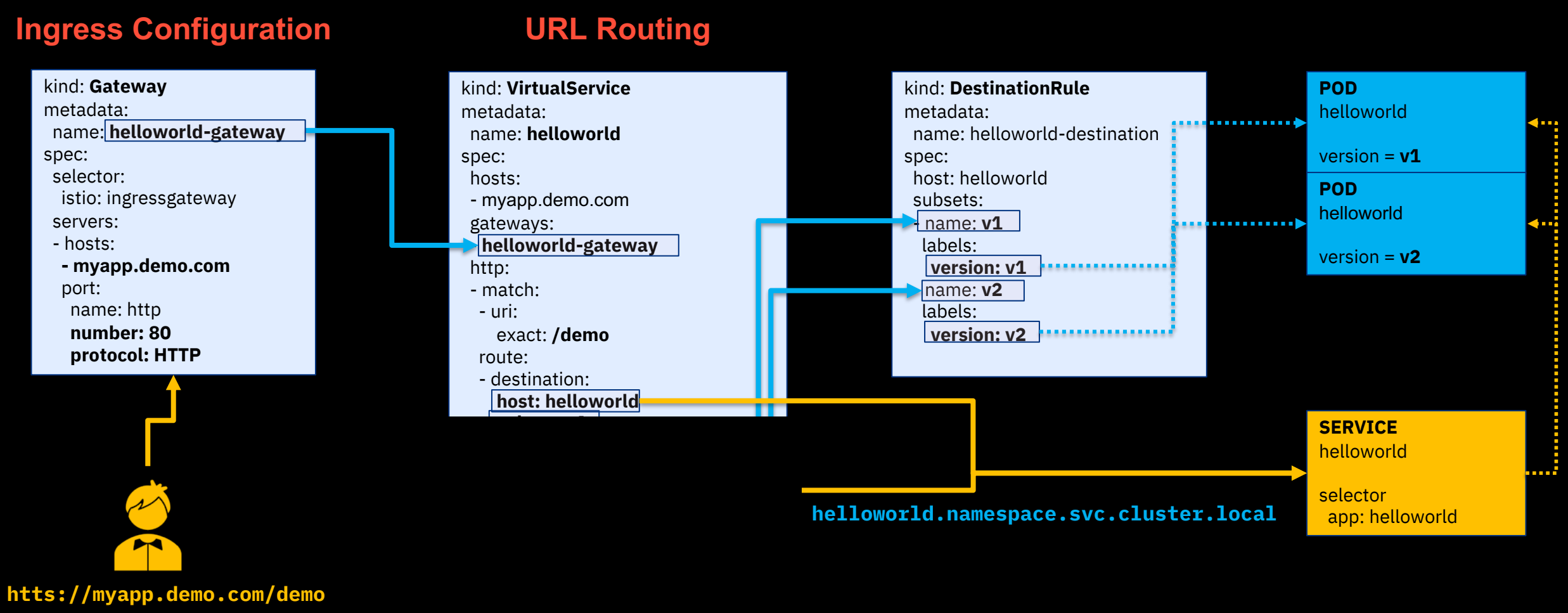
ISTIO

Custom resource definitions



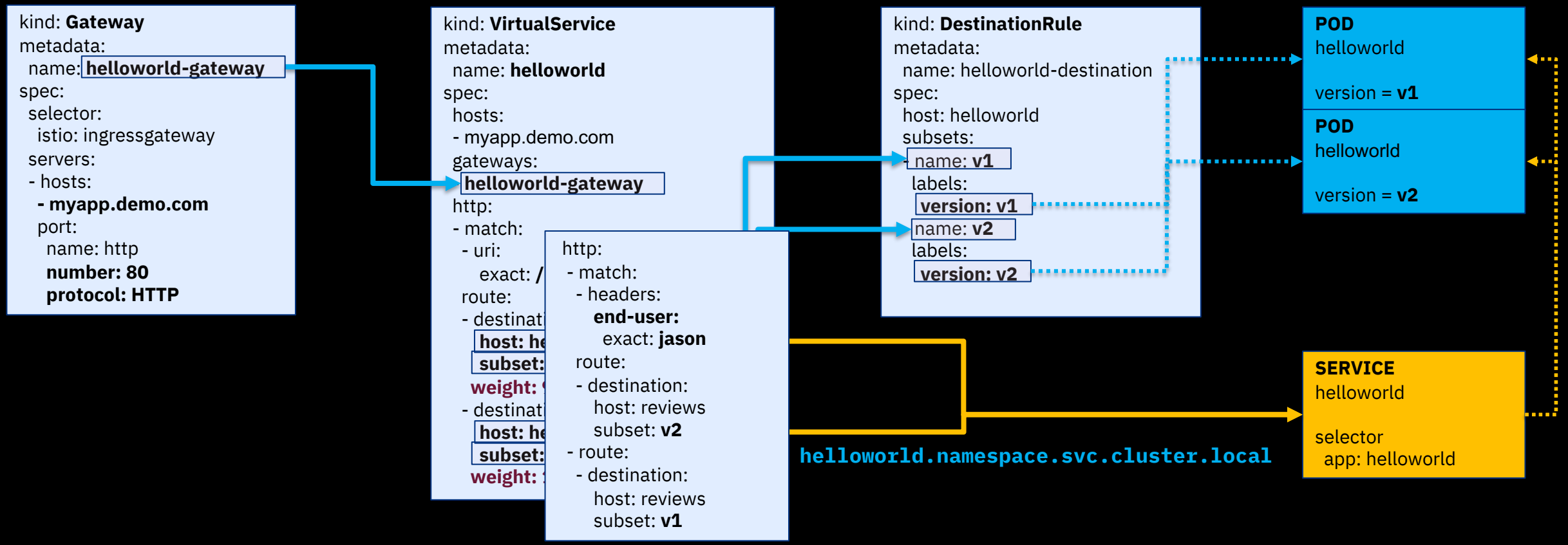
ISTIO

Custom resource definitions



ISTIO

Custom resource definitions



ISTIO
Sidecar injection

Manual Injection

```
kubectl apply -f <(istioctl kube-inject -f myapp.yaml)
```

Automatic Injection

For automatic sidecar injection, Istio relies on Mutating Admission Webhooks.
Label the namespace where you are deploying the app with `istio-injection=enabled`

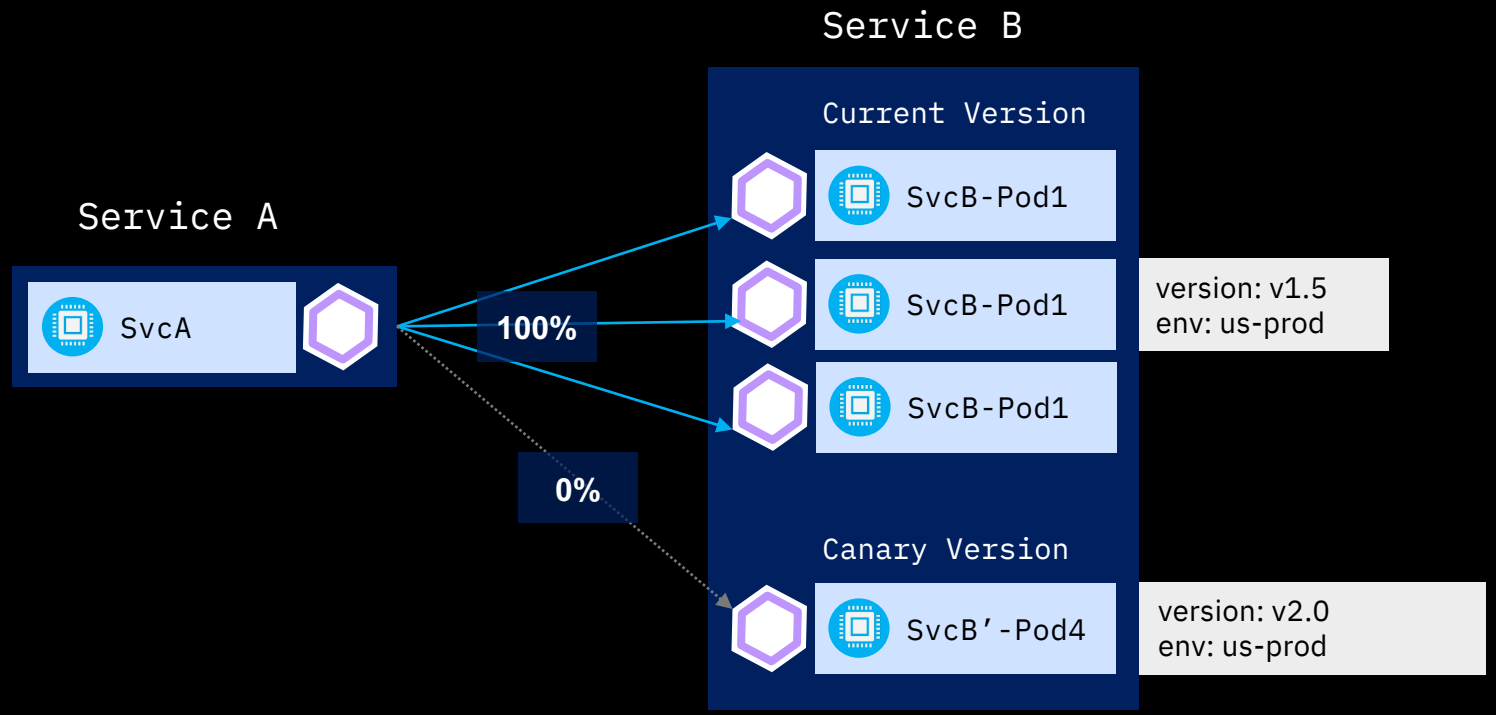
```
root@please1:~# kubectl get namespaces --show-labels
NAME                 STATUS        AGE   LABELS
default              Active        35d   istio-injection=enabled
dev-namespace        Active        35d   <none>
godemo               Terminating   16d   istio-injection=enabled
istio-system         Active        35d   icp=system
kube-public          Active        35d   <none>
kube-system          Active        35d   icp=system
platform             Active        35d   <none>
prod-namespace       Active        35d   <none>
services             Active        35d   <none>
test-namespace       Active        35d   <none>
```

```
kind: Deployment
metadata:
  name: no-injection
spec:
  template:
    metadata:
      annotations:
        sidecar.istio.io/inject: "false"
    spec:
      containers:
      - name: no-injection
        image: nginx
```

Addressing DevOps Challenges

#	CHALLENGE	ISTIO SOLUTION
CHALLENGE 1	ROLL OUT NEW VERSION WITHOUT DOWNTIME OR CHANGING CODE	TRAFFIC CONTROL
CHALLENGE 2	HOW TO DO CANARY TESTING	TRAFFIC SPLITTING
CHALLENGE 3	HOW TO DO A/B TESTING	TRAFFIC STEERING
CHALLENGE 4	THINGS DON'T ALWAYS GO CORRECTLY IN PRODUCTION…	TRAFFIC MIRRORING RESILIENCY RESILIENCY TESTING
CHALLENGE5	HOW CAN I LIMIT RATE FOR SOME OF MY SERVICES?	RATE LIMITING
CHALLENGE 6	I NEED TO VIEW AND MONITOR WHAT IS GOING ON WHEN CRISIS ARISES	TELEMETRY
CHALLENGE 7	HOW CAN I SECURE MY SERVICES?	AUTHENTICATION AUTHORIZATION CALICO

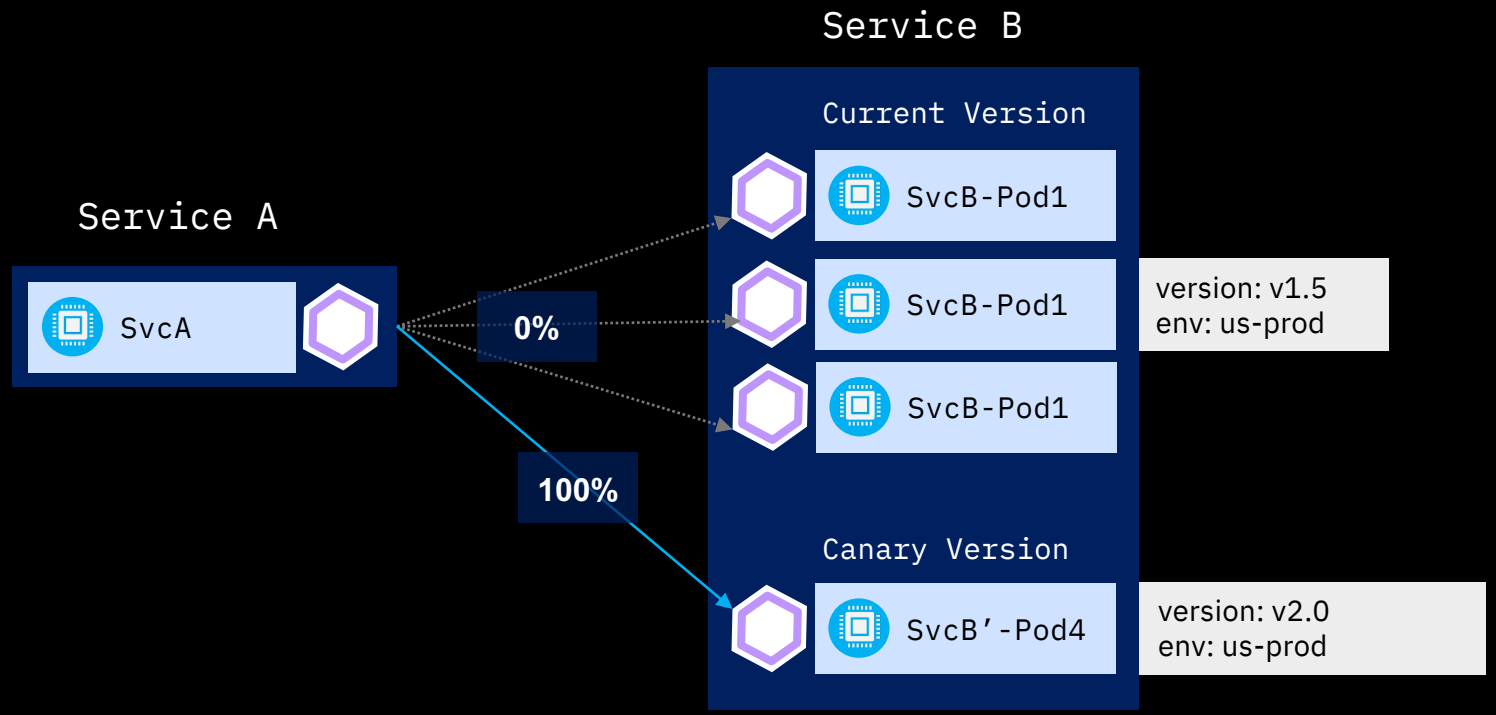
Traffic Control




```
kind: VirtualService
destination:
  serviceB.cluster.local
match:
  source:
    serviceA.cluster.local
route:
  - labels:
    version: v1.5
    env: us-prod
  weight: 100
```

CHALLENGE 1
ROLL OUT NEW VERSION WITHOUT DOWNTIME OR CHANGING CODE

Traffic Control

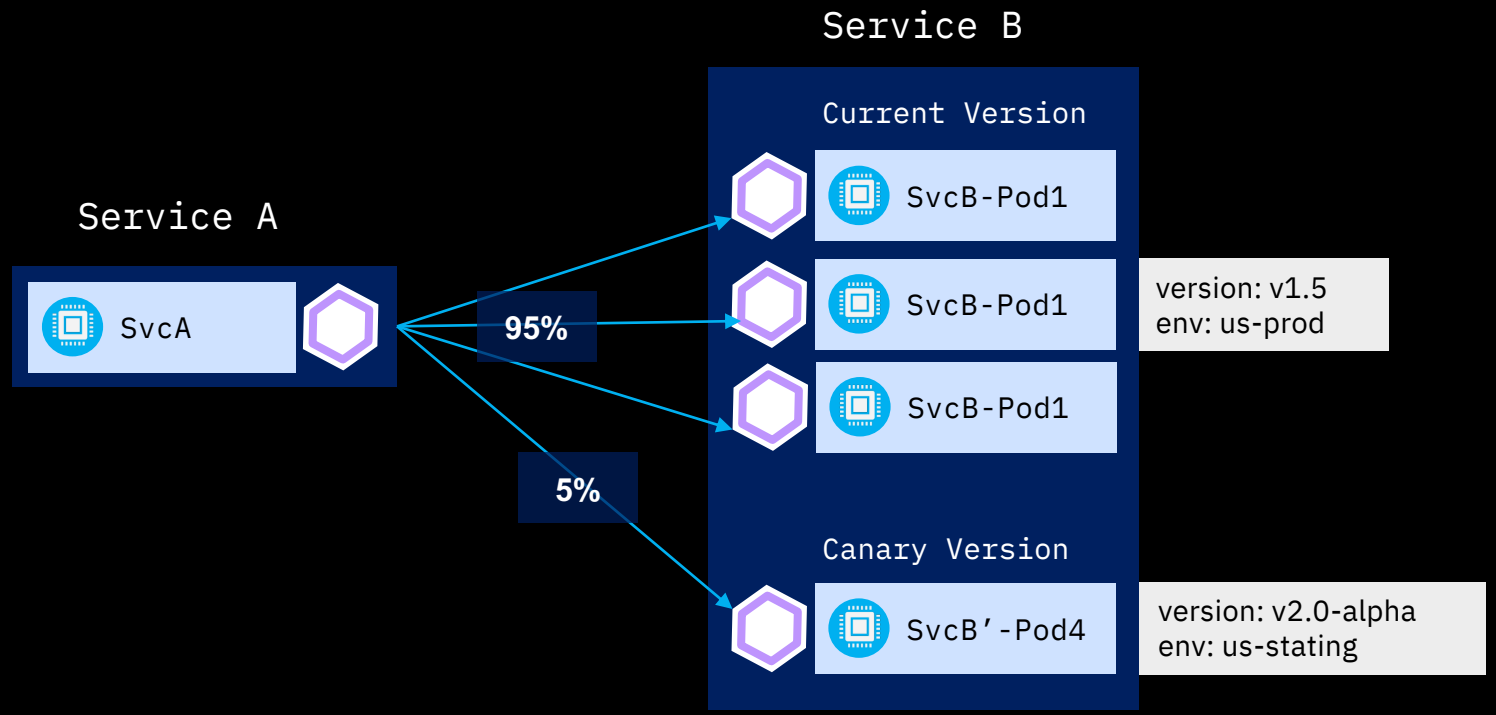



```
kind: VirtualService
destination:
  serviceB.cluster.local
match:
  source:
    serviceA.cluster.local
route:
  - labels:
    version: v2.0
    env: us-prod
  weight: 100
```

Update

CHALLENGE 1
ROLL OUT NEW VERSION WITHOUT DOWNTIME OR CHANGING CODE

Traffic Splitting

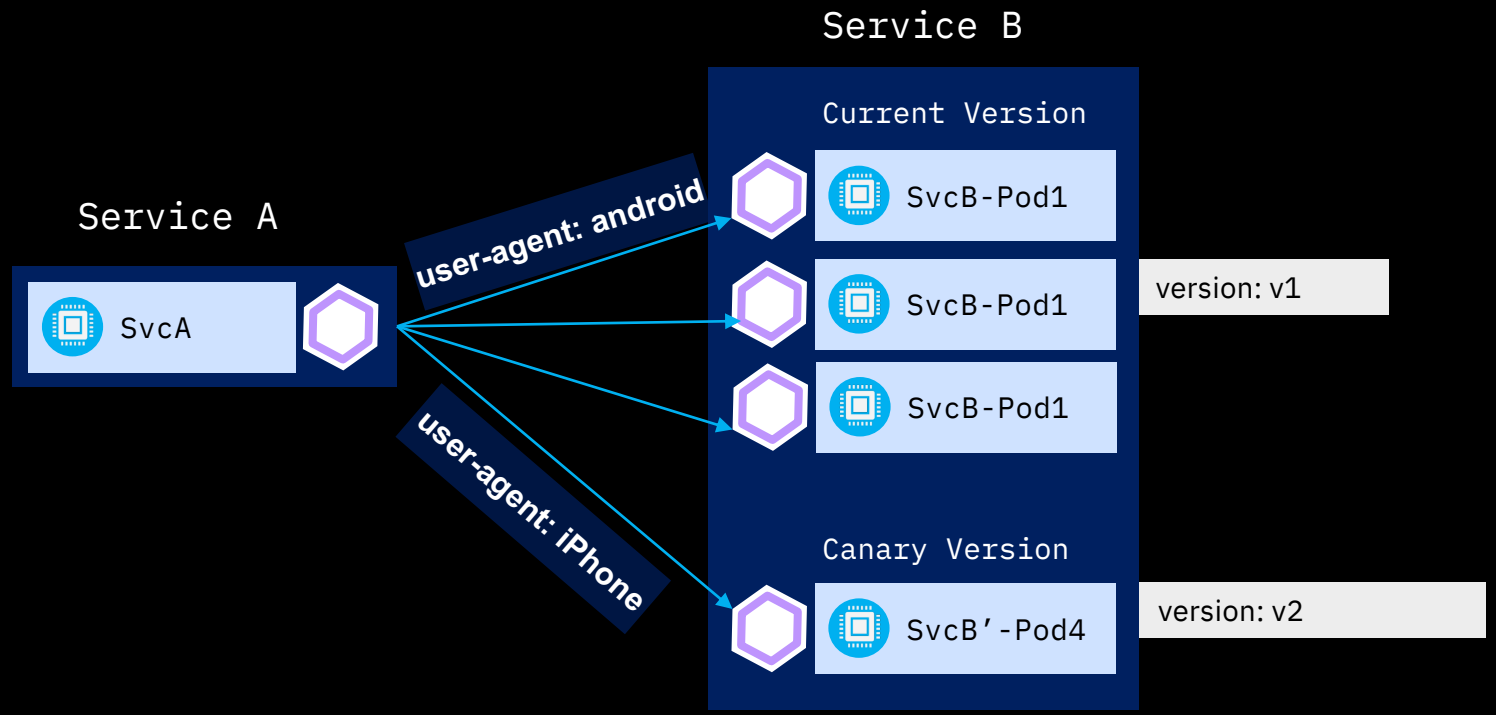



```
kind: VirtualService
destination:
  serviceB.cluster.local
match:
  source:
    serviceA.cluster.local
route:
  - labels:
    version: v1.5
    env: us-prod
  weight: 95
  - labels:
    version: v2.0-alpha
    env: us-staging
  weight: 5
```

CHALLENGE 2
HOW TO DO CANARY TESTING

Routing not based on the request content.
Staged rollouts with %-based traffic splits.

Traffic Steering

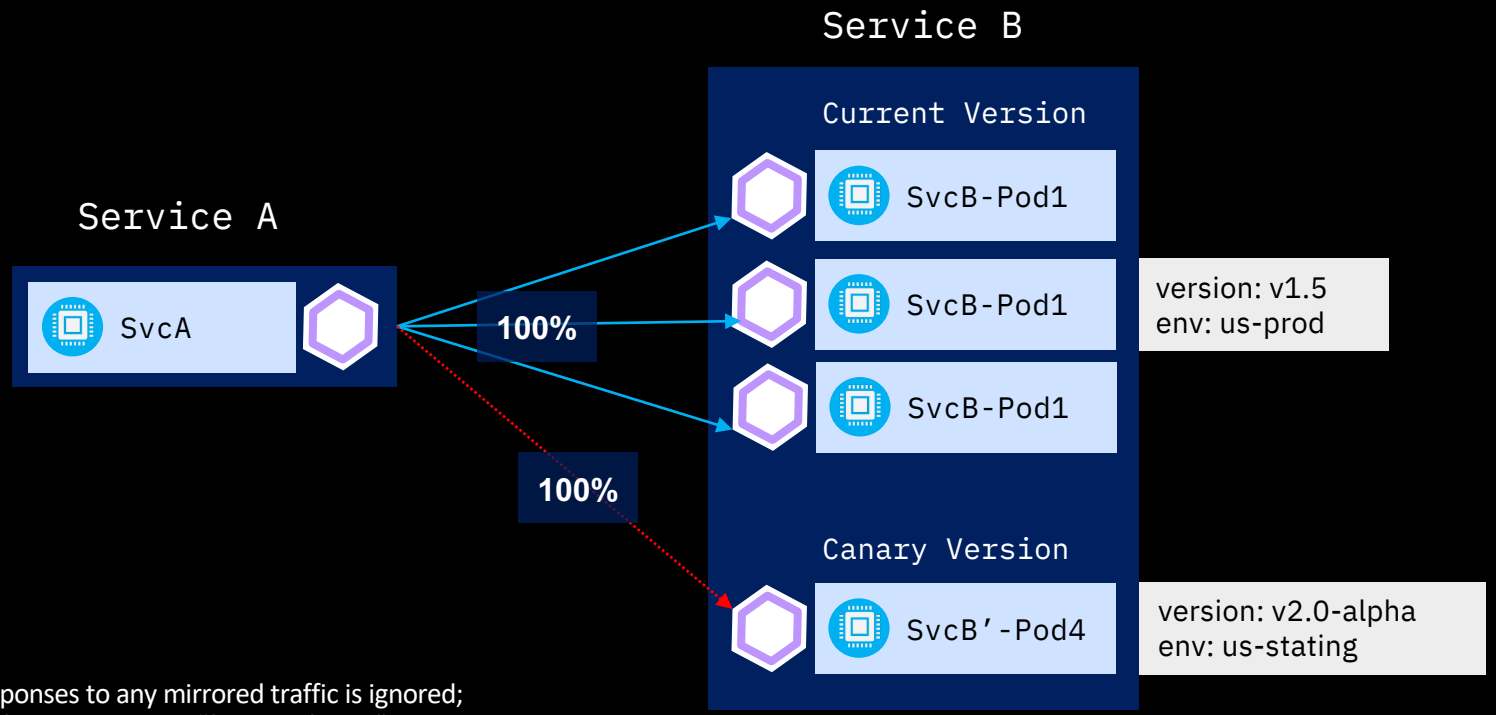



```
kind: VirtualService
destination:
 serviceB.cluster.local
match:
  httpHeaders:
    user-agent:
      regex:
        ^(.*?;)?(iPhone)(;.*)?$
 route:
- labels:
  version: v2
```

Routing based on the request content

CHALLENGE 3
HOW TO DO A/B TESTING

Traffic Mirroring



Responses to any mirrored traffic is ignored;
traffic is mirrored as "fire-and-forget"
You'll need to have the 0-weighted route to hint to
Istio to create the proper Envoy

```
kind: VirtualService
destination:
  serviceB.cluster.local
match:
  source:
    serviceA.cluster.local
route:
  - labels:
    version: v1.5
    env: us-prod
  weight: 100
  - labels:
    version: v2.0-alpha
    env: us-staging
  weight: 0
  mirror:
    name: httpbin
    labels:
      version: v2.0-alpha
      env: us-staging
```

CHALLENGE 4
THINGS DON'T ALWAYS GO CORRECTLY IN PRODUCTION...

Resiliency

Istio adds fault tolerance to your application without any changes to code

```
// Circuit breakers
destination: serviceB.example.cluster.local
policy:
- labels:
    version: v1
  circuitBreaker:
    simpleCb:
      maxConnections: 100
      httpMaxRequests: 1000
      httpMaxRequestsPerConnection: 10
      httpConsecutiveErrors: 7
      sleepWindow: 15m
      httpDetectionInterval: 5m
```

Resilience features

- Timeouts
- Retries with timeout budget
- Circuit breakers
- Health checks
- AZ-aware load balancing w/ automatic failover
- Control connection pool size and request load

CHALLENGE 4
THINGS DON'T ALWAYS GO CORRECTLY IN PRODUCTION...

Resiliency
Circuit Breakers

Istio

Connection pool

- Limits connections for reviews to invoke ratings
- **Limited to 1 concurrent connection, 1 request per connection (One concurrent request total)**
  - While requests are using all of the connections in a pool, any new requests are rejected

Outlier detection

- **If there are 3 requests in 2 seconds, reviews will be ejected for 3 minutes**
  - Request 1 will take more than 2 seconds, blocking the connection during that time
  - Request 2 won't get a connection, which will generate the first error
  - Request 3 won't get a connection, which will generate the second error, causing ejection

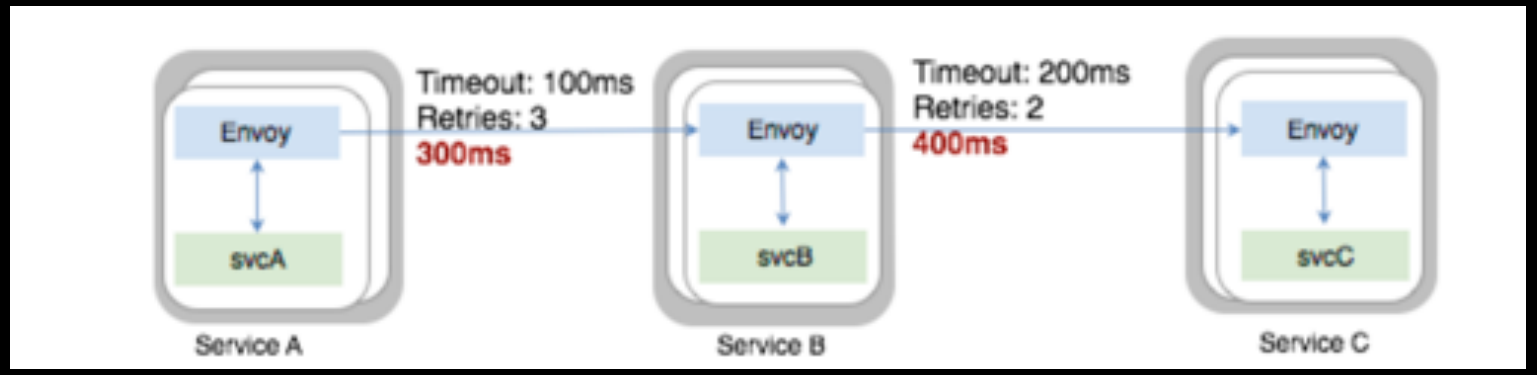
```
kind: DestinationRule
host: reviews
trafficPolicy:
  connectionPool:
    tcp:
      maxConnections: 1
    http:
      http1MaxPendingRequests: 1
      maxRequestsPerConnection: 1
  outlierDetection:
    consecutiveErrors: 2
    interval: 2s
    baseEjectionTime: 3m
    maxEjectionPercent: 100
```

CHALLENGE 4
THINGS DON'T ALWAYS GO CORRECTLY IN PRODUCTION...

Resiliency
Fault injection

Fault injection can be used for testing
- Faults are caused on a percentage of requests
- Faults can cause a request delay or failure

In this example, ratings is being invoked
- All of the requests from bar have a 2 second timeout
- 40% of the requests from bar also have a 5 second delay



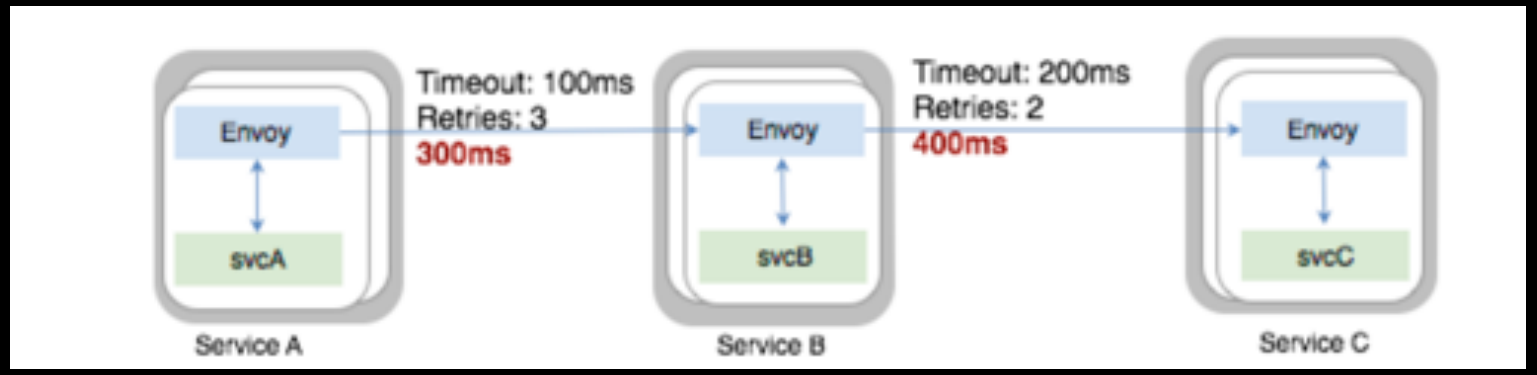

```
kind: VirtualService
hosts:
  - ratings
http:
- match:
  - headers:
      end-user:
        exact: bar
  fault:
    delay:
      percent: 40
      fixedDelay: 5s
  timeout: 2s
  route:
  - destination:
      host: ratings
```

CHALLENGE 4
HOW DO I INJECT FAULT TO MY MICROSERVICES TO PREPARE MYSELF?

Resiliency
Fault injection

Fault injection can be used for testing
- Faults are caused on a percentage of requests
- Faults can cause a request delay or failure

In this example, ratings is being invoked
- 20% of the requests from foo get an HTTP 500 error
- All other requests (not foo or bar) have a 4 second timeout



CHALLENGE 4
HOW DO I INJECT FAULT TO MY MICROSERVICES TO PREPARE MYSELF?

```
kind: VirtualService
hosts:
  - ratings
http:
- match:
  - headers:
      end-user:
        exact: foo
  fault:
    abort:
      percent: 20
      httpStatus: 500
  route:
  - destination:
      host: ratings
- route:
  - destination:
      host: ratings
  timeout: 4s
```

Timeout is measured after the host is invoked, it is calculated after delay period has passed.

The 40% of requests from bar will time out after 7 seconds (5 sec delay + 2 sec timeout)

Rate limiting

Istio protects your application from rogue actors by imposing ratelimits

```
Quotas:
- name: requestcount.quota.istio-system
  maxAmount: 5000
  validDuration: 1s
  overrides:
  - dimensions:
      destination: ratings
      source: reviews
      sourceVersion: v3
    maxAmount: 1
    validDuration: 1s
  - dimensions:
      destination: ratings
    maxAmount: 100
    validDuration: 1s
```

Rate limit

- Configurable limits with overrides
- Multiple rate limiting backends
- Conditional rate limiting

CHALLENGE 5
HOW CAN I LIMIT RATE FOR SOME OF MY SERVICES?

Rate limiting

Every distinct rate limit configuration represents a counter.
If the number of requests in the last validDuration duration exceed maxAmount, Mixer returns a RESOURCE_EXHAUSTED message to the proxy.

Global rate limit of 500 calls per second.

If “reviews” is called, it’s limited to one call every 5 seconds.

If “reviews” is called from 10.28.11.20, it’s limited to 99 calls per seconds.

```
kind: handler
quotas:
- name: requestcountquota.instance.istio-system
  maxAmount: 500
  validDuration: 1s

  overrides:

  - dimensions:
      destination: reviews
    maxAmount: 1
    validDuration: 5s

  - dimensions:
      destination: productpage
      source: "10.28.11.20"
    maxAmount: 99
    validDuration: 1s
```

CHALLENGE 5
HOW CAN I LIMIT RATE FOR SOME OF MY SERVICES?

Telemetry

Monitoring & tracing should not be an afterthought in the infrastructure

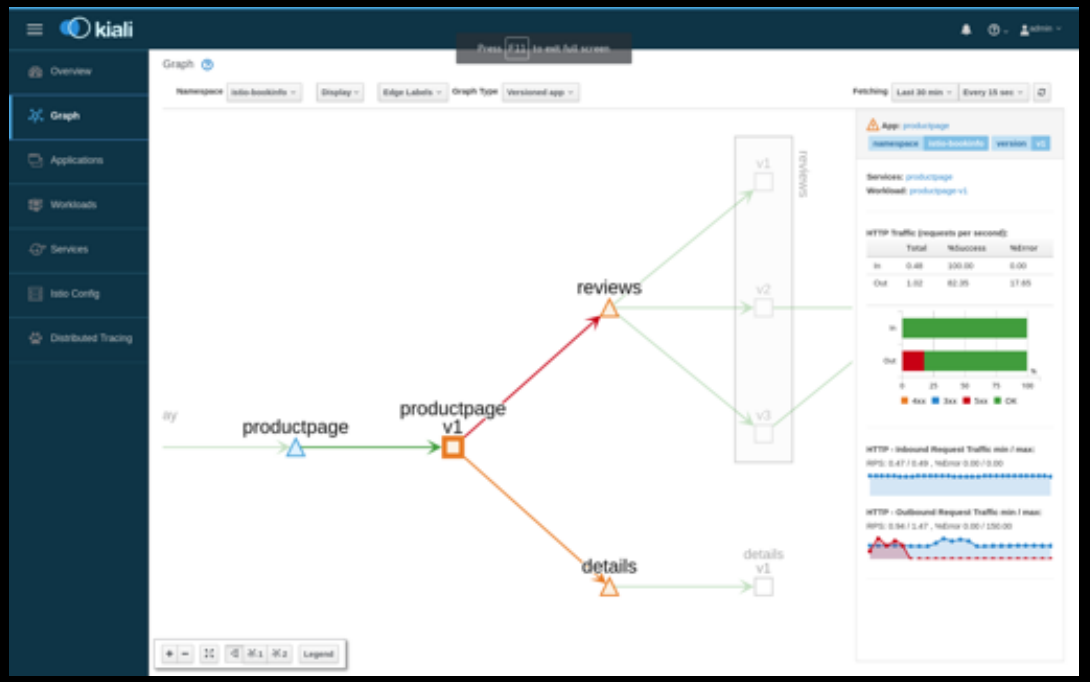
Goals

- Metrics without instrumenting apps
- Consistent metrics across fleet
- Trace flow of requests across services
- Portable across metric backend providers

CHALLENGE 6
I NEED TO VIEW WHAT IS GOING ON WHEN CRISIS ARISES

Kiali

Kiali (greek κιάλι)
monocular or *spyglass*
Visualise the service mesh topology, features like circuit breakers or request rates

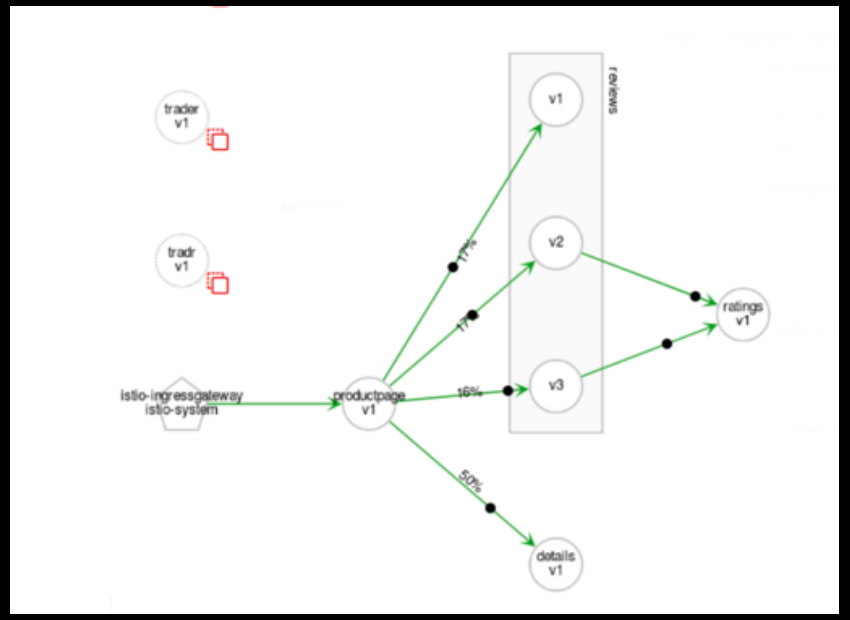


Features

Graph
- Health
- Types
- Side Panel
- Traffic Animation

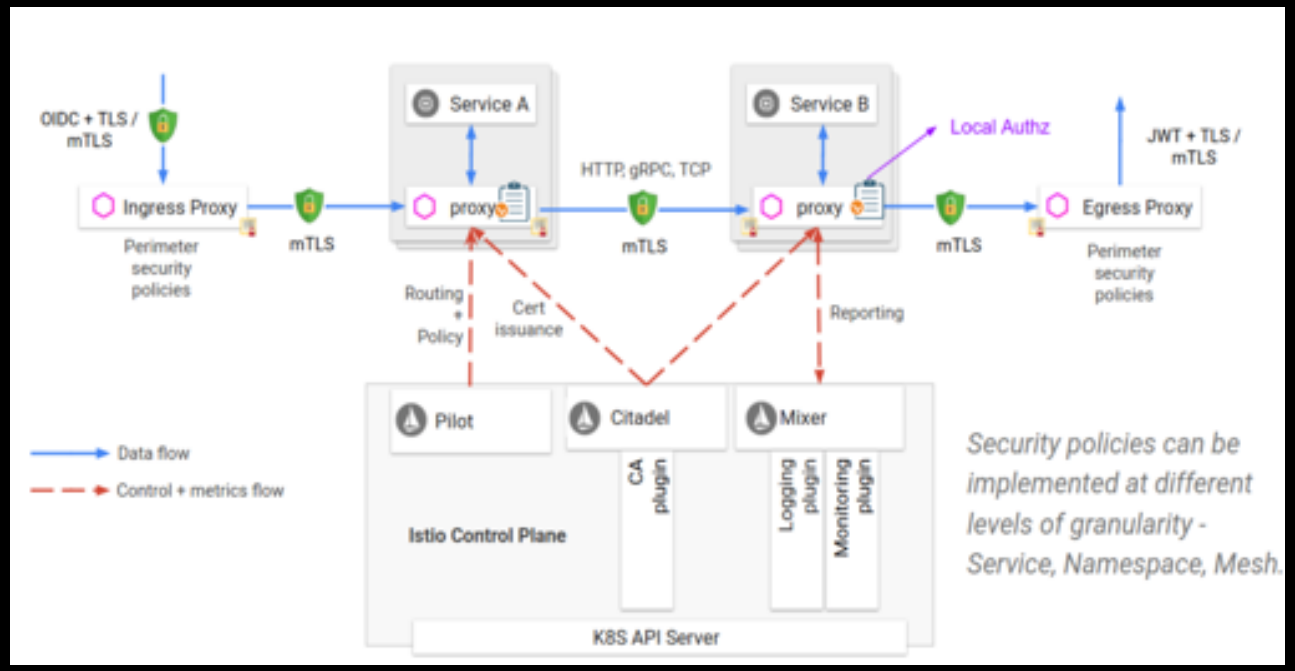
Applications, Workloads and Services
- Detailed Metrics

Traffic Routing
Istio compliance
Istio Configuration



CHALLENGE 6
I NEED TO VIEW WHAT IS GOING ON WHEN CRISIS ARISES

Security



Authentication

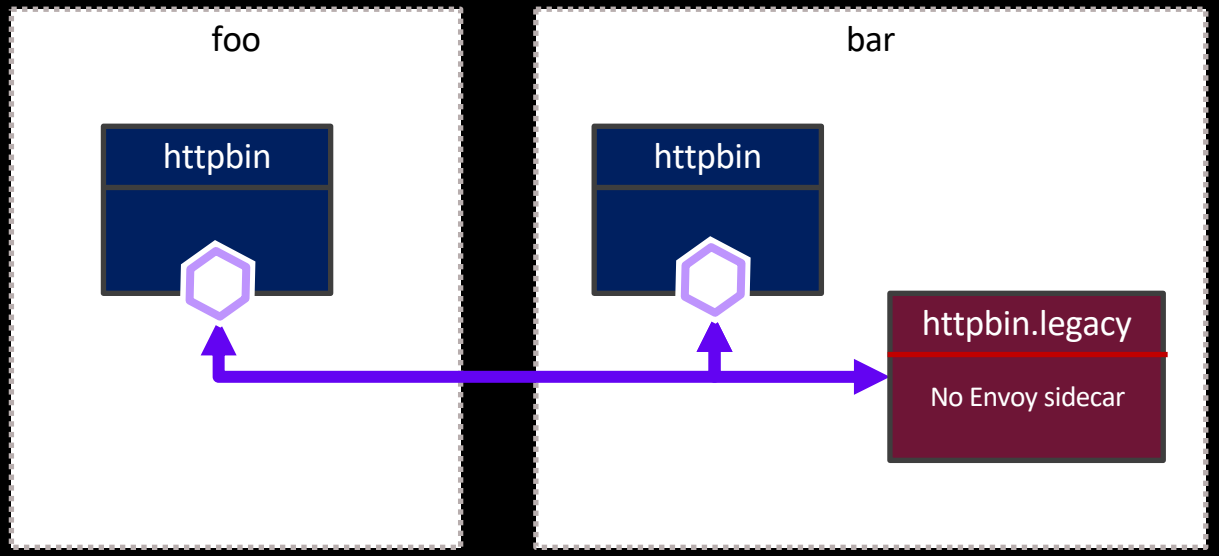
- Transport authentication, also known as service-to-service authentication
- Origin authentication, also known as end-user authentication

Authorization

- Based on RBAC
- Namespace-level, service-level and method-level access control for services

CHALLENGE 7
HOW CAN I SECURE MY SERVICES?

Security - Authentication

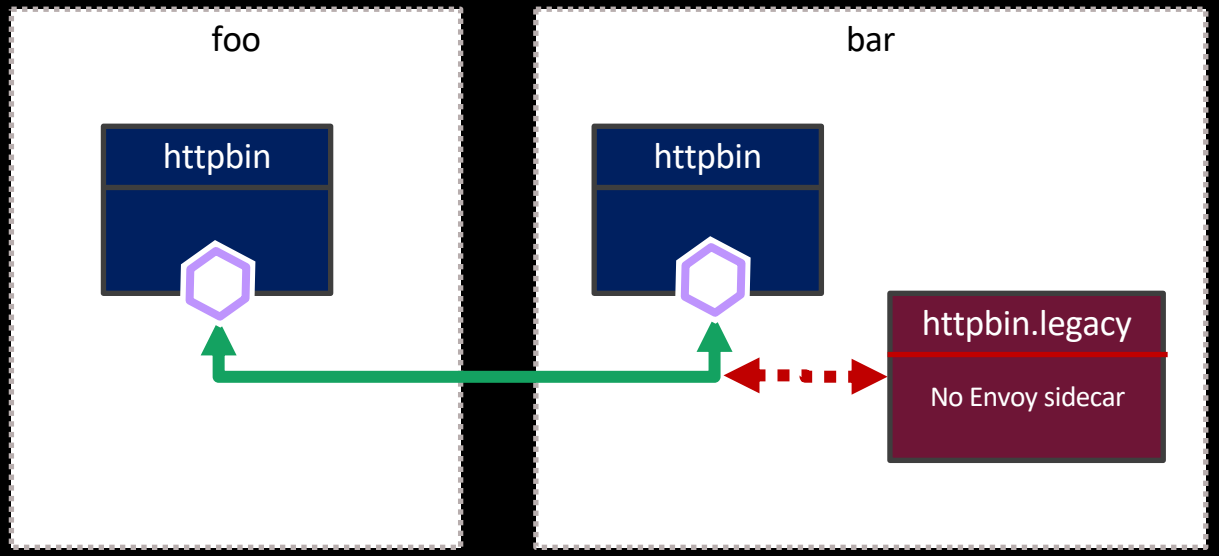


EXAMPLE

CHALLENGE 7
HOW CAN I SECURE MY SERVICES?

Not encrypted
Encrypted (TLS)
No communication

Security - Authentication

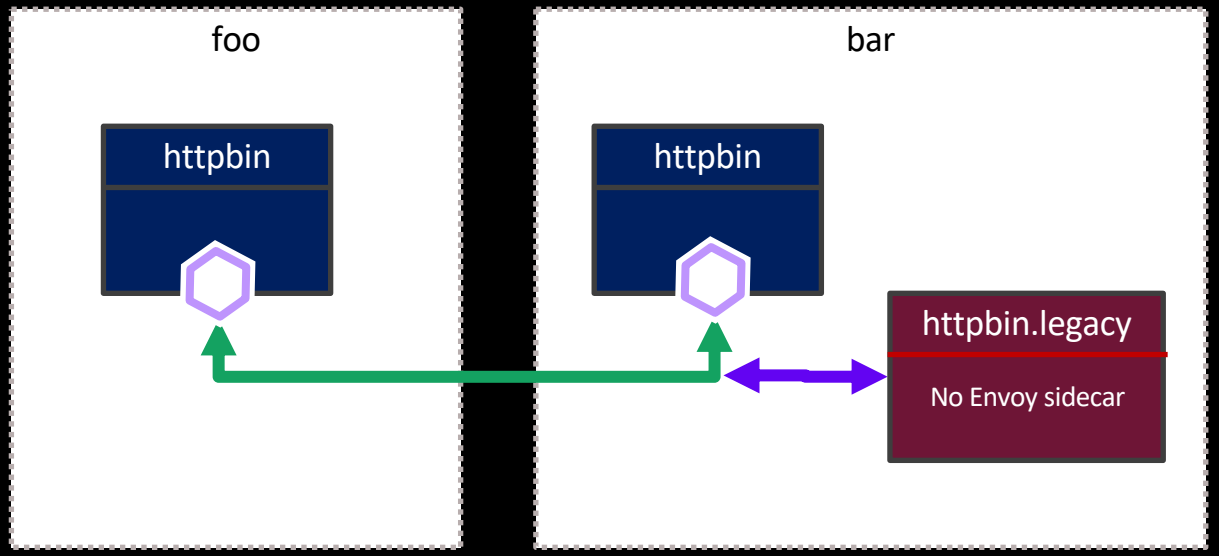



```
kind: DestinationRule
metadata:
  name: "default"
spec:
  host: "*.local"
  trafficPolicy:
    tls:
      mode: ISTIO_MUTUAL
```

CHALLENGE 7
HOW CAN I SECURE MY SERVICES?

Not encrypted
Encrypted (TLS)
No communication

Security - Authentication




```
kind: DestinationRule
metadata:
 name: "httpbin-legacy"
spec:
 host: "httpbin.legacy.svc.cluster.local"
 trafficPolicy:
   tls:
     mode: DISABLE
```

CHALLENGE 7
HOW CAN I SECURE MY SERVICES?

Not encrypted
Encrypted (TLS)
No communication

Security - Authorization

Istio Role Based Access

- **OFF**: Istio authorization is disabled.
- **ON**: enabled for all services in the mesh.
- **ON_WITH_INCLUSION**: enabled for all services specified in the inclusion field.
- **ON_WITH_EXCLUSION**: enabled for all services except the ones in the exclusion field.

```
kind: RbacConfig
metadata:
  name: my-user
spec:
  mode: 'ON_WITH_INCLUSION'
  inclusion:
    services:
    - "webapp.default.svc.cluster.local"
    - "frontend.default.svc.cluster.local"
    - "feedback.default.svc.cluster.local"
```

CHALLENGE 7
HOW CAN I SECURE MY SERVICES?

Security - Authorization

Istio Role Based Access

- **services**: A list of service names.
- **methods**: A list of HTTP method names (GET, POST,…)
- **paths**: HTTP paths (in the form of /packageName.serviceName/methodName)
- **constraints**: additional conditions for your rules.

```
kind: ServiceRole
metadata:
  name: service-viewer
spec:
  rules:
  - services:
    - "webapp.default.svc.cluster.local"
    - "frontend.default.svc.cluster.local"
    methods: ["GET", "HEAD"]
    paths: ["*"]
    constraints:
    - key: request.headers[version]
      values: ["v1", "v2"]
```

CHALLENGE 7
HOW CAN I SECURE MY SERVICES?

Security - Authorization

Access for:

- Service in **Namespace “default” only accessible by authenticated users** and services
- User: "*" assigns the ServiceRole to all (both authenticated and unauthenticated)

```
kind: ServiceRoleBinding
metadata:
  name: binding-products-all-authenticated-users
spec:
  subjects:
  - properties:
      source.principal: "*"
  - properties:
      source.namespace: "default”
  roleRef:
    kind: ServiceRole
    name: ”service-viewer"
```

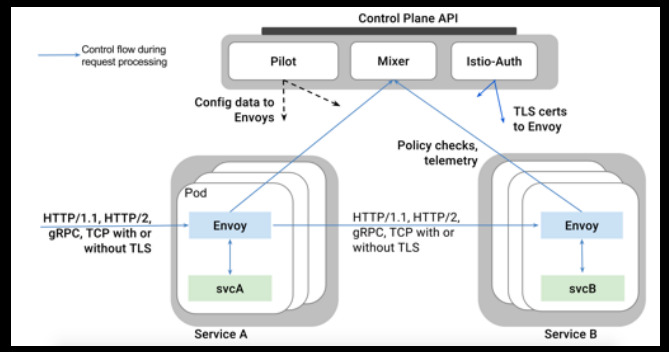
CHALLENGE 7
HOW CAN I SECURE MY SERVICES?

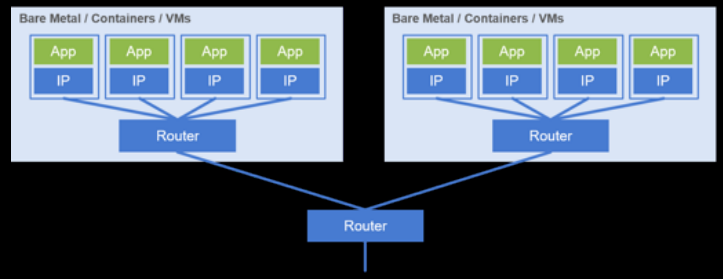
Security
Using Istio in concert with Calico

Istio

PROJECT CALICO

“RPC” — L7	Layer	“Network” — L3-4
Userspace	Implementation	Kernel
Pod	Enforcement Point	Node

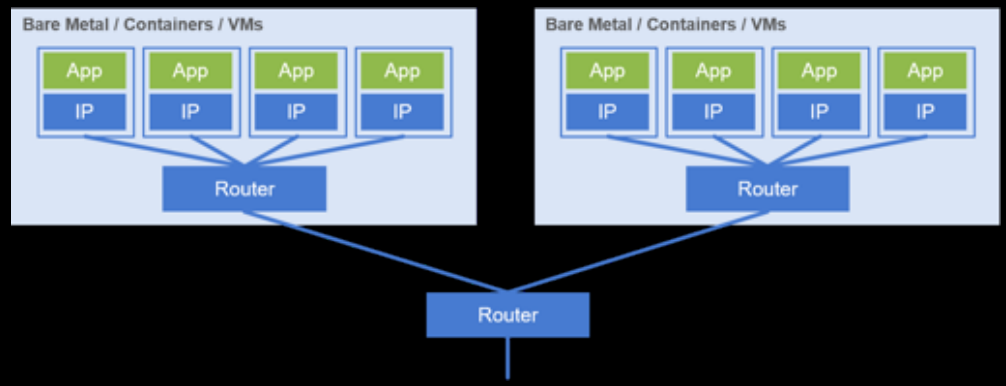




Security
Using Istio in concert with Calico

Operates at **Layer 3**, which is the network layer

- Has the advantage of being **universal** (DNS, SQL, real-time streaming, ...)
- Can extend beyond the service mesh (including to **bare metal or VM** endpoints not under the control of Kubernetes).
- Calico's policy is enforced at the host node, outside the network namespace of the guest pods.
- Based on **iptables**, which are packet filters implemented in the standard Linux kernel, it is extremely fast.



Security
Using Istio in concert with Calico

“RPC” — L7	Layer	“Network” — L3-4
Userspace	Implementation	Kernel
Pod	Enforcement Point	Node
Ideal for applying policy in support of operational goals, like service routing, retries, circuit-breaking, etc	Strengths	Universal, highly efficient, and isolated from the pods, making it ideal for applying policy in support of security goals

Service Mesh - Bad Idea ?

A Service Mesh is not always the right solution…

- **Service Meshes are Opinionated**
  They are a *platform* solution. “Work their way”
- **Service Meshes are Complex**
  Adds considerable complexity with sidecars and control plane
- **Service Meshes can be Slow**
  Routing traffic through a series of proxies can get painfully slow (about 700 nodes → reflector)
- **Service Meshes are for Developers**
  Focused primarily on Developer view.

https://glasnostic.com/blog/comparing-service-meshes-linkerd-vs-istio

Getting started

- Go to https://istio.io/

Download ISTIO Release

With Kubectl

```
$ kubectl apply -f install/kubernetes/helm/istio/templates/crds.yaml

$ kubectl apply -f install/kubernetes/istio-demo.yaml
```

With HELM Templating

```
$ kubectl create namespace istio-system

$ helm template --name istio
      --namespace istio-system
      --set grafana.enabled=true
      --set servicegraph.enabled=true
      --set kiali.enabled=true > istio.yaml

$ kubectl apply -f istio.yaml
```

Useful links

- Web istio.io
- Twitter: @Istiomesh
- Istio 101: https://github.com/IBM/istio101
- Traffic management using Istio: https://ibm.co/2F7xSnf
- Resiliency and fault-tolerance using Istio: https://bit.ly/2qStF2B
- Reliable application roll out and operations using Istio: https://bit.ly/2K9IRQX

QUESTIONS?

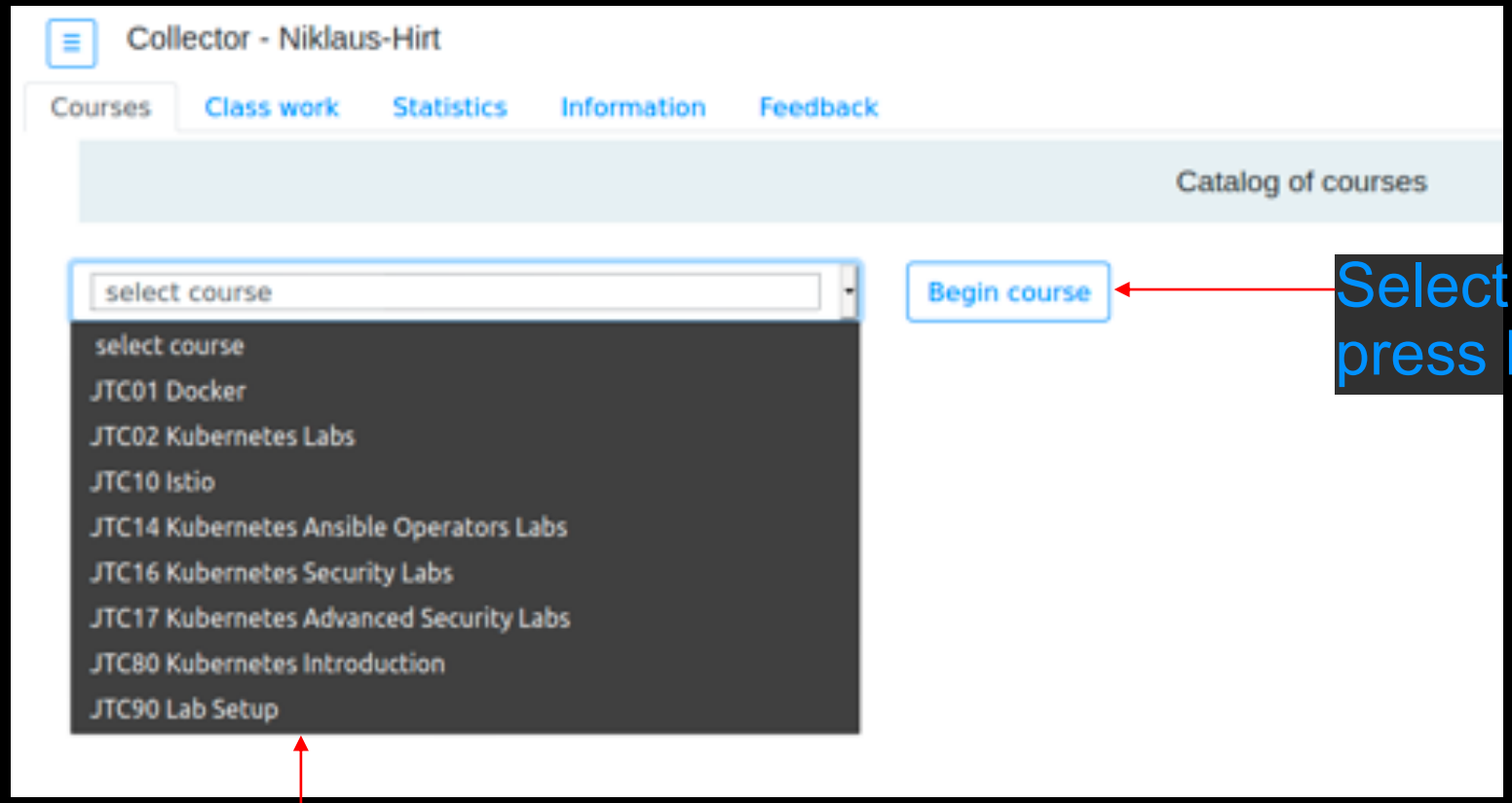
Kubernetes Workshop Series

Mesh Networking - *Hands On*

02

Starting Course JTC10 ISTIO

Name will be shown



Select course and press button to begin

Current course catalog

JTC10 Istio

Lab 1 - Istio Introduction

Lab 2 - Installing Istio

Lab 3 - Deploy the Bookinfo App

Lab 4 - Monitoring with Kiali

Lab 5 - Traffic flow management

Lab 6 - Access policy enforcement

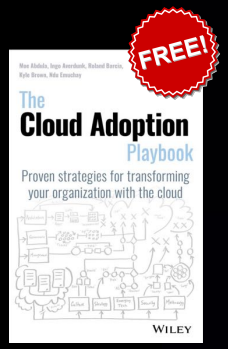
Lab 7 - Telemetry data aggregation

READY SET GO!!!!

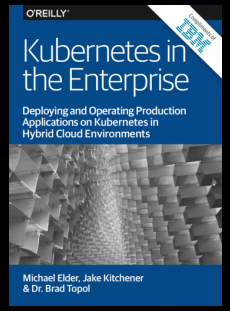
Duration: 90 mins

QUESTIONS?

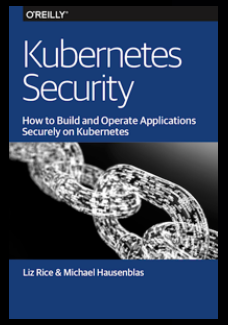
Kubernetes – Some Reading Tips



The de facto guide to improving your enterprise with the cloud, created by distinguished members of our Solution Engineering team
http://ibm.biz/playbook



Deploying and Operating Production Applications on Kubernetes in Hybrid Cloud Environments
https://ibm.co/2LQketN (excerpt)



https://kubernetes-security.info/

Videos, sources and documentation will be available here:

All Workshop Recordings
https://www.youtube.com/channel/UCIS0jmGOQrG2AKKPkTJYj9w/videos

https://github.com/niklaushirt/k8s_training_public

https://github.com/niklaushirt/training

Before you go...

We will collect some feedback.

Please make sure you can access https://kahoot.it/
either on your PC of Phone.

You will get the Game PIN
later in the training.

See you in September!

Kubernetes Workshop Series
Kubernetes

Offering extended
New courses will be added

Enjoy the holidays!

Niklaus Hirt

nikh@ch.ibm.com

@nhirt

THANK YOU!!!!