Kubernetes from Zero to Hero

Slide:https://reurl.cc/EKV2b1

Repo:https://reurl.cc/L1a027

Hello!

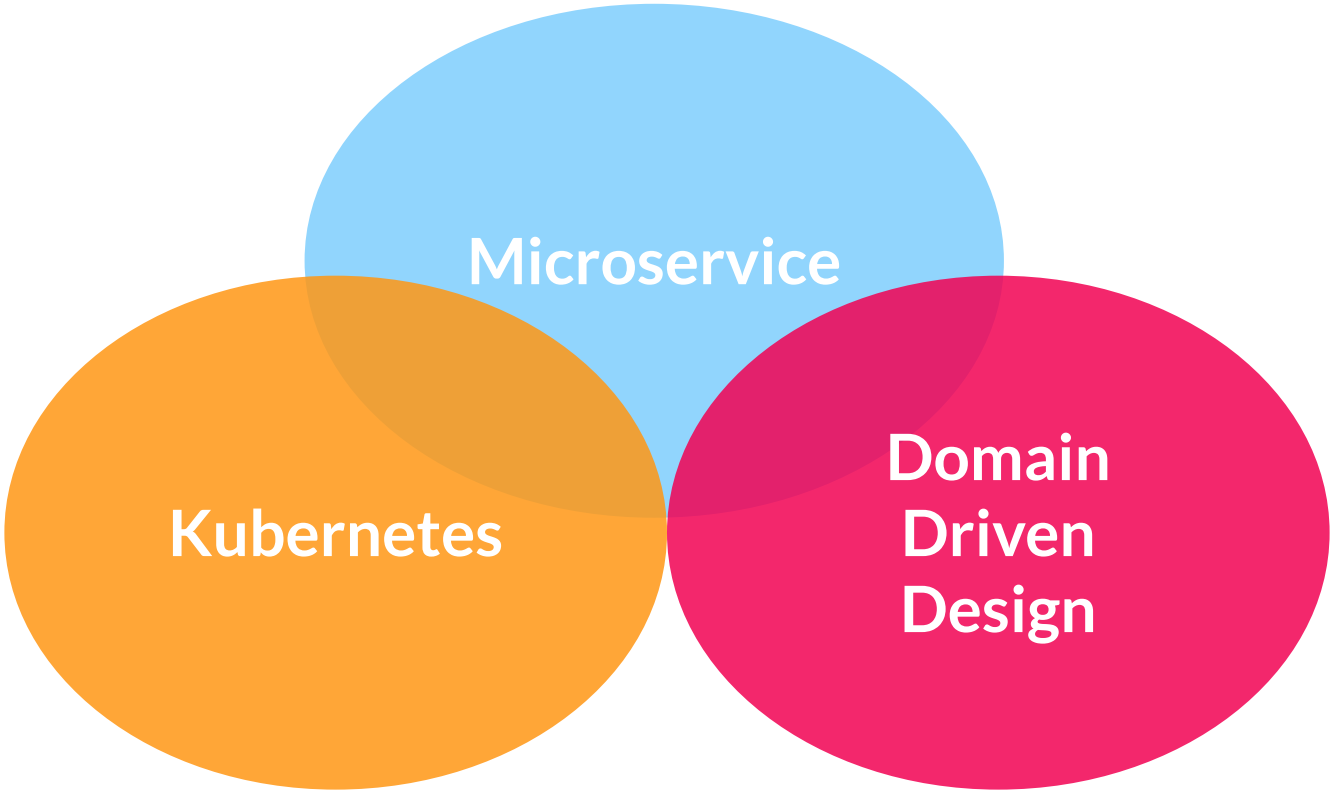
HashiCorp User Group
TAIPEI

https://bit.ly/taipei-hug

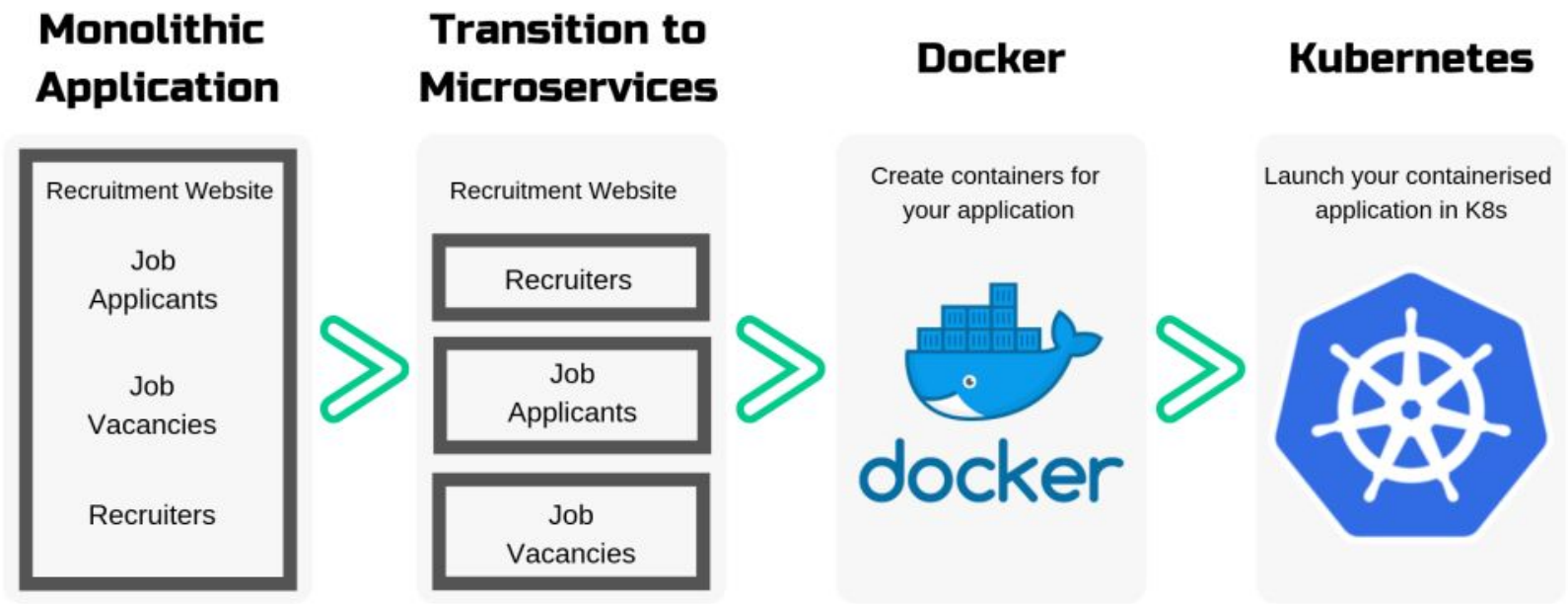
https://t.me/TaiwanHashiCorpUserGroup

Have You Ever Used Kubernetes?

Why Kubernetes?


Microservice
Kubernetes
Domain
Driven
Design

代誌不是憨人想的那麼簡單



Outline

CH00 EKS Setup

CH01 Kubernetes 101

CH02 Deploy Application by Helm

CH03 Accelerate Development by Skaffold

CH04 IAM Role for Service Account

CH05 Service Mesh by Istio

CHoo

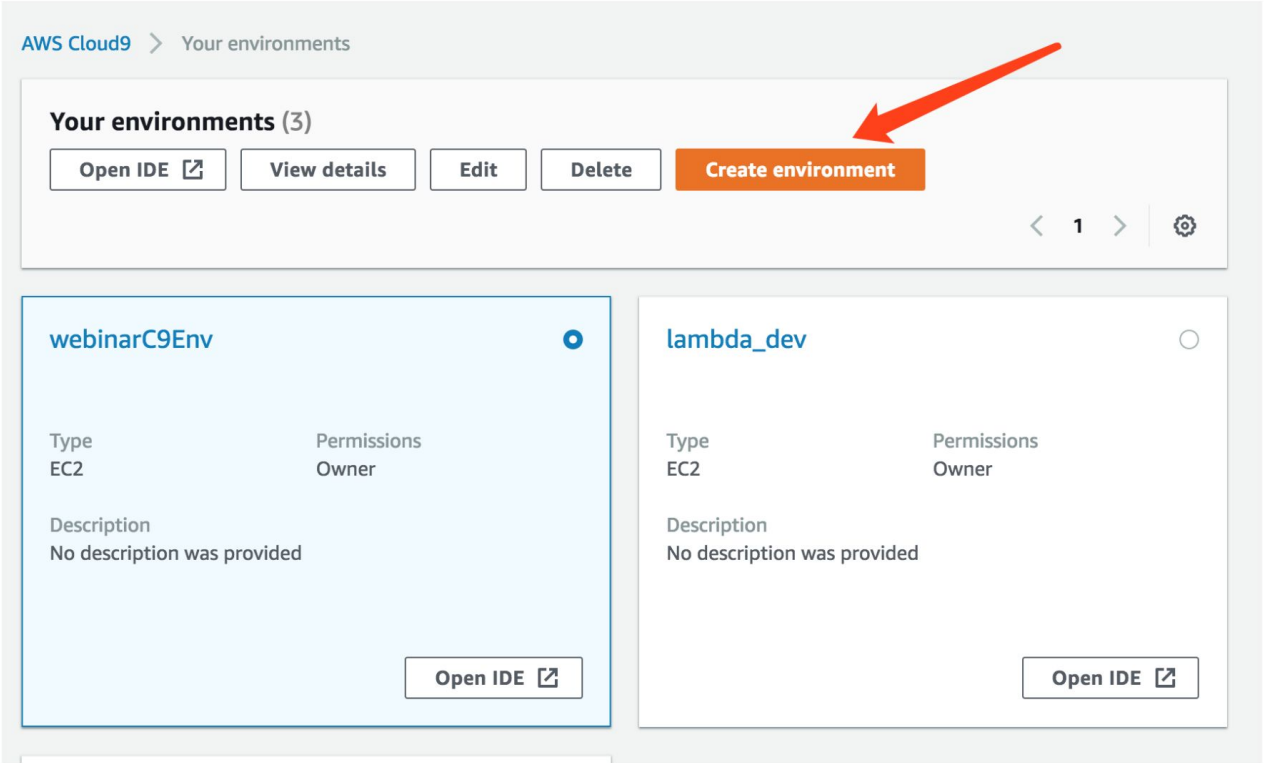
EKS Setup

Setup Cloud9

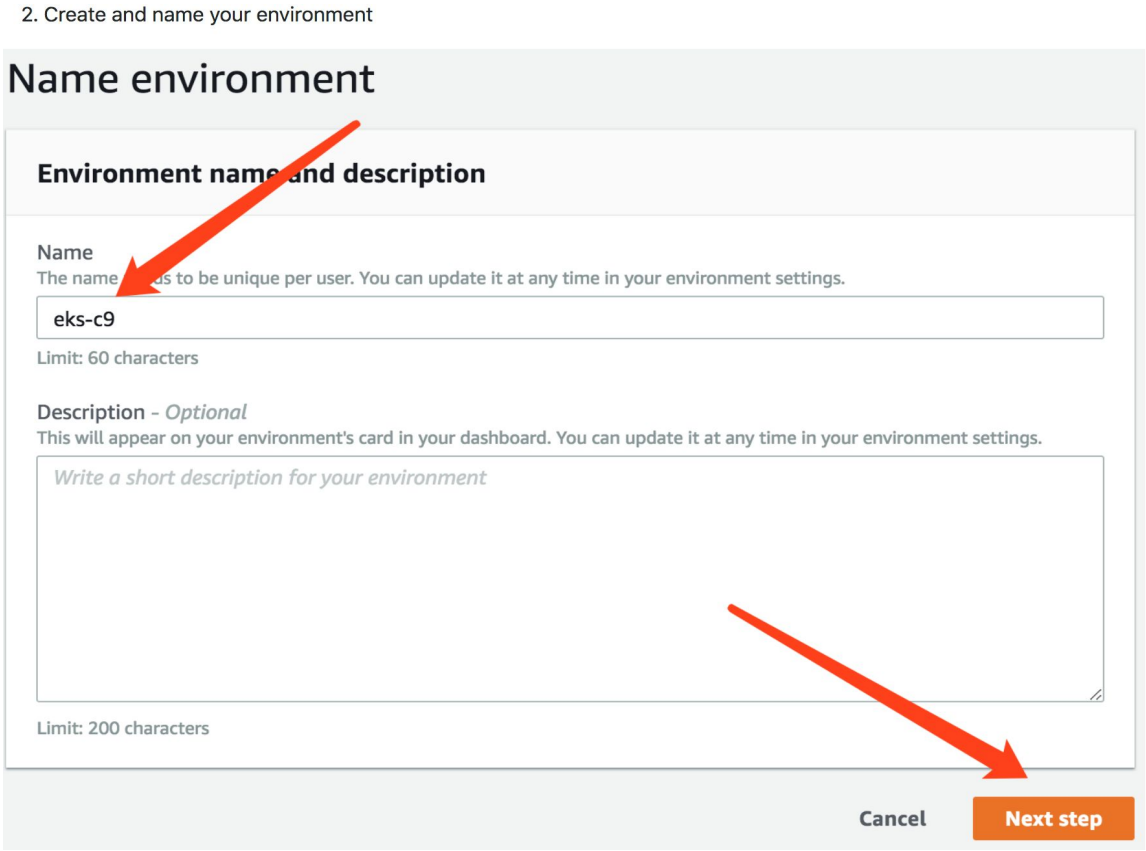
Following pictures of installation guide comes from:
https://github.com/pahud/amazon-eks-workshop

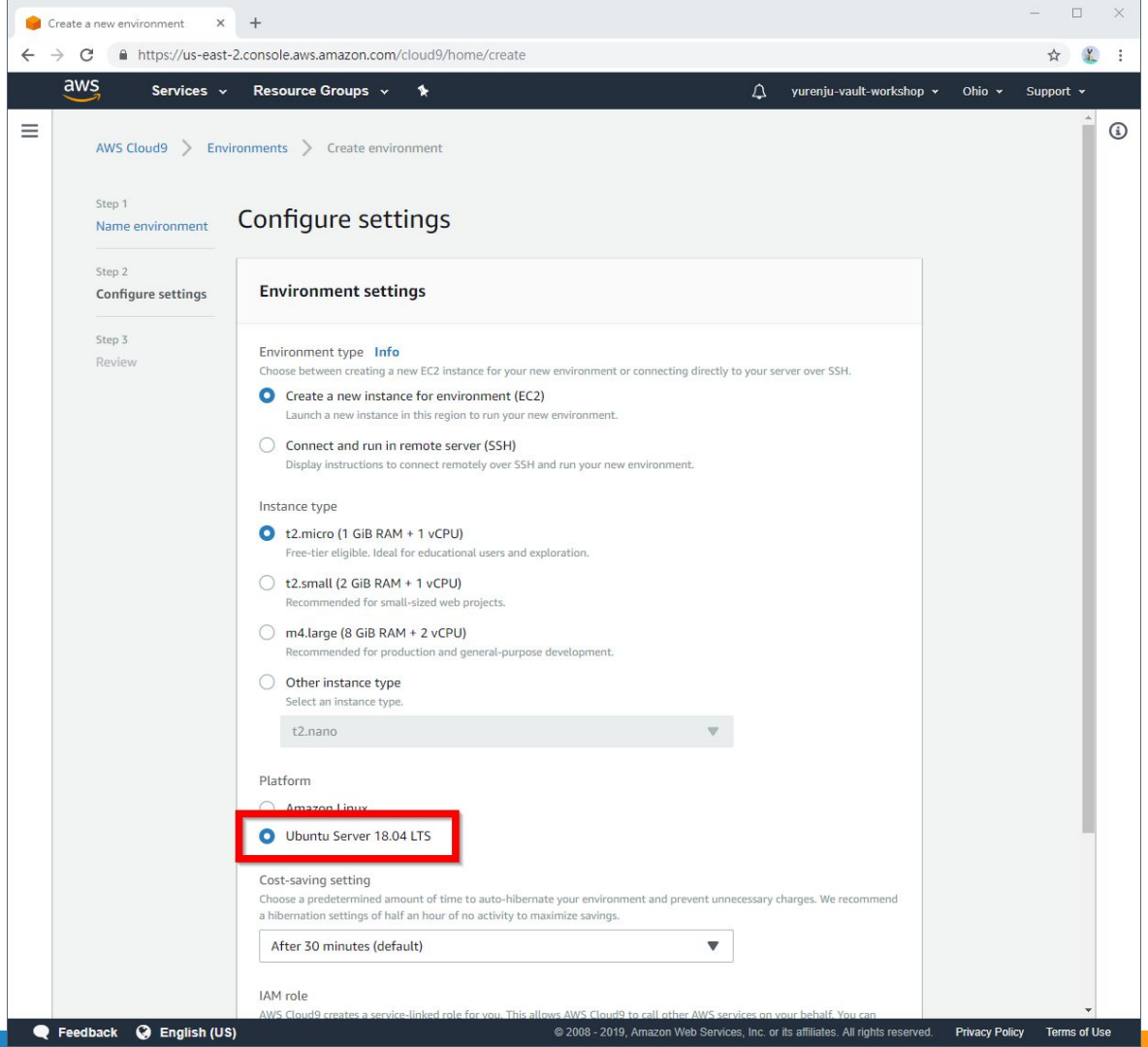
Create Environment

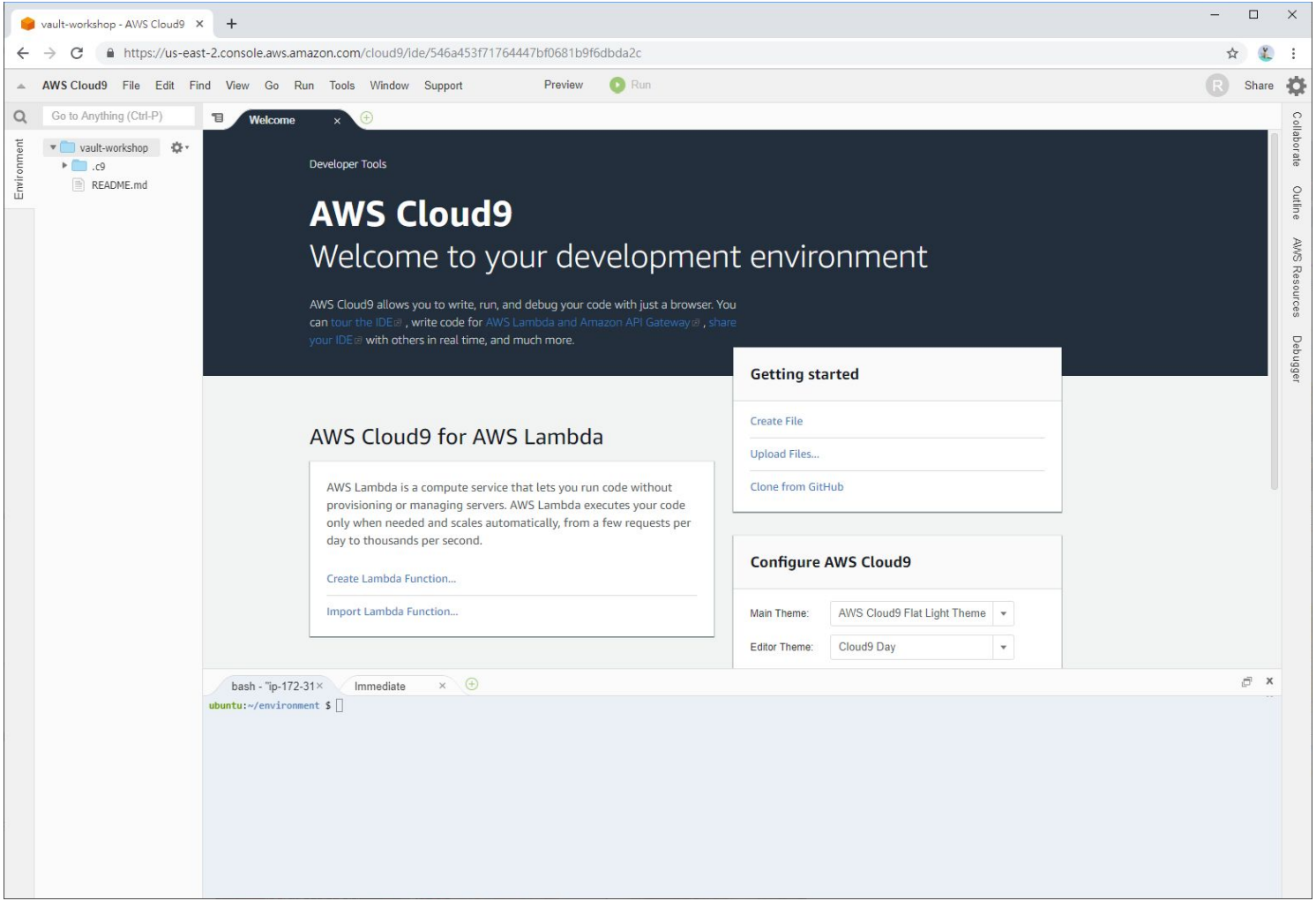
1. Spin up your Cloud9 IDE from AWS console.



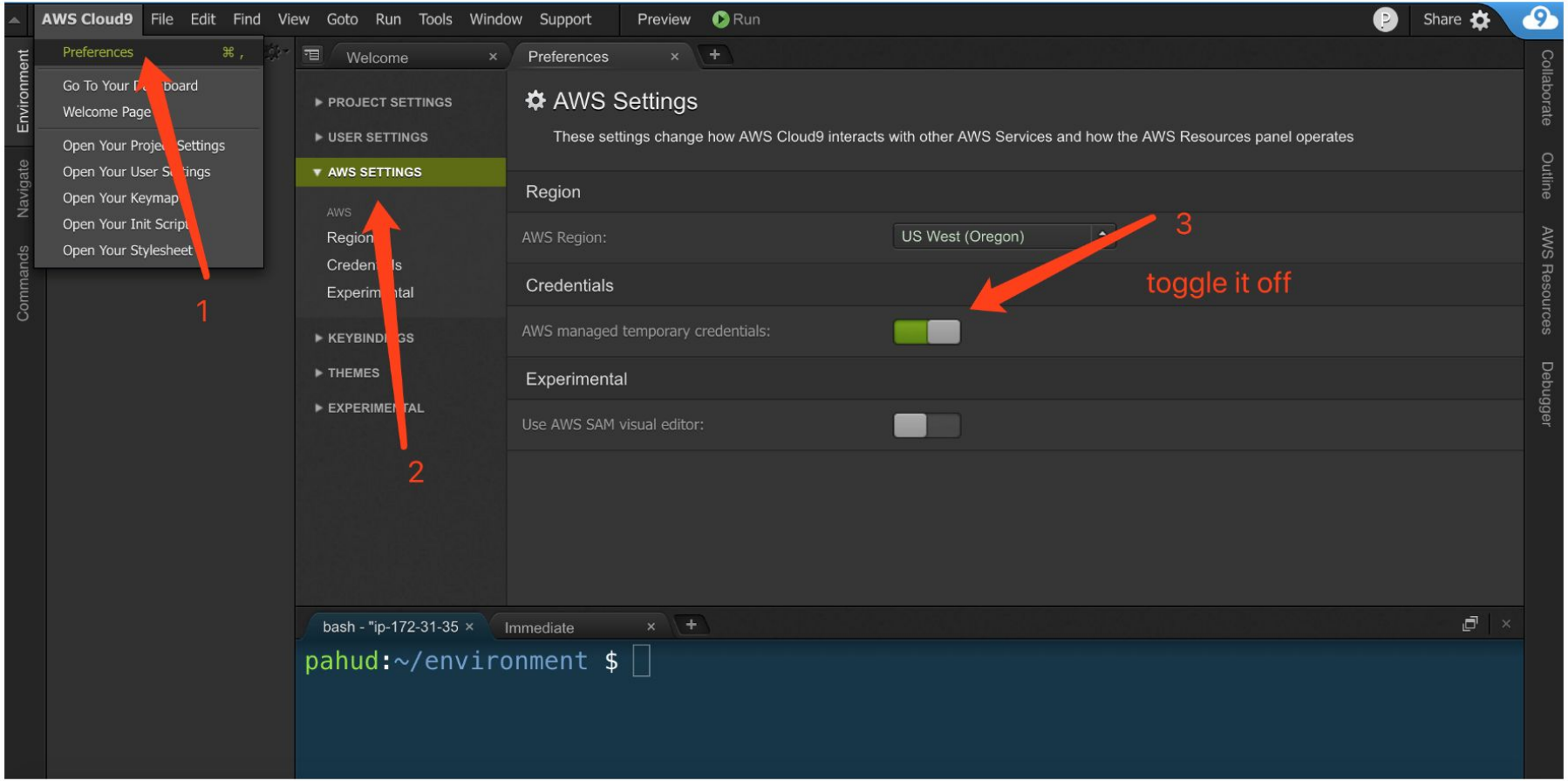
Name Environment



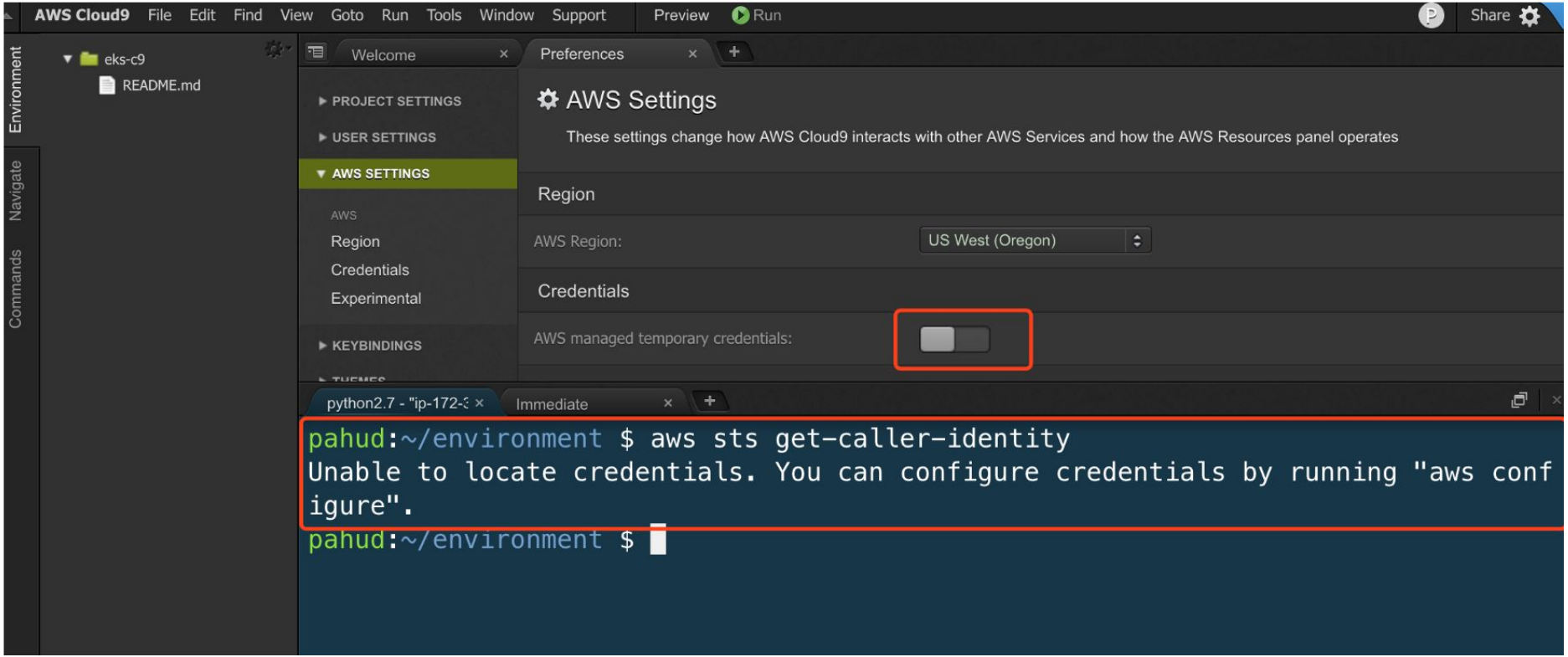

Create a new environment
https://us-east-2.console.aws.amazon.com/cloud9/home/create
aws
Services
Resource Groups
yurenju-vault-workshop
Ohio
Support
AWS Cloud9 > Environments > Create environment
Step 1
Name environment
Step 2
Configure settings
Step 3
Review
Configure settings
Environment settings
Environment type Info
Choose between creating a new EC2 instance for your new environment or connecting directly to your server over SSH.
Create a new instance for environment (EC2)
Launch a new instance in this region to run your new environment.
Connect and run in remote server (SSH)
Display instructions to connect remotely over SSH and run your new environment.
Instance type
t2.micro (1 GiB RAM + 1 vCPU)
Free-tier eligible. Ideal for educational users and exploration.
t2.small (2 GiB RAM + 1 vCPU)
Recommended for small-sized web projects.
m4.large (8 GiB RAM + 2 vCPU)
Recommended for production and general-purpose development.
Other instance type
Select an instance type.
t2.nano
Platform
Amazon Linux
Ubuntu Server 18.04 LTS
Cost-saving setting
Choose a predetermined amount of time to auto-hibernate your environment and prevent unnecessary charges. We recommend a hibernation settings of half an hour of no activity to maximize savings.
After 30 minutes (default)
IAM role
Feedback
English (US)
© 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use


vault-workshop - AWS Cloud9
https://us-east-2.console.aws.amazon.com/cloud9/ide/546a453f71764447bf0681b9f6dbda2c
AWS Cloud9 File Edit Find View Go Run Tools Window Support
Preview
Run
Share
Go to Anything (Ctrl-P)
Environment
vault-workshop
.c9
README.md
Welcome
Developer Tools
AWS Cloud9
Welcome to your development environment
AWS Cloud9 allows you to write, run, and debug your code with just a browser. You can tour the IDE, write code for AWS Lambda and Amazon API Gateway, share your IDE with others in real time, and much more.
Getting started
Create File
Upload Files...
Clone from GitHub
AWS Cloud9 for AWS Lambda
AWS Lambda is a compute service that lets you run code without provisioning or managing servers. AWS Lambda executes your code only when needed and scales automatically, from a few requests per day to thousands per second.
Create Lambda Function...
Import Lambda Function...
Configure AWS Cloud9
Main Theme:
AWS Cloud9 Flat Light Theme
Editor Theme:
Cloud9 Day
bash - "ip-172-31
Immediate
ubuntu:~/environment $
Collaborate
Outline
AWS Resources
Debugger

5. We need to turn off the Cloud9 temporarily provided IAM credentials.



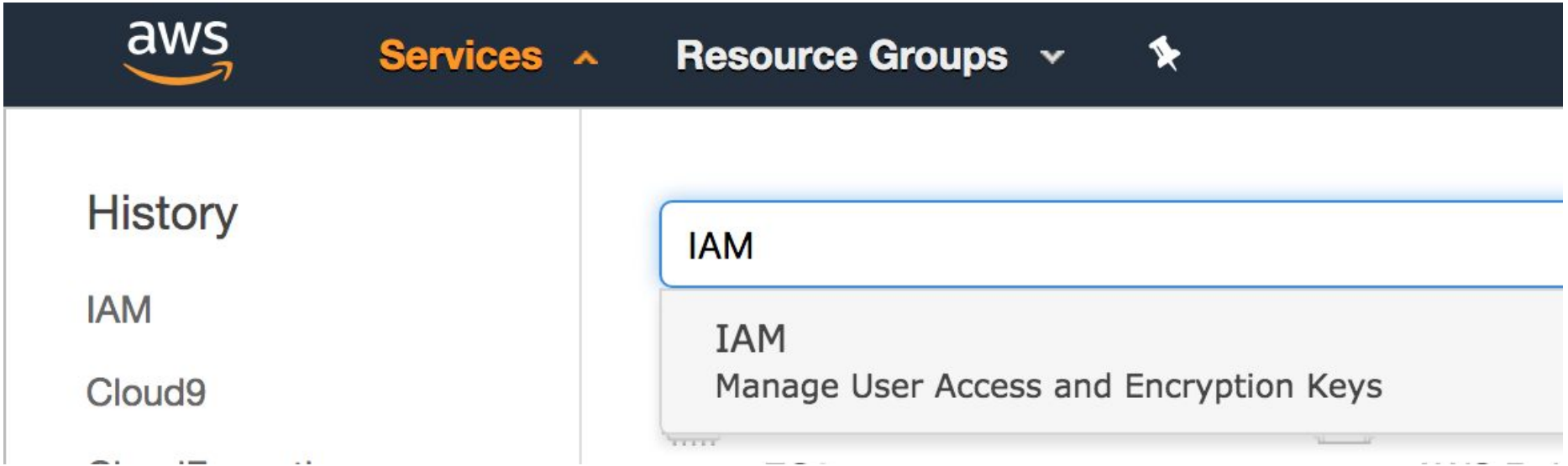
6. When you turn off the temporary credentials, you should not be able to un AWS CLI now.



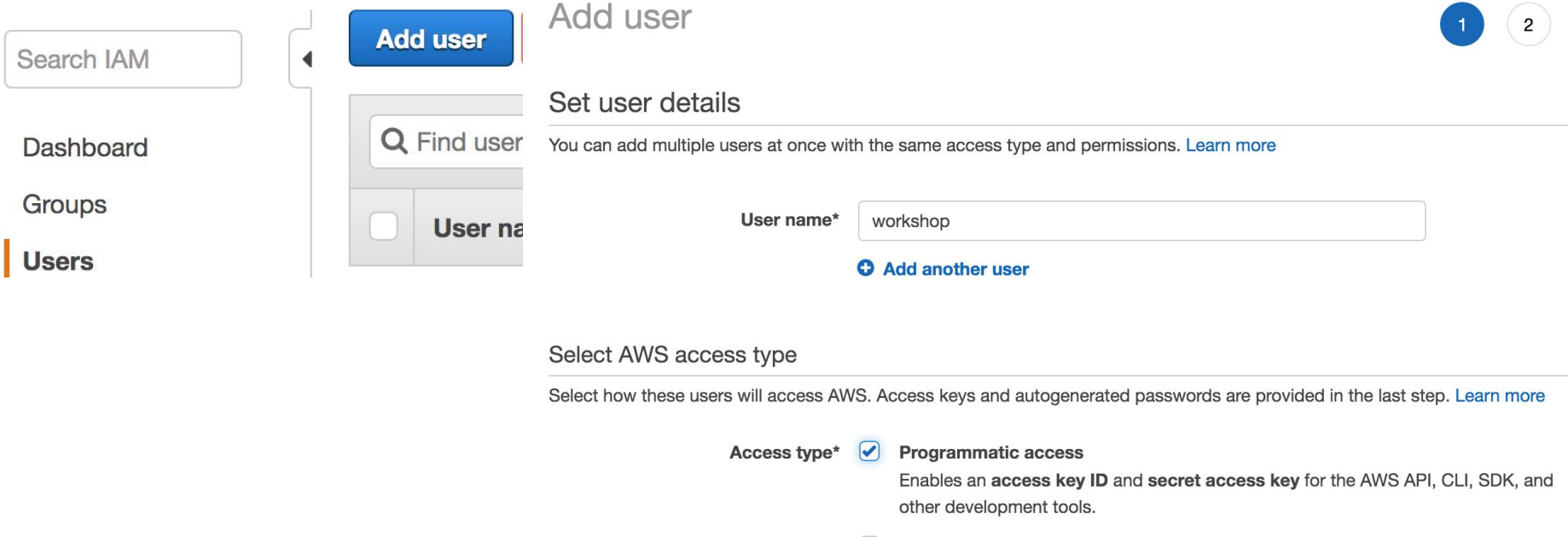
```
pahud:~/environment $ aws sts get-caller-identity
Unable to locate credentials. You can configure credentials by running "aws conf
igure".
pahud:~/environment $ aws configure
AWS Access Key ID [None]: [redacted]
AWS Secret Access Key [None]: [redacted]
Default region name [None]: us-west-2
Default output format [None]:
pahud:~/environment $ aws sts get-caller-identity
{
    "Account": "9[redacted]",
    "UserId": "AIDAJVHX3XBRH4E4UGWWK",
    "Arn": "arn:aws:iam::9[redacted]:user/pahud"
}
pahud:~/environment $
```

7. execute '**aws configure**' to configure the credentials for your IAM user. Make sure this IAM User has **AdministratorAccess** and run '**aws sts get-caller-identity**' - you should be able to see the returned JSON output like this.

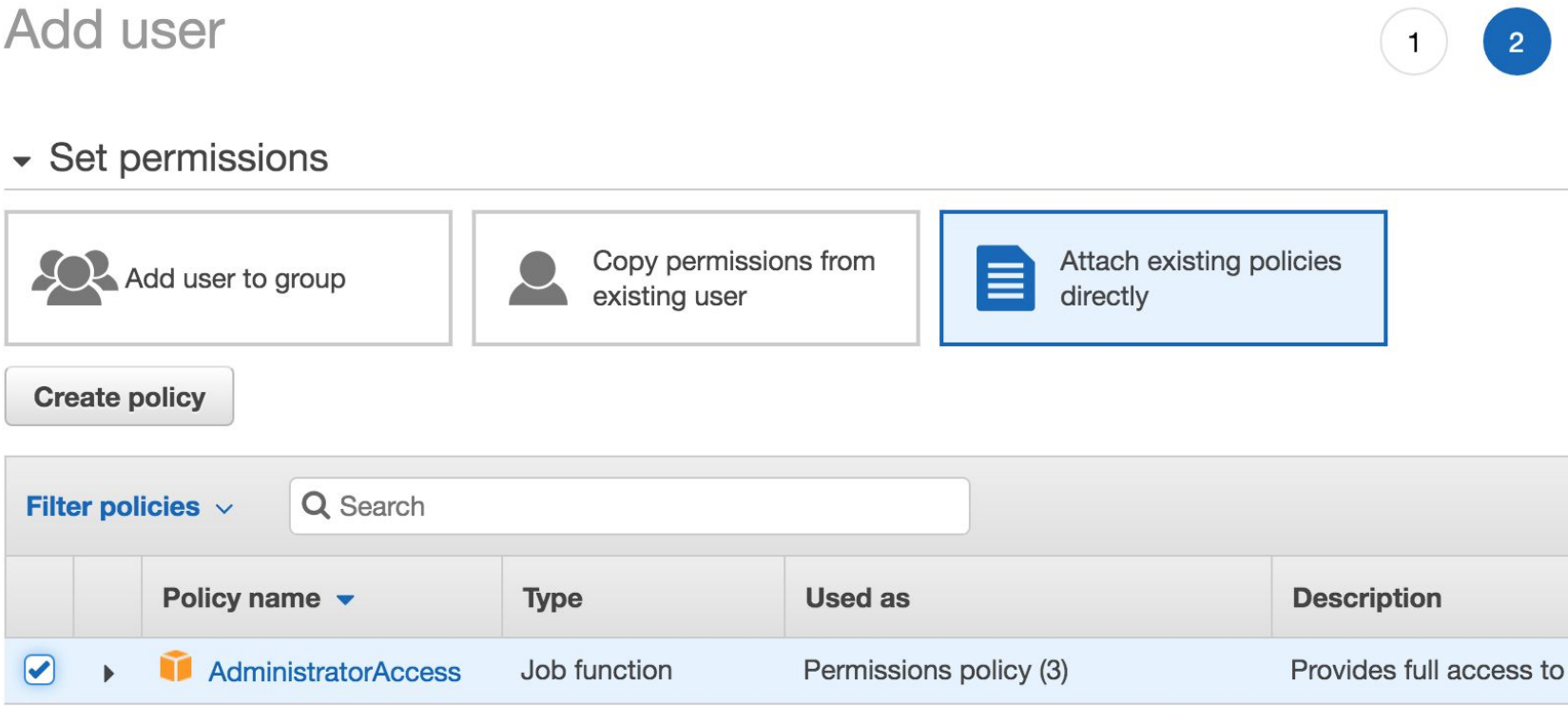
Create IAM Key if You Have No One (1/4)



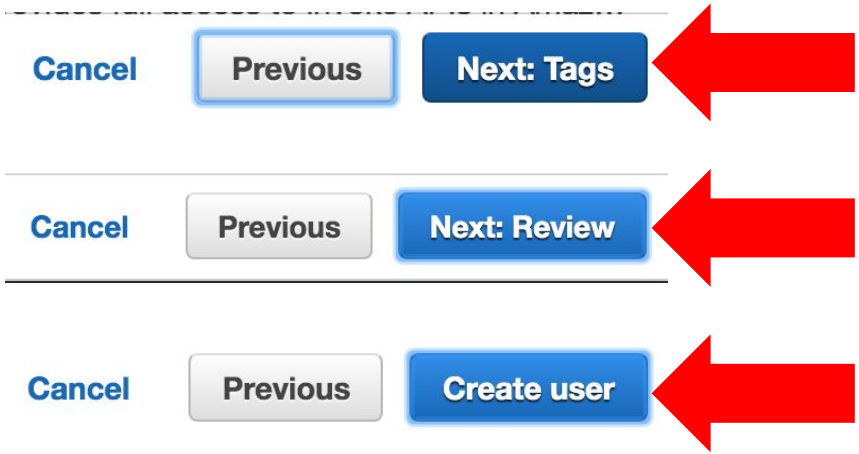
Create IAM Key if You Have No One (2/4)



Create IAM Key if You Have No One (3/4)



Create IAM Key if You Have No One (4/4)



Download .csv

		User	Access key ID	Secret access key
▸	✓	workshop	AKIAR4TOPPAWJYW5EALL	********* Show

```
$(~/environment)
$ git clone https://github.com/Taipei-HUG/eks-workshop.git
$ cd eks-workshop

$(~/environment/eks-workshop)
$ cd setup

$(~/environment/eks-workshop/setup)
$ ./dowload-binary.sh
$ ./eks.sh
```

```
# Attention
# 離開前務必記得要把雲端資源都刪掉
# 離開前務必記得要把雲端資源都刪掉
# 離開前務必記得要把雲端資源都刪掉
$(~/environment/eks-workshop/setup)
$ ./clean-up.sh
$(~/environment/eks-workshop/irsa/terraform)
$ terraform destroy -auto-approve
# 執行完務必要到 aws console 檢查
# 執行完務必要到 aws console 檢查
# 執行完務必要到 aws console 檢查
```

CH01

Kubernetes 101

Kubernetes

- Open Source
  - Container Automation Framework
  - Container Orchestrator
- Open API Based on Google's experiences
- Manage applications, not machines

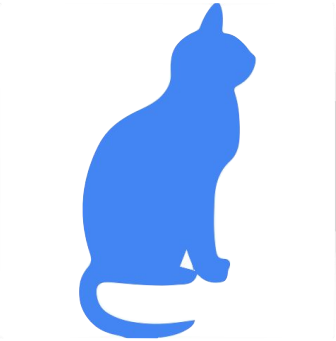
Kuberenetes

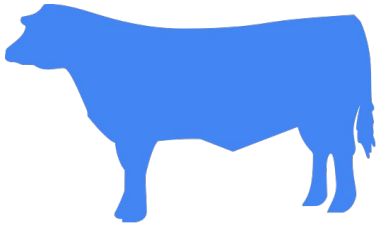
Container management, scheduling, and service discovery.

- API driven application management
- Agents monitor endpoints for state changes
- Controllers enforce desired state
- Labels identify resources

Kubernetes Concept

Pet vs Cattle

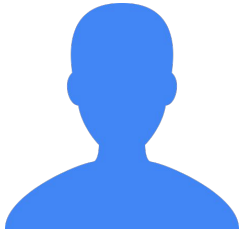




- Has a name
- Is unique or rare
- Personal Attention
- If it gets ill, you make it better

- Has a number
- One is much like any other
- Run as a group
- If it gets ill, you make hamburgers

Children vs Adult



- Go upstairs
- Get undressed
- Put on pajamas
- Brush your teeth
- Pick out 2 stories

- Go get some sleep

Manually States

./create_docker_images.sh

./launch_frontend.sh x 3

./launch_services.sh x 2

./launch_backend.sh x 1

Desire States

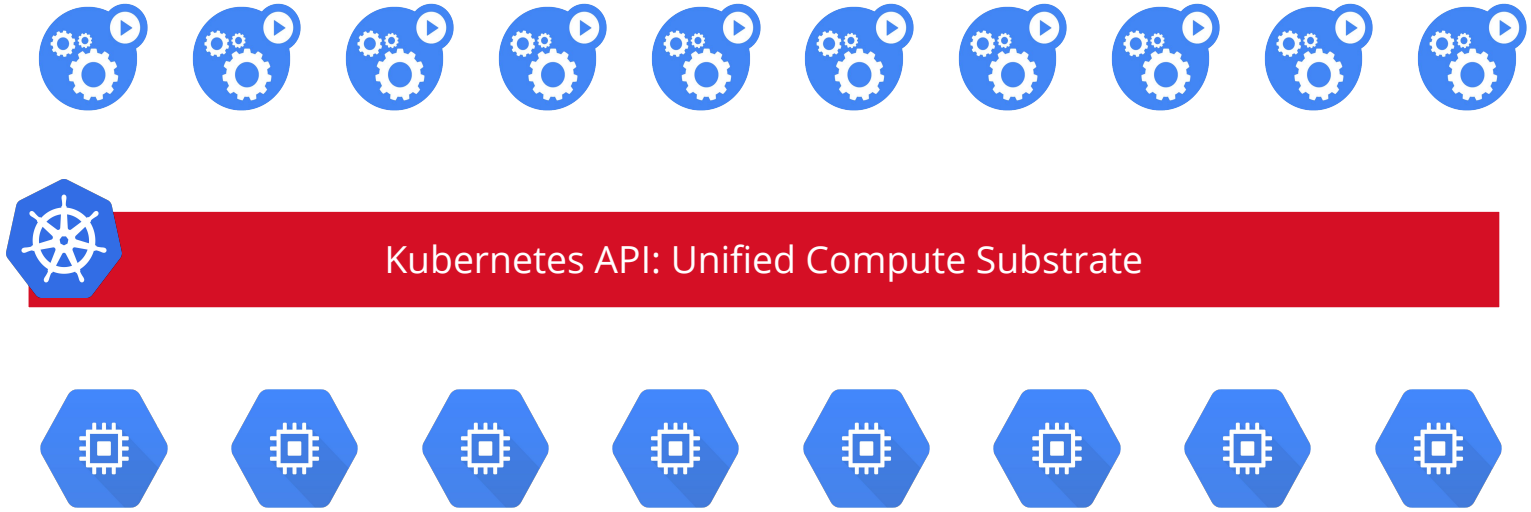
There should be:

3 Frontends

2 Services

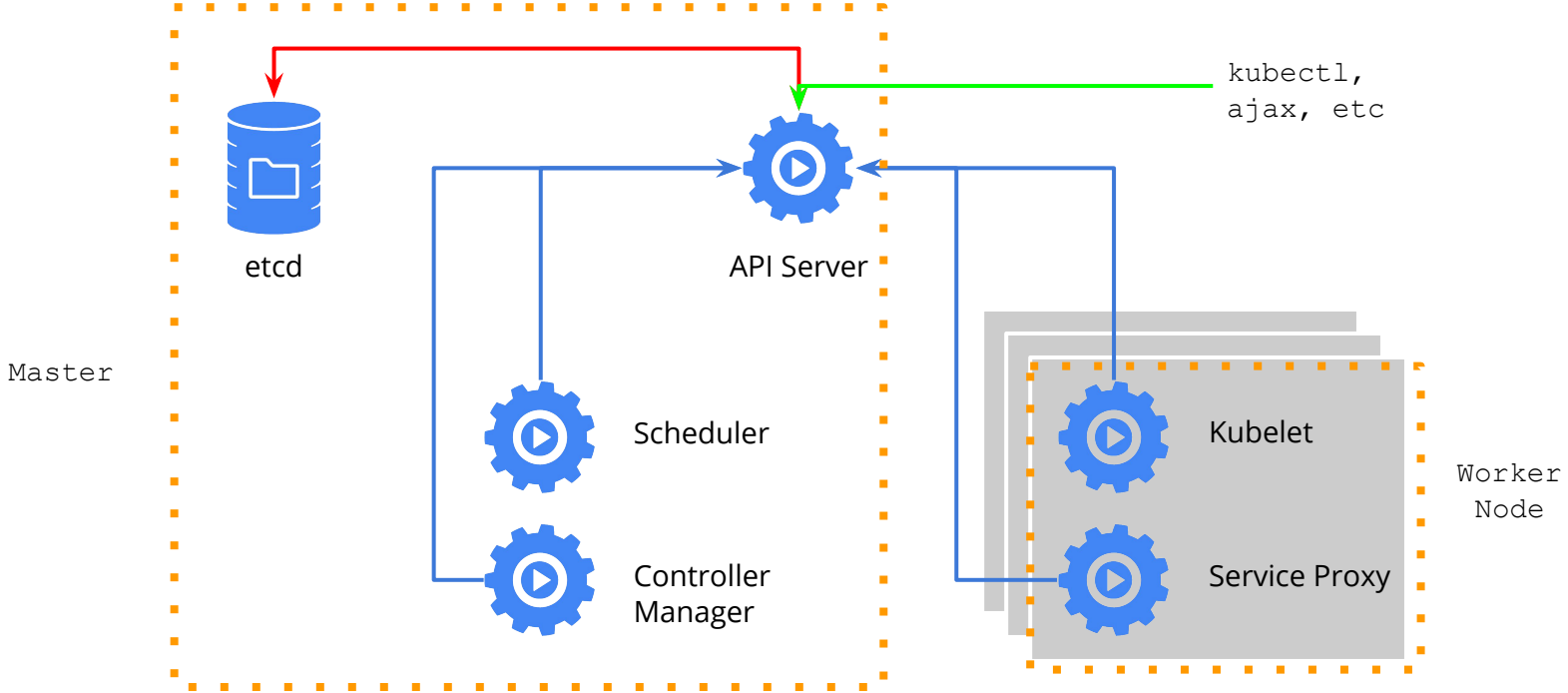
1 Backend

Container Platform



Homogenous Machine Fleet (Virtual or Physical)

Kubernetes Architechture



Kubernetes Resource

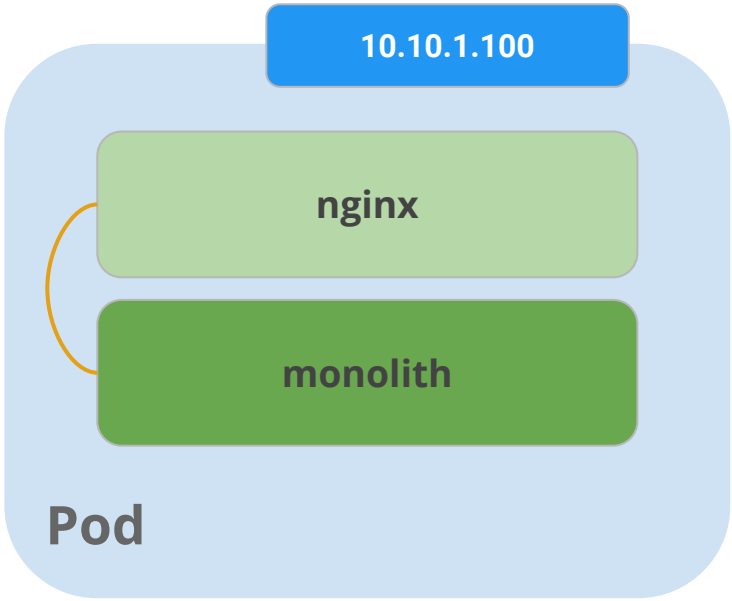
Primary

- Container
- Pod
- Labels
- Selector
- Deployment
- Service
- Ingress
- ServiceAccount

Pod

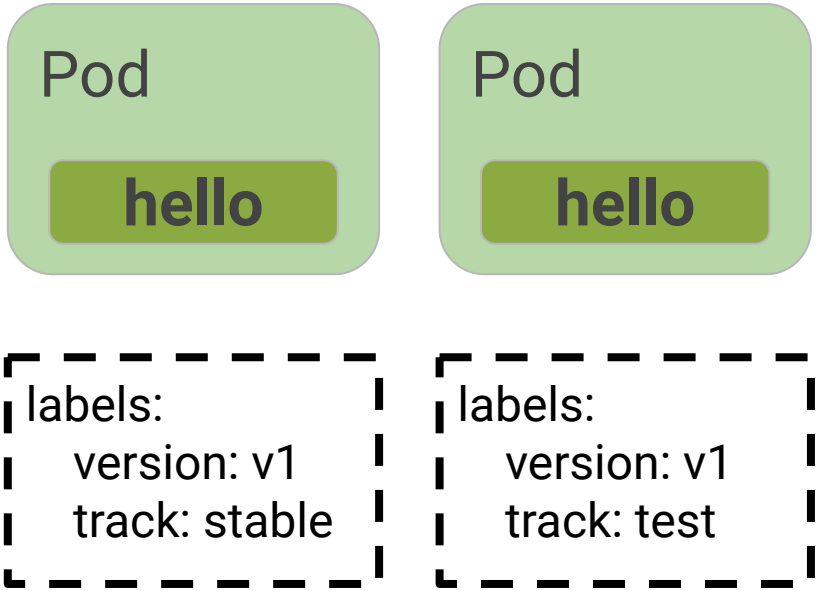
Logical Application

- One or more containers and volumes
- Shared namespaces
- One IP per pod



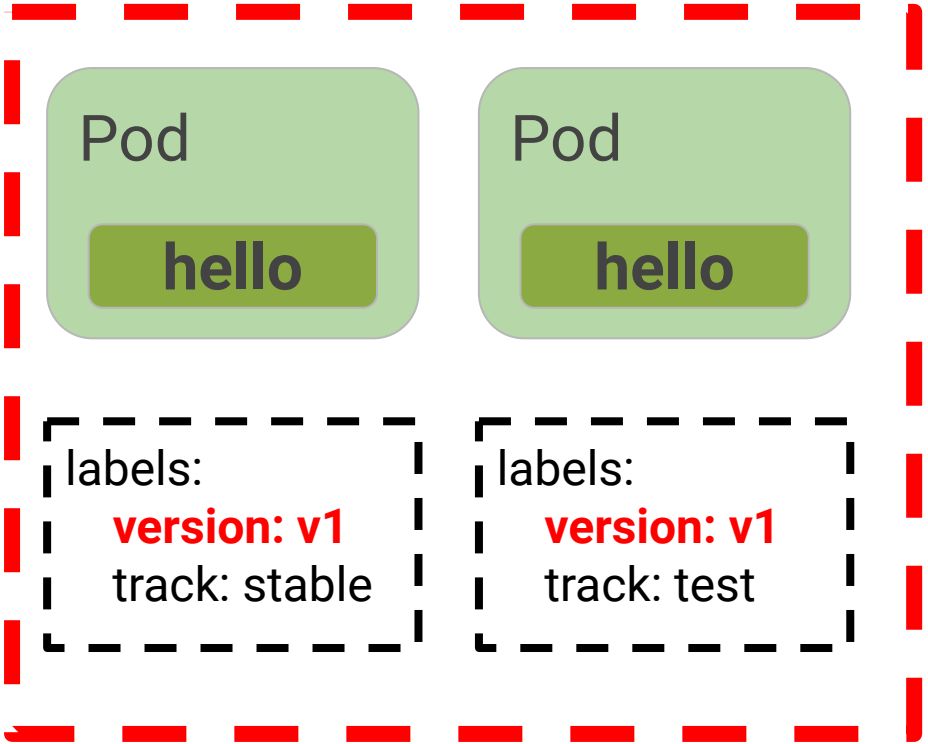
Label

- Arbitrary meta-data attached to Kubernetes object



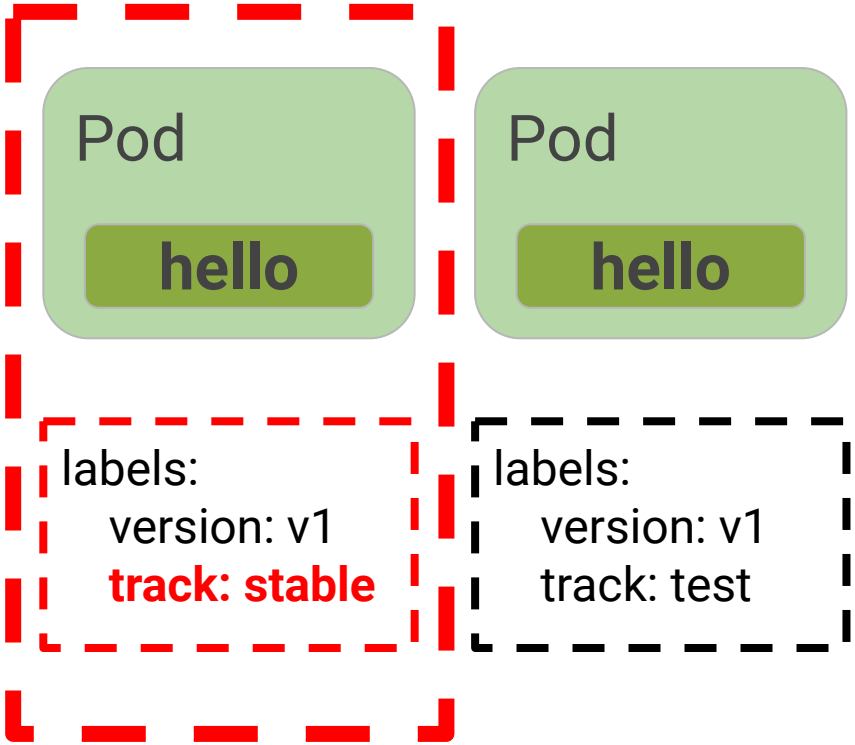
Selector

selector: “version=v1”



Selector

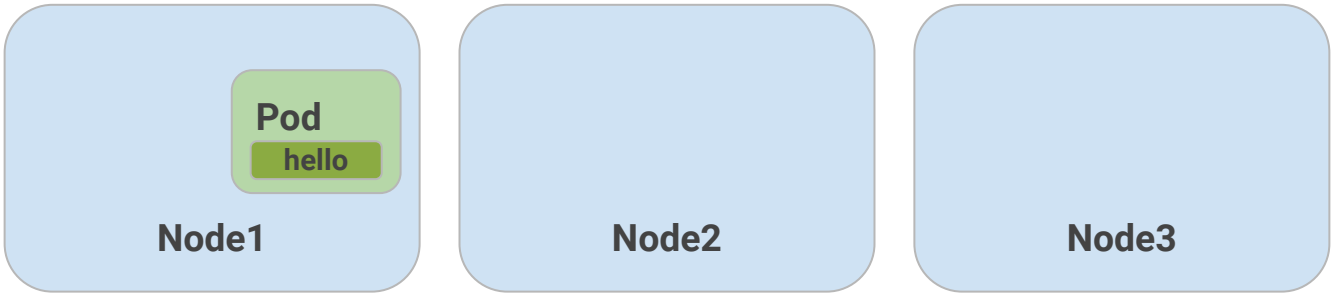
selector: “track=stable”



Deployment

Drive current state towards desired state

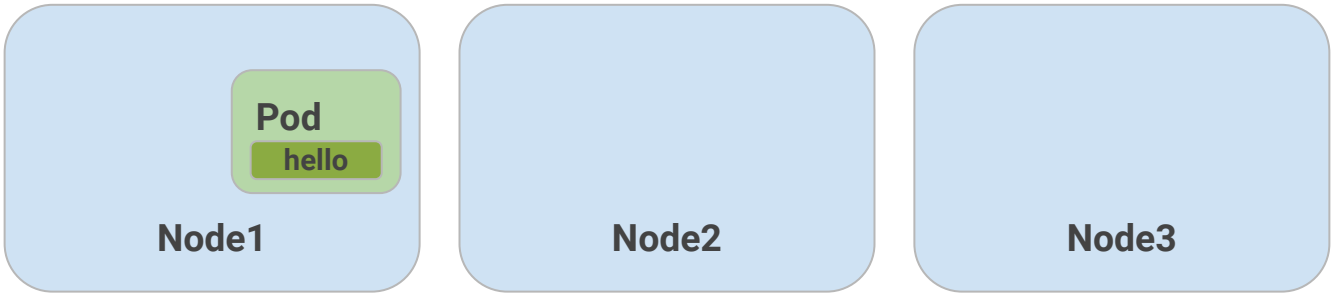
app: hello
replicas: 1



Deployment

Drive current state towards desired state

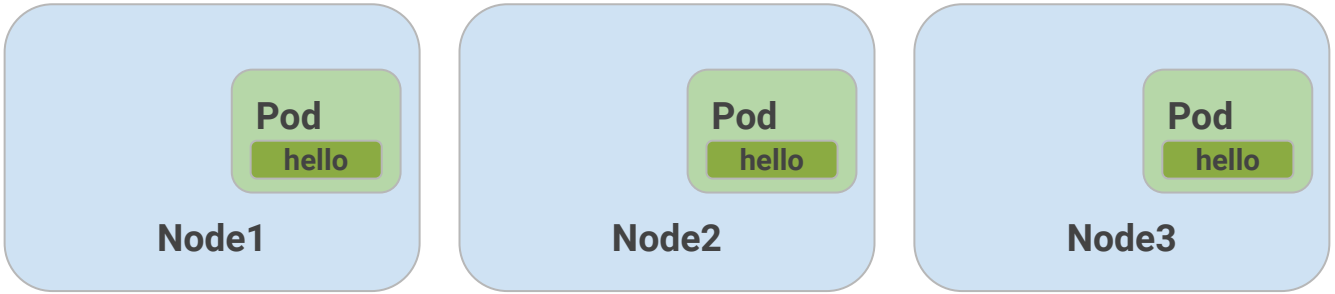
app: hello
replicas: 3



Deployment

Drive current state towards desired state

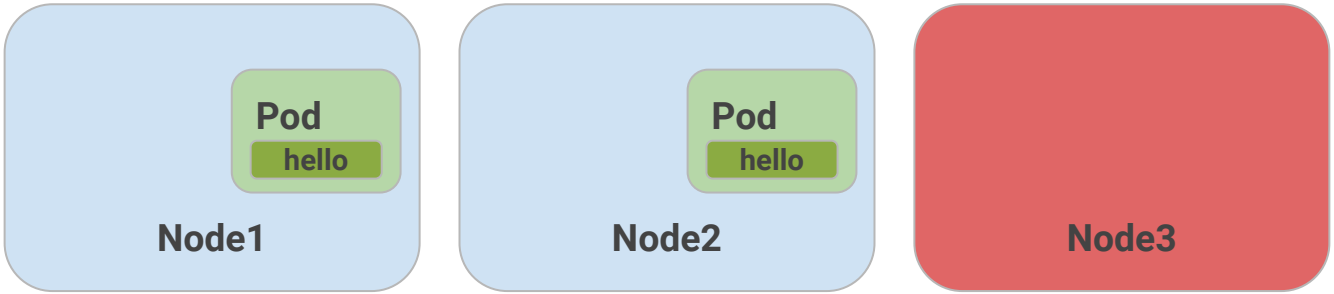
app: hello
replicas: 3



Deployment

Drive current state towards desired state

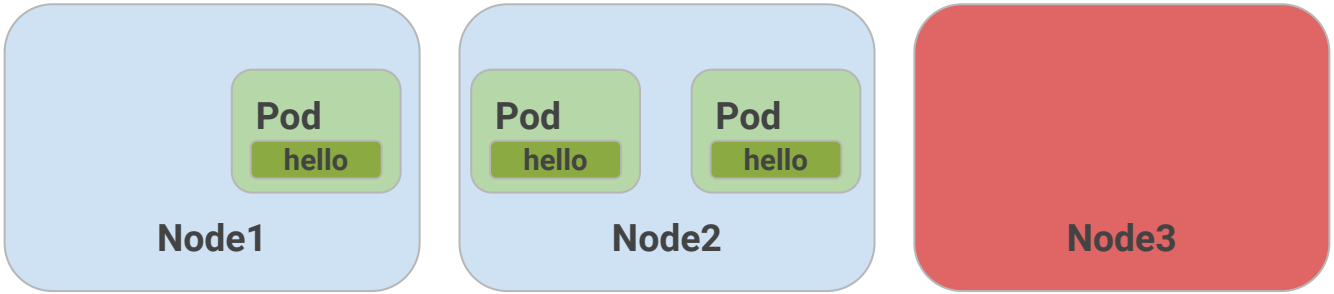
app: hello
replicas: 3



Deployment

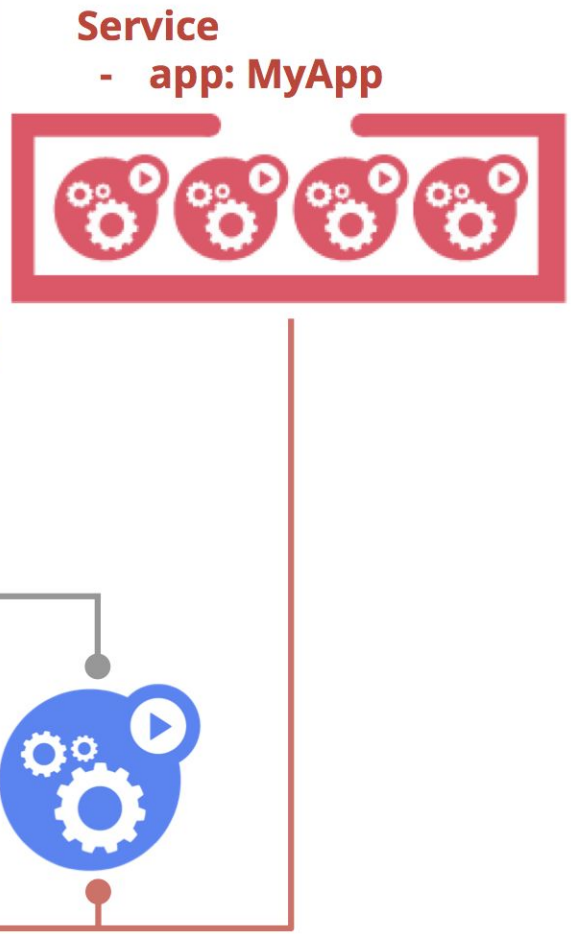
Drive current state towards desired state

app: hello
replicas: 3

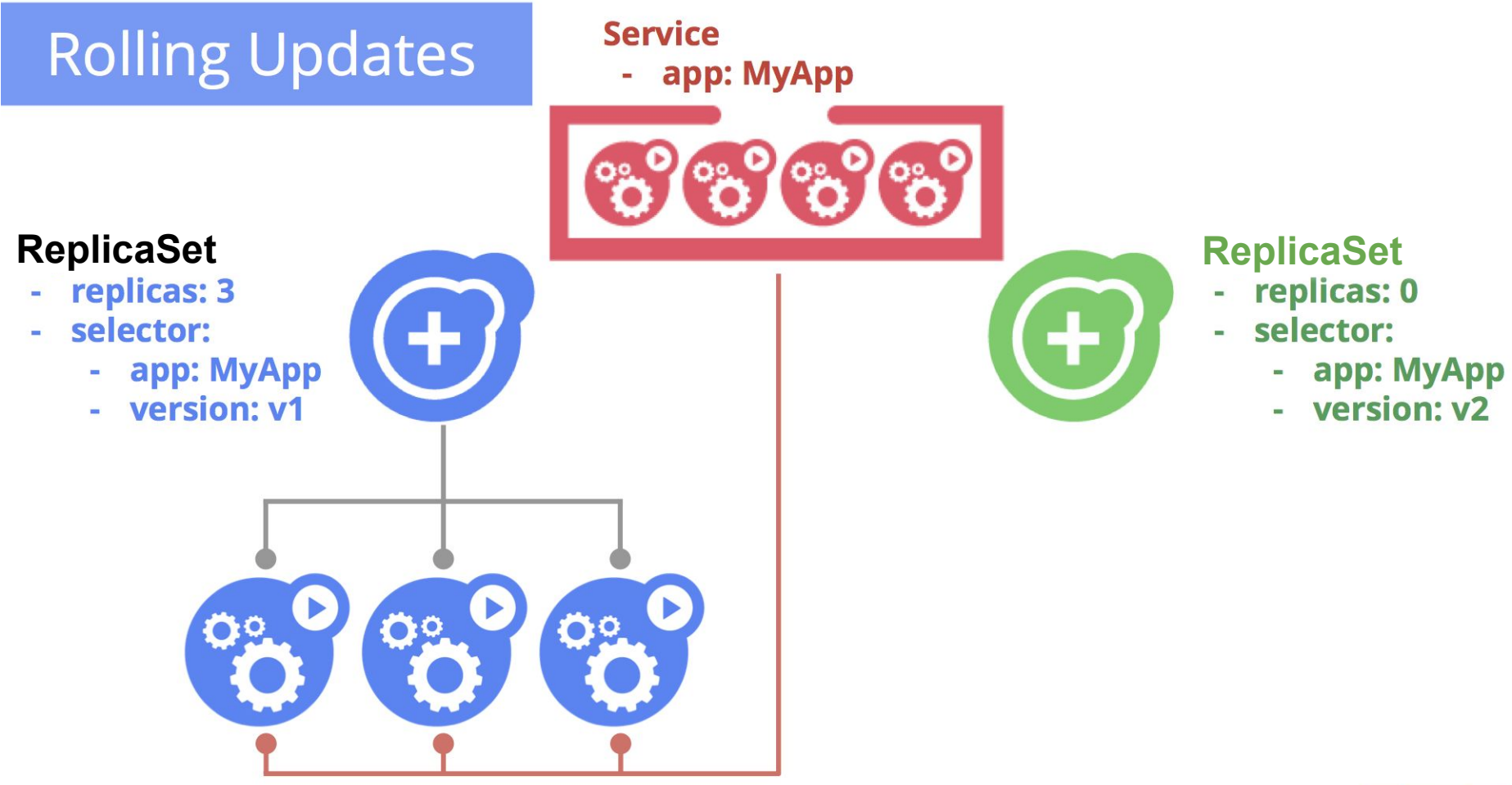


Rolling Update

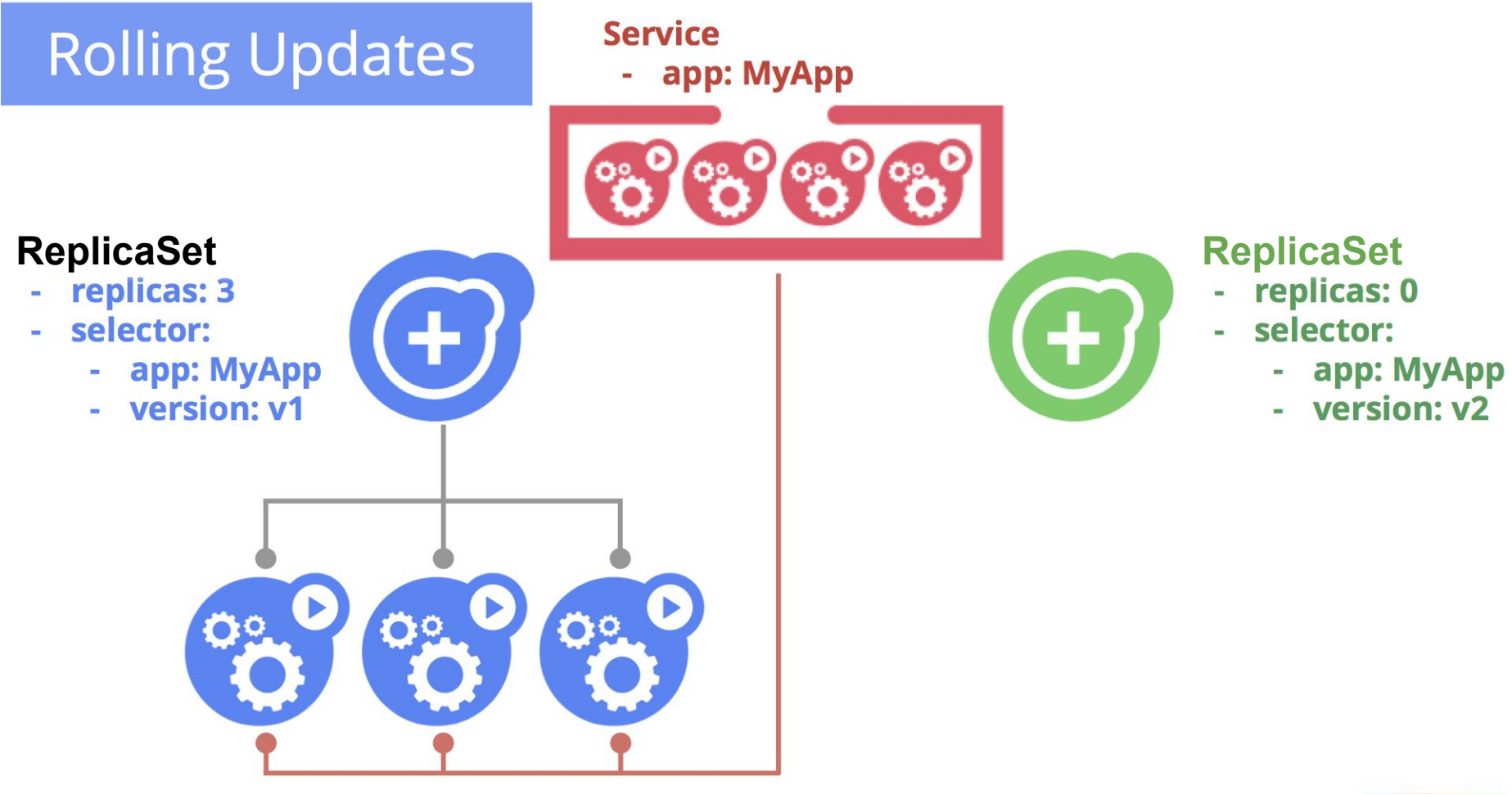
Rolling Updates



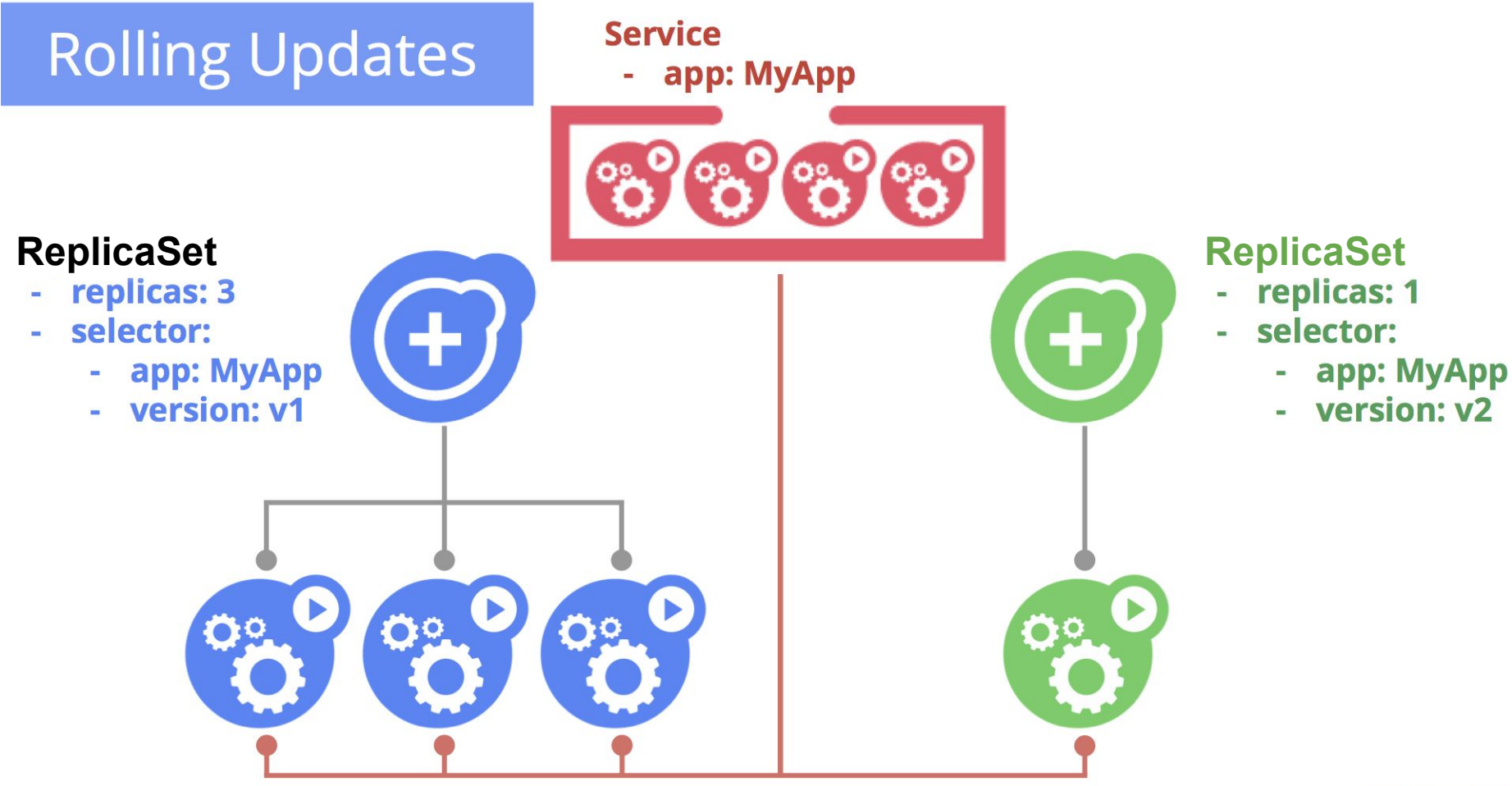
Rolling Updates



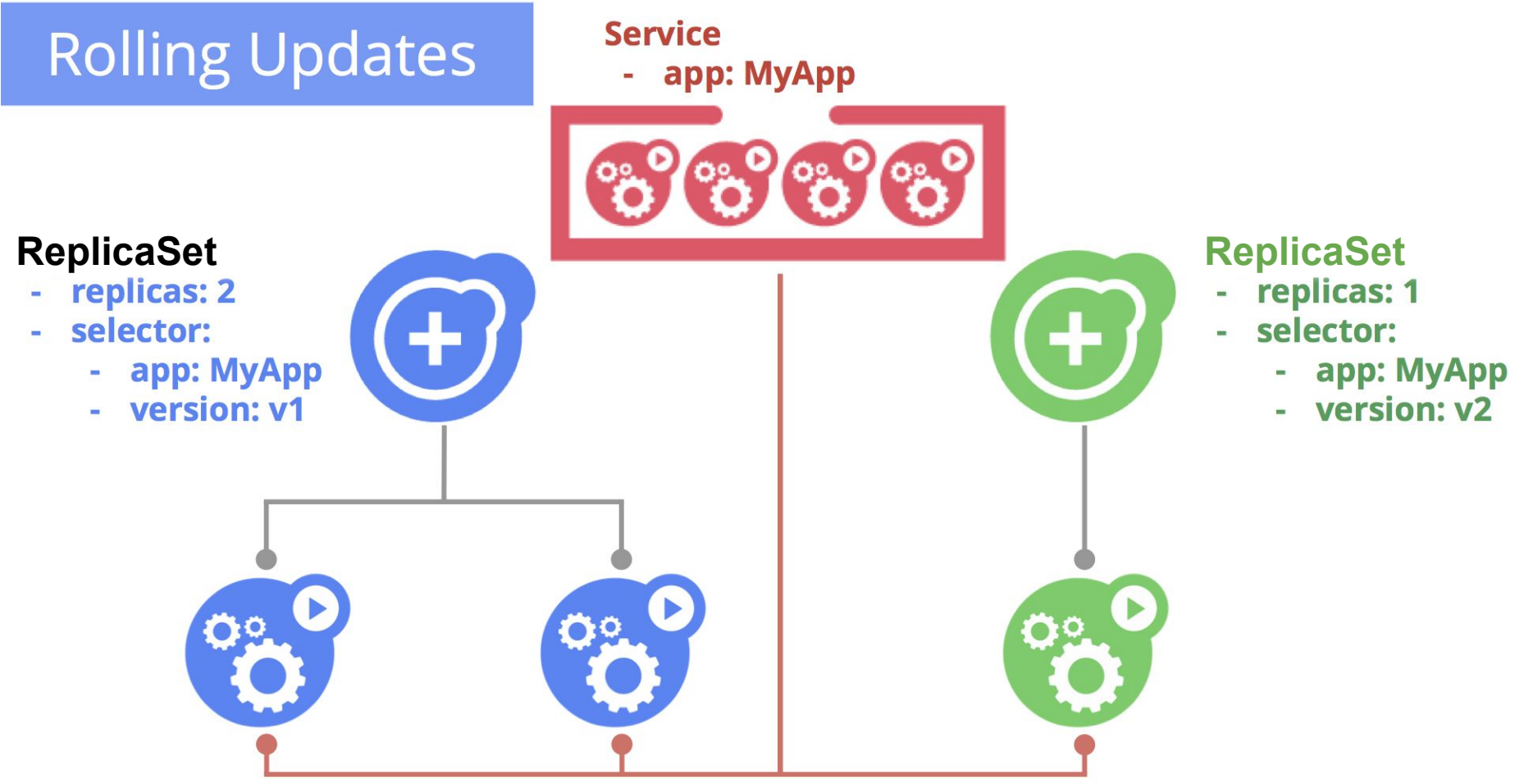
Rolling Updates



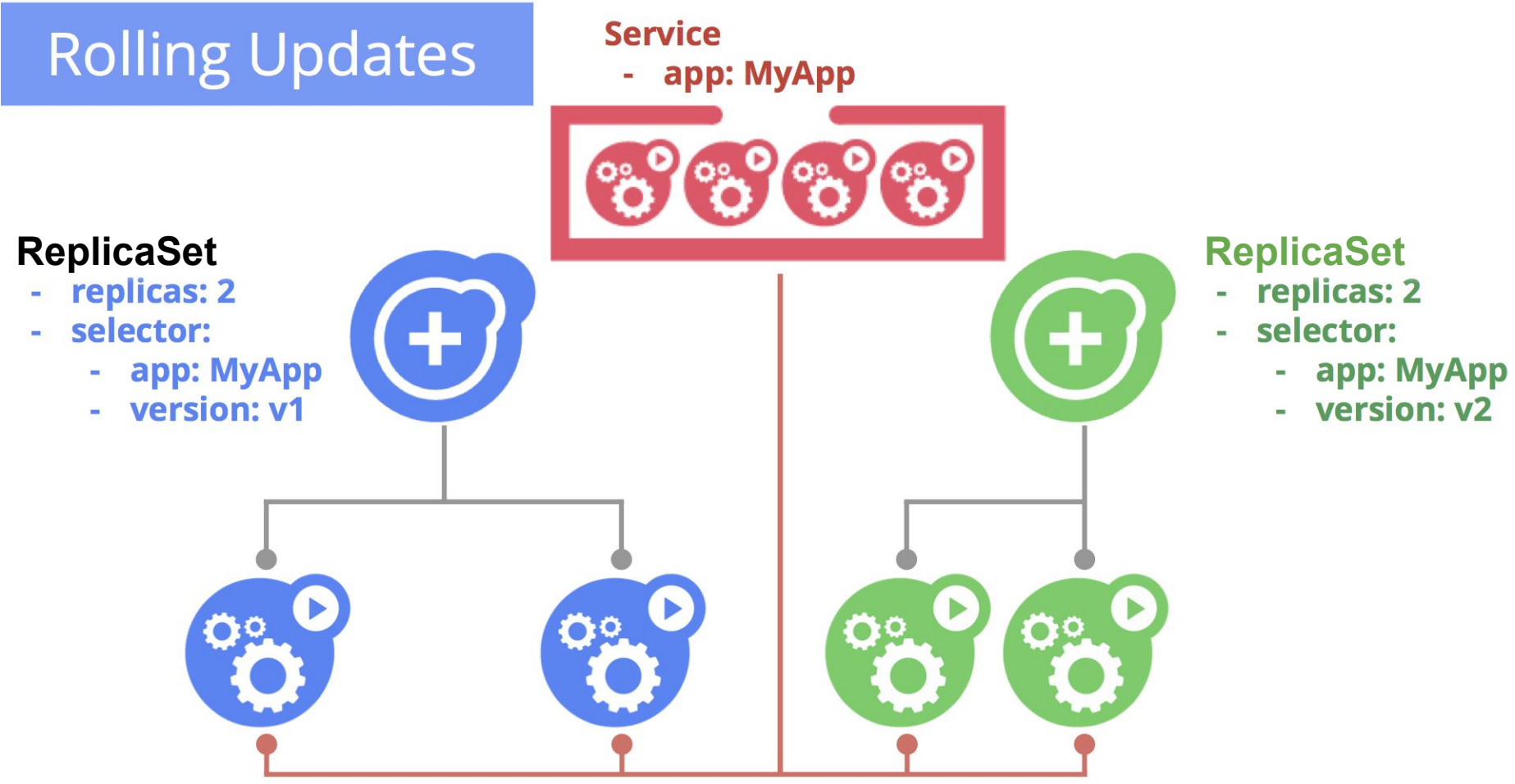
Rolling Updates



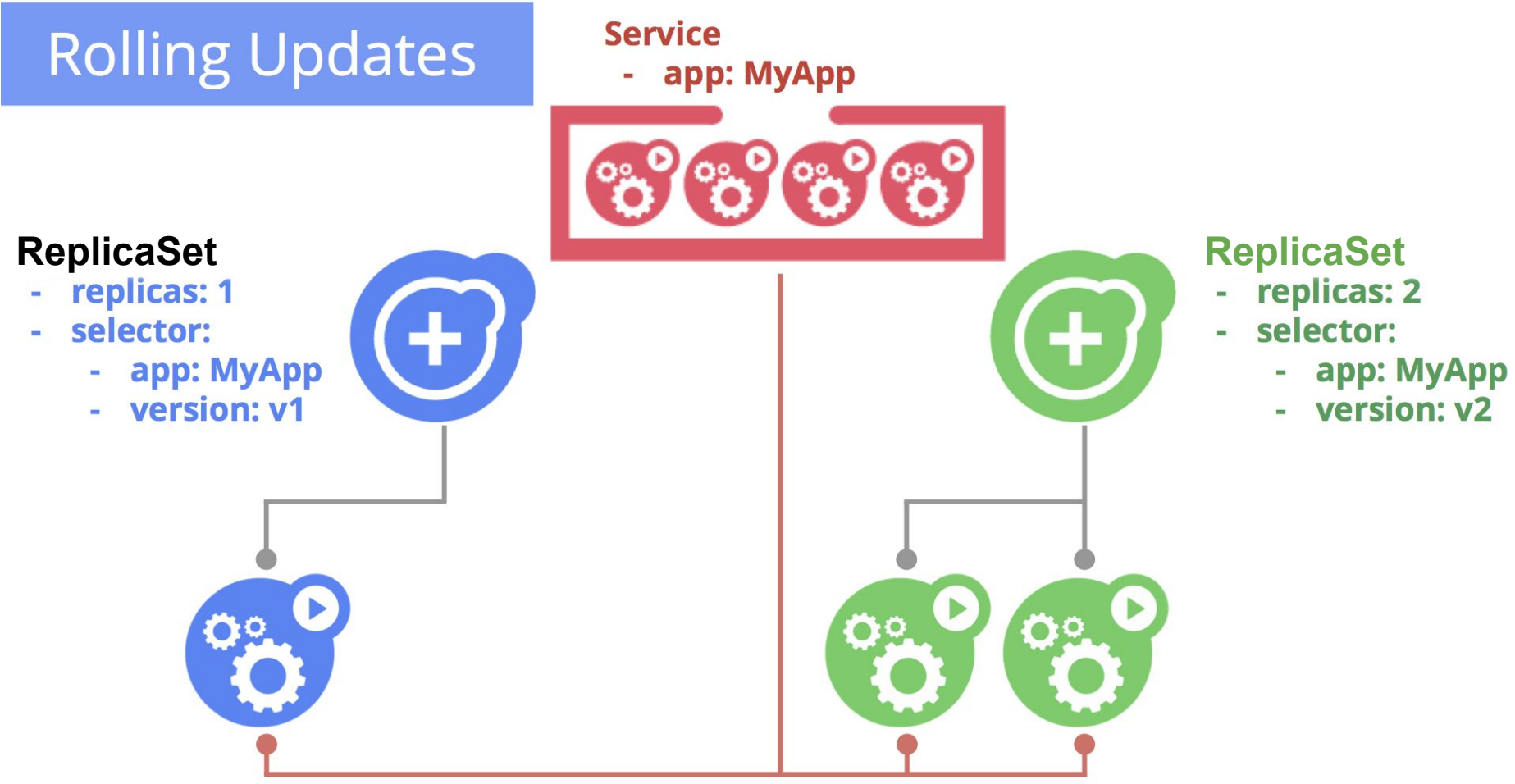
Rolling Updates



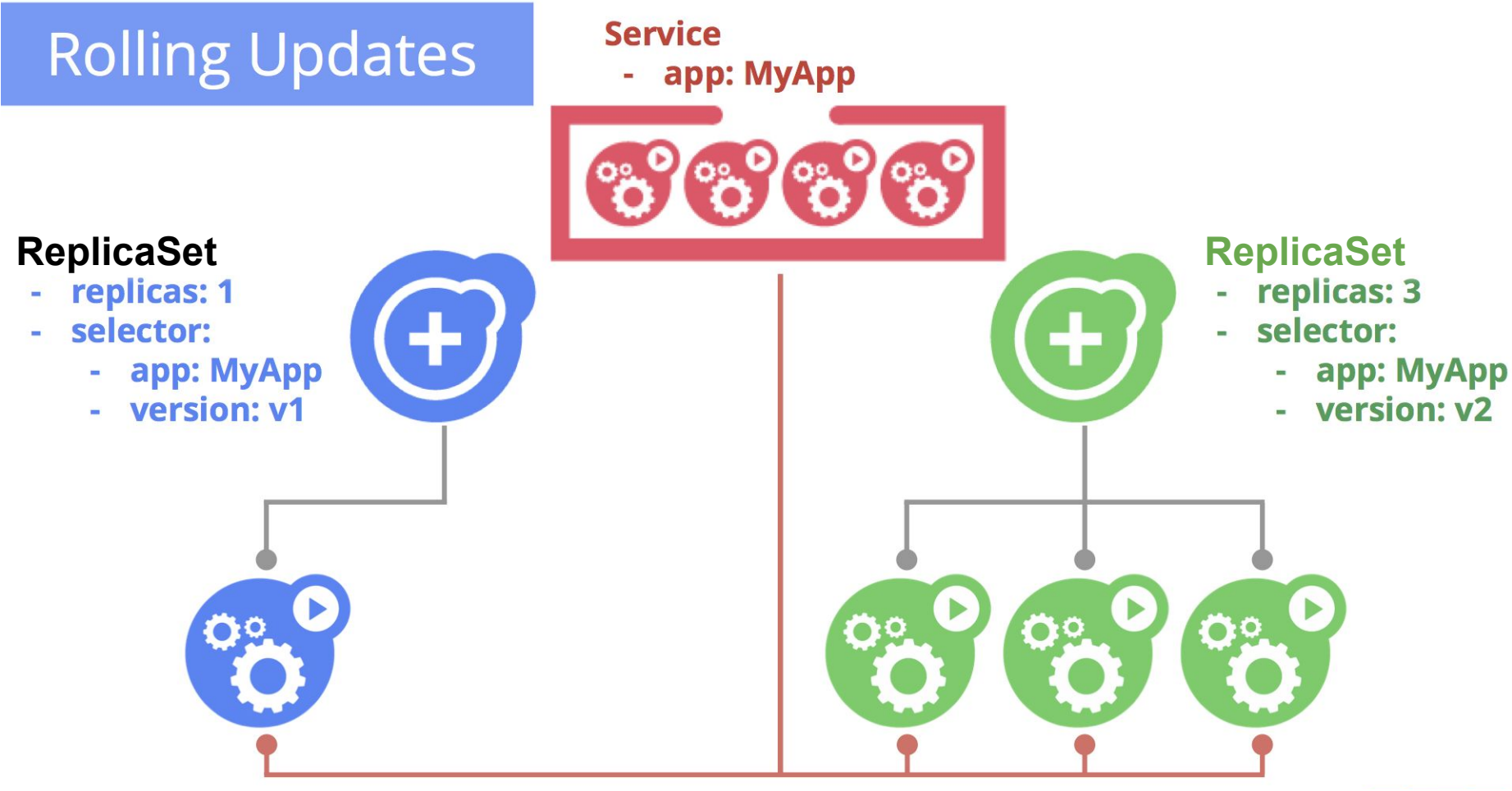
Rolling Updates



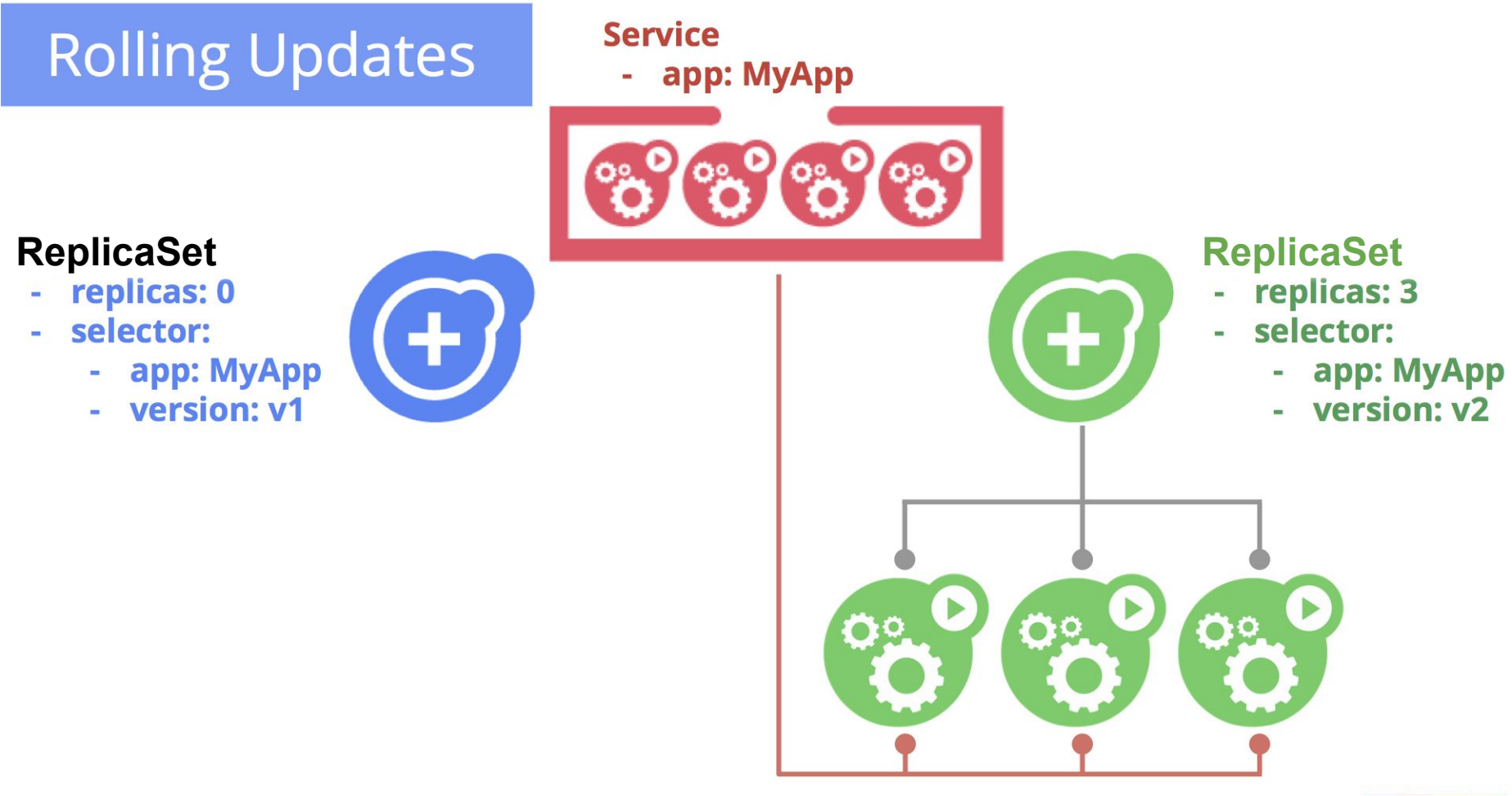
Rolling Updates



Rolling Updates

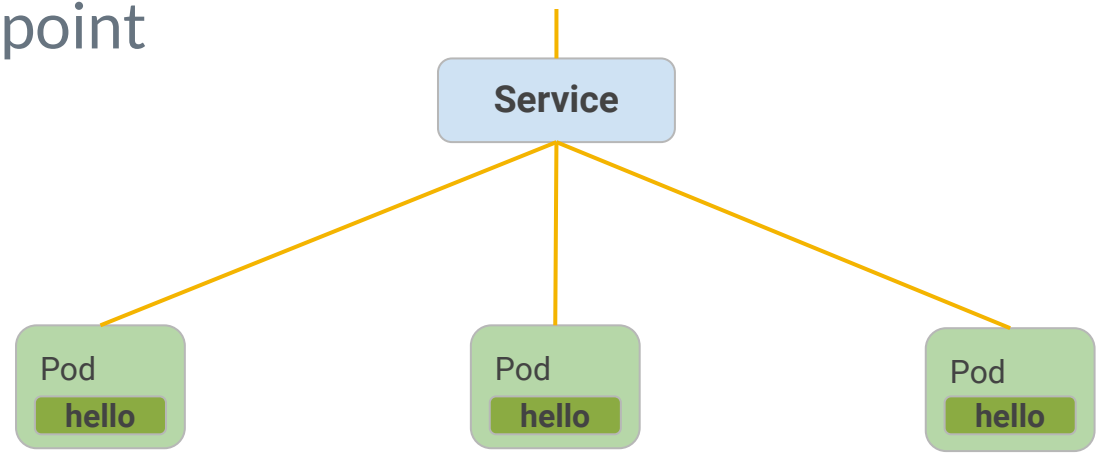


Rolling Updates

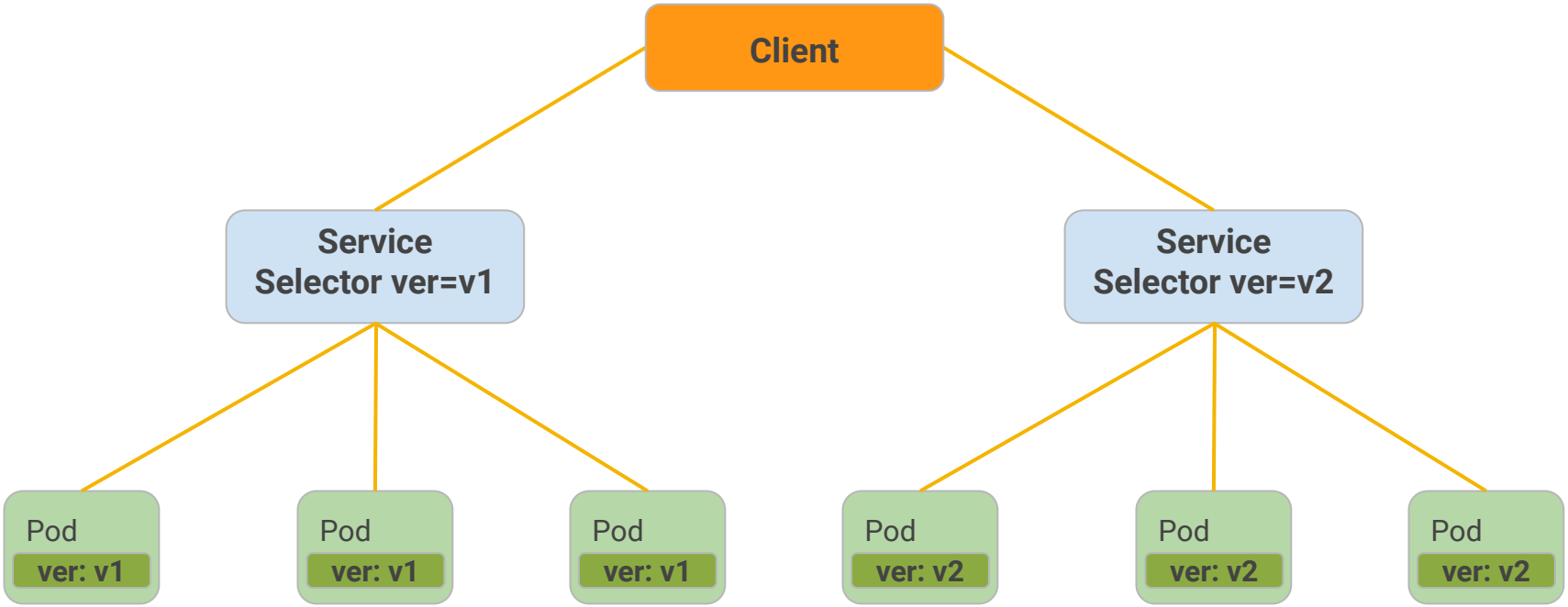


Service

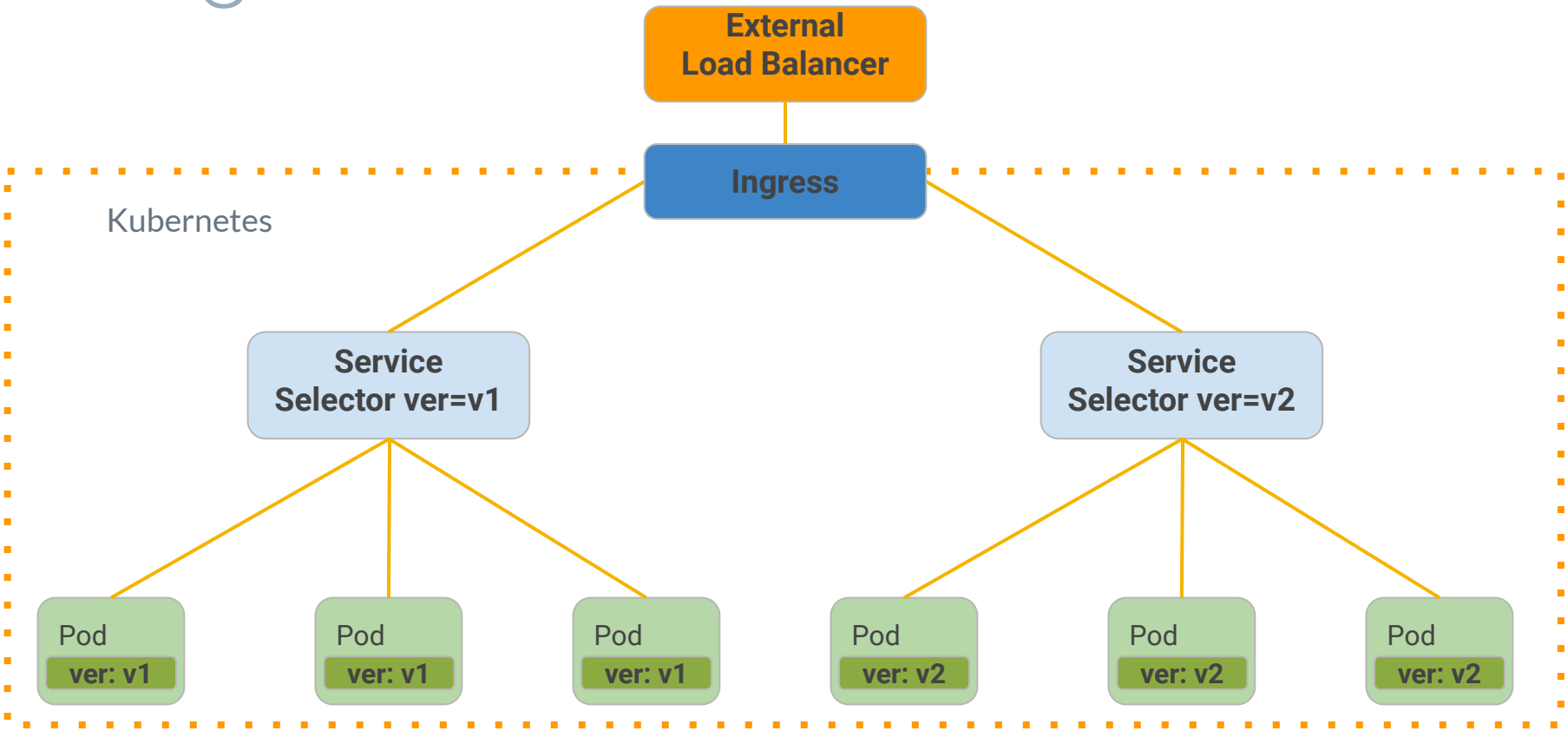
- Persistent Endpoint for Pods
- Use Labels to Select Pods
- Internal or External IP



Service

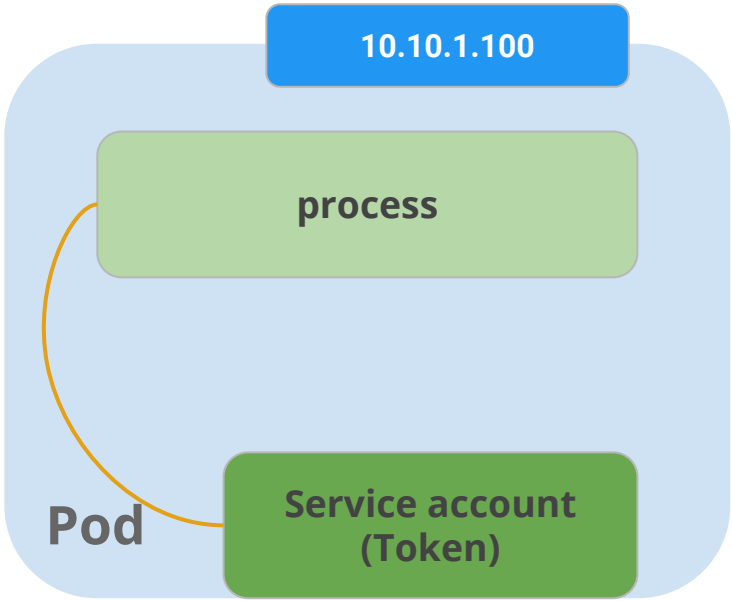

Client
Service
Selector ver=v1
Service
Selector ver=v2
Pod
ver: v1
Pod
ver: v1
Pod
ver: v1
Pod
ver: v2
Pod
ver: v2
Pod
ver: v2

Ingress


External
Load Balancer
Ingress
Kubernetes
Service
Selector ver=v1
Service
Selector ver=v2
Pod
ver: v1
Pod
ver: v1
Pod
ver: v1
Pod
ver: v2
Pod
ver: v2
Pod
ver: v2

Service Account

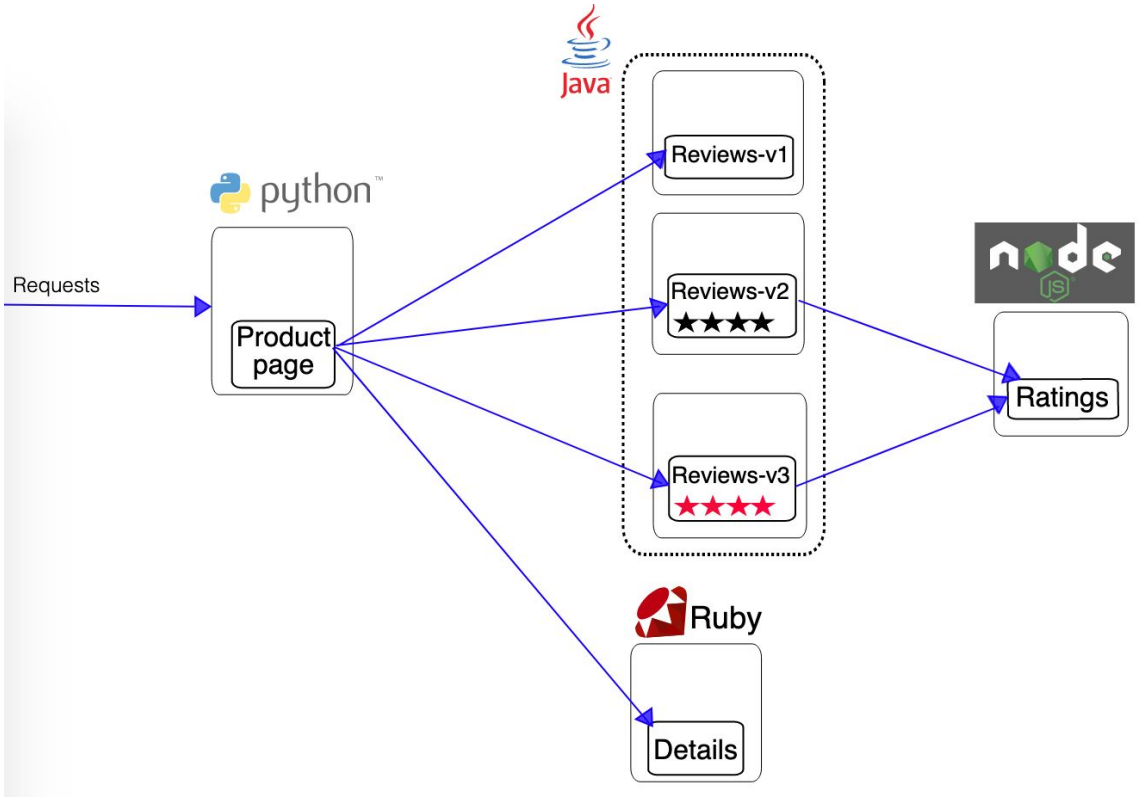
The identity for the process in pod to access resource



```
$(~/environment/eks-workshop)

$ cd setup
$(~/environment/eks-workshop/setup)
$ ./helm.sh
$ ./istio.sh
$ source demo-app.sh

$ cd ../sample
$(~/environment/eks-workshop/sample)
$ skaffold run
```

Demo App

Book Info


Java
python
Requests
Product page
Reviews-v1
Reviews-v2
★★★★
Reviews-v3
★★★★
Ratings
Ruby
Details

```
$(~/environment/eks-workshop)

$ export POD_NAME=$(kubectl get pods --namespace default -l
"app.kubernetes.io/name=ratings,app.kubernetes.io/instance=
ratings" -o jsonpath="{.items[0].metadata.name}")

$ kubectl logs -f ${POD_NAME} -c ratings

$ kubectl port-forward $POD_NAME 8080:9080

$ curl 127.0.0.1:8080/health
```

```
$(~/environment/eks-workshop)

$ export POD_NAME=$(kubectl get pods --namespace default -l
"app.kubernetes.io/name=ratings,app.kubernetes.io/instance=
ratings" -o jsonpath="{.items[0].metadata.name}")

$ kubectl scale deployemnt rating --replicas=2

$ kubectl delete pod ${POD_NAME}
```

CH02

Deploy Application by Helm

Why Helm?

- Kubernetes YAMLs are complex
  - Pods
  - Persistent volumes
  - Configs and secrets
  - Networking
- Templating
  - Development, staging and production
- Packaging
  - Like **apt** and **yum**

Helm v2 Architecture

- **The Helm Client**
  - A command-line client for end users.
- **The Tiller Server**
  - An in-cluster server that interacts with the Helm client, and interfaces with the Kubernetes API server.

What is a Chart?

- A Helm package is called a Chart
- A collection of files that describe a related set of Kubernetes resources

What is a Release?

- When a chart is installed, Tiller (the Helm server) creates a ***release*** to track that installation.
- A single chart may be installed many times into the same cluster, and create many different releases.

Helm Initialization

This will validate that `helm`'s local environment is set up correctly (and set it up if necessary). Then it will install `tiller` into the `kube-system` namespace.

```
$ helm init
```

In case we have to upgrade `tiller`

```
$ helm init --upgrade
```

Sample Charts

```
$ cd eks-workshop/sample/helm-charts
```

```
.
├── bookinfo
├── details
├── productpage
├── ratings
└── reviews
```

Public Charts

You can find various charts in the official chart repository

```
$ helm search
```

You can also input some keywords

```
$ helm search redis
```

Release Management

```
Usage: helm install CHART [flags]
Install from a chart repository
$ helm install stable/redis

or
Install from a local path (--debug prints verbose
logs)
$ helm install ./productpage --debug

You can also dry-run it

$ helm install ./productpage --debug --dry-run
```

Release Management

Show all releases

```
$ helm list --all
```

or

```
$ helm ls -a
```

Show DEPLOYED releases only

```
$ helm list
```

Release Management

```
Usage: helm delete [flags] RELEASE_NAME [...]

To dry-run a deletion

$ helm delete --purge --dry-run my-release


To delete a release

$ helm delete --purge my-release
```

Release Management

To install and name your release
It's recommended to name your releases

```
$ helm install ./productpage --name productpage
```

To install your release into a specific namespace

```
$ kubectl create namespace my-ns
$ helm install ./productpage --name productpage-my-ns
--namespace my-ns
```

Release Upgrading

```
Usage: helm upgrade RELEASE CHART [flags]
```

```
$ helm upgrade productpage ./productpage --set
replicaCount=2
```

```
To see the overridden value
```

```
$ helm upgrade productpage ./productpage --set
replicaCount=3 --debug --dry-run | grep replicas
```

Release Upgrading

The widely used flags in CI/CD

--install installs if the release doesn't exist
--reset-values resets the values to the ones built into the chart
--atomic make the installation process purges chart on fail
--debug enables the verbose output
--values specifies values in a YAML file or a URL(can specify multiple)

```
$ helm upgrade productpage ./productpage \
    --install \
    --reset-values \
    --atomic \
    --debug \
    --values productpage-prod.yaml
```

Release Upgrading

Let's try

```
$ cat << EOF >> productpage-prod.yaml
replicaCount: 3
EOF

$ helm upgrade productpage ./productpage \
    --install \
    --reset-values \
    --atomic \
    --debug \
    --values productpage-prod.yaml
```

Chart Development

- **How to create a chart?**
- How does a chart work?

Creating a new chart

```
Usage: helm create CHART
```

```
$ helm create my-chart
```

The Chart File Structure

```
bookinfo/
  Chart.yaml            # A YAML file containing information about the chart
  LICENSE               # OPTIONAL: A plain text file containing the license for the chart
  README.md             # OPTIONAL: A human-readable README file
  requirements.yaml     # OPTIONAL: A YAML file listing dependencies for the chart
  values.yaml           # The default configuration values for this chart
  charts/               # A directory containing any charts upon which this chart depends.
  templates/            # A directory of templates that, when combined with values,
                        # will generate valid Kubernetes manifest files.
  templates/NOTES.txt   # OPTIONAL: A plain text file containing short usage notes
```

The Chart.yaml

```
apiVersion:  The chart API version, always "v1" (required)
name:        The name of the chart (required)
version:     A SemVer 2 version (required)
kubeVersion: A SemVer range of compatible Kubernetes versions (optional)
description: A single-sentence description of this project (optional)
keywords:
  - A list of keywords about this project (optional)
home: The URL of this project's home page (optional)
sources:
  - A list of URLs to source code for this project (optional)
maintainers: # (optional)
  - name: The maintainer's name (required for each maintainer)
    email: The maintainer's email (optional for each maintainer)
    url: A URL for the maintainer (optional for each maintainer)
```

The Chart.yaml

```
engine: gotpl # The name of the template engine (optional, defaults to gotpl)
icon:       A URL to an SVG or PNG image to be used as an icon (optional).
appVersion: The version of the app that this contains (optional). This needn't be
            SemVer.
deprecated: Whether this chart is deprecated (optional, boolean)
tillerVersion: The version of Tiller that this chart requires. This should be
               expressed as a SemVer range: ">2.0.0" (optional)
```

Chart Dependencies (Subcharts)

- A **requirements.yaml** file is a simple file for listing your dependencies.

```
dependencies:
    - name: productpage
      version: 1.0.0
      repository: https://eks-workshop.com/charts
    - name: details
      version: 1.0.0
      repository: file://../details
    - name: reviews
      version: 1.0.0
      repository: file://../reviews
```

Chart Dependencies Updating

```
Usage: helm dependency build [flags] CHART
```

```
$ helm dependency build ./bookinfo
```

or

```
$ helm dep build ./bookinfo
```

Install a Chart with subcharts

Let's try

```
$ helm install --name bookinfo ./bookinfo
```

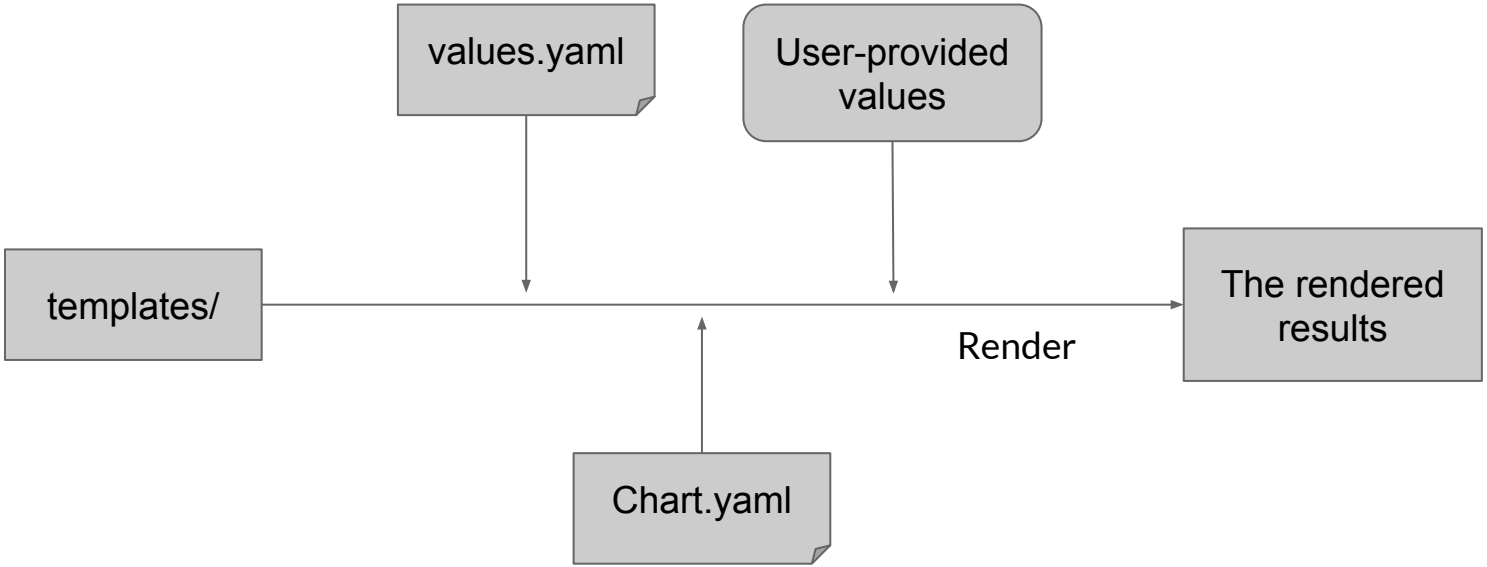
Chart Development

- How to create a chart?
- **How does a chart work?**

Chart Rendering

- When evaluating a chart, Tiller will render all of the files in the **templates/** directory.
- Tiller then collects the results of those templates and sends them on to Kubernetes.

Chart Rendering



Kubernetes Service Template

```
apiVersion: v1
kind: Service
metadata:
  name: {{ include "productpage.fullname" . }}  # templates in _helpers.tpl
  labels:
{{ include "productpage.labels" . | indent 4 }} # templates in _helpers.tpl
spec:
  type: {{ .Values.service.type }}              # In values.yaml
  ports:
    - port: {{ .Values.service.port }}          # In values.yaml
      targetPort: http
      protocol: TCP
      name: http
  selector:
    app.kubernetes.io/name: {{ include "productpage.name" . }} # templates in _helpers.tpl
    app.kubernetes.io/instance: {{ .Release.Name }}            # From the release properties
```

Built-in Objects

- Release
  - This object describes the release itself. It has several objects inside of it.
- Values
  - Values passed into the template from the **values.yaml** file and from user-supplied files.
- Chart
  - The content of **Chart.yaml**.
- Files
  - This provides access to all non-special files in a chart.
- Capabilities
  - This provides information about what capabilities the Kubernetes cluster supports.
- Template
  - Contains information about the current template that is being executed.

Doc: https://helm.sh/docs/chart_template_guide/#built-in-objects

Kubernetes Service Template

- How did Helm process this kind of values?
  - These are **Named Templates**.
  - See **templates/_helpers.tpl**

```
metadata:
  name: {{ include "productpage.fullname" . }}
  labels:
{{ include "productpage.labels" . | indent 4 }}
```

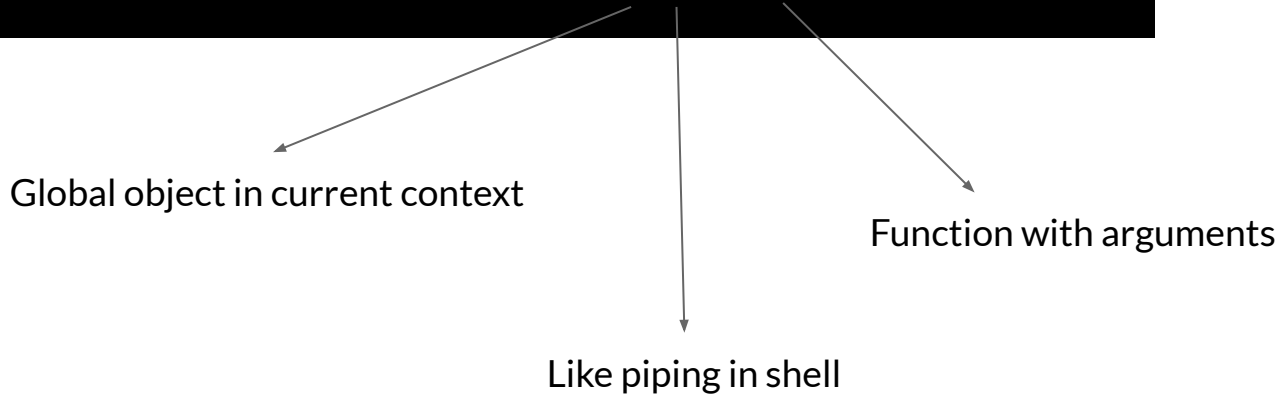
Doc: https://helm.sh/docs/chart_template_guide/#named-templates

```
{{/*
Create a default fully qualified app name.
We truncate at 63 chars because some Kubernetes name fields are limited to this (by the DNS naming spec).
If release name contains chart name it will be used as a full name.
*/}}
{{- define "productpage.fullname" -}}
{{- if .Values.fullnameOverride -}}
{{- .Values.fullnameOverride | trunc 63 | trimSuffix "-" -}}
{{- else -}}
{{- $name := default .Chart.Name .Values.nameOverride -}}
{{- if contains $name .Release.Name -}}
{{- .Release.Name | trunc 63 | trimSuffix "-" -}}
{{- else -}}
{{- printf "%s-%s" .Release.Name $name | trunc 63 | trimSuffix "-" -}}
{{- end -}}
{{- end -}}
{{- end -}}
```

```
{{/*
Common labels
*/}}
{{- define "productpage.labels" -}}
app.kubernetes.io/name: {{ include "productpage.name" . }}
helm.sh/chart: {{ include "productpage.chart" . }}
app.kubernetes.io/instance: {{ .Release.Name }}
{{- if .Chart.AppVersion }}
app.kubernetes.io/version: {{ .Chart.AppVersion | quote }}
{{- end }}
app.kubernetes.io/managed-by: {{ .Release.Service }}
{{- end -}}
```

Kubernetes Service Template

```
metadata:
  name: {{ include "productpage.fullname" . }}
  labels:
{{ include "productpage.labels" . | indent 4 }}
```



Kubernetes Service Template

Let's try to render service.yaml in the productpage chart

```
$ helm template ./productpage -x templates/service.yaml
```

Let's override some values and try again

```
$ helm template ./productpage -x templates/service.yaml
--name my-release --set service.port=1234
```

What did you see?

Controlling subcharts

We can also render a single file in subcharts

```
$ helm template bookinfo -x
charts/productpage/templates/service.yaml
```

Controlling subcharts

Passing values to subcharts

```
$ helm template bookinfo -x
charts/productpage/templates/deployment.yaml \
--set productpage.replicaCount=5 | grep replicas
```

Prefix the value with the subchart name

Doc: https://helm.sh/docs/chart_template_guide/#overriding-values-of-a-child-chart

Controlling subcharts

Let's try

```
$ helm upgrade bookinfo ./bookinfo --install
--reset-values --atomic \
--set productpage.replicaCount=2 \
--set reviews.replicaCount=2 \
--set details.replicaCount=2
```

Now get the pods

```
$ kubectl get pods
```

Did you see it?

Helm v3

- Almost ready for release (R.C.)
- Major changes
  - `tiller` is gone
  - `helm init` is gone
  - `requirements.yaml` is merged into `Chart.yaml`
  - Chart types
    - **library** chart: provides utilities or functions
    - **application** chart: standard chart
  - More CRDs friendly

CH03

Accelerate Development by Skaffold

Kubernetes is great, but ……

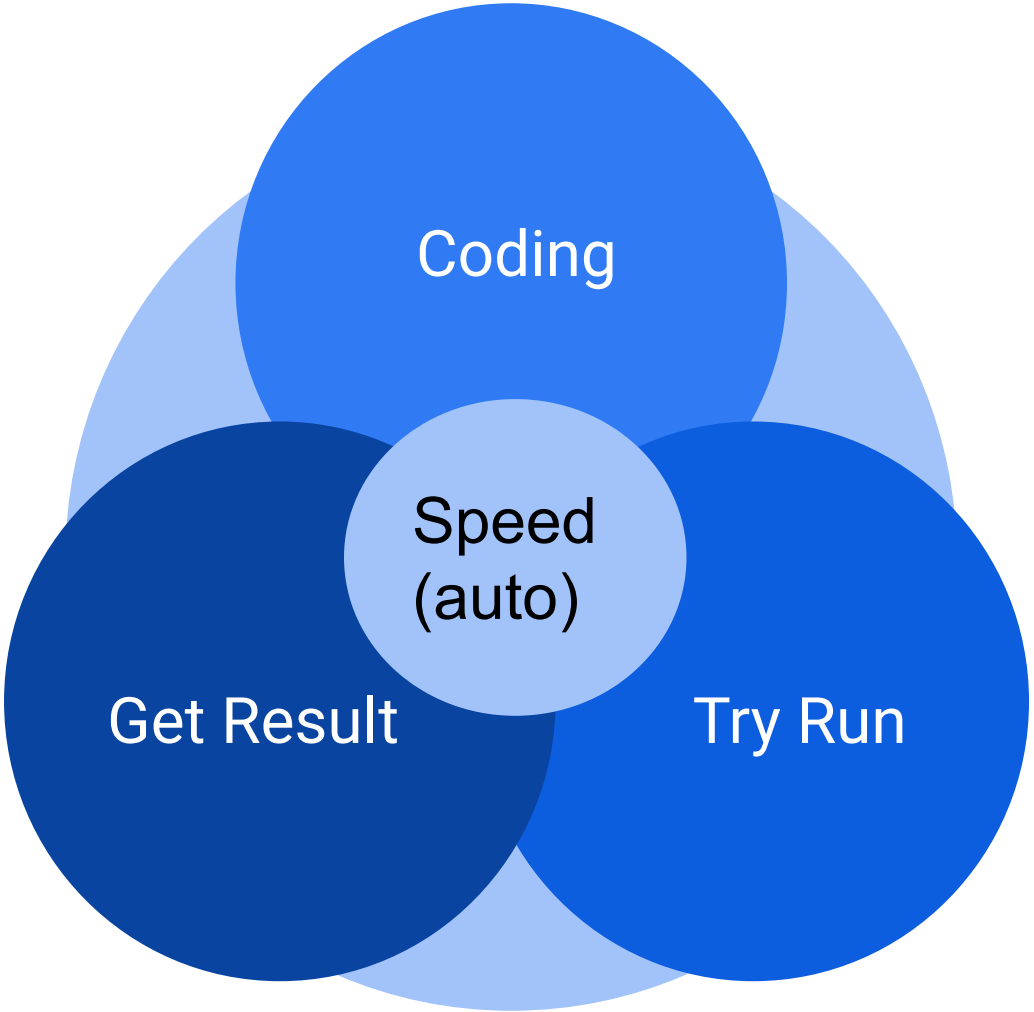
- A Whole New World for developer
- Why Does Developing on Kubernetes Suck?
- General Developing Workflow
  - Developing
  - Run
  - Get Result

Steps In Kubernetes

- Coding
- Build Image
- Push Image
- Deploy the image into Kubernetes
  - Helm
- Get Result in
  - Observation
  - Debugging

Steps In Kubernetes

```
$ #coding
$ docker build -t ${image}:${tag}
$ docker push ${image}:${tag}
$ helm upagrade --install ${release} ${chart}
$ kubectl get po
$ kubectl logs -f ${podName}
$ kubectl exec -it ${podName} sh
…….
```

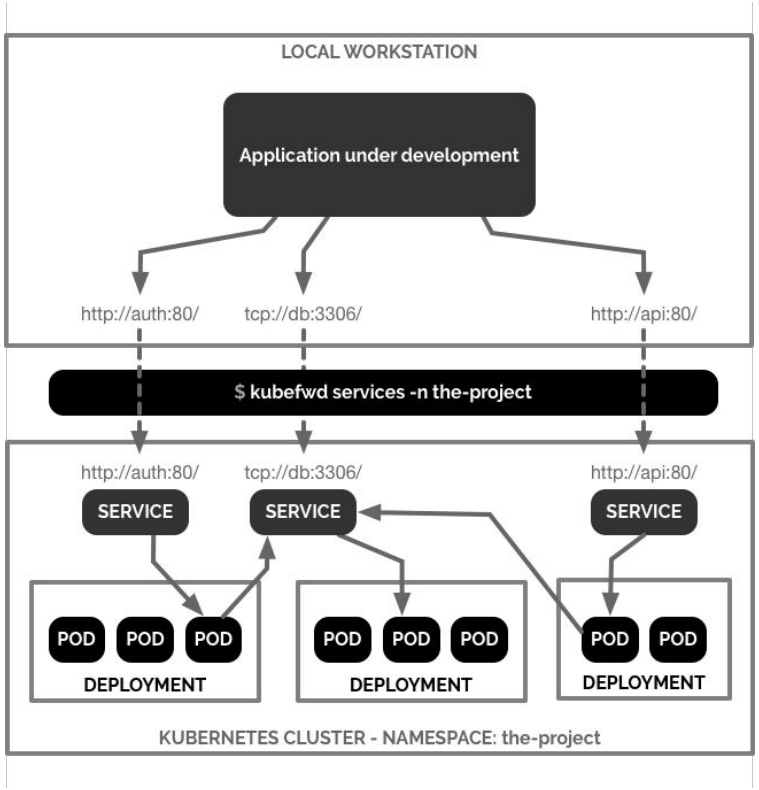

Coding
Speed
(auto)
Get Result
Try Run

Observation & Debugging

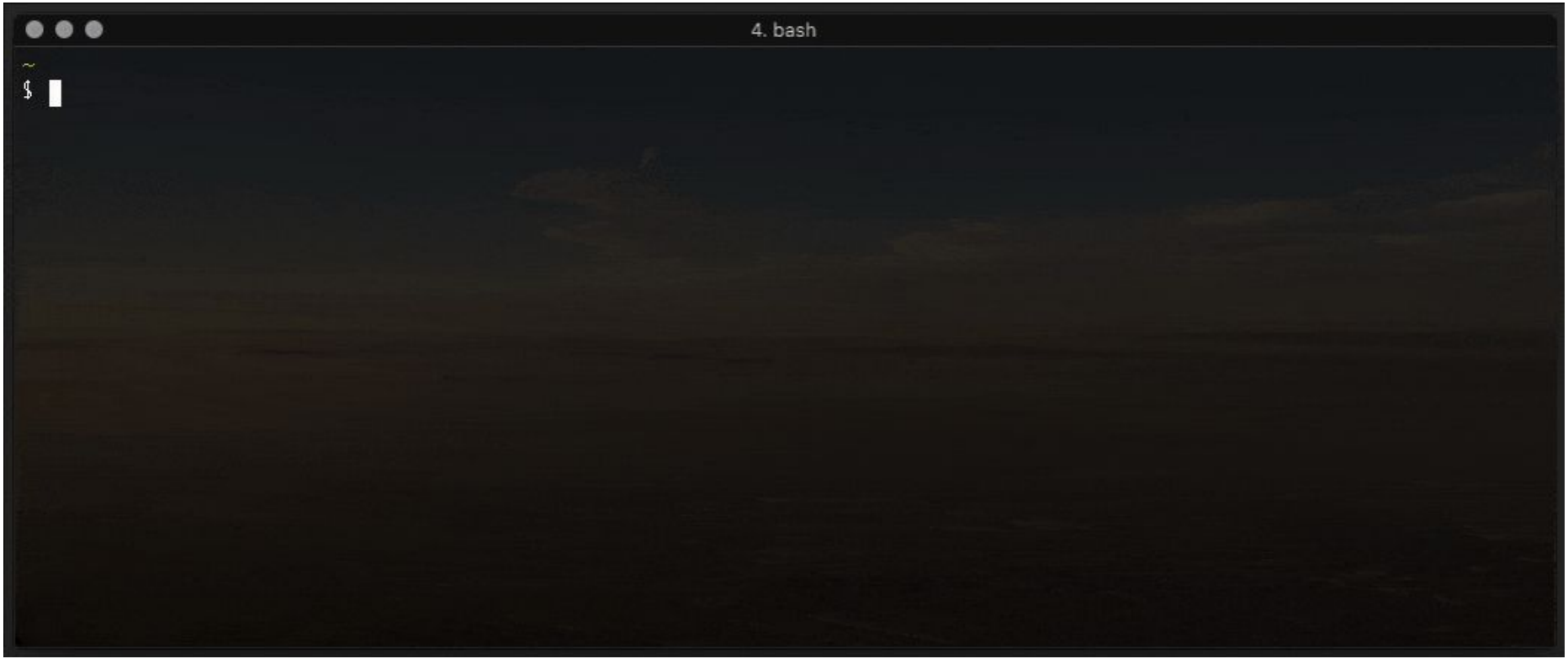
- Access component in Kubernetes
  - kubectl port-forward
  - kubectl exec
  - kubectl log
  - resource usage ……
  - state changes

kubefwd

- kubectl port-forward one command for on pod/sevice only
- kubefwd
  - Automatically port-forward all service
  - Modify host /etc/hosts
  - Access component



kubefwd



```
$(~/environment/eks-workshop)
$ sudo -E kubefwd services
```

kubebox

- A Terminal UI for escaping from kubectl
  - Switch Pod & Container with arrow
  - See log & resource
  - Exec shell

kubebox

https://github.com/astefanutti/kubebox

General	
`q`, `Ctrl`+`q`	Exit [3]
`n`	Change current namespace
`r`	Remote shell into container
`m`	Memory usage
`c`	CPU usage
`t`	Network usage
Log	
`g`, `⇧`+`g`	Move to top / bottom
`Ctrl`+`u`, `Ctrl`+`d`	Move one page up / down

How to accelerate more ?

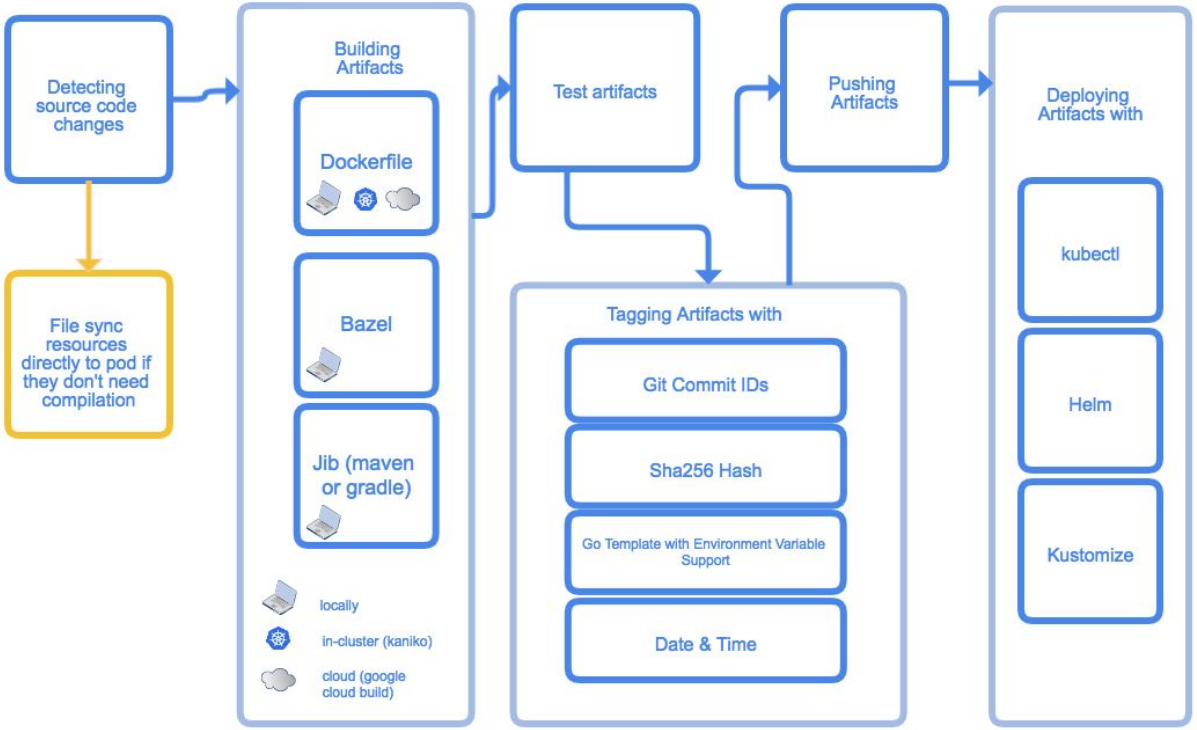
Automatic Workflow

- Makes it easy to build applications that run on Kubernetes.

- The "inner loop" of a developer's workflow: as they hack on code, but before code is committed to version control.

Skaffold workflow

Skaffold pipeline



Skaffold pipeline: build

- Dockerfile
  - locally with Docker
  - in-cluster with Kaniko
- Jib Maven and Gradle
  - locally
- Bazel locally
- Custom script locally
- CNCF Buildpacks

Skaffold pipeline: test

- https://github.com/GoogleContainerTools/container-structure-test

```
schemaVersion: 2.0.0

fileExistenceTests:
  - name: 'no go binary'
    path: '/usr/bin/go'
    shouldExist: false
```

Skaffold pipeline: tag

- gitCommit
  - use git commit IDs as tags
- sha256
  - use sha256 hashes of contents as tags
- envTemplate
  - use values of environment variables as tags
- dateTime
  - use date and time values as tags

Skaffold pipeline: Deploy

- kubectl
- kustomize
- helm

Skaffold pipeline: Deploy:helm

```
deploy:
  helm:
    releases:
    - name: reviews
      chartPath: helm-charts/reviews
    - name: ratings
      chartPath: helm-charts/ratings
      values:
        image: eks-workshop-bookinfo-ratings
      imageStrategy:
        helm: {}
    flags:
      install:
        - --atomic
      upgrade:
        - --reset-values
        - --atomic
```

Skaffold pipeline: Port Forwarding

- Skaffold will perform automatic port forwarding for resources that it manages:
- Or configure as following

```
portForward:
- resourceType: deployment
  resourceName: myDep
  namespace: mynamespace  #
  port: 8080 #
  localPort: 9000 # *Optional*
```

Skaffold operation

- skaffold build
  - Just build artifact (docker image)
- skaffold deploy
  - Just Deploy the application into Kubernetes cluster.
- skaffold delete
  - Just delete the application into Kubernetes cluster.

Skaffold operation

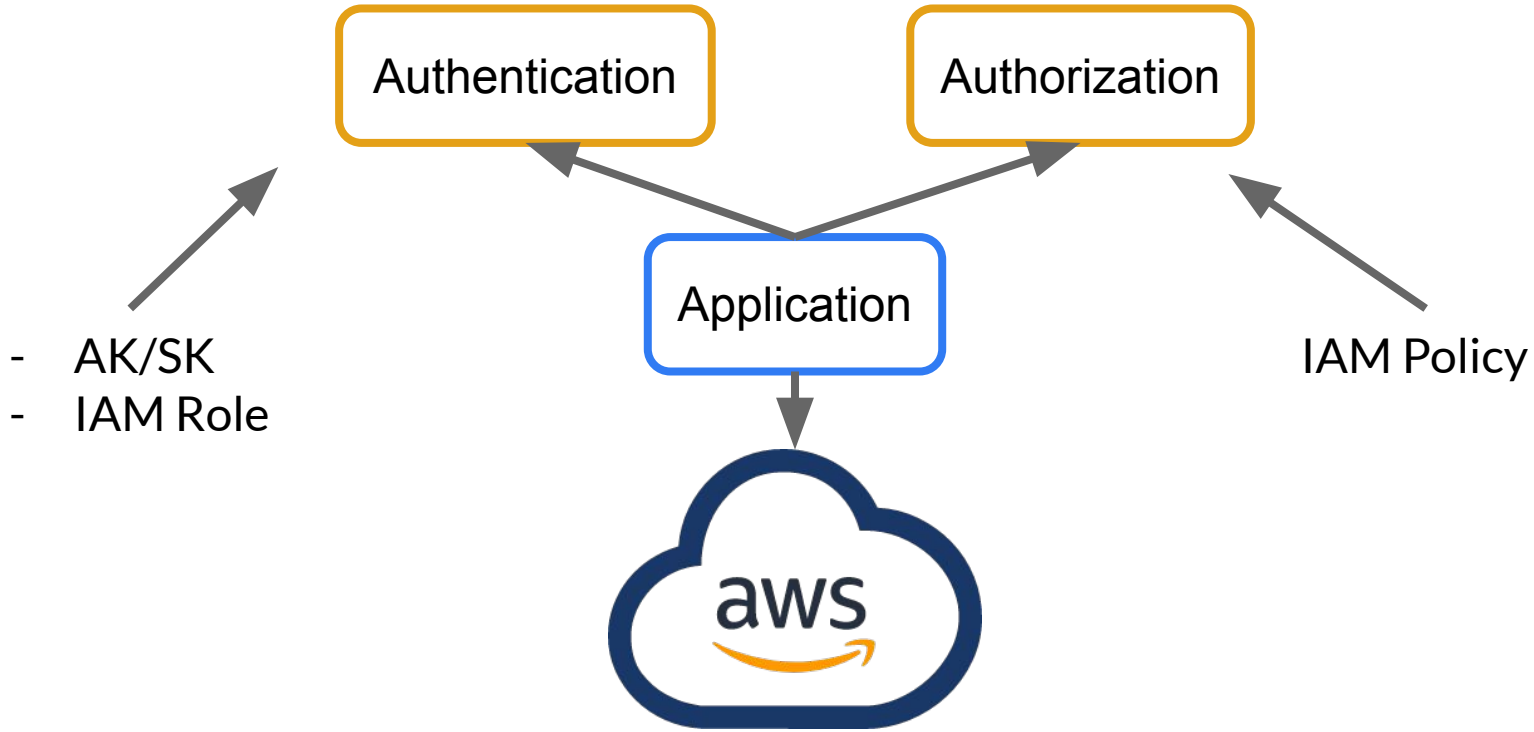
- skaffold dev
  - the continuous development mode, so that **every time you make changes to the source code, skaffold will build and deploy your application.**
- skaffold run
  - the standard mode, instructs skaffold to build and deploy your application exactly once.

```
$(~/environment/eks-workshop/sample)
$ skaffold dev

$ skaffold dev --port-forward --tail=false
$ curl 127.0.0.1:8080/health
$ curl 127.0.0.1:8081/health
$ curl 127.0.0.1:8082/health
$ curl 127.0.0.1:8083/health

$ skaffold dev --trigger='notify'
```

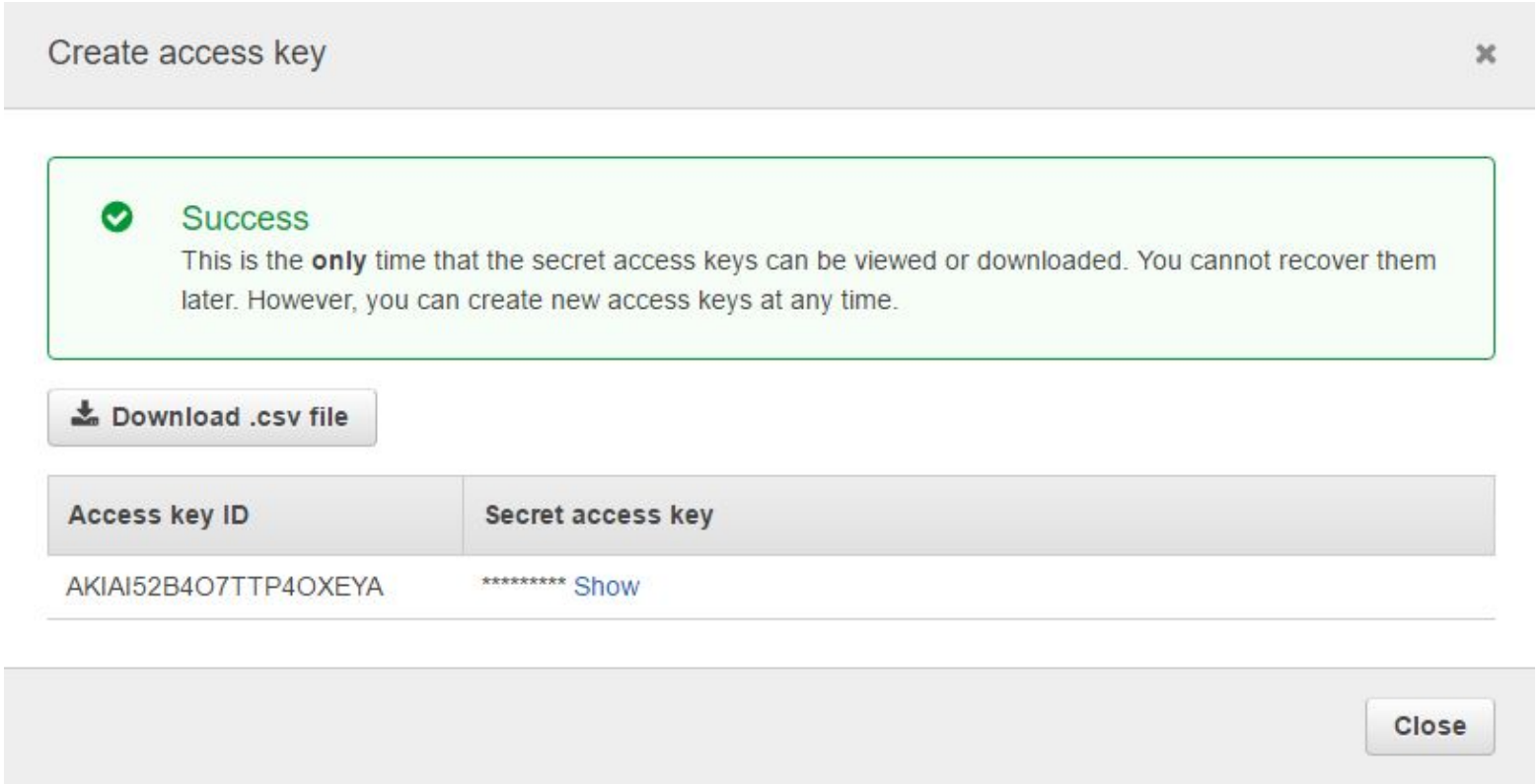
CH04

IAM Role for Service Account

How Application Access AWS Resource



Access Key ID/Secret Access Key



IAM Role

IAM Role

1

Trust

Who can assume this role

Defined by the role trust policy

2

Permissions

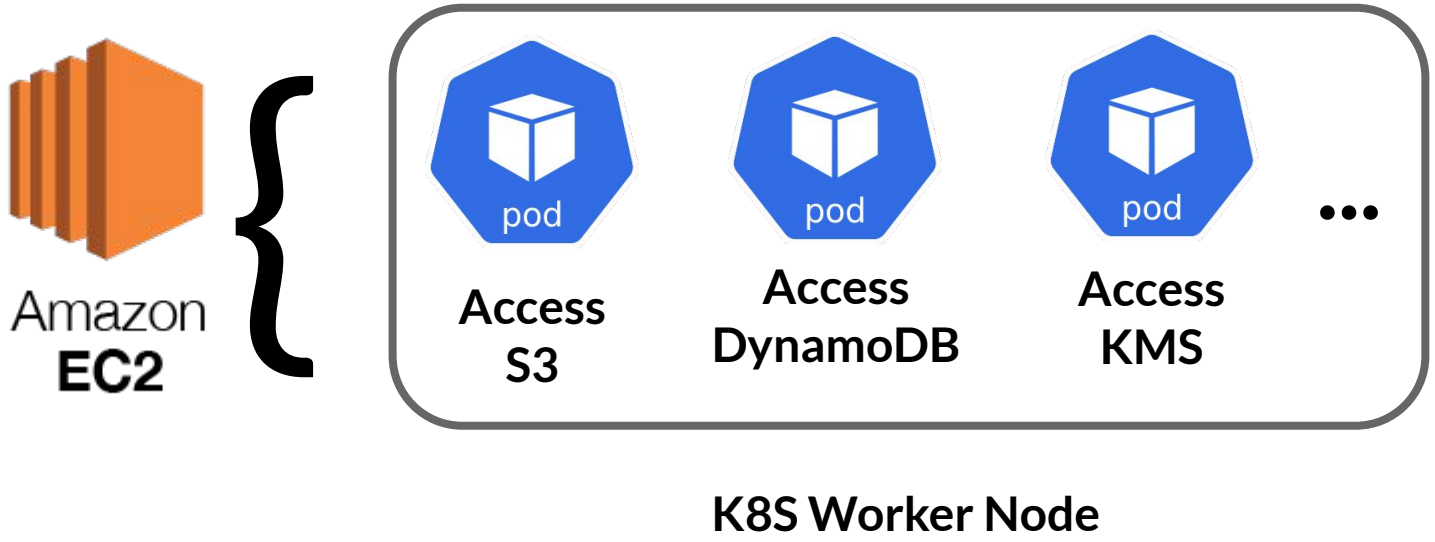
What you can do after assuming a role

Defined by IAM permissions policies

Which one is Better?

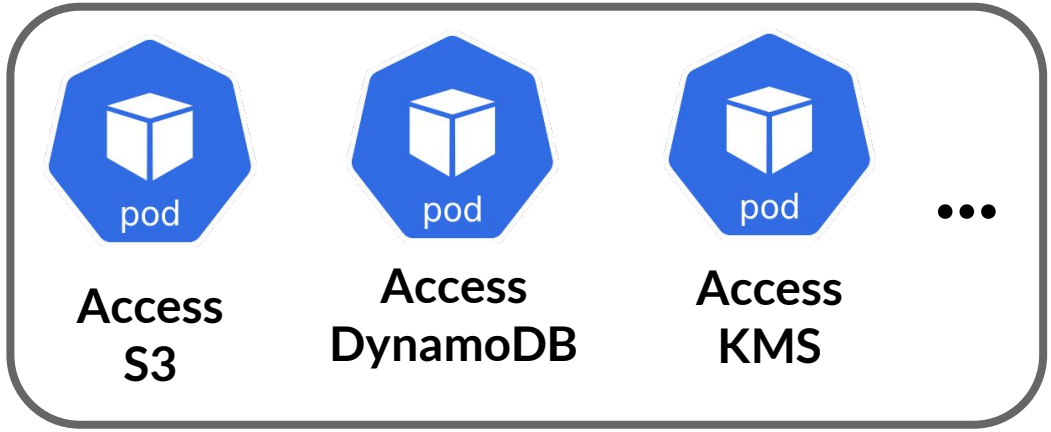
- It Depends…
- Inside AWS
- Outside AWS

K8S Use IAM Role Before...



K8S Use IAM Role Before...

IAM Policy
- **S3**
- **DynamoDB**
- **KMS**
- **...**



Third Party Tool

- Kube2iam
- Kiam
- Zalando

A Year Later...

EKS IAM Roles for Service Accounts (Pods) #23

Opened in **aws/containers-roadmap**

pauncejones commented on Dec 6, 2018

Update 1/9/19:

After talking about this internally, we've been working on a proposed solution for this. Below is a writeup on what we're thinking, and we've included some example scripts so you can get a feel for how we expect this to work.

Show more

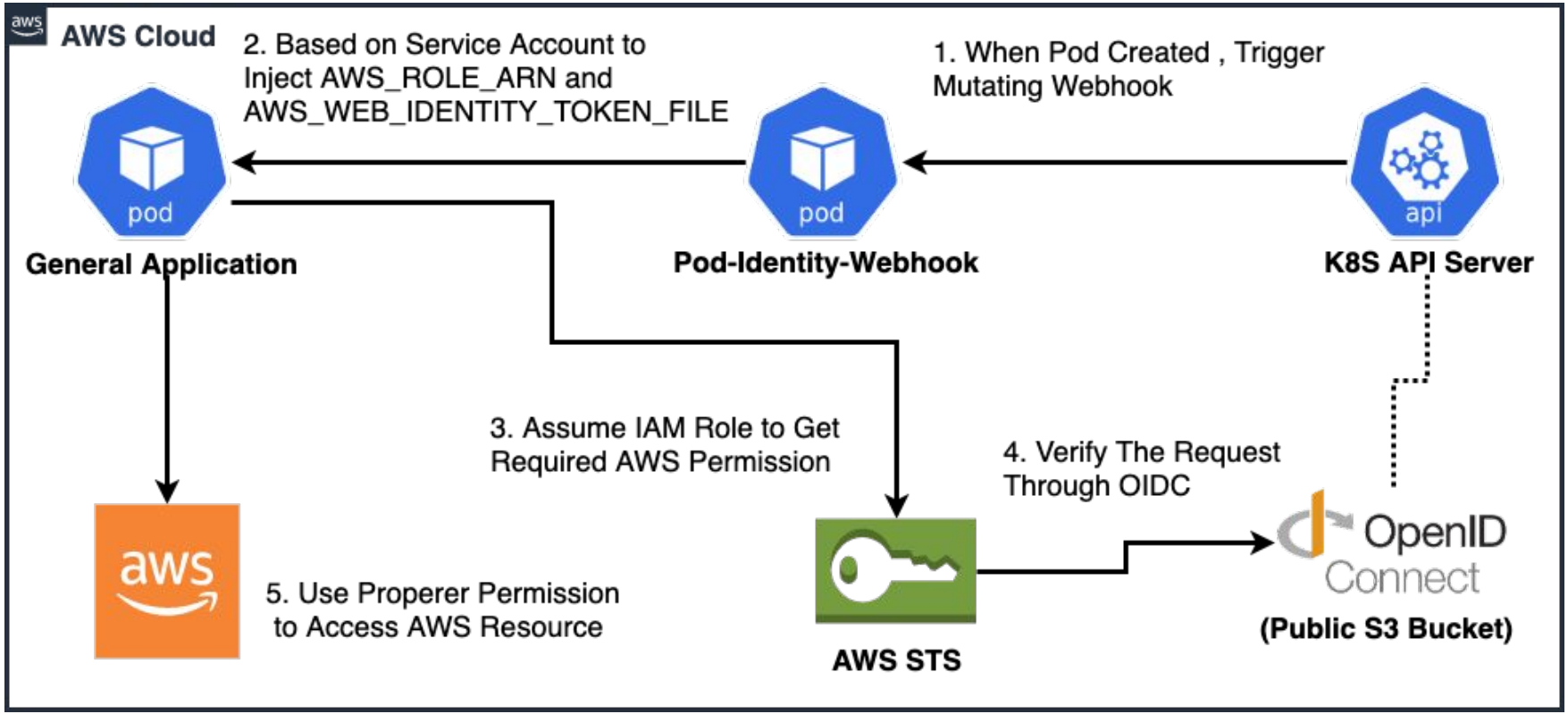
106

👍 434 👎 4 😄 22 🎉 33 ❤️ 50 🚀 19 👀 25

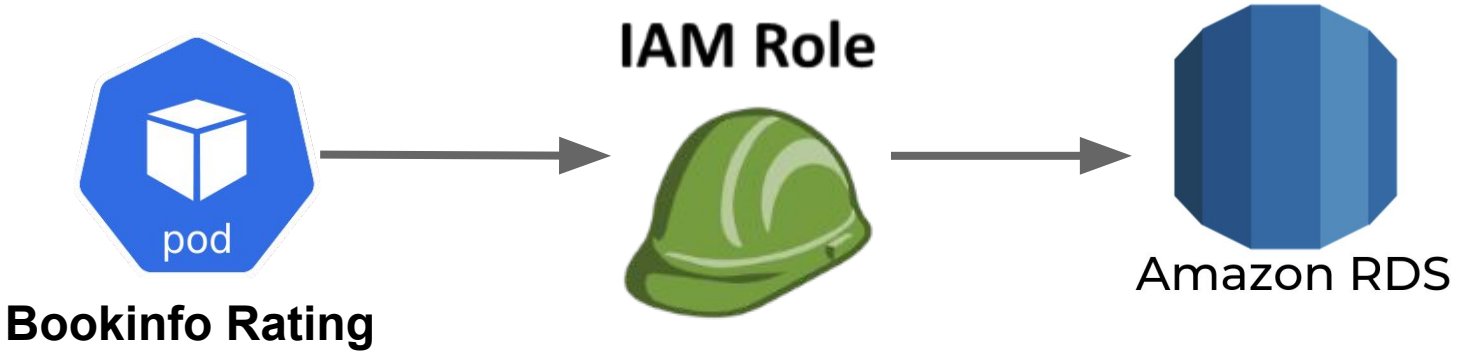
What is ServiceAccount?

- Kube2iam
- Kiam
- Zalando

IAM Role for Service Account



Learning by Doing



Create AWS RDS (MySQL) By TF

```
$(~/environment/eks-workshop/irsa/terraform)
$ terraform init
$ terraform apply -auto-approve
…
Outputs:
db_address = xxxxxxx.xxxxxxx.us-west-2.rds.amazonaws.com
db_password = WKi)EnlnG30EZB!t
db_policy = arn:aws:iam::1234567890:policy/test-irsa-rds
db_port = 3306
db_username = x5ceTNwv
```

Setup Database User for Ratings

```
$ mysql -h ${db_address} -P ${db_port} -u ${db_username} -p
Enter password:

mysql> CREATE USER irsa IDENTIFIED WITH
AWSAuthenticationPlugin as 'RDS';

mysql> GRANT ALL PRIVILEGES ON test.* TO 'irsa'@'%';
```

Create Database Scheme

```
mysql> USE test;

mysql> CREATE TABLE `ratings` (
   `ReviewID` INT NOT NULL,
   `Rating` INT,
   PRIMARY KEY (`ReviewID`)
);
```

Insert Data Into Database

```
mysql> INSERT INTO ratings (ReviewID, Rating) VALUES (1,
5);
mysql> INSERT INTO ratings (ReviewID, Rating) VALUES (2,
4);
```

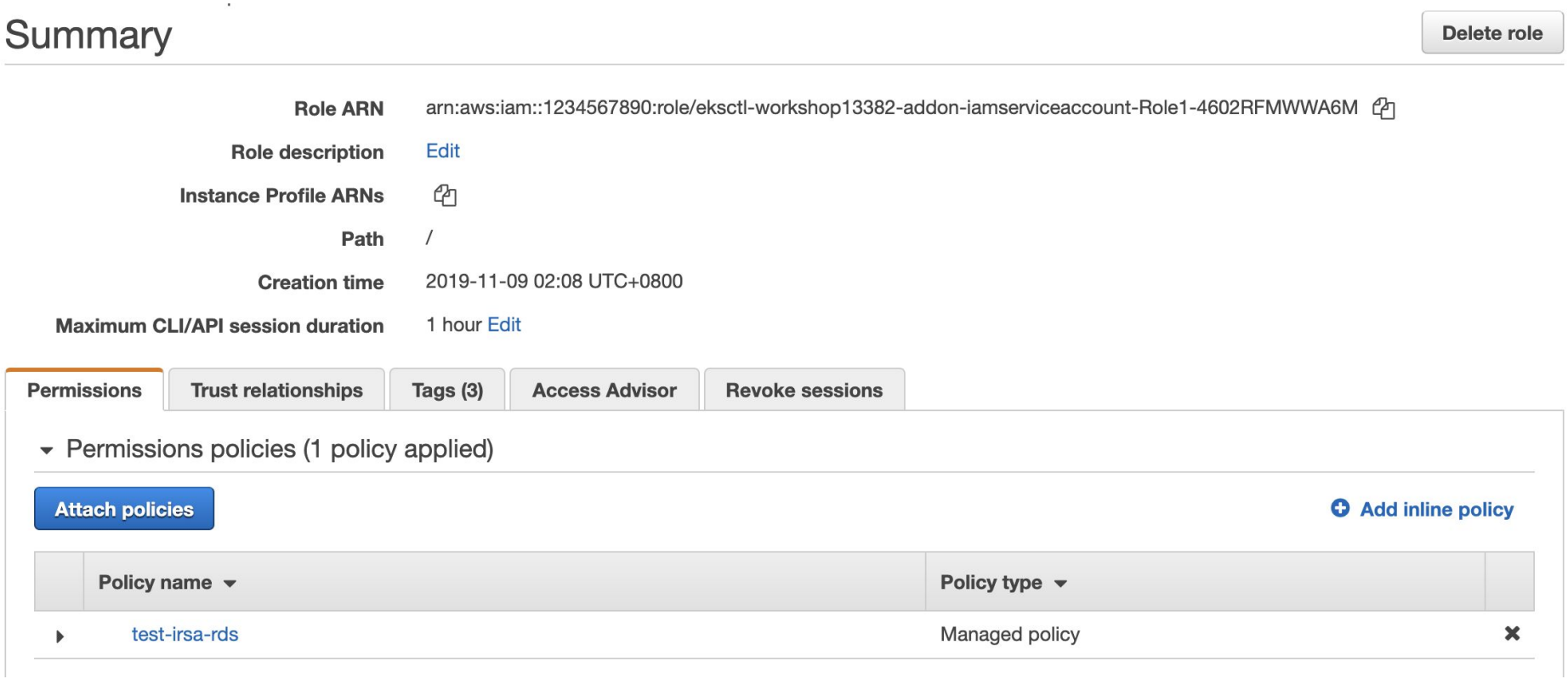
Create ServiceAccount W IAM Role

```
$ eksctl get cluster
$ eksctl create iamserviceaccount \
                --name bookinfo-ratings-irsa \
                --namespace default \
                --cluster ${cluster_name} \
                --attach-policy-arn ${db_policy} \
                --approve
```

Create ServiceAccount W IAM Role

```
$ kubectl describe serviceaccount bookinfo-ratings-irsa

Name:                bookinfo-ratings-irsa
Namespace:           default
Labels:              <none>
Annotations:
eks.amazonaws.com/role-arn=arn:aws:iam::1234567890:role/eks
ctl-workshop13382-addon-iamserviceaccount-Role1-4602RFMWWA6
M
```

Check the IAM Role



Modify Ratings Helm Chart (SA)

```
(~/environment/eks-workshop/sample/helm-charts/ratings/values.yaml)
serviceAccount:
  # Specifies whether a service account should be created
  #create: true
  create: false
  # The name of the service account to use.
  # If not set and create is true, a name is generated using the fullname template
  #name: bookinfo-ratings
  name: bookinfo-ratings-irsa
```

Modify Ratings Helm Chart (Env)

```
env:
  - name: DB_TYPE
    value: "mysql"
  - name: MYSQL_DB_HOST
    value: "xxx.xxx.us-west-2.rds.amazonaws.com"
  - name: MYSQL_DB_PORT
    value: "3306"
  - name: MYSQL_DB_USER
    value: "irsa"
  - name: AWS_REGION
    value: "us-west-2"
  - name: SERVICE_VERSION
    value: "v2"
```

Deploy Ratings by Skaffold

```
$ (~/environment/eks-workshop/sample)
$ skaffold run
```

Verify by API

```
$ curl http://ratings:9080/health
{"status":"Ratings is healthy"}

$ curl http://ratings:9080/ratings/1
{"id":1,"ratings":{"Reviewer1":5,"Reviewer2":4}}
```

How it Work?

```
$ kubectl describe pod ratings-77dff74649-r7xnl

…
AWS_ROLE_ARN:
arn:aws:iam::1234567890:role/eksctl-workshop13382-addon-iam
serviceaccount-Role1-4602RFMWWA6M
AWS_WEB_IDENTITY_TOKEN_FILE:
/var/run/secrets/eks.amazonaws.com/serviceaccount/token
…
```

How it Work?

```
(~/environment/eks-workshop/sample/src/ratings/ratings.js)
…
    if (process.env.DB_TYPE === 'mysql') {
      var signer = new AWS.RDS.Signer();

      signer.getAuthToken({ // uses the IAM role access
keys to create an authentication token
        region: awsRegion,
        hostname: hostName,
        port: portNumber,
        username: username
…
```

CH05

Service Mesh by Istio

What is Service Mesh ?
What is Service Mesh for ?

Service Mesh

It is a **dedicated infrastructure layer** for making **service-to-service** communication:

- **Safe**
- **Fast**
- **Reliable**

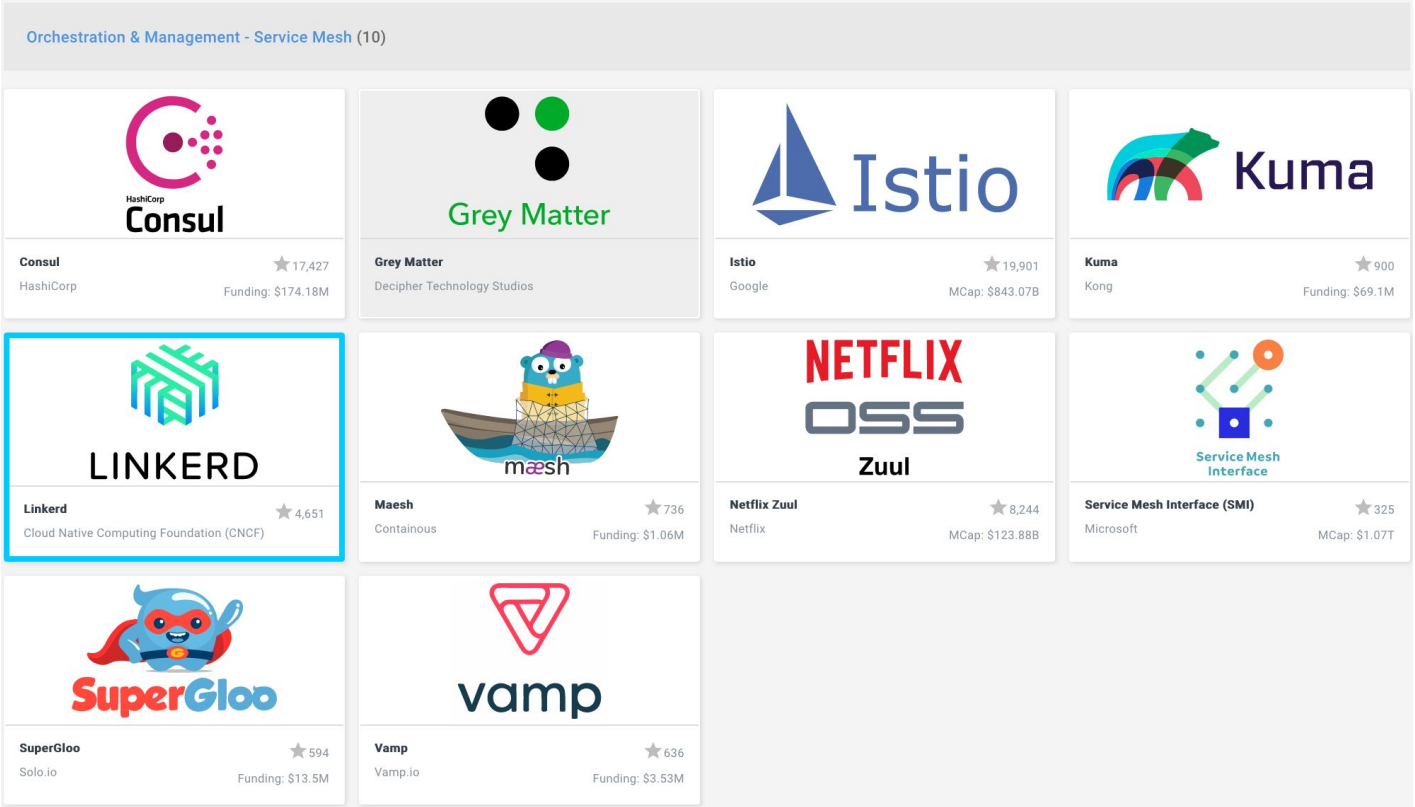
A NETWORKING MODEL?

- networking model layer of abstraction above TCP/IP
- Make **service communication** could be **monitored**, **managed** and **controlled**

Why Need?

- It comes with the rise of
  - Cloud Native
    - Dynamic Service
  - Microservices
    - Complicated service communication

One important chapter in CNCF



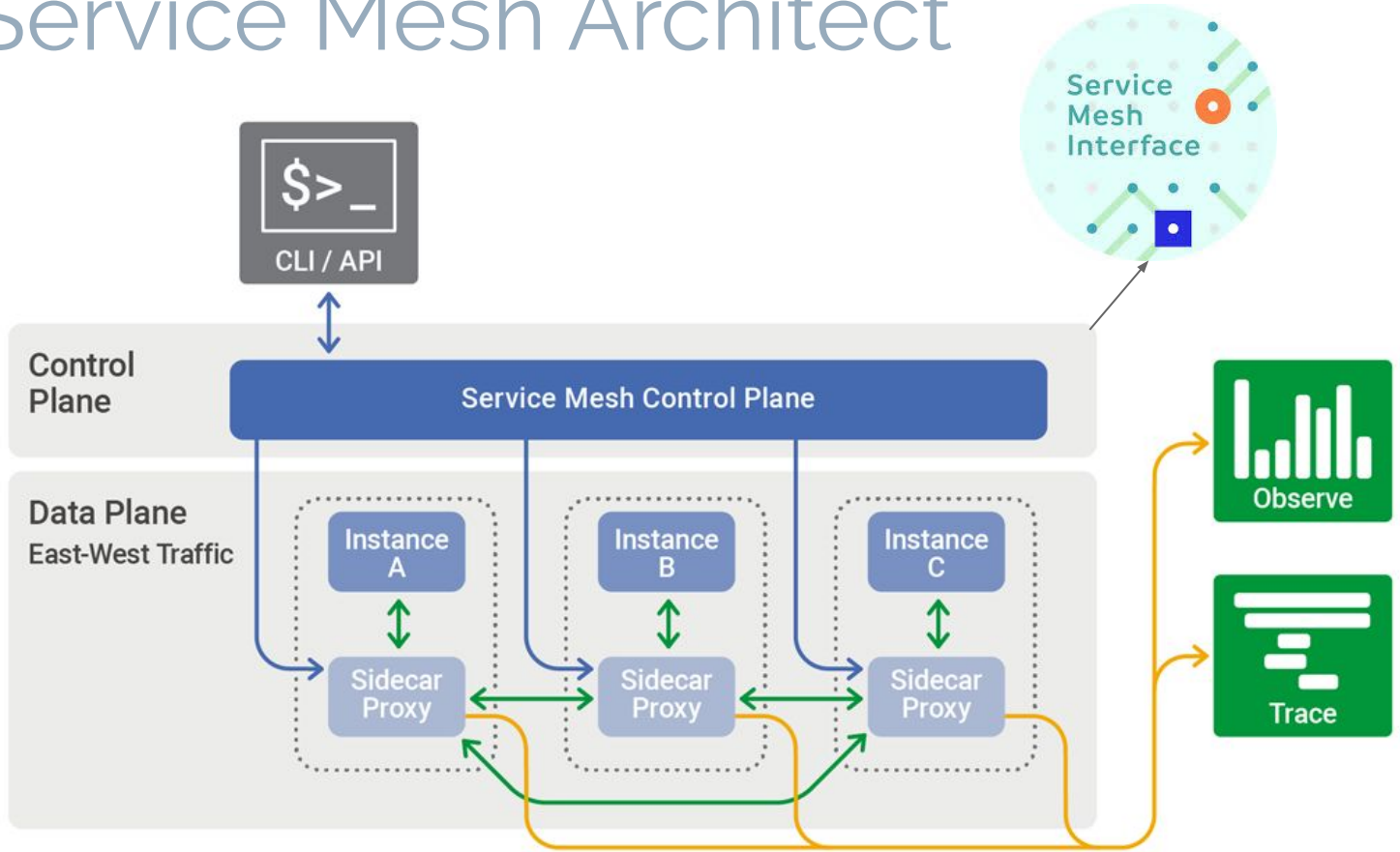
Kubernetes is good

- Application
  - Interact with service discovery
  - Easy to scale
  - Easy to update

But ……

- Lack of
  - Visibility
  - Policy Enforcement
  - Traffic Control
  - Security Connection
- Service Mesh Coming

Service Mesh Architect

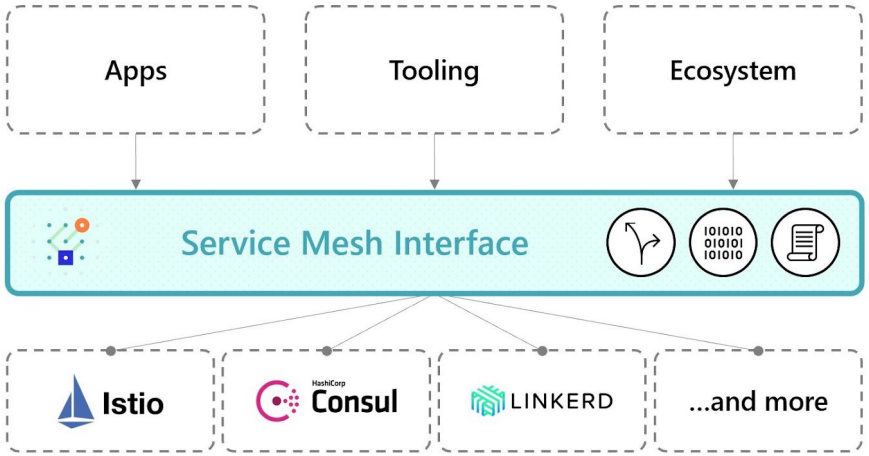


Service Mesh Interface

Standardized
Standard interface for service mesh on Kubernetes

Simplified
Basic feature set to address most common scenarios

Extensible
Support for new features as they become widely available

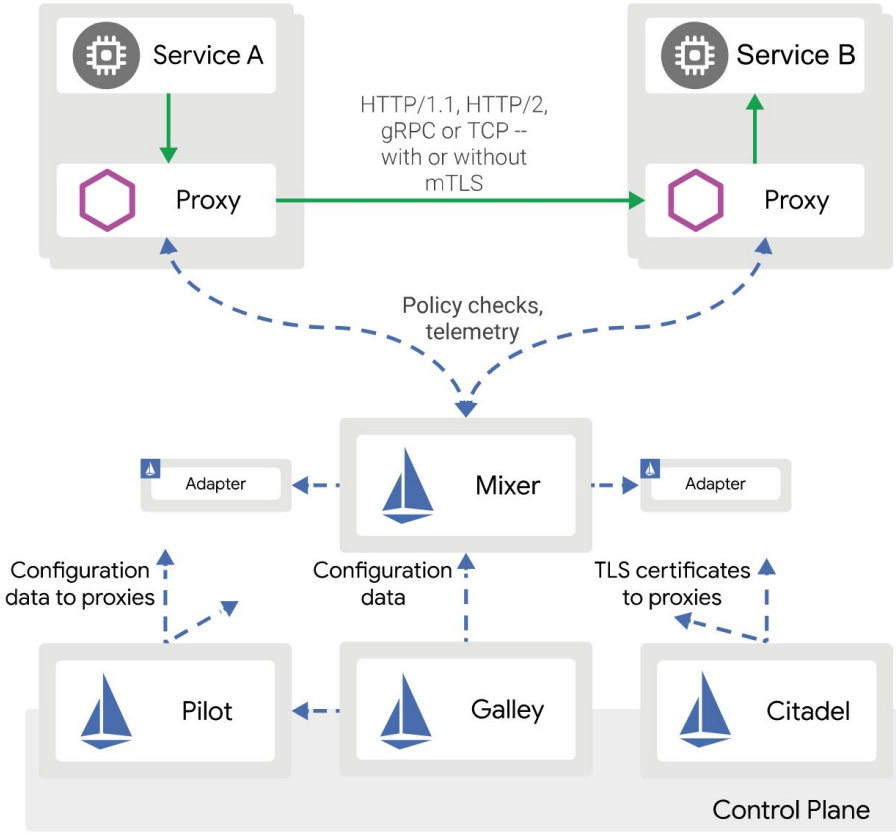


Service Mesh: Istio

- Combined efforts of IBM, Google, and Lyft.
- 4 goals
  - Connect
  - Secure
  - Control
  - Observe

Istio Architecture

Istio Architecture

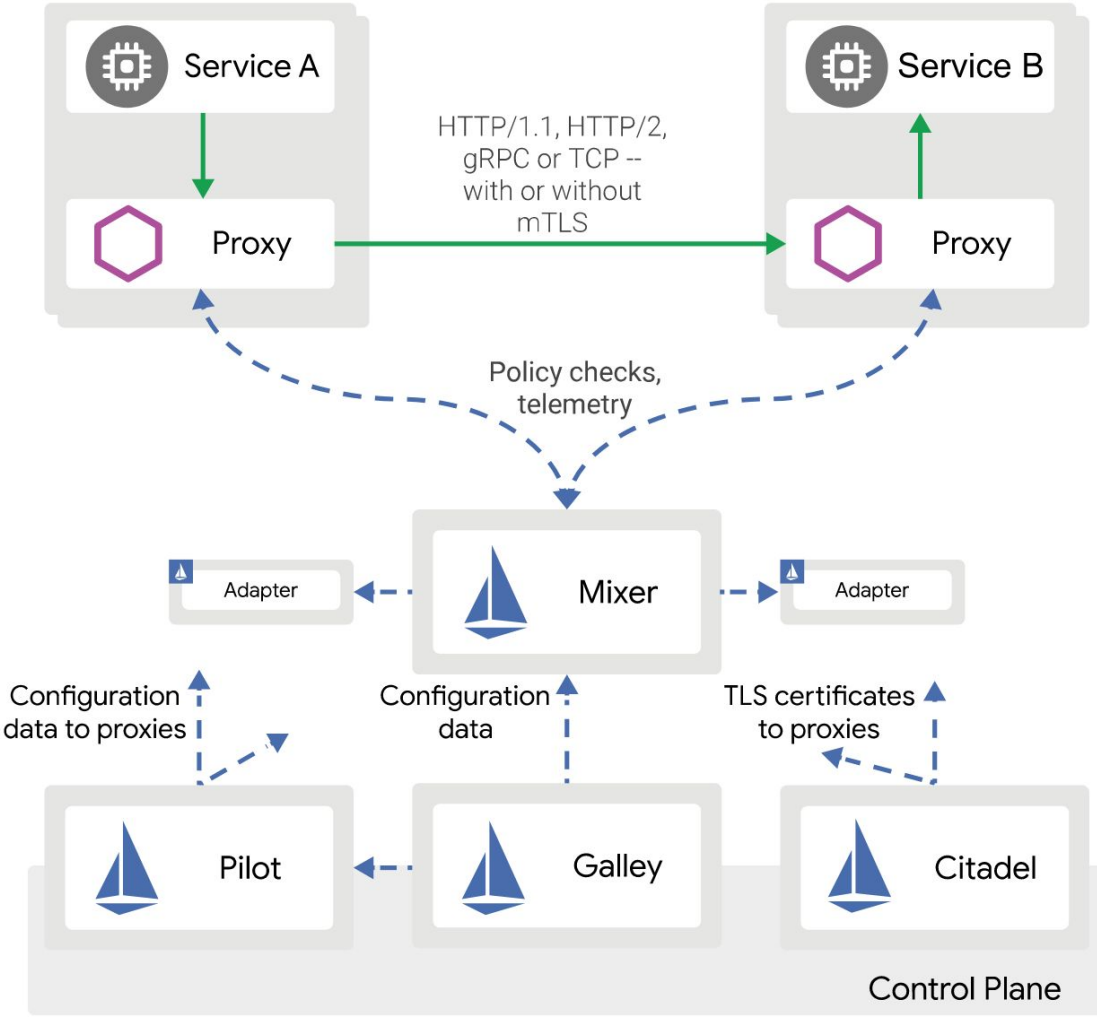


Istio Architecture

- Proxy
- Mixer
  - Adapter
- Pilot
- Gallery
- Citadel

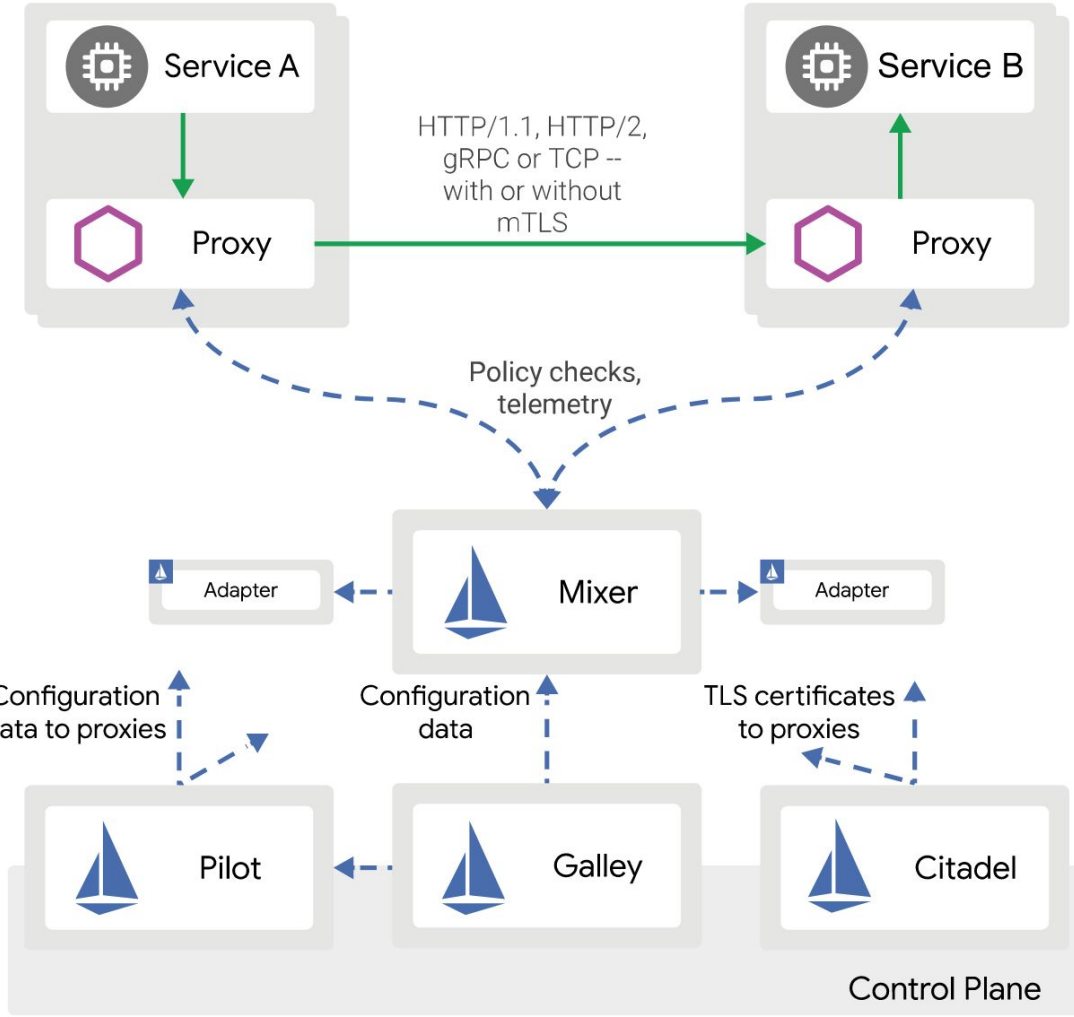
Proxy: Envoy

- high-performance proxy
- Envoy is deployed as a sidecar the same Kubernetes pod.



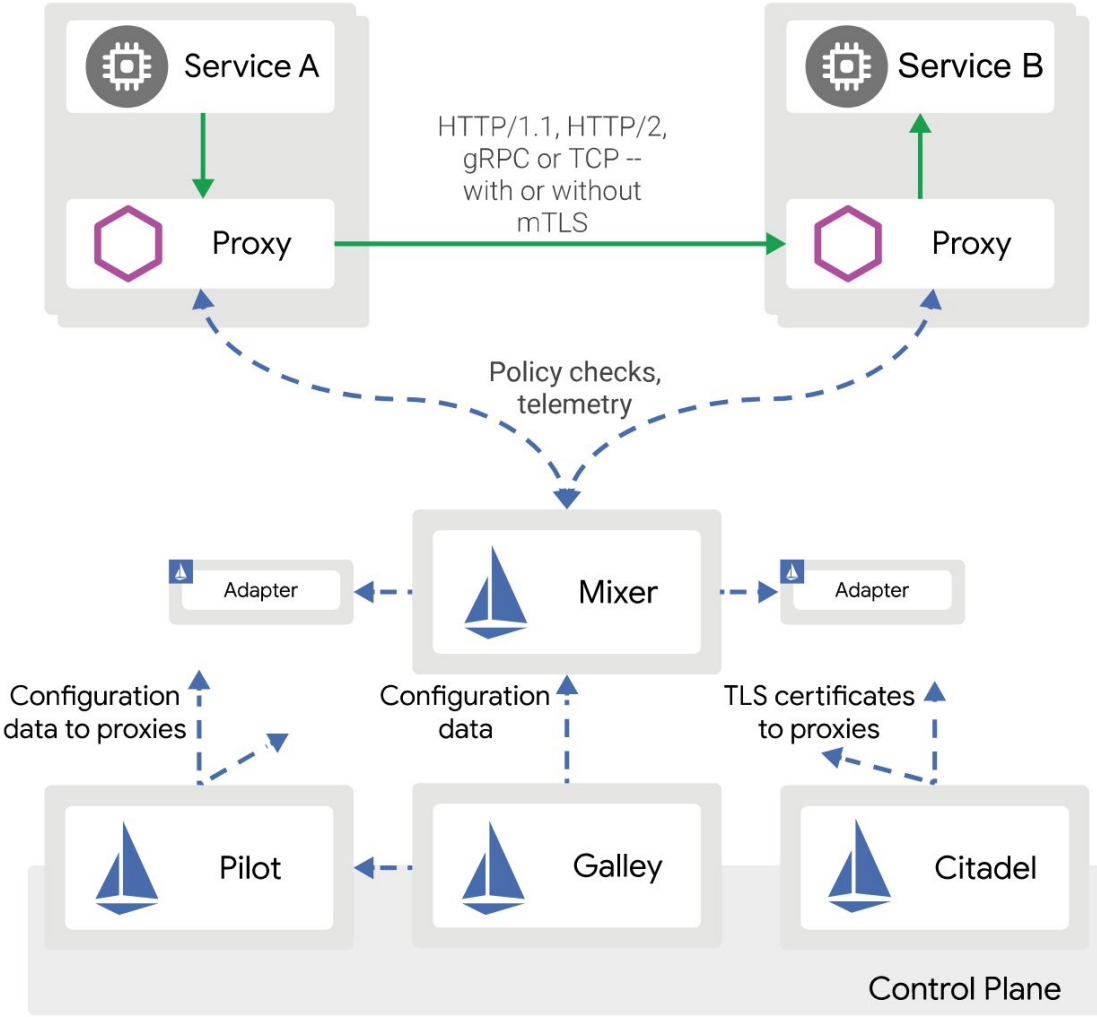
Mixer

- access control
- usage policies
- collects telemetry data from the Envoy proxy and other services.



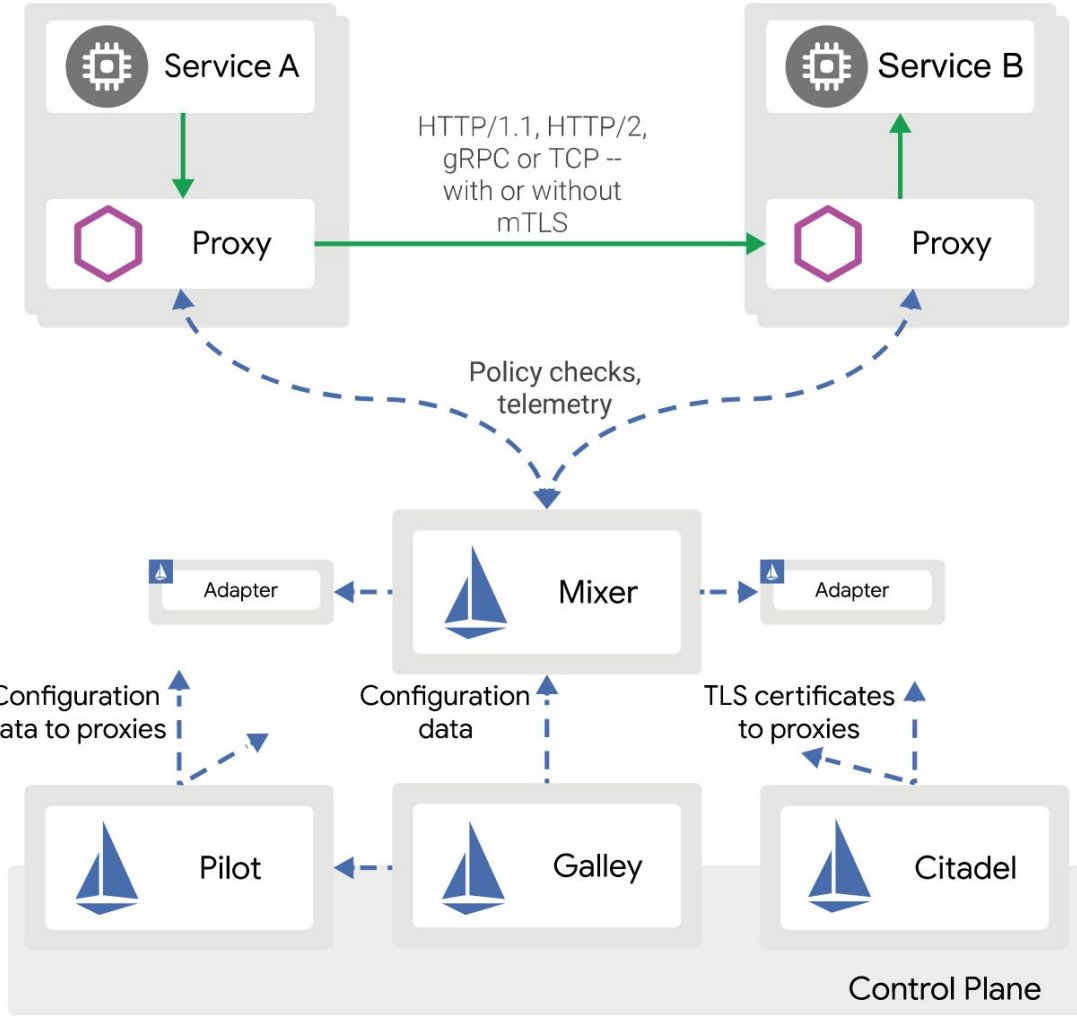
Pilot

- converts high level routing rules into Envoy-specific configurations
  - Service discovery
  - Traffic management capabilities
  - Resiliency



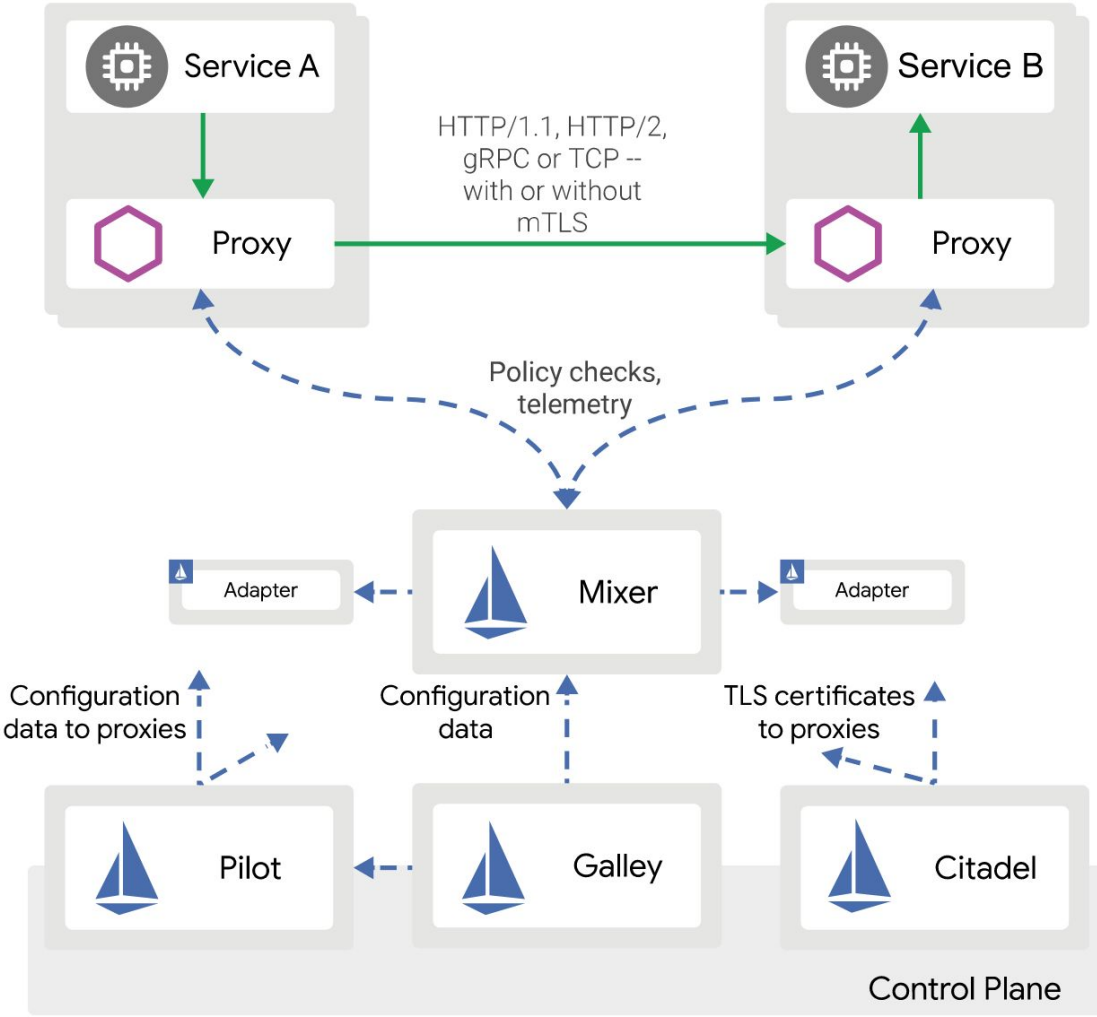
Gallery

- Configuration
  - Validation
  - Ingestion
- Istio CRDs
- Kubernetes Resources



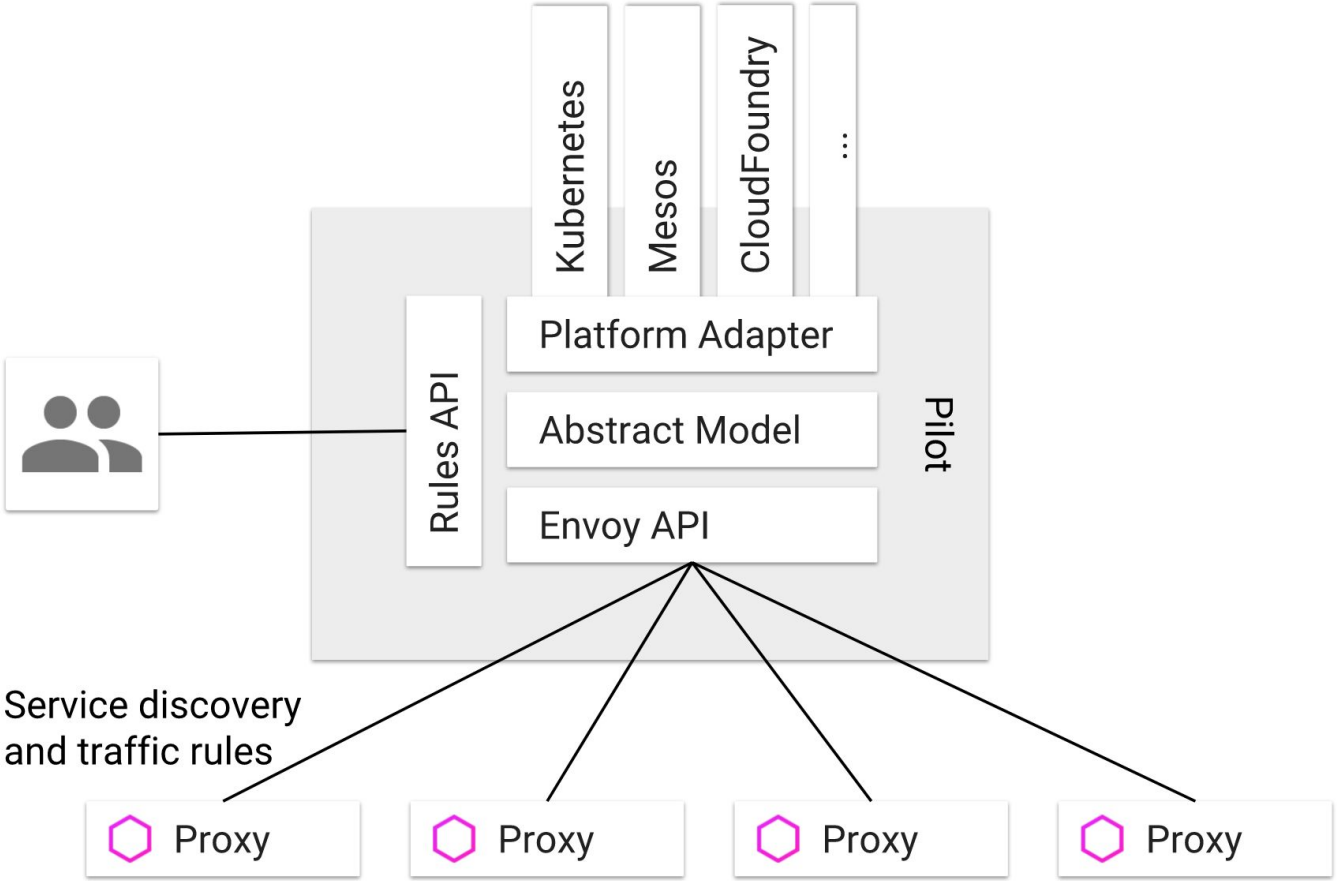
Citadel

- Strong authentication with built-in identity
  - service-to-service
  - end-user
- Credential management
- Can use Citadel to upgrade unencrypted traffic in the service mesh.

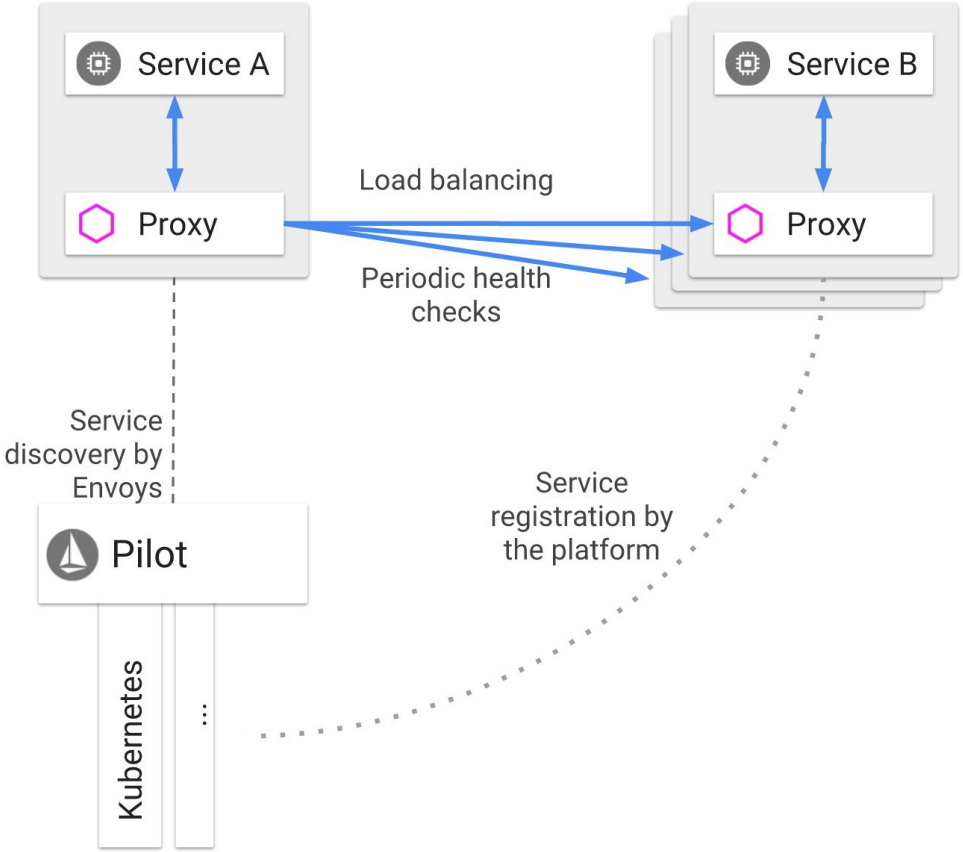


Traffic Management

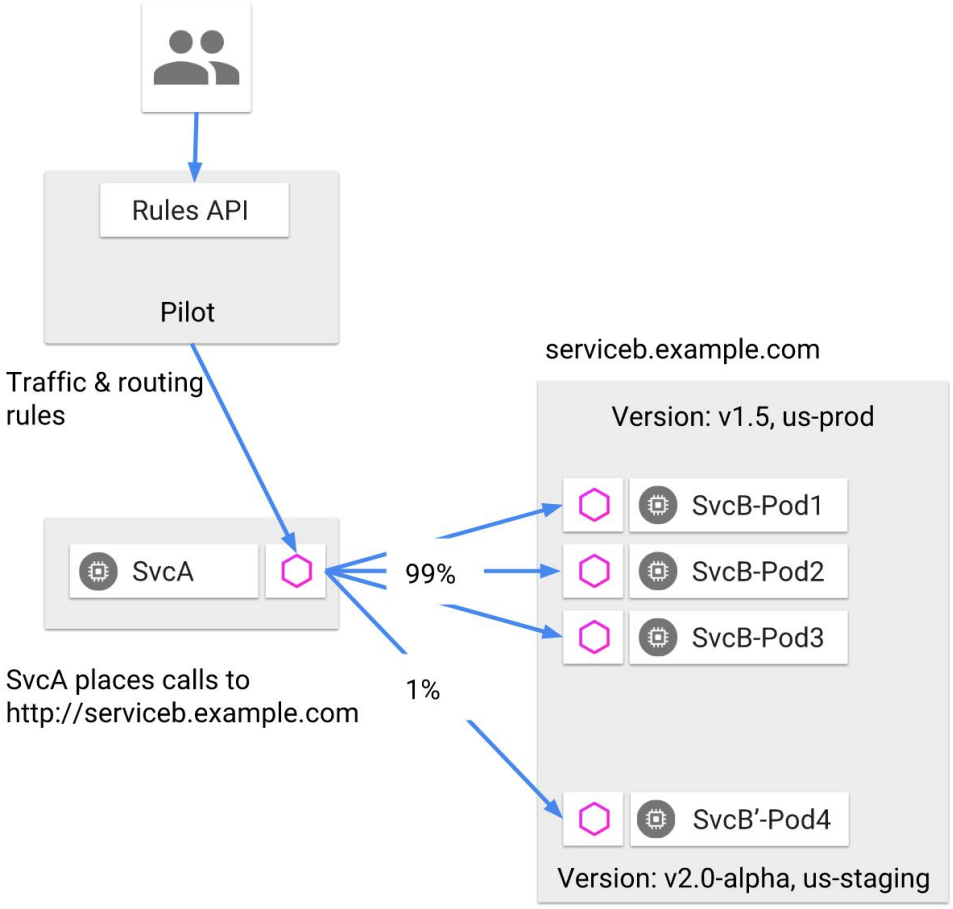
Pilot & Envoy


Kubernetes
Mesos
CloudFoundry
...
Platform Adapter
Abstract Model
Envoy API
Rules API
Pilot
Service discovery
and traffic rules
Proxy
Proxy
Proxy
Proxy

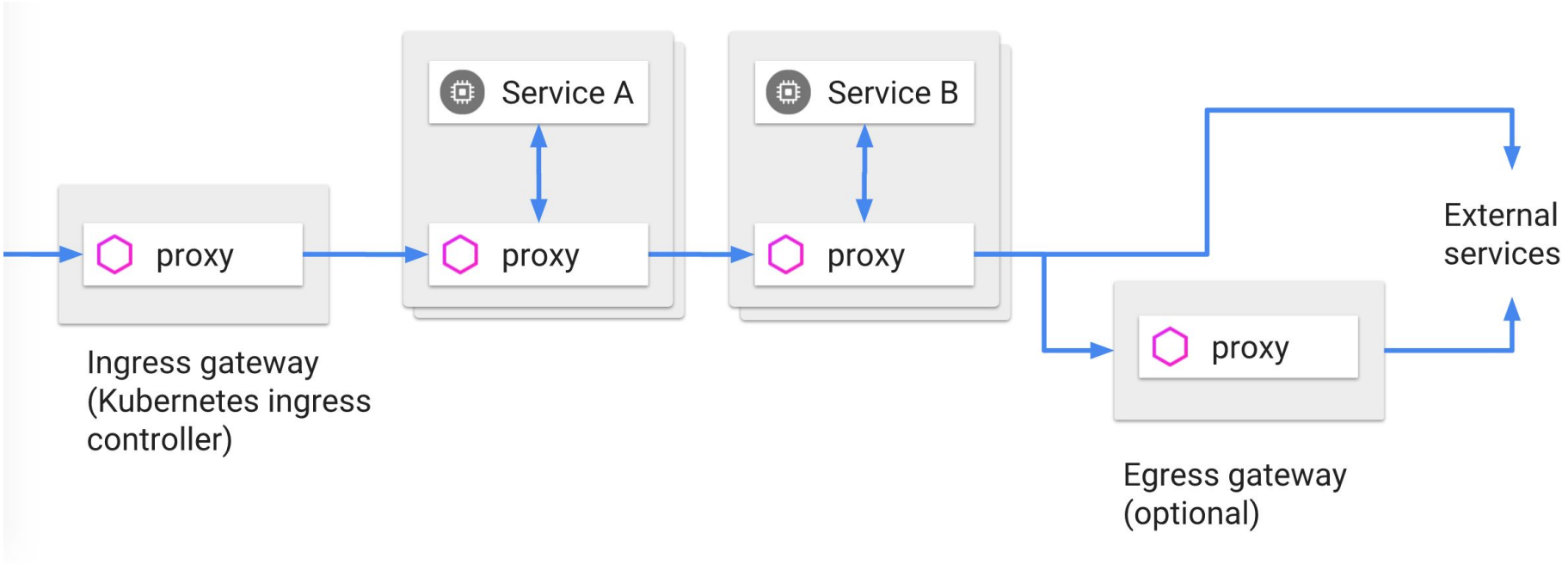
Discovery and load balancing



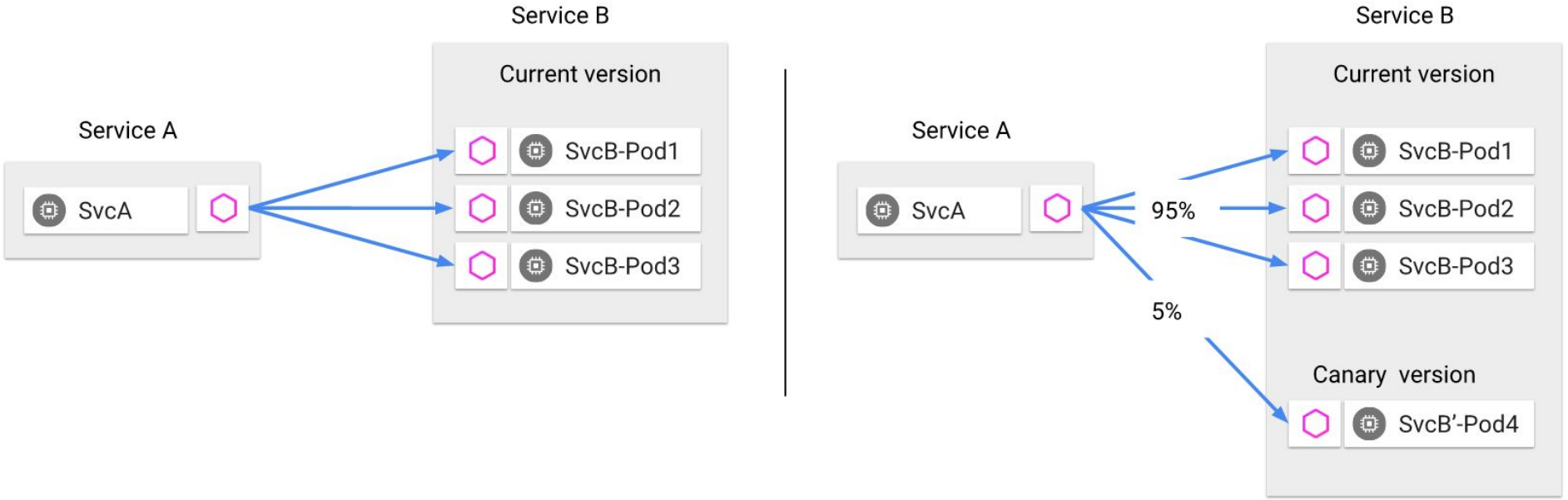
Communication between services



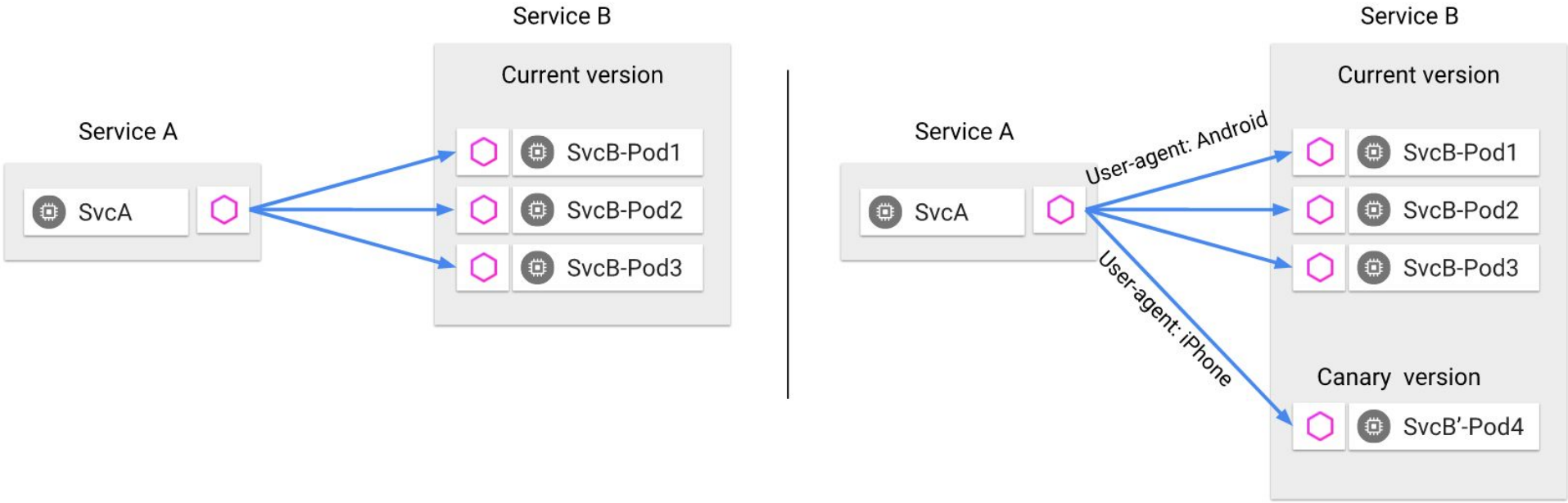
Ingress and egress



Canary

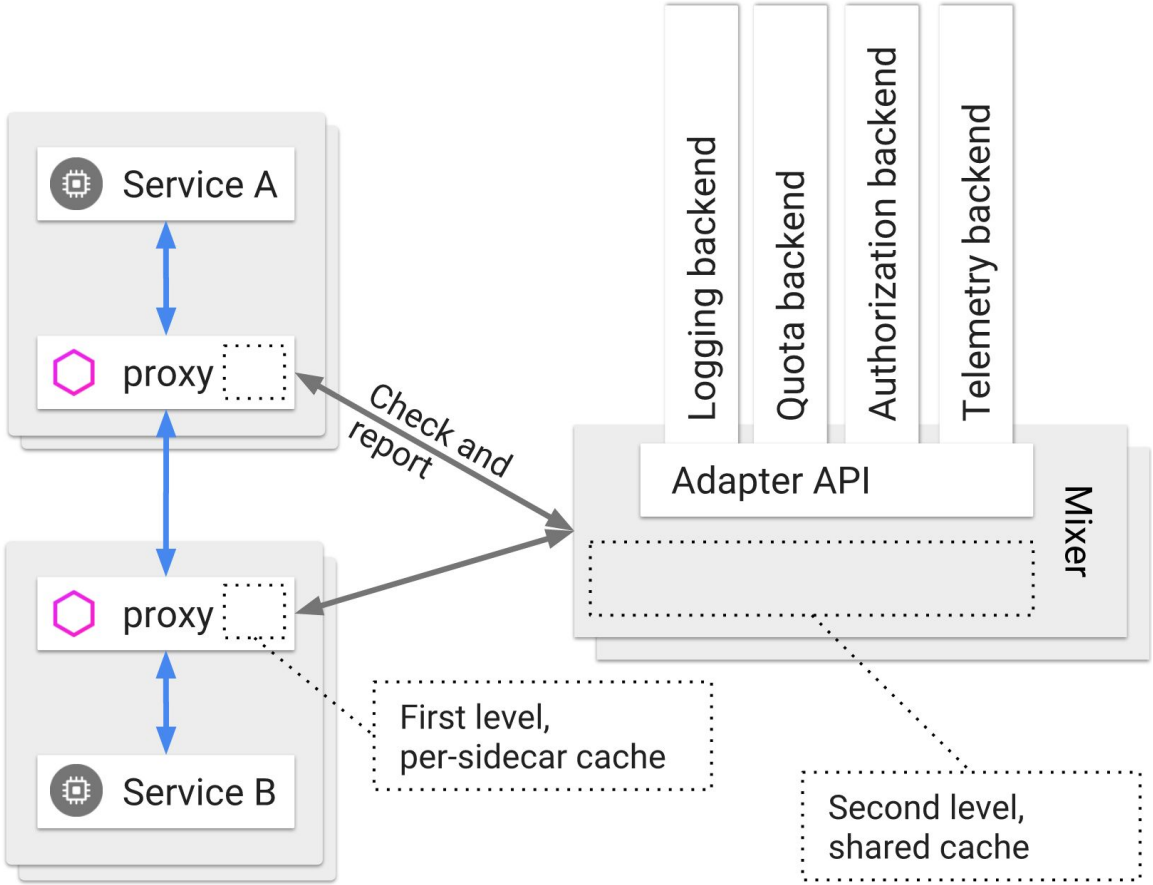

Service B
Current version
Service A
SvcA
SvcB-Pod1
SvcB-Pod2
SvcB-Pod3
Service B
Current version
Service A
SvcA
95%
5%
SvcB-Pod1
SvcB-Pod2
SvcB-Pod3
Canary version
SvcB'-Pod4

Header matching

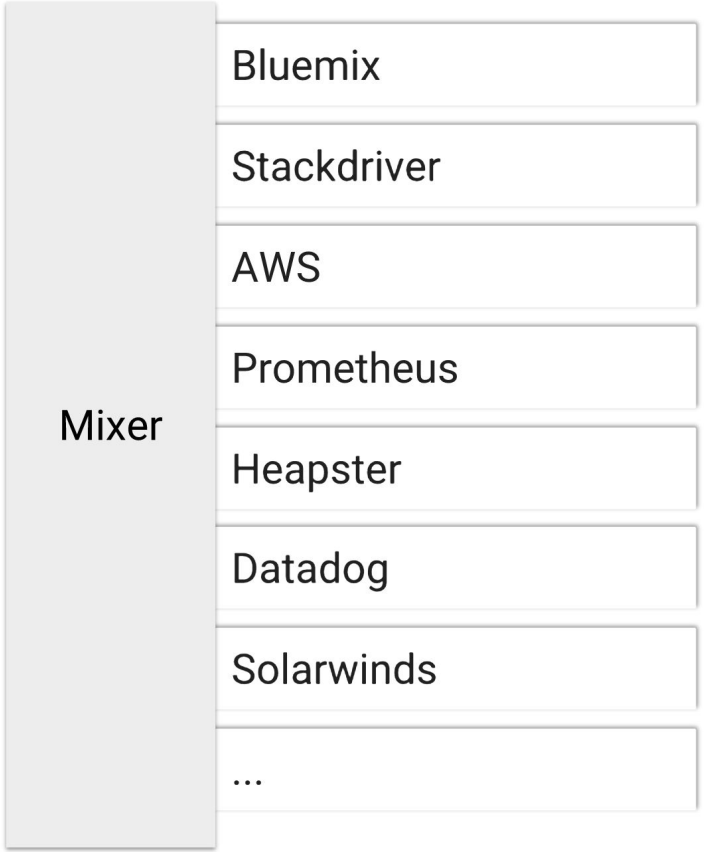


Policies & Telemetry

Policies & Telemetry

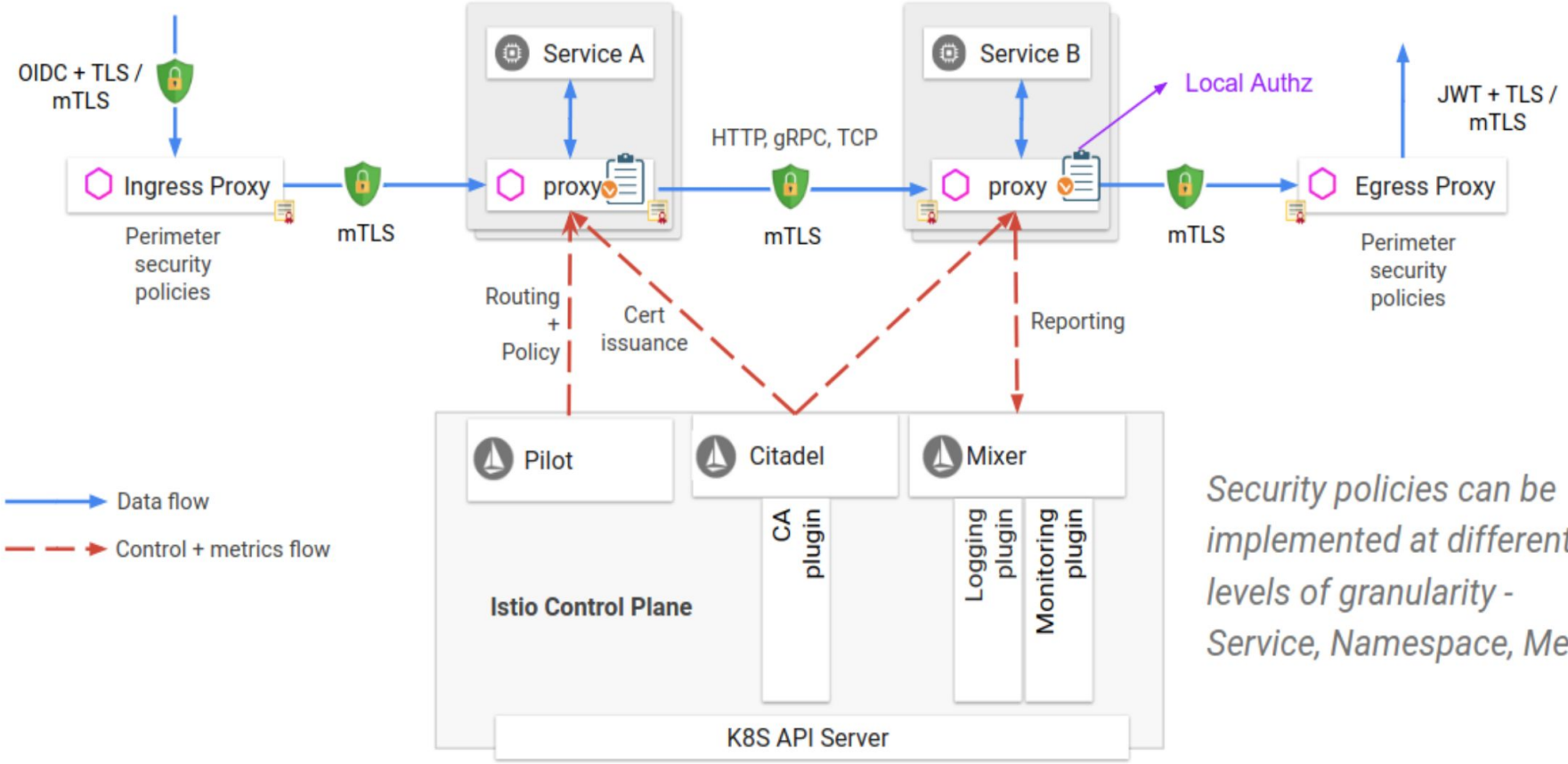


Policies & Telemetry



Security

Security



Security policies can be implemented at different levels of granularity - Service, Namespace, Mesh.

Istio component duty

- Proxy
  - Control Traffic
- Pilot
  - Apply setting to proxy (Rule, Policy …)
- Mixer
  - Get report(metrics, log ...) from proxy
- Citadel
  - Manage the certificate in the traffic

Istio Integrated

- Kiali
- Grafana
- Prometheus
- Jaeger

```
$(~/environment/eks-workshop/service-mesh/01-demo-ui)

$ ./install.sh

$ ../get-links.sh

$ ../ab.sh productpage

$ ./uninstall.sh
```

kiali

Topology

See your services communicate

Health

Quickly identify issues

Metrics

Chart Istio and App performance

Tracing

Follow requests with Jaeger Distributed Tracing

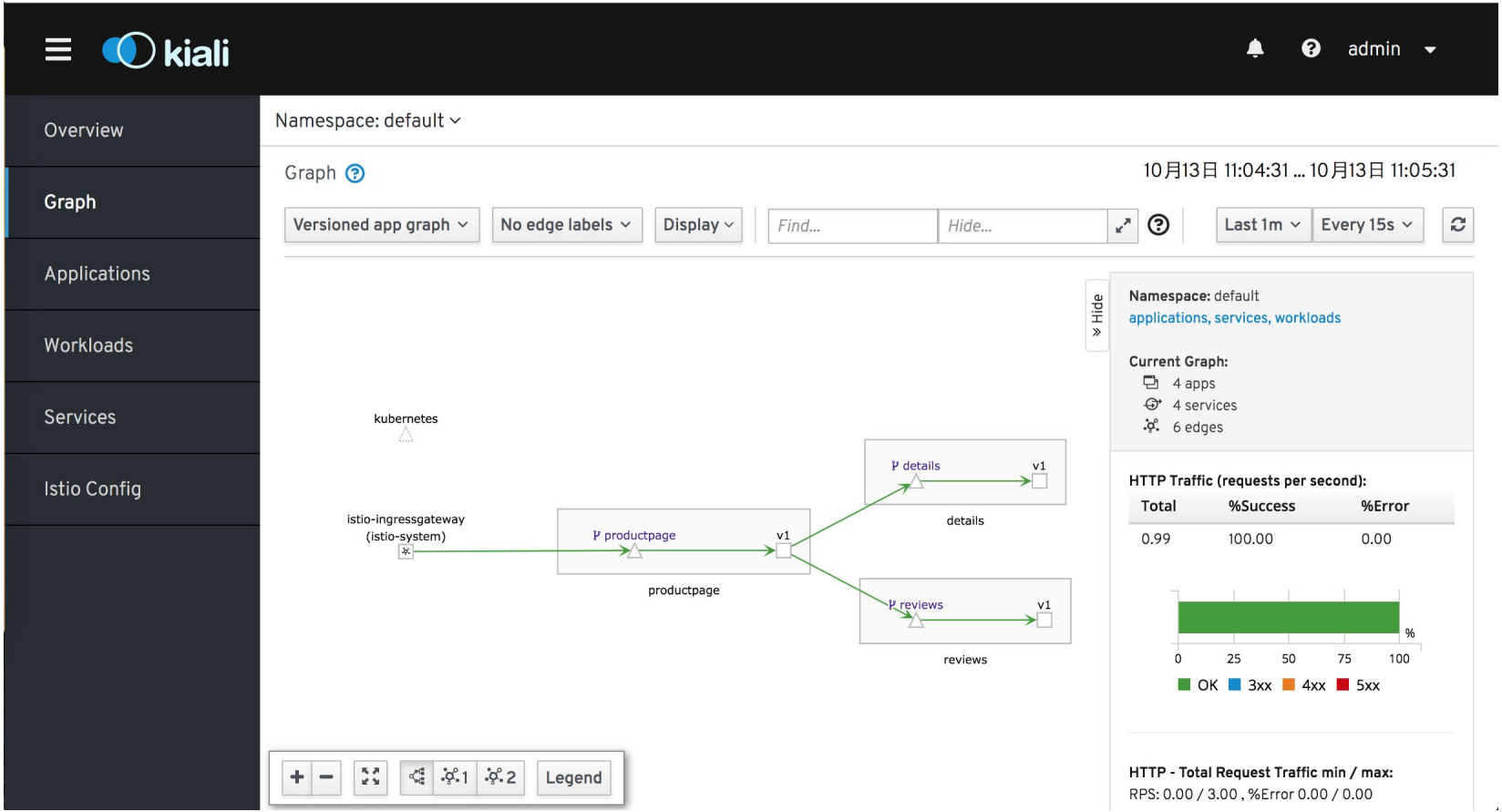
Validations

Detect advanced misconfigurations

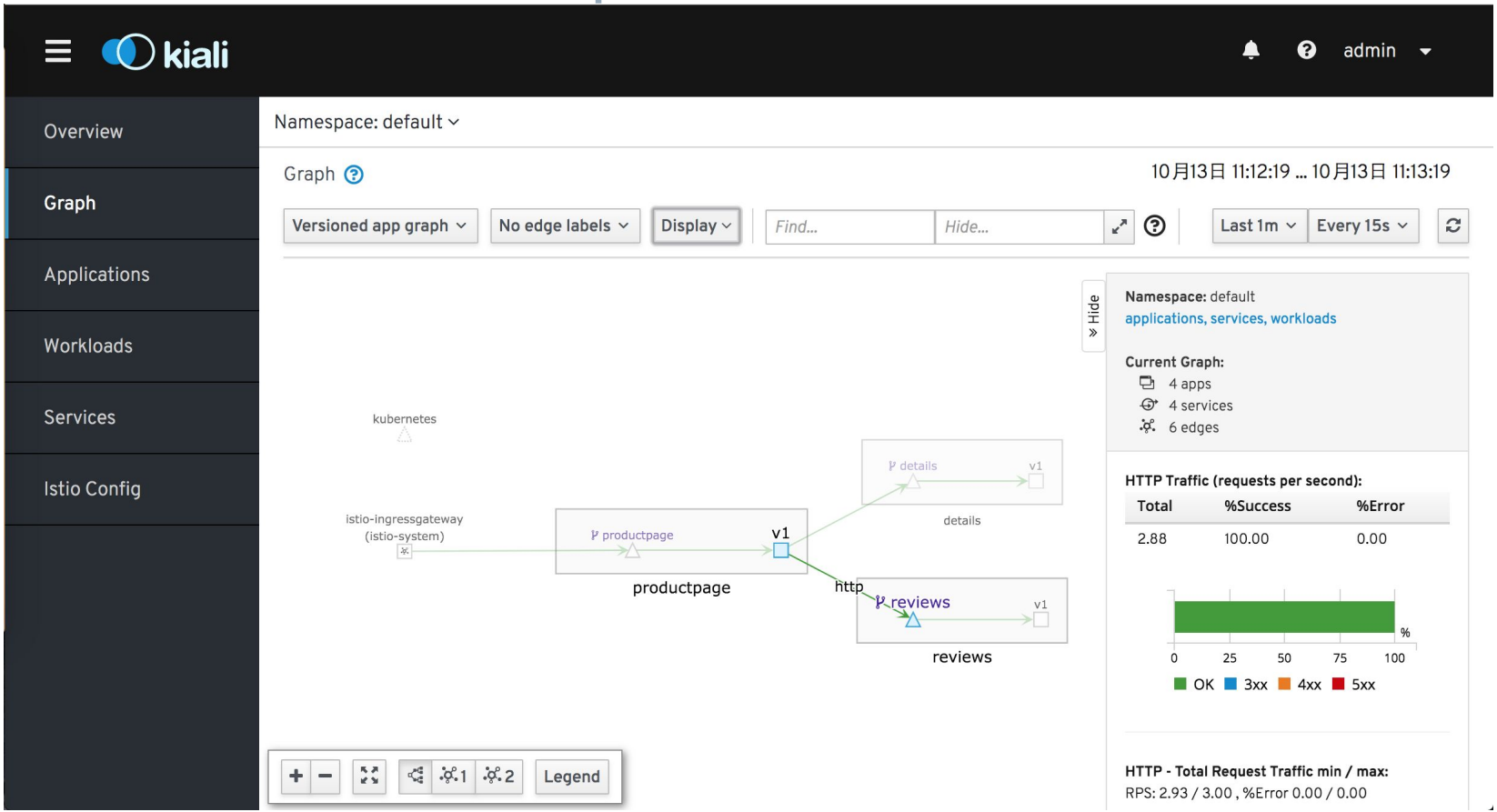
Wizards

Easily configure Istio routing

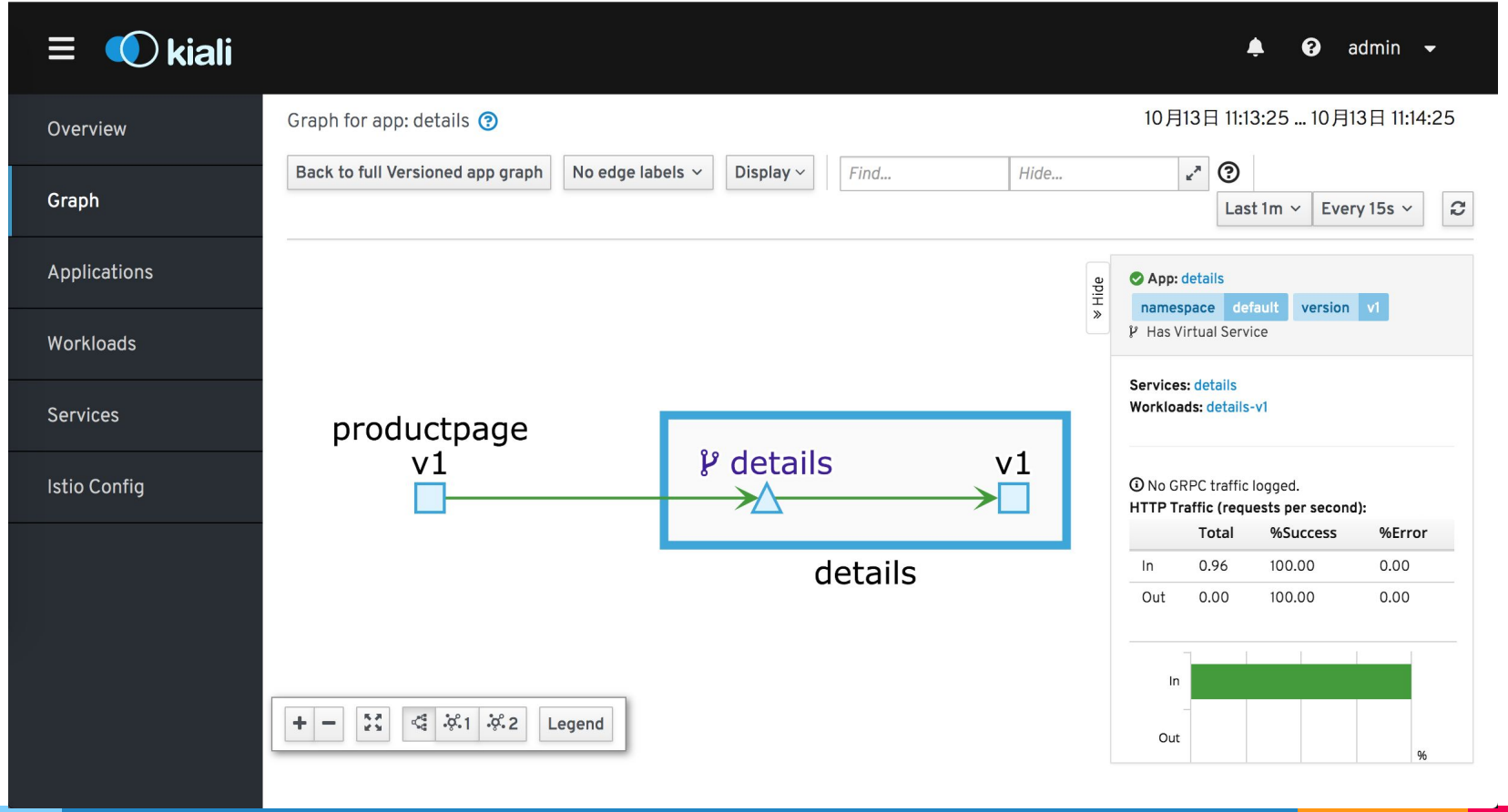
Kiali : Graph


kiali
admin
Overview
Graph
Applications
Workloads
Services
Istio Config
Namespace: default
Graph
10月13日 11:04:31 ... 10月13日 11:05:31
Versioned app graph
No edge labels
Display
Find...
Hide...
Last 1m
Every 15s
Hide
kubernetes
istio-ingressgateway
(istio-system)
productpage
v1
productpage
details
v1
details
reviews
v1
reviews
Legend
Namespace: default
applications, services, workloads
Current Graph:
4 apps
4 services
6 edges
HTTP Traffic (requests per second):
Total
%Success
%Error
0.99
100.00
0.00
0
25
50
75
100
%
OK
3xx
4xx
5xx
HTTP - Total Request Traffic min / max:
RPS: 0.00 / 3.00 , %Error 0.00 / 0.00

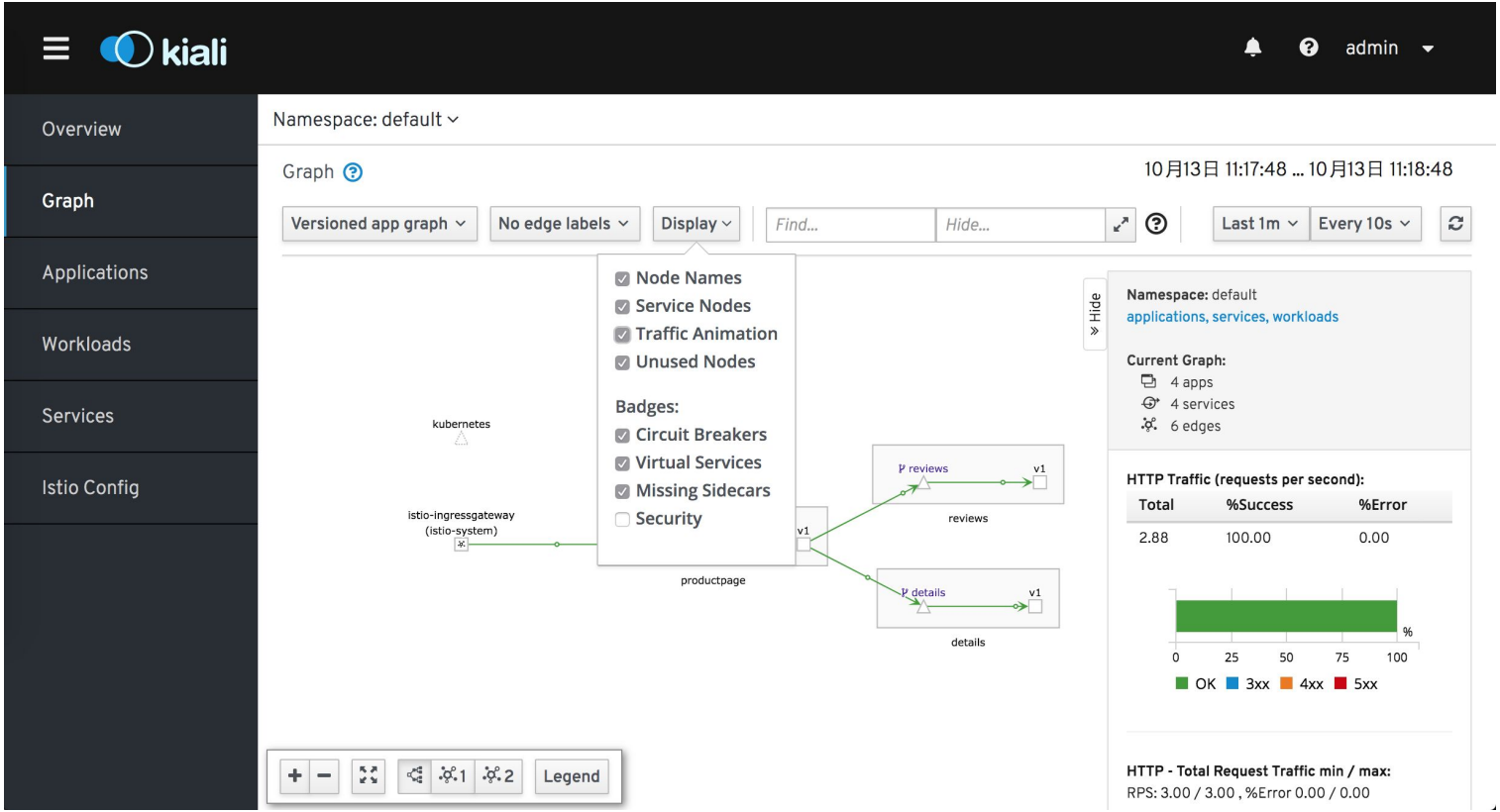
Kiali : Graph : Health



Kiali : Graph : Drill Down

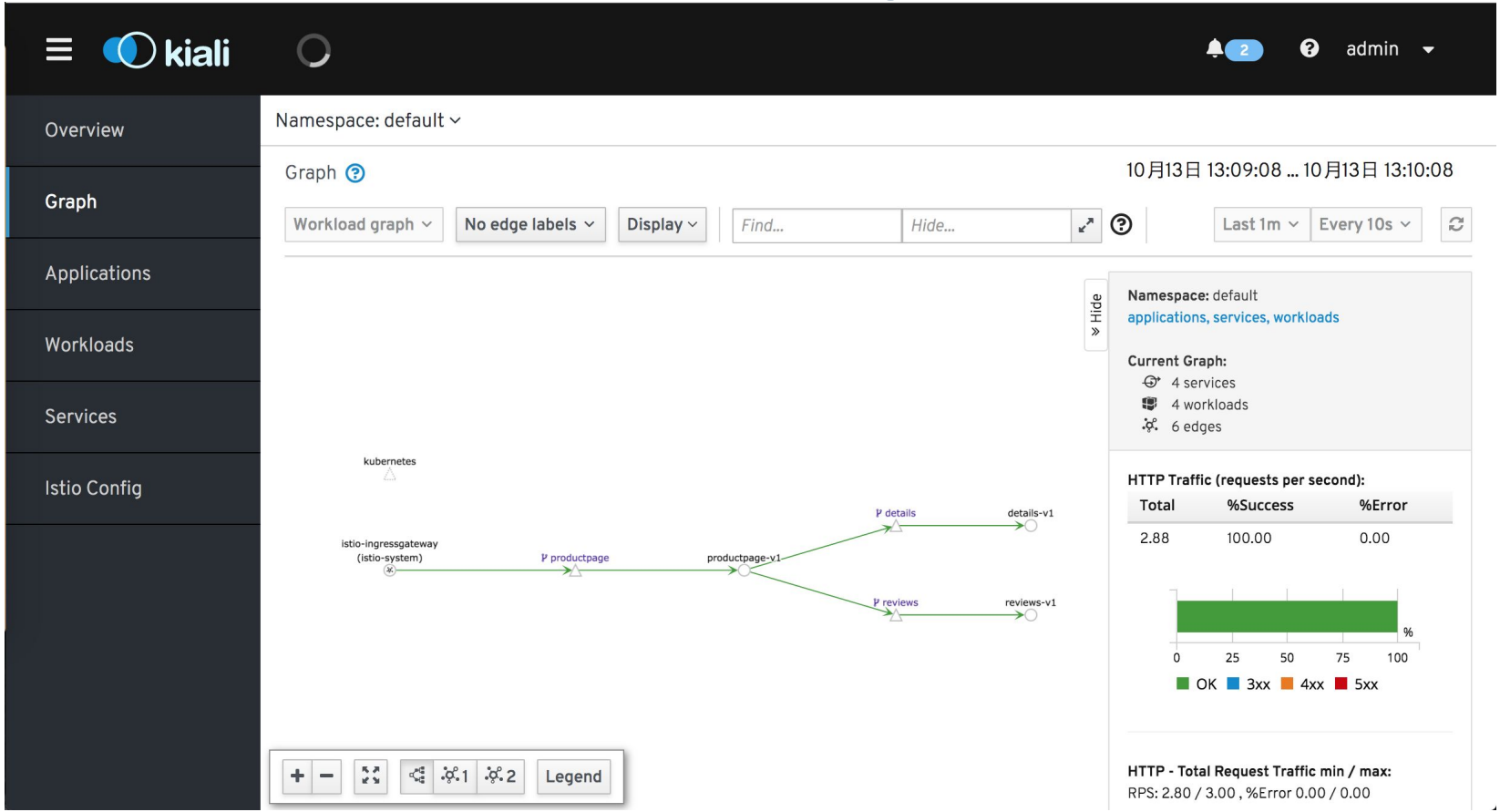


Kiali : Graph : Traffic Animation

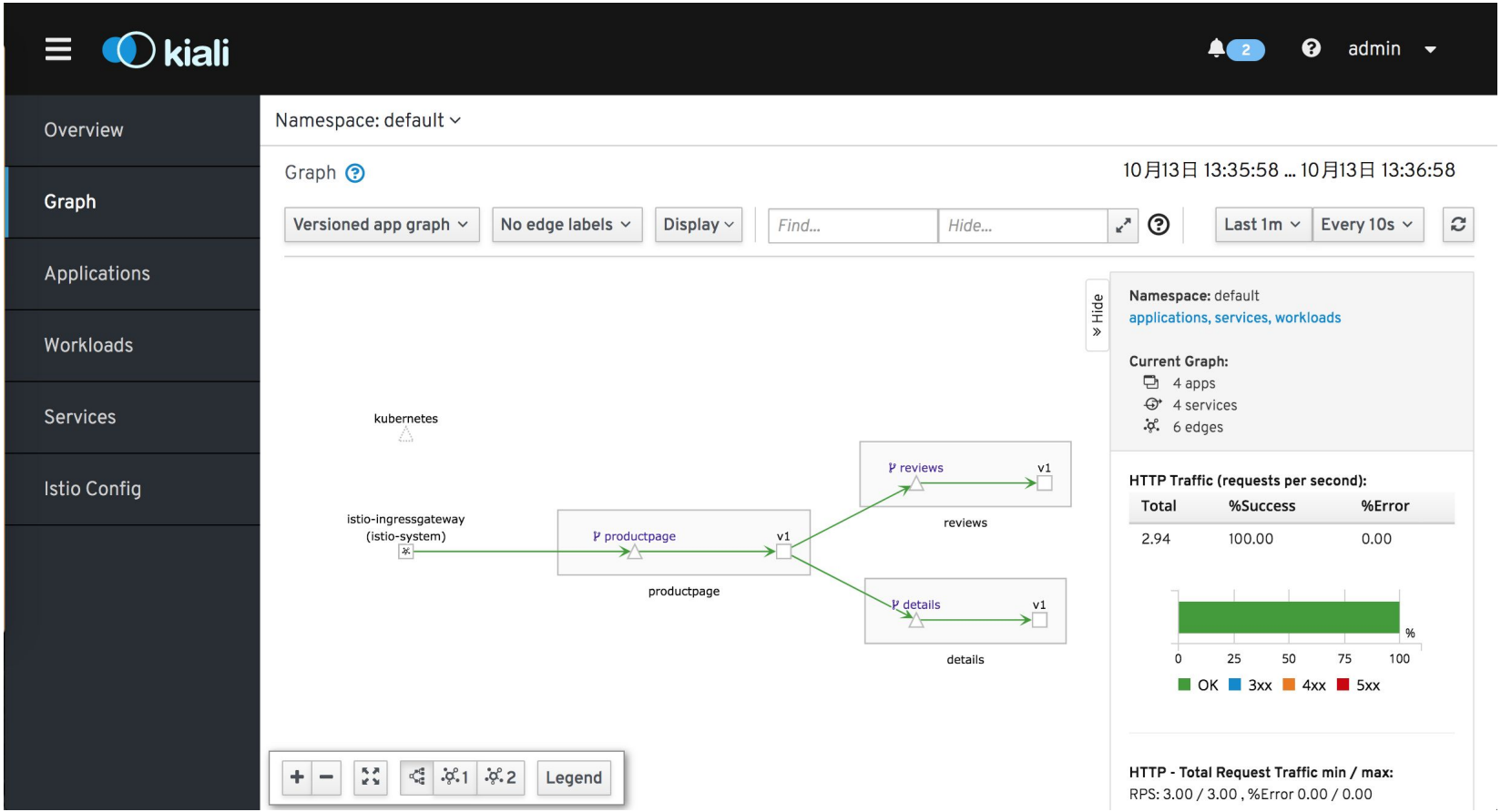


https://www.kiali.io/images/documentation/features/kiali_traffic_animation_thumb.mp4

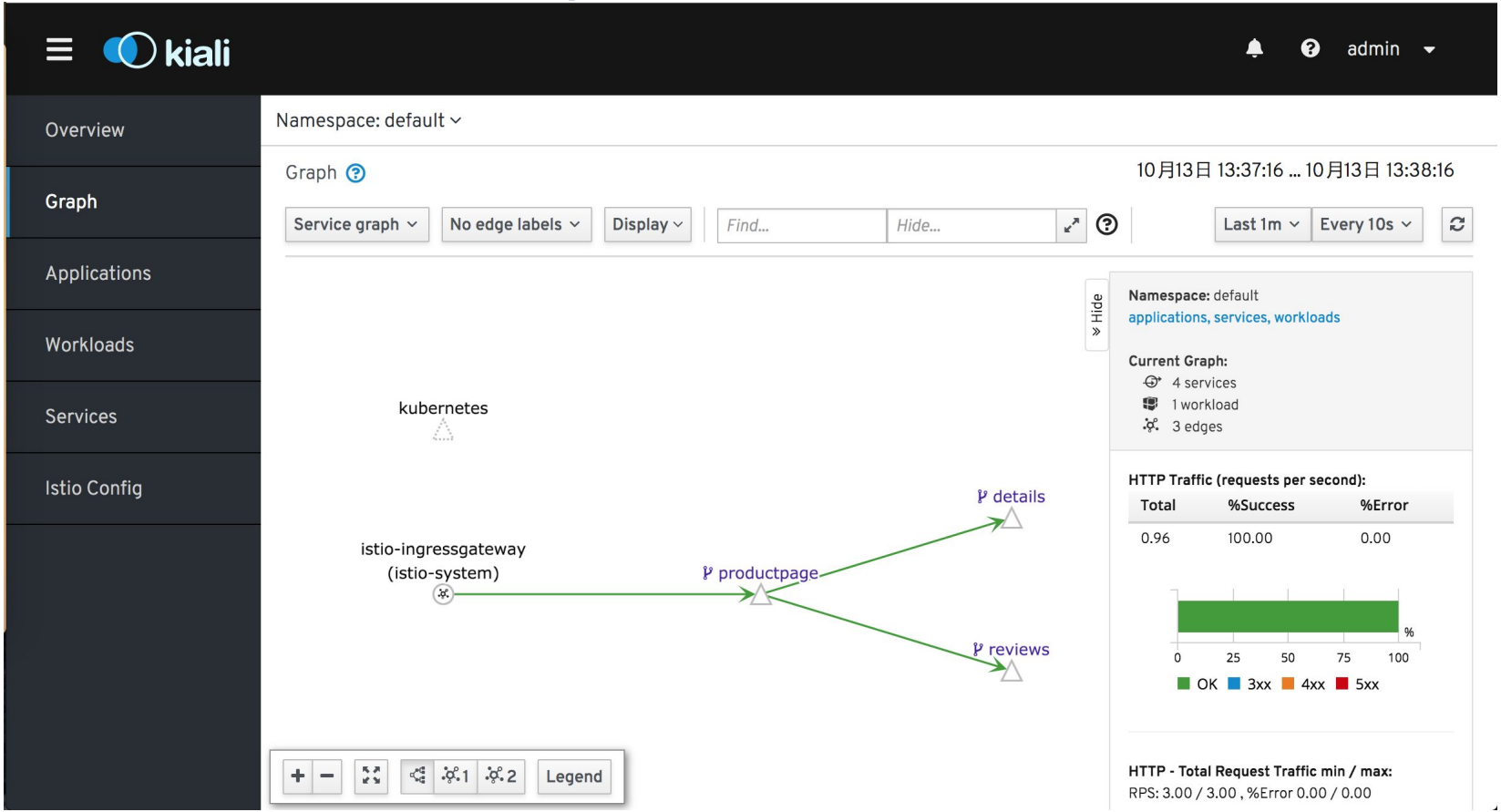
Kiali : Graph : Graph Type : workload



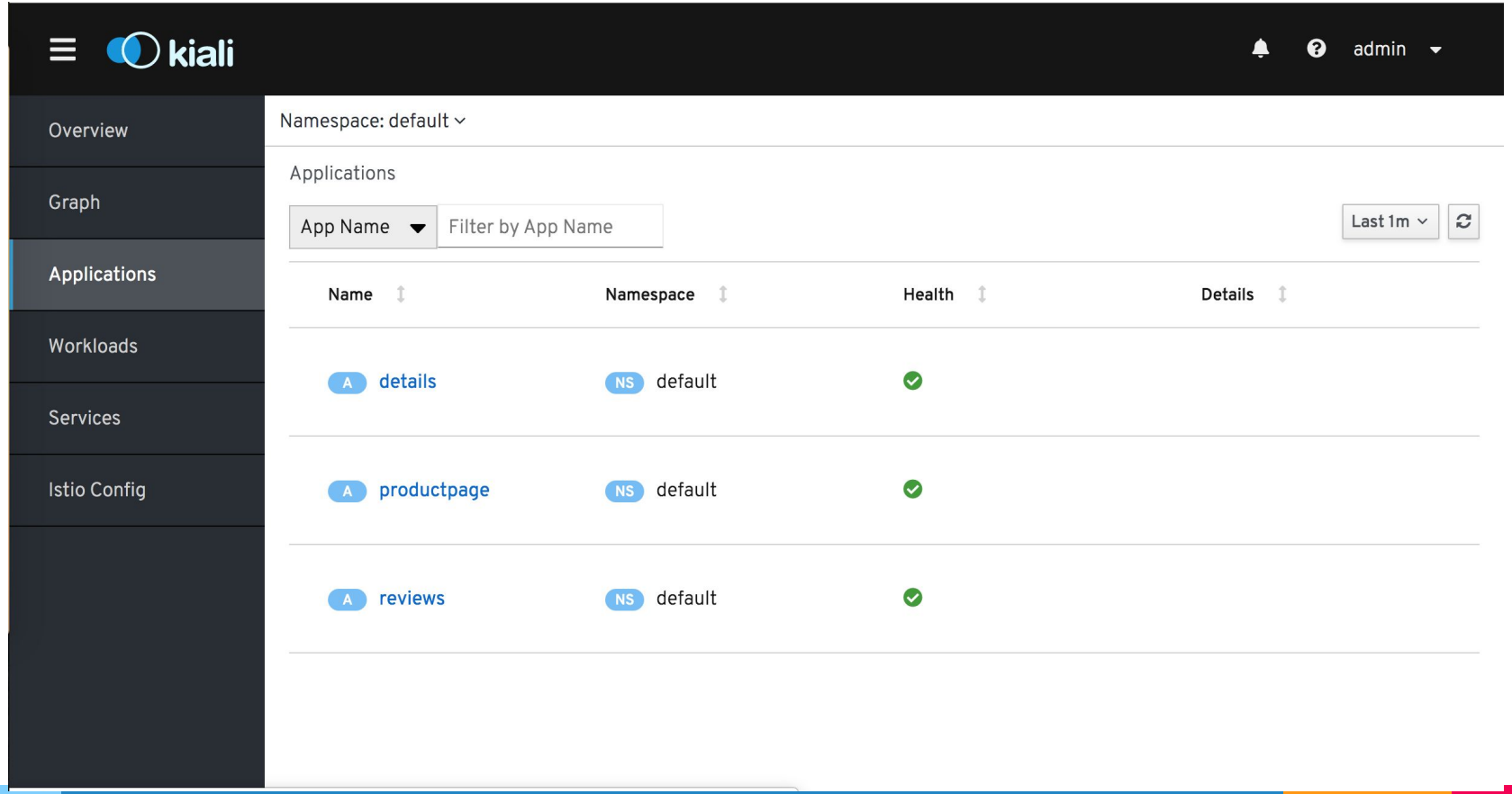
Kiali : Graph Type : versioned app



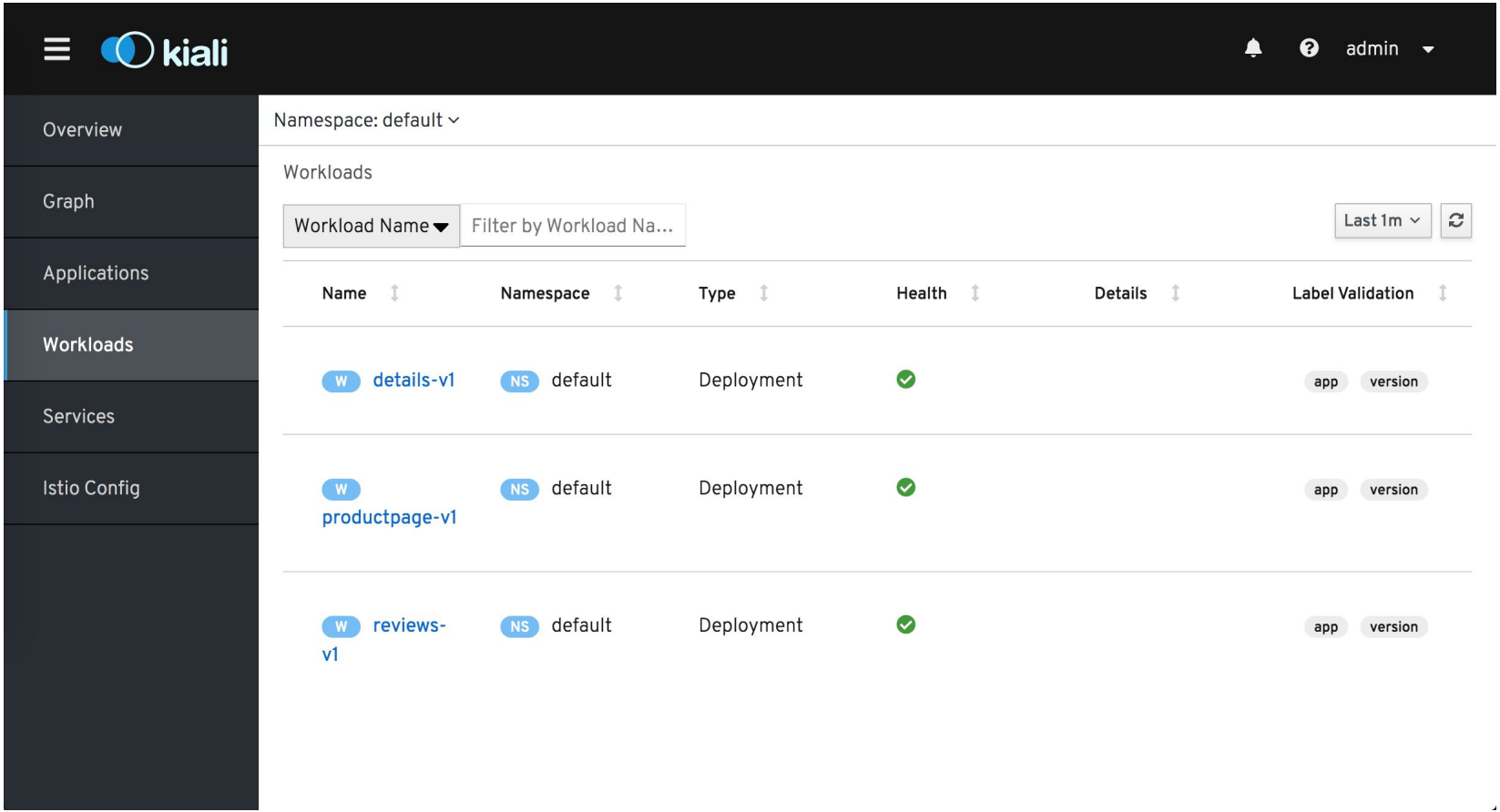
Kiali : Graph Type : service



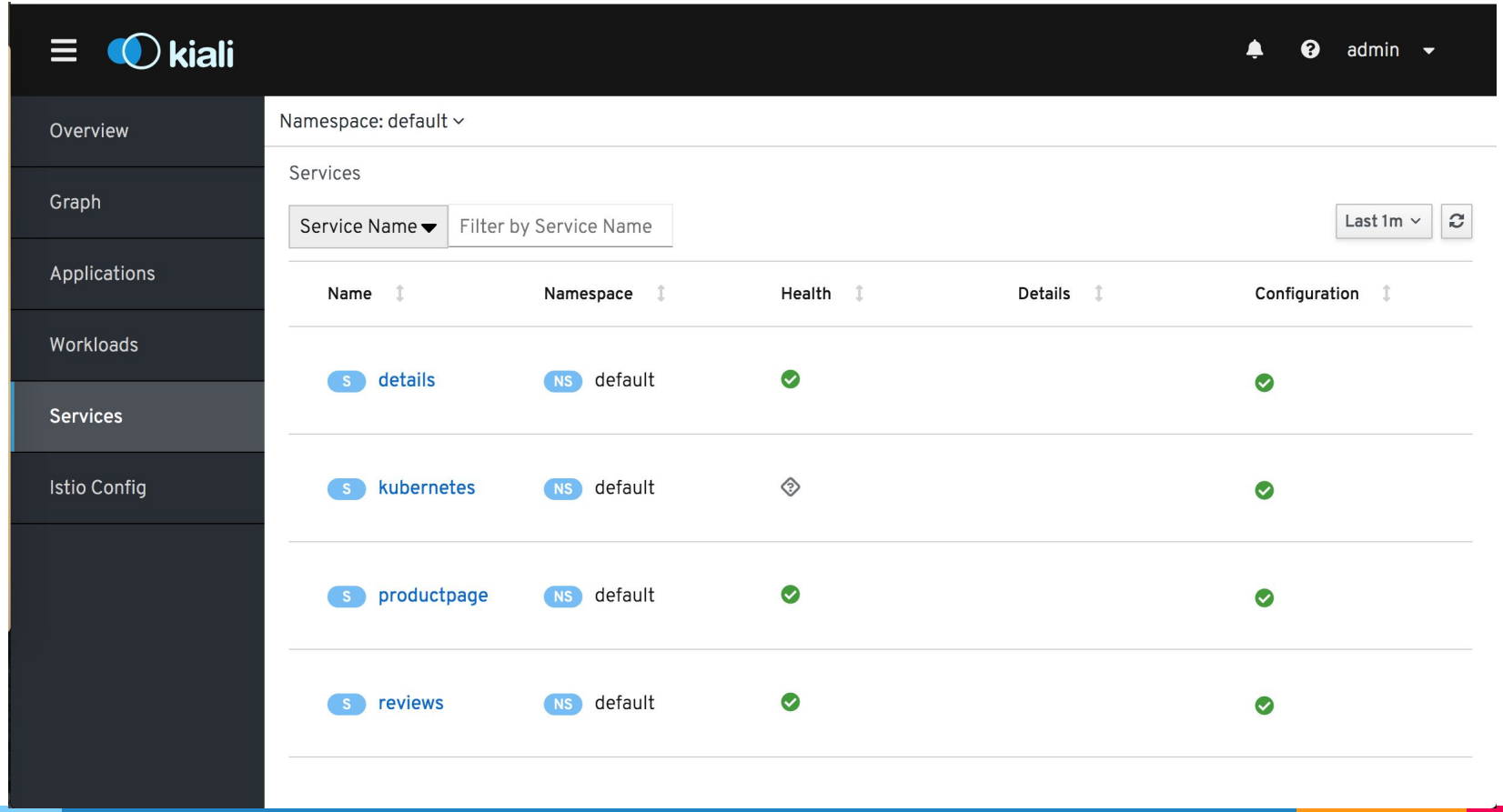
Kiali : Detail Views : Applications



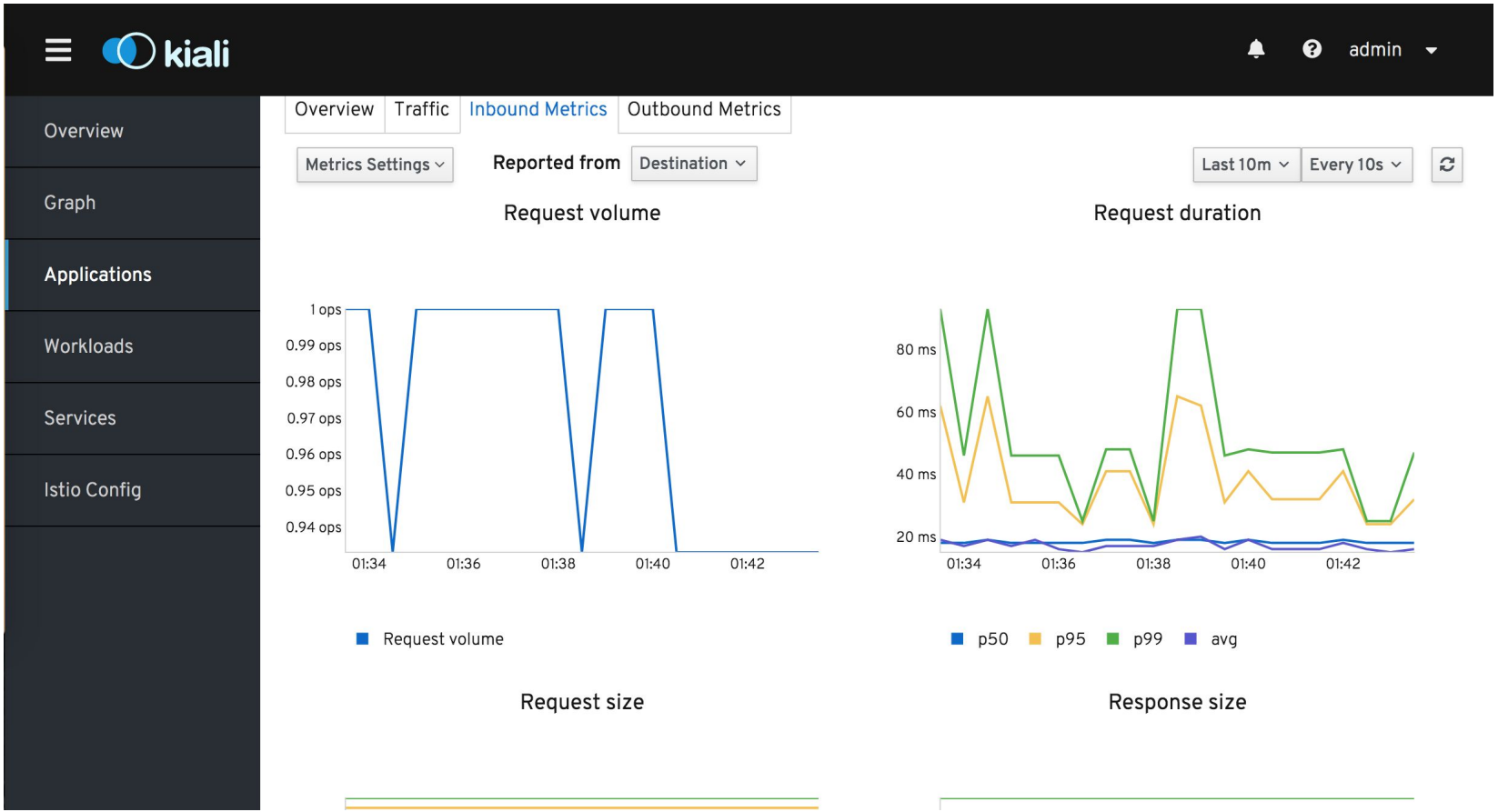
Kiali : Detail Views : Workloads



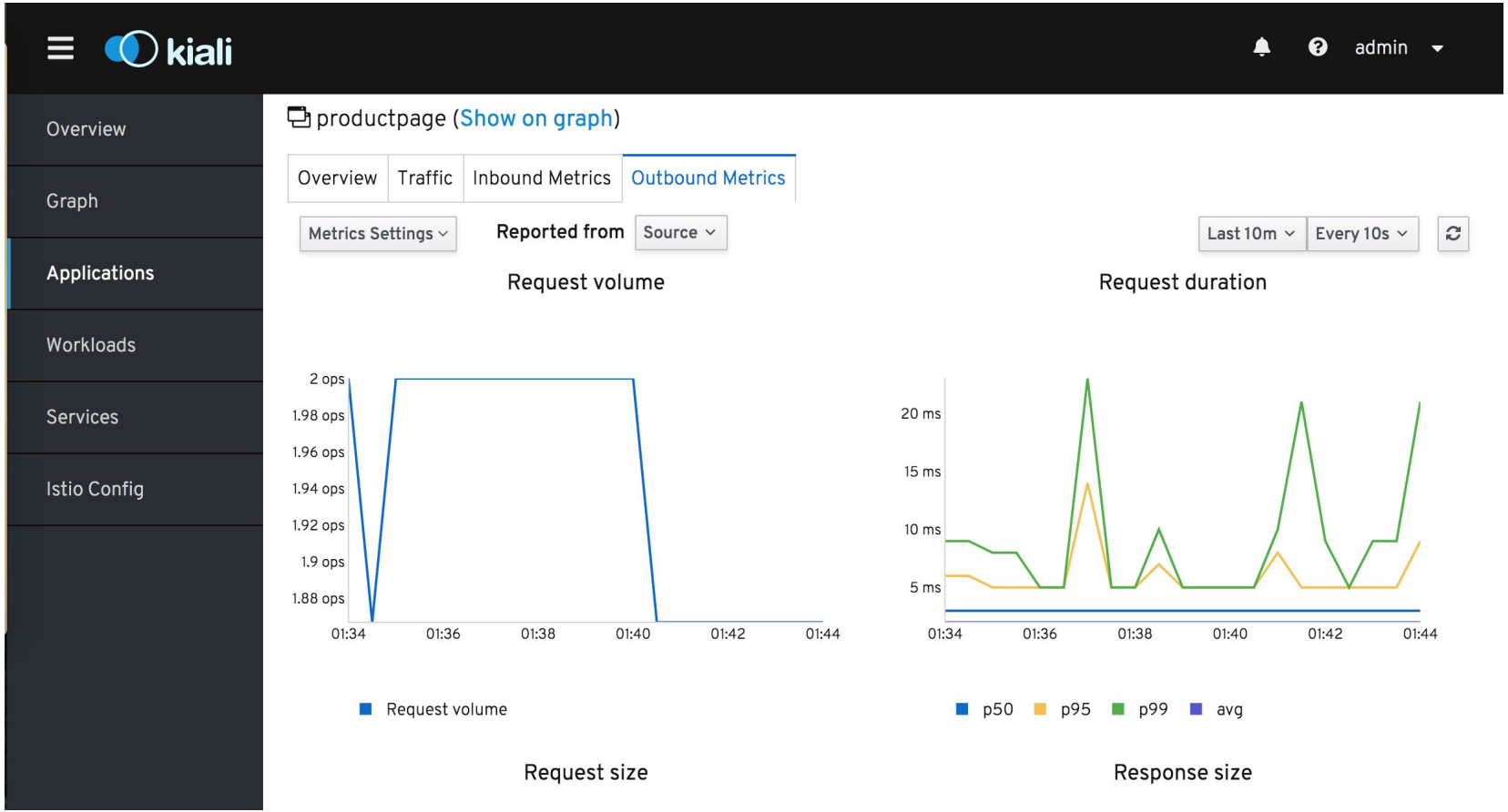
Kiali : Detail Views : Services



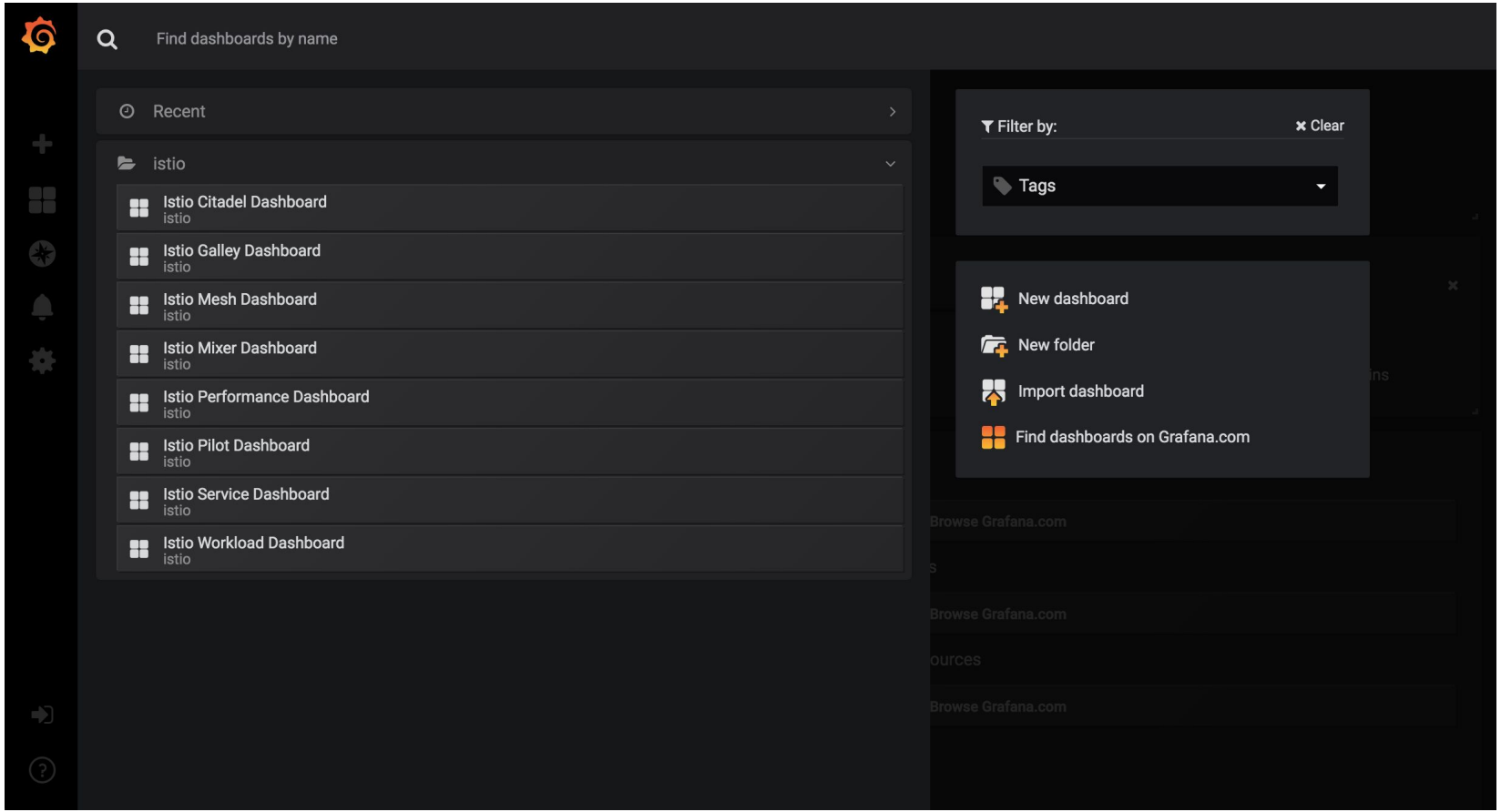
Kiali : Detail Views : Metrics



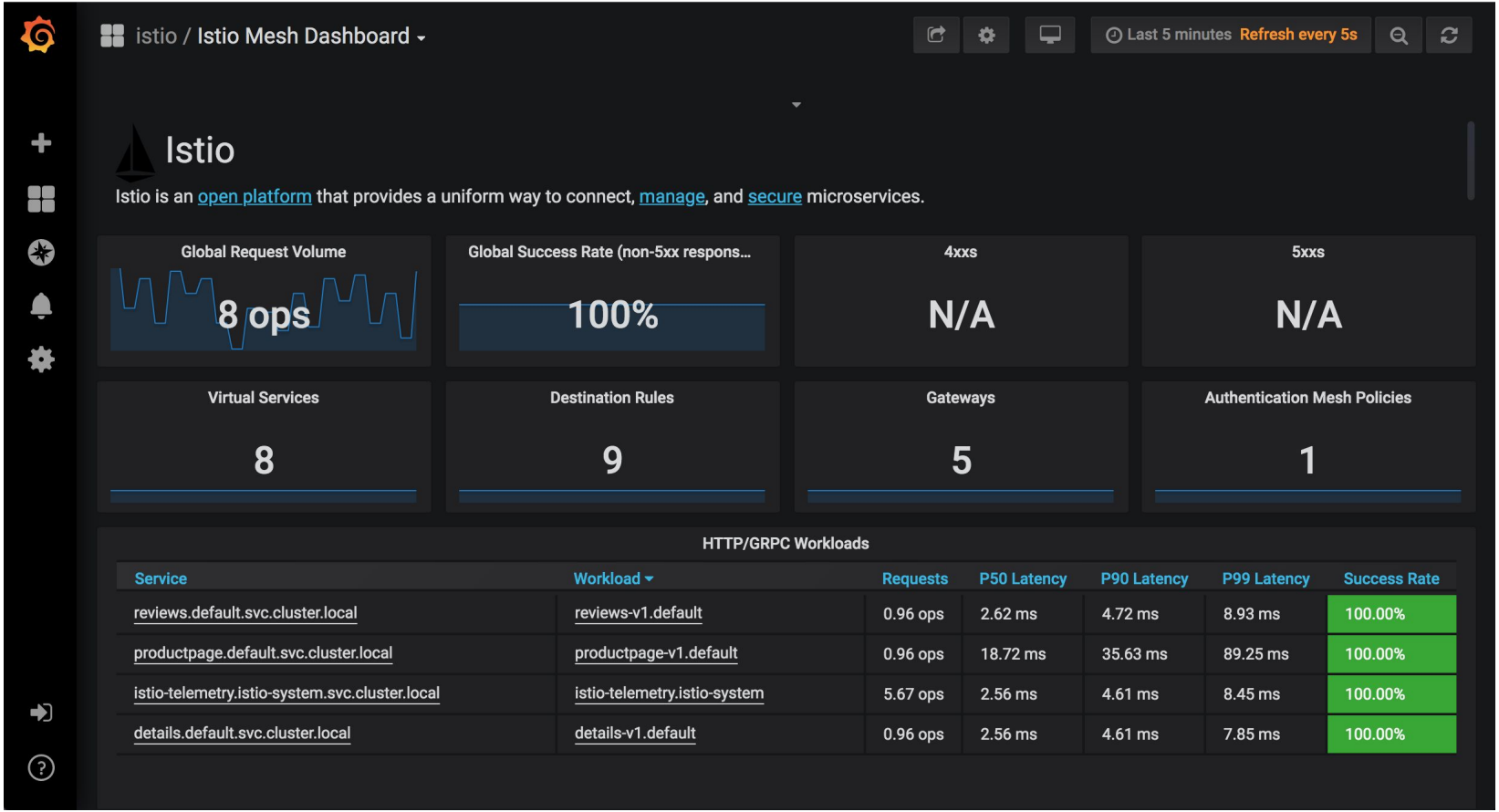
Kiali : Detail ViewsMetrics



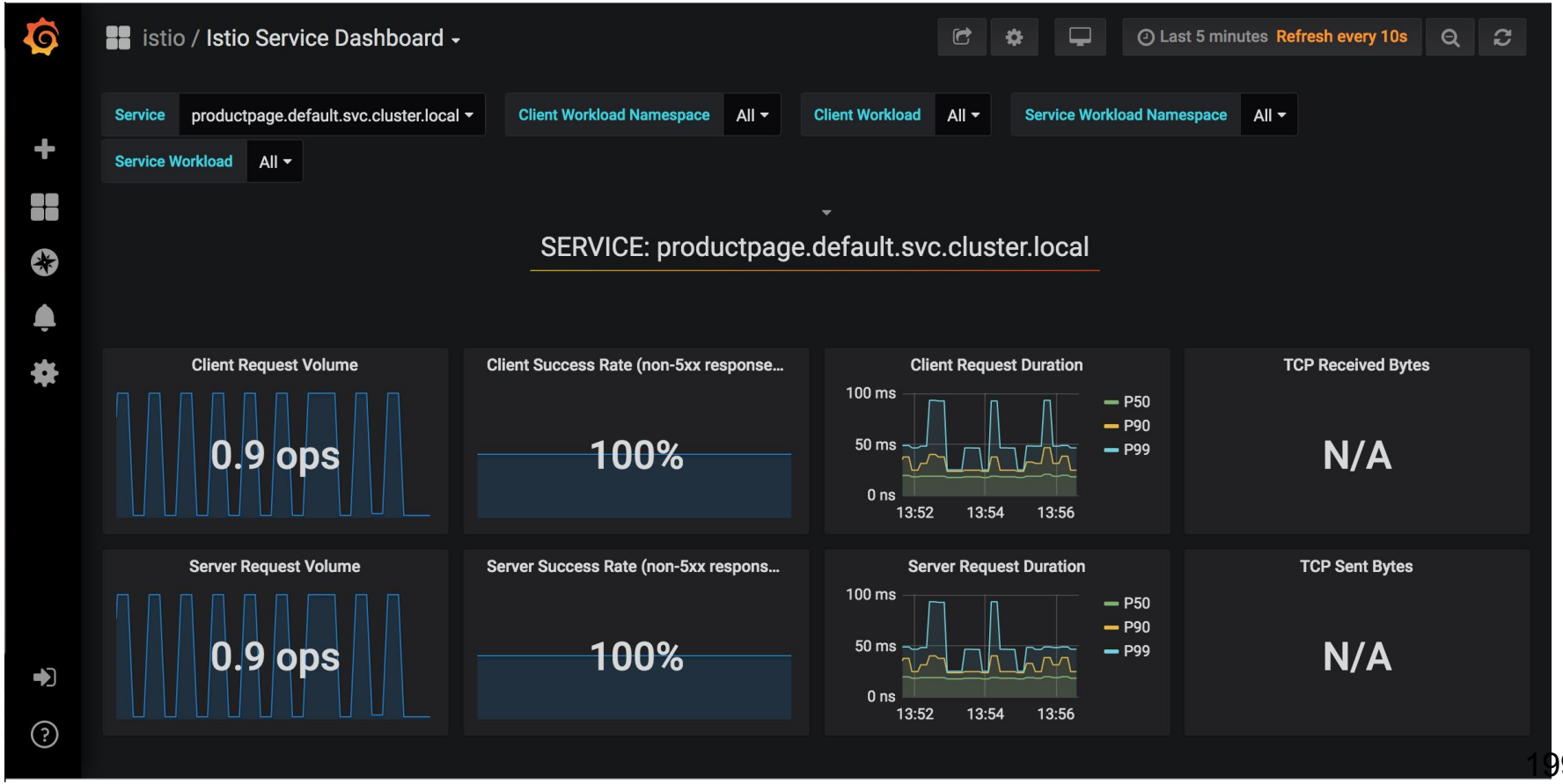
Grafana Dashboard



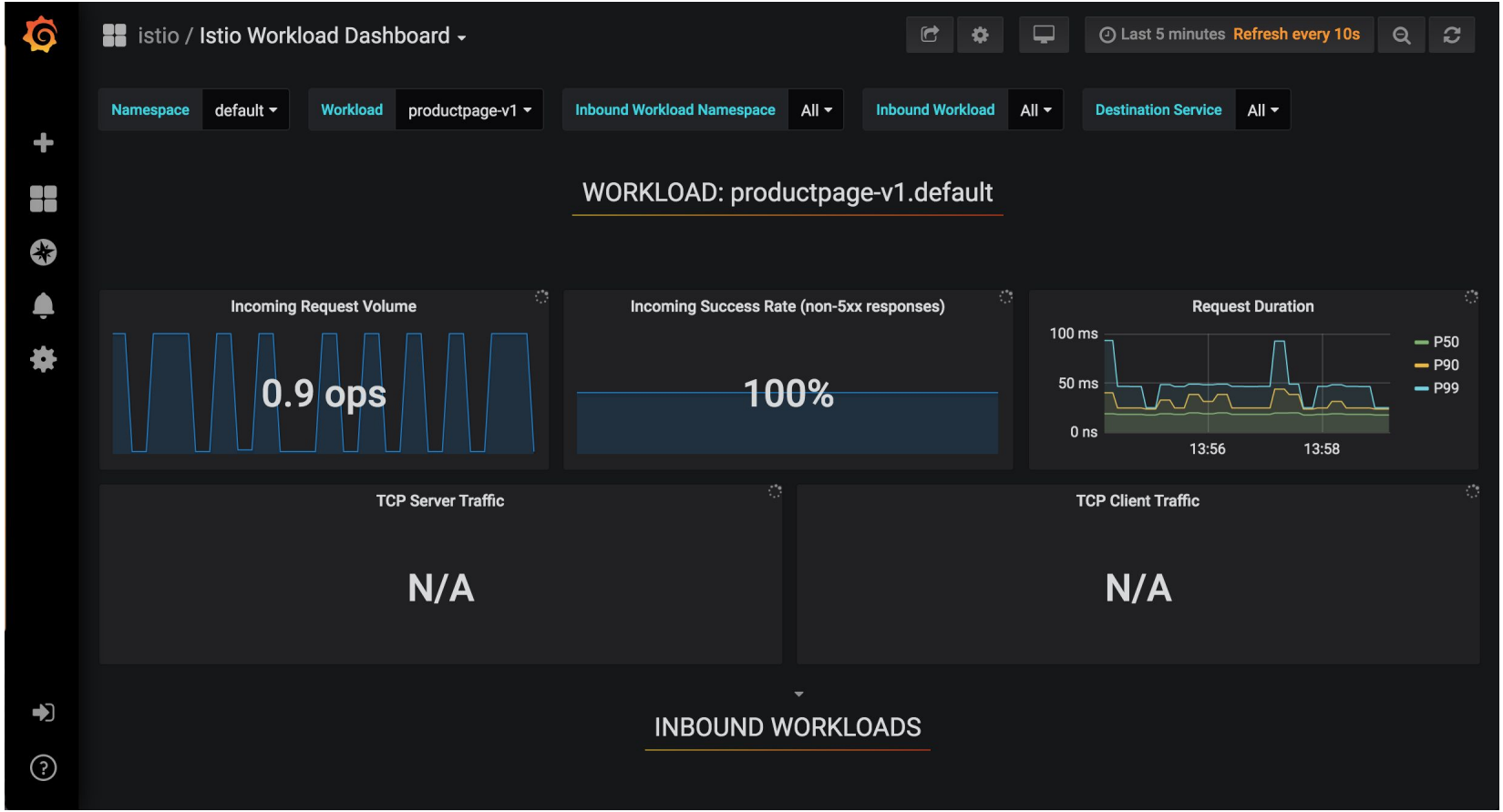
Istio Mesh Dashboard



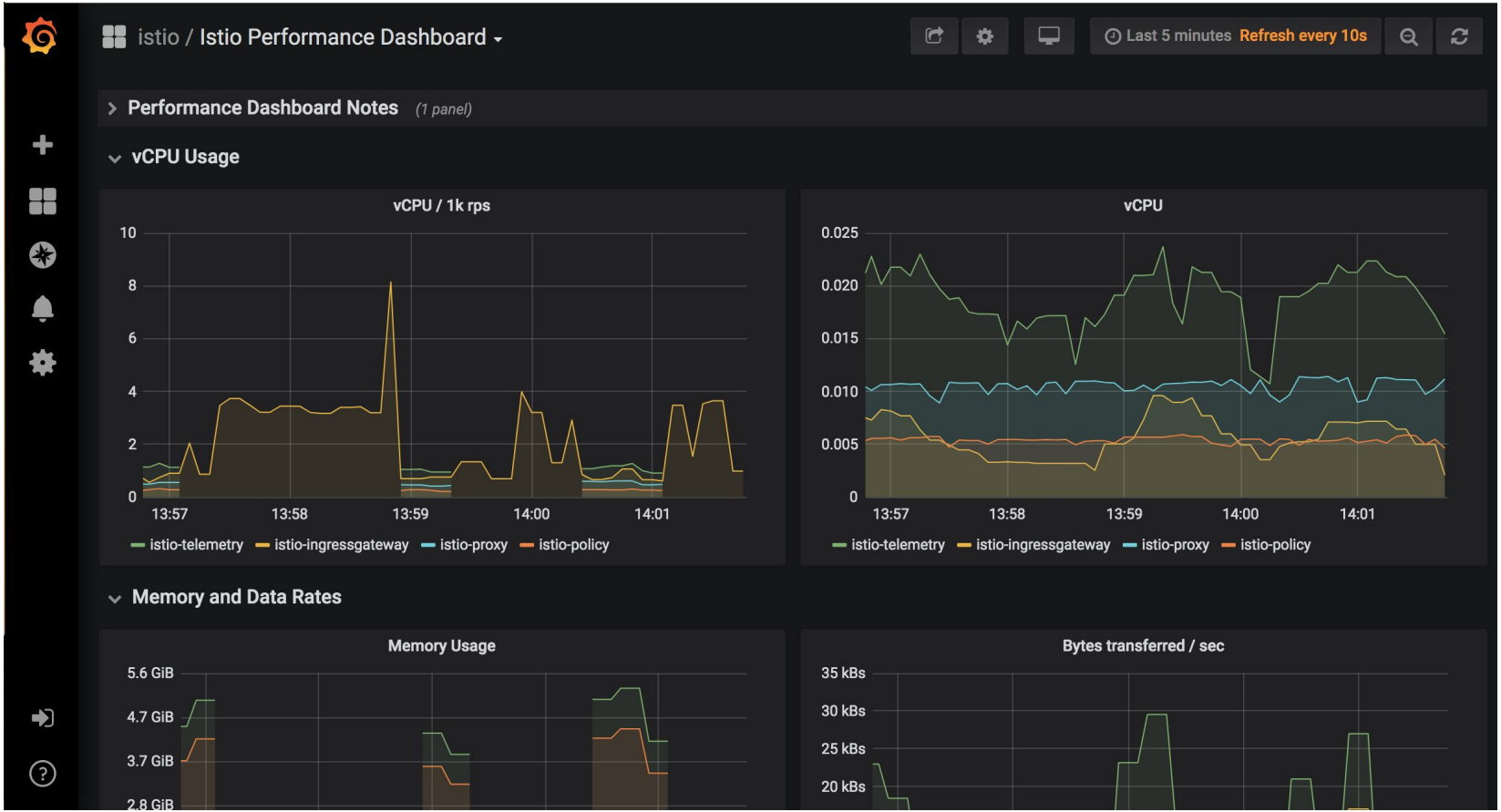
Istio Service Dashboard



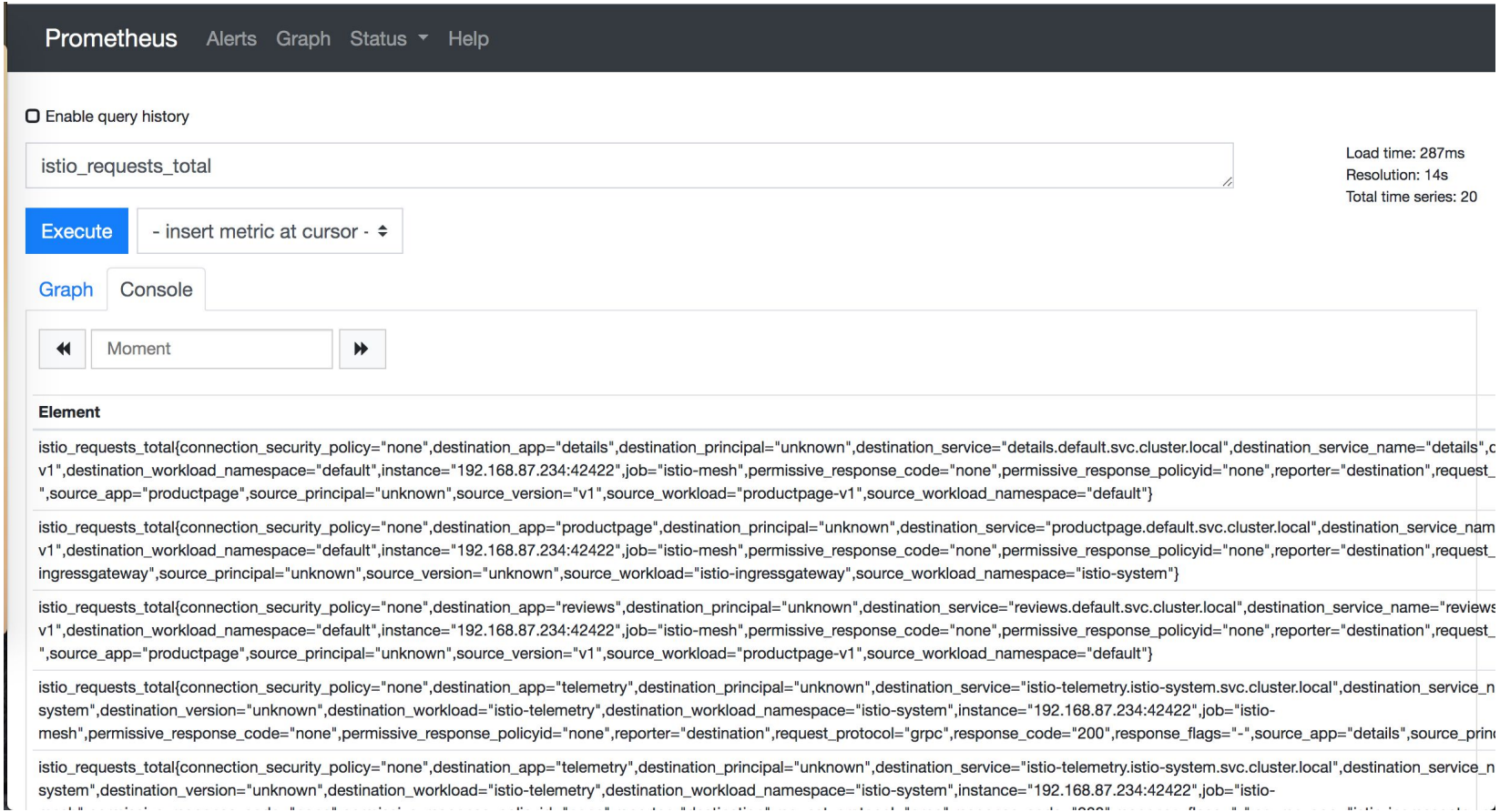
Istio Workload Dashboard



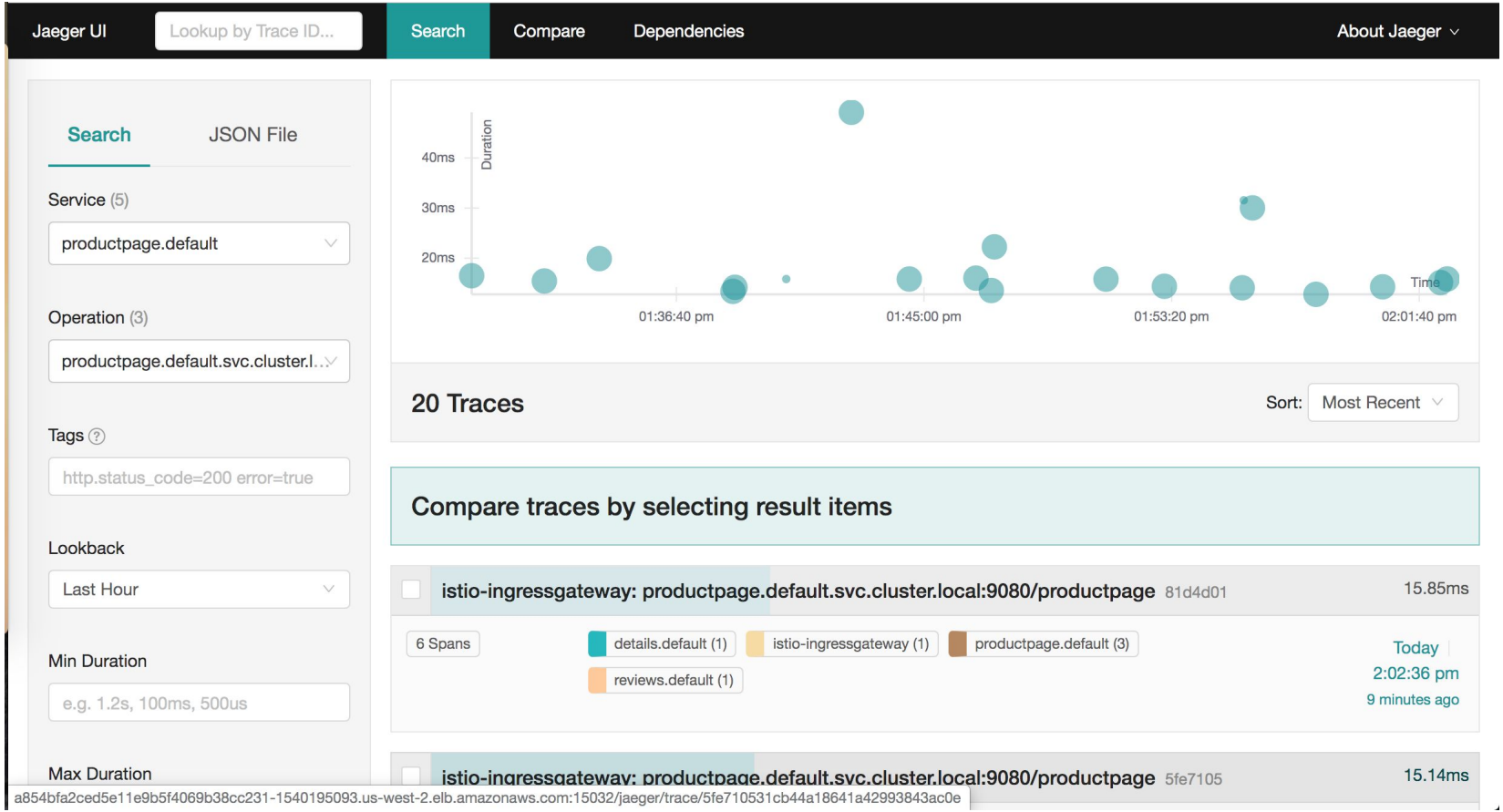
Istio Performance Dashboard



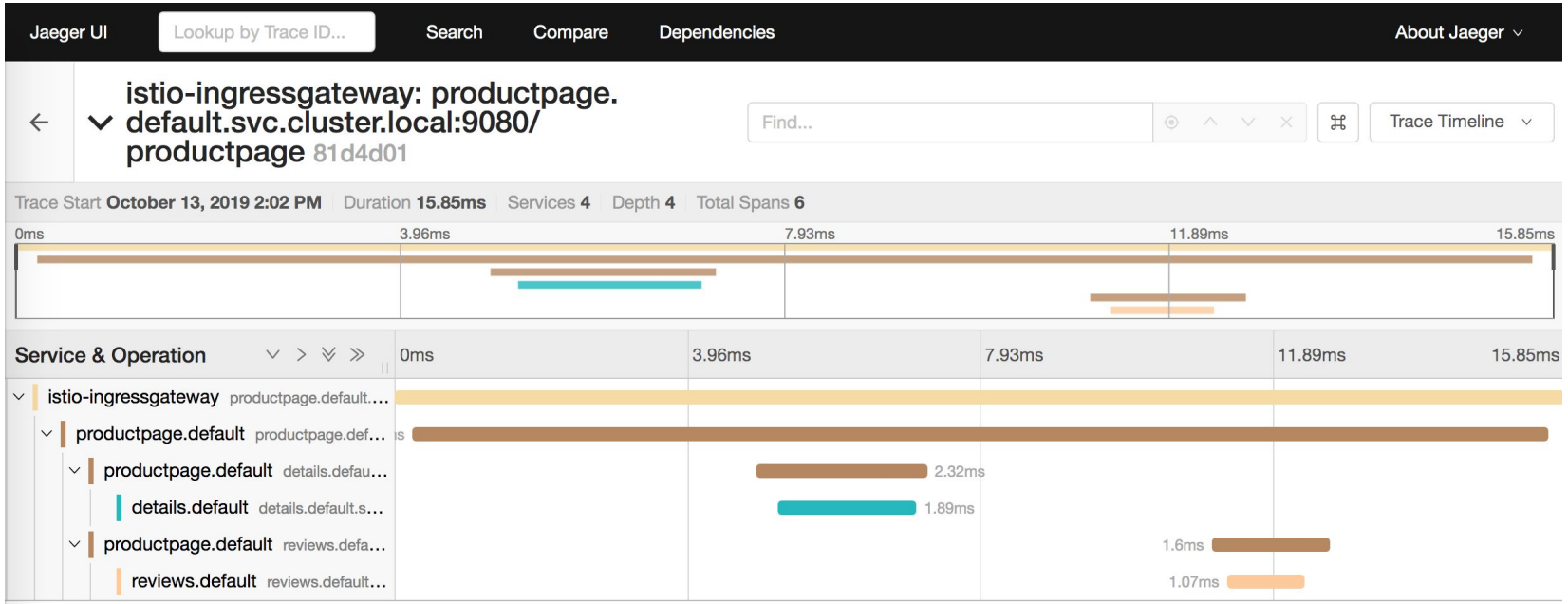
Prometheus


Prometheus
Alerts
Graph
Status
Help
Enable query history
istio_requests_total
Load time: 287ms
Resolution: 14s
Total time series: 20
Execute
- insert metric at cursor -
Graph
Console
Moment
Element
istio_requests_total{connection_security_policy="none",destination_app="details",destination_principal="unknown",destination_service="details.default.svc.cluster.local",destination_service_name="details",c
v1",destination_workload_namespace="default",instance="192.168.87.234:42422",job="istio-mesh",permissive_response_code="none",permissive_response_policyid="none",reporter="destination",request_
",source_app="productpage",source_principal="unknown",source_version="v1",source_workload="productpage-v1",source_workload_namespace="default"}
istio_requests_total{connection_security_policy="none",destination_app="productpage",destination_principal="unknown",destination_service="productpage.default.svc.cluster.local",destination_service_nam
v1",destination_workload_namespace="default",instance="192.168.87.234:42422",job="istio-mesh",permissive_response_code="none",permissive_response_policyid="none",reporter="destination",request_
ingressgateway",source_principal="unknown",source_version="unknown",source_workload="istio-ingressgateway",source_workload_namespace="istio-system"}
istio_requests_total{connection_security_policy="none",destination_app="reviews",destination_principal="unknown",destination_service="reviews.default.svc.cluster.local",destination_service_name="reviews
v1",destination_workload_namespace="default",instance="192.168.87.234:42422",job="istio-mesh",permissive_response_code="none",permissive_response_policyid="none",reporter="destination",request_
",source_app="productpage",source_principal="unknown",source_version="v1",source_workload="productpage-v1",source_workload_namespace="default"}
istio_requests_total{connection_security_policy="none",destination_app="telemetry",destination_principal="unknown",destination_service="istio-telemetry.istio-system.svc.cluster.local",destination_service_n
system",destination_version="unknown",destination_workload="istio-telemetry",destination_workload_namespace="istio-system",instance="192.168.87.234:42422",job="istio-
mesh",permissive_response_code="none",permissive_response_policyid="none",reporter="destination",request_protocol="grpc",response_code="200",response_flags="-",source_app="details",source_prin
istio_requests_total{connection_security_policy="none",destination_app="telemetry",destination_principal="unknown",destination_service="istio-telemetry.istio-system.svc.cluster.local",destination_service_n
system",destination_version="unknown",destination_workload="istio-telemetry",destination_workload_namespace="istio-system",instance="192.168.87.234:42422",job="istio-

Tracing


Jaeger UI
Lookup by Trace ID...
Search
Compare
Dependencies
About Jaeger
Search
JSON File
Service (5)
productpage.default
Operation (3)
productpage.default.svc.cluster.l...
Tags
http.status_code=200 error=true
Lookback
Last Hour
Min Duration
e.g. 1.2s, 100ms, 500us
Max Duration
Duration
40ms
30ms
20ms
01:36:40 pm
01:45:00 pm
01:53:20 pm
02:01:40 pm
Time
20 Traces
Sort:
Most Recent
Compare traces by selecting result items
istio-ingressgateway: productpage.default.svc.cluster.local:9080/productpage 81d4d01
15.85ms
6 Spans
details.default (1)
istio-ingressgateway (1)
productpage.default (3)
reviews.default (1)
Today
2:02:36 pm
9 minutes ago
istio-ingressgateway: productpage.default.svc.cluster.local:9080/productpage 5fe7105
15.14ms
a854bfa2ced5e11e9b5f4069b38cc231-1540195093.us-west-2.elb.amazonaws.com:15032/jaeger/trace/5fe710531cb44a18641a42993843ac0e

Tracing

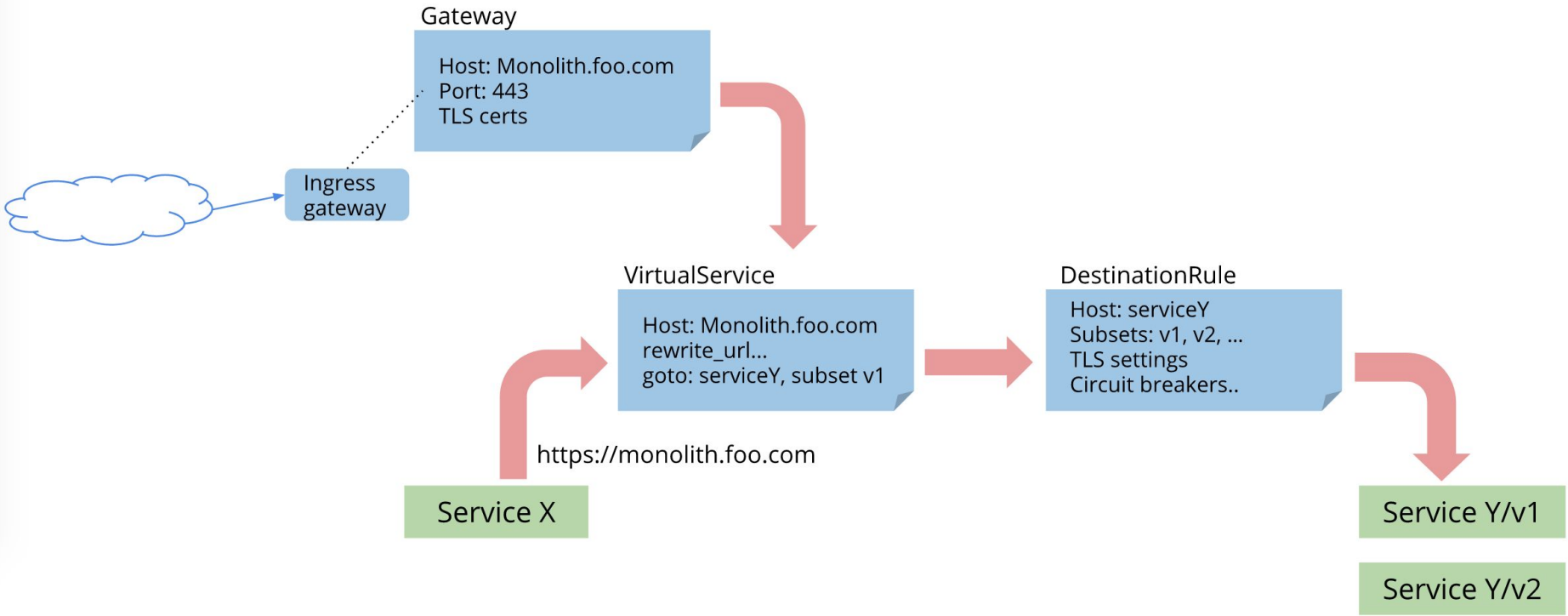

Jaeger UI
Lookup by Trace ID...
Search
Compare
Dependencies
About Jaeger
istio-ingressgateway: productpage.default.svc.cluster.local:9080/productpage 81d4d01
Find...
Trace Timeline
Trace Start October 13, 2019 2:02 PM
Duration 15.85ms
Services 4
Depth 4
Total Spans 6
0ms
3.96ms
7.93ms
11.89ms
15.85ms
Service & Operation
istio-ingressgateway productpage.default....
productpage.default productpage.def...
productpage.default details.defau...
details.default details.default.s...
productpage.default reviews.defa...
reviews.default reviews.default...
2.32ms
1.89ms
1.6ms
1.07ms

Traffice Management Setting

Istio routing configuration

- Gateway
- VirtualService
- DestinationRule
- ServiceEntry

Routing configuration



Gateway

- overcomes the Ingress shortcomings
- separating L4-L6 spec from L7.

```yaml
apiVersion: networking.istio.io/v1alpha3
kind: Gateway
metadata:
  name: bookinfo-gateway
spec:
  servers:
  - port:
      number: 443
      name: https
      protocol: HTTPS
    hosts:
    - bookinfo.com
    tls:
      mode: SIMPLE
      serverCertificate: /tmp/tls.crt
      privateKey: /tmp/tls.key
```

VirtualService

- must be defined for the same host and bound to the Gateway configuration

```yaml
apiVersion: networking.istio.io/v1alpha3
kind: VirtualService
metadata:
  name: bookinfo
spec:
  hosts:
    - bookinfo.com
  gateways:
  - bookinfo-gateway # <---- bind to gateway
  http:
  - match:
    - uri:
        prefix: /reviews
    route:
    ...
```

VirtualService

- Integrate RouteRule now
- hosts can be specified using wildcard for all matching services.

```
apiVersion: networking.istio.io/v1alpha3
kind: VirtualService
metadata:
  name: bookinfo
spec:
  hosts:
    - bookinfo.com
  http:
  - match:
    - uri:
        prefix: /reviews
    route:
    - destination:
        host: reviews
  - match:
    - uri:
        prefix: /ratings
    route:
    - destination:
        host: ratings
  ...
```

DestinationRule

- DestinationRule defines addressable subsets of the corresponding destination host.

```yaml
apiVersion: networking.istio.io/v1alpha3
kind: DestinationRule
metadata:
  name: reviews
spec:
  host: reviews
  trafficPolicy:
    loadBalancer:
      simple: RANDOM
  subsets:
  - name: v1
    labels:
      version: v1
  - name: v2
    labels:
      version: v2
    trafficPolicy:
      loadBalancer:
        simple: ROUND_ROBIN
  - name: v3
    labels:
      version: v3
```

ServiceEntry

- To add external entries into the service registry in Istio

```yaml
apiVersion: networking.istio.io/v1alpha3
kind: ServiceEntry
metadata:
  name: foo-ext
spec:
  hosts:
  - foo.com
  ports:
  - number: 80
    name: http
    protocol: HTTP
```

Demo

- External Traffic
- Canary Deployment
- Fault Injection
- Telemetry

```
$(~/environment/eks-workshop/service-mesh/02-canary-percent
)

$ ./install.sh

$ ../get-links.sh

$ ../ab.sh productpage

$ ./uninstall.sh
```

```
$(~/environment/eks-workshop/service-mesh/03-canary-header)

$ ./install.sh

$ ../get-links.sh

$ ../ab.sh productpage

$ ./uninstall.sh
```

```
$(~/environment/eks-workshop/service-mesh/04-fault-injection)

$ ./install.sh

$ ../get-links.sh

$ ../ab.sh productpage

$ ./uninstall.sh
```

```
$(~/environment/eks-workshop/service-mesh/05-telemetry-log)

$ ./install.sh

$ ../get-links.sh

$ ./show-log.sh

$ ./uninstall.sh
```

Thanks!

Any questions?

You can find me at: