10 Kube Commandments

We've been in the game for years

That in itself is admirable

There's rules to this biz

We wrote y'all a manual

A step-by-step conf talk for you to get...

Your clusters on track

And not your releases pushed back

Bryan Liles

Staff Software Engineer
Heptio

Lots of years Years of Experience
@bryanl

Carlos Amedee

Senior Software Engineer
DigitalOcean

Observability
Cloud Compute Services
Systems Engineering
@cagedmantis

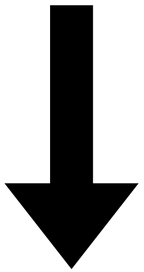
Rule Number Uno

To go fast, you must start slow

Rule Number Uno
To go fast, you must start
deliberately

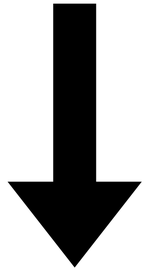
Public Cloud **Datacenter** **Your Desktop**

Public Cloud



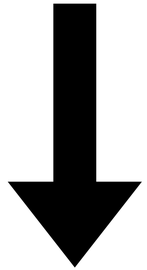
- GKE on Google Cloud
- AKS on Azure
- *lots of vendors*

Datacenter



- kubeadm
- *lots of vendors*

Your Desktop



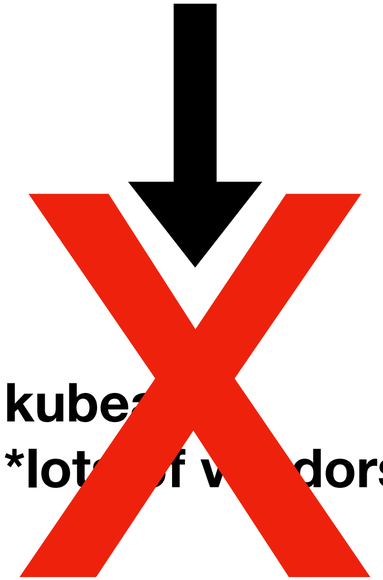
- Minikube
- Minik8s
- Docker for Mac or Windows

Not Declarative

Public Cloud

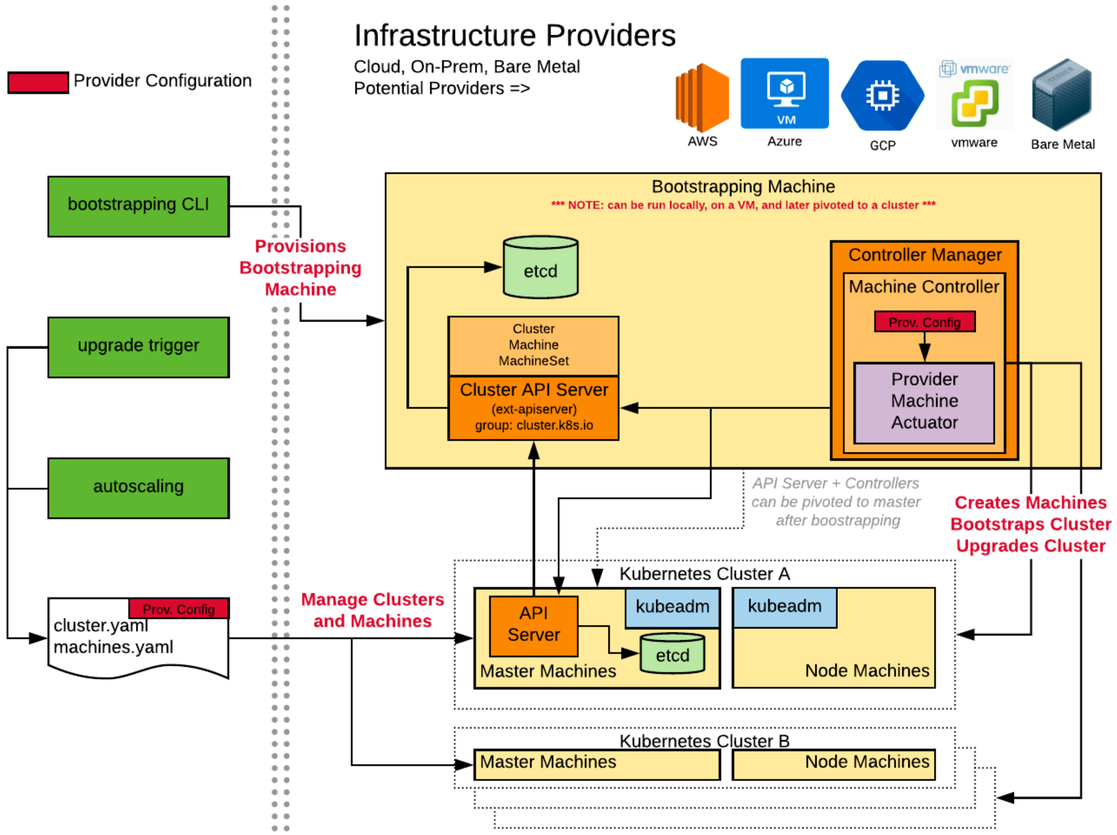
- GKE on Google Cloud
- AKS on Azure
- *lots of vendors*

Datacenter



- kubeadm
- *lots of vendors*

Cluster API


Provider Configuration
Infrastructure Providers
Cloud, On-Prem, Bare Metal
Potential Providers =>
AWS
Azure
GCP
vmware
Bare Metal
bootstrapping CLI
upgrade trigger
autoscaling
Prov. Config
cluster.yaml
machines.yaml
Provisions Bootstrapping Machine
Bootstrapping Machine
*** NOTE: can be run locally, on a VM, and later pivoted to a cluster ***
etcd
Cluster
Machine
MachineSet
Cluster API Server
(ext-apiserver)
group: cluster.k8s.io
Controller Manager
Machine Controller
Prov. Config
Provider
Machine
Actuator
API Server + Controllers can be pivoted to master after boostrapping
Creates Machines
Bootstraps Cluster
Upgrades Cluster
Manage Clusters and Machines
Kubernetes Cluster A
API Server
kubeadm
kubeadm
etcd
Master Machines
Node Machines
Kubernetes Cluster B
Master Machines
Node Machines

Number Two
Always let them know your next move

Your next move is the images you'll deploy to your cluster


Build Image
Host Image

docker build

docker build

- buildah
- img
- GCP Container Builder

Why are you still building your containers with root privileges?

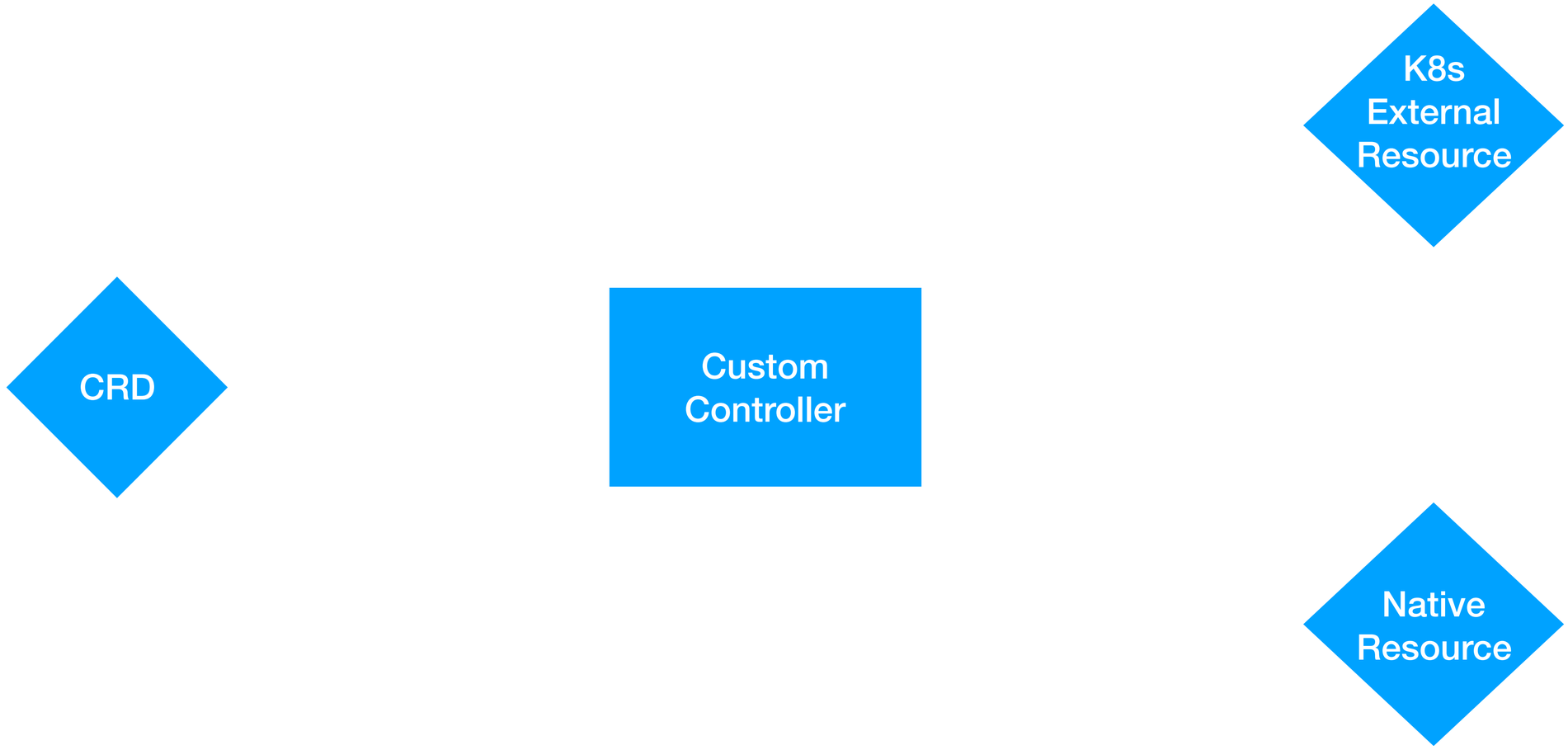
Rule Number Three
Never trust nobody: Hookup up that Pod Security Policy

```yaml
apiVersion: extensions/v1beta1
kind: PodSecurityPolicy
metadata:
  name: how-not-to-get-robbed
spec:
  privileged: false
  runAsUser:
    rule: MustRunAsNonRoot
  seLinux:
    rule: RunAsAny
  fsGroup:
    rule: RunAsAny
  supplementalGroups:
    rule: RunAsAny
  volumes:
    - nfs
```

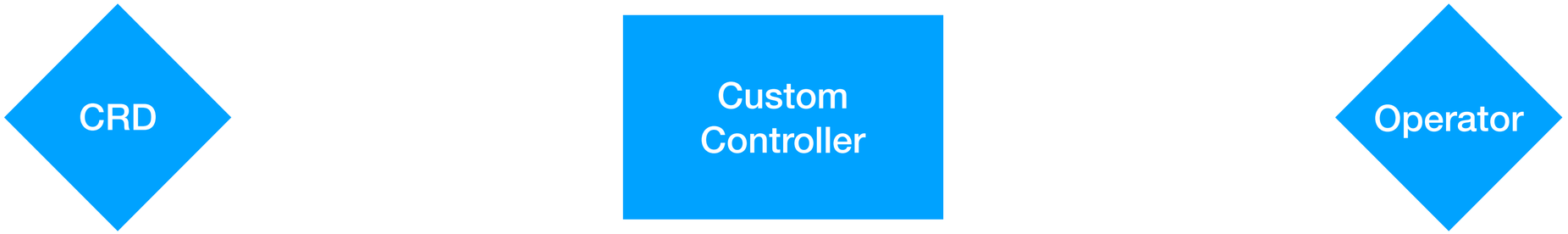
Number Four
I know you heard this before: Never get high off what Kube supplies

Custom Resource Definition

Custom Resources Pattern



Custom Resources Pattern



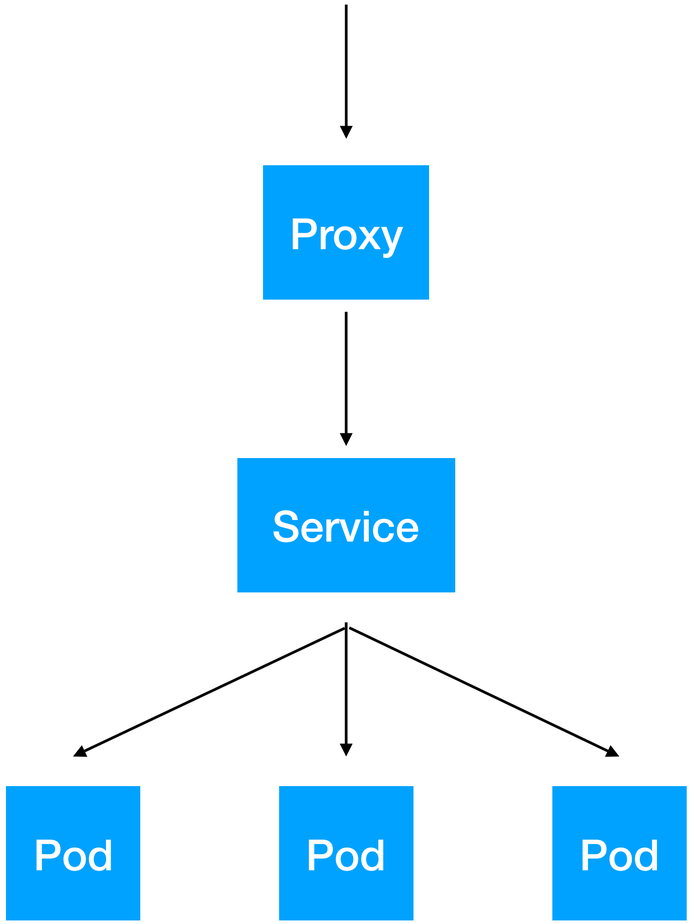
Custom Resources Pattern

Rule Number Five
Communicating With Pods
Never Mix Internal and External Traffic

Ingress Traffic

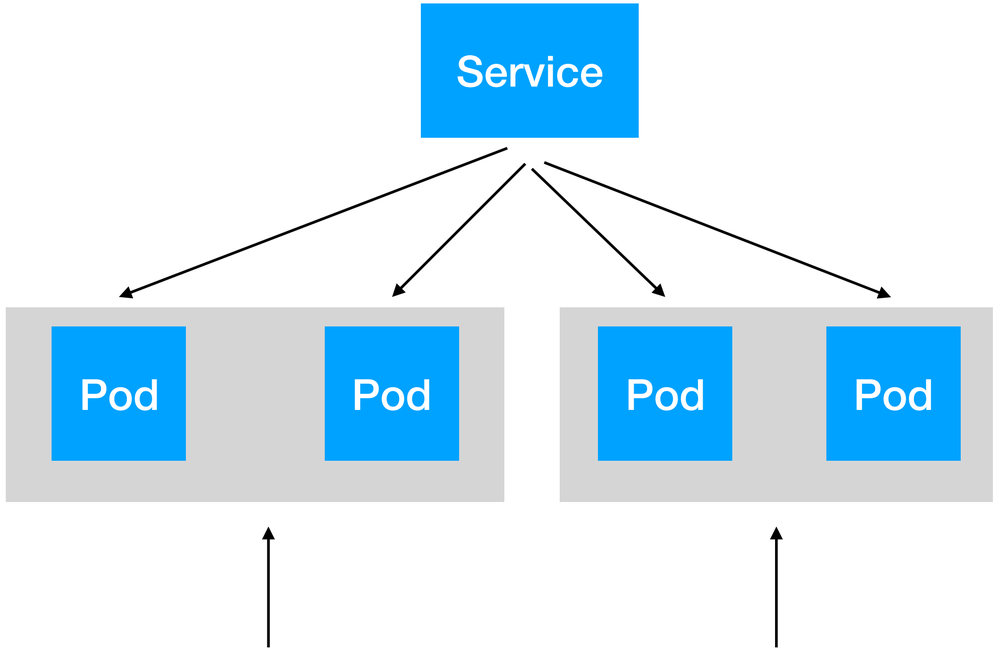
Cluster IP

```
apiVersion: v1
kind: Service
metadata:
  name: sample-service
spec:
  selector:
    app: sample-app
  type: ClusterIP
  ports:
  - name: http
    port: 80
    targetPort: 80
    protocol: TCP
```



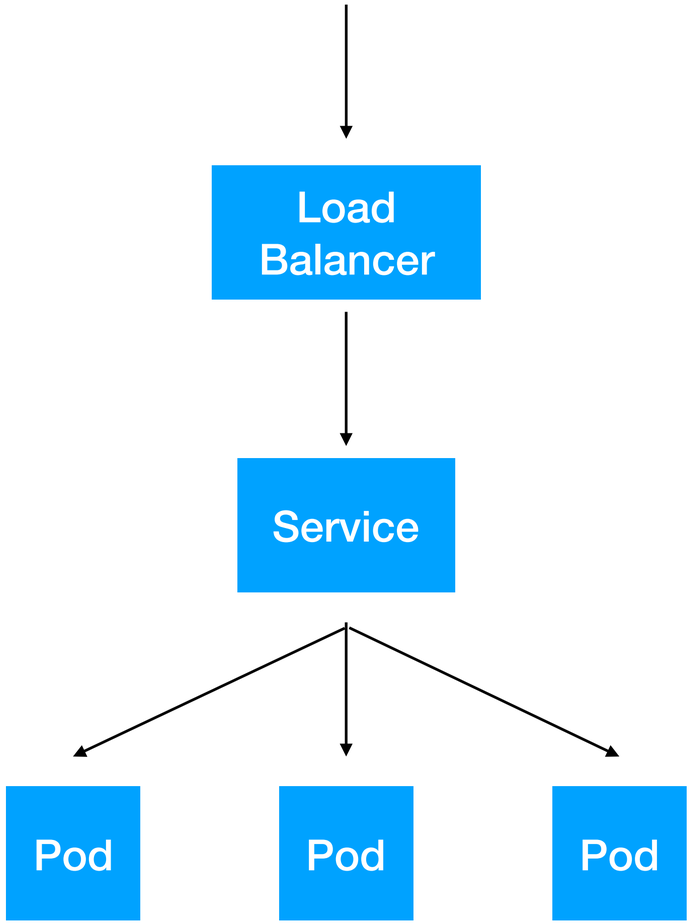
Node Port

```
apiVersion: v1
kind: Service
metadata:
  name: my-nodeport-service
spec:
  selector:
    app: my-app
  type: NodePort
  ports:
  - name: http
    port: 80
    targetPort: 80
    nodePort: 30036
    protocol: TCP
```



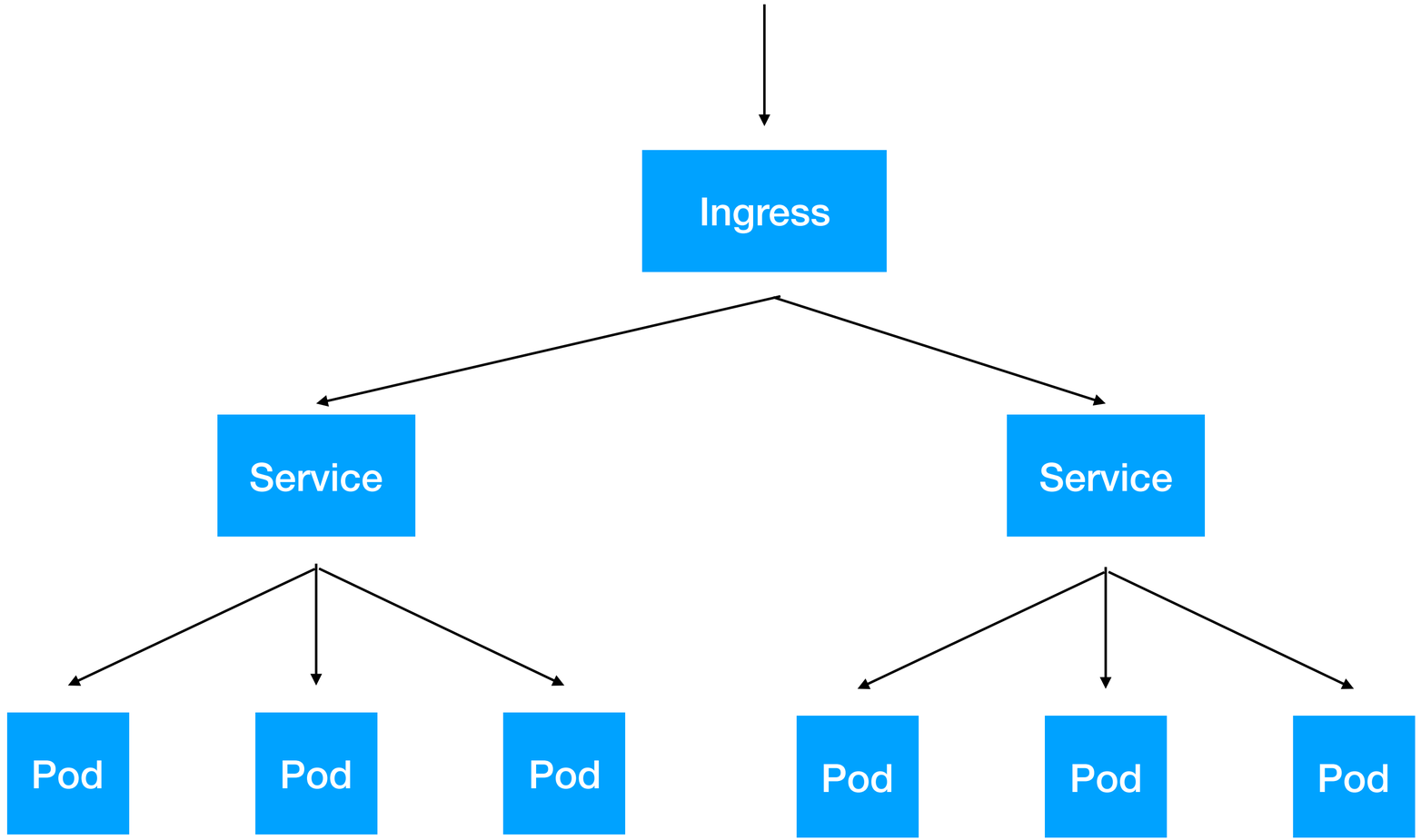
Load Balancer

```
apiVersion: v1
kind: Service
metadata:
  name: sample-lb
spec:
  selector:
    app: some-app
  type: LoadBalancer
  ports:
  - name: http
    port: 80
    targetPort: 80
    protocol: TCP
```



Ingress

```
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: my-ingress
spec:
  backend:
    serviceName: other
    servicePort: 8080
  rules:
  - host: foo.mydomain.com
    http:
      paths:
      - backend:
          serviceName: foo
          servicePort: 8080
  - host: mydomain.com
    http:
      paths:
      - path: /bar/*
        backend:
          serviceName: bar
          servicePort: 8080
```



Egress Traffic

Egress

```
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: sample-network-policy
spec:
  podSelector:
    matchLabels:
      role: my-app
  policyTypes:
  - Egress
  egress:
  - to:
    - ipBlock:
        cidr: 10.0.0.0/24
```

Service Mesh

Rule Number Six
If You Think You Know What's Happening In Your Cluster… Forget it.

Observability

What’s happening in your cluster?

What’s happening on your cluster?

Metrics and Alerting

Logging

Distributed Tracing

Observability Dashboard

Horizontal Pod Autoscaler

Rule Number Seven

Keep your storage and the business rules to manage it completely separated.

Storage

Easily create your own storage implementation

Persistent Volume Snapshots

Number 8: Using Tools

Package Management

Configuration Management

Package Management

- Helm 2
- Bounds of YAML

Configuration Management

- ksonnet
- Pulumi
- Ballerina

Ballerina

Home » Learn » API Docs » ballerinax/kubernetes pa

TABLE OF CONTENTS

ballerinax/kubernetes pacl

PACKAGE DETAIL

Records

Record	Description
ConfigMap	Kubernetes Confi
ConfigMapMount	Secret volume mo
DeploymentConfiguration	Kubernetes deplo
FileConfig	External file type f
IngressConfiguration	Kubernetes ingre
JobConfig	value:"Kubernete
PersistentVolumeClaimConfig	Kubernetes Persi
PersistentVolumeClaims	Persistent Volume
PodAutoscalerConfig	Kubernetes Horiz
Secret	Kubernetes secret volume mount
SecretMount	Secret volume mount configurations for kubernetes
ServiceConfiguration	Kubernetes service configuration

Annotations

```
// Deploy 3 replicas of an nginx pod
import * as k8s from "@pulumi/kubernetes";

function deploy(name, replicas, pod) {
   return new k8s.apps.v1beta1.Deployment(name, {
      spec: {
         selector: { matchLabels: pod.metadata.labels },
         replicas: replicas,
         template: pod
       }
   });
}

const nginxServer = deploy("nginx", 3, {
   metadata: { labels: { app: "nginx" } },
   spec: {
      containers: [{ name: "nginx",
                     image: "nginx:1.15-alpine" }]
   }
});
```

Other types of tools?

- skaffold
- kustomize

Number 9: Extending Kubernetes

What happens if you get an API for free?

What happens when you outgrow the Kubernetes API?

Number 10: A live word called refinement -- Building On Kubernetes

"On top of Kubernetes"

"On Kubernetes"

Follow these rules

You'll have mad bread to break up

If not, 24 hours of on-call with constant wake ups.