

# Kubernetes für Entwickler und Architekten

think  
tecture

Thorsten Hans

@ThorstenHans

Consultant



# Thorsten Hans

Consultant @ Thinktecture

#Azure

#Kubernetes

#CloudNative

#Terraform

thorsten.hans@thinktecture.com

thinktecture.com

thorsten-hans.com

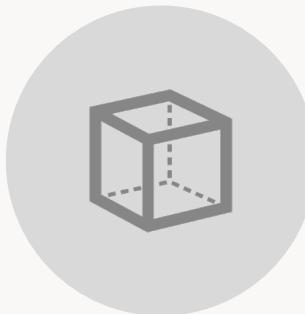
@ThorstenHans



# Talking Points



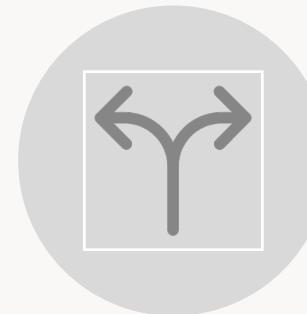
Intro



Running Applications



Adopting Platform  
Capabilities



Exposing Applications  
to the public

# Kubernetes

## Open-Source Container Orchestrator

- Initially founded by Google
- Cloud Native Computing Foundation (CNCF) maintains and manages Kubernetes
- Written in go
- Kubernetes is Cloud-Agnostic

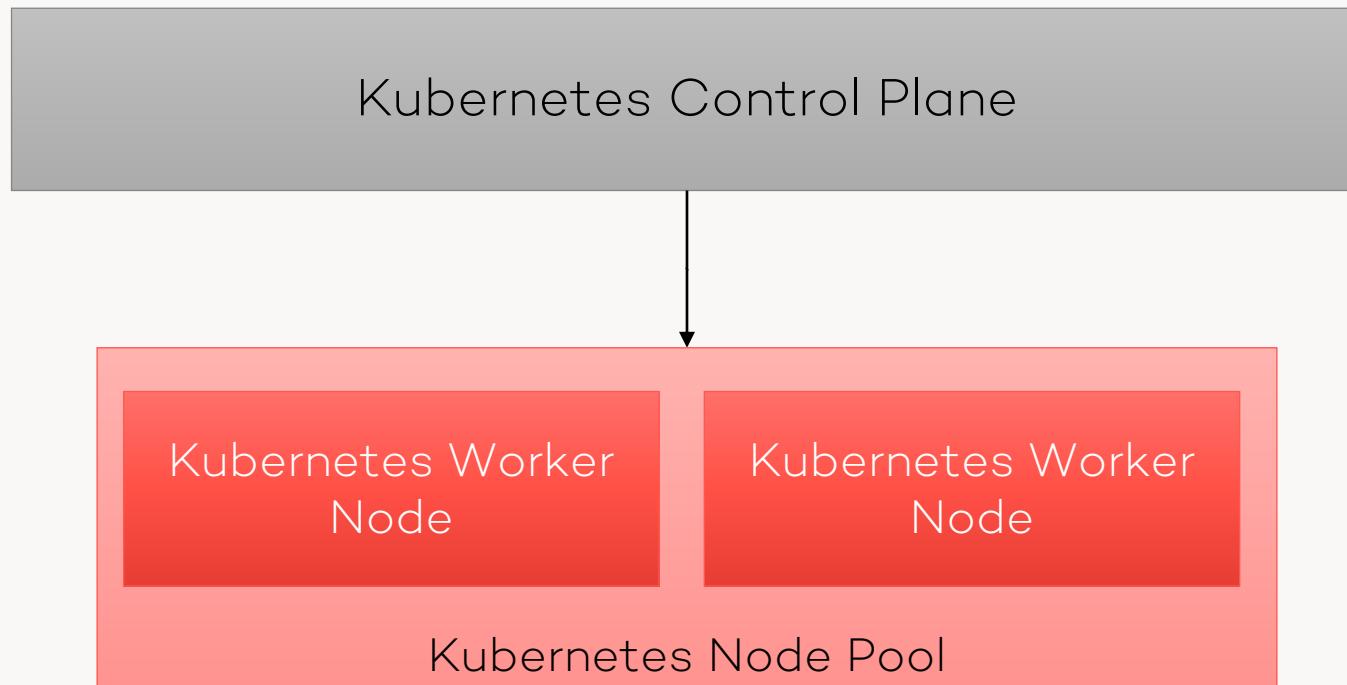
# Kubernetes

## Open-Source Container Orchestrator

- Runs containerized workloads
- Takes care about networking & isolation of workloads
- Abstracts away hardware internas from developers
- Container Lifecycle Management

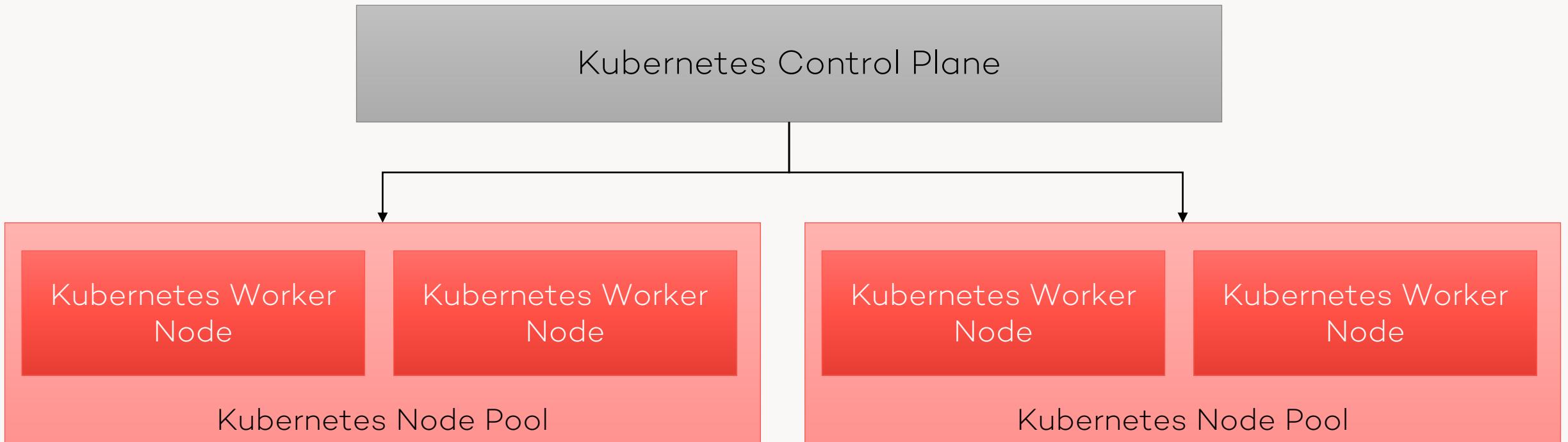
# Kubernetes

## Cluster Infrastructure



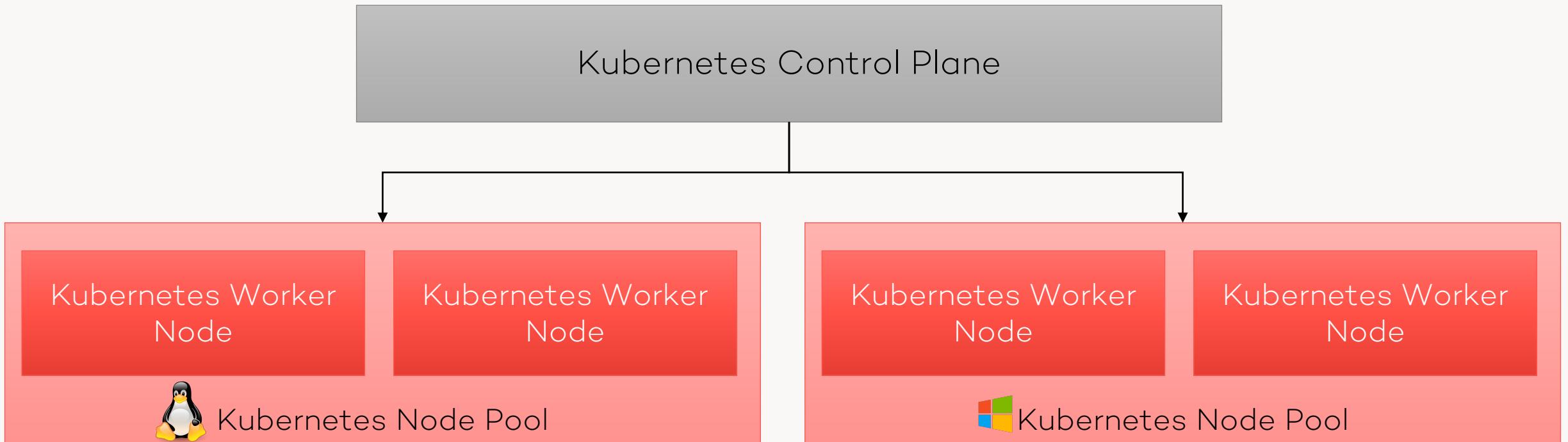
# Kubernetes

## Cluster Infrastructure



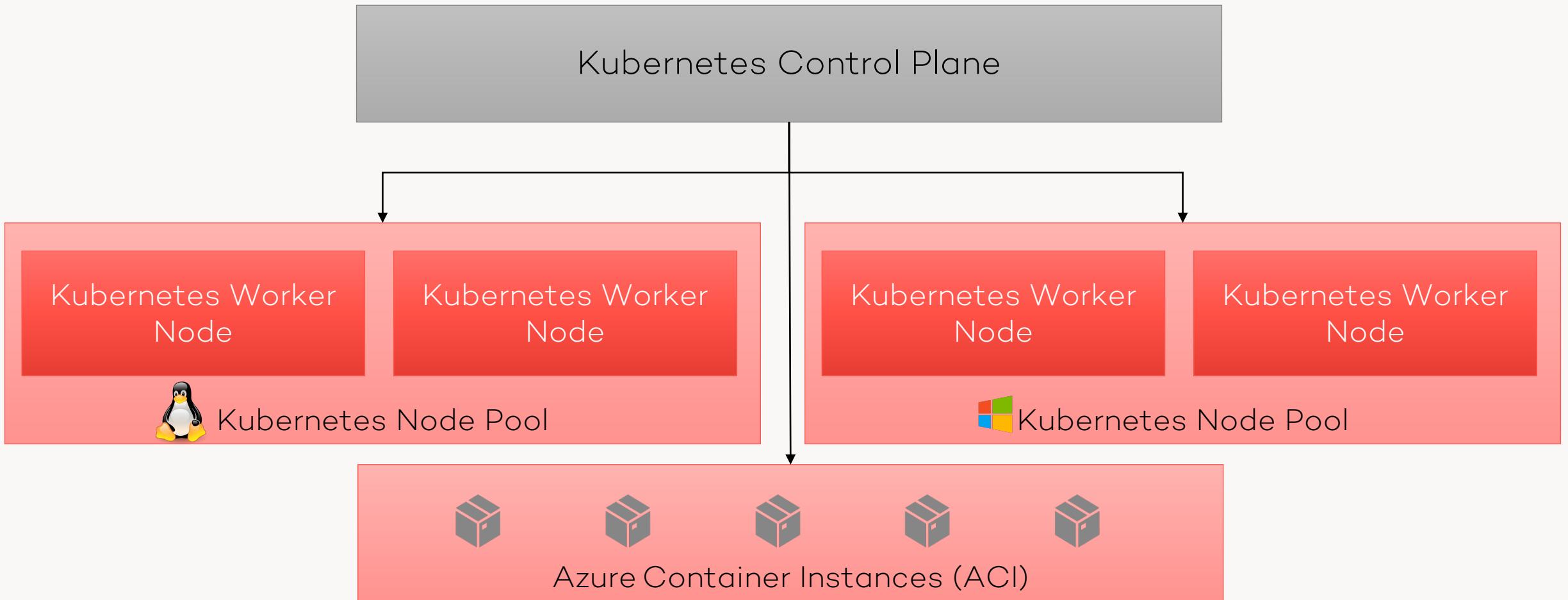
# Kubernetes

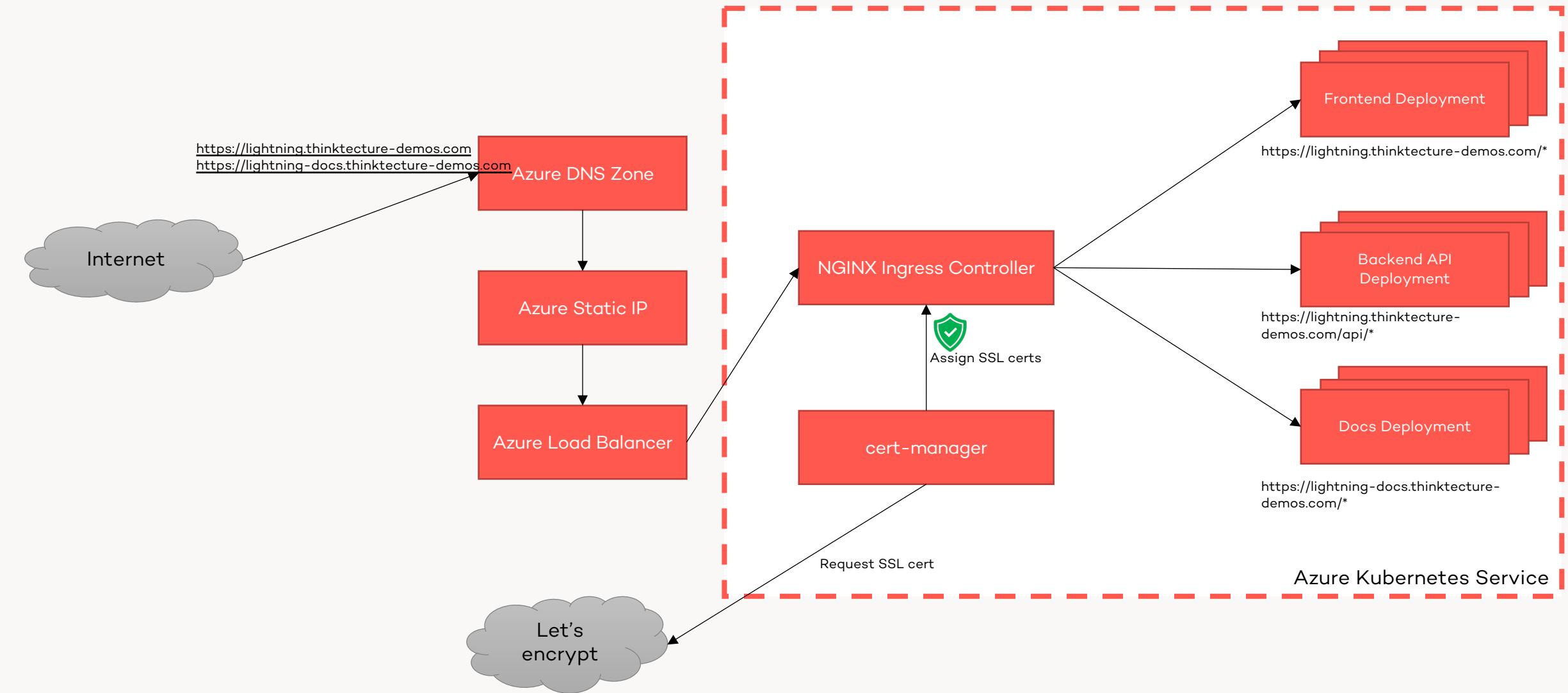
## Cluster Infrastructure

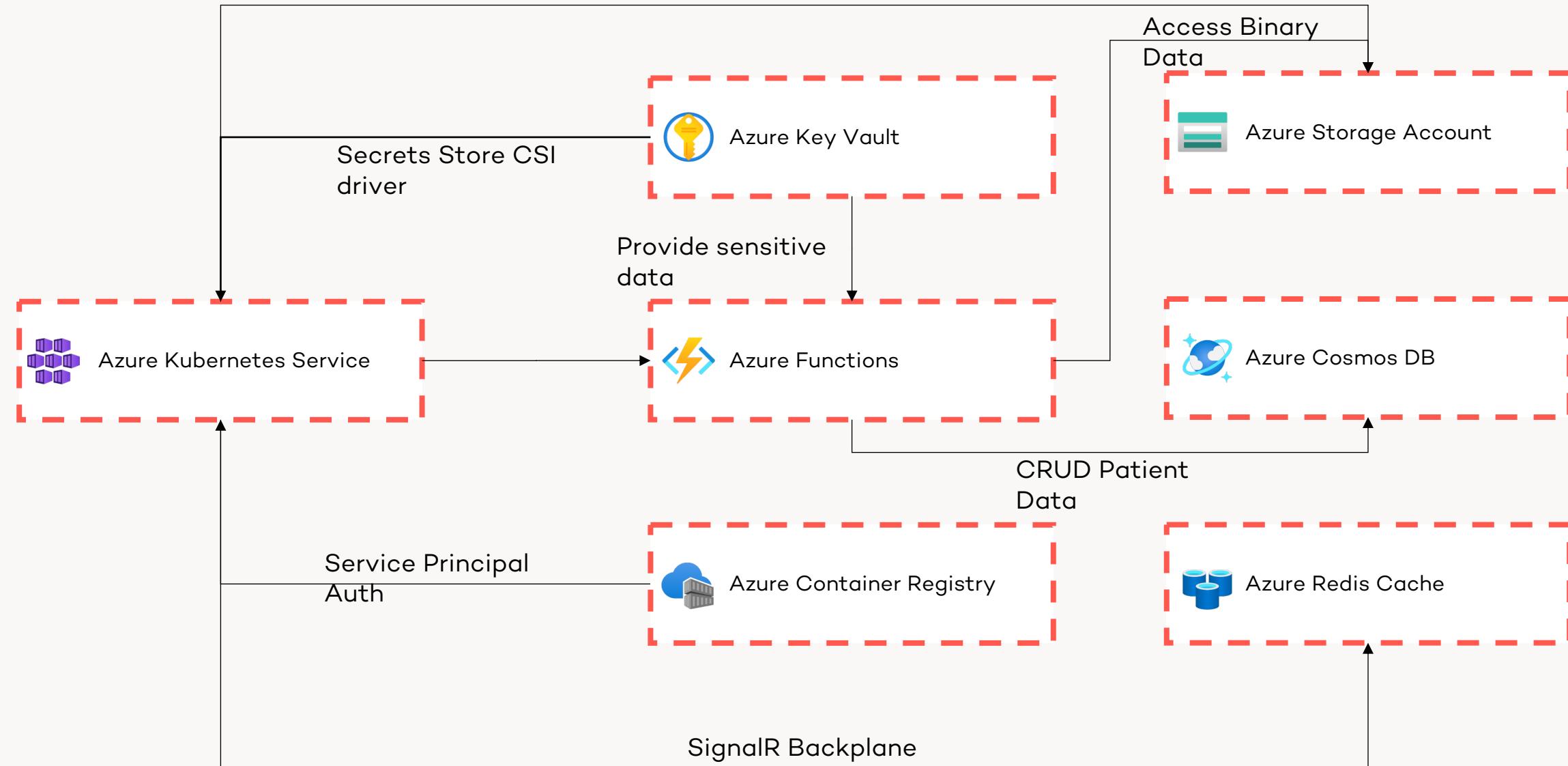


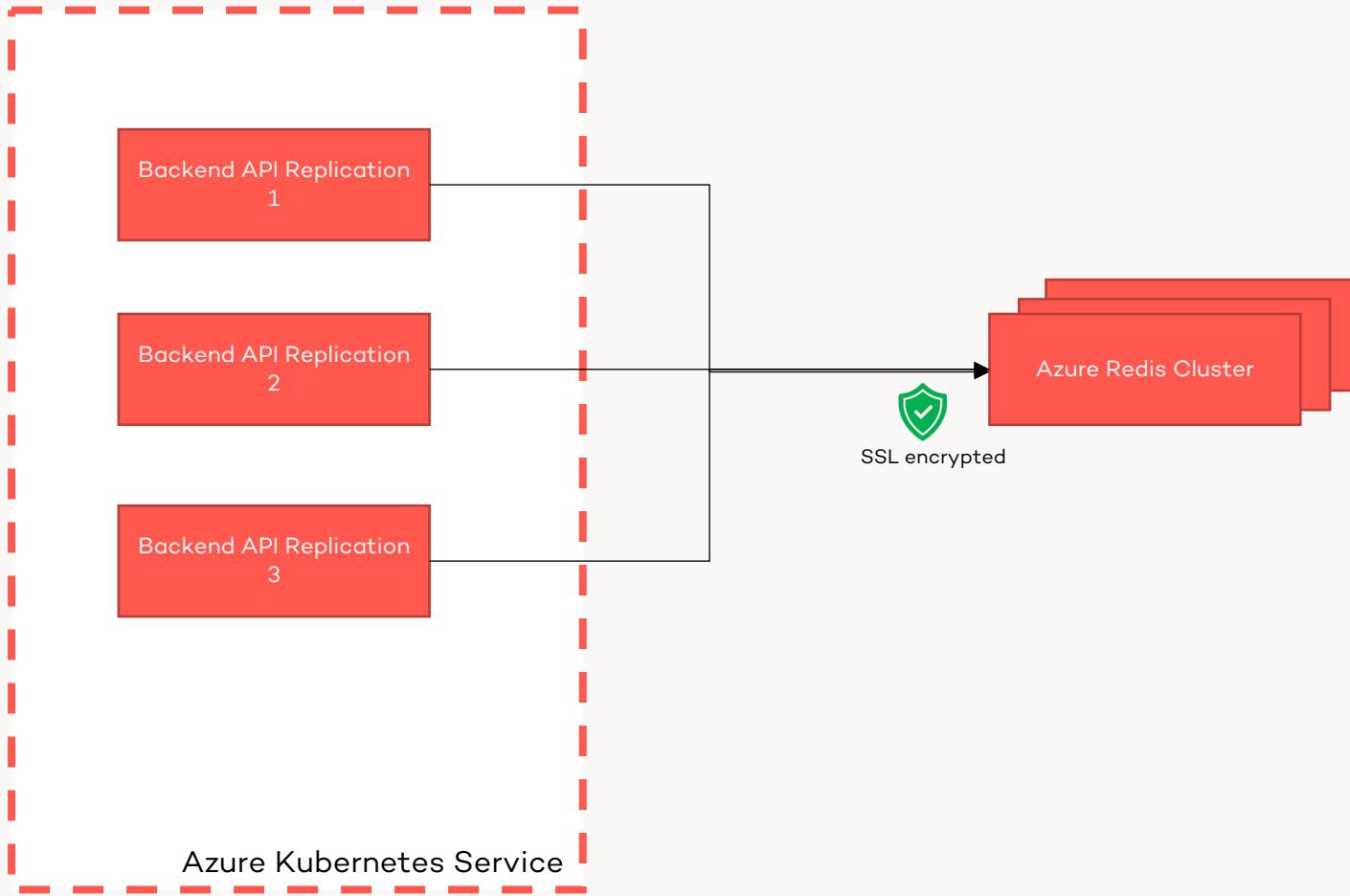
# Kubernetes

## Cluster Infrastructure









# Kubernetes Cluster

## Hosting Opportunities

- Public cloud offerings
  - Azure
  - Amazon AWS
  - Google Compute Cloud
  - Digital Ocean
- On-Premises datacenter
- Locally
  - Kind, microk8s, minikube, ...

## Demo

### Creating a Kubernetes Cluster in Azure / locally

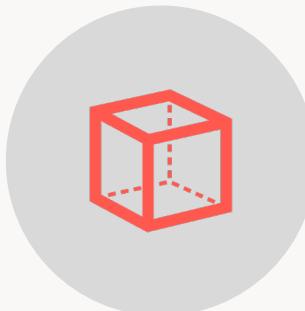
- Create a new Kubernetes Cluster in Azure  
using Azure CLI
- Create a new Kubernetes Cluster locally  
using Kind
- Install Kubectl locally



# Talking Points



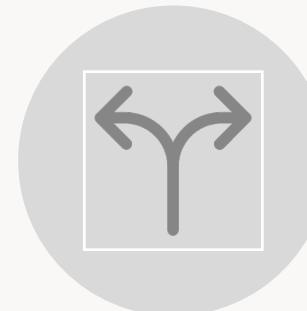
Intro



Running Applications



Adopting Platform  
Capabilities



Exposing Applications  
to the public

# Running Applications

## Kubernetes Core Capabilities

- Pod is the smallest unit of work in Kubernetes
- 1 Pod can contain multiple Docker Containers
  - All containers of a Pod run on the same host
- More complex and powerful variations are ReplicaSets and Deployments

## Demo

### Running Applications in Kubernetes

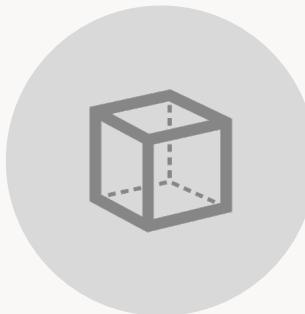
- Pod Spec
- Deployment Spec
- General kubectl operations



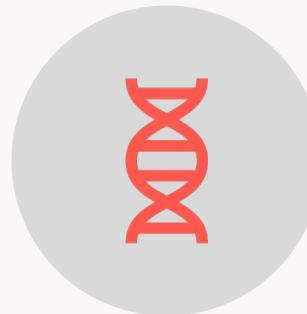
# Talking Points



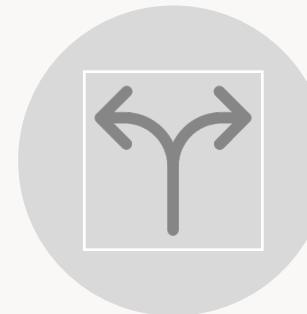
Intro



Running Applications



Adopting Platform  
Capabilities



Exposing Applications  
to the public

# Adopting Platform Capabilities

## ConfigMaps and Secrets

- Inject configuration values into your application containers
- For sensitive values consider using Kubernetes Secrets
- Simple Key-Value storage principles
- Kuberentes Secrets are just encoded

# Adopting Platform Capabilities

## Ressource Requests and Limits

- Specify minimal requirements per container
- Specify maximum resource utilization per container
- Mandatory for each piece of the overall application, deployed to a Kubernetes cluster

# Adopting Platform Capabilities

## Readiness and Liveness Probes

- Let Kubernetes monitor and heal your applications
- Probes support different approaches to verify application health
  - HTTP
  - TCP
  - Process Execution

## Demo

### Adopting Platform Capabilities

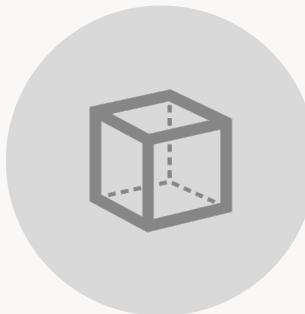
- Create Secrets and ConfigMaps
- Use Secrets and ConfigMaps
- Add support for Health Probes
- Define Resource Requests and Limits



# Talking Points



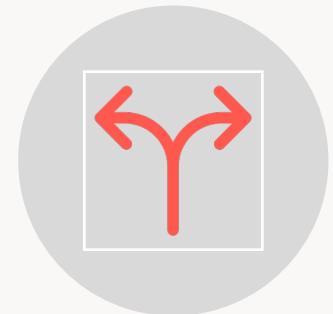
Intro



Running Applications



Adopting Platform  
Capabilities



Exposing Applications  
to the public

# Exposing Applications to the public

## Ingress

- Ingress Controllers route requests from the internet
- NGINX Ingress is an open-source Ingress Controller
- Ingress declarations specify routing
- Ingress Controller routes traffic

# Exposing Applications to the public

## CertManager

- Use CertManager to provide SSL certificates
- Flexible, cluster-wide certificate management solution
- Issues certificates to services
- Support for
  - ACME (Let's Encrypt)
  - HashiCorp Vault
  - Venafi
  - self signed and internal certificate authorities.

## Demo

### Exposing Applications to the public

- Install NGINX Ingress
- Create Ingress definitions

