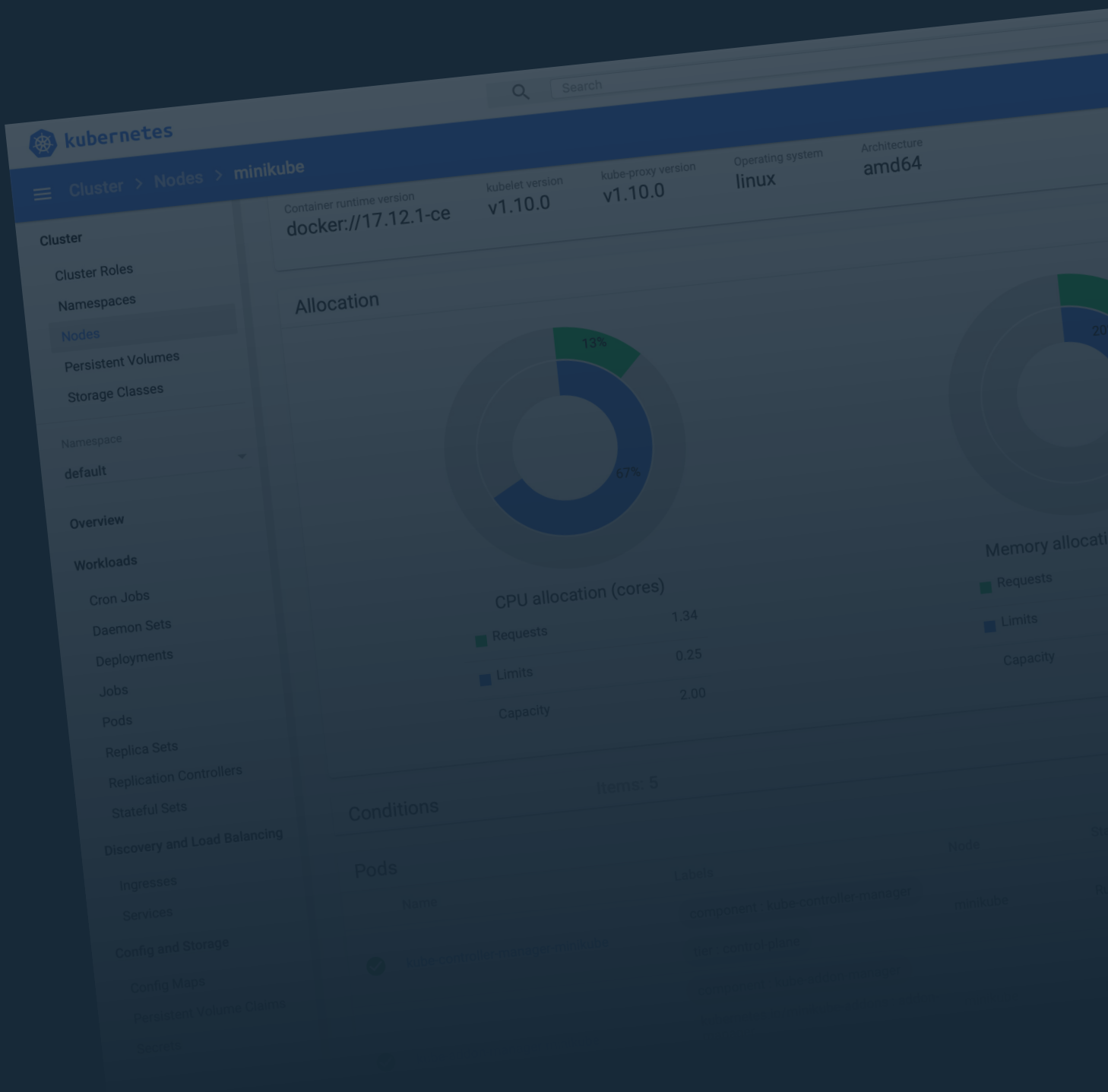
# Kubernetes Workshop




DigIO

# Kubernetes Workshop

## Realise the benefits of Kubernetes for deploying, scaling and managing containerised applications.

### This workshop will cover:

- Local tooling setup and cluster creation.
- Architecture walkthrough and component responsibilities.
- Declarative management and the Kubernetes resource types.
- Targeting multiple environments through resource templating.
- Management access control and RBAC.
- Ingress and network policies.
- Observability.

### Upon completion of this workshop, participants will be able to answer the following questions:

**What is it, and when should I use it?**
Understand how Kubernetes works, and the problems it can be applied to.

**How do I use it?**
Implement common use cases in hands on labs, and become familiar with the tools used to implement them.

**What do I do next?**
Leave with a working environment and sample applications to support further self-paced learning.

### Audience:

Developers or platform engineers who wish to understand the features that Kubernetes provides and how to deploy and run containerised applications on it.

No prior understanding of Kubernetes is required, however a working knowledge of Docker in a local environment and a basic understanding of networking concepts such as load balancing is highly desirable.

### Duration:

2 days

### Number of attendees:

8-10 per session

DigIO

# Kubernetes Workshop

## Contents



DigIO

# Kubernetes Workshop

## Overview

This training aims to provide a crash course in Kubernetes and the fundamental primitives available to us, enabling the capabilities of Service Mesh technologies like Istio.

We go through the most pertinent aspects of Kubernetes which will give us sufficient context to understand how Istio works and how we can enable application deployment and lifecycle management.

The key benefits of Istio are demonstrated through sophisticated traffic steering and observability capabilities, with enhanced security through authentication (JWT, mTLS) and authorization (RBAC).

Given the time available to cover such a large problem space, we'll skip certain parts of the training which can be covered in a self-paced fashion. Made possible via the distribution of the training material in a self-contained git repository and deployed into a running Kubernetes cluster locally.

## Prerequisites

Admin control of machine to install dependencies.

Install Docker-for-desktop available here.

Run make bootstrap this will install dependencies and validate the level of access on your given machine (admin permissions required).

Run make build.

DigIO

# Kubernetes Workshop

## Workshop outline

### Bootstrap 1-1.5 hours

**The instantiation of local Kubernetes based development environment, including the following:**

- Best-practice setting for docker-for-desktop.
- Installing CLI tooling (kubectl, helm, kustomize, kubectx, kubens, et al).
- Management of Kubernetes context for interacting with local cluster.
- Networking settings for training and labs.
- Build of demo apps ready for local deployment.

### Kubernetes - Architecture 1-1.5 hours

**Introduce fundamental Kubernetes components, architectural overview. Explain key components and their associated interactions. The benefits of given architectural decisions. Key concepts will include:**

- Kubernetes architecture.
- Component & Interactions (kube-api, controller, kubelet, kube-proxy et al).
- Locally running cluster and how we interact/control environment.

**Outcomes**

- Understand the interactions within a Kubernetes cluster and the different components in a sufficient way to make basic reasonings about clusters state and the basic of how a running workload is made possible.

DigIO

# Kubernetes Workshop

## Workshop outline

### Kubernetes - Resources  2-3.5 hours

**Overview of key Kubernetes resources and how we can go about building feasible services and the scenarios on where different resources should be used. This includes a theoretical breakdown as well as practical application component. The key theory includes the following:**

Overview key service resources (Job, CronJob, Deployment, ReplicaSet, ReplicaController, Service).

Over of config management resources (ConfigMap, Secret).

### Workshops  2-3.5 hours

Deploy both Job and CronJob, interact with deployed resources and their status.

Deploy a Deployment resources and interact with deployment, experiment with making this deployment available from outside the cluster.

Deploy a Simple Application and interact with running deployment via CLI & Browser. Involves combining a Deployment + Service.

Extend the "Simple" application, adding additional configuration via Secret + ConfigMap resources.

Templating - Kustomize based approach to templating resources.

Templating - Helm based approach to templating resources.

Rolling updates via Deployment resource.

### Outcomes

Having developed Kubernetes resources, deployed them to a running cluster and explored the available tools to evaluate the state of the cluster and running resources, a degree of comfort will be achieved with the primitive core resources available in Kubernetes.

A demonstration of available templating tooling will show how the available resources can be joined to facilitate more "production" or streamlined ways of managing application deployments.

DigIO

# Kubernetes Workshop

## Workshop outline

### Kubernetes - API & Access  1-1.5 hours

**Run through the Kubernetes API and what it means to apply resources in reality and the steps involved in these resources persisting within a cluster.**

**We also look at the available tooling to interact with the Kubernetes API, one of the most basic being kubectl and how this is achieved through kubeconfig.**

**We look at the available mechanisms of controlling access and how we can enable permissions in a fine-grained way to the Kubernetes API and its available endpoints.**

**Workshop  30 minutes**

Exploring Role Based Access Control (RBAC).

**Outcomes**

Understand how the Kubernetes API is made available and how access can be controlled in a fine-grained way, enabling a model of least-privilege and best-practices.

### Kubernetes - Networking & Ingress  2 hours

**Having carried out a number of deployments of Kubernetes resources we look to demystify the networking within Kubernetes and how we go about making services accessible outside the cluster.**

**We then also explore briefly the ways in which can control access within the cluster between different workloads and namespaces.**

**Workshop  1-1.5 hours**

NetworkPolicy.

Simple App Ingress.

Basic Monitoring options.

**Outcomes**

Understanding how Pod-to-Pod Networking is possible in Kubernetes and the various ways this can be configured. Capable of reasoning about network isolation and preventing unwanted access. Understanding how to achieve ingress into your cluster to allow access to your running services, and what exactly this translates to in terms of network flow.

DigIO