

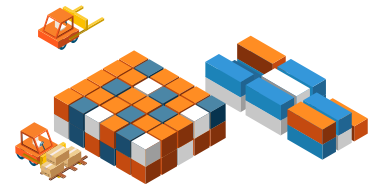
# Kubernetes: mastering application deployment and scaling

Learn Docker and Kubernetes to deploy, run, and manage containerized applications.



With the explosive momentum of Docker, Kubernetes has become the de-facto standard for orchestrating and managing containerized apps in production. In this course, you'll take an app, build it into a docker container then use Kubernetes to deploy, scale, and update it. This course is perfect if you wish to:

- ✓ Get started with Kubernetes in your next project and you need to quickly get up to speed in deploying and scaling your Node.js, Java, .NET, Scala, etc. microservices
- ✓ Design and architect micro services from the ground up leveraging the strength of Kubernetes
- ✓ Understand the why of all the hype around Kubernetes and convince your boss to switch to the dark side of container orchestration
- ✓ Decide which one of the dozen ways to install Kubernetes is right for your team
- ✓ Speak the Kubernetes lingo and communicate with the rest of the industry



## AUDIENCE

- Software developers
- Architects
- Deployment engineers

## PREREQUISITES

- Strong experience using Linux command line
- Understanding of the software development lifecycle
- A general understanding of virtualization concepts
- Experience with networking concepts
- Understand how a simple web application works



## COURSE DETAILS

- Duration: 3 days
- Hours: 9:30am - 5:00pm
- Bring your own laptop

(MacOSX or Linux recommended)

# Day 1

In the first day you'll learn the basics of running Docker containers and how to deploy them to Kubernetes. You will be introduced to the key concepts of Kubernetes, its architecture and how best to leverage the tools available to manage it.

## 1. DOCKER REFRESHER

- What is Docker ?
- Installing Docker
- Running containers
- Publishing to and downloading from Docker registry
- Mounting volumes
- Writing Dockerfiles
- Building Docker images
- Docker layers and caching
- Expose and binding ports
- Containers lifecycle
- Injecting environment variables
- Debugging running containers
- Docker networking model
- Linking containers
- Docker compose

## 4. K8S ARCHITECTURE

- Physical architecture
- Nodes
- API Server
- kubelet
- Etcd
- Scheduler
- Node, replication, endpoints, service account & token controllers
- Architecture for high availability

## 2. INTRODUCTION TO K8S

- What is Kubernetes?
- Installing k8s
- Packaging an application as a Docker container
- Declaring healthchecks
- Logical architecture
- Defining a Pod
- Defining readiness and liveness probes
- Defining a Deployment
- Defining a Service
- Defining an ingress
- Scaling
- Simulating failures

## 5. NETWORKING MODEL

- Networking architecture
- Overlay network
- Endpoints
- Services
- kube-proxy
- kube-dns
- Service types
- Ingress

## 3. DEPLOYMENT STRATEGIES

- Rolling updates
- Deployment history & rollbacks
- Selectors and labels
- Canary deployments
- Blue-green deployments



# Day 2

The second day is focused on developing real world architectures on Kubernetes. You'll learn how to containerise and deploy applications that use different architectural styles. You'll also learn advanced debugging techniques and be able to inspect and solve common issues you're likely to encounter on the platform.

## 1. TROUBLESHOOTING

- Logging architecture
- Debugging pods
- Debugging deployments
- Debugging services
- Debugging ingress
- Attaching to a running container
- Forwarding ports
- Telepresence

## 2. MONITORING

- Horizontal Pod Autoscaling (HPA)
- Metrics server
- Resource metrics API
- Custom resource metrics API
- Daemon sets
- Quotas & limits

## 3. PERSISTENCE & STORAGE

- Managing stateful applications
- Persistent volume
- Persistent volume claim
- Dynamic provisioning
- Managing configurations
- Managing secrets
- Stateful sets

## 4. MICROSERVICES IN K8S

- Designing architectures with k8s
- Service discovery
- Architecting microservices with point-to-point style
- Architecting microservices with an api gateway
- Architecting microservices with message broker style

## 5. TEMPLATING RESOURCES

- Helm
- Installing helm
- Architecture
- Creating charts
- Managing charts
- Templating
- Parametrised deployments
- Debugging
- Creating helpers
- Updating charts
- Managing deployments and rollbacks
- Linking dependency amongst resources
- Requiring dependency as charts
- Sharing chart with repositories

## 5. K8S IN GOOGLE CLOUD

- Installing gcloud
- Creating a project
- Creating the cluster
- Deploying an application
- Adding more team members



# Day 3

The third day explores advanced concepts such as custom scheduling and extending Kubernetes with custom controllers. Also, it covers multi-cloud deployments using Kubernetes federation.

## 1. ADVANCED SCHEDULING

- Scheduler architecture
- Quotas
- Predicated and priorities
- Attaching pods to nodes
- Node affinity/anti-affinity
- Pod affinity/anti-affinity
- Taints and tolerations
- Jobs
- Custom scheduler

## 2. SECURITY

- Accessing the API
- Authentication
- Authorization
- RBAC
- Admission controllers
- Security contexts
- Pod Security Policies
- Network policies

## 3. KUBERNETES FEDERATION

- Architecture • kubefed
- Creating a federation
- Cluster API resource
- Deploying an application in a federation
- Latency based routing
- Spreading Replicas in Underlying Clusters

## 4. EXTENDING KUBERNETES

- Kubectl plugins
- API Access Extensions
- Creating resources
- Managing resources with controllers
- Custom Scheduler
- Custom API and API Aggregator
- Network plugins
- Storage plugins

