



# Kubernetes for App Developers

Course LFD459

3 Days

Instructor-led, Hands on

## Course Information

This course will teach you how to containerize, host, deploy, and configure an application in a multi-node cluster. It also serves as preparation for the Certified Kubernetes Application Developer (CKAD) exam.

## At Course Completion

Upon successful completion of this course, students will understand and acquire the following skills:

- Kubernetes Architecture
- Build
- Design
- Deployment Configuration
- Security
- Exposing Applications
- Troubleshooting

## Prerequisites

To get the most out of this course, you should have:

Basic Linux command line and file editing skills and be familiar with using a programming language (such as Python, Node.js, Go). Knowledge of Cloud Native application concepts and architectures is helpful for this course.

Please note, Kubernetes Administration (LFS458) is not a pre-requisite for this course. There is overlap in the course materials as each one is designed to stand alone and aligns with the related exams.

## Course Outline

### Module 1: Introduction

- Objectives
- Who You Are
- The Linux Foundation
- Linux Foundation Training
- Certification Programs and Digital Badging

Contact ISInc for more information at 916.920.1700 or by visiting our website at <http://www.isinc.com>



- Preparing Your System
- Course Registration
- Labs

## **Module 2: Kubernetes Architecture**

- What Is Kubernetes?
- Components of Kubernetes
- Challenges
- The Borg Heritage
- Kubernetes Architecture
- Terminology
- Master Node
- Minion (Worker) Nodes
- Pods
- Services
- Controllers / Operators
- Single IP per Pod
- Networking Setup
- CNI Network Configuration File
- Pod-to-Pod Communication
- Cloud Native Computing Foundation
- Resource Recommendations
- Labs

## **Module 3: Build**

- Container Options
- Containerizing an Application
- Creating the Dockerfile
- Hosting a Local Repository
- Creating a Deployment
- Running Commands in a Container
- Multi-Container Pod
- readinessProbe
- livenessProbe
- Testing
- Labs

## **Module 4: Design**

- Traditional Applications: Considerations
- Decoupled Resources

Contact ISINC for more information at 916.920.1700 or by visiting our website at <http://www.isinc.com>

- Transience
- Flexible Framework
- Managing Resource Usage
- Using Label Selectors
- Multi-Container Pods
- Sidecar Container
- Adapter Container
- Ambassador
- Points to Ponder
- Jobs
- Labs

## **Module 5: Deployment Configuration**

- Volumes Overview
- Introducing Volumes
- Volume Spec
- Volume Types
- Shared Volume Example
- Persistent Volumes and Claims
- Persistent Volume
- Persistent Volume Claim
- Dynamic Provisioning
- Secrets
- Using Secrets via Environment Variables
- Mounting Secrets as Volumes
- Portable Data with ConfigMaps
- Using ConfigMaps
- Deployment Configuration Status
- Scaling and Rolling Updates
- Deployment Rollbacks
- Labs

## **Module 6: Security**

- Security Overview
- Accessing the API
- Authentication
- Authorization
- ABAC
- RBAC
- RBAC Process Overview
- Admission Controller



- Security Contexts
- Pod Security Policies
- Network Security Policies
- Network Security Policy Example
- Default Policy Example
- Labs

## **Module 7: Exposing Applications**

- Service Types
- Services Diagram
- Service Update Pattern
- Accessing an Application with a Service
- Service without a Selector
- ClusterIP
- NodePort
- LoadBalancer
- ExternalName
- Ingress Resource
- Ingress Controller
- Service Mesh
- Labs

## **Module 8: Troubleshooting**

- Troubleshooting Overview
- Basic Troubleshooting Steps
- Ongoing (Constant) Change
- Basic Troubleshooting Flow: Pods
- Basic Troubleshooting Flow: Node and Security
- Basic Troubleshooting Flow: Agents
- Monitoring
- Logging Tools
- Monitoring Applications
- System and Agent Logs
- Conformance Testing
- More Resource
- Labs

## **Module 9: Closing and Evaluation Survey**

- Evaluation Survey

Contact ISInc for more information at 916.920.1700 or by visiting our website at <http://www.isinc.com>