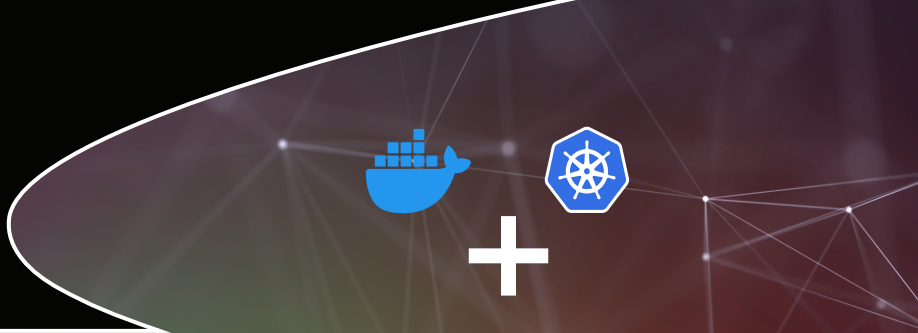


Docker & Kubernetes Advanced



Containerization and DevOps are revolutionizing the Information Technology world and introducing new paradigms into everything related to network-aware application development and deployment. This course continues on from CENGN's Docker & Kubernetes Basics course, taking the learner on an in-depth journey that starts with container creation, adds storage and networking, secures that container and configures monitoring for an Internet accessible Kubernetes-based service.



Audience:

- Software Engineer or Architect
- Network Engineer or Architect
- Cloud Engineer or Architect
- DevOps/DevSec Engineer



Delivery Mode: Learn on your own schedule with self-paced online training and labs



Duration: Learners will need approximately 20 hours to complete the course. Learners will have access to the online content and labs for 4 weeks



Hands-on Labs: This course features hands-on labs using a 3-node cluster hosted on CENGN's multi-vendor cloud. The labs build in sequence, starting with new container creation and adding functionality to produce a secure, web-accessible deployment using industry best practices



Recommended Prerequisites:

This course is best suited for learners with the following knowledge and skills

- Intermediate experience with networking
- Intermediate experience Linux, including command line interface
- Moderate level of understanding of virtualization
- Basic skills in containerization ([CENGN Docker and Kubernetes Basics](#))
- Basic understanding of DevOps ([CENGN Introduction to DevOps](#))



Learner Support: The CENGN Academy team of subject matter experts will be available to support you while you take this course. We will answer your questions, confirm your labs, and check in with you after your course to assist with your badge exam preparations

Exam and Digital Badge

Learners who complete this course are ready the CENGN Docker + Kubernetes Advanced exam. Those who successfully complete the exam will earn a CENGN Docker & Kubernetes Level 2 digital badge, which can be posted on LinkedIn and other social media

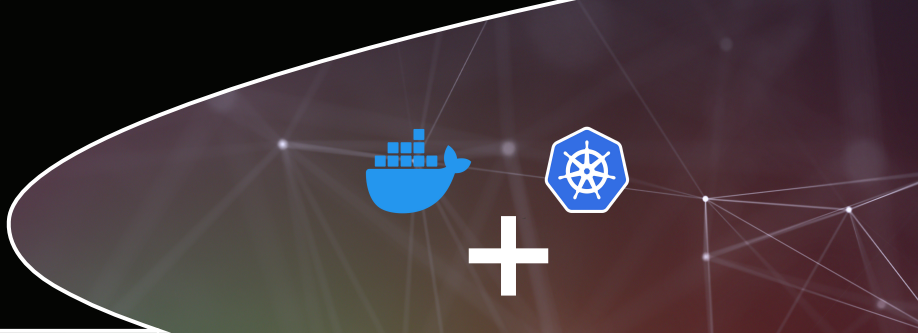


Developed in collaboration with CloudOps

Course Objectives

After completing this course, the learner will be able to:

- Describe container runtime and network interface architectures
- Explain how Docker, Containerd, and Kubernetes interact with the Linux kernel
- Outline how Container Network Interfaces (CNIs) build on core Linux networking functionality
- Apply Docker/Containerd functionality to Kubernetes concepts
- Identify the CNI's role in Kubernetes service discovery
- Create a containerized application using security, storage, service, and monitoring best practices
- Perform admin-level tasks for an existing application
- Explain how business structure impacts application development



Course Content

Module 1 – Container Recap

- Explain core Linux functionality needed by container engines
- Summarize key Linux concepts related to containerization
- Describe the architecture of low-level and high-level runtimes
- Summarize the interaction between applications and the host operating system

Module 2 – Creating Containers

- Explain how container images are created and run
- Describe container image layers
- Build a base container image from scratch

Module 3 – Networking Containers

- Explain core Linux network functionality
- Describe the container network model
- Manage container networks using Docker

Module 4 – Kubernetes Review

- Summarize basic Kubernetes concepts such as pods, nodes, and services
- Compare and contrast Kubernetes and Docker
- Outline advanced Kubernetes concepts such as networking and storage
- Perform basic tasks with the Kubernetes CLI

Module 5 – Kubernetes Services

- Outline how container models build on core Linux functionality
- Summarize the functionality of standard Kubernetes services
- Explain Kubernetes network components

Module 6 – Storage and Stateful Workloads

- Explain the different Kubernetes storage options
- Create and deploy storage for stateless applications
- Create and apply policies for dynamic Kubernetes storage

Module 7 – Kubernetes Network Add-ons

- List different CNIs and their features
- Describe the CNI's role in Kubernetes service discovery
- Explain the sidecar Kubernetes architecture
- Apply CNI deployments to Kubernetes environments
- Create Network policies to safely allow calls to containers

Module 8 – RBAC, Security, and Scheduling

- Identify primary attack vectors in a Kubernetes Cluster
- Demonstrate the ability to schedule and limit cluster resources
- Explain how to implement custom resource definitions

Module 9 – CI/CD Tools and Best Practices

- Implement core upgrades and deployment strategies in Kubernetes
- Describe key features and functionality of Helm
- Implement a Helm chart in a CI/CD pipeline

Module 10 – Monitoring in Kubernetes

- Describe the challenges in monitoring a distributed system
- Explain container and node monitoring
- Deploy metric, logging, and tracing exporters
- Create basic dashboards and visualization tools of deployments