

Docker and Kubernetes Platform Operations

Development clusters, enterprise identity, security and delivery

FA-0740

DevOps, Cloud & Infrastructure · Cybersecurity

Introduction

Explore Docker containers and Kubernetes platform operations through development-cluster exercises and enterprise integration examples. Topics progress from storage and networking to workload exposure, identity, RBAC, admission policy, monitoring, backup and delivery pipelines.

Use a compatible lab stack to examine how these components work together. Enterprise and cloud integration examples provide implementation practice, not a claim that the lab constitutes a fully secured production platform.

Learning outcomes

- Operate Docker containers and select suitable storage and networking options.
- Create and inspect a kind development cluster and deploy selected workloads.
- Explain service exposure, load balancing and external DNS integration.
- Configure and test selected identity, RBAC and admission-policy examples.
- Distinguish Kubernetes audit logging from runtime detection and application-log analysis.
- Perform guided resource/volume backup and restore checks.
- Explain the roles of CI/CD, GitOps and onboarding components in a platform.

Prerequisites

- Linux command-line, networking and basic application-deployment familiarity.

- A compatible container runtime, kubectl and access to the designated lab environment.
- Basic understanding of identity and access control for the enterprise-security modules.

Training outline

1. Module 1 — Docker Fundamentals

- Containerisation concepts, Docker setup and CLI use.
- Images and container lifecycle.
- Persistent data: volumes, bind mounts and tmpfs; choose the appropriate option.
- Docker networking and user-defined bridge networks.

2. Module 2 — Kubernetes Development Clusters

- Kubernetes components and objects; control plane, worker nodes and API server.
- Install kind and create/review a compatible local development cluster.
- Deploy a sample application and inspect its objects.
- Introduce load-balancing and ingress components appropriate to the lab.

3. Module 3 — Services and External Access

- Expose workloads through Kubernetes Services.
- Compare Layer 4 and Layer 7 load balancing.
- Explore MetalLB where suitable for the lab and externalDNS for selected DNS integration.
- Explain provider, permissions and routing dependencies.

4. Module 4 — Enterprise Authentication

- How Kubernetes associates API requests with identities.
- OpenID Connect concepts and compatible lab configuration.
- Impersonation for selected managed-cluster integrations, including the necessary authorisation controls.
- Compare OpenUnison-based integration with a direct impersonation example.

5. Module 5 — RBAC, Namespaces and Audit

- Roles, role bindings and mapping enterprise identities to resource permissions.

- Namespace-based access boundaries and the limits of namespace-only multi-tenancy.
- Kubernetes API audit logging.
- Use audit records and an audit2rbac example where compatible to investigate policy requirements; review generated permissions.

6. Module 6 — Securing a Cluster UI

- UI authentication, permissions and exposure risks.
- Use a supported cluster UI such as Headlamp; compare the archived Kubernetes Dashboard historically.
- Explore reverse-proxy and OpenUnison integration patterns where supported by the chosen lab stack.
- Verify identity propagation and RBAC rather than assuming the UI supplies authorisation.

7. Module 7 — Pod Security and OPA

- Historical PodSecurityPolicy and its removal; current Pod Security Admission and Pod Security Standards.
- Apply selected namespace security controls and examine exemptions.
- Dynamic admission webhooks and Open Policy Agent concepts.
- Rego and OPA Gatekeeper constraints; enforce a selected memory/resource-policy example.
- Compare built-in standards with custom policy checks and test admission outcomes.

8. Module 8 — Runtime Detection and Log Analysis

- Distinguish API auditing, runtime events and application logs.
- Falco configuration, deployment and selected rule/event examples.
- Choose a supported modern eBPF or kernel-module driver for the environment.
- Introduce Elasticsearch/Fluentd/Kibana log-analysis integration in version-compatible context.

9. Module 9 — Backup and Restore

- Kubernetes recovery scope and backup dependencies.
- Separate etcd control-plane snapshots from workload-resource and persistent-volume backups.
- Set up compatible Velero components and storage-provider integration.
- Use the Velero CLI to back up and restore a selected workload; inspect resource and data recovery.

10. Module 10 — Platform Provisioning and Delivery

- Design a sample delivery pipeline and prepare the cluster.
- Deploy or inspect compatible GitLab, Tekton and Argo CD examples and distinguish their roles.
- Explore OpenUnison-based project onboarding.
- Review permissions, integration dependencies and the boundaries between development labs and production operations.