Kubernetes Essentials

- 3 days
- Lecture & Labs

Course Overview

This class prepares students for the Certified Kubernetes Application Developer (CKAD) exam. Kubernetes is a Cloud Orchestration Platform providing reliability, replication, and stability while maximizing resource utilization for applications and services. By the conclusion of this hands-on training you will go back to work with all necessary commands and practical skills to empower your team to succeed, as well as gain knowledge of important concepts like Kubernetes architecture and container orchestration. We prioritize covering all objectives and concepts necessary for passing the Certified Kubernetes Application Developer (CKAD) exam. You will build, command, and configure a high availability Kubernetes environment capable of demonstrating all "K8s" features discussed and demonstrated in this course. Your three days of intensive, hands-on training will conclude with a mock CKAD exam that matches the real thing.

Review this course online at https://www.alta3.com/courses/k8s-3day

Who Should Attend

- Anyone who plans to work with Kubernetes at any level or tier of involvement
- Any company or individual who wants to advance their knowledge of the cloud environment
- Application Developers
- Operations Developers
- IT Directors/Managers

What You'll Learn

All topics required by the CKAD exam, including:

- Deploy applications to a Kubernetes cluster
- Use Kubernetes primitives to implement common deployment strategies (e.g. blue/green or canary)
- Define, build and modify container images
- Implement probes and health checks
- Understand multi-container Pod design patterns (e.g. sidecar, init and others)
- Understand ConfigMaps
- Create & consume Secrets
- Troubleshooting and debugging tools
- Provide and troubleshoot access to applications via services
- Use Ingress rules to expose applications
- AI LLM prompt engineering for relevant configuration snippets and solutions

Outline

From Containers to Kubernetes

- 💬 Lecture: Kubernetes Architecture
- 💻 Lecture + Lab: Define, build and modify container images

- 💬 Lecture: Pods and the Control Plane
- 💻 Lecture + Lab: Deploy Kubernetes using Ansible

Pod Basics

- 💬 Lecture: Namespaces and Fundamental Kubectl Commands
- 💬 Lecture: Understanding YAML
- 💬 Lecture: Pod Manifests
- 💻 Lecture + Lab: Create and Configure Basic Pods
- 💬 Lecture: Understanding API Versioning and Deprecations

Container Health, Security, and Observability

- 💬 Lecture: Kubectl port-forward
- 💻 Lecture + Lab: Debugging via kubectl port-forward
- 💬 Lecture: Kubectl exec and cp
- 💻 Lecture + Lab: Performing Commands inside a Pod
- 💬 Lecture: Readiness and Liveness Probes
- 💻 Lecture + Lab: Implement Probes and Health Checks
- 💬 Lecture: Pod Security Contexts
- 💻 Lecture + Lab: Understanding Security Contexts

Resource Management

- 💬 Lecture: Limits, Requests, and Namespace ResourceQuotas
- 💻 Lecture + Lab: Understanding and Defining Resource Requirements, Limits and Quotas
- 💻 Lecture + Lab: Kubectl Top and Application Monitoring
- 💬 Lecture: Admission Controller
- 💻 Lecture + Lab: Create a LimitRange AdmissionController

RBAC

- 💬 Lecture: Role Based Access Control
- 💻 Lecture + Lab: Service Accounts
- 💬 Lecture: Contexts
- 💻 Lecture + Lab: Cluster Access with Kubernetes Context

Logging

- 💬 Lecture: Utilize Container Logs
- 💻 Lecture + Lab: Kubectl Log Command
- 💬 Lecture: Advanced Logging Techniques

Ephemeral Storage

- 💬 Lecture: ConfigMaps and Volume Mounting
- 💻 Lecture + Lab: Consistent Configuration with ConfigMaps
- 💬 Lecture: Secrets
- 💻 Lecture + Lab: Create and Consume Secrets

Persistent Storage

- 💬 Lecture: Persistent Volumes, Claims, and StorageClasses
- 💻 Lecture + Lab: Using PersistentVolumeClaims for Storage

Multi-Container Pod Design

- 💬 Lecture: Why Use Multi-Container Pods?
- 💻 Lecture + Lab: Creating Ephemeral Storage For Fluentd Logging Sidecar
- 💬 Lecture: Init Containers
- 💻 Lecture + Lab: Understand the Init Container Multi-Container Pod Design Pattern

Deployments

- 💬 Lecture: Labels
- 💻 Lecture + Lab: Understanding Labels and Selectors
- 💬 Lecture: ReplicaSets
- 💬 Lecture: Purpose and Advantages of Deployments
- 💻 Lecture + Lab: Writing a Deployment Manifest
- 💬 Lecture: Deployment Version Control
- 💻 Lecture + Lab: Performing Rolling Updates and Rollbacks with Deployments
- 💬 Lecture: Blue/Green and Canary Deployments
- 💻 Lecture + Lab: Advanced Deployment Strategies
- 💬 Lecture: Horizontal Scaling with Deployments

Jobs and CronJobs

- 💬 Lecture: Jobs and CronJobs
- 💻 Lecture + Lab: Running and Executing a Job

NetworkPolicy

- 💬 Lecture: Controlling Connectivity with NetworkPolicy
- 💻 Lecture + Lab: Namespace Network Policy

Services and Ingress

- 💬 Lecture: Networking with Services
- 💻 Lecture + Lab: Provide and troubleshoot access to applications via services
- 💬 Lecture: Ingress Controllers
- 💻 Lecture + Lab: Use Ingress Rules to Expose Applications

The Helm Package Manager

- 💬 Lecture: Helm
- 💻 Lecture + Lab: Using the Helm Package Manager to Deploy Existing Packages

Extending Kubernetes

- 💬 Lecture: Custom Resource Definitions
- 💻 Lecture + Lab: Custom Resource Definitions (CRDs)

CKAD

- 💬 Lecture: Tips to Pass your CKAD Exam!

Bonus Lectures & Labs

- 💻 Lecture + Lab: Isolating Resources with Kubernetes Namespaces
- 💬 Lecture: Persistent Volumes with CSI
- 💻 Lecture + Lab: CSI Storage Solution: NFS
- 💬 Lecture: Taints, Tolerations, and Pod Affinity

- 💻 Lecture + Lab: Tainted Nodes and Tolerations
- 💬 Lecture: Kubectl get and sorting
- 💻 Lecture + Lab: Listing Resources with kubectl get
- 💻 Lecture + Lab: Examining Resources with kubectl describe
- 💬 Lecture: Annotations
- 💻 Lecture + Lab: Insert an Annotation
- 💻 Lecture + Lab: Create and Configure a ReplicaSet
- 🏆 Challenge: Horizontal Pod Autoscaler
- 💬 Lecture: Networking Plugins
- 💬 Lecture: Hostnames and FQDNs
- 💻 Lecture + Lab: Utilizing FQDNs
- 💻 Lecture + Lab: Troubleshooting

Prerequisites

Next Courses

- CKA (https://alta3.com/courses/cka)
- Developing Microservices (https://alta3.com/courses/microservices)