Version : **2026.01**

Last update : 2026/02/02 14:02

Preparing for the CKA (Certified Kubernetes Administrator) Certification

Prerequisites

Hardware

- One computer (MacOS, Linux, Windows™ or Solaris™),
- AZERTY FR or QWERTY US keyboard,
- 4 GB RAM minimum,
- 2-core processor minimum,
- Headphones or earphones,
- A microphone (optional).

Software

- Web Chrome version 72+ or
- Microsoft Edge version 79+ or
- Firefox version 65+.

Internet

- **Fast** Internet access (4G minimum) **WITHOUT** using a proxy,
- **Unblocked** access to ports 80 and 443 at: https://ittraining.trainercentralsite.com/#/home and *.ittraining.team.

Curriculum

- **DOE301 - Creating Kubernetes clusters**
  - Container Orchestration
  - Introduction to Kubernetes (k8s)
    - Control Plane
    - Controller
    - Nodes (Minions)
  - LAB #1 - Creating a Kubernetes cluster with Virtual Machines
    - 1.1 - Overview
    - 1.2 - Connecting to the kubemaster
    - 1.3 - Testing the network
    - 1.4 - Initializing the Cluster Controller
    - 1.5 - Installing a Network Extension for communication between PODs
    - 1.6 - Connecting workers to the Controller
    - 1.7 - K8s and High Availability
  - LAB #2 - Creating a Kubernetes cluster with Minikube
    - 2.1 - Introducing Minikube
    - 2.2 - Installing Minikube
    - 2.3 - Configuring Minikube
    - 2.4 - Installing Docker
    - 2.5 - Installing kubectl
    - 2.6 - The minikube addons command
    - 2.7 - The minikube dashboard addon

- **DOE302 - Managing Pods, Replication Controllers, ReplicaSets, Deployments, Maintenance and Cluster Updates**
  - LAB #1 - Creating a pod
    - 1.1 - Introducing a pod
    - 1.2 - Manual pod creation
    - 1.3 - Creating a pod using a YAML file
      - apiVersion
      - kind
      - metadata

- spec
- Using the YAML file
- LAB #2 - Using Replication Controllers and ReplicaSets
  - 2.1 - Replication Controllers
    - Overview
    - Implementation
  - 2.2 - ReplicaSets
    - Overview
    - Implementation
- LAB #3 - Deployment management
  - 3.1 - Overview
  - 3.2 - Implementation
    - Rollouts
    - Rolling Updates
    - Rollbacks
- LAB #4 - Maintenance
  - 4.1 - The drain command
  - 4.2 - The uncordon command
- LAB #5 - Managing Cluster Updates
  - 5.1 - Updating kubeadm
  - 5.2 - Updating Workers

- **DOE303 - The kubectl, krew and kustomize commands**
  - LAB #1 - Using the kubectl command
    - 1.1 - Getting help with kubectl commands
    - 1.2 - Obtaining information about the Cluster
      - The version command
      - The cluster-info command
      - The api-versions command
      - The api-resources command
    - 1.3 - Obtaining information about nodes
      - The describe node command
      - The top command
    - 1.4 - Obtaining information about Pods

- The describe pod command
- The top command
  - 1.5 - Working with the kubectl command
    - The apply command
    - The create command
    - The get command
    - Using Options
    - The exec command
    - Imperative commands
  - LAB #2 - Managing kubectl plugins with the krew command
    - 2.1 - Installing krew
    - 2.2 - Viewing the list of plugins
    - 2.3 - Installing and using plugins
    - 2.4 - Updating and deleting plugins
  - LAB #3 - Managing patches with the kustomize command

- **DOE304 - Working with Pods and Containers**
  - LAB #1 - Application Configuration
    - 1.1 - Overview
    - 1.2 - Creating a ConfigMap
    - 1.3 - Creating a Secret
    - 1.4 - Using ConfigMaps and Secrets
      - Using Environment variables
      - Using Configuration Volumes
  - LAB #2 - Container Resource Management
    - 2.1 - Overview
    - 2.2 - Resource Requests
    - 2.3 - Resource Limits
  - LAB #3 - Container supervision
    - 3.1 - Overview
    - 3.2 - Liveness Probes
      - The exec Probe
      - The httpGet Probe
    - 3.3 - Startup Probes

- 3.4 - Readiness Probes
    - LAB #4 - Restart Policy Management
        - 4.1 - Overview
        - 4.2 - Always
        - 4.3 - OnFailure
        - 4.4 - Never
    - LAB #5 - Creating Multi-container Pods
        - 5.1 - Overview
        - 5.2 - Implementation
    - LAB #6 - Init containers
        - 6.1 - Overview
        - 6.2 - Implementation
    - LAB #7 - Scheduling
        - 7.1 - Overview
        - 7.2 - Implementation
    - LAB #8 - DaemonSets
        - 8.1 - Overview
        - 8.2 - Implementation
    - LAB #9 - Static Pods
        - 9.1 - Overview
        - 9.2 - Implementation

- **DOE305 - Network, Service and Microservices Architecture Management**
    - LAB #1 - Network and Service Management
        - 1.1 - Overview of Network Extensions
        - 1.2 - DNS K8s
            - Overview
            - Implementation
        - 1.3 - Network Policies
            - Overview
            - Implementation
        - 1.4 - Services
            - Overview
            - Implementation

- The NodePort service
- The ClusterIP service
  - 1.5 - Services and the K8s DNS
    - Overview
    - Implementation
  - 1.6 - K8s Ingress management
    - Overview
    - Implementation
- LAB #2 - Microservices Architecture Management
  - 2.1 - Overview
  - 2.2 - Creating Deployments
  - 2.3 - Creating Services
  - 2.4 - Deploying the Application
  - 2.5 - Scaling Up

- **DOE306 - Managing Volumes with Kubernetes**
  - Overview
    - Volumes
    - Persistent Volumes
    - Volume Types
  - LAB #1 - Using K8s Volumes
    - 1.1 - Volumes and volumeMounts
    - 1.2 - Sharing volumes between containers
  - LAB #2 - Persistent Volumes
    - 2.1 - Storage Classes
    - 2.2 - Persistent Volumes
    - 2.3 - Persistent Volume Claims
    - 2.4 - Using a PersistentVolumeClaim in a pod
    - 2.5 - Resizing a PersistentVolumeClaim

- **DOE307 - Troubleshooting K8s**
  - LAB #1 - The API Server
    - 1.1 - Connection Refused
    - 1.2 - System Pod Logs
  - LAB #2 - The Nodes

- 2.1 - NotReady Status
  - LAB #3 - Pods
    - 3.1 - The ImagePullBackOff Error
    - 3.2 - The CrashLoopBackOff Error
  - LAB #4 - Containers
    - 4.1 - The exec Command
  - LAB #5 - Networking
    - 5.1 - kube-proxy and DNS
    - 5.2 - The netshoot Container

- **DOE308 - Introduction to Securing K8s**
  - LAB #1 - Role Based Access Control and TLS Certificates
    - 1.1 - Overview
    - 1.2 - The /etc/kubernetes/manifests/kube-apiserver.yaml File
    - 1.3 - Creating a serviceAccount
    - 1.4 - Creating a User
    - 1.5 - TLS Certificates
  - LAB #2 - Pod Security Implementation
    - 2.1 - Overview
    - 2.2 - Kubernetes Security Context
    - 2.3 - Kubernetes Network Policies
    - 2.4 - Kubernetes Resource Allocation Management

- **DOE309 - Package Management for Kubernetes with Helm**
  - Overview
  - LAB #1 - Working with Helm
    - 1.1 - Installing Helm
    - 1.2 - The helm search hub Command
    - 1.3 - Searching the Artifact Hub
    - 1.4 - Adding and Deleting a Repository
    - 1.5 - The helm search repo Command
    - 1.6 - The helm show Command
    - 1.7 - Installing a Chart
    - 1.8 - The helm get Command
    - 1.9 - Using NOTES

- 1.10 - The helm upgrade Command
      - 1.11 - The helm history Command
      - 1.12 - The helm rollback Command
      - 1.13 - The helm uninstall Command
  - LAB #2 - Monitoring Kubernetes with the EFK Stack
      - 2.1 - Overview
      - 2.2 - Installing the elasticsearch Chart
      - 2.3 - Installing fluentd-elasticsearch Chart
      - 2.4 - Installing the kibana Chart
      - 2.5 - Generating Logs in Kubernetes
      - 2.6 - Visualizing Data with Kibana

- **DOE310 - Exam - Acquired knowledge**
  - 60 questions from a pool of 104 questions