

Kubernetes for App Developers (LFD459)

Cours officiel, préparation à l'examen CKAD

Cours Pratique de 3 jours - 21h

Réf : DKU - Prix 2024 : 2 480€ HT

Avec cette formation, vous apprendrez à conteneuriser, héberger, déployer et configurer une application dans un cluster à plusieurs nœuds. En commençant par un simple script Python, vous définirez les ressources de l'application et utiliserez les primitives principales pour créer, surveiller et dépanner des applications évolutives dans Kubernetes. En travaillant avec des plug-ins de réseau, la sécurité et le stockage en nuage, vous serez exposé à de nombreuses fonctionnalités nécessaires au déploiement d'une application dans un environnement de production.

OBJECTIFS PÉDAGOGIQUES

À l'issue de la formation l'apprenant sera en mesure de :

Conteneuriser et déployer un nouveau script Python

Configurer le déploiement avec ConfigMaps, Secrets et SecurityContexts

Comprendre la conception de pods multi-conteneurs

Configurer les sondes pour la santé des pods

Mettre à jour et annuler une application

Implémenter des services et des stratégies de réseau

Utiliser PersistentVolumeClaims (PVC) pour la persistance d'état

CERTIFICATION

Cette formation, vous prépare à l'examen Certified Kubernetes Application Developer (CKAD). Les sujets abordés sont directement alignés sur les domaines de connaissances testés par le programme CKAD certifié de la Cloud Native Computing Foundation, et augmenteront considérablement votre capacité à obtenir la certification.

LE PROGRAMME

dernière mise à jour : 09/2021

1) Introduction

- Objectifs.
- Qui êtes-vous ?
- Fondation Linux.
- Formation Linux Foundation.
- Programmes de certification et badges numériques.
- Préparation de votre système.

2) Architecture Kubernetes

- Qu'est-ce que Kubernetes ?
- Composants de Kubernetes.
- Défis. L'héritage Borg.
- Architecture Kubernetes.
- Terminologie.
- Nœud maître. Nœuds Minion (worker).
- Pods. Services. Contrôleurs.
- IP unique par pod.

PARTICIPANTS

Consultants, développeurs, architectes, DevOps, chefs de projet.

PRÉREQUIS

Compétences de base en ligne de commande Linux et en édition de fichiers. Être familiarisé avec un langage de programmation (Python). Connaissances des architectures d'applications cloud native.

COMPÉTENCES DU FORMATEUR

Les experts qui animent la formation sont des spécialistes des matières abordées. Ils sont agréés par l'éditeur et sont certifiés sur le cours. Ils ont aussi été validés par nos équipes pédagogiques tant sur le plan des connaissances métiers que sur celui de la pédagogie, et ce pour chaque cours qu'ils enseignent. Ils ont au minimum trois à dix années d'expérience dans leur domaine et occupent ou ont occupé des postes à responsabilité en entreprise.

MODALITÉS D'ÉVALUATION

Évaluation des compétences visées en amont de la formation.

Évaluation par le participant, à l'issue de la formation, des compétences acquises durant la formation.

Validation par le formateur des acquis du participant en précisant les outils utilisés : QCM, mises en situation...

À l'issue de chaque stage, ITTCERT fournit aux participants un questionnaire d'évaluation du cours qui est ensuite analysé par nos équipes pédagogiques. Les participants réalisent aussi une évaluation officielle de l'éditeur. Une feuille d'émargement par demi-journée de présence est fournie en fin de formation ainsi qu'une attestation de fin de formation si le participant a bien assisté à la totalité de la session.

MOYENS PÉDAGOGIQUES ET TECHNIQUES

Les ressources pédagogiques utilisées sont les supports et les travaux pratiques officiels de l'éditeur.

MODALITÉS ET DÉLAIS D'ACCÈS

L'inscription doit être finalisée 24 heures avant le début de la formation.

ACCESSIBILITÉ AUX PERSONNES HANDICAPÉES

Vous avez un besoin spécifique d'accessibilité ? Contactez Mme FOSSE, référente handicap, à l'adresse suivante psh-accueil@orsys.fr pour étudier au mieux votre demande et sa faisabilité.

- Configuration du réseau.
- Fichier de configuration réseau Container Network Interface (CNI).
- Communication entre pods.
- Cloud Native Computing Foundation.
- Recommandations de ressources.

Travaux pratiques : Mise en application.

3) Build

- Options de conteneur.
- Conteneurisation d'une application.
- Création du fichier Docker.
- Hébergement d'un référentiel local.
- Création d'un déploiement.
- Exécution de commandes dans un conteneur.
- Pod multi-conteneurs.
- ReadinessProbe. LivenessProbe.

Mise en application.

4) Conception

- Applications traditionnelles : considérations.
- Ressources découplées.
- Fugacité.
- Cadre flexible.
- Gestion de l'utilisation des ressources.
- Pods multi-conteneurs.
- Conteneur Sidecar.
- Conteneur adaptateur.
- Ambassador.
- Points à considérer.
- Jobs.

5) Configuration de déploiement

- Vue d'ensemble des volumes.
- Présentation des volumes.
- Spec de volumes.
- Types de volumes.
- Exemple de volumes partagés.
- Volumes persistants et réclamations.
- PersistentVolume.
- PersistentVolumeClaim (PVC).
- Provisionnement dynamique.
- Secrets.
- Utilisation de secrets via des variables d'environnement.
- Secrets de montage sous forme de volumes.
- Données portables avec ConfigMaps.
- Utilisation de ConfigMaps.
- Statut de configuration de déploiement.
- Scaling et mises à jour progressives.
- Rollbacks de déploiement.

Travaux pratiques : Mise en application.

6) Sécurité

- Vue d'ensemble de la sécurité.
- Accéder à l'interface de programmation des applications (API).
- Authentification.
- Autorisation.

- ABAC. RBAC.
- Vue d'ensemble du contrôle d'accès basé sur les rôles (RBAC).
- Contrôleur d'admission.
- Contextes de sécurité.
- Politiques de sécurité du pod.
- Politiques de sécurité réseau.
- Exemple de politique de sécurité réseau.
- Exemple de politique par défaut.

Travaux pratiques : Mise en application.

7) Exposer des applications

- Types de service.
- Schéma de services.
- Modèle de mise à jour de service.
- Accession à une application avec un service.
- Service sans sélecteur.
- ClusterIP.
- NodePort.
- LoadBalancer.
- Nom externe.
- Ressource d'entrée. Contrôleur d'entrée.

8) Dépannage

- Vue d'ensemble du dépannage.
- Étapes de dépannage de base.
- Changement en cours (constant).
- Procédure de dépannage de base : pods.
- Procédure de dépannage de base : nœud et sécurité.
- Procédure de dépannage de base : agents.
- Surveillance.
- Outils de journalisation.
- Applications de surveillance.
- Journaux système et agent.
- Tests de conformité.

Travaux pratiques : Mise en application.

LES DATES

CLASSE À DISTANCE

2024 : 17 juin, 22 juil., 26 août, 23 sept., 21 oct., 25 nov., 16 déc.

PARIS

2024 : 21 mai, 17 juin, 26 août, 23 sept., 21 oct., 25 nov., 16 déc.