+33 (0)1 48 10 09 46
formation@ambient-it.net
140 Avenue Jean Lolive 93500 Pantin - France

Mis à jour le 20/01/2025

S'inscrire

Formation Kubernetes Avancé : Administration en Production

3 jours (21 heures)

PRÉSENTATION

Notre formation Kubernetes avancée vous offre une maîtrise approfondie pour administrer et sécuriser des clusters Kubernetes en production.

Dans cette formation vous apprendrez à monitorer, administrer, gérer les utilisateurs et déployer cette infrastructure en production.

Vous apprendrez à automatiser l'installation, gérer la scalabilité et la sécurité, paramétrer RBAC, utiliser des add-ons avancés (Istio, Cilium), et adopter des pratiques GitOps.

Cette formation apporte des bases solides pour passer les certifications CKS et CKA.

Pour avoir un bon score, nous vous conseillons de poursuivre avec notre préparation à la certification CKA et notre préparation à la certification CKS. À la suite de ces cours d'une journée, vous pourrez passer ces certifications gratuitement.

Cette formation vous présentera la toute dernière version de Kubernetes (à la date de rédaction de l'article : Kubernetes 1.32).

OBJECTIFS

- Comprendre comment utiliser Kubernetes
- Maîtriser les concepts avancés de Kubernetes pour l'dministration de clusters en production
- Configurer la sécurité (RBAC, politiques réseau, gestion des secrets)
- Utiliser des outils avancés comme Istio, Helm, et des pratiques GitOps
- Superviser, monitorer et résoudre les problèmes des clusters Kubernetes
- Préparer les certifications CKA et CKS.
- Maîtriser les fonctionnements avancés des réseaux sur Kubernetes

- Paramétrer la sécurité d’un cluster Kubernetes

PUBLIC VISÉ

- Administrateurs systèmes
- Architecte infrastructure
- DevOps

PRÉ-REQUIS

- De préférence, avoir suivi notre formation Kubernetes
- Connaissance de base d’un système Unix et du fonctionnement des conteneurs
- Tester Mes Connaissances

RECOMMANDATIONS DE LECTURE AVANT ET APRÈS LA FORMATION

- « Kubernetes : Up and Running » de Kelsey Hightower, Brendan Burns et Joe Beda
- « The Kubernetes Book » de Nigel Poulton
- « Cloud Native DevOps with Kubernetes » de John Arundel
- « Kubernetes Cookbook: Building Cloud Native Applications » de S. Goasguen et Michael Hausenblas
- La page Github de Kubernetes

PROGRAMME DE NOTRE FORMATION KUBERNETES AVANCÉE

INTRODUCTION AUX MICRO-SERVICES

- Applications monolithique Vs micro-services
  - Caractéristiques d'une Application Monolithique
  - Avantages d'une Application Monolithique
  - Inconvénients d'une Application Monolithique
  - Diagramme illustratif d'une Application Monolithique
- Qu'est-ce qu'un Micro-service ?
  - Principes clés des micro-services
  - Avantages des Micro-services
  - Défis des Micro-services
  - Outils et Technologies pour les Micro-Services
  - Micro-services et Kubernetes
  - Comparaison avec une Architecture Micro-services

ADMINISTRATION DE KUBERNETES EN PRODUCTION

- Kubeadm : Un outil de déploiement Kubernetes
  - Qu'est-ce que Kubeadm ?
  - Autres outils pour déployer Kubernetes
- Configuration avancée de kubeadm
- Travaux pratiques
- Déploiement d'un cluster Kubernetes haute disponibilité
- Mise en place automatisée d’un cluster Kubernetes On-Premise
- Sécurisation d’un cluster Kubernetes On-Premise pour la production
- Mise en place de la haute disponibilité pour le Control-Plane
- Mise à jour automatisée en mode Rolling Update du Control-Plane et des nœuds Kubernetes
- Virtuosité dans l’utilisation de kubectl pour la CKAD
- Intégration continue dans le Cloud avec kind
- Les runtimes: crictl, Docker et Containerd

LES COMPOSANTS DU CONTROL PLANE ET DES NŒUDS DE TRAVAIL

- Introduction
- Composants du Control Plane
- API Server (kube-apiserver)
- etcd
- Scheduler (kube-scheduler)
- Controller Manager (kube-controller-manager)
  - Cloud Controller Manager (cloud-controller-manager)
- Composants des nœuds de travail
- Fonctionnement de la boucle de réconciliation et du Controller Kubernetes
  - La boucle de réconciliation
  - Fonctionnement des contrôleurs Kubernetes
- Fonctionnement interne de l'API Server : Authentification, Autorisation et Admission Control
  - Fonctionnement interne de l'API Server
  - Gestion des contrôleurs d'admission
  - Extension du cycle de vie du serveur d'API avec les Webhooks d'Admission
- Extension du cycle de vie du serveur d’API avec les MutatingAdmissionWebhook et les ValidatingAdmissionWebhook
- Configuration déclarative
- Groupement implicite ou dynamique
- Cinématique de création d’un Pod à partir d’un Deployment
- Kube-proxy, fonctionnement avancé du réseau virtuel des services
- Service discovery avec CoreDNS

GESTION DES ACCÈS AVEC RBAC ET UTILISATEURS

- Introduction à RBAC
  - Qu'est-ce que RBAC ?
  - Pourquoi utiliser RBAC ?
- Concepts de base de RBAC
  - Role, ClusterRole, RoleBinding et ClusterRoleBinding
- GrouApi, Ressources et verbes
  - Concepts clés de RBAC dans Kubernetes
  - Groupes d'API (apiGroups)
  - Ressources
  - Verbes
  - Relations dans RBAC

- Gestion des utilisateurs et RBAC
  - Prérequis et hypothèses
  - Objects de l'API RBAC
    - Cas d'utilisation
  - Création d'utilisateurs et authentification avec les certificats clients X.509
- Authentification : certificats, tokens
- Gestion des utilisateurs et de leurs authorisations
  - Installation de KREW
  - rakkess
  - kubect-who-can
  - rbac-lookup
  - RBAC Manager

LimitRange et ResourceQuota dans Kubernetes

- Introduction aux Namespaces dans Kubernetes
  - Les namespaces offrent les avantages suivants
- Gestion des LimitRange
  - Qu'est-ce que LimitRange ?
  - Configuration d'un LimitRange
- Gestion des ResourceQuota
  - Qu'est-ce que ResourceQuota ?
- Limitation des ressources par utilisateurs : Contexte et Solutions
  - Utilisation de ResourceQuotaScopes avec des labels
- Scopes dans Kubernetes
  - Types de Scopes disponibles
  - Utilisation des scopes
  - Utilisation de scopeSelector
- PriorityClass dans Kubernetes
  - Fonctionnalité de PriorityClass
  - Configuration de PriorityClass
  - Utilisation de PriorityClass dans les Pods
  - Préemption avec PriorityClass
- TP : LimitRange, resourcequota

LES NETWORK POLICIES DANS KUBERNETES

- Introduction aux Network Policies
  - Qu'est-ce qu'une Network Policy ?
  - Composants d'une Network Policy
  - Syntaxe de base d'une Network Policy
- Travaux pratiques
  - TP1 : Ouvrir le port 80 sur un pod
  - TP2 : Restreindre le trafic entre les prods
  - TP3 : Restreindre le trafic entre les pods et namespace
  - TP4 : Autoriser le trafic sortant vers l'extérieur du cluster

INFRASTRUCTURE AS CODE, GITOPS

- Comprendre le IaC
  - Principes de base de l'IaC
  - IaC dans Kubernetes
  - Outils IaC pour Kubernetes
  - Avantages de l'IaC dans Kubernetes
- Comprendre le GitOps
  - Principes fondamentaux du GitOps
  - Fonctionnement du GitOps
  - Outils populaires de GitOps
  - Avantages du GitOps
  - Exemple de workflow GitOps avec Kubernetes
  - Conclusion
- Tour d’horizon des gestionnaires de packages pour Kubernetes Helm, Kustomize
- Qu'est-ce que Helm ?
- Qu'est-ce que Kustomize ?
- Comparaison Helm vs Kustomize
- Automatiser les déploiements avec Flux et ArgoCD
  - Flux
  - ArgoCD
  - Comparaison

INGRESS CONTROLLERS ET NGINX INGRESS CONTROLLER

- Qu'est-ce qu'un Ingres Controller ?
  - Fonctionnalités principales des Ingress Controllers
- NGINX Ingress Controller
  - Fonctionnalités principales du NGINX Ingress Controller
  - Installation et configuration du NGINX Ingress Controller
  - Installer le NGINX Ingress Controller
  - Installation via Helm
- TP : Ingress controller, Ingress

RÉSEAUX – SERVICE MESH

- Comprendre ISTIO, Cilium et les Ingress Controllers
- Choix d’un Add-On réseau sécurisé et performant
- Déployer des ingress, Gateways, route pour les applications
- Administrer les flux réseaux

SÉCURITÉ

- Sécuriser l’exécution des processus Unix dans les Pods
- SecurityContext
  - Mode privileged
  - Linux Capabilities
  - Sécurisation des processus Unix
- Industrialiser la sécurité des Pods avec les PodSecurityPolicies
- Industrialiser la sécurité du réseau (L4) avec les NetworkPolicies
- Industrialiser la gestion des certificats avec Cert-Manager
- Découvrir OPA et Falco

QUALITÉ DE SERVICE

- Utilisation optimale des ressources matérielles grâce aux Requests et Limits
- Classes de QoS
  - Guaranteed
  - Burstable
  - BestEffort
- Contrôle d'allocation des ressources par Namespace avec les ResourceQuota
- Contrôle d'allocation des ressources par Pod avec les LimitRange

OPTIMISATION DU SCHEDULER

- Contrôle de la planification avec les Labels et les Affinités
- NodeSelector, NodeAffinity, PodAffinity, PodAntiAffinity
- Taints and Tolerations

LES OPERATORS

- Présentation des méthodes d'extension de Kubernetes : les Operators
- Comprendre l'utilisation des ressources CRD
- Ajouter des API customisées à Kubernetes: les CustomResourceDefinitions
- Déployer une stack de monitoring avec l'opérateur Prometheus Kube state metrics

MONITORING

- Objectifs de surveillance et de journalisation
- Automatiser le monitoring avec l'opérateur Prometheus
- Obtenir et agréger les métriques de votre cluster et de vos applications
- Visualiser et interagir avec vos données avec Grafana

GESTION DU STOCKAGE EN PRODUCTION

- Comprendre le stockage hyperconvergé et hautement disponible
- Déploiement de ceph avec rook operator
- Déployer le stockage NAS

OPERATORS, HELM & EFK (+1 JOUR)

- Présentation des méthodes d'extension de Kubernetes : les Operators
- Ajouter des API customisées à Kubernetes: les CustomResourceDefinitions
- Créer ses opérateurs avec l'Operator-Framework et l'Operator-SDK
- Helm 2 et Helm 3
- Gestion des logs avec la pile EFK (ElasticSearch, Fluentd, Kibana)

INTRODUCTION À ISTIO & LINKERD (+1 jour - uniquement sur demande

en équipe)

- Service Mesh
- ISTIO
- LINKERD2 (Conduit)

Modules Cloud Complémentaires

Préparer la production (1 journée)

- Pipeline de CI/CD: théorie et mise en oeuvre (GihubActions/ArgoCD)
- Les Services Mesh: fonctionnement et cas pratique avec Istio
- Ingress: fonctionnement et cas pratique avec nginx-controller

Services de gestion de conteneurs du Cloud public ou Multi-Cloud: les exemples de Google Kubernetes Engine et de Rancher (½ journée)

Outils avancés de déploiement pour Kubernetes (½ journée)

- Les Operators
- Ajouter des API customisées à Kubernetes: les CustomResourceDefinitions
- Créer ses opérateurs avec l'Operator-Framework et l'Operator-SDK
- Helm: présentation et exemple avec la gestion des logs EFK (ElasticSearch, Fluentd, Kibana)

Accompagnement et conseil sur des cas pratiques proposés par les stagiaires (½ journée à 1 jour)

Sociétés concernées

Cette formation s'adresse à la fois aux particuliers ainsi qu'aux entreprises, petites ou grandes, souhaitant former ses équipes à une nouvelle technologie informatique avancée ou bien à acquérir des connaissances métiers spécifiques ou des méthodes modernes.

Positionnement à l'entrée en formation

Le positionnement à l'entrée en formation respecte les critères qualité Qualiopi. Dès son

inscription définitive, l'apprenant reçoit un questionnaire d’auto-évaluation nous permettant d’apprécier son niveau estimé sur différents types de technologies, ses attentes et objectifs personnels quant à la formation à venir, dans les limites imposées par le format sélectionné. Ce questionnaire nous permet également d’anticiper certaines difficultés de connexion ou de sécurité interne en entreprise (intraentreprise ou classe virtuelle) qui pourraient être problématiques pour le suivi et le bon déroulement de la session de formation.

Méthodes pédagogiques

Stage Pratique : 60% Pratique, 40% Théorie. Support de la formation distribué au format numérique à tous les participants.

Organisation

Le cours alterne les apports théoriques du formateur soutenus par des exemples et des séances de réflexions, et de travail en groupe.

Validation

À la fin de la session, un questionnaire à choix multiples permet de vérifier l’acquisition correcte des compétences.

Sanction

Une attestation sera remise à chaque stagiaire qui aura suivi la totalité de la formation.