

FORMATION KUBERNETES ET LES MICRO-SERVICES (3 jours + 1 à 2 jours en option)

PRÉSENTATION

Le logiciel libre Kubernetes (communément appelé « K8s ») est désormais le standard en terme d'orchestration de conteneurs. Cet outil vous permettra d'entrer dans l'ère "Cloud Native" et d'exposer à grande échelle vos applications de manière sûre, reproductible et flexible. Vous apprendrez également à faire évoluer vos applications vers le standard micro-service, modulaire et scalable. Plébiscité par les géants de la Silicon Valley, K8s est géré par une gouvernance responsable liée à Cloud Native Computing Foundation (une entité de la Fondation Linux). Kubernetes fournit une « plateforme pour automatiser le déploiement, la mise à l'échelle et la mise en production de conteneurs d'applications sur des grappes de serveurs ». Il supporte de multiples moteurs d'exécution de conteneurs dont Docker, Rocket et Singularity.

Cette formation vous présentera la toute dernière version de [Kubernetes](#) (à la date de rédaction de ce programme : [Kubernetes 1.17](#)).

OBJECTIFS

- Comprendre l'intérêt et les caractéristiques des micro-services.
- Savoir déployer des applications "Cloud-Native"
- Maîtriser kubeadm, l'outil d'installation de Kubernetes
- Comprendre comment déployer Kubernetes en production
- Déployer à grande échelle ses application sur un cluster ou dans le cloud.
- Permettre le passage à l'échelle automatique des applications
- Mettre en place la haute-disponibilité et l'auto-réparation des services logiciels
- Optimiser le stockage de grandes masses de données avec les volumes
- Disposer d'un monitoring avancé de l'infrastructure et des applications
- Automatiser la mise à jour des versions logicielles de ses applications

PUBLIC VISÉ

Développeurs, Architectes, Administrateurs systèmes, DevOps

PRÉ-REQUIS

Connaissances de base d'un système Unix

PROGRAMME

INTRODUCTION AUX MICRO-SERVICES

- Les bonnes pratiques: la méthodologie des “12 facteurs”
- Application monolithique versus Micro-services
- Faire évoluer une application vers les micro-services

RAPPELS SUR LES CONTENEURS

- Vue d'ensemble de Docker
- Vue d'ensemble des conteneurs
- Installer et exécuter des images Docker
- Interagir avec des conteneurs
- Créer ses propres images
- Différence entre les dépôts privés et publics

KUBERNETES: LES BASES

- Créer un cluster Kubernetes: sur votre poste de travail, dans votre datacenter ou dans le cloud.
- Architecture et composants de Kubernetes (côté Control Plane et Node)
- Cycle de vie d'une requête kubectl
- Déployer une application sur plusieurs machines
- Explorer une application
- Passage à l'échelle

KUBERNETES: LES CONCEPTS FONDAMENTAUX

- Vue d'ensemble des Pods
- Interagir avec les Pods
- Configuration et sécurité d'une application (ConfigMaps et Secrets)
- Vue d'ensemble des Services (ClusterIP, NodePort, LoadBalancer, Headless)

- Créer ses propres services pour exposer ses applications
- Exposer une application sur le réseau
- Organiser ses Pods avec les Labels

DEPLOYER SES MICRO-SERVICES

- Stratégies de déploiement en mode “Cloud-Native”
- Stratégies de calcul intensif (Jobs)
- Cas pratique: déploiements avec kubectl et yaml
- Stratégies de passage à l'échelle (Replicasets et Daemonsets)
- Cas pratique: utilisation des réplicas
- Cas pratique: installation d'un gestionnaire de journaux distribués
- Stratégie de mise à jour logicielle (Deployments)
- Cas pratique: Rolling update
- Gérer simplement ses mises à jour applicatives
- Techniques avancées : déploiement blue/green, canary

CONCEPTS AVANCÉS

- Stockages volatiles et persistants (PersistentVolume/PersistentVolumeClaim)
- Techniques de supervision avancées: Prometheus
- Déploiement des bases de données distribuées (StatefulSet)
- Cas pratique: installation de MongoDB et Redis en mode distribué

EN OPTION

- Les Services Mesh: fonctionnement et cas pratique avec Istio (½ jour)
- Ingress: fonctionnement et cas pratique avec Traefik (½ jour)
- Accompagnement et conseil sur des cas pratiques proposés par les stagiaires (½ à 1 jour)
- Services de gestion de conteneurs du Cloud public ou Multi-Cloud: les exemples de Google Kubernetes Engine et de Rancher (½ journée)