L’essentiel du cloud native

Prix : 750 €HT

Durée : 1 jour

Code de Référence : CNESS

Catalogue Cloud Native et DevOps

Les conteneurs et leur orchestration avec Kubernetes sont devenus les fondements des architectures Cloud Native modernes. Cette journée de formation donne aux participants une vision claire et opérationnelle de ces technologies : comprendre pourquoi et comment les conteneurs ont transformé le cycle de vie applicatif, maîtriser les briques essentielles de Docker et Kubernetes, et poser les bases d’une stratégie d’intégration CaaS (Container as a Service) adaptée à leur organisation.

La formation alterne apports théoriques et démonstrations illustratives.

Objectifs de la formation

A l’issue de cette formation, vous serez capable de :

- Expliquer les enjeux et les apports des conteneurs dans un système d’information moderne.
- Distinguer les principales technologies de conteneurs : Docker, containerd, Podman, et leurs cas d’usage.
- Décrire l’architecture technique de Kubernetes et ses ressources fondamentales : pods, deployments, services.
- Identifier les critères de choix d’une plateforme CaaS selon les contraintes de son organisation.
- Esquisser les bases d’une stratégie d’intégration Cloud Native et DevOps dans son SI.

Public

Cette formation s’adresse aux DSI, architectes techniques, ingénieurs système et réseau, chefs de projets IT, administrateurs systèmes, développeurs et consultants Cloud et DevOps.

Prérequis

Connaissance de la terminologie et des concepts d'architecture informatique.
Connaissances de base en Cloud et en environnements Linux.

Programme de la formation

Les démonstrations, exemples et travaux pratiques sont donnés à titre indicatif et pourront être adaptés selon les besoins et le contexte de la session.

Jour 1
Matinée – Histoire, technologies et architecture des conteneurs
Chapitre 1 – Origines et fondements des conteneurs

- Pourquoi les conteneurs ont-ils émergé ? État de l'art et évolution
- En quoi changent-ils la manière de packager, déployer et maintenir une application ?
- Conteneurs vs machines virtuelles : différences de fond et complémentarité
- Cas d'usage adaptés et non adaptés aux conteneurs

Démonstration :

- Illustration du cycle de vie d'un conteneur : image, conteneur, exécution

Chapitre 2 – Panorama des technologies et acteurs

- Briques du moteur de conteneurs : Docker Engine, containerd, CRI-O, Podman
- Alternatives à Docker et raisons de migrer
- Acteurs du marché CaaS : Docker, Red Hat, Rancher, AWS, Google Cloud, Microsoft Azure
- Gouvernance de l'écosystème conteneurs : CNCF et OCI

Démonstration :

- Passage de Docker à containerd pour illustrer l'interopérabilité

Chapitre 3 – Fonctionnement technique d'un conteneur

- Image, couche, volume : définitions et interactions
- Isolation : namespaces et cgroups
- Communication entre conteneurs : réseau, DNS, ports
- Persistance des données dans Docker

Démonstration :

- Construction d'une image Docker personnalisée et persistance via volume

Après-midi – Kubernetes, stratégie CaaS et vision Cloud Native
Chapitre 4 – Introduction à l'orchestration avec Kubernetes

- Pourquoi un orchestrateur ? Problèmes résolus par Kubernetes
- Architecture technique de Kubernetes : control plane, worker nodes, etcd
- Ressources fondamentales : pods, deployments, services, ingress
- Extensibilité et personnalisation de Kubernetes

Démonstration :

- Déploiement d'un cluster Kubernetes avec minikube
- Présentation des principales ressources Kubernetes

Chapitre 5 – Cadrer son projet CaaS et choisir sa plateforme

- Critères pour sélectionner les premières applications à conteneuriser
- Cloud, on-premise, edge : où démarrer ? Peut-on faire du CaaS sans Kubernetes ?
- Critères de choix d'une distribution : AKS, GKE, EKS, OpenShift, Rancher
- Supervision, logging, sécurité, sauvegarde : les incontournables

Démonstration :

- Déploiement d'une application complète sur Kubernetes avec autoscaling
- Présentation d'outils d'observabilité

Chapitre 6 – Vers une stratégie Cloud Native et DevOps

- Impact du CaaS sur le cycle de vie applicatif et la CI/CD
- GitOps et Infrastructure as Code : Terraform, Argo CD, dans un modèle Cloud Native
- Culture DevOps : faire évoluer l'organisation entre développement, opérations et gouvernance
- Multi-cloud, interopérabilité et futur du CaaS : WebAssembly, eBPF, serverless conteneurisé

Démonstration :

- Déploiement entièrement automatisé en pipeline CI/CD

Méthodes pédagogiques

- Séminaire ponctué de démonstrations illustratives pour ancrer les concepts théoriques abordés.

- Le formateur alterne méthode démonstrative, interrogative et active, en s'appuyant sur les contextes des participants pour enrichir les échanges.

- Le format d'une journée privilégie une vision claire et opérationnelle des conteneurs, de Kubernetes et des principaux choix d'architecture Cloud Native.

Les supports de formation seront fournis aux participants au cours de la formation au format PDF.

Méthodes d'évaluation des acquis

Avant la formation :

- Questionnaire de positionnement et d'auto-évaluation des compétences, complété individuellement par chaque stagiaire avant la session.

En cours de formation :

- Points d’étapes réguliers du formateur sur la compréhension des stagiaires.
- Vérification des acquis lors des démonstrations, échanges guidés et mises en situation.

Après la formation « à chaud » :

- Questionnaire d’auto-évaluation des compétences complété individuellement par chaque stagiaire en fin de session.
- Questionnaire de satisfaction complété individuellement en fin de session.
- Le compte rendu formateur complété par le formateur.

Après la formation « à froid » :

- Questionnaire de satisfaction complété individuellement quelques semaines après la session.

Un certificat de réalisation de fin de formation est remis au stagiaire lui permettant de faire valoir le suivi de la formation.