

Docker : administration avancée

Durée: 2 jours

1215 €
HT

14 au 15 mars
24 au 25 mai

20 au 21 septembre
28 au 29 novembre

Public:

Administrateurs, chefs de projet et toute personne souhaitant maîtriser les concepts avancés de Docker

Objectifs:

Savoir configurer les fonctionnalités avancées de Docker : la sécurité, les configurations multi-hôtes, la création de registres privés, le provisioning de services dans le cloud, ...

Connaissances préalables nécessaires:

Il est demandé aux participants de connaître les bases du système Unix/Linux et les bases de Docker, ou d'avoir suivi le stage "Docker : mise en oeuvre".

Programme:

Docker engine : Fonctionnalités, installation et configuration

Le service Docker : Docker daemon : rôle, configuration des principales options.
Option socket pour les accès en réseau.
Variables d'environnement : DOCKER_HOST, et DOCKER_TLS_VERIFY
Option storage-driver :
définition des formats de stockage des images.
Gestion de noeuds avec l'option -cluster-advertise
Travaux pratiques :
configuration des accès réseau et de clusters Docker

Création d'un registry privé : Présentation de Docker Trusted Registry (DTR).
Architecture. Containers et volumes propres au DTR
Pilotage par UCP (Universal Control Plane).
Travaux pratiques :
installation d'un dépôt privé.
Gestion des images du DTR, des droits d'accès.

Docker : administration avancée

- Administration en production** : Applications multi-containers avec Compose: définition de l'environnement applicatif, déclaration des services dans docker-compose.yml, exécution avec docker-compose.
Méthodes d'administration de containers en production.
Orchestration avec Docker Machine.
Travaux pratiques :
exemples de provisioning en environnement mixte, dans le cloud et sur des machines physiques.
Présentation de Swarm pour le clustering : fonctionnalités, gestion de clusters docker, équilibrage de charge, répartition de tâches, gestion de services répartis, ..
- Sécurité** : Analyse des points à risques : le noyau, le service Docker, les containers, .. et des types de dangers : déni de service, accès réseau non autorisés, ..
Mécanismes de protection :
pile réseau propre à chaque container, limitations de ressources par les cgroups, restrictions des droits d'accès sur les sockets, politique de sécurité des containers.
Travaux pratiques :
mise en évidence de failles de sécurité et des bonnes pratiques à adopter.
Sécurisation des clients par des certificats
Principe, et mise en oeuvre avec openssl.
Fiabilité des images déployées dans Docker: présentation de Content Trust pour signer les images.
Exercices pratiques :
activation de Content Trust,
variable d'environnement DOCKER_CONTENT_TRUST,
Création et déploiement d'images signées.
Configuration réseau, sécurité et TLS