Date de mise à
jour 30/01/2024 Version 1.2

FORMATION : Docker
REFERENCE : LmsiDocker
TARIF INTER : 2000€/ht soit 2400€/ttc
TARIF INTRA : 1600 €/ht soit 1920 €/ttc

Date de mise à jour
30/01/2024 Version 1.2

Indicateur annuel de satisfaction 100% **Nombre de stagiaires ayant participés** 1

OBJECTIFS DE LA FORMATION

Connaître les caractéristiques d'un conteneur Linux et découvrir Docker -Installer et utiliser Docker, connaître les fonctionnalités avancées - Maîtriser les images Docker et leur cycle de vie, rédiger des "Dockerfile" - Connaître et configurer une Registry (publique et privée) - Maîtriser les notions réseaux de Docker (drivers, links) - Comprendre et maîtriser la persistance des données (drivers, volumes) - Maîtriser la notion de service et stack Docker avec Docker-compose - Utiliser Docker Swarm pour déployer une stack orientée Production - Maîtriser les bonnes pratiques - Maîtriser la sécurité de sa plateforme docker

A QUI S'ADRESSE CETTE FORMATION

PROFIL

Administrateurs de systèmes ainsi que les techniciens chargés du support mais également toute personne impliquée dans la sécurité du système d'information.

PRE-REQUIS

Notions Bases de l'administration systèmes sous Linux et Windows

LES PLUS DE CETTE FORMATION

Cette formation permet de comprendre la conteneurisation des applications

MODALITES PEDAGOGIQUES

La FOAD chez LMSI ne se contente pas de «dispenser» des cours. Un réel suivi pédagogique est effectué grâce aux outils d'échange, ce qui permet un ajustement permanent des contenus à la nécessité des auditeurs.
«Chat», mail, forums privés avec le formateur, plages horaires de contacts téléphoniques, visio-conférence: tous les moyens nécessaires peuvent être mis en oeuvre pour que l'auditeur soit accompagné.
Avant le début de la formation le stagiaire n'a pas d'exercice à réaliser dans sa démarche d'apprentissage.
C'est un élément fondamental de réussite qui n'est jamais négligé.
La formation sera jalonnée de tp sur des machines qui sont mises à disposition et que le stagiaire pour prendre en main à distance via l'outil de visio conférence. Ces tp seront suivis par le formateur.
Les situations de handicap seront étudiés au cas par cas.

MOYENS DE FORMATION

Formateur expert du domaine, un ordinateur par personne

MODALITES DE REALISATION

Classe à distance (prévoir une connexion adsl 8Mb/s minimum)ou présentiel, la formation se déroule sur site chez le client.

MODALITES D'EVALUATION

Au cours de la formation, suivi du responsable de projet. Évaluation écrite à l'issue de la formation où le participantdonnera ses impressions par écrit et à l'oral.

NOTRE ENGAGEMENT QUALITE

ORGANISATION

DUREE

2 jours soit 14 heures

EFFECTIF

Minimum : 1 personne **Maximum : 6 personnes**

La session peut être annulée 48h avant la date de réalisation s'il n'y a qu'un seul participant, car pour nous la formation sécurité est aussi un moment de partage des bonnes pratiques entre les différents stagiaires.

SUIVI

Feuille d'émargement signée par ½ journée, évaluation de satisfaction et d'auto-évaluation stagiaire. Attestation de stage

INTERVENANT

Damien Lamy, intervient régulièrement dans diverses entreprises depuis plus de 10 ans, membre de la réserve citoyenne de cyberdéfense et l'auteur de la vidéo Kali Linux aux éditions ENI, réalise des audits réseaux depuis 20 ans.

RESPONSABLE PEDAGOGIQUE ET ADMIN.

Damien LAMY 04 58 17 28 58 dlamy@lmsi.eu

Contact HANDICAP

Damien LAMY 04 58 17 28 58 dlamy@lmsi.eu

DATES DE FORMATION

Pour toute demande de calendrier adressez un mail f ormation@lmsi.eu en précisant le nom du stage.

CONTACT ET MODALITES D'INSCRIPTION

Pour s'inscrire à la formation ou recevoir des informations envoyer un mail à l'adresse f ormation@lmsi.eu en précisant lecode de la formation.
Contact Lamy Damien 04 58 17 28 58

COORDONNES DE L'ORGANISME DE FORMATION

Lmsi sasu Capital 1000€ http://www.lmsi.eu 418 chemin des soupirs 26210 Epinouze Organisme de formation enregistré 84260258126Siret: 817 999 709 00014 Code APE 6202A

FORMATION : Docker
REFERENCE : LmsiDocker
TARIF INTER : 2000€/ht soit 2400€/ttc
TARIF INTRA : 1600 €/ht soit 1920 €/ttc

LES MODULES DE FORMATION

MODULE 1 – Introduction (Jour 1 matin)

- Les différentes formes de virtualisation et leur concept
- Présentation des avantages et des cas d'utilisation des conteneurs
- Présentation de Docker et de son architecture

MODULE 2 – Prendre en main Docker (Jour 1 matin)

- Installer Docker
- Utiliser les commandes de base du client Docker
- Expliquer un conteneur et son cycle de vie
- Instancier un conteneur (mode interactif, mode détaché)
- Administrer et superviser un conteneur depuis le docker host (exec, inspect, logs...)

MODULE 3 - Manipuler des images Docker (Jour 1 apm)

- Présentation du concept d'images Docker (Docker Hub, images personnalisées)
- Les différentes méthodes de conception d'une image Docker
- Créer une image à partir d'un conteneur (commit)
- Créer une image à partir d'un Dockerfile
- Les instructions dans un Dockerfile (FROM, COPY, ADD, EXPOSE, ENTRYPOINT, CMD)
- Gérer le cycle de vie des images (labels, tags, versionning mineur/majeur)
- Sélectionner et récupérer une image depuis la communauté "Docker Hub"
- Le concept des layers et du cache (optimisation)
- La registry et le stockage des images (registry privée, registry "Docker Hub")

MODULE 4 - Configurer le réseau pour Docker (Jour 1 apm)

- Le conteneur dans son réseau (stack réseau Docker)
- Le port forwarding (PAT)
- Liaisonner des conteneurs (links)
- Les différents réseaux proposés par Docker (drivers, les impacts et cloisonnements)

MODULE 5 - Gérer les systèmes de fichiers pour Docker (Jour 1 apm)

- Le principe de volumes associés à un conteneur
- Créer et persister des volumes docker
- Gérer les modèles de configuration et leurs bonnes pratiques

MODULE 6 - Réaliser une Infrastructure As Code avec Docker (Jour 1 apm)

- Introduction au DevOps
- Docker-compose : la solution pour créer, assembler et administrer son service de conteneurs
- Mettre en place un contrôle de l'exécution

MODULE 7 - Appréhender le déploiement à grande échelle avec Docker (Jour 2 matin)

- Les enjeux
- Docker-machine (créer rapidement sa plateforme Docker avant mise en production)
- L'orchestrateur Swarm : nodes, services
- Déploiement de services et stacks dans un Swarm
- Comment sécuriser l'infrastructure Docker (TLS/SSL, Apparmor et SeLinux)
- Interface de management (Portainer)
- Présentation des outils de déploiements et de gestion de configuration (ansible, puppet, salt)
- Présentation des différents orchestrateurs

MODULE 8 - Optimiser la conception d'images (Jour 2 matin)

- Rappels sur la conception des "Dockerfiles
- Développer une conception et une gestion fine du cycle de vie des images
- Justifier la gestion du cache avec les "layers"
- Rompre avec le systeme d'idempotence
- Construite une image en "multi-stage builds"
- Contrôler l'état applicatif dans l'image
- Identifier les projets communautaires incontournables : analyse, métrics, reverse-proxy, sécurité

MODULE 9 - Restructurer docker-compose (Jour 2 apm)

- Rappels sur les concepts
- Assembler les ressources (les services, les réseaux, les volumes)
- Intégrer intelligemment les variables d'environnement
- Adapter les contextes de build : "Dockerfile
- Résoudre les dépendances entre services
- Mettre en place un contrôle de l'exécution
- Industrialiser une stack docker-compose

MODULE 10 - Construire un environnement de production : clustering, orchestration et monitoring Docker (Jour 2 apm)

- Définir les enjeux d'un orchestrateur
- Expliquer Swarm et ses fonctionnalités
- Illustrer la notion de nodes (manager, worker)
- Examiner le réseau et les "topologies mesh"

MODULE 11 - Gérer le cycle de vie approfondi des conteneurs (Jour 2 apm)

- Consolider les ressources à travers les commandes docker update
- Délimiter les domaines d'exécutions des conteneurs (les labels, placements par node)
- Utiliser les mécanismes de "rolling update" et "rollback"

MODULE 12 - Appréhender la sécurité pour Docker (Jour 2 apm)

- Sécuriser la plateforme avec TLS/SSL (client, hotes, registry)
- Identifier les risques : noyau, service Docker, containers, déni de service, accès réseau
- Utiliser des mécanismes de protection : "subnet" spécifique par application, limitations de ressources par les "cgroups", restrictions des droits d'accès sur les sockets, politique de sécurité des containers
- Examiner les "events" docker
- Fiabiliser les images déployées dans Docker : présentation de "Content Trust" pour signer les images