

Catalogue de Formations

@mjbright CONSULTING

training@mjbright.net

Adaptation

Toutes les formations peuvent être adaptées à vos besoins sur:

- les sujets abordés
- la durée, le fuseau horaire (max 8h entre 7h à 23h59 CET si virtuel)
- Par demi-journée ou journée
- Le nombre de jours peut-être réduit ou étendues
- en anglais ou français (éventuellement avec supports en anglais)
- en virtuelle de préférence, ou chez vous ou en salle de louée
- selon les normes Qualiopi si besoin

Legende

Disponible

Sur demande

Plutard

Indexe de Formations

Categorie	Formation
Linux	[1j] BELOW LIN-001-FR Introduction a Linux - en ligne de commande / shell
Generale	[1j] BELOW GEN-001-FR Conteneurs pour les Dirigeants
Kubernetes	[2j] BELOW K8S-001-FR Introduction a Kubernetes
	[4j] <u>BELOW</u> K8S-002-FR Administration Kubernetes
	[3j] <u>BELOW</u> K8S-003-FR Development avec Kubernetes
	[4j] BELOW K8S-004-FR Securite de Kubernetes
	[2j] BELOW K8S-010-FR Kubernetes Avance
	[1j] <u>BELOW</u> K8S-011-FR Introduction a Helm
	[1j] BELOW K8S-012-FR Kubernetes Policy Management (OPA)
	[3j] BELOW K8S-013-FR Kubernetes Policy Management (Kyverno)

	[2j] BELOW K8S-014-FR Deploying Kubernetes with Istio Service Mesh
	[2j] BELOW K8S-015-FR Deploying Kubernetes with Linkerd Service Mesh
	[1j] BELOW K8S-004-FR Preparation a l'examen CKA
	[1j] BELOW K8S-005-FR Preparation a l'examen CKAD
	[2j] BELOW K8S-006-FR Preparation a l'examen KCNA
	[4j] BELOW K8S-007-FR Preparation a l'examen CKS
Docker	[2j] BELOW DO-001-FR Introduction a Docker
	[2j] BELOW DO-002-FR Docker Avance
Terraform / OpenTofu	[2j] BELOW TF-000-FR Introduction a Terraform (CORE)
	[1j] BELOW TF-001-FR Introduction a Terraform (AWS)
	[1j] BELOW TF-002-FR Introduction a Terraform (Azure)
	[1j] BELOW TF-003-FR Introduction a Terraform (Google Cloud)
	[1j] BELOW TF-004-FR Introduction a Terraform (Oracle Cloud)
	[1j] BELOW TF-005-FR Introduction a Terraform (OpenStack)
	[1j] BELOW TF-006-FR Introduction a Terraform (Proxmox)
	[2j] BELOW TF-011-FR Terraform Avance (AWS)
	[2j] BELOW TF-012-FR Terraform Avance (Azure)
	[2j] BELOW TF-013-FR Terraform Avance (Google Cloud)
	[2j] BELOW TF-014-FR Terraform Avance (Oracle Cloud)
	[2j] BELOW TF-015-FR Terraform Avance (OpenStack)
	[2j] BELOW TF-016-FR Terraform Avance (Proxmox)
	[2j] BELOW TF-021-FR Introduction a OpenTofu (AWS)
Configuration Management	
	[2j] BELOW AN-002-FR Introduction a Ansible

	[3j] BELOW AN-003-FR Introduction a Ansible
Policy Management	[2j] BELOW POL-001-FR Introduction to Policy Management with OPA/Rego
	[1j] BELOW K8S-010-FR Kubernetes Policy Management (OPA)
	[3j] BELOW K8S-011-FR Kubernetes Policy Management (Kyverno)
WebAssembly	[2j] BELOW WASM-001-FR Introduction a WebAssembly

Formations

Kubernetes

[4j] K8S-002-FR Administration Kubernetes

K8S-002-FR Administration Kubernetes

Objectifs

L'objectif de cette formation est d'acquérir les bases sur l'administration de Kubernetes, dans le but de:

- * Apprendre sur l'orchestration des conteneurs
- * Maîtriser l'installation, configuration d'une plateforme Kubernetes
- * Connaître les principaux types d'objets gérés par Kubernetes
- * Comprendre comment déployer sur Kubernetes
- * Savoir déboguer et observer les applications ainsi que le cluster Kubernetes même

La formation serait constituée d'environ 50% théorie et 50% travaux pratiques.

Note: Bien que la formation peut-être une bonne préparation pour l'examen CKA, elle peut être associée à la formation *"K8S-004-FR Preparation a l'examen CKA"*

C'est une journée dédiée aux compétences nécessaires pour réussir la certification.

Nous conseillons de faire une pause entre les deux formations.

Acquis à l'issue de la formation

Le participant à cette formation aura acquis des bases pour administrer une plateforme Kubernetes, en particulier:

- * Installation, Configuration, Mise à jour d'un Cluster Kubernetes
- * Comment déployer et mettre à jour des applications sur Kubernetes
- * Comment exposer des applications
- * Déploiement des volumes de données externes
- * Observation, Débogage d'applications, du cluster
- * La sécurité, Les meilleurs pratiques

Public concerné et pré-requis

Cette formation vise des ingénieurs qui débutent avec l'administration de Kubernetes.

Les participants doivent venir avec leur propre PC et avoir accès à Internet:

- * via un navigateur pour se connecter à des sites hébergés chez AWS
- * ainsi qu'à des VMs EC2 hébergées chez AWS par SSH: moyens de connexion alternatif peuvent être fournis

Pour tirer pleinement parti de cette formation, les participants:

- * doivent être à l'aise pour travailler avec la ligne de commande
- * avoir de notions de processus Linux
- * avoir de notions de conteneurs et Docker
- * Disposer d'un client SSH tel que le client openssh sur Linux, macOS, WSL ou Putty sur Windows

En cas de manquement il est conseillé de d'abord suivre la formation "*LIN-001-FR - Introduction a Linux - en ligne de commande / shell*" ou equivalent.

Evaluation

En début de formation, nous vérifions l'expérience dans la domaine ainsi que les attentes de chaque participant.

Si en mode Qualiopi

En début de formation, une auto évaluation est effectuée pour évaluer le niveau, ainsi que les attentes de chaque participant. Ainsi nous pouvons organiser au mieux les différents groupes de travail.

Le formateur effectue ensuite une évaluation continue à l'aide d'exercices pratiques.

À la fin de la formation, les participants auto-évaluent leur progression, et le formateur fournit une évaluation ainsi que des axes d'amélioration.

En option, un accompagnement post-formation peut être proposé, faisant l'objet d'un devis complémentaire.

Programme

Proposé sur 4 jours, mais peut-être étendu a 5 jours avoir plus de temps pour les travaux pratiques et ainsi mieux assimiler les concepts et la pratique.

A partir d'un exemple concret, nous regarderons les aspects suivants

MODULE: Principes et Concepts de Kubernetes:

- * Reprise de principes de Conteneurs
- * Pourquoi les Pods - comparaison avec les conteneurs ?
- * Lancer et Interagir avec un Pod (ports, exec, logs)
- * Résilience à l'échelle "a la Kubernetes"
- * Principes de bases: couplage faibles, état désiré, Namespaces, Pods, Controllers, Services
- * Installation d'un cluster Kubernetes (kubeadm) et CNI (Cilium)
- * Le modele reseau
- * Kubeconfig
- * Processus de boot, static pods

MODULE: Déploiement d'applications avec kubernetes:

- * L'architecture
- * Pods - init container, multi-container patterns
- * Gestion de ressources par Container, par Namespace
- * API Kubernetes - API groups, explain, accès API
- * Workload controllers
- * Mise à jour d'applications
- * Stockage: basic, par PV/PVC, dynamic

MODULE: Extensions/outils, Exposer des applications:

- * ConfigMaps: creation, utilisation, mise à jour
- * Secrets: création, utilisation, mise à jour
- * Outils d'extension - Helm, CRDs, Operators
- * Exposer des applications:
 - * Services
 - * Ingress controller
 - * Gateway API
- * ServiceMesh

MODULE: Observabilité, Débogage, Sécurité, HA:

- * Observabilité
 - * MetricsServer, Dashboard
 - * Logging - EFK, Loki
 - * Metrics - Prometheus/grafana
- * Troubleshooting
 - * Application troubleshooting avec challenge
- * Sécurité
 - * SecurityContext & PodSecurityStandard
 - * APIServer controles - Authentication, Authorisation, Admission Control
 - * Policy Engines: OPA, Kyverno
 - * Networkpolicies
- * Cluster HA
- * Conseils pour l'examen CKA

K8S-003-FR Development avec Kubernetes

Objectifs

L'objectif de cette formation est d'acquérir les bases de l'utilisation et développement d'applications sur Kubernetes, dans le but de:

- * Apprendre sur l'orchestration des conteneurs
- * Maîtriser la création et hébergement d'images d'applications
- * Comment développer les applications en suivant les principes du Cloud Natif
- * Connaître les principaux types d'objets gérés par Kubernetes
- * Comprendre comment déployer sur Kubernetes
- * Savoir déboguer et observer les applications sur Kubernetes
- * Comment sécuriser ses applications, Les meilleurs pratiques

La formation serait constituée d'environ 50% théorie et 50% travaux pratiques.

Note: Bien que la formation peut-être une bonne préparation pour l'examen CKAD, elle peut être associée à la formation "*K8S-005-FR Préparation à l'examen CKAD*"

C'est une journée dédiée aux compétences nécessaires pour réussir la certification.

Nous conseillons de faire une pause entre les deux formations.

Acquis à l'issue de la formation

Le participant à cette formation aura acquis des bases pour administrer une plateforme Kubernetes, en particulier:

- * Comment créer des images de conteneurs pour les applications
- * Comment héberger des images de conteneurs
- * Comment déployer et mettre à jour des applications sur Kubernetes
- * Comment exposer des applications
- * Déploiement des volumes de données externes
- * Observation, Débogage d'applications

Public concerné et pré-requis

Cette formation vise des ingénieurs qui débutent avec le développement d'applications pour Kubernetes.

Les participants doivent venir avec leur propre PC et avoir accès à Internet:

- * via un navigateur pour se connecter à des sites hébergés chez AWS
- * ainsi qu'à des VMs EC2 hébergées chez AWS par SSH: moyens de connexion alternatifs peuvent être fourni

Pour tirer pleinement parti de cette formation, les participants:

- * doivent être à l'aise pour travailler avec la ligne de commande
- * avoir de notions de processus Linux
- * avoir de notions de conteneurs et Docker
- * Disposer d'un client SSH tel que le client openssh sur Linux, macOS, WSL ou Putty sur Windows

En cas de manquement il est conseillé de d'abord suivre la formation "*LIN-001-FR - Introduction a Linux - en ligne de commande / shell*" ou equivalent.

Evaluation

En début de formation, nous vérifions l'expérience dans la domaine ainsi que les attentes de chaque participant.

Si en mode Qualiopi:

En début de formation, une auto évaluation est effectuée pour évaluer le niveau, ainsi que les attentes de chaque participant. Ainsi nous pouvons organiser au mieux les différents groupes de travail.

Le formateur effectue ensuite une évaluation continue à l'aide d'exercices pratiques.

À la fin de la formation, les participants auto-évaluent leur progression, et le formateur fournit une évaluation ainsi que des axes d'amélioration.

En option, un accompagnement post-formation peut être proposé, faisant l'objet d'un devis complémentaire.

Programme

Proposé sur 3 jours, mais peut-être étendu a 4 jours avoir plus de temps pour les travaux pratiques et ainsi mieux assimiler les concepts et la pratique.

A partir d'un exemple concret, nous regarderons les aspects suivants

MODULE: Principes et Concepts de Kubernetes:

- * Reprise de principes de Conteneurs
- * Pourquoi les Pods - comparaison avec les conteneurs ?
- * Lancer et Interagir avec un Pod (ports, exec, logs)
- * Résilience à l'échelle "a la Kubernetes"
- * Principes de bases: couplage faibles, état désiré, Namespaces, Pods, Controllers, Services
- * Installation d'un cluster Kubernetes (kubeadm) et CNI (Cilium)
- * Le modele reseau
- * Kubeconfig
- * Processus de boot, static pods

MODULE: Déploiement d'applications avec kubernetes:

- * L'architecture
- * Pods - init container, multi-container patterns
- * Gestion de ressources par Container, par Namespace
- * API Kubernetes - API groups, explain, accès API
- * Workload controllers
- * Mise à jour d'applications
- * Stockage: basic, par PV/PVC, dynamic
- * ConfigMaps: creation, utilisation, mise à jour
- * Secrets: création, utilisation, mise à jour
- * Installation et packaging d'applications avec Helm

MODULE: Exposer des applications, Observabilité, Débogage, Sécurité:

- * Exposer des applications:
 - * Services

- * Ingress controller
- * Gateway API
- * Observabilité
 - * MetricsServer
 - * Logging
 - * Prometheus/grafana
- * Débogage
 - * Application troubleshooting avec challenge
- * Sécurité
 - * SecurityContext & PodSecurityStandard
 - * APIServer controls - Authentication, Authorisation, Admission Control
 - * Networkpolicies
- * Conseils pour l'examen CKAD

K8S-020-FR Preparation a l'examen CKA

Objectifs

L'objectif de cette formation d'une journée est de préparer pour passer l'examen "*CKA - Certified Kubernetes Administrator*" de la Linux Foundation.

Les stagiaires doivent déjà maîtriser la théorie et la pratique de Kubernetes - ayant suivi le "*K8S-002-FR Administration Kubernetes*" par exemple ou équivalent.

Les stagiaires seraient mis en situation avec des tâches à accomplir - voir plus bas pour plus de détails.

La formation serait constituée principalement (90%) travaux pratiques.

Acquis à l'issue de la formation

Le participant à cette formation serait en bon mesure de passer la certification

Public concerné et pré-requis

Cette formation vise des ingénieurs qui ont déjà une première expérience d'administration de Kubernetes.

Note: De suivre K8S-002-FR et K8S-004-FR seule n'est pas suffisant pour passer l'examen - il faut pratiquer. Nous fournissons des astuces, la pratique et des ressources pour bien préparer la certification.

Les participants doivent venir avec leur propre PC et avoir accès à Internet:

- * via un navigateur pour se connecter à des sites hébergés chez AWS
- * ainsi qu'à des VMs EC2 hébergées chez AWS par SSH: moyens de connexion alternatifs peuvent être fourni

Pour tirer pleinement parti de cette formation, les participants:

- * doivent être à l'aise pour travailler avec la ligne de commande
- * avoir de notions de processus Linux
- * avoir de notions de conteneurs et Docker
- * Disposer d'un client SSH tel que le client openssh sur Linux, macOS, WSL ou Putty sur Windows

En cas de manquement il est conseillé de d'abord suivre la formation "*LIN-001-FR - Introduction a Linux - en ligne de commande / shell*" ou équivalent.

Evaluation

En début de formation, nous vérifions l'expérience dans la domaine ainsi que les attentes de chaque

participant.

Programme

Au long de la journée, les stagiaires seront mis en situation avec des tâches à accomplir - similaires à ceux rencontrés en examen - du genre

* Implémenter un scénario:

- * Un déploiement où les Pods accèdent à un ConfigMap monté en Volume
- * Un DaemonSet où les Pods accèdent à un Secret par des variables d'environnement
- * Empêcher des Pods en Namespace 'frontend' de communiquer directement avec des Pods du Namespace 'backend'

* Déboguer des scénarios:

- * un Service qui ne répond pas aux requêtes, ou de façon intermittente
- * Un Pod qui ne démarre pas, ou un Conteneur qui redémarre sans cesse
- * kubectl qui fait timeout

K8S-021-FR Preparation a l'examen CKAD

Objectifs

L'objectif de cette formation d'une journée est de préparer pour passer l'examen "*CKAD - Certified Kubernetes Application Developer*" de la Linux Foundation.

Les stagiaires doivent déjà maîtriser la théorie et la pratique de Kubernetes - ayant suivi le "*K8S-003-FR Development avec Kubernetes*" par exemple ou équivalent.

Les stagiaires seraient mis en situation avec des tâches à accomplir - voir plus bas pour plus de détails.

La formation serait constituée principalement (90%) travaux pratiques.

Acquis à l'issue de la formation

Le participant à cette formation serait en bon mesure de passer la certification

Public concerné et pré-requis

Cette formation vise des ingénieurs qui ont déjà une première expérience de développement d'applications pour et utilisation d'un cluster Kubernetes.

Note: De suivre K8S-003-FR et K8S-005-FR seule n'est pas suffisant pour passer l'examen - il faut pratiquer. Nous fournissons des astuces, la pratique et des ressources pour bien préparer la certification.

Les participants doivent venir avec leur propre PC et avoir accès à Internet:

- * via un navigateur pour se connecter à des sites hébergés chez AWS
- * ainsi qu'à des VMs EC2 hébergées chez AWS par SSH: moyens de connexion alternatifs peuvent être fourni

Pour tirer pleinement parti de cette formation, les participants:

- * doivent être à l'aise pour travailler avec la ligne de commande
- * avoir de notions de processus Linux
- * avoir de notions de conteneurs et Docker
- * Disposer d'un client SSH tel que le client openssh sur Linux, macOS, WSL ou Putty sur Windows

En cas de manquement il est conseillé de d'abord suivre la formation "*LIN-001-FR - Introduction a Linux - en ligne de commande / shell*" ou equivalent.

Evaluation

En début de formation, nous vérifions l'expérience dans la domaine ainsi que les attentes de chaque participant.

Programme

Au long de la journée, les stagiaires seraient mis en situation avec des tâches à accomplir - similaires à ceux rencontrés en examen - du genre

* Implémenter un scénario:

- * Un déploiement où les Pods accèdent à un ConfigMap monté en Volume
- * Un DaemonSet où les Pods accèdent à un Secret par des variables d'environnement
- * Empêcher des Pods en Namespace 'frontend' de communiquer directement avec des Pods du Namespace 'backend'

* Déboguer des scénarios:

- * un Service qui ne répond pas aux requêtes, ou de façon intermittente
- * Un Pod qui ne démarre pas, ou un Conteneur qui redémarre sans cesse
- * kubectl qui fait timeout

K8S-011-FR Introduction a Helm

Objectifs

L'objectif de cette formation est de permettre aux participants d'acquérir les bases de l'utilisation de l'outil Helm pour créer et installer des Charts d'applications sous Kubernetes. Les principaux objectifs sont les suivants :

- * Faciliter la gestion d'applications sur Kubernetes
- * Personnaliser les applications au moment de leur installation

La formation est équilibrée entre la théorie (50%) et les travaux pratiques (50%).

Acquis à l'issue de la formation

À la fin de la formation, les participants auront acquis des compétences pour administrer les applications sur une plateforme Kubernetes en utilisant Helm, notamment :

- * Installation, configuration, mise à jour d'une application Kubernetes
- * Création et test d'un chart Helm

Public concerné et pré-requis

Cette formation vise des ingénieurs qui ont déjà une première expérience de développement d'applications pour et utilisation d'un cluster Kubernetes.

Les participants doivent venir avec leur propre PC et avoir accès à Internet:

- * via un navigateur pour se connecter à des sites hébergés chez AWS
- * ainsi qu'à des VMs EC2 hébergées chez AWS par SSH: moyens de connexion alternatifs peuvent être fourni

Pour tirer pleinement parti de cette formation, les participants:

- * doivent être à l'aise pour travailler avec la ligne de commande
- * avoir de notions de processus Linux
- * avoir de notions de conteneurs et Docker
- * Disposer d'un client SSH tel que le client openssh sur Linux, macOS, WSL ou Putty sur Windows

En cas de manquement il est conseillé de d'abord suivre la formation *"LIN-001-FR - Introduction a Linux - en ligne de commande / shell"* ou equivalent.

Evaluation

En début de formation, nous vérifions l'expérience dans la domaine ainsi que les attentes de chaque participant.

Si en mode Qualiopi:

En début de formation, une auto évaluation est effectuée pour évaluer le niveau, ainsi que les attentes de chaque participant. Ainsi nous pouvons organiser au mieux les différents groupes de travail.

Le formateur effectue ensuite une évaluation continue à l'aide d'exercices pratiques.

À la fin de la formation, les participants auto-évaluent leur progression, et le formateur fournit une évaluation ainsi que des axes d'amélioration.

En option, un accompagnement post-formation peut être proposé, faisant l'objet d'un devis complémentaire.

Programme

A partir d'un exemple concret, nous regarderons les aspects suivants

MODULE: Introduction à Helm & Concepts de base:

- * Présentation de Helm : Qu'est-ce que c'est et pourquoi l'utiliser?
- * Comparaison de Helm Charts avec les alternatives (k8s yaml, kustomize)
- * Charts : Structure et composition
- * Helm Repositories : ArtifactHub, Chart Museum, Repositories
- * Comment installer une application avec Helm

EXERCISE: Installation:

- * Création de cluster Kubernetes via script simplifié
- * Installation de Helm
- * Recherche d'un Repo & d'un Chart via l'ArtifactHub et le CLI Helm
- * Installation du Chart sur un cluster Kubernetes

MODULE: Helm Commands et Workflow:

- * Structure d'un Chart
- * Commandes Helm : helm install, helm upgrade, helm rollback
- * Utilisation de valeurs pour personnaliser les installations
- * Mise à jour d'une application
- * Fonctions de template, variables, conditionnels

EXERCISE: Création d'un Chart simple:

- * Récupération d'un chart pour examiner sa structure
- * Création d'une structure de base de Chart.
- * Installation personnalisée d'un Chart avec l'option '--set'
- * Installation personnalisée d'un Chart avec '-f values.yaml'
- * Utilisation de l'option dry-run

MODULE: Helm Advanced:

- * Mise à jour d'une application
- * Quelques commandes helm bien utiles
- * Meilleures Pratiques de Helm
- * Utilisation des hooks pre-/post-operation
- * Tests de Charts
- * Emballage et dépendances

EXERCISE: Mise à jour et Rollback:

- * Gestion des releases avec Helm.
- * Mise à jour d'un Chart avec de nouvelles valeurs.
- * Exécution de rollbacks pour revenir à une version précédente.
- * Tests des Charts

MODULE: Questions et Réponses, Feedback:

- * Répondre aux questions des participants.
- * Recueillir des commentaires pour améliorer la formation.

Terraform

[2j] TF-000-FR Introduction a Terraform (CORE)

Description

Cette formation, proposée sous forme de formation de 2 jours, ou 4 demi-journées, présente aux étudiants les avantages de développer « Infrastructure as Code » avec Terraform.

L'intention est de suivre cette formation de 2 jours par une formation distincte d'une journée qui applique les concepts appris au fournisseur de cloud préféré du client (AWS, Azure, Google Cloud, Oracle) ou à un autre environnement virtualisé (Proxmox, OpenStack).

Note: Ces formations sont proposées sous forme de Terraform, mais peuvent également être dispensées sous forme de formations OpenTofu.

Cette formation initiale de 2 jours est la manière recommandée pour apprendre les bases de Terraform :

* Il utilise principalement Docker ou des Providers locaux pour les exercices de base - Cela permet d'assimiler les concepts Terraform importants à un rythme accéléré

Terraform permet de gérer le déploiement de différents types d'infrastructures via le plugin « Provider » approprié - ce cours utilise le fournisseur Docker pour pouvoir expérimenter dix fois plus rapidement qu'en utilisant des Cloud Providers ou même des Hyperviseurs.

Les étudiants apprécieront la facilité avec laquelle les ressources d'infrastructure sont définies de manière déclarative - en utilisant HCL v2 - « HashiCorp Configuration Language » - permettant de créer, mettre à jour ou détruire des ressources de manière idempotente.

Il est recommandé de combiner cette formation de 2 jours avec une ou plusieurs des formations d'une journée spécifiques au prestataire suivantes :

- * TF-001-FR Terraform Introduction (AWS)
- * TF-002-FR Terraform Introduction Terraform (Azure)
- * TF-003-FR Terraform Introduction Terraform (Google Cloud)
- * TF-004-FR Terraform Introduction Terraform (Oracle Cloud)
- * TF-005-FR Terraform Introduction Terraform (OpenStack)
- * TF-006-FR Terraform Introduction Terraform (Proxmox)

Pre-requisites

- * Soyez à l'aise en travaillant en ligne de commande et en éditant des fichiers
- * Notions de base de Cloud, Linux, Conteneurs
- * Utilisation d'un client ssh, par ex. openssh sous Linux, macOS ou Windows (ou Putty)

Included

- * Supports de cours et laboratoires : 50 % de travaux pratiques, 50 % de présentation et de démonstrations

- * Accès à un environnement de laboratoire temporaire
- * Accès à un document évolutif couvrant diverses ressources d'apprentissage Terraform

Objectives

- * Apprenez à utiliser Terraform pour mettre en place diverses ressources, de manière déclarative
- * Apprenez les nombreuses subtilités de Terraform/HCLv2 pour écrire des configurations déclaratives de qualité
 - Sachez où trouver des informations sur les autres fournisseurs, modules pour Google Cloud, Azure etc...

Programme

MODULE: Introduction to Infrastructure as Code:

- * HashiCorp “free to use” ecosystem
- * Infrastructure as Code, Config Management, Idempotence
- * Terraform
- * Installation

MODULE: Terraform Workflow:

- * The plan
- * Applying and re-applying plans
- * Destroying resources
- * Various sub-commands

MODULE: HCL Configurations:

- * Providers
- * Variables
- * Resources

MODULE: Variable types:

- * Variables, passing values to the configuration, Locals
- * Basic and complex types
- * Functions

MODULE: Control Structures:

- * Count, ternary, for_in, for_each
- * Dynamic Blocks
- * Templates

MODULE: Terraform Registry:

- * Provider Data Sources
- * Modules
- * Using Modules
- * Writing your own Modules

MODULE: In Practice:

- * Variable validation
- * Debugging
- * 3rd-party tools

MODULE: State:

- * Local State
- * Using “remote state” for working in teams

MODULE: Importation of foreign ressources:

- Importation of resources created outside Terraform:
- * terraform import
 - * Import blocks

MODULE: In Production:

- * Provisioners (Local-exec, File, Remote-exec)
- * Terraform Best practices
- * Tooling: Linters, scanners, testers
- * Terraform test

MODULE: HashiCorp Terraform Eco-system:

- * HCP Terraform & Enterprise
- * Terraform CDK
- * Waypoint, Boundary
- * Terraform Certification

[1j] TF-001-FR Terraform Introduction (AWS)

Description

Cette formation proposée en formation d'1 journée, ou 2 demi-journées, applique au Cloud AWS les principes de Terraform déjà appris dans la formation « TF-000-EN Terraform Introduction (CORE) » de 2 jours aux ressources.

Cette formation sera davantage orientée vers la pratique que vers la théorie : elle n'est pas destinée à enseigner AWS, mais à expérimenter l'application de Terraform aux ressources AWS.

Note: Suivre « TF-000-FR Terraform Introduction (CORE) » est un prérequis pour cette formation spécifique au cloud.

Terraform permet de gérer le déploiement de différents types d'infrastructures via le plugin « Provider » approprié - ce cours utilise le fournisseur AWS permettant de gérer de nombreuses ressources cloud AWS.

Les étudiants apprécieront la facilité avec laquelle les ressources d'infrastructure sont définies de manière déclarative permettant de créer, de mettre à jour ou de détruire des ressources de manière idempotente.

Terraform utilise HCL v2 – « HashiCorp Configuration Language » – pour définir les ressources à créer pour 1 ou plusieurs fournisseurs.

Note: le même cours est disponible pour d'autres fournisseurs

Pre-requisites

- * Soyez à l'aise en travaillant en ligne de commande et en éditant des fichiers
- * Notions de base de Cloud, Linux, Conteneurs
- * Utilisation d'un client ssh, par ex. openssh sous Linux, macOS ou Windows (ou Putty)

Compris

- * Supports de cours et laboratoires : 50 % de travaux pratiques, 50 % de présentation et de démonstrations
- * Accès à un environnement de laboratoire temporaire
- * Accès à un document évolutif couvrant diverses ressources d'apprentissage Terraform

Objectives

- * Apprenez à utiliser Terraform pour mettre en place diverses ressources AWS, de manière déclarative
- * Apprenez à utiliser les modules, sources de données et outils spécifiques à AWS

Note: les ressources spécifiques AWS suivantes sont proposées, mais peuvent être adaptées aux besoins des clients.

Programme

MODULE: Review

- * Infrastructure as Code principes
- * Terraform & OpenTofu workflows

MODULE: Working with Containers

- * Managing AWS ECS containers with Terraform
- * Using Data Sources with AWS ECS

MODULE: Working with VMs

- * Managing AWS EC2 virtual machines with Terraform
- * Using Data Sources with AWS EC2

MODULE: Working with modules

- * Terraform registry: Working with existing terraform modules for AWS
- * Writing modules for AWS: Creating clusters of VMs

MODULE: Templates

- * Creating useful templates (ssh_config, ansible inventory, reports)

MODULE: In Practice

- * Variable validation
- * Debugging
- * 3rd-party tools

MODULE: State

- * Local State
- * Using AWS/S3+DynamoDB for “remote state”

MODULE: Importation of foreign ressources

- * Importation of AWS resources
- * Move of AWS resources

MODULE: Auto-scaling & Load-Balancing

- * AWS EC2 ASG - Autoscaling Groups
- * AWS EC2 ALB - Application Load Balancer

MODULE: Other AWS resources

- * Lambda, VPC, EIP, S3, EBS, IAM, RDS

MODULE: In Production

- * Provisioners (Local-exec, File, Remote-exec)
- * Provider Aliases

[1j] TF-002-FR Terraform Introduction (Azure)

Description

Cette formation proposée en formation d'1 journée, ou 2 demi-journées, applique au Cloud Azure les principes de Terraform déjà appris dans la formation « TF-000-EN Terraform Introduction (CORE) » de 2 jours aux ressources.

Cette formation sera davantage orientée vers la pratique que vers la théorie : elle n'est pas destinée à enseigner Azure, mais à expérimenter l'application de Terraform aux ressources AWS.

Note: Suivre « TF-000-FR Terraform Introduction (CORE) » est un prérequis pour cette formation spécifique au cloud.

Terraform permet de gérer le déploiement de différents types d'infrastructures via le plugin « Provider » approprié - ce cours utilise le fournisseur Azure permettant de gérer de nombreuses ressources cloud Azure.

Les étudiants apprécieront la facilité avec laquelle les ressources d'infrastructure sont définies de manière déclarative permettant de créer, de mettre à jour ou de détruire des ressources de manière idempotente.

Terraform utilise HCL v2 – « HashiCorp Configuration Language » – pour définir les ressources à créer pour 1 ou plusieurs fournisseurs.

Note: le même cours est disponible pour d'autres fournisseurs

Pre-requisites

- * Soyez à l'aise en travaillant en ligne de commande et en éditant des fichiers
- * Notions de base de Cloud, Linux, Conteneurs
- * Utilisation d'un client ssh, par ex. openssh sous Linux, macOS ou Windows (ou Putty)

Compris

- * Supports de cours et laboratoires : 50 % de travaux pratiques, 50 % de présentation et de démonstrations
- * Accès à un environnement de laboratoire temporaire
- * Accès à un document évolutif couvrant diverses ressources d'apprentissage Terraform

Objectives

- * Apprenez à utiliser Terraform pour mettre en place diverses ressources Azure, de manière déclarative
- * Apprenez à utiliser les modules, sources de données et outils spécifiques à Azure

Note: les ressources spécifiques Azure suivantes sont proposées, mais peuvent être adaptées aux besoins des clients.

Programme

MODULE: Review

- * Infrastructure as Code principes
- * Terraform & OpenTofu workflows

MODULE: Working with Containers

- * Managing Azure ACI containers with Terraform
- * Using Data Sources with Azure ACI

MODULE: Working with VMs

- * Managing Azure virtual machines with Terraform
- * Using Data Sources with Azure VMs

MODULE: Working with modules

- * Terraform registry:: Working with existing terraform modules for Azure
- * Writing modules for Azure: Creating clusters of VMs

MODULE: Templates

- * Creating useful templates (ssh_config, ansible inventory, reports)

MODULE: In Practice

- * Variable validation
- * Debugging
- * 3rd-party tools

MODULE: State

- * Local State
- * Using Azure Blob Storage for “remote state”

MODULE: Importation of foreign resources

- * Importation of Azure resources
- * Move of Azure resources

MODULE: Auto-scaling & Load-Balancing

- * Azure VM ScaleSets
- * terraform-azurerm-loadbalancer

MODULE: Other Azure resources

- * Azure functions, Azure VPC, Azure Public IP, Blob Storage, AKS

MODULE: In Production

- * Provisioners (Local-exec, File, Remote-exec)
- * Provider Aliases

Ansible

[2j] AN-002-FR Introduction to Ansible

Description

Cette formation proposée sous forme de formation de 2 jours, ou 4 demi-journées, offre une introduction complète à la gestion de configuration sous Ansible.

****Note:**** Cette formation existe également sous forme de formation de 3 jours « AN-003-EN Introduction à Ansible » permettant plus de temps pour les TP et également l'ajout de modules “*Ansible Vault*” et “*Network Automation*”.

Cette formation comprend beaucoup de pratique ainsi que de théorie.
Tous les exercices pratiques seront effectués sur des machines virtuelles et des conteneurs Linux.

Les étudiants apprécieront la facilité avec laquelle les logiciels peuvent être installés et maintenus de manière fiable, reproductible et réutilisable au sein des équipes grâce à la nature largement déclarative des Playbooks et rôles Ansible. Les playbooks peuvent être réexécutés, garantissant des résultats reproductibles en raison de la nature idempotente d'Ansible.

Pre-requisites

- * Soyez à l'aise en travaillant en ligne de commande et en éditant des fichiers
- * Notions de base de Cloud, Linux, Conteneurs
- * Utilisation d'un client ssh, par ex. openssh sous Linux, macOS ou Windows (ou Putty)

Compris

- * Supports de cours et laboratoires : 50 % de travaux pratiques, 50 % de présentation et de démonstrations
- * Accès à un environnement de laboratoire temporaire
- * accès à un document évolutif regroupant diverses ressources d'apprentissage

Objectifs

- * Apprenez à utiliser Ansible pour effectuer la configuration logicielle et la maintenance des machines/appareils
- * Découvrez les nombreux modules intégrés, collections de modules et rôles pour Ansible
- * Découvrez les meilleures pratiques pour travailler avec Ansible

Programme

MODULE: Introduction

- * Pourquoi Ansible?
- * Concepts & Termes
- * Agentless Architecture
- * Inventory:
 - * Static & Dynamique
 - * Host & Group Patterns

MODULE: Deploying with Ansible

- * Installation de Ansible
- * Fichiers de configuration YAML
- * Playbooks
 - * Modules Ansible
 - * Commandes Ad-Hoc

- * Inventory Dynamique:
 - * Scripted
 - * Plugins

EXERCISE

- * Deploiement Ansible
- * Commandes Ad-Hoc
- * Inventories Dynamiques

MODULE: Playbooks

- * Structure de Playbook
- * Ordre d'exécution - Host & Task
- * Categories de Modules
 - * Command
 - * File Manipulation
 - * Network Modules
 - * Packaging Modules
 - * System Storage
 - * Account Management
 - * Security
 - * Services

EXERCISE

- * Playbook Basiques
- * Playbooks: Command Modules
- * Playbooks: Common Modules

MODULE: Variables

- * au runtime
- * dans Playbooks
- * register task output, debug module
- * dans inventory: host_vars, global_vars
- * Scope & precedence
- * Facts & magic variables
- * dans included files, roles
- * Output: callback plugins

EXERCISE

- * Variables & Facts
- * Inclusions

MODULE: Jinja Templating

- * Jinja Expressions
- * Builtin.template module
- * Filtres
- * Methodes
- * Conditionels
- * Lookup plugins
- * Structures de Control

EXERCISE

- * Templates Jinja

MODULE: Control Structures

- * Loops & Variables (with)
- * Conditionels (when)
- * Handlers
- * Tags
- * Error Handling

EXERCISE

- * Task Control

MODULE: Roles

- * Usage
- * Creation de Roles
- * Deploiement de Roles avec Ansible Galaxy

EXERCISE

- * Conversion Playbooks a Roles
- * Creation de Roles from Scratch
- * Ansible Galaxy Roles

[3j] AN-003-FR Introduction to Ansible

Description

Cette formation proposée sous forme de formation de 3 jours, ou 6 demi-journées, offre une introduction complète à la gestion de configuration sous Ansible.

****Note:**** Cette formation existe également sous forme de formation de 2 jours « AN-002-EN Introduction à Ansible » excluant les modules “*Ansible Vault*” et “*Network Automation*”.

Cette formation comprend beaucoup de pratique ainsi que de théorie.
Tous les exercices pratiques seront effectués sur des machines virtuelles et des conteneurs Linux.

Les étudiants apprécieront la facilité avec laquelle les logiciels peuvent être installés et maintenus de manière fiable, reproductible et réutilisable au sein des équipes grâce à la nature largement déclarative des Playbooks et rôles Ansible. Les playbooks peuvent être réexécutés, garantissant des résultats reproductibles en raison de la nature idempotente d'Ansible.

Pre-requisites

- * Soyez à l'aise en travaillant en ligne de commande et en éditant des fichiers
- * Notions de base de Cloud, Linux, Conteneurs
- * Utilisation d'un client ssh, par ex. openssh sous Linux, macOS ou Windows (ou Putty)

Compris

- * Supports de cours et laboratoires : 50 % de travaux pratiques, 50 % de présentation et de démonstrations
- * Accès à un environnement de laboratoire temporaire
- * accès à un document évolutif regroupant diverses ressources d'apprentissage

Objectifs

- * Apprenez à utiliser Ansible pour effectuer la configuration logicielle et la maintenance des machines/appareils
- * Découvrez les nombreux modules intégrés, collections de modules et rôles pour Ansible
- * Découvrez les meilleures pratiques pour travailler avec Ansible

Programme

MODULE: Introduction

- * Pourquoi Ansible?

- * Concepts & Termes
- * Agentless Architecture
- * Inventory:
 - * Static & Dynamique
 - * Host & Group Patterns

MODULE: Deploying with Ansible

- * Installation de Ansible
- * Fichiers de configuration YAML
- * Playbooks
 - * Modules Ansible
 - * Commandes Ad-Hoc
 - * Inventory Dynamique:
 - * Scripted
 - * Plugins

EXERCISE

- * Deploiement Ansible
- * Commandes Ad-Hoc
- * Inventories Dynamiques

MODULE: Playbooks

- * Structure de Playbook
- * Ordre d'exécution - Host & Task
- * Categories de Modules
 - * Command
 - * File Manipulation
 - * Network Modules
 - * Packaging Modules
 - * System Storage
 - * Account Management
 - * Security
 - * Services

EXERCISE

- * Playbook Basiques
- * Playbooks: Command Modules
- * Playbooks: Common Modules

MODULE: Variables

- * au runtime
- * dans Playbooks
- * register task output, debug module
- * dans inventory: host_vars, global_vars
- * Scope & precedence
- * Facts & magic variables
- * dans included files, roles
- * Output: callback plugins

EXERCISE

- * Variables & Facts
- * Inclusions

MODULE: Jinja Templating

- * Jinja Expressions
- * Builtin.template module
- * Filtres
- * Methodes
- * Conditionels

- * Lookup plugins
- * Structures de Control

EXERCISE

- * Templates Jinja

MODULE: Control Structures

- * Loops & Variables (with)
- * Conditionals (when)
- * Handlers
- * Tags
- * Error Handling

EXERCISE

- * Task Control

MODULE: Roles

- * Usage
- * Creation de Roles
- * Deploiement de Roles avec Ansible Galaxy

EXERCISE

- * Conversion Playbooks a Roles
- * Creation de Roles from Scratch
- * Ansible Galaxy Roles

MODULE: Optimisation

- * Types de Connection
- * Delegation
- * Parallelisme
- * Callback Plugins

EXERCISE

- * Optimization d'Ansible

MODULE: Secrets with Ansible Vault

- * Configuration Vault
- * Vault IDs
- * Utilisation de Vault

EXERCISE

- * Ansible Vault

MODULE: Network Automation

- * Exemples Simple de Module Reseau
- * Debugage de Modules Reseaux
- * Modules ios
- * Exemple de Modules Simple IOS

WebAssembly

TBD: Introduction to WebAssembly

From ChatGPT:

Jour 1: Introduction à WebAssembly (8 heures)

MODULE: C'est quoi WebAssembly? (1 heure):

- Définition de WebAssembly (WASM)
- Historique du développement de WASM
- Enjeux liés à l'utilisation de WASM
- Avantages de WebAssembly

MODULE: Cloud Natif et WebAssembly (1 heure):

- Introduction au cloud natif
- Intégration de WebAssembly dans les architectures cloud natives
- Exemples d'entreprises utilisant WASM dans des environnements cloud

MODULE: Overview de la Technologie (2 heures):

- Gestion de la mémoire dans WebAssembly
- Isolation des exécutions
- Utilisations en navigateur
- Utilisations en ligne de commande
- Cas d'usage variés de WebAssembly

MODULE: Travaux Pratiques (2 heures):

- Mise en place de l'environnement de développement WASM
- Exécution de premiers programmes WASM
- Exploration des outils de développement et des compilateurs

Jour 2: Modules WASM, Réseau et Outils (8 heures)

MODULE: Les Modules WebAssembly (2 heures):

- Comprendre la structure des modules WASM
- Interaction avec les modules depuis d'autres langages
- Développement de modules WASM personnalisés

MODULE: Le Réseau et WebAssembly (2 heures):

- Utilisation de WebAssembly dans le contexte réseau
- Exemples de communication entre modules WASM
- Cas d'usage spécifiques liés au réseau

MODULE: Outils et Runtimes pour WebAssembly (2 heures):

- Exploration de wasmtime et d'autres runtimes
- Différents compilateurs pour WebAssembly
- Utilisation de WASI, WASIX, et WASI Preview2

MODULE: Travaux Pratiques (2 heures):

- Utilisation des outils et runtimes présentés
- Expérimentation avec les différentes options de compilation
- Manipulation de modules WASM dans des environnements de test

MODULE: Bonus: Avancées et Futur de WebAssembly (1 heure):

- Observabilité avec wasmedge et observesdk
- Utilisation de WebAssembly avec Docker, HashiCorp Nomad, Kubernetes, Fermyon Spin, Envoy et Open Policy Agent
- Perspectives sur l'avenir de WebAssembly

MODULE: Conclusion et Questions (1 heure):

- Récapitulation des points clés
- Réponses aux questions des participants
- Remise de ressources supplémentaires pour approfondir les connaissances