

Projet de pipelines CI/CD

Le module de **Pipelines et livraisons continues** consiste en un projet utilisant le code développé lors du cours **DevOps** afin de l'orchestrer dans des pipelines Jenkins.

1 - Installation de Jenkins:

Cette installation se fait manuellement, sur une instance EC2 de type t2.micro, en suivant la documentation officielle : <https://www.jenkins.io/doc/book/installing/linux/>

L'instance EC2 doit être publique afin que tous les membres du groupe puissent y accéder. Créez un compte nominatif pour chaque membre du groupe.

2 - Versionning du code:

Les TPs du cours Devops nous ont permis d'écrire du code:

- **Ansible**: Pour automatiser l'installation d'apache httpd, choisir un port d'écoute et installer un site web statique.
- **Packer**: Pour la création de l'AMI contenant le site web statique.
- **Terraform**: Pour automatiser la création de tout le réseau AWS + l'installation du site web statique sur un Auto scaling groupe et éventuellement son exposition via un Load Balancer.

Tout le code développé doit être déposé sur un dépôt **git** afin que les pipelines Jenkins puissent le cloner et l'exécuter.

3 - Travail minimal attendu:

3.1 - AMI:

Cette pipeline permet de construire une AMI contenant un serveur apache httpd et un site web.

Cette pipeline prend en entrée:

- L'environnement (dev, preprod, prod,...): Cette valeur doit être visible, soit dans des tags ou dans le nom de l'AMI.
- Le port sur lequel écoute le serveur httpd et qui nous permettra d'accéder au site web: Cette valeur doit être visible, soit dans des tags ou dans le nom de l'AMI.
- Le nom du projet: il doit être utilisé pour construire le nom final de l'AMI.

3.2 - Infrastructure Réseau:

Cette pipeline permet de créer tous les composants réseau (un nouveau vpc, des subnet, ...) qui vont accueillir le site web.

Elle devra prendre en paramètres: ***l'environnement*** et le **nom du projet**: ces deux paramètres doivent apparaître dans les noms et les tags de toutes les ressources AWS créés

3.3 - Le site Web

Cette Pipeline permet de déployer le site web en exploitant les ressources créées par les deux pipelines plus haut

3.4 - Destruction:

Pour chaque pipeline plus haut, créez une pipeline de nettoyage permettant de détruire les ressources.

4 - Axes d'amélioration:

- Les pipelines créées sont nombreuses, et doivent être lancées manuellement dans un ordre spécifique (AMI puis Réseau puis déploiement du site). Comment faire pour simplifier leur lancement ?
- Vous avez probablement créé les paramètres d'entrée de vos pipelines via l'UI de jenkins. Comment faire pour les intégrer directement dans le code de la pipeline?
- Vous avez développé probablement votre pipeline dans l'UI jenkins. Mais finalement, la pipeline n'est que du code, et ce code évoluera forcément avec le temps. Comment faire pour 'versionner' ce code et simplifier la création de la pipeline ?
- Vous avez construit des pipelines de **Livraison continue**, comment faire pour les transformer en pipelines de **Déploiement continu**?
- Souvent, quand vous travaillez sur des projets conséquents, l'instance de Jenkins est installée sur un compte AWS, et les environnements sur lesquels doivent agir les pipelines sont dans d'autres comptes AWS. Pourriez-vous faire en sorte que vos pipelines créent et installent votre site web sur le compte AWS de l'un de vos collègues ?