**Projet Kubernetes**

Guide d’utilisation

Cours :

DevSecOps – ESILV

Participants :

Reizène Ben Hassine

Maureen Drouard

Date :

12/12/2024

Sommaire

[Présentation 3](#_Toc184840241)

[1. Configuration des ConfigsMaps et Secrets 4](#_Toc184840242)

[a) Fichier Secret 4](#_Toc184840243)

[b) Fichier Configmap 4](#_Toc184840244)

[2. Gestion des logs 4](#_Toc184840245)

[3. Monitoring 5](#_Toc184840246)

## Présentation

Cette documentation consiste à expliquer comment utiliser Kubernetes et Docker pour lancer notre application web en Node.js et MySQL de gestion des utilisateurs.

**Objectifs :**

* Déployer une BDD comme un pod Kubernetes avec persistance des données
* Déployer une app web conteneurisée en utilisant Docker
* Créer des services pour la communication interne et externe entre les composants de l’application

**Outils utilisés :**

* Docker : pour les conteneurs
* Minikube : pour créer et gérer les cluster Kubernetes
* Kubectl : pour interagir avec les clusters
* VS Code : pour éditer les fichiers de configurations

## Configuration des ConfigsMaps et Secrets

Les Configmaps et secrets sont les yaml permettant de stocker les données de configuration et les informations sensibles de l’application web.

### Fichier Secret

Dans le fichier Secret,yaml, modifier les valeurs des variables :

* DB\_HOST: adresse de la bdd
* DB\_USER: nom d’utilisateur
* DB\_PASSWORD: mots de passes
* DB\_NAME: nom de la base de données

Modifier les valeurs en base 64 directement dans le fichier yaml et faites la commande « kubectl apply -f secret.yaml »

### Fichier Configmap

Dans le fichier mysql-init-configmap.yaml, modifier le nom de la base de données si besoin :

Une image contenant texte, capture d’écran, Police, affichage

Description générée automatiquement

Ensuite faites « kubectl apply -f mysql-init-configmap.yaml » et  « kubectl delete pod -l app=mysql ».

## Gestion des logs

Un cas de problème, il est possible avec kubernetes de superviser et de diagnostiquer les problèmes.

Pour inspecter les logs :

* kubectl logs -l app=web
* kubectl logs -l app=mysql

Pour surveiller l’état des ressources :

* kubectl get all
* kubectl describe pod <pod-name>

Pour debugguer pour un pod :

* kubectl exec -it <pod-name> -- bash

## Monitoring

Avec Kubernetes, vous avez accès à un dashboard grâce à la commande Minikube dashboard. Ici, vous trouverez toutes vos ressources, leur fonctionnement,