

LAB 5 : Manipulation des objets Deployments

Conditions préalables

- Cluster Kubernetes fonctionnel

Exercice 1 : Deployment.

Dans cet exercice, vous allez créer un Deployment et l'exposer à l'extérieur du cluster via un service de type NodePort.

1. Spécification d'un Deployment

Créez un fichier `vote_deployment.yaml` définissant un Deployment ayant les propriétés suivantes:

- nom: vote
- nombre de replicas: 3
- définition d'un selector sur le label `app:vote`
- spécification du Pod:
 - label `app:vote`
 - un container nommé `vote` basé sur l'image `instavote/vote` et exposant le port 80

2. Création du Deployment

Utilisez `kubectl` pour créer le Deployment

3. Status du Deployment

A l'aide de `kubectl`, examinez le status du Deployment `vote`.

A partir de ces informations, que pouvez-vous dire par rapport au nombre de Pods gérés par ce Deployment ?

4. Status des Pods associés

A l'aide de `kubectl`, lister les Pods associés à ce Deployment.

5. Exposition des Pods du Deployment

Créez un Service permettant d'exposer les Pods du Deployment à l'extérieur du cluster

Conseils:

- Vous pourrez commencer par créer une spécification pour le Service, en spécifiant que le selector doit permettre de regrouper les Pods ayant le label `app:vote`.
- Utilisez un service de type `NodePort`, vous pourrez par exemple le publier sur le port 31001 des nodes du cluster
- le container basé sur l'image `instavote/vote` tourne sur le port 80, ce port devra donc être référencé en tant que `targetPort` dans la spécification du Service.

Note: n'hésitez pas à vous reporter à l'exercice sur les Services de type NodePort que nous avons vu précédemment

Une fois le service créé, vous pourrez accéder à l'interface de vote sur `http://IP:31001` où IP est l'adresse IP d'une machine du cluster Kubernetes.

Attention: cette interface n'est pas branchée à un backend, il n'est pas encore possible de voter, si vous cliquez sur l'un des choix, vous obtiendrez une erreur.

Exercice 2 : rolling update.

Dans cet exercice, vous allez créer un Deployment et effectuer un rolling update.

1. Création d'un Deployment

A l'aide de la commande `kubectl run`, créez un Deployment avec les propriétés suivantes:

- nom: `www`
- 3 replicas d'un Pod avec un container basé sur `nginx:1.12`

Spécifiez l'option `--record=true` à la fin de la commande afin de conserver l'historique des commandes de mises à jour du Deployment.

2. Liste des ressources

- Listez les ressources créées par la commande précédente (Deployment, ReplicaSet, Pod).

Note: utilisez une seule fois la commande `kubectl` pour lister l'ensemble des ressources.

3. Mise à jour de l'image

- Mettez l'image `nginx` à jour avec la version `1.14-alpine`

4. Liste des ressources

Une nouvelle fois, listez les ressources.

- Que constatez-vous ?

5. Historique des mises à jour

- Listez les mises à jour (= révisions) du Deployment.

Note: utilisez la commande `kubectl rollout...`