

Exercice 7 : Mise à jour Deployment

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 create`, créez un Deployment

- nommé `www`
- définissant un Pod basé sur l'image `nginx:1.16`

Note: la commande `kubectl create` fait partie des commandes dites impératives, elle permet de créer un Deployment sans passer par un fichier de spécification en yaml. Cette approche permet d'aller vite, elle est à utiliser dans un contexte de développement ou de debugging.

2. Scaling

La commande `kubectl create` ne permet pas de spécifier le nombre de réplicas à la création d'un Deployment. Par défaut le Deployment créé à l'étape précédente a donc un seul réplica.

Changez ce nombre de replicas de façon à en avoir 3.

Note: pour cela vous pourrez avoir besoin de la commande `$ kubectl scale` L'aide en ligne `$ kubectl scale --help` donne quelques exemples d'utilisation.

3. Liste des ressources

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

4. Mise à jour de l'image

Mettez l'image `nginx` à jour avec la version `nginx:1.16-alpine`

Note: spécifiez l'option `--record` afin de conserver l'historique de la mise à jour

5. Liste des ressources

Une nouvelle fois, listez les ressources.

Que constatez vous ?

6. Historique des mises à jour

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

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

7. Effectuez un rollback

Faites un rollback et vérifiez que le Deployment est maintenant basé sur la version précédente de l'image (*nginx:1.16*)

8. Cleanup

Supprimez le Deployment *www*