

Lab 4 - Communiquer avec les Pods via les Services ClusterIP et NodePort

COMPTE RENDU

YOUSSEF BEN GHORBEL

Table des matières

Objectifs	2
Pré-requis	2
Étapes à suivre	2
1. Créer un Namespace.....	2
2. Créer un Deployment	2
3. Vérifier le contenu des Pods.....	3
4. Création du Service ClusterIP.....	4
5. Tester le Service ClusterIP	4
6. Créer un Service de type NodePort	4
7. Configurer le cluster pour exposer le NodePort	5
8. Tester le Service NodePort.....	5

Objectifs

- Écrire une configuration Service de type ClusterIP.
- Réaliser une communication entre deux Pods.
- Écrire une configuration Service de type NodePort.
- Réaliser une communication depuis l'extérieur du cluster K8s.

Pré-requis

- Assurez-vous que le Namespace mynamespaceexercice1 soit supprimé

```
kubectl delete namespace mynamespaceexercice2
```

Étapes à suivre

1. Créer un Namespace

- Créez le fichier mynamespaceexercice3.yaml

```
apiVersion: v1
kind: Namespace
metadata:
  name: mynamespaceexercice3
```

- Appliquer le fichier :

```
kubectl apply -f exercice2-deployment/mynamespaceexercice3.yaml
```

```
root@localhost exercice3-service-clust
namespace/mynamespaceexercice3 created
root@localhost exercice3-service-clust
```

2. Créer un Deployment

- Créez le fichier mydeploymentforservice.yaml

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: mydeploymentforservice
spec:
  replicas: 3
  selector:
    matchLabels:
```

```

    app: mypod
template:
  metadata:
    labels:
      app: mypod
  spec:
    containers:
    - name: mycontainer
      image: nginx:latest
      ports:
      - containerPort: 80
      lifecycle:
        postStart:
          exec:
            command: ["/bin/sh", "-c", "echo $HOSTNAME >
/usr/share/nginx/html/index.html"]

```

- Appliquez cette configuration

```

kubectl apply -f exercice3-service-clusterip-
nodeport/mydeploymentforservice.yaml -n mynamespaceexercice3

```

```

[root@localhost exercice3-service-clusterip-nodeport]# kubectl apply -f
exercice3-service-clusterip-nodeport/mydeploymentforservice.yaml -n mynamespaceexercice3
deployment.apps/mydeploymentforservice created
[root@localhost exercice3-service-clusterip-nodeport]#

```

3. Vérifier le contenu des Pods

- Lister les Pods et vérifier le contenu de index.html :

```

kubectl get pod -n mynamespaceexercice3

```

```

[root@localhost exercice3-service-clusterip-nodeport]# kubectl get pod -n mynamespaceexercice3
NAME                                READY   STATUS    RESTARTS   AGE
mydeploymentforservice-548c9d97dc-7rflh  1/1     Running   0           4m53s
mydeploymentforservice-548c9d97dc-zfszw  1/1     Running   0           4m53s
mydeploymentforservice-548c9d97dc-zhrwr  1/1     Running   0           4m53s
[root@localhost exercice3-service-clusterip-nodeport]#

```

```

kubectl exec -it -n mynamespaceexercice3 mydeploymentforservice-
548c9d97dc-7rflh -- more /usr/share/nginx/html/index.html

```

```

mydeploymentforservice-548c9d97dc-7rflh -- more /usr/share/nginx/html/index.html
mydeploymentforservice-548c9d97dc-7rflh

```

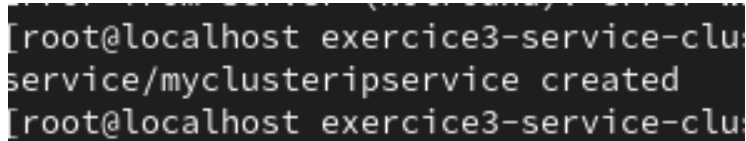
4. Création du Service ClusterIP

- Créer le fichier myclusteripservice.yaml :

```
apiVersion: v1
kind: Service
metadata:
  name: myclusteripservice
spec:
  selector:
    app: mypod
  type: ClusterIP
  ports:
  - protocol: TCP
    targetPort: 80
    port: 8080
```

- Appliquer le fichier :

```
kubectl apply -f exercice3-service-clusterip-
nodeport/myclusteripservice.yaml -n mynamespaceexercice3cd
```



```
[root@localhost exercice3-service-clu
service/myclusteripservice created
[root@localhost exercice3-service-clu
```

5. Tester le Service ClusterIP

- Créer un Pod de test et interroger le Service :

```
kubectl run podtest --image=alpine:latest -- /bin/sh -c "while
true; do wget -qO- 10.43.77.141:8080; sleep 1; done"
```

6. Créer un Service de type NodePort

- Créer un fichier mynodeportservice.yaml :

```
apiVersion: v1
kind: Service
metadata:
  name: mynodeportservice
spec:
```

```

selector:
  app: mypod
type: NodePort
ports:
- protocol: TCP
  targetPort: 80
  port: 8080
  nodePort: 30001

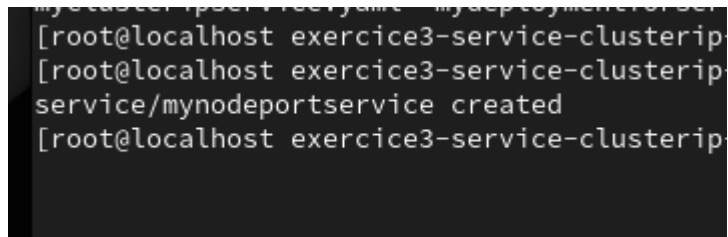
```

- Appliquer le fichier :

```

kubectl apply -f exercice3-service-clusterip-
nodeport/mynodeportservice.yaml -n mynamespaceexercice3

```



```

[myclusteripserver]$ kubectl apply -f mynodeportservice.yaml
[root@localhost exercice3-service-clusterip-
nodeport]$ kubectl apply -f mynodeportservice.yaml -n mynamespaceexercice3
service/mynodeportservice created
[root@localhost exercice3-service-clusterip-
nodeport]$

```

7. Configurer le cluster pour exposer le NodePort

- Modifier le cluster K3d pour ajouter l'écoute sur le port 30001 :

```

k3d cluster edit mycluster --port-add 30001:30001@server:0

```

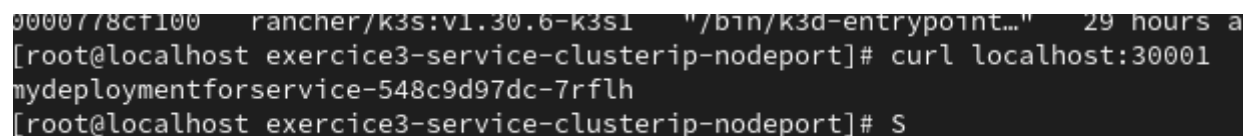
8. Tester le Service NodePort

- Interroger le Service NodePort avec cURL :

```

curl localhost:30001

```



```

0000778cf100 rancher/k3s:v1.30.6-k3s1 "/bin/k3d-entrypoint..." 29 hours a
[root@localhost exercice3-service-clusterip-nodeport]# curl localhost:30001
mydeploymentforservice-548c9d97dc-7rflh
[root@localhost exercice3-service-clusterip-nodeport]# S

```