



Léo Bouffard
Cyprien Touraine

Montage cluster Kubernetes avec Kind

1. Ouvrir Docker Desktop afin d'activer Docker
2. Créez un dossier à la racine C: de votre ordinateur nommé tools
3. Récupérez kind.exe en exécutant : `curl -Lo %USERPROFILE%\Downloads\kind.exe https://kind.sigs.k8s.io/dl/latest/kind-windows-amd64`
4. changez l'emplacement du fichier kind.exe vers votre dossier créé : `move %USERPROFILE%\Downloads\kind.exe C:\tools`
5. Ajoutez ce dossier au Path:
 - a. Cliquez droit sur *Ce PC* > *Propriétés* > *Paramètres système avancés* > *Variables d'environnement*.
 - b. Modifiez la variable Path et ajoutez le chemin du dossier contenant kind.exe
6. Allez dans le dossier du projet
7. Ouvrez un terminal dans ce dernier
8. Exécutez la requête suivante : `kind create cluster --config kind-config.yaml`
9. Le résultat suivant devrez s'afficher :

```
PS C:\Users\Utilisateur\Desktop\kubernetes\4KUBE-PROJECT> kind create cluster --config kind-config.yaml
Creating cluster "kind" ...
  • Ensuring node image (kindest/node:v1.31.2) ...
  ✓ Ensuring node image (kindest/node:v1.31.2) ...
  • Preparing nodes ...
  ✓ Preparing nodes ...
  • Writing configuration ...
  ✓ Writing configuration ...
  • Starting control-plane ...
  ✓ Starting control-plane ...
  • Installing CNI ...
  ✓ Installing CNI ...
  • Installing StorageClass ...
  ✓ Installing StorageClass ...
  • Joining worker nodes ...
  ✓ Joining worker nodes ...
Set kubect context to "kind-kind"
You can now use your cluster with:

kubectl cluster-info --context kind-kind

Thanks for using kind! 😊
PS C:\Users\Utilisateur\Desktop\kubernetes\4KUBE-PROJECT> kubectl get nodes
NAME                STATUS    ROLES    AGE     VERSION
kind-control-plane  Ready    control-plane  2m8s   v1.31.2
kind-worker         Ready    <none>      116s   v1.31.2
kind-worker2        Ready    <none>      115s   v1.31.2
PS C:\Users\Utilisateur\Desktop\kubernetes\4KUBE-PROJECT>
```

10. Il faut maintenant attribuer le label worker aux deux nœuds workers voulus, afin que les déploiements se passent sans problème, modifier les noms des nodes si différents :
 - a. Exécutez : `kubectl label nodes kind-worker kubernetes.io/role=worker`
 - b. Exécutez : `kubectl label nodes kind-worker2 kubernetes.io/role=worker`
11. Si les noms des workers ne sont pas les mêmes, adaptez la ligne de commande
12. Vous pouvez maintenant déployer vos yaml !
13. Pour avoir accès à l'application depuis localhost:8080, exécutez : `kubectl port-forward service/fleetman-webapp 8080:80`

Test crash d'un nœud :

1. Après avoir déployé vos différents yaml
2. Arrêtez l'un des deux pods en exécutant par exemple : `docker stop kind-worker2`

3. Observez qu'en rechargeant la page, l'application marche encore

Déploiement et suppression des ressources :

Pour déployer les ressources, exécutez : `deploy.bat`

Pour supprimer les ressources, exécutez : `deploy.bat`

Ces deux fichiers contiennent les lignes de commandes pour réaliser les différentes actions

Information importante :

Les véhicules peuvent mettre un certain temps avant d'apparaître, mais ils finissent par arriver.