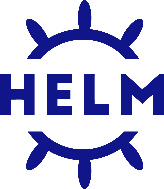
**Simulation de l'environnement de production localement :**



1. **Technologies utilisées :**

****



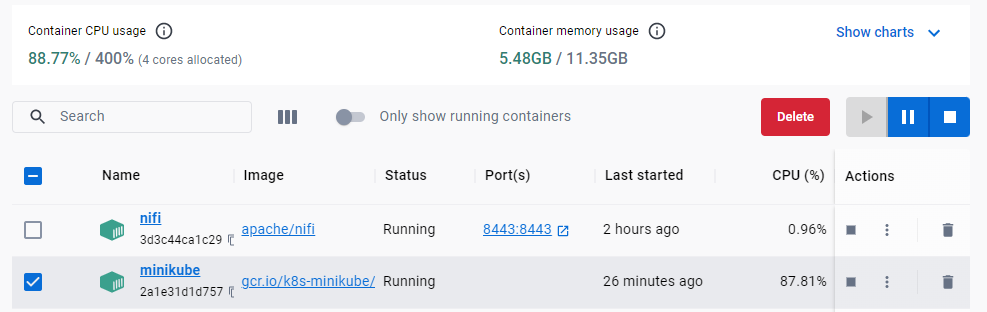
**NiFiKop**

1. **Installation de Minikube, Kubectl et Helm :**

Minikube nous a permis de créer une version miniature d'un cluster Kubernetes au sein de notre propre machine, fournissant une plateforme de test fiable. Cela nous a permis d'expérimenter, de développer et de déployer des applications dans un environnement de type Kubernetes, tout en évitant la complexité d'une infrastructure complète. Kubectl, d'autre part, s'est révélé indispensable pour interagir avec le cluster Kubernetes, nous permettant de déployer et de gérer des ressources, de surveiller l'état des pods et de contrôler divers aspects du cluster. Enfin, l'utilisation de Helm a grandement facilité la gestion des déploiements d'applications en automatisant la configuration, la gestion des dépendances et la mise à l'échelle des applications.

**a) Commandes**

**b) Aperçu**



1. **Configuration du cluster avec Nifikop et Zookeeper :**

Nifikop, un opérateur Kubernetes spécialement conçu pour Apache NiFi, a joué un rôle essentiel dans la simplification du déploiement et de la gestion des instances NiFi au sein du cluster. En utilisant le graphique Helm de Nifikop, nous avons pu configurer et allouer de manière transparente des ressources, garantissant des performances optimales tout en respectant les contraintes de ressources. Cela a accéléré le processus de configuration de NiFi au sein du cluster, facilitant ainsi les phases de test et de développement. Zookeeper, quant à lui, s'est révélé être un composant crucial pour soutenir les opérations de NiFi. Déployé à l'aide de Helm, Zookeeper a établi un service de coordination résilient et distribué, indispensable pour maintenir la fiabilité des systèmes distribués de NiFi.



1. **Création du cluster et des pods :**
2. **Commandes**

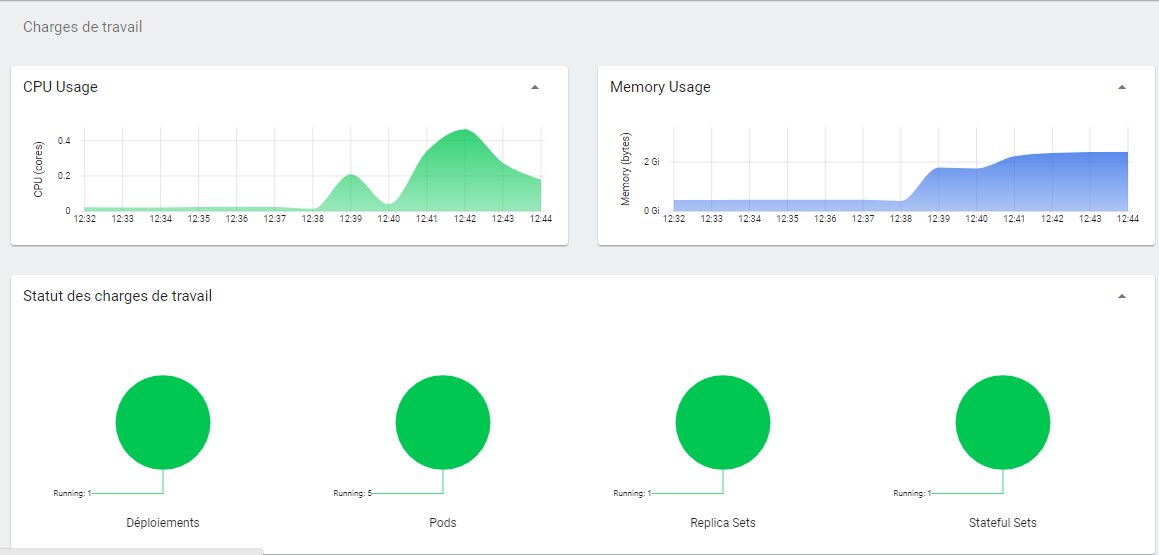
kubectl get pod -n nifi

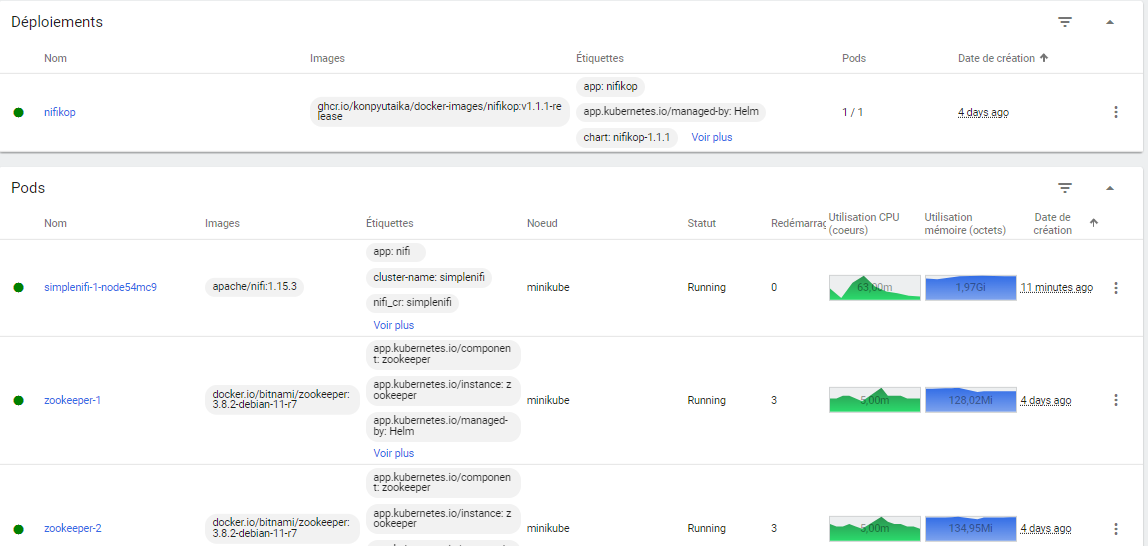
kubectl create -n nifi -f C:\persistentnificluster.yaml

1. **Aperçu du fichier de configuration**



1. **Aperçu global :**





1. **Accès** [**http://localhost:8080**](http://localhost:8080)**:**

