

LongHorn

Déployer une solution de stockage

Installation des prérequis sur TOUS LES NOEUDS :

```
sudo apt-get update
sudo apt-get install -y open-iscsi nfs-common
sudo systemctl enable --now iscsid
```

Installation de Helm sur le master 1 :

```
curl -fsSL -o get_helm.sh https://raw.githubusercontent.com/helm/helm/main/scripts/get-helm-3
bash get_helm.sh
```

Préparation et déploiement de Longhorn :

```
helm repo add longhorn https://charts.longhorn.io
helm repo update
```

Dans le fichier : /root/longhorn-values.yaml :

```
defaultSettings:
  defaultDataPath: "/opt/longhorn"
```

Déploiement de longhorn :

```
helm install longhorn longhorn/longhorn \
  --namespace longhorn-system \
  --create-namespace \
```

```
-f /root/longhorn-values.yaml
```

Résultat :

LAST DEPLOYED: Wed Apr 15 22:49:57 2026

NAMESPACE: longhorn-system

STATUS: deployed

REVISION: 1

TEST SUITE: None

NOTES:

Longhorn is now installed on the cluster!

Please wait a few minutes for other Longhorn components such as CSI deployments, Engine Images, and Instance Managers to be initialized.

Visit our documentation at <https://longhorn.io/docs/>

Vérifications :

```
kubectl get pods -n longhorn-system
```

NAME	READY	STATUS	RESTARTS	AGE
csi-attacher-5bcb65bf95-44fv7	1/1	Running	1 (66s ago)	2m27s
csi-attacher-5bcb65bf95-jvqxj	1/1	Running	0	2m27s
csi-attacher-5bcb65bf95-pgf8h	1/1	Running	0	2m27s
csi-provisioner-5d498c6944-d65sl	1/1	Running	0	2m27s
csi-provisioner-5d498c6944-dfl96	1/1	Running	0	2m27s
csi-provisioner-5d498c6944-wvmvs	1/1	Running	0	2m27s
csi-resizer-6c6474c996-9k94v	1/1	Running	0	2m26s
csi-resizer-6c6474c996-mjtsj	1/1	Running	0	2m27s
csi-resizer-6c6474c996-t94bs	1/1	Running	0	2m26s
csi-snapshotter-5cc5fb45d9-pxz5r	1/1	Running	0	2m26s
csi-snapshotter-5cc5fb45d9-swr6x	1/1	Running	0	2m26s
csi-snapshotter-5cc5fb45d9-wsdh6	1/1	Running	0	2m26s
engine-image-ei-75a03ec3-8r6p4	1/1	Running	0	3m23s
engine-image-ei-75a03ec3-nmbdp	1/1	Running	0	3m23s
engine-image-ei-75a03ec3-tl4x2	1/1	Running	0	3m23s

```

engine-image-ei-75a03ec3-vcq2q          1/1  Running  0          3m23s
instance-manager-8904e58b7c0d881b70d813ddd4fe86f4  1/1  Running  0          2m54s
instance-manager-9176dd8e42515fe6146f445c7a1bc3ad  1/1  Running  0          2m53s
instance-manager-9805597396c5db2820496e3f125bc6bc  1/1  Running  0          2m53s
instance-manager-eaeed3ca06747e29c0f3aa8b9557c5c8  1/1  Running  0          2m53s
longhorn-csi-plugin-7vtx8                3/3  Running  0          2m26s
longhorn-csi-plugin-brx6d                3/3  Running  0          2m26s
longhorn-csi-plugin-d4fxc                3/3  Running  0          2m26s
longhorn-csi-plugin-tsrrg                3/3  Running  0          2m26s
longhorn-driver-deployer-746f54969f-9gmlb        1/1  Running  0          4m5s
longhorn-manager-b7q7k                    2/2  Running  0          4m5s
longhorn-manager-fk6f4                    2/2  Running  0          4m5s
longhorn-manager-xghbf                    2/2  Running  0          4m5s
longhorn-manager-xt4xq                    2/2  Running  0          4m5s
longhorn-ui-5df99fc477-7zt5s             1/1  Running  0          4m5s
longhorn-ui-5df99fc477-tzwdp             1/1  Running  0          4m5s

```

Accéder à l'interface graphique :

```
kubectl patch svc longhorn-frontend -n longhorn-system -p '{"spec": {"type": "NodePort"}}'
```

Afficher le port :

```
kubectl get svc longhorn-frontend -n longhorn-system
```

On va utiliser le port :

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
longhorn-frontend	NodePort	10.102.85.80	<none>	80:32555/TCP	6m22s

Exemple ici : <http://192.168.1.104:32555/#/dashboard>

On peut utiliser n'importe quelle IP du cluster.

Expand All	Delete	Edit Node	Status	Readiness	Name	Replicas	Allocated	Used	Size	Tags	Operation
+	<input type="checkbox"/>		Schedulable	Ready	k8s-master-1 10.10.156.49	0	0 / 69.95 Gi	2.28 / 99.93 Gi	70 Gi +30 Gi Reserved		⋮
+	<input type="checkbox"/>		Schedulable	Ready	k8s-worker-1 10.10.230.8	0	0 / 69.95 Gi	2.28 / 99.93 Gi	70 Gi +30 Gi Reserved		⋮
+	<input type="checkbox"/>		Schedulable	Ready	k8s-worker-2 10.10.140.8	0	0 / 69.95 Gi	2.28 / 99.93 Gi	70 Gi +30 Gi Reserved		⋮
+	<input type="checkbox"/>		Schedulable	Ready	k8s-worker-3 10.10.69.201	0	0 / 69.95 Gi	2.28 / 99.93 Gi	70 Gi +30 Gi Reserved		⋮

Tester le stockage (pour aller plus loin) :

Déclaration du PVC :

hello-pvc.yaml

```

apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: longhorn-hello-pvc
spec:
  accessModes:
    - ReadWriteOnce
  storageClassName: longhorn # On utilise le driver Longhorn
resources:
  requests:
    storage: 1Gi # 1 Go c'est largement assez pour un test

```

Création d'un conteneur utilisant le PVC :

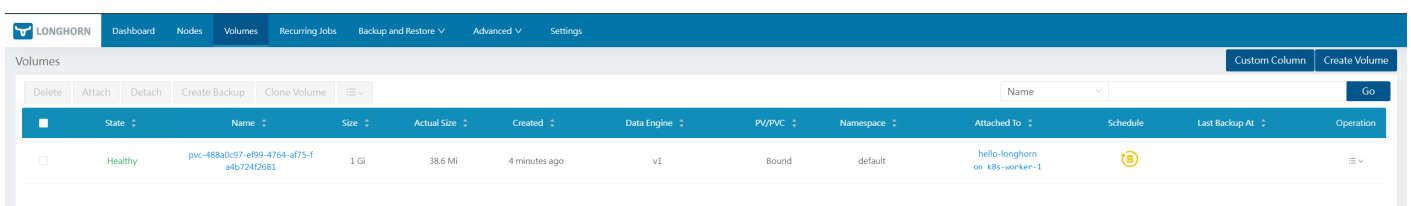
hello-app.yaml

```
apiVersion: v1
kind: Pod
metadata:
  name: hello-longhorn
spec:
  containers:
  - name: hello-world
    image: busybox
    # On écrit la date dans /data/coucou.txt en boucle
    command: ["sh", "-c", "while true; do date >> /data/coucou.txt; echo 'Donnée écrite !'; sleep 5; done"]
  volumeMounts:
  - name: storage-volume
    mountPath: /data
  volumes:
  - name: storage-volume
    persistentVolumeClaim:
      claimName: longhorn-hello-pvc
```

Application des changements :

```
kubectl apply -f hello-pvc.yaml
kubectl apply -f hello-app.yaml
```

Côté interface graphique :



The screenshot shows the Longhorn web interface. The top navigation bar includes 'Dashboard', 'Nodes', 'Volumes', 'Recurring Jobs', 'Backup and Restore', 'Advanced', and 'Settings'. The 'Volumes' section is active, displaying a table of volumes. The table has columns for State, Name, Size, Actual Size, Created, Data Engine, PV/PVC, Namespace, Attached To, Schedule, Last Backup At, and Operation. A single volume is listed with the name 'pvc-488a0c37-e99-4764-a75-fa9b724f2681', a size of 1 Gi, and an actual size of 38.6 Mi. It is in a 'Healthy' state and is attached to the 'hello-longhorn' pod on the 'ks-worker-1' node.

State	Name	Size	Actual Size	Created	Data Engine	PV/PVC	Namespace	Attached To	Schedule	Last Backup At	Operation
Healthy	pvc-488a0c37-e99-4764-a75-fa9b724f2681	1 Gi	38.6 Mi	4 minutes ago	v1	Bound	default	hello-longhorn on ks-worker-1			

Suppression du pod :

```
kubectl delete pod hello-longhorn
```

Suppression du pvc :

```
kubectl delete pvc longhorn-hello-pvc
```

Revision #2

Created 20 April 2026 15:40:15 by Nehemie

Updated 20 April 2026 15:49:52 by Nehemie