

Guía para instalar Portainer CE en Kubernetes con almacenamiento persistente usando local-path-provisioner en Arch Linux

1. Preparación en Arch Linux (nodos del cluster)

El provisioner `local-path` usa un directorio en el nodo para almacenar datos persistentes. Por defecto usa `/opt/local-path-provisioner`.

Pasos:

- Asegúrate que en cada nodo físico o VM de tu cluster exista el directorio:

```
sudo mkdir -p /opt/local-path-provisioner
```

- Otorga permisos de escritura para evitar problemas de acceso:

```
sudo chmod 777 /opt/local-path-provisioner
```

2. Instalación del provisioner local-path en Kubernetes

Este provisioner permite que PVC con StorageClass `local-path` funcionen, creando volúmenes persistentes usando el directorio de cada nodo.

Ejecuta:

```
kubectl apply -f https://raw.githubusercontent.com/rancher/local-path-  
provisioner/master/deploy/local-path-storage.yaml
```

Esto crea:

- Namespace `local-path-storage`
- ServiceAccount, Roles y ClusterRoles para el provisioner
- Deployment del provisioner
- StorageClass `local-path`

3. Verifica la instalación y que el provisioner esté corriendo

```
kubectl get pods -n local-path-storage  
kubectl logs -n local-path-storage <nombre-del-pod-local-path-provisioner>  
kubectl get storageclass
```

Debes ver el StorageClass `local-path` y el provisioner corriendo sin errores.

4. Establecer `local-path` como StorageClass por defecto (opcional)

Si deseas que los PVC usen `local-path` por defecto, ejecuta:

```
kubectl patch storageclass local-path -p '{"metadata":  
{"annotations":{"storageclass.kubernetes.io/is-default-class":"true"}}}'
```

5. Crear PersistentVolumeClaim (PVC) para Portainer

Archivo `pvc-portainer.yml`:

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: portainer
  namespace: portainer
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 50Gi
  storageClassName: local-path
```

Aplica el PVC con:

```
kubectl apply -f pvc-portainer.yml
```

6. Instalación y despliegue de Portainer CE

Usando Helm (recomendado):

```
helm repo add portainer https://portainer.github.io/k8s/
helm repo update

helm upgrade --install --create-namespace -n portainer portainer portainer/portainer \
  --set persistence.storageClass=local-path \
  --set persistence.size=50Gi \
  --set image.tag=lts
```

Usando YAML (alternativo):

```
kubectl apply -n portainer -f https://downloads.portainer.io/ce-lts/portainer.yaml
```

7. Resolución de problemas comunes

- Si el PVC queda en estado `Pending`, verifica que:
 - El provisioner `local-path` esté corriendo correctamente.
 - El directorio `/opt/local-path-provisioner` exista en todos los nodos y tenga permisos adecuados.
 - No exista un PVC previo con configuración incompatible (en ese caso, elimina el PVC y vuelve a crear).
- Para eliminar PVC problemáticos:

```
kubectl delete pvc portainer -n portainer  
kubectl apply -f pvc-portainer.yml
```

- El pod de Portainer puede quedar en estado `Pending` si el PVC no está enlazado a un PV.

8. Verificación final

- Revisa el estado de los pods y servicios:

```
kubectl get pods -n portainer  
kubectl get svc -n portainer
```

- Accede a Portainer a través del navegador con la IP del nodo y el puerto NodePort asignado (por ejemplo, `https://<IP-Nodo>:30779`).

Resumen de comandos útiles

```
# En cada nodo físico o VM  
sudo mkdir -p /opt/local-path-provisioner  
sudo chmod 777 /opt/local-path-provisioner
```

```
# Instalar provisioner local-path
kubectl apply -f https://raw.githubusercontent.com/rancher/local-path-
provisioner/master/deploy/local-path-storage.yaml

# Verificar provisión
kubectl get pods -n local-path-storage
kubectl get storageclass

# Opcional: hacer local-path storageclass default
kubectl patch storageclass local-path -p '{"metadata":
{"annotations":{"storageclass.kubernetes.io/is-default-class":"true"}}}'

# Crear PVC para Portainer
kubectl apply -f pvc-portainer.yml

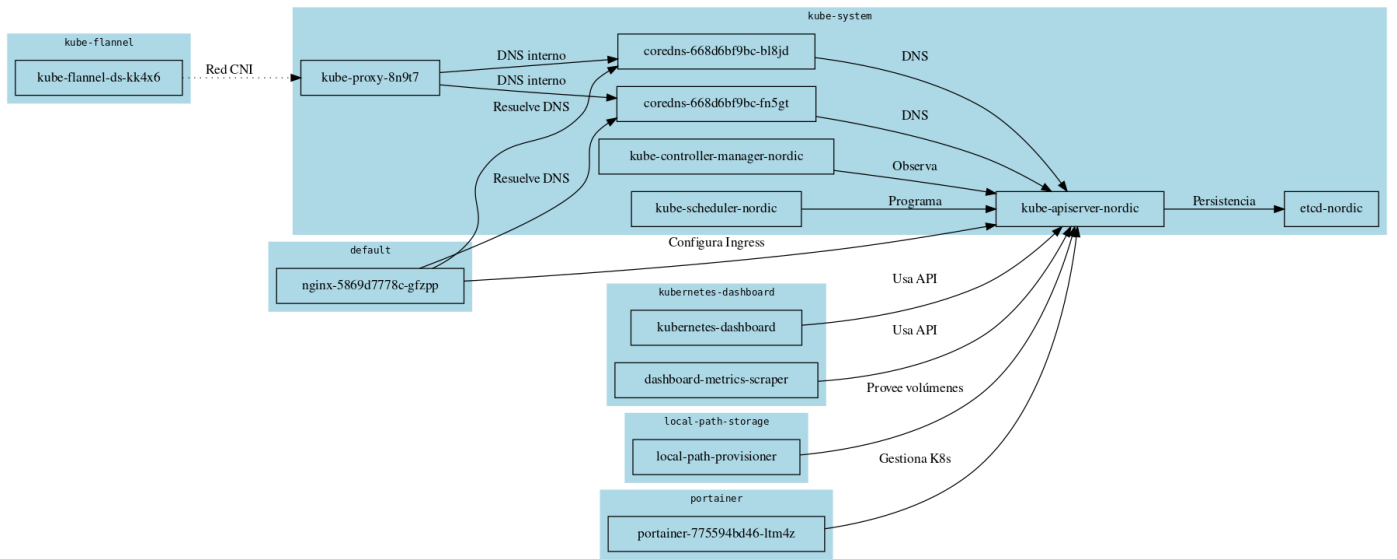
# Instalar Portainer con Helm
helm repo add portainer https://portainer.github.io/k8s/
helm repo update
helm upgrade --install --create-namespace -n portainer portainer portainer/portainer --set
persistence.storageClass=local-path --set persistence.size=50Gi --set image.tag=lts

# Alternativa: instalar Portainer con YAML
kubectl apply -n portainer -f https://downloads.portainer.io/ce-lts/portainer.yaml

# Si es necesario, eliminar PVC problemático
kubectl delete pvc portainer -n portainer
```

Arquitectura de la infraestructura

Al final los servicios quedan de esta forma!



Y listo tenemos el servicio de portainer corriendo en nuestro baremetal.

Environment summary

Dashboard

Environment info

Environment	local
URL	kubernetes.default.svc
Tags	-

8 Namespaces

7 Services

18 ConfigMaps

1 Volume

8 Applications

0 Ingresses

5 Secrets

Applications

Application list

Name	Stack	Namespace	Image	Application Type	Status	Published	Created
dashboard-metrics-scraper	External	kubernetes-dashboard	kubernetes/metrics-scraper:v1.0.8	Deployment	Replicated 1 / 1	Yes	2025-07-15 00:47:20
kube-flannel-ds	External	kube-flannel	ghcr.io/flannel-io/flannel:v0.27.0	DaemonSet	Global 1 / 1	No	2025-07-15 00:33:58
kubernetes-dashboard	External	kubernetes-dashboard	kubernetes/dashboard:v2.7.0	Deployment	Replicated 1 / 1	Yes	2025-07-15 00:47:20
local-path-provisioner	External	local-path-storage	rancher/local-path-provisioner:v0.0.31	Deployment	Replicated 1 / 1	No	2025-07-15 01:01:11
nginx	External	default	nginx	Deployment	Replicated 1 / 1	Yes	2025-07-15 00:36:15