

## Use NFS for Persistent Volumes



SHEIKH KAMRAN MUNEER · May 19, 2023 · 2 min read

PLAY THIS ARTICLE

▶ 0:00 / 2:25 SPEED 1X

### TABLE OF CONTENTS

#### Lab Setup:

1. Setup an NFS server
2. Install the CSI driver for NFS
3. Create a StorageClass for NFS

### Installation On Microk8s-NFS for Persistent Volumes

#### Lab Setup:

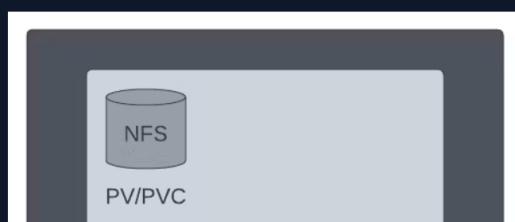
```
19/05/2023 12:03:59 /home/mobaxterm ssh generic@192.168.1.14
generic@192.168.1.14's password:
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.15.0-72-generic x86_64)

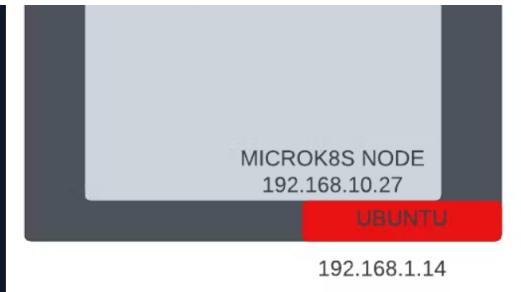
 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

System information as of Fri May 19 04:06:12 AM UTC 2023

System load: 0.9228515625 Processes: 118
Usage of /: 28.5% of 9.75GB Users logged in: 0
Memory usage: 5%
Swap usage: 0% IPv4 address for ensps3: 192.168.1.14
IPv4 address for ensps8: 192.168.10.27
```

Bridge Network - use for access terminal  
NAT - Underlay Network - Node IP Address





## 1. Setup an NFS server

```
sudo apt-get install nfs-kernel-server
```

```
root@microk8s2:~# sudo apt-get install nfs-kernel-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  keyutils libnfsidmap1 nfs-common rpcbind
Suggested packages:
  watchdog
The following NEW packages will be installed:
  keyutils libnfsidmap1 nfs-common nfs-kernel-server rpcbind
0 upgraded, 5 newly installed, 0 to remove and 48 not upgraded.
Need to get 521 kB of archives.
After this operation, 1,973 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```

Create a directory to be used for NFS:

```
sudo mkdir -p /srv/nfs
```

```
sudo chown nobody:nogroup /srv/nfs
```

```
sudo chmod 0777 /srv/nfs
```

```
root@microk8s2:~# sudo mkdir -p /srv/nfs
sudo chown nobody:nogroup /srv/nfs
sudo chmod 0777 /srv/nfs
root@microk8s2:~#
root@microk8s2:~#
root@microk8s2:~#
```

Make sure that the IP addresses of all your MicroK8s nodes are able to mount this share.

```
sudo mv /etc/exports /etc/exports.bak
```

```
echo '/srv/nfs 192.168.10.0/24(rw,sync,no_subtree_check)' | sudo tee /etc/exports
```

```
root@microk8s2:~# microk8s kubectl get no --no-headers
NAME      STATUS   ROLES    AGE     VERSION   INTERNAL-IP   EXTERNAL-IP   OS-IMAGE          KERNEL-VERSION   CONTAINER-RUNTIME
microk8s2  Ready    <none>   102m   v1.26.4   192.168.10.27  <none>        Ubuntu 22.04.2 LTS   5.15.0-72-generic   containerd://1.6.15
root@microk8s2:~# 
root@microk8s2:~# 
root@microk8s2:~# echo '/srv/nfs 192.168.10.0/24(rw,sync,no_subtree_check)' | sudo tee /etc/exports
echo: command not found
root@microk8s2:~# echo '/srv/nfs 192.168.10.0/24(rw,sync,no_subtree_check)' | sudo tee /etc/exports
/srv/nfs 192.168.10.0/24(rw,sync,no_subtree_check)
root@microk8s2:~# 
root@microk8s2:~#
```

Restart server.

```
sudo systemctl restart nfs-kernel-server
```

```
root@microk8s2:~#
root@microk8s2:~# sudo systemctl restart nfs-kernel-server
root@microk8s2:~#
root@microk8s2:~#
```

## 2. Install the CSI driver for NFS

```
microk8s enable helm3 microk8s helm3 repo add csi-driver-nfs
```

```
https://raw.githubusercontent.com/kubernetes-csi/csi-driver-nfs/master/charts
```

```
microk8s helm3 repo update
```

```
root@microk8s2:~# microk8s enable helm3
microk8s helm3 repo add csi-driver-nfs https://raw.githubusercontent.com/kubernetes-csi/csi-driver-nfs/master/charts
microk8s helm3 repo update
Infer repository core for addon helm3
Addon core/helm3 is already enabled
"csi-driver-nfs" has been added to your repositories ...
Hang tight while we grab the latest from your chart repositories ...
...Successfully got an update from the "csi-driver-nfs" chart repository
Update Complete. *appy Helmng!
root@microk8s2:~#
```

install the Helm chart under the `kube-system` namespace with

```
microk8s helm3 install csi-driver-nfs csi-driver-nfs/csi-driver-nfs \ --namespace kube-
system \ --set kubeletDir=/var/snap/microk8s/common/var/lib/kubelet
```

```
root@microk8s2:~# microk8s helm3 install csi-driver-nfs csi-driver-nfs/csi-driver-nfs \
--namespace kube-system \
--set kubeletDir=/var/snap/microk8s/common/var/lib/kubelet
NAME: csi-driver-nfs
LAST DEPLOYED: Fri May 19 06:06:15 2023
NAMESPACE: kube-system
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
The CSI NFS Driver is getting deployed to your cluster.

To check CSI NFS Driver pods status, please run:

  kubectl --namespace=kube-system get pods --selector="app.kubernetes.io/instance=csi-driver-nfs" --watch
root@microk8s2:~#
```

CSI controller and node pods to come up

```
microk8s kubectl --namespace=kube-system get pods --
selector="app.kubernetes.io/instance=csi-driver-nfs" --watch
```

```
root@microk8s2:~# microk8s kubectl --namespace=kube-system get pods --selector="app.kubernetes.io/instance=csi-driver-nfs" --watch
NAME          READY   STATUS    RESTARTS   AGE
csi-nfs-controller-f9bd9fc-b494t  3/3     Running   0          2m26s
csi-nfs-node-29nkz   3/3     Running   0          2m26s
root@microk8s2:~#
```

```
microk8s kubectl wait pod --selector app.kubernetes.io/name=csi-driver-nfs --for
condition=ready --namespace kube-system
```

```
root@microk8s2:~# microk8s kubectl wait pod --selector app.kubernetes.io/name=csi-driver-nfs --for condition=ready --namespace kube-system
pod/csi-nfs-controller-f9bd9fc-b494t condition met -->
pod/csi-nfs-node-29nkz condition met -->
root@microk8s2:~#
```

Verification CSI Drivers.

```
microk8s kubectl get csidrivers
```

```
root@microk8s2:~# microk8s kubectl get csidrivers
NAME          ATTACHEQUIRED   PODINFOUMOUNT   STORAGECAPACITY   TOKENREQUESTS   REQUIRESRERUBLISH   MODES   AGE
nfs.csi.k8s.io  false           false           false            <unset>        false             Persistent   10m
root@microk8s2:~#
```

### 3. Create a StorageClass for NFS

Create a Kubernetes Storage Class that uses the `nfs.csi.k8s.io` CSI driver

```
# sc-nfs.yaml
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: nfs-csi
provisioner: nfs.csi.k8s.io
parameters:
  server: 192.168.10.27
  share: /srv/nfs
reclaimPolicy: Delete
volumeBindingMode: Immediate
mountOptions:
  - hard
  - nfsvers=4.1
  ~
  ~
```

```
microk8s kubectl apply -f sc-nfs.yaml
```

```
root@microk8s2:~/nfs# ls -l
total 4
-rw-r--r-- 1 root root 265 May 19 06:27 sc-nfs.yaml
root@microk8s2:~/nfs# microk8s kubectl apply -f sc-nfs.yaml
storageclass.storage.k8s.io/nfs-csi created
root@microk8s2:~/nfs#
```

```
root@microk8s2:~/nfs# microk8s kubectl get no -o wide
NAME      STATUS   ROLES      AGE   VERSION   INTERNAL-IP     EXTERNAL-IP   OS-IMAGE    KERNEL-VERSION   CONTAINER-RUNTIME
microk8s2 Ready    <none>    134m  v1.26.4   192.168.10.27  <none>        Ubuntu 22.04.2 LTS   5.15.0-72-generic  containerd://1.6.15
root@microk8s2:~/nfs# vim sc-nfs.yaml
root@microk8s2:~/nfs# cat sc-nfs.yaml
# sc-nfs.yaml

apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: nfs-csi
provisioner: nfs.csi.k8s.io
parameters:
  server: 192.168.10.27
  share: /srv/nfs
reclaimPolicy: Delete
volumeBindingMode: Immediate
mountOptions:
  - hard
  - nfsv3=4.1
root@microk8s2:~/nfs#
```

Persistent Volume Claim Yml.

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: my-pvc
spec:
  storageClassName: nfs-csi
  accessModes: [ReadWriteOnce]
  resources:
    requests:
      storage: 1Gi
~
```

```
microk8s kubectl apply -f pvc1.yaml
```

```
microk8s kubectl get pv
```

```
microk8s kubectl get pvc
```

```
root@microk8s2:~/nfs# microk8s kubectl get pvc
NAME      STATUS   VOLUME                                     CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM           STORAGECLASS   REASON   AGE
pvc-c628e7d5-a23d-4c45-b3cb-b770ae0adb61   1Gi        RWO            Delete          Bound   default/my-pvc   nfs-csi        7m34s
root@microk8s2:~/nfs#
root@microk8s2:~/nfs#
root@microk8s2:~/nfs#
root@microk8s2:~/nfs#
root@microk8s2:~/nfs#
root@microk8s2:~/nfs#
root@microk8s2:~/nfs# microk8s kubectl get pvc
NAME      STATUS   VOLUME                                     CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM           STORAGECLASS   AGE
my-pvc   Bound   pvc-c628e7d5-a23d-4c45-b3cb-b770ae0adb61   1Gi        RWO            Delete          Bound   default/my-pvc   nfs-csi        25m
root@microk8s2:~/nfs#
```

End Of Lab - Good Luck!



## Subscribe to my newsletter

Read articles from **Integrating Jenkins with GitHub Webhooks** directly inside your inbox. Subscribe to the newsletter, and don't miss out.

sheikhkamranmuneerj@gmail.com

SUBSCRIBE

Docker

kubernetes-persistent-volumes

AWS NFS

Nfs

KUBER



WRITTEN BY

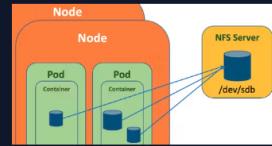
**SHEIKH KAMRAN MUNEER**

[Add Profile Bio](#)

ARTICLE SERIES  
**kubernetes**

### Use NFS for Persistent Volumes

- 1 Installation On Microk8s-NFS for Persistent Volumes Lab Setup: 1. Setup an NFS server sudo apt-get...



### Running Kubernetes locally on Linux with Microk8s

- 2 MICROK8S Installation:  
<https://kubernetes.io/blog/2019/11/26/running-kubernetes-locally-on-linux-with-microk8s>...

# MicroK8s

©2023 Integrating Jenkins with GitHub Webhooks

[Archive](#) · [Privacy policy](#) · [Terms](#)

[Publish with Hashnode](#)

Powered by [Hashnode](#) - Home for tech writers and readers