Running Kubernetes locally on Linux with Microk8s

SHEIKH KAMRAN MUNEER · May 19, 2023 · 1 min read

PLAY THIS ARTICLE

0:00 / 0:49

SPEED 1X

TABLE OF CONTENTS

MICROK8S Installation:

https://kubernetes.io/blog/2019/11/26/running-kubernetes-locally-on-linux-with-microk8s/

```
sudo snap install microk8s --classic
```

```
2. root@microk8s2: ~
root@microk8s2:~# sudo snap install microk8s --classic
Download snap "core18" (2745) from channel "stable"
```

```
sudo microk8s.status
```

```
root@microk8s2:~# sudo snap install microk8s --classic
microk8s (1.26/stable) v1.26.4 from Canonical✓ installed
root@microk8s2:~# sudo microk8s.status
microk8s is running
high-availability: no
  datastore master nodes: 127.0.0.1:19001
  datastore standby nodes: none
addons:
  enabled:
    ha-cluster          # (core) Configure high availability on the current node
    helm                # (core) Helm - the package manager for Kubernetes
    helm3               # (core) Helm 3 - the package manager for Kubernetes
  disabled:
    cert-manager        # (core) Cloud native certificate management
    community           # (core) The community addons repository
    dashboard           # (core) The Kubernetes dashboard
```

```
    dns                  # (core) CoreDNS
    gpu                  # (core) Automatic enablement of Nvidia CUDA
    host-access          # (core) Allow Pods connecting to Host services smoothly
    hostpath-storage     # (core) Storage class; allocates storage from host directory
    ingress              # (core) Ingress controller for external access
    kube-ovn             # (core) An advanced network fabric for Kubernetes
    mayastor             # (core) OpenEBS MayaStor
    metallb              # (core) Loadbalancer for your Kubernetes cluster
    metrics-server       # (core) K8s Metrics Server for API access to service metrics
    minio                # (core) MinIO object storage
    observability        # (core) A lightweight observability stack for logs, traces and metrics
    prometheus           # (core) Prometheus operator for monitoring and logging
    rbac                 # (core) Role-Based Access Control for authorisation
    registry             # (core) Private image registry exposed on localhost:32000
    storage              # (core) Alias to hostpath-storage add-on, deprecated
root@microk8s2:~#
```

`microk8s enable dns dashboard host-access hostpath-storage ingress rbac storage metrics-server`

```
root@microk8s2:~# microk8s enable dns dashboard host-access hostpath-storage ingress rbac storage metrics-server
Infer repository core for addon dns
Infer repository core for addon dashboard
Infer repository core for addon host-access
Infer repository core for addon hostpath-storage
Infer repository core for addon ingress
Infer repository core for addon rbac
Infer repository core for addon storage
Infer repository core for addon metrics-server
Enabling DNS
Using host configuration from /run/systemd/resolve/resolv.conf
Applying manifest
serviceaccount/coredns created
configmap/coredns created
deployment.apps/coredns created
service/kube-dns created
clusterrole.rbac.authorization.k8s.io/coredns created
clusterrolebinding.rbac.authorization.k8s.io/coredns created
```

```
root@microk8s2:~# microk8s status
microk8s is running
high-availability: no
  datastore master nodes: 127.0.0.1:19001
  datastore standby nodes: none
addons:
  enabled:
    dns                  # (core) CoreDNS
    ha-cluster           # (core) Configure high availability on the current node
    helm                 # (core) Helm - the package manager for Kubernetes
    helm3                # (core) Helm 3 - the package manager for Kubernetes
    host-access          # (core) Allow Pods connecting to Host services smoothly
    hostpath-storage     # (core) Storage class; allocates storage from host directory
    rbac                 # (core) Role-Based Access Control for authorisation
    storage              # (core) Alias to hostpath-storage add-on, deprecated
  disabled:
    cert-manager         # (core) Cloud native certificate management
    community            # (core) The community addons repository
    dashboard            # (core) The Kubernetes dashboard
    gpu                  # (core) Automatic enablement of Nvidia CUDA
    ingress              # (core) Ingress controller for external access
    kube-ovn             # (core) An advanced network fabric for Kubernetes
    mayastor             # (core) OpenEBS MayaStor
    metallb              # (core) Loadbalancer for your Kubernetes cluster
    metrics-server       # (core) K8s Metrics Server for API access to service metrics
    minio                # (core) MinIO object storage
    observability        # (core) A lightweight observability stack for logs, traces and metrics
    prometheus           # (core) Prometheus operator for monitoring and logging
    registry             # (core) Private image registry exposed on localhost:32000
```

Verification Commands:

`microk8s.kubectl get nodes`

`microk8s.kubectl get nodes -o wide`

`microk8s kubectl get ns`

`microk8s kubectl get svc`

`microk8s kubectl get deploy`

```
root@microk8s2:~# microk8s kubectl get no
NAME         STATUS   ROLES    AGE   VERSION
microk8s2    Ready    <none>   25m   v1.26.4
root@microk8s2:~# microk8s kubectl get no -o wide
NAME         STATUS   ROLES    AGE   VERSION   INTERNAL-IP      EXTERNAL-IP   OS-IMAGE             KERNEL-VERSION      CONTAINER-RUNTIME
microk8s2    Ready    <none>   25m   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:~# microk8s kubectl get ns
NAME              STATUS   AGE
kube-system       Active   25m
kube-public       Active   25m
kube-node-lease   Active   25m
default           Active   25m
ingress           Active   4m2s
root@microk8s2:~#
root@microk8s2:~# microk8s kubectl get svc
NAME         TYPE        CLUSTER-IP     EXTERNAL-IP   PORT(S)   AGE
kubernetes   ClusterIP   10.152.183.1   <none>        443/TCP   25m
root@microk8s2:~#
root@microk8s2:~# microk8s kubectl get deploy
No resources found in default namespace.
root@microk8s2:~#
root@microk8s2:~# microk8s kubectl get ingress
No resources found in default namespace.
root@microk8s2:~#
```

Node IP Address

End Of Lab: Good Day!

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

Kubernetes Docker WordPress SQL microk8s

WRITTEN BY
SHEIKH KAMRAN MUNEER
Add Profile Bio

ARTICLE SERIES
kubernetes

1 **Running Kubernetes locally on Linux with Microk8s**
MICROK8S Installation: https://kubernetes.io/blog/2019/11/26/running-kubernetes-locally-on-linux-wit...