Docker and Kubernetes: The Complete Guide

Minikube Setup on Linux

These instructions should be valid for Debian, Ubuntu, or Mint Linux distributions. Your experience may vary if using other distributions such as RHEL, Arch, non-desktop distributions like Ubuntu server, or lightweight distros which may omit many expected tools.

Install Minikube

In your terminal run the following:

```
curl -LO https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64
```

```
sudo install minikube-linux-amd64 /usr/local/bin/minikube
```

Starting Minikube and Testing Installation

After you have successfully installed Minikube we need to start and test the cluster to make sure everything is working correctly.

1. Add your user to the docker group

Note - If this step was performed when Docker was installed, it can be skipped.

In your terminal, run:

```
sudo usermod -aG docker $USER && newgrp docker
```

Log out of the user profile and log back in so these changes take effect. If running inside a VM, you will need to restart the entire machine, not just log out.

2. Start with the default driver:

In your terminal, run:

```
minikube start
```

Your output should look similar to this:

```
dev@machine:~$ minikube start
😄  minikube v1.17.1 on Ubuntu 20.04 (amd64)
✨  Automatically selected the docker driver
👍  Starting control plane node minikube in cluster minikube
🔥  Creating docker container (CPUs=2, Memory=2200MB) ...
🐳  Preparing Kubernetes v1.20.2 on Docker 20.10.2 ...
    ▪ Generating certificates and keys ...
    ▪ Booting up control plane ...
    ▪ Configuring RBAC rules ...
🔎  Verifying Kubernetes components...
🌟  Enabled addons: storage-provisioner, default-storageclass
🏄  Done! kubectl is now configured to use "minikube" cluster and "default" namespace
by default
dev@machine:~$
```

2. Check Minikube Status

After you see a **Done!** message in your terminal, run `minikube status` to make sure the cluster is healthy. Pay particular attention that the **apiserver** is in a "**Running**" state.

```
dev@machine:~$ minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured
timeToStop: Nonexistent

dev@machine:~$
```

3. Install kubectl

In your terminal run the following:

```
curl -LO "https://dl.k8s.io/release/$(curl -L -s
https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl"

sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl
```

4. Test kubectl

Lastly, open up your terminal and make sure that you can run `kubectl version`

```
dev@machine:~$ kubectl version
Client Version: version.Info{Major:"1", Minor:"20", GitVersion:"v1.20.4", GitCommit:"e
87da0bd6e03ec3fea7933c4b5263d151aafd07c", GitTreeState:"clean", BuildDate:"2021-02-18T
16:12:00Z", GoVersion:"go1.15.8", Compiler:"gc", Platform:"linux/amd64"}
Server Version: version.Info{Major:"1", Minor:"20", GitVersion:"v1.20.2", GitCommit:"f
aecb196815e248d3ecfb03c680a4507229c2a56", GitTreeState:"clean", BuildDate:"2021-01-13T
13:20:00Z", GoVersion:"go1.15.5", Compiler:"gc", Platform:"linux/amd64"}
dev@machine:~$
```

Note - the client and server can be off by one minor version without error or issue.