Kubernetes Installation Guide

Aymen Boubetana

Introduction

Kubernetes is an open-source platform designed for automating deployment, scaling, and operations of application containers. This guide explains how to install Kubernetes on various platforms.

1. Prerequisites

Ensure the following requirements are met before installing Kubernetes:

Operating System

Supported versions include:

- Linux distributions: Ubuntu, CentOS, Debian, Fedora, etc.
- Windows or macOS (via a virtual machine or Kubernetes distributions like Minikube).

Hardware Requirements

- At least 2 CPUs (4 CPUs recommended).
- 2GB of RAM (minimum) or 8GB (recommended).
- 20GB of free disk space.

Software Requirements

- Docker (or another container runtime).
- A Kubernetes command-line tool (e.g., `kubectl`).

—

2. Installation on Different Platforms

2.1. Using Minikube (Recommended for Beginners)

Minikube is a lightweight Kubernetes distribution for local testing.

1. **Install Minikube:**

```
curl -LO https://storage.googleapis.com/minikube
    /releases/latest/minikube-linux-amd64
sudo install minikube-linux-amd64 /usr/local/bin
    /minikube
```

2. **Start Minikube:**

```
minikube start
```

This command sets up a local Kubernetes cluster.

3. **Verify Installation:**

```
kubectl get nodes
```

This command should display the nodes in your cluster.

—

2.2. Installing Kubernetes on Ubuntu

For a production-ready setup, follow these steps:

1. **Update the System:**

```
sudo apt update
sudo apt install -y apt-transport-https curl
```

2. **Add Kubernetes Repository:**

```
curl -s https://packages.cloud.google.com/apt/
    doc/apt-key.gpg | sudo apt-key add -
sudo apt-add-repository "deb␣http://apt.
    kubernetes.io/␣kubernetes-xenial␣main"
```

3. **Install Kubernetes Tools:**

```
sudo apt update
sudo apt install -y kubelet kubeadm kubectl
sudo apt-mark hold kubelet kubeadm kubectl
```

4. **Initialize Kubernetes:**

```
sudo kubeadm init
```

5. **Configure kubectl for the Current User:**

```
mkdir -p $HOME/.kube
sudo cp -i /etc/kubernetes/admin.conf $HOME/.
   kube/config
sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

6. **Install a Pod Network Add-On:** Choose a network solution like `Weave` or `Calico`. For example, to install Weave:

```
kubectl apply -f https://cloud.weave.works/k8s/
   net?k8s-version=$(kubectl version | base64 |
   tr -d '\n')
```

—

2.3. Using Kubernetes on Cloud Platforms

- **Google Kubernetes Engine (GKE):** GKE provides a managed Kubernetes service. Visit the GKE documentation for details.

- **Amazon EKS:** Use AWS to set up Kubernetes. Visit the EKS documentation.

- **Azure Kubernetes Service (AKS):** Azure offers Kubernetes as a service. See the AKS documentation.

—

3. Post-Installation Steps

1. **Verify the Kubernetes Cluster:**

```
kubectl cluster-info
```

2. **Deploy a Test Application:**

```
kubectl create deployment nginx --image=nginx
kubectl expose deployment nginx --type=NodePort
   --port=80
kubectl get services
```

   Access the application via the displayed NodePort.

3. **Manage Pods and Nodes:** Use `kubectl` commands to interact with your cluster:

```
kubectl get pods
kubectl get nodes
```

—

4. Troubleshooting

- **kubectl Command Not Found:** Ensure `kubectl` is installed and in your PATH.
- **Cluster Not Starting:** Verify prerequisites and check logs:

```
kubectl logs <pod-name>
```

—

5. Additional Resources

- Kubernetes Documentation: https://kubernetes.io/docs/
- Troubleshooting Guide: Debugging Kubernetes