Kubernetes Cluster Setup Notes on Ubuntu Server

1. Minikube Setup (Using Docker Driver)

Step 1: System Update

```
sudo apt update && sudo apt upgrade -y
```

Step 2: Install Docker

```
sudo apt install -y docker.io
sudo systemctl start docker
sudo systemctl enable docker
```

Step 3: Add User to Docker Group

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

> Logout and login again to apply group changes.

Step 4: Install kubectl CLI

```
curl -LO "https://dl.k8s.io/release/$(curl -s
https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl"
chmod +x kubectl
sudo install kubectl /usr/local/bin/
kubectl version --client
```

Step 5: Install Minikube

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

Step 6: Start Minikube with Docker Driver

```
minikube start --driver=docker
```

Step 7: Verify Installation

```
minikube status
kubectl get nodes
```

2. Kind Setup (on Ubuntu Server)

Step 1: System Update

```
sudo apt update && sudo apt upgrade -y
```

Step 2: Install Docker

```
sudo apt install -y docker.io
sudo systemctl start docker
sudo systemctl enable docker
```

Step 3: Add User to Docker Group

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

Step 4: Install kubectl

```
curl -LO "https://dl.k8s.io/release/v1.26.0/bin/linux/amd64/kubectl"
chmod +x kubectl
sudo mv kubectl /usr/local/bin/
kubectl version --client
```

Step 5: Install Kind

```
curl -Lo ./kind https://kind.sigs.k8s.io/dl/v0.18.0/kind-linux-amd64
chmod +x ./kind
sudo mv ./kind /usr/local/bin/kind
kind --version
```

Step 6: Create Kind Cluster

```
kind create cluster --name my-cluster
```

Step 7: Verify Cluster

```
kubectl get nodes
kubectl cluster-info
```

Step 8: Interact with Cluster

```
kubectl get all
kubectl get services
```

Step 9: Delete Kind Cluster

```
kind delete cluster --name my-cluster
```

3. EKS Cluster Setup on Ubuntu EC2 (AWS)

Step 1: IAM Setup

- Create IAM User: eks-admin with AdministratorAccess
- Generate Access Key and Secret Access Key

Step 2: Launch Ubuntu EC2 (Region: us-west-2)

```
ssh -i "your-key.pem" ubuntu@<EC2-Public-IP>
```

Step 3: Install AWS CLI v2

```
curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o
"awscliv2.zip"
sudo apt install unzip
unzip awscliv2.zip
sudo ./aws/install -i /usr/local/aws-cli -b /usr/local/bin --update
```

Step 4: Configure AWS CLI

```
aws configure
```

> Enter Access Key, Secret Key, Region (us-west-2), and output format (json)

Step 5: Install kubectl

```
curl -o kubectl https://amazon-eks.s3.us-west-
2.amazonaws.com/1.19.6/2021-01-05/bin/linux/amd64/kubectl
chmod +x ./kubectl
sudo mv ./kubectl /usr/local/bin
kubectl version --short --client
```

Step 6: Install eksctl

```
curl --silent --location
"https://github.com/weaveworks/eksctl/releases/latest/download/eksctl_$
(uname -s)_amd64.tar.gz" | tar xz -C /tmp
sudo mv /tmp/eksctl /usr/local/bin
eksctl version
```

Step 7: Create EKS Cluster

```
eksctl create cluster --name my-cluster --region us-west-2 --node-type
t2.micro --nodes-min 2 --nodes-max 2
```

Step 8: Update kubeconfig

```
aws eks update-kubeconfig --region us-west-2 --name my-cluster
```

Step 9: Verify Cluster

```
kubectl get nodes
kubectl cluster-info
```

Step 10: Deploy Manifests (Optional)

```
kubectl create namespace two-tier-ns
kubectl apply -f .
kubectl delete -f .
```

Step 11: Delete EKS Cluster

```
eksctl delete cluster --name my-cluster --region us-west-2
```

Jitendra Kumar Jat