

## Module 7: Kubernetes Assignment-1

### 1. Deploy a Kubernetes cluster for 3 nodes

Requisites: 3 aws instances

One for master node.

Two instaces for worker/slave nodes.

- master: 172.31.86.175
- slave1: 172.31.86.239
- slave2: 172.31.85.113

- connected to master node

```
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.2.0-1017-aws x86_64)

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

System information as of Wed Dec 20 01:59:45 UTC 2023

System load:  0.06689453125      Processes:            121
Usage of /:   20.6% of 7.57GB    Users logged in:     0
Memory usage: 6%                IPv4 address for eth0: 172.31.80.55
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status
```

i-0564f046b39aecfa4 (k8s master)

PublicIPs: 35.171.153.48 PrivateIPs: 172.31.80.55

```
ubuntu@ip-172-31-80-55:~$ vi docker.sh
ubuntu@ip-172-31-80-55:~$ bash docker.sh
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base pigz runc ubuntu-fan
Suggested packages:
  ifupdown aufs-tools cgroupfs-mount | cgroup-lite debootstrap docker-doc rinse zfs-fuse | zfsutils
The following NEW packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base docker.io pigz runc ubuntu-fan
0 upgraded, 8 newly installed, 0 to remove and 26 not upgraded.
Need to get 69.7 MB of archives.
After this operation, 267 MB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 pigz amd64 2.6-1 [63.6 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 bridge-utils amd64 1.7-lubuntu3 [34.4 kB]
```

```

ubuntu@ip-172-31-80-55:~$ vi k8s.sh
ubuntu@ip-172-31-80-55:~$ bash k8s.sh
k8s.sh: line 1: install.sh: command not found
install: target 'k8s.sh' is not a directory
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Fetched 119 kB in 0s (243 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
26 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages will be upgraded:
  binutils binutils-common binutils-x86-64-linux-gnu cryptsetup cryptsetup-bin cryptsetup-initramfs kpartx libbinutils libc-bin libc6
  libcryptsetup12 libctf-nobfd0 libctf0 libssh-4 locales multipath-tools openssh-client openssh-server openssh-sftp-server systemd-hwe-hwdb

```

```

ubuntu@ip-172-31-80-55:~$ mkdir -p $HOME/.kube
ubuntu@ip-172-31-80-55:~$ sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
cp: overwrite '/home/ubuntu/.kube/config'?
ubuntu@ip-172-31-80-55:~$ sudo chown $(id -u):$(id -g) $HOME/.kube/config

```

- Docker is installed on all nodes

```

ubuntu@ip-172-31-86-175:~$ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Get:4 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal/universe amd64 Packages [8628 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal/universe Translation-en [5124 kB]
Get:7 http://security.ubuntu.com/ubuntu focal-security/main amd64 Packages [2640 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal/universe amd64 c-n-f Metadata [265 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal/multiverse amd64 Packages [144 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal/multiverse Translation-en [104 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal/multiverse amd64 c-n-f Metadata [9136 B]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 Packages [3021 kB]
Get:13 http://security.ubuntu.com/ubuntu focal-security/main Translation-en [405 kB]
Get:14 http://security.ubuntu.com/ubuntu focal-security/main amd64 c-n-f Metadata [13.2 kB]
Get:15 http://security.ubuntu.com/ubuntu focal-security/restricted amd64 Packages [2451 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/main Translation-en [488 kB]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 c-n-f Metadata [17.2 kB]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/restricted amd64 Packages [2569 kB]
Get:19 http://security.ubuntu.com/ubuntu focal-security/restricted Translation-en [342 kB]

```

```

ubuntu@ip-172-31-86-175:~$ sudo apt-get install docker.io -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base libidn11 pigz runc ubuntu-fan
Suggested packages:
  ifupdown aufs-tools cgroupfs-mount | cgroup-lite debootstrap docker-doc rinse zfs-fuse | zfsutils
The following NEW packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base docker.io libidn11 pigz runc ubuntu-fan
0 upgraded, 9 newly installed, 0 to remove and 55 not upgraded.
Need to get 63.2 MB of archives.
After this operation, 267 MB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal/universe amd64 pigz amd64 2.4-1 [57.4 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal/main amd64 bridge-utils amd64 1.6-2ubuntu1 [30.5 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 runc amd64 1.1.7-0ubuntu1~20.04.1 [3819 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 containerd amd64 1.7.2-0ubuntu1~20.04.1 [32.5 MB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal/main amd64 dns-root-data all 2019052802 [5300 B]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal/main amd64 libidn11 amd64 1.33-2.2ubuntu2 [46.2 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 dnsmasq-base amd64 2.80-1.1ubuntu1.7 [315 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/universe amd64 docker.io amd64 24.0.5-0ubuntu1~20.04.1 [26.4 MB]

```

- After completion of docker installation kubernetes is installed on master and slave nodes
- To install kubernetes shellscript is created with the following script
  - `sudo apt update`
  - `sudo apt upgrade -y`
  - `sudo apt install -y curl apt-transport-https ca-certificates software-properties-common`
  - `curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add -`
  - `sudo add-apt-repository "deb http://apt.kubernetes.io/ kubernetes-xenial main"`
  - `sudo swapoff -a`
  - `sudo apt update`
  - `sudo apt install -y kubelet kubeadm kubectl`

```
ubuntu@ip-172-31-86-175:~$ sudo nano k8s.sh
ubuntu@ip-172-31-86-175:~$ bash k8s.sh
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu focal-security InRelease
Reading package lists... Done
Building dependency tree
Reading state information... Done
55 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree
```

- After installation kubeadm is initiated “`$ sudo kubeadm init`”,
- At the end of initialization we get admin token which is to be run on worker nodes to join them to the cluster

```
Your Kubernetes control-plane has initialized successfully!

To start using your cluster, you need to run the following as a regular user:

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

Alternatively, if you are the root user, you can run:

  export KUBECONFIG=/etc/kubernetes/admin.conf

You should now deploy a pod network to the cluster.
Run "kubectl apply -f [podnetwork].yaml" with one of the options listed at:
  https://kubernetes.io/docs/concepts/cluster-administration/addons/

Then you can join any number of worker nodes by running the following on each as root:
```

```
kubeadm join 172.31.86.175:6443 --token wq44h7.7yl47nbfkbiwjmdl \
--discovery-token-ca-cert-hash sha256:fe2bab943e153497d2539f5773182c028168b678028c3c0059f6edc8daee5d05
```

- After kubernetes installation, following commands need to be run in only in master

```
mkdir -p $HOME/.kube
```

```
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
```

```
sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

```
kubectl apply -f https://github.com/weaveworks/weave/releases/download/v2.8.1/weave-
daemonset-k8s.yaml
```

```
ubuntu@ip-172-31-86-175:~$ mkdir -p $HOME/.kube
ubuntu@ip-172-31-86-175:~$ sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
ubuntu@ip-172-31-86-175:~$ sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
cp: overwrite '/home/ubuntu/.kube/config'? ^[^X^C
ubuntu@ip-172-31-86-175:~$ sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

```
ubuntu@ip-172-31-86-175:~$ kubectl apply -f https://github.com/weaveworks/weave/releases/download/v2.8.1/weave-daemonset-k8s.yaml
serviceaccount/weave-net created
clusterrole.rbac.authorization.k8s.io/weave-net created
clusterrolebinding.rbac.authorization.k8s.io/weave-net created
role.rbac.authorization.k8s.io/weave-net created
rolebinding.rbac.authorization.k8s.io/weave-net created
daemonset.apps/weave-net created
```

- Now on slave 1 run the admin token to join the node to cluster

```
ubuntu@ip-172-31-86-239:~$ sudo kubeadm join 172.31.86.175:6443 --token wq44h7.7yl47nbfbkiwjmdl --discovery-token-ca-cert-hash sha256:fe2bab943e153497d2539f5773182c028168b678028c3c0059f6edc8daee5d05
[preflight] Running pre-flight checks
[preflight] Reading configuration from the cluster...
[preflight] FYI: You can look at this config file with 'kubectl -n kube-system get cm kubeadm-config -o yaml'
[kubelet-start] Writing kubelet configuration to file "/var/lib/kubelet/config.yaml"
[kubelet-start] Writing kubelet environment file with flags to file "/var/lib/kubelet/kubeadm-flags.env"
[kubelet-start] Starting the kubelet
[kubelet-start] Waiting for the kubelet to perform the TLS Bootstrap...

This node has joined the cluster:
* Certificate signing request was sent to apiservert and a response was received.
* The Kubelet was informed of the new secure connection details.

Run 'kubectl get nodes' on the control-plane to see this node join the cluster.
```

- Similarly on slave2 also run the admin token

```
ubuntu@ip-172-31-85-113:~$ sudo kubeadm join 172.31.86.175:6443 --token wq44h7.7yl47nbfbkiwjmdl --discovery-token-ca-cert-hash sha256:fe2bab943e153497d2539f5773182c028168b678028c3c0059f6edc8daee5d05
[preflight] Running pre-flight checks
[preflight] Reading configuration from the cluster...
[preflight] FYI: You can look at this config file with 'kubectl -n kube-system get cm kubeadm-config -o yaml'
[kubelet-start] Writing kubelet configuration to file "/var/lib/kubelet/config.yaml"
[kubelet-start] Writing kubelet environment file with flags to file "/var/lib/kubelet/kubeadm-flags.env"
[kubelet-start] Starting the kubelet
[kubelet-start] Waiting for the kubelet to perform the TLS Bootstrap...

This node has joined the cluster:
* Certificate signing request was sent to apiservert and a response was received.
* The Kubelet was informed of the new secure connection details.

Run 'kubectl get nodes' on the control-plane to see this node join the cluster.
```

- Kubernetes cluster is successfully deployed

```
ubuntu@ip-172-31-86-175:~$ kubectl get nodes
NAME                STATUS    ROLES    AGE   VERSION
ip-172-31-85-113    Ready    <none>    3m17s v1.28.2
ip-172-31-86-175    Ready    control-plane 45m   v1.28.2
ip-172-31-86-239    Ready    <none>    3m59s v1.28.2
ubuntu@ip-172-31-86-175:~$
```

## 2. Create a NGINX deployment of 3 replicas

- A manifest file is created to deploy nginx with 3 replicas

```
ubuntu@ip-172-31-86-175:~$ sudo hostnamectl set-hostname k8-master
ubuntu@ip-172-31-86-175:~$ vi nginxdeployment.yml
ubuntu@ip-172-31-86-175:~$ cat nginxdeployment.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx
        ports:
        - containerPort: 80
```

- The nginx deployment is deployed to cluster using kubectl apply command

```
ubuntu@ip-172-31-86-175:~$ kubectl apply -f nginxdeployment.yml
deployment.apps/nginx-deployment created
ubuntu@ip-172-31-86-175:~$ kubectl get po
NAME                                READY   STATUS    RESTARTS   AGE
nginx-deployment-7c5ddbdf54-6j788  1/1     Running   0           11s
nginx-deployment-7c5ddbdf54-8jwgh  1/1     Running   0           11s
nginx-deployment-7c5ddbdf54-cmws6   1/1     Running   0           11s
```

```
ubuntu@ip-172-31-86-175:~$ kubectl apply -f nginxdeployment.yml
deployment.apps/nginx-deployment created
ubuntu@ip-172-31-86-175:~$ kubectl get po
NAME                                READY   STATUS    RESTARTS   AGE
nginx-deployment-7c5ddbdf54-6j788  1/1     Running   0           11s
nginx-deployment-7c5ddbdf54-8jwgh  1/1     Running   0           11s
nginx-deployment-7c5ddbdf54-cmws6   1/1     Running   0           11s
ubuntu@ip-172-31-86-175:~$ kubectl get deploy
NAME                READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment    3/3     3             3           12m
ubuntu@ip-172-31-86-175:~$
```