NAMA : Hanif al zikri

NIM : 2301082009

BUILD 1 NODES & 2 CONTAINERS

Running Linux di Cloud AWS – EC2 (Ubuntu)

1. Master-Node (Kubernetes)

a. **Langkah 1: Update Package Index**

Pastikan sistem Ubuntu Anda terupdate:

sudo apt update

b. **Langkah 2: Download** kubectl **Binary**

Unduh versi terbaru kubectl dari repositori resmi Kubernetes:

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

c. **Langkah 3: Berikan Hak Akses Eksekusi**

chmod +x kubectl

d. **Langkah 4: Pindahkan ke** /usr/local/bin

sudo mv kubectl /usr/local/bin/

e. **Langkah 5: Verifikasi Instalasi**

kubectl version --client

Output yang muncul menunjukkan versi kubectl yang terinstal.

f. **2. Instalasi** minikube

minikube memungkinkan Anda menjalankan Kubernetes secara lokal.

g. **Langkah 1: Install Dependensi**

Pastikan curl dan socat sudah terinstal:

sudo apt install -y curl socat

h. **Langkah 2: Download & Install Minikube**

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

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

i. **Langkah 3: Verifikasi Instalasi**

minikube version

```
root@ip-10-0-8-148:/home/ubuntu# minikube version
minikube version: v1.35.0
commit: dd5d320e41b5451cdf3c01891bc4e13d189586ed-dirty
root@ip-10-0-8-148:/home/ubuntu#
```

```
root@ip-10-0-8-148:/home/ubuntu# kubectl version
Client Version: v1.32.3
Kustomize Version: v5.5.0
Server Version: v1.32.0
root@ip-10-0-8-148:/home/ubuntu#
```

Pastikan output menunjukkan versi Minikube yang terinstal.

j. **Jalankan Minikube**
minikube start --force

k. **Langkah 1: Start Minikube Cluster**

minikube start --driver=docker

l. **Catatan**:

- Jika belum ada driver (seperti Docker, VirtualBox, dll.), Minikube akan menggunakan docker sebagai default.
- Pastikan Docker sudah terinstal **(sudo apt install docker.io**).
- Jika ingin menggunakan driver lain (misal VirtualBox), ganti docker dengan virtualbox.

m. **Langkah 2: Periksa Status Cluster**

minikube status

Output:

type: Control Plane

host: Running

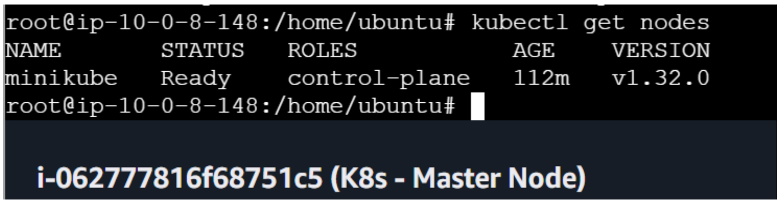
kubelet: Running

apiserver: Running

kubeconfig: Configured

n. **Langkah 3: Uji dengan** kubectl

kubectl get nodes



2. **Container (Worker Node 1)**

Bagian 1: Instalasi Docker di Ubuntu (EC2)

1. **Update package**

   sudo apt update && sudo apt upgrade -y

2. **Install dependensi**

   sudo apt install -y ca-certificates curl gnupg lsb-release

3. **Tambahkan GPG key Docker**

   sudo mkdir -p /etc/apt/keyrings

   curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg

4. **Tambahkan repository Docker**

   echo \

   "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] \

   https://download.docker.com/linux/ubuntu \

   $(lsb_release -cs) stable" | \

   sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

5. **Install Docker Engine**

   sudo apt update

   sudo apt install -y docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin

6. **Tambahkan user EC2 ke grup docker (agar tidak perlu sudo)**

   sudo usermod -aG docker $USER

7. **(Opsional) Logout dan login kembali, atau jalankan:**

   newgrp docker

8. **Cek versi Docker:**

   docker –version

```
root@ip-10-0-14-65:/home/ubuntu# docker version | less
```

i-0a8e1d0e8b3e14b70 (hWorker - Node)

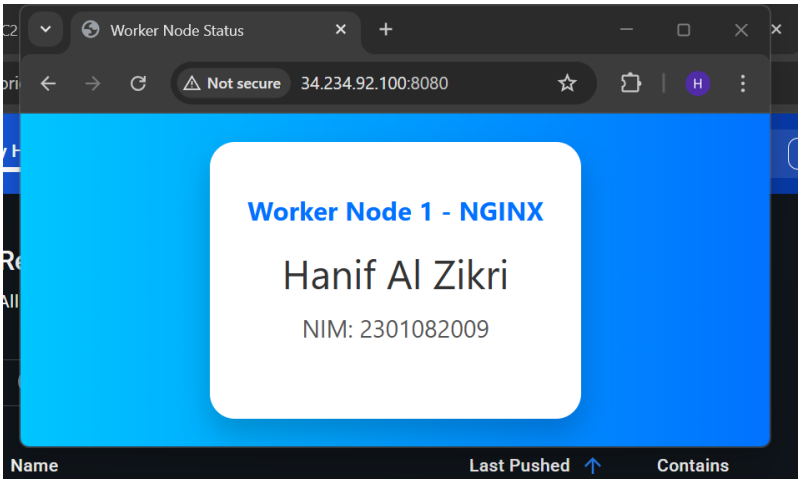
PublicIPs: 34.234.92.100 PrivateIPs: 10.0.14.65

```
Client: Docker Engine - Community
 Version:           28.1.1
 API version:       1.49
 Go version:        go1.23.8
 Git commit:        4eba377
 Built:             Fri Apr 18 09:52:10 2025
 OS/Arch:           linux/amd64
 Context:           default

Server: Docker Engine - Community
 Engine:
  Version:          28.1.1
  API version:      1.49 (minimum version 1.24)
  Go version:       go1.23.8
  Git commit:       01f442b
  Built:            Fri Apr 18 09:52:10 2025
  OS/Arch:          linux/amd64
  Experimental:     false
```

9. **Hasil running container**
   Container – 1

```
root@ip-10-0-14-65:/home/ubuntu# docker ps
CONTAINER ID   IMAGE            COMMAND
MES
de04f5b6facc   h-worker-node1   "/docker-entrypoint.…"
inx-web
root@ip-10-0-14-65:/home/ubuntu#
```



Container – 2

```
CONTAINER ID   IMAGE          COMMAND                  CREATED          STATUS          PORTS
S
573da0279b08   h-worker-2     "/run.sh"                49 seconds ago   Up 48 seconds   0.0.0.0:3000->3000/tcp,
```