Mohammad Saeed Pourmazar

https://github.com/MohammadSaeedPourmazar

https://gitlab.com/MohammadSaeedPourmazar

https://medium.com/@MohammadSaeedPourmazar

https://dev.to/MohammadSaeedPourmazar

https://www.youtube.com/@MohammadSaeedPourmazar

https://www.instagram.com/MohammadSaeedPourmazar

https://www.facebook.com/MohammadSaeedPourmazar

https://www.linkedin.com/in/MohammadSaeedPourmazar/

https://orcid.org/0009-0008-9383-419X

Install Kubernetes On Ubuntu (Worker-Nodes)

**** Ensure Docker is installed on your system ****

1. Update System:

First: update your system packages to the latest version:

sudo apt update

sudo apt upgrade -y

sudo reboot

2. Set Hostname:

Set your VM's hostname to Worker.Node:

sudo hostnamectl set-hostname Worker.Node.1

sudo hostnamectl set-hostname Worker.Node.2

3. Update /etc/hosts:

Edit the /etc/hosts file to include the VM's IP and hostname:

sudo nano /etc/hosts

*Add the following line * don't forget Master.Nodes if you have * :*

IP Address Master.Node

IP Address Worker.Node

4. Disable Swap:

Kubernetes requires swap to be disabled:

sudo swapoff -a

sudo nano /etc/fstab

** Comment out any swap entries (add # in front) **

sudo mount -a

The sudo mount -a command is used to mount all file systems listed in /etc/fstab that are not currently mounted. This is useful when you've made changes to /etc/fstab and want to apply them without rebooting.

5. Load Kernel Modules:

Load the necessary kernel modules for Kubernetes networking:

sudo tee /etc/modules-load.d/containerd.conf <<EOF

overlay

br_netfilter

EOF

sudo modprobe overlay

sudo modprobe br_netfilter

6. Set Kernel Parameters:

Configure the kernel parameters for Kubernetes:

sudo tee /etc/sysctl.d/kubernetes.conf <<EOF

net.bridge.bridge-nf-call-ip6tables = 1

net.bridge.bridge-nf-call-iptables = 1

net.ipv4.ip_forward = 1

EOF

sudo sysctl --system

7. Install Docker (Containerd):

Install containerd for container runtime:

sudo apt install -y curl gnupg2 software-properties-common apt-transport-https ca-certificates

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

sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"

sudo apt update

sudo apt install -y containerd.io

Configure containerd:

containerd config default | sudo tee /etc/containerd/config.toml >/dev/null 2>&1

sudo sed -i 's/SystemdCgroup = false/SystemdCgroup = true/g' /etc/containerd/config.toml

sudo systemctl restart containerd

sudo systemctl enable containerd

8. Install Kubernetes:

Install kubeadm, kubelet, and kubectl (The Latest):

sudo apt-get update

sudo apt-get install -y apt-transport-https ca-certificates curl gpg

curl -fsSL https://pkgs.k8s.io/core:/stable:/v1.32/deb/Release.key | sudo gpg --dearmor -o /etc/apt/keyrings/kubernetes-apt-keyring.gpg

echo 'deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg] https://pkgs.k8s.io/core:/stable:/v1.32/deb/ /' | sudo tee /etc/apt/sources.list.d/kubernetes.list

sudo apt-get update

sudo apt-get install -y kubelet kubeadm kubectl

sudo apt-mark hold kubelet kubeadm kubectl

Installing Specific Versions of Kubernetes Components:

To install a specific version of kubeadm, kubelet, kubectl(e.g., v1.30.0)follow these steps:

1. Add the Correct Kubernetes Repository:

sudo apt-get update

sudo apt-get install -y curl

sudo mkdir -p /etc/apt/keyrings

curl -fsSL https://pkgs.k8s.io/core:/stable:/v1.30/deb/Release.key | sudo tee /etc/apt/keyrings/kubernetes-apt-keyring.asc

echo "deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.asc] https://pkgs.k8s.io/core:/stable:/v1.30/deb/ /" | sudo tee /etc/apt/sources.list.d/kubernetes.list

2. Update the package list again:

sudo apt-get update

3. Install specific versions of kubelet, kubeadm, and kubectl:

sudo apt-get install -y kubelet=1.30.0-1.1 kubeadm=1.30.0-1.1 kubectl=1.30.0-1.1

4. Prevent these packages from being automatically updated:

sudo apt-mark hold kubelet kubeadm kubectl

9. Reseting Kubernetes Setup and Synchronize Time Using NTP:

sudo apt-get install ntp

sudo apt-get install ntpdate

sudo ntpdate ntp.ubuntu.com

10. Retrieve the Token From The Master Node:

Run the following command on the master node to get the join token:

kubeadm token create --print-join-command

11. Run The kubeadm Join Command On The Worker Node From Master Node:

**sudo kubeadm join <MASTER_IP>:6443 --token <TOKEN> **

--discovery-token-ca-cert-hash sha256:<CERT_HASH>