|  |
| --- |
| * #!/bin/sh * curl -fsSL https://get.docker.com -o get-docker.sh * sh get-docker.sh * sudo usermod -aG docker azureuser * sudo swapoff -a |

Installation of Kubernetes

Create 2 or 3 VM’s by giving the following commands in advance

vm's connect to PowerShell

* docker info
* sudo -i
* wget https://storage.googleapis.com/golang/getgo/installer\_linux
* chmod +x ./installer\_linux
* ./installer\_linux
* source ~/.bash profile
* git clone https://github.com/Mirantis/cri-dockerd.git
* cd cri-dockerd
* mkdir bin
* go build -o bin/cri-dockerd
* mkdir -p /usr/local/bin
* install -o root -g root -m 0755 bin/cri-dockerd /usr/local/bin/cri-dockerd
* cp -a packaging/systemd/\* /etc/systemd/system
* sed -i -e 's,/usr/bin/cri-dockerd,/usr/local/bin/cri-dockerd,' /etc/systemd/system/cri-docker.service
* systemctl daemon-reload
* systemctl enable cri-docker.service
* systemctl enable --now cri-docker.socket
* cd ~
* sudo apt-get update
* sudo apt-get install -y apt-transport-https ca-certificates curl
* sudo curl -fsSLo /etc/apt/keyrings/kubernetes-archive-keyring.gpg https://packages.cloud.google.com/apt/doc/apt-key.gpg
* echo "deb [signed-by=/etc/apt/keyrings/kubernetes-archive-keyring.gpg] https://apt.kubernetes.io/ kubernetes-xenial main" | 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

### now doing only master node

* kubeadm --help
* kubeadm init --pod-network-cidr "10.244.0.0/16"
* kubeadm init --pod-network-cidr "10.244.0.0/16" --cri-socket "unix:///var/run/cri-dockerd.sock"

### copy the output of above command from to start to token in a notepad

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

### above three commands from notepad only

* kubectl get nodes
* kubectl get nodes -w
* kubectl apply -f <https://github.com/flannel-io/flannel/releases/latest/download/kube-flannel.yml>
* kubectl get nodes
* kubectl get nodes -w

### Now doing in Nodes

Paste the command from notepad and add this in between

--cri-socket "unix:///var/run/cri-dockerd.sock"

### Then check in master node by using these commands

* kubectl get nodes
* kubectl get nodes -w

### create manifest files and execution

* curl <https://jsonplaceholder.typicode.com/todos/1>
* curl https://jsonplaceholder.typicode.com/todos
* kubectl api-resources

### go to kubectl cheat sheet

* Kubectl (tab button)
* source <(kubectl completion bash)
* Kubectl get p
* echo "source <(kubectl completion bash)" >> ~/.bashrc
* Exit

###Reconnect the VM’s again. Create a manifest file and upload into git hub

* git clone (paste github code)
* cd (manifest file)
* Ls
* Kubectl apply -f (yaml file name)
* kubectl get pods
* Kubectl api-resources
* kubectl api-resources |grep pod
* Kubectl get po (or) kubectl get pods
* Kubectl get pods -o wide
* Kubectl describe pods
* Kubectl describe pods (name in yaml file)
* Kubectl get pods -o yaml
* kubectl delete -f http.yaml
* kubectl get po
* Kubectl apply -f (yaml file name)
* kubectl get pods
* Kubectl get pods -o wide
* Kubectl get po (name in yaml file) -o yaml
* Kubectl apply -f (yaml file name)
* Kubectl get pods -o wide (check ip address)
* kubectl get po
* kubectl delete pods alpine httpd-pod
* git pull

### Apply the following commands to excute the manifest file

* kubectl apply -f (file name)
* kubectl get po
* kubectl get po -o wide