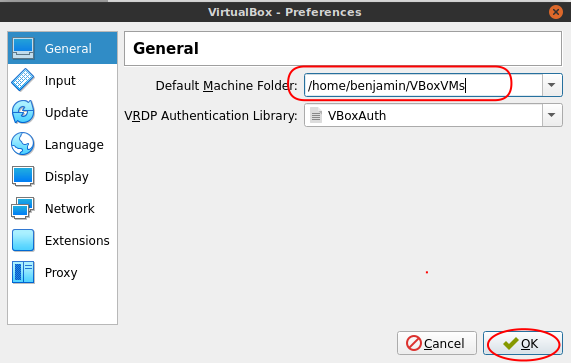
## Install Virtual Box Image

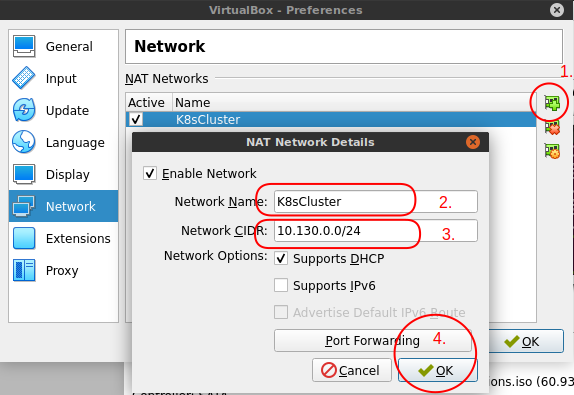
1.



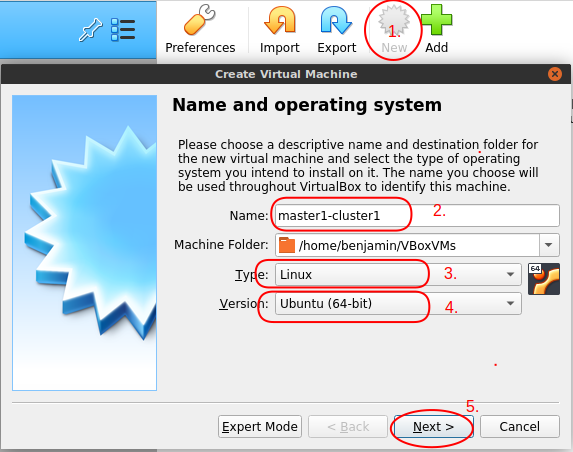
2.



3.



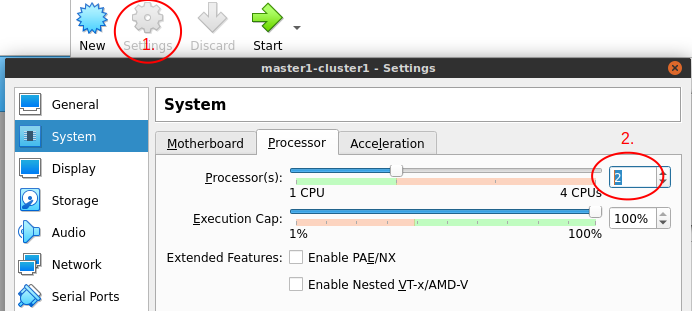
4.



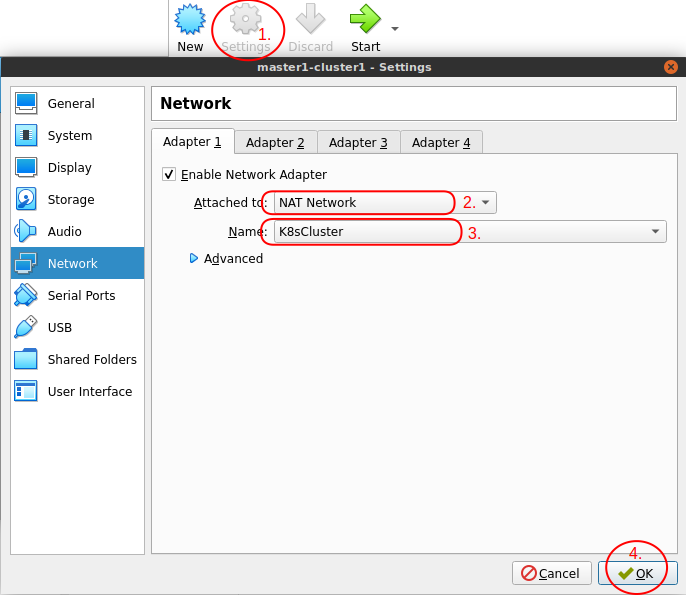
5. 2048 MB (Memory size) - Next->, (Hard Disk) - Create

6. (Hard disk file type) - Next ->, (Storage on physical hard disk) - Next->,   
 (File location and size) (15.00 GB) - Create

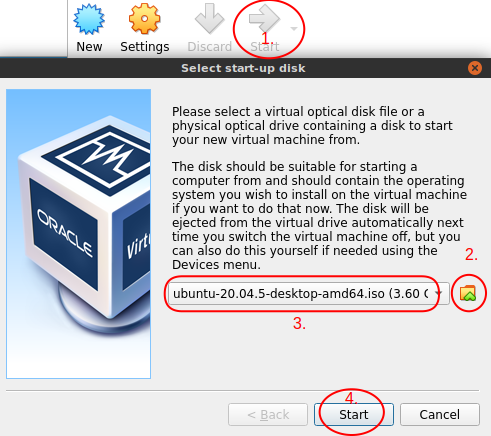
7.



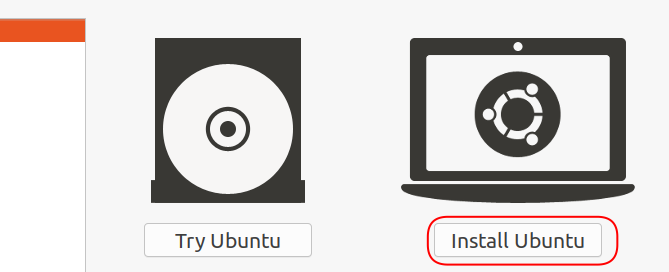
8.



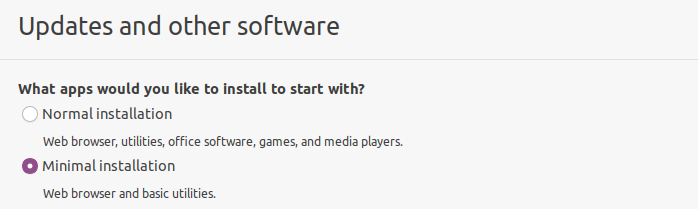
9.



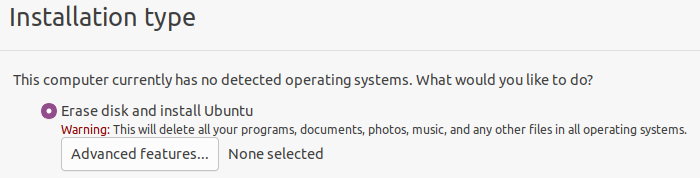
10.



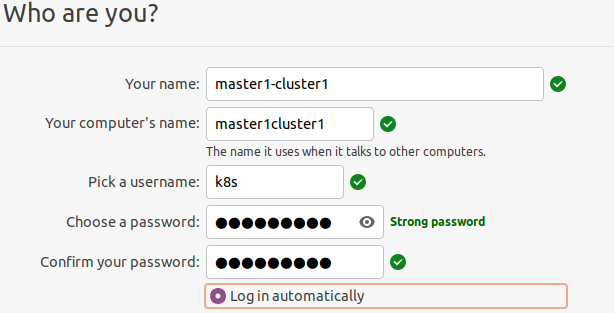
11. Continue ->



12. Continue ->



13. Install Now, …

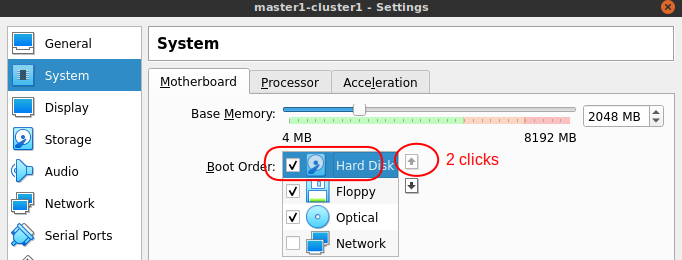


14. Continue, and then click the Reboot button

15. Open the Terminal and disable swap  
sudo swapoff -a  
sudo sed -i '/swap/d' /etc/fstab

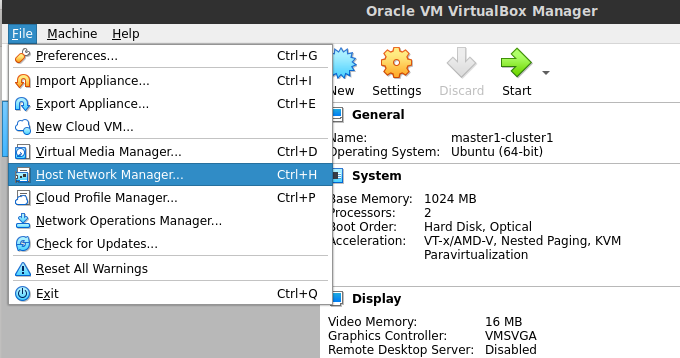
**NOTE**: Run all system and software updates

16. Shutdown VM (Close, Power off the machine, and Change boot order)

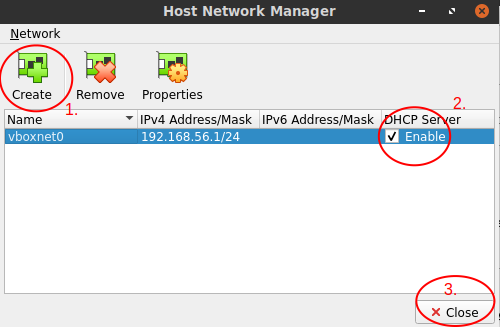


## Install OpenSSH

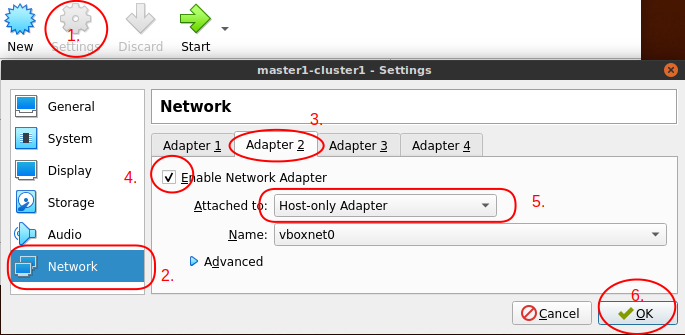
1. (Install OpenSSH on each VM)



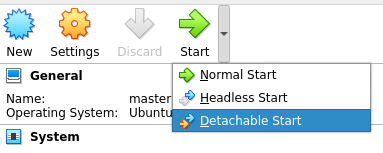
2.



3.



4.



5. Open the Terminal

sudo apt update  
sudo apt upgrade -y  
sudo apt install curl openssh-server -y

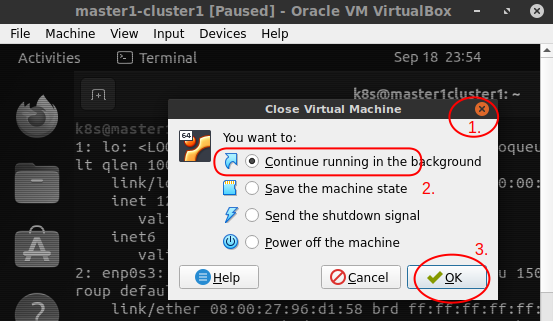
6. Type ip addr show

Find local network inet ip (192.168.xx.xx)

7. Open your Local Terminal and type (Test SSH connection)

ssh k8s@192.168.xx.xx

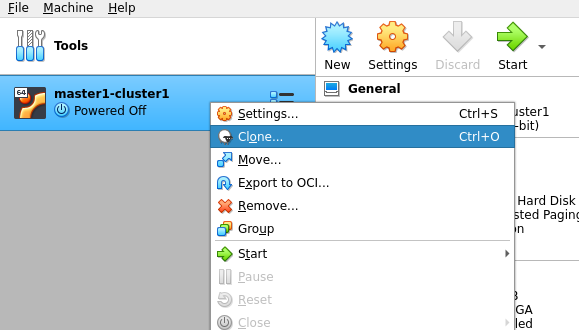
8. Close VM



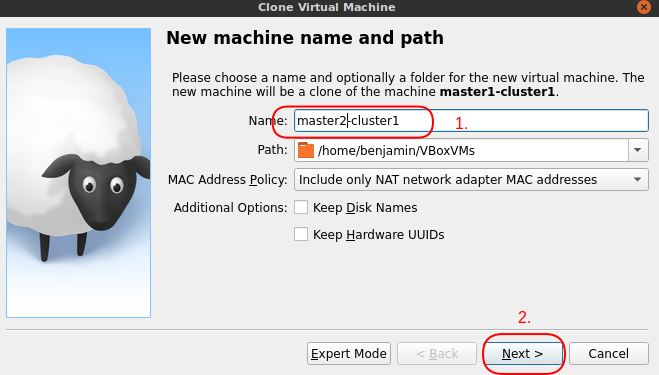
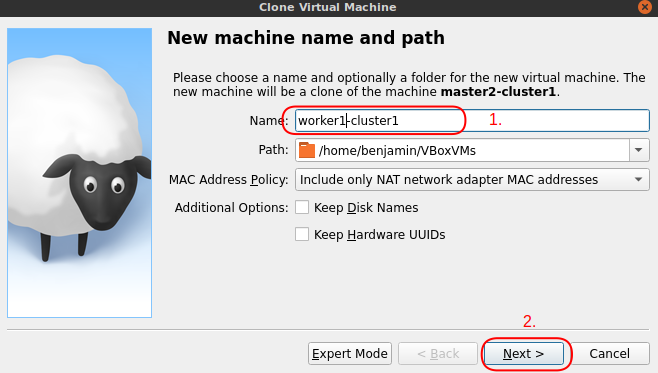
## Clone VM

## Create multiple workers or/and masters

1.

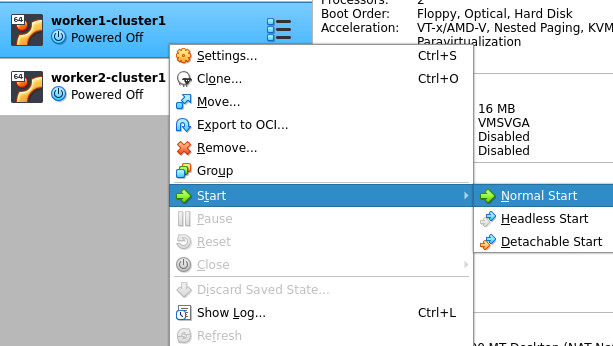


2.

3. (Clone Type) – Clone

4.



5. Open Terminal

sudo hostnamectl set-hostname worker1-cluster1

## Install Kubernetes

##### Login as root user sudo su -

##### Install Kubernetes container orchestration CRI-O

modprobe overlay

modprobe br\_netfilter

cat >>/etc/sysctl.d/kubernetes.conf<<EOF  
 net.bridge.bridge-nf-call-iptables = 1

net.bridge.bridge-nf-call-ip6tables = 1  
 net.ipv4.ip\_forward = 1

EOF

sysctl --system

export OS=xUbuntu\_22.04  
export VER=1.22  
export WEBSITE=http://download.opensuse.org  
  
echo "deb $WEBSITE/repositories/devel:/kubic:/libcontainers:/stable:/cri-o:/$VER/$OS/ /" | tee -a /etc/apt/sources.list.d/cri-0.list  
  
echo "deb $WEBSITE/repositories/devel:/kubic:/libcontainers:/stable/$OS/ /" | tee -a /etc/apt/sources.list.d/libcontainers.list

curl -L $WEBSITE/repositories/devel:/kubic:/libcontainers:/stable:/cri-o:/$VER/$OS/Release.key | apt-key add -

curl -L $WEBSITE/repositories/devel:/kubic:/libcontainers:/stable/$OS/Release.key | apt-key add -  
  
apt-get update  
apt-get install -y cri-o cri-o-runc  
  
systemctl daemon-reload  
systemctl enable crio  
systemctl start crio  
systemctl status crio

1. Add Apt repository:  
   curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | apt-key add -  
   echo "deb https://apt.kubernetes.io/ kubernetes-xenial main" > /etc/apt/sources.list.d/kubernetes.list
2. Update apt package index, install kubelet, kubeadm and kubectl, and pin their version:  
   apt-get update  
   apt-get install -y kubelet kubeadm kubectl  
   apt-mark hold kubelet kubeadm kubectl

##### Initialize Kubernetes Cluster for the Master Node Only! kubeadm init --upload-certs

(In case [ERROR FileContent--proc-sys-net-bridge-bridge-nf-call-iptables]: /proc/sys/net/bridge/bridge-nf-call-iptables does not exist)  
modprobe overlay  
modprobe br\_netfilter

1. **Setup Kubernetes config**su k8s  
   mkdir -p $HOME/.kube

**Master Node**  
 sudo cp /etc/kubernetes/admin.conf $HOME/.kube/config  
**Other Nodes:**  
 scp k8s@{{MASTER\_IP}}:/home/k8s/.kube/config $HOME/.kube/config  
  
chown $(id -u):$(id -g) $HOME/.kube/config

1. Setup Container Network Interface  
   curl https://docs.projectcalico.org/manifests/calico.yaml > ~/calico.yaml  
   kubectl apply -f ~/calico.yaml  
   rm ~/calico.yaml
2. Remove the "NoSchedule" is a default control plane taint in worker nodes.

kubectl taint nodes --all node-role.kubernetes.io/control-plane-

Verify:

kubectl describe node | grep -i taint -A1

1. Run on the **master node** a join node command:  
   kubeadm token create --print-join-command
2. Run the output join command on the **other nodes**

sudo -i su

PASTE join command: kubeadm join 10.130…

NOTE: For any errors, reload kubeadm and try joining again: kubeadm reset –f

1. Verify installation  
   kubectl cluster-info

kubectl get nodes -o wide

## Troubleshooting

sudo -i su  
kubeadm reset -f

kubeadm init phase certs all

kubeadm init phase kubeconfig all

systemctl restart kubelet