Building a Kubernetes 1.20 Cluster with Kubeadm

Introduction

https://app.linuxacademy.com/dashboard

This lab will allow you to practice the process of building a new Kubernetes cluster. You will be given a set of Linux servers, and you will have the opportunity to turn these servers into a functioning Kubernetes cluster. This will help you build the skills necessary to create your own Kubernetes clusters in the real world.

Solution

Log in to the lab server using the credentials provided:

```
ssh cloud_user@<PUBLIC_IP_ADDRESS>
```

Install Packages

1. Log into the Control Plane Node *(Note: The following steps must be performed on all three nodes.).*
2. Create configuration file for containerd:

```
3. cat <<EOF | sudo tee /etc/modules-load.d/containerd.conf
4. overlay
5. br_netfilter
```

```
EOF
```

6. Load modules:

```
7. sudo modprobe overlay
```

```
sudo modprobe br_netfilter
```

8. Set system configurations for Kubernetes networking:

```
9. cat <<EOF | sudo tee /etc/sysctl.d/99-kubernetes-cri.conf
10.  net.bridge.bridge-nf-call-iptables = 1
11.  net.ipv4.ip_forward = 1
```

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

```
EOF
```

13. Apply new settings:

```
sudo sysctl --system
```

14. Install containerd:

```
sudo apt-get update && sudo apt-get install -y containerd
```

15. Create default configuration file for containerd:

```
sudo mkdir -p /etc/containerd
```

16. Generate default containerd configuration and save to the newly created default file:

```
sudo containerd config default | sudo tee /etc/containerd/config.toml
```

17. Restart containerd to ensure new configuration file usage:

```
sudo systemctl restart containerd
```

18. Disable swap:

```
sudo swapoff -a
```

19. Disable swap on startup in `/etc/fstab`:

```
sudo sed -i '/ swap / s/^\(.*\)$/#\1/g' /etc/fstab
```

20. Install dependency packages:

```
sudo apt-get update && sudo apt-get install -y apt-transport-https curl
```

21. Download and add GPG key:

```
curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo ap
t-key add -
```

22. Add Kubernetes to repository list:

```
23.  cat <<EOF | sudo tee /etc/apt/sources.list.d/kubernetes.list
24.  deb https://apt.kubernetes.io/ kubernetes-xenial main
```

```
EOF
```

25. Update package listings:

```
sudo apt-get update
```

26. Install Kubernetes packages (Note: If you get a dpkg lock message, just wait a minute or two before trying the command again):

```
sudo apt-get install -y kubelet=1.20.1-00 kubeadm=1.20.1-00 kubectl=1.2
0.1-00
```

27. Turn off automatic updates:

```
sudo apt-mark hold kubelet kubeadm kubectl
```

28. Log into both Worker Nodes to perform previous steps.

Initialize the Cluster

1. Initialize the Kubernetes cluster on the control plane node using kubeadm *(Note: This is only performed on the Control Plane Node)*:

```
sudo kubeadm init --pod-network-cidr 192.168.0.0/16
```

2. Set kubectl access:

```
3. mkdir -p $HOME/.kube
4. sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
```

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

5. Test access to cluster:

```
kubectl version
```

Install the Calico Network Add-On

6. On the Control Plane Node, install Calico Networking:

```
kubectl apply -f https://docs.projectcalico.org/manifests/calico.yaml
```

7. Check status of Calico components:

```
kubectl get pods -n kube-system
```

Join the Worker Nodes to the Cluster

1. In the Control Plane Node, create the token and copy the kubeadm join command *(NOTE:The join command can also be found in the output from `kubeadm init` command)*:

```
kubeadm token create --print-join-command
```

2. In both Worker Nodes, paste the kubeadm join command to join the cluster:

```
sudo kubeadm join <join command from previous command>
```

3. In the Control Plane Node, view cluster status (Note: You may have to wait a few moments to allow the cluster to become ready):

```
kubectl get nodes
```

Conclusion

Congratulations — you have completed this hands-on lab!

kubeadm join 172.31.42.232:6443 --token erq2vv.twuyexezlzh73ihw \

--discovery-token-ca-cert-hash sha256:4831c8c608d8adf738ed131fb19e88eecf5f8bb7d0a74660a995c18f128ec09f

Control Plane Node

```
root@4339acb1731c:~# history
    1  cat <<EOF | sudo tee /etc/modules-load.d/containerd.conf
overlay
br_netfilter
EOF

    2  sudo modprobe overlay
    3  sudo modprobe br_netfilter
    4  cat <<EOF | sudo tee /etc/sysctl.d/99-kubernetes-cri.conf
net.bridge.bridge-nf-call-iptables = 1
net.ipv4.ip_forward = 1
net.bridge.bridge-nf-call-ip6tables = 1
EOF

    5  sudo sysctl --system
    6  sudo apt-get update && sudo apt-get install -y containerd
    7  sudo mkdir -p /etc/containerd
    8  sudo containerd config default | sudo tee /etc/containerd/config.toml
    9  sudo systemctl restart containerd
   10  sudo swapoff -a
   11  sudo sed -i '/ swap / s/^\(.*\)$/#\1/g' /etc/fstab
   12  sudo apt-get update && sudo apt-get install -y apt-transport-https curl
   13  curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add -
   14  cat <<EOF | sudo tee /etc/apt/sources.list.d/kubernetes.list
deb https://apt.kubernetes.io/ kubernetes-xenial main
EOF
```

```
  15  sudo apt-get update
  16  sudo apt-get install -y kubelet=1.20.1-00 kubeadm=1.20.1-00 kubectl=1.20.1-00
  17  sudo apt-mark hold kubelet kubeadm kubectl
  18  sudo kubeadm init --pod-network-cidr 192.168.0.0/16
  19  mkdir -p $HOME/.kube
  20  sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
  21  sudo chown $(id -u):$(id -g) $HOME/.kube/config
  22  kubectl version
  23  kubectl apply -f https://docs.projectcalico.org/manifests/calico.yaml
  24  kubectl get pods -n kube-system
  25 kubectl
  26  history
```

Workernode-1

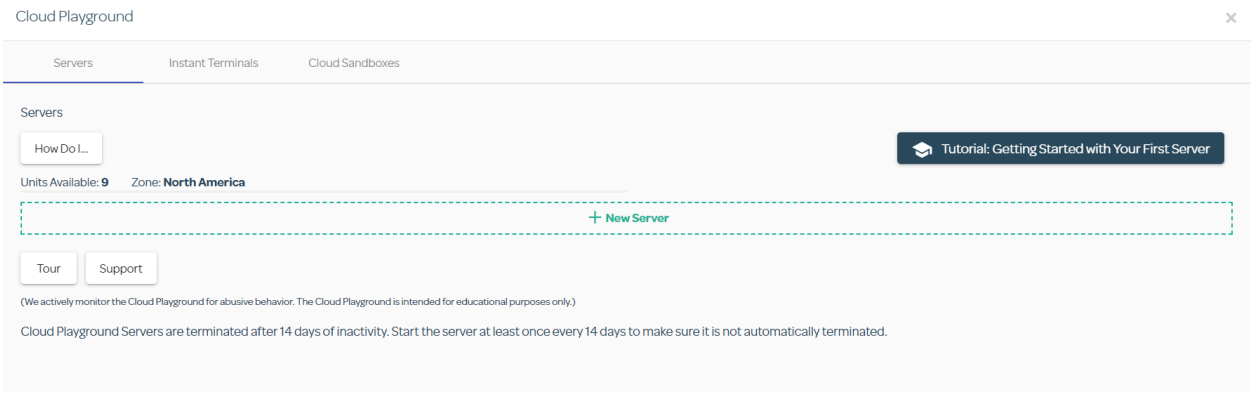
```
root@dafa9889821c:~# history
   1  cat <<EOF | sudo tee /etc/modules-load.d/containerd.conf
overlay
br_netfilter
EOF

   2  sudo modprobe overlay
   3  sudo modprobe br_netfilter
   4  cat <<EOF | sudo tee /etc/sysctl.d/99-kubernetes-cri.conf
net.bridge.bridge-nf-call-iptables = 1
net.ipv4.ip_forward = 1
net.bridge.bridge-nf-call-ip6tables = 1
EOF
```

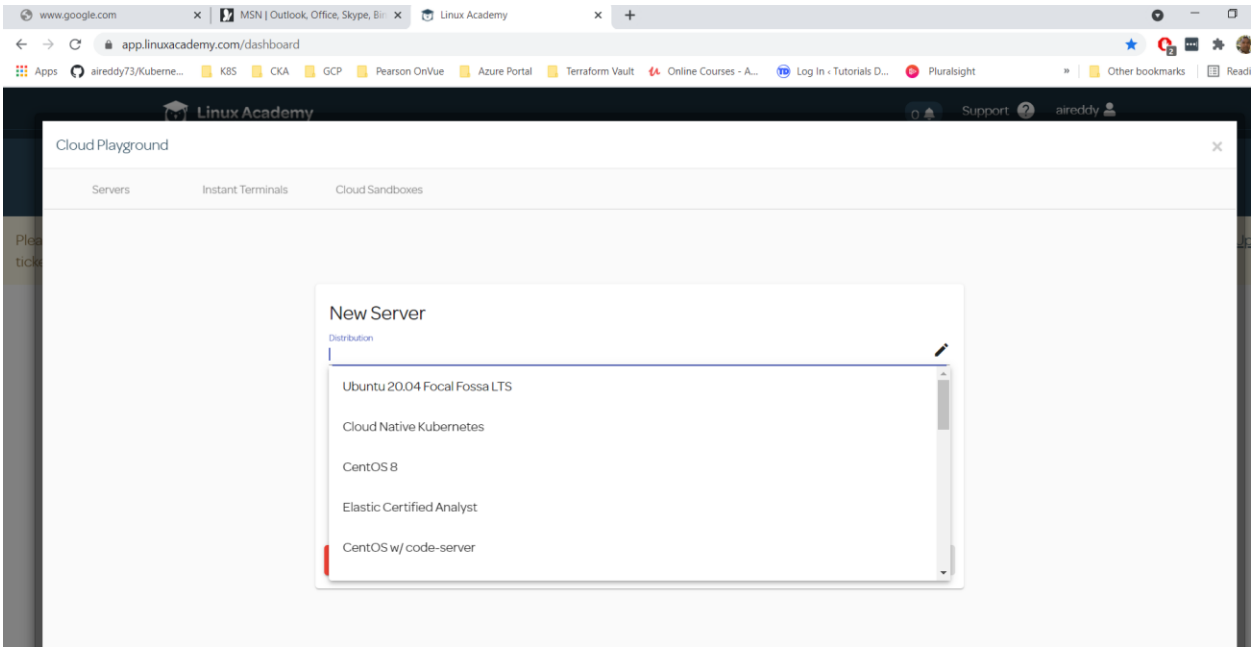
```
    5  sudo sysctl --system
    6  sudo apt-get update && sudo apt-get install -y containerd
    7  sudo mkdir -p /etc/containerd
    8  sudo containerd config default | sudo tee /etc/containerd/config.toml
    9  sudo systemctl restart containerd
   10  sudo swapoff -a
   11  sudo sed -i '/ swap / s/^\(.*\)$/#\1/g' /etc/fstab
   12  sudo apt-get update && sudo apt-get install -y apt-transport-https curl
   13  curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add -
   14  cat <<EOF | sudo tee /etc/apt/sources.list.d/kubernetes.list
deb https://apt.kubernetes.io/ kubernetes-xenial main
EOF

   15  sudo apt-get update
   16  sudo apt-get install -y kubelet=1.20.1-00 kubeadm=1.20.1-00 kubectl=1.20.1-00
   17  sudo apt-mark hold kubelet kubeadm kubectl
   18  kubeadm join 172.31.42.232:6443 --token erq2vv.twuyexezlzh73ihw     --discovery-token-ca-cert-
hash sha256:4831c8c608d8adf738ed131fb19e88eecf5f8bb7d0a74660a995c18f128ec09f
   19  kubectl
   20  history
```

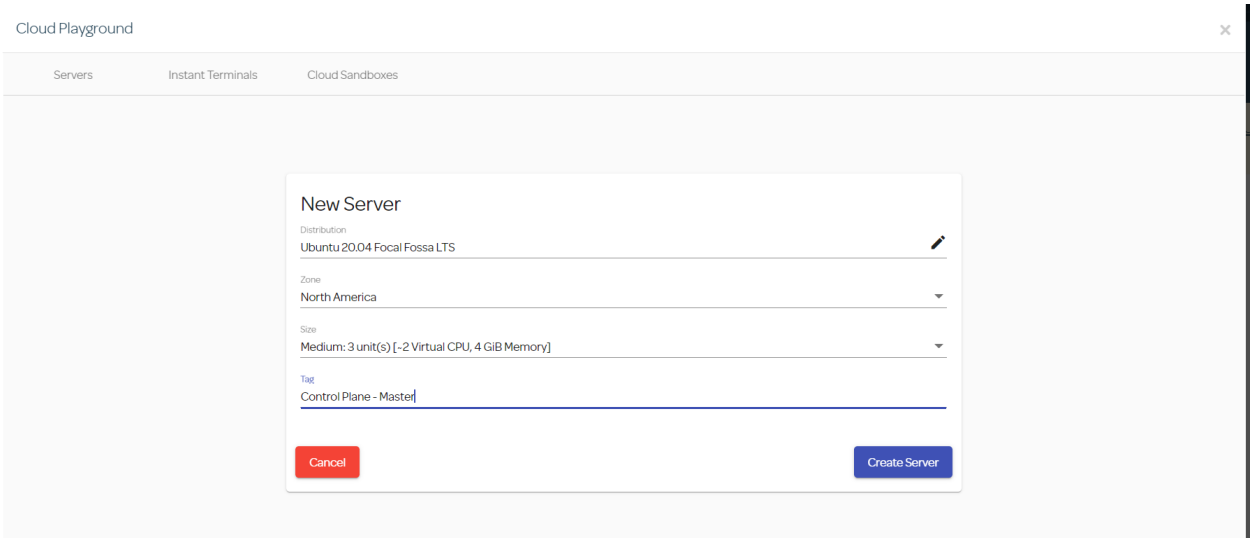
Worker Node 2

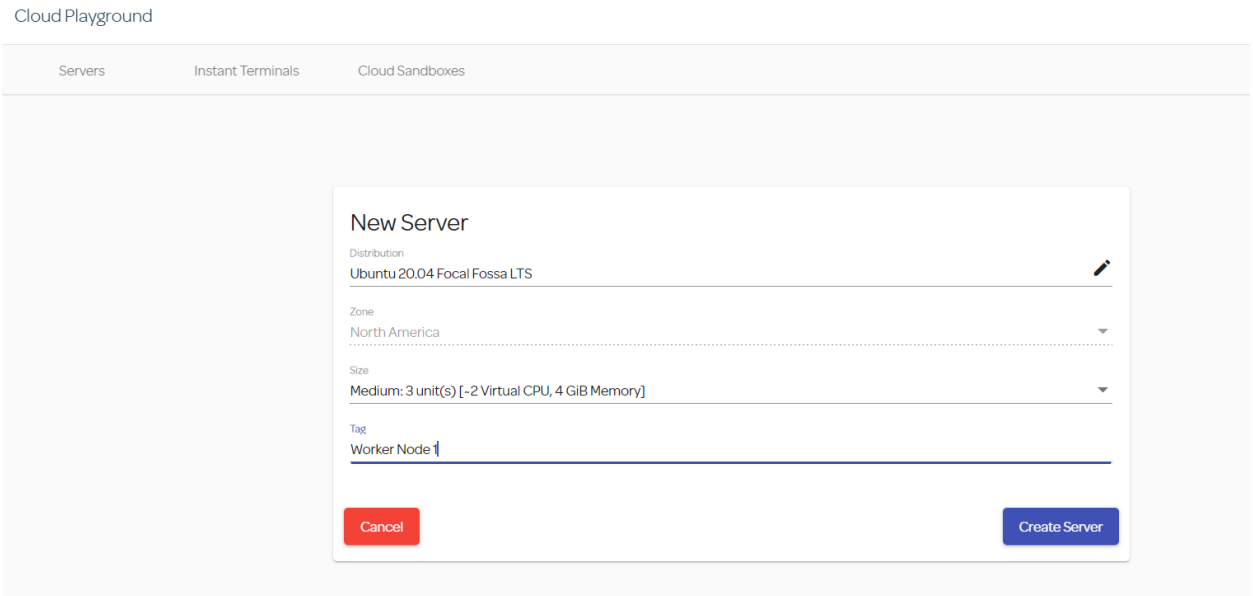
```
root@ef4a42879a1c:~# history
    1  cat <<EOF | sudo tee /etc/modules-load.d/containerd.conf
overlay
br_netfilter
```

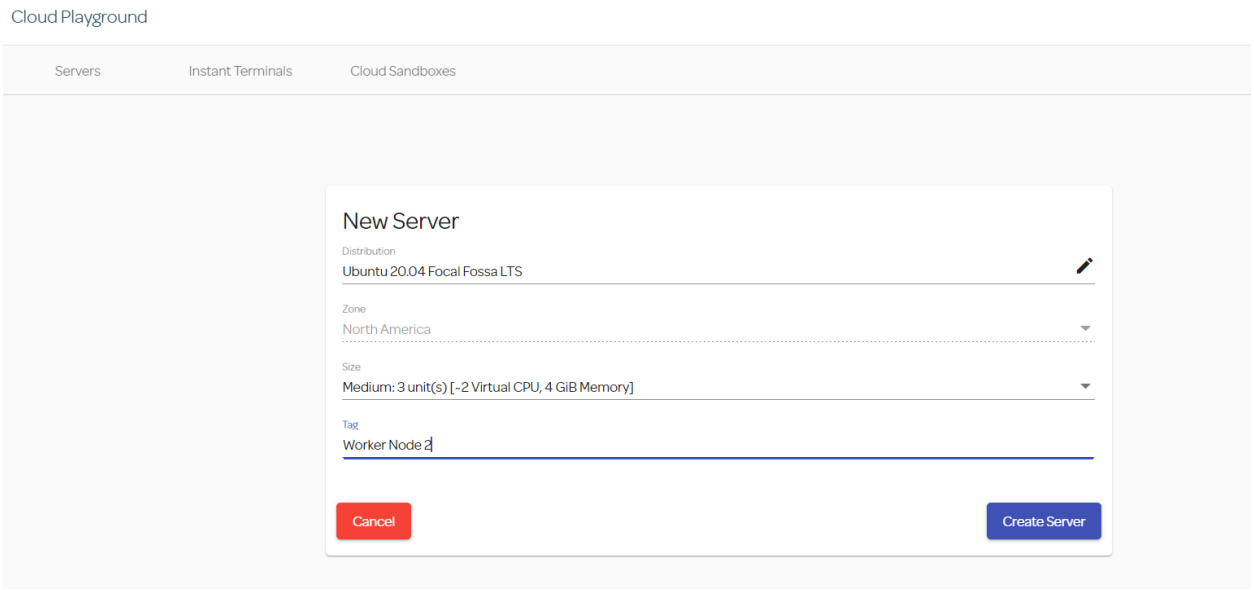
```
EOF
    2  sudo modprobe overlay
    3  sudo modprobe br_netfilter
    4  cat <<EOF | sudo tee /etc/sysctl.d/99-kubernetes-cri.conf
net.bridge.bridge-nf-call-iptables = 1
net.ipv4.ip_forward = 1
net.bridge.bridge-nf-call-ip6tables = 1
EOF
    5  sudo sysctl --system
    6  sudo apt-get update && sudo apt-get install -y containerd
    7  sudo mkdir -p /etc/containerd
    8  sudo containerd config default | sudo tee /etc/containerd/config.toml
    9  sudo systemctl restart containerd
   10  sudo swapoff -a
   11  sudo sed -i '/ swap / s/^\(.*\)$/#\1/g' /etc/fstab
   12  sudo apt-get update && sudo apt-get install -y apt-transport-https curl
   13  curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add -
   14  cat <<EOF | sudo tee /etc/apt/sources.list.d/kubernetes.list
deb https://apt.kubernetes.io/ kubernetes-xenial main
EOF
   15  sudo apt-get update
   16  sudo apt-get install -y kubelet=1.20.1-00 kubeadm=1.20.1-00 kubectl=1.20.1-00
   17  sudo apt-mark hold kubelet kubeadm kubectl
   18  kubeadm join 172.31.42.232:6443 --token erq2vv.twuyexezlzh73ihw     --discovery-token-ca-cert-
hash sha256:4831c8c608d8adf738ed131fb19e88eecf5f8bb7d0a74660a995c18f128ec09f
   19  kubectl
   20  history
root@ef4a42879a1c:~#
```

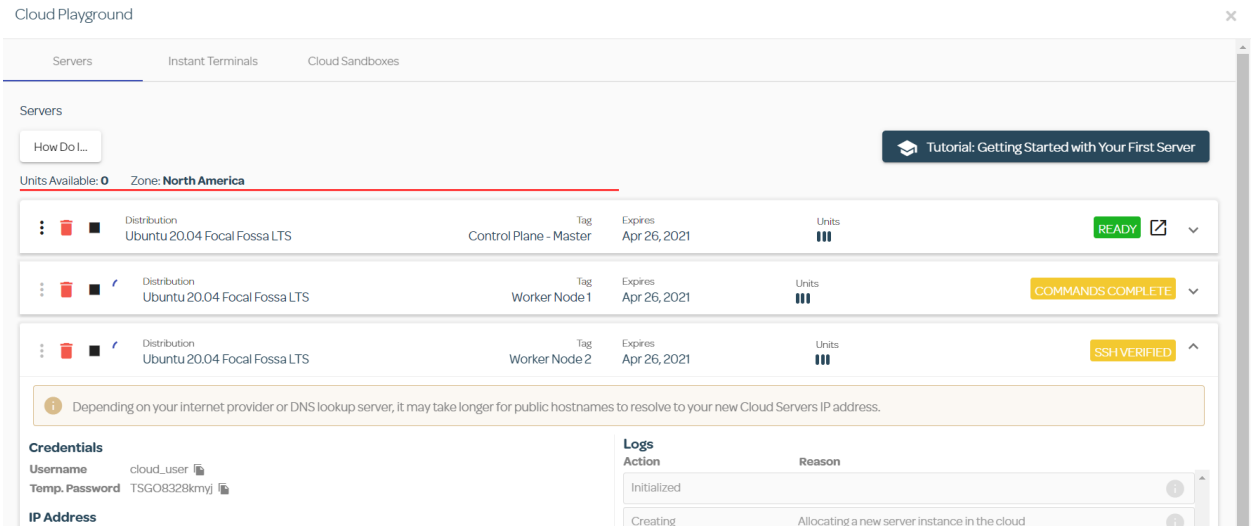

Cloud Playground
Servers
Instant Terminals
Cloud Sandboxes
Servers
How Do I...
Tutorial: Getting Started with Your First Server
Units Available: 9
Zone: North America
New Server
Tour
Support
(We actively monitor the Cloud Playground for abusive behavior. The Cloud Playground is intended for educational purposes only.)
Cloud Playground Servers are terminated after 14 days of inactivity. Start the server at least once every 14 days to make sure it is not automatically terminated.


www.google.com
MSN | Outlook, Office, Skype, Bi
Linux Academy
app.linuxacademy.com/dashboard
Apps
aireddy73/Kuberne...
K8S
CKA
GCP
Pearson OnVue
Azure Portal
Terraform Vault
Online Courses - A...
Log In ‹ Tutorials D...
Pluralsight
Other bookmarks
Linux Academy
Support
aireddy
Cloud Playground
Servers
Instant Terminals
Cloud Sandboxes
New Server
Distribution
Ubuntu 20.04 Focal Fossa LTS
Cloud Native Kubernetes
CentOS 8
Elastic Certified Analyst
CentOS w/ code-server

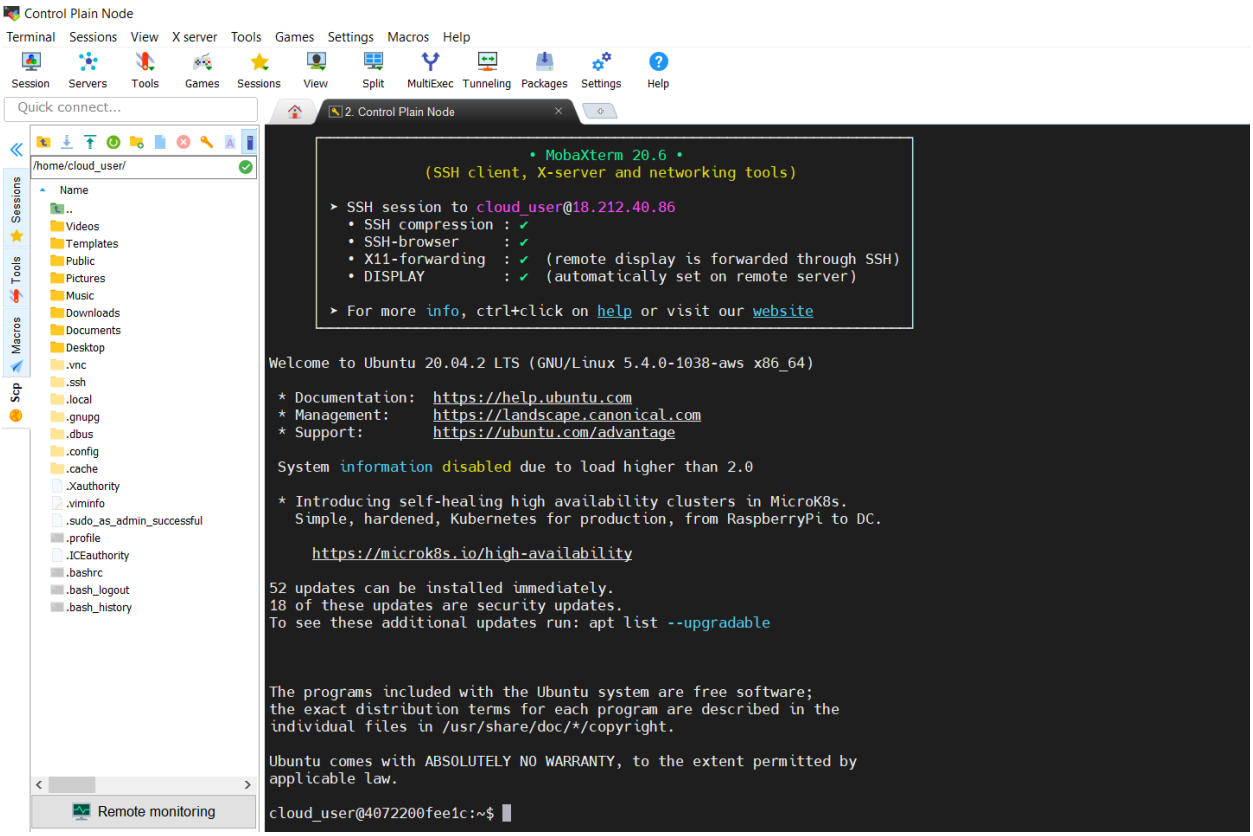

Cloud Playground
Servers
Instant Terminals
Cloud Sandboxes
New Server
Distribution
Ubuntu 20.04 Focal Fossa LTS
Zone
North America
Size
Medium: 3 unit(s) [-2 Virtual CPU, 4 GiB Memory]
Tag
Control Plane - Master
Cancel
Create Server

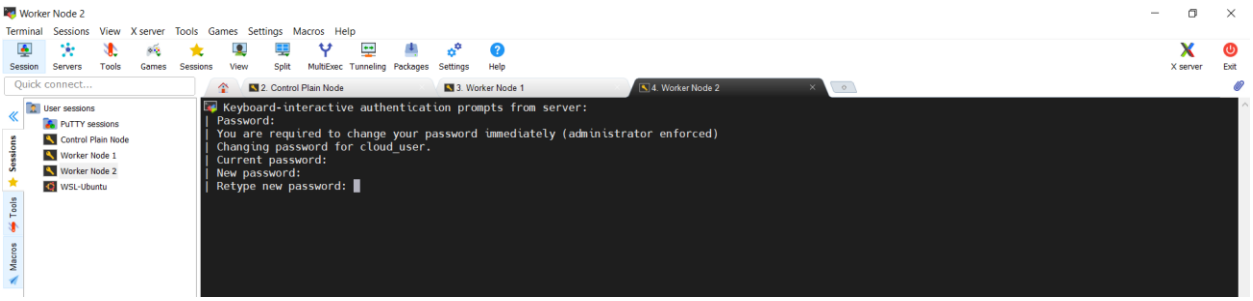

Cloud Playground
Servers
Instant Terminals
Cloud Sandboxes
New Server
Distribution
Ubuntu 20.04 Focal Fossa LTS
Zone
North America
Size
Medium: 3 unit(s) [-2 Virtual CPU, 4 GiB Memory]
Tag
Worker Node 1
Cancel
Create Server

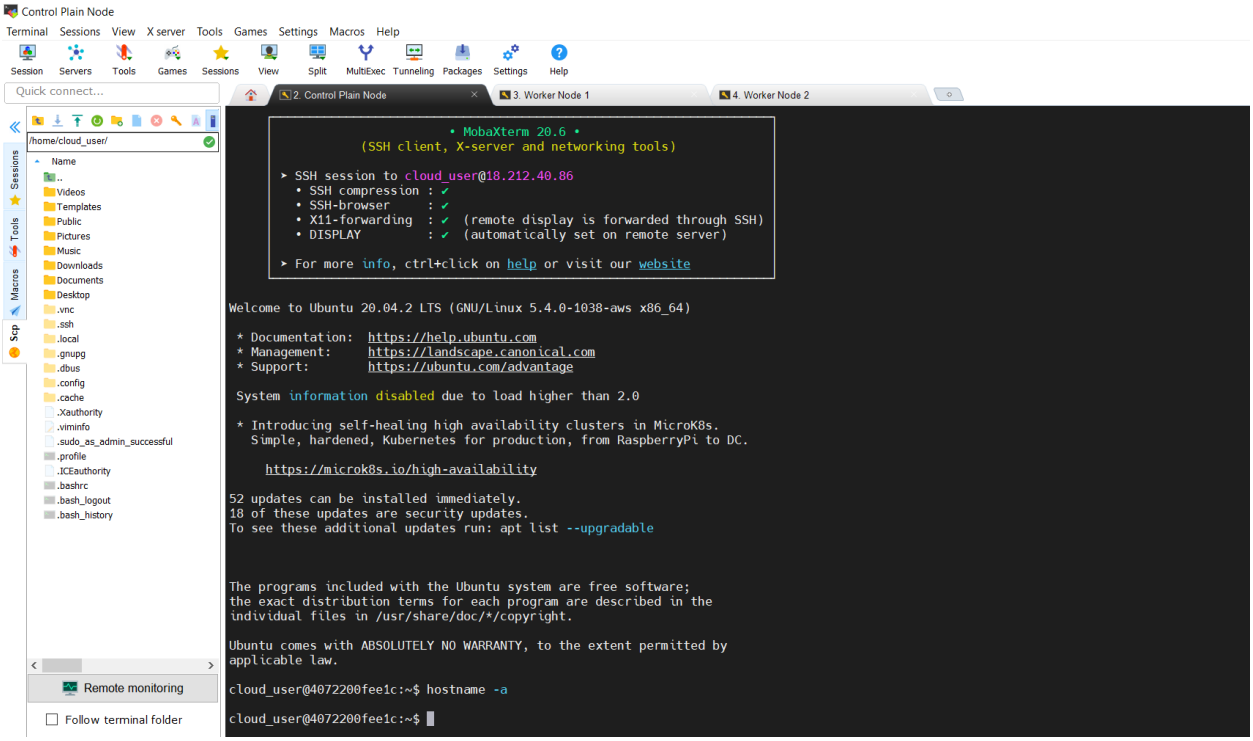

Cloud Playground
Servers
Instant Terminals
Cloud Sandboxes
New Server
Distribution
Ubuntu 20.04 Focal Fossa LTS
Zone
North America
Size
Medium: 3 unit(s) [-2 Virtual CPU, 4 GiB Memory]
Tag
Worker Node 2
Cancel
Create Server

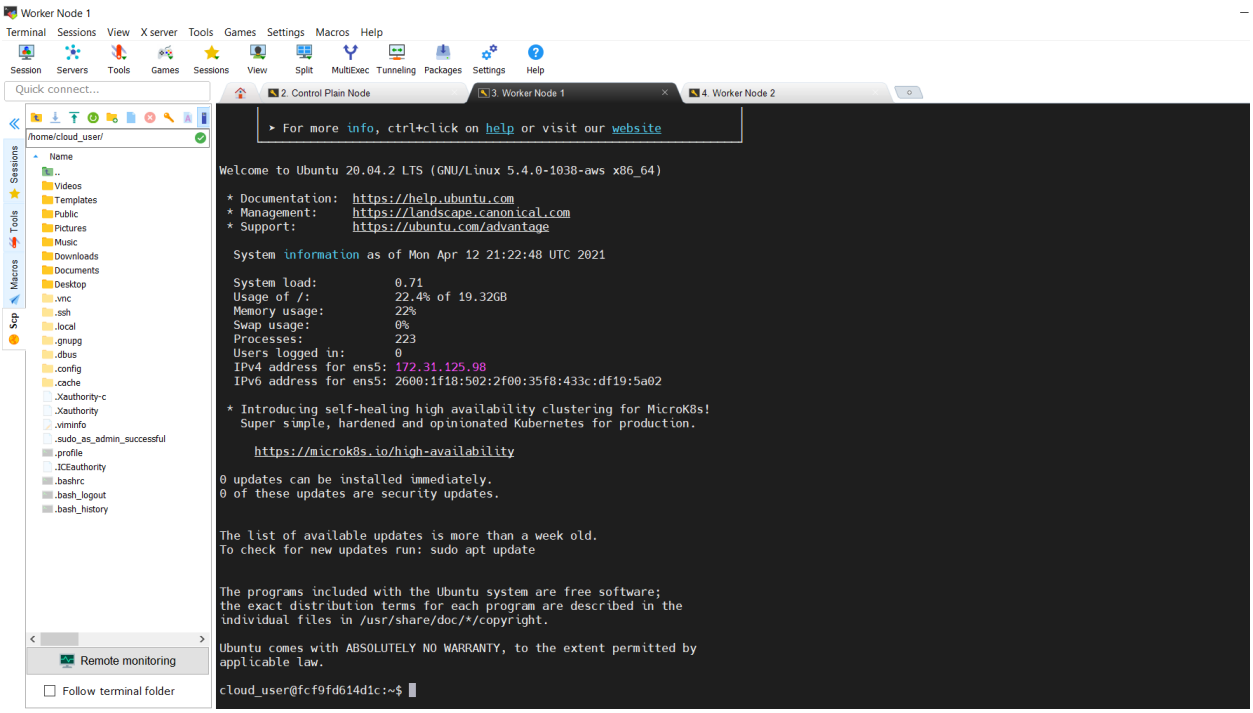

Cloud Playground
Servers
Instant Terminals
Cloud Sandboxes
Servers
How Do I...
Tutorial: Getting Started with Your First Server
Units Available: 0
Zone: North America
Distribution
Ubuntu 20.04 Focal Fossa LTS
Tag
Control Plane - Master
Expires
Apr 26, 2021
Units
READY
Distribution
Ubuntu 20.04 Focal Fossa LTS
Tag
Worker Node 1
Expires
Apr 26, 2021
Units
COMMANDS COMPLETE
Distribution
Ubuntu 20.04 Focal Fossa LTS
Tag
Worker Node 2
Expires
Apr 26, 2021
Units
SSH VERIFIED
Depending on your internet provider or DNS lookup server, it may take longer for public hostnames to resolve to your new Cloud Servers IP address.
Credentials
Username
cloud_user
Temp. Password
TSGO8328kmyj
IP Address
Logs
Action
Reason
Initialized
Creating
Allocating a new server instance in the cloud


Control Plain Node
Terminal
Sessions
View
X server
Tools
Games
Settings
Macros
Help
Session
Servers
Tools
Games
Sessions
View
Split
MultiExec
Tunneling
Packages
Settings
Help
X server
Exit
Quick connect...
2 Control Plain Node
User sessions
PuTTY sessions
Control Plain Node
Worker Node 1
Worker Node 2
WSL-Ubuntu
Sessions
Keyboard-interactive authentication prompts from server:
Password:


Control Plain Node
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect...
2. Control Plain Node
/home/cloud_user/
Name
..
Videos
Templates
Public
Pictures
Music
Downloads
Documents
Desktop
.vnc
.ssh
.local
.gnupg
.dbus
.config
.cache
.Xauthority
.viminfo
.sudo_as_admin_successful
.profile
.ICEauthority
.bashrc
.bash_logout
.bash_history
Remote monitoring
• MobaXterm 20.6 •
(SSH client, X-server and networking tools)
➤ SSH session to cloud_user@18.212.40.86
• SSH compression : ✔
• SSH-browser : ✔
• X11-forwarding : ✔ (remote display is forwarded through SSH)
• DISPLAY : ✔ (automatically set on remote server)
➤ For more info, ctrl+click on help or visit our website
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.4.0-1038-aws x86_64)
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/advantage
System information disabled due to load higher than 2.0
* Introducing self-healing high availability clusters in MicroK8s.
Simple, hardened, Kubernetes for production, from RaspberryPi to DC.
https://microk8s.io/high-availability
52 updates can be installed immediately.
18 of these updates are security updates.
To see these additional updates run: apt list --upgradable
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
cloud_user@4072200fee1c:~$


Worker Node 2
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help X server Exit
Quick connect...
2. Control Plain Node
3. Worker Node 1
4. Worker Node 2
User sessions
PuTTY sessions
Control Plain Node
Worker Node 1
Worker Node 2
WSL-Ubuntu
Keyboard-interactive authentication prompts from server:
| Password:
| You are required to change your password immediately (administrator enforced)
| Changing password for cloud_user.
| Current password:
| New password:
| Retype new password:


Control Plain Node
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect...
2. Control Plain Node
3. Worker Node 1
4. Worker Node 2
/home/cloud_user/
Name
Videos
Templates
Public
Pictures
Music
Downloads
Documents
Desktop
.vnc
.ssh
.local
.gnupg
.dbus
.config
.cache
.Xauthority
.viminfo
.sudo_as_admin_successful
.profile
.ICEauthority
.bashrc
.bash_logout
.bash_history
Remote monitoring
Follow terminal folder
• MobaXterm 20.6 •
(SSH client, X-server and networking tools)
➤ SSH session to cloud_user@18.212.40.86
• SSH compression : ✓
• SSH-browser : ✓
• X11-forwarding : ✓ (remote display is forwarded through SSH)
• DISPLAY : ✓ (automatically set on remote server)
➤ For more info, ctrl+click on help or visit our website
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.4.0-1038-aws x86_64)
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/advantage
System information disabled due to load higher than 2.0
* Introducing self-healing high availability clusters in MicroK8s.
Simple, hardened, Kubernetes for production, from RaspberryPi to DC.
https://microk8s.io/high-availability
52 updates can be installed immediately.
18 of these updates are security updates.
To see these additional updates run: apt list --upgradable
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
cloud_user@4072200fee1c:~$ hostname -a
cloud_user@4072200fee1c:~$


Worker Node 1
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect...
2. Control Plain Node
3. Worker Node 1
4. Worker Node 2
/home/cloud_user/
Name
Videos
Templates
Public
Pictures
Music
Downloads
Documents
Desktop
.vnc
.ssh
.local
.gnupg
.dbus
.config
.cache
.Xauthority-c
.Xauthority
.viminfo
.sudo_as_admin_successful
.profile
.ICEauthority
.bashrc
.bash_logout
.bash_history
Remote monitoring
Follow terminal folder
➤ For more info, ctrl+click on help or visit our website
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.4.0-1038-aws x86_64)
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/advantage
System information as of Mon Apr 12 21:22:48 UTC 2021
System load: 0.71
Usage of /: 22.4% of 19.32GB
Memory usage: 22%
Swap usage: 0%
Processes: 223
Users logged in: 0
IPv4 address for ens5: 172.31.125.98
IPv6 address for ens5: 2600:1f18:502:2f00:35f8:433c:df19:5a02
* Introducing self-healing high availability clustering for MicroK8s!
Super simple, hardened and opinionated Kubernetes for production.
https://microk8s.io/high-availability
0 updates can be installed immediately.
0 of these updates are security updates.
The list of available updates is more than a week old.
To check for new updates run: sudo apt update
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
cloud_user@fcf9fd614d1c:~$

Cluster setup with Kubeadm

Try to create Kubeadm cluster with below mentioned links instead of copy paste the commands. There is no need to disable swap but there is no harm in disabling swap (These commands you cannot find in documentation).

1. Disable swap:

```
sudo swapoff -a
```

2. Disable swap on startup in `/etc/fstab`:

```
sudo sed -i '/ swap / s/^\(.*\)$/#\1/g' /etc/fstab
```

https://kubernetes.io/docs/setup/production-environment/tools/kubeadm/install-kubeadm/#letting-iptables-see-bridged-traffic

https://kubernetes.io/docs/setup/production-environment/container-runtimes/#container-runtimes

https://kubernetes.io/docs/setup/production-environment/tools/kubeadm/install-kubeadm/#installing-kubeadm-kubelet-and-kubectl

kubeadm init --pod-network-cidr 192.168.0.0/16

https://kubernetes.io/docs/setup/production-environment/tools/kubeadm/create-cluster-kubeadm/#initializing-your-control-plane-node

https://kubernetes.io/docs/setup/production-environment/tools/kubeadm/high-availability/

In this example we are using Weave Net:

```
kubectl apply -f "https://cloud.weave.works/k8s/net?k8s-version=$(kubectl version
| base64 | tr -d '\n')"
```

https://kubernetes.io/docs/reference/setup-tools/kubeadm/kubeadm-token/#cmd-token-create

https://kubernetes.io/docs/reference/setup-tools/kubeadm/kubeadm-join/

Cluster upgrade

https://kubernetes.io/docs/tasks/administer-cluster/kubeadm/kubeadm-upgrade/