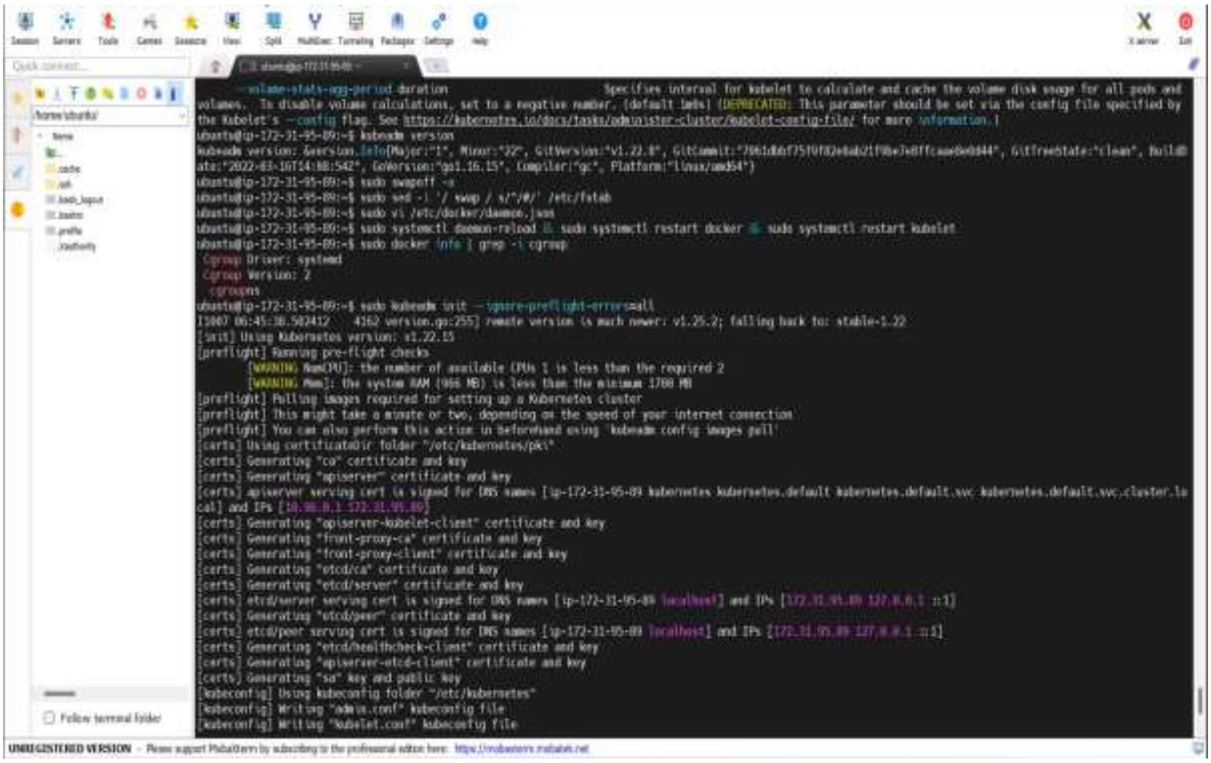
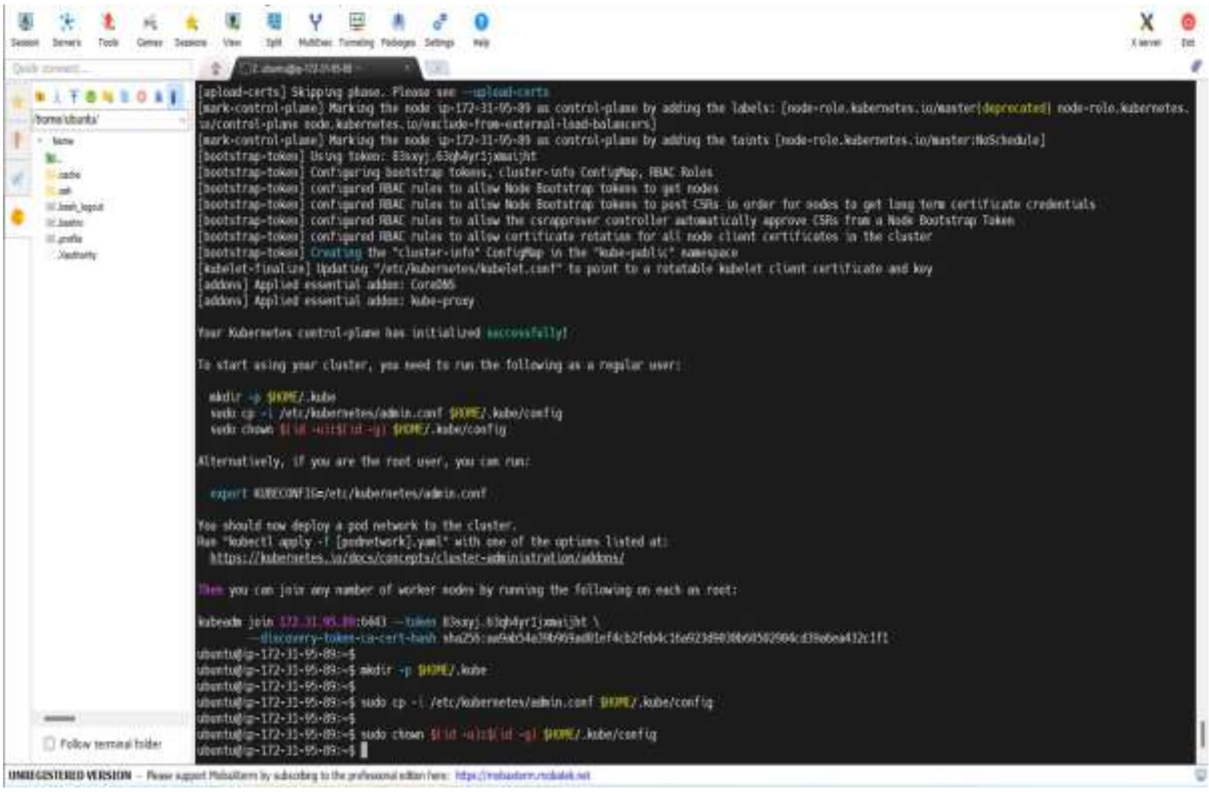
C group install



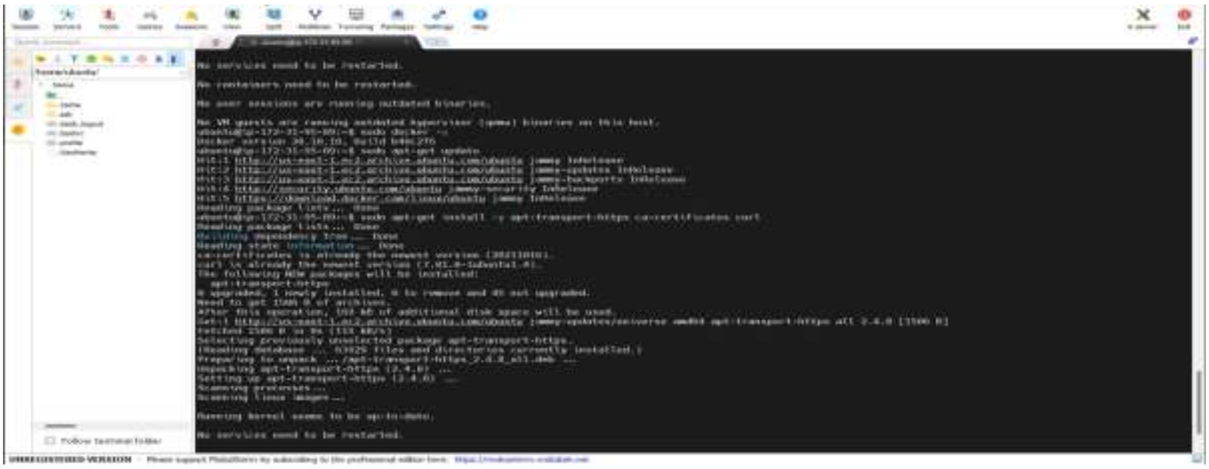
```
      --volume-stats-agg-period duration                    Specifies interval for kubelet to calculate and cache the volume disk usage for all pods and
volumes.  To disable volume calculations, set to a negative number. (default 1m0s) (DEPRECATED: This parameter should be set via the config file specified by
the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)
ubuntu@ip-172-31-95-89:~$ kubeadm version
kubeadm version: &version.Info{Major:"1", Minor:"22", GitVersion:"v1.22.8", GitCommit:"7061dbbf75f9f82e8ab21f9be7e8ffcaae8e0d44", GitTreeState:"clean", BuildD
ate:"2022-03-16T14:08:54Z", GoVersion:"go1.16.15", Compiler:"gc", Platform:"linux/amd64"}
ubuntu@ip-172-31-95-89:~$ sudo swapoff -a
ubuntu@ip-172-31-95-89:~$ sudo sed -i '/ swap / s/^/#/' /etc/fstab
ubuntu@ip-172-31-95-89:~$ sudo vi /etc/docker/daemon.json
ubuntu@ip-172-31-95-89:~$ sudo systemctl daemon-reload && sudo systemctl restart docker && sudo systemctl restart kubelet
ubuntu@ip-172-31-95-89:~$ sudo docker info | grep -i cgroup
 Cgroup Driver: systemd
 Cgroup Version: 2
  cgroupns
ubuntu@ip-172-31-95-89:~$ sudo kubeadm init --ignore-preflight-errors=all
I1007 06:45:38.502412    4162 version.go:255] remote version is much newer: v1.25.2; falling back to: stable-1.22
[init] Using Kubernetes version: v1.22.15
[preflight] Running pre-flight checks
        [WARNING NumCPU]: the number of available CPUs 1 is less than the required 2
        [WARNING Mem]: the system RAM (966 MB) is less than the minimum 1700 MB
[preflight] Pulling images required for setting up a Kubernetes cluster
[preflight] This might take a minute or two, depending on the speed of your internet connection
[preflight] You can also perform this action in beforehand using 'kubeadm config images pull'
[certs] Using certificateDir folder "/etc/kubernetes/pki"
[certs] Generating "ca" certificate and key
[certs] Generating "apiserver" certificate and key
[certs] apiserver serving cert is signed for DNS names [ip-172-31-95-89 kubernetes kubernetes.default kubernetes.default.svc kubernetes.default.svc.cluster.lo
cal] and IPs [10.96.0.1 172.31.95.89]
[certs] Generating "apiserver-kubelet-client" certificate and key
[certs] Generating "front-proxy-ca" certificate and key
[certs] Generating "front-proxy-client" certificate and key
[certs] Generating "etcd/ca" certificate and key
[certs] Generating "etcd/server" certificate and key
[certs] etcd/server serving cert is signed for DNS names [ip-172-31-95-89 localhost] and IPs [172.31.95.89 127.0.0.1 ::1]
[certs] Generating "etcd/peer" certificate and key
[certs] etcd/peer serving cert is signed for DNS names [ip-172-31-95-89 localhost] and IPs [172.31.95.89 127.0.0.1 ::1]
[certs] Generating "etcd/healthcheck-client" certificate and key
[certs] Generating "apiserver-etcd-client" certificate and key
[certs] Generating "sa" key and public key
[kubeconfig] Using kubeconfig folder "/etc/kubernetes"
[kubeconfig] Writing "admin.conf" kubeconfig file
[kubeconfig] Writing "kubelet.conf" kubeconfig file
```

Cluster creation



```
[upload-certs] Skipping phase. Please see --upload-certs
[mark-control-plane] Marking the node ip-172-31-95-89 as control-plane by adding the labels: [node-role.kubernetes.io/master(deprecated) node-role.kubernetes.
io/control-plane node.kubernetes.io/exclude-from-external-load-balancers]
[mark-control-plane] Marking the node ip-172-31-95-89 as control-plane by adding the taints [node-role.kubernetes.io/master:NoSchedule]
[bootstrap-token] Using token: 83sxyj.63qh4yr1jxmaijht
[bootstrap-token] Configuring bootstrap tokens, cluster-info ConfigMap, RBAC Roles
[bootstrap-token] configured RBAC rules to allow Node Bootstrap tokens to get nodes
[bootstrap-token] configured RBAC rules to allow Node Bootstrap tokens to post CSRs in order for nodes to get long term certificate credentials
[bootstrap-token] configured RBAC rules to allow the csrapprover controller automatically approve CSRs from a Node Bootstrap Token
[bootstrap-token] configured RBAC rules to allow certificate rotation for all node client certificates in the cluster
[bootstrap-token] Creating the "cluster-info" ConfigMap in the "kube-public" namespace
[kubelet-finalize] Updating "/etc/kubernetes/kubelet.conf" to point to a rotatable kubelet client certificate and key
[addons] Applied essential addon: CoreDNS
[addons] Applied essential addon: kube-proxy

Your Kubernetes control-plane has initialized successfully!

To start using your cluster, you need to run the following as a regular user:

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

Alternatively, if you are the root user, you can run:

  export KUBECONFIG=/etc/kubernetes/admin.conf

You should now deploy a pod network to the cluster.
Run "kubectl apply -f [podnetwork].yaml" with one of the options listed at:
  https://kubernetes.io/docs/concepts/cluster-administration/addons/

Then you can join any number of worker nodes by running the following on each as root:

kubeadm join 172.31.95.89:6443 --token 83sxyj.63qh4yr1jxmaijht \
        --discovery-token-ca-cert-hash sha256:aa9ab54a39b969ad81ef4cb2feb4c16a923d9030b60502904cd39a6ea432c1f1
ubuntu@ip-172-31-95-89:~$
ubuntu@ip-172-31-95-89:~$ mkdir -p $HOME/.kube
ubuntu@ip-172-31-95-89:~$
ubuntu@ip-172-31-95-89:~$ sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
ubuntu@ip-172-31-95-89:~$
ubuntu@ip-172-31-95-89:~$ sudo chown $(id -u):$(id -g) $HOME/.kube/config
ubuntu@ip-172-31-95-89:~$ █
```

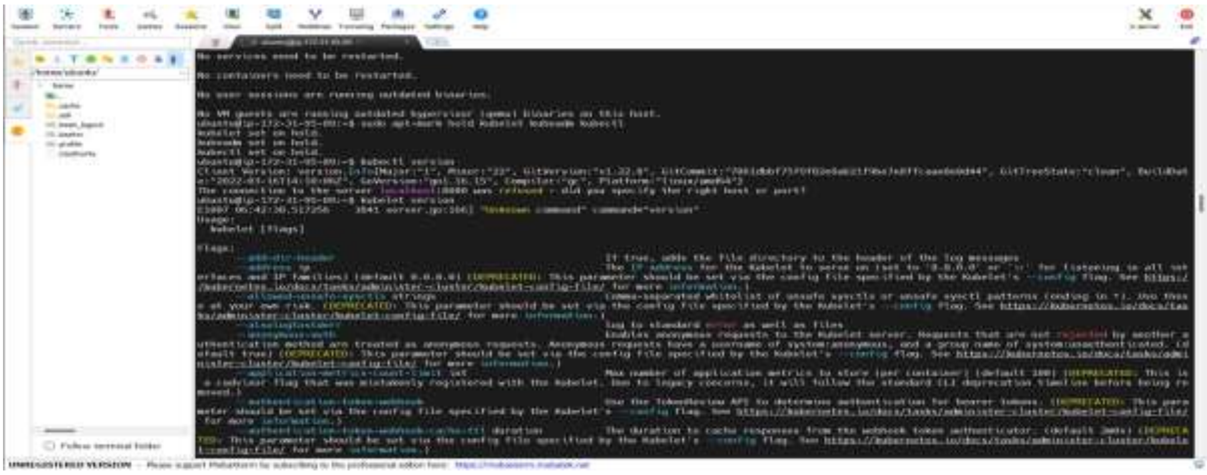
Docker version



```
No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-95-89:~$ sudo docker -v
Docker version 20.10.10, build [illegible]
ubuntu@ip-172-31-95-89:~$ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:5 https://download.docker.com/linux/ubuntu jammy InRelease
Reading package lists... Done
ubuntu@ip-172-31-95-89:~$ sudo apt-get install -y apt-transport-https ca-certificates curl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20211016).
curl is already the newest version (7.81.0-1ubuntu1.4).
The following NEW packages will be installed:
  apt-transport-https
0 upgraded, 1 newly installed, 0 to remove and 45 not upgraded.
Need to get 1506 B of archives.
After this operation, 169 kB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 apt-transport-https all 2.4.8 [1506 B]
Fetched 1506 B in 0s (113 kB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 63925 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_2.4.8_all.deb ...
Unpacking apt-transport-https (2.4.8) ...
Setting up apt-transport-https (2.4.8) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

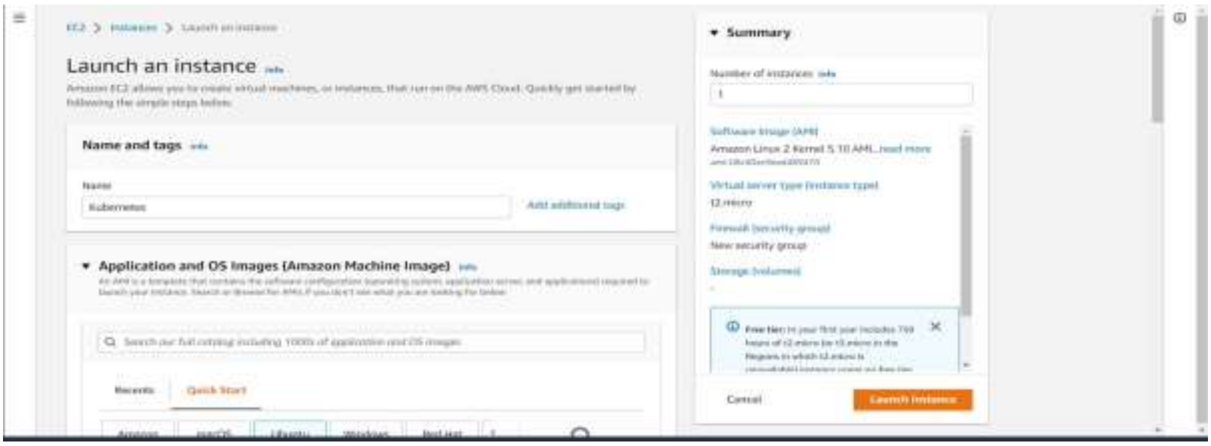
Kubectl install



```
No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-95-89:~$ sudo apt-mark hold kubelet kubeadm kubectl
kubelet set on hold.
kubeadm set on hold.
kubectl set on hold.
ubuntu@ip-172-31-95-89:~$ kubectl version
Client Version: version.Info{Major:"1", Minor:"22", GitVersion:"v1.22.8", GitCommit:"7061dbbf75f9f82e8ab21f9be7e8ffcaaeb0d44", GitTreeState:"clean", BuildDat
e:"2022-03-16T14:10:06Z", GoVersion:"go1.16.15", Compiler:"gc", Platform:"linux/amd64"}
The connection to the server localhost:8080 was refused - did you specify the right host or port?
ubuntu@ip-172-31-95-89:~$ kubelet version
E1007 06:42:30.517256    3841 server.go:166] "unknown command" command="version"
Usage:
  kubelet [flags]

Flags:
      --add-dir-header                                     If true, adds the file directory to the header of the log messages
      --address ip                                         The IP address for the Kubelet to serve on (set to '0.0.0.0' or '::' for listening in all int
erfaces and IP families) (default 0.0.0.0) (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://
/kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)
      --allowed-unsafe-sysctls strings                     Comma-separated whitelist of unsafe sysctls or unsafe sysctl patterns (ending in *). Use thes
e at your own risk. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tas
ks/administer-cluster/kubelet-config-file/ for more information.)
      --alsologtostderr                                    log to standard error as well as files
      --anonymous-auth                                     Enables anonymous requests to the Kubelet server. Requests that are not rejected by another a
uthentication method are treated as anonymous requests. Anonymous requests have a username of system:anonymous, and a group name of system:unauthenticated. (d
efault true) (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/admi
nister-cluster/kubelet-config-file/ for more information.)
      --application-metrics-count-limit int                Max number of application metrics to store (per container) (default 100) (DEPRECATED: This is
 a cadvisor flag that was mistakenly registered with the Kubelet. Due to legacy concerns, it will follow the standard CLI deprecation timeline before being re
moved.)
      --authentication-token-webhook                       Use the TokenReview API to determine authentication for bearer tokens. (DEPRECATED: This para
meter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/
 for more information.)
      --authentication-token-webhook-cache-ttl duration    The duration to cache responses from the webhook token authenticator. (default 2m0s) (DEPRECA
TED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubele
t-config-file/ for more information.)
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

Kubernetes Instance



EC2 > Instances > Launch an instance

Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name: Kubernetes

Add additional tags

Application and OS Images (Amazon Machine Image) Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below.

Search our full catalog including 1000s of application and OS images

Recents | Quick Start

Amazon | macOS | Ubuntu | Windows | Red Hat

Summary

Number of instances Info: 1

Software Image (AMI): Amazon Linux 2 Kernel 5.10 AMI...read more
ami-[illegible]

Virtual server type (instance type): t2.micro

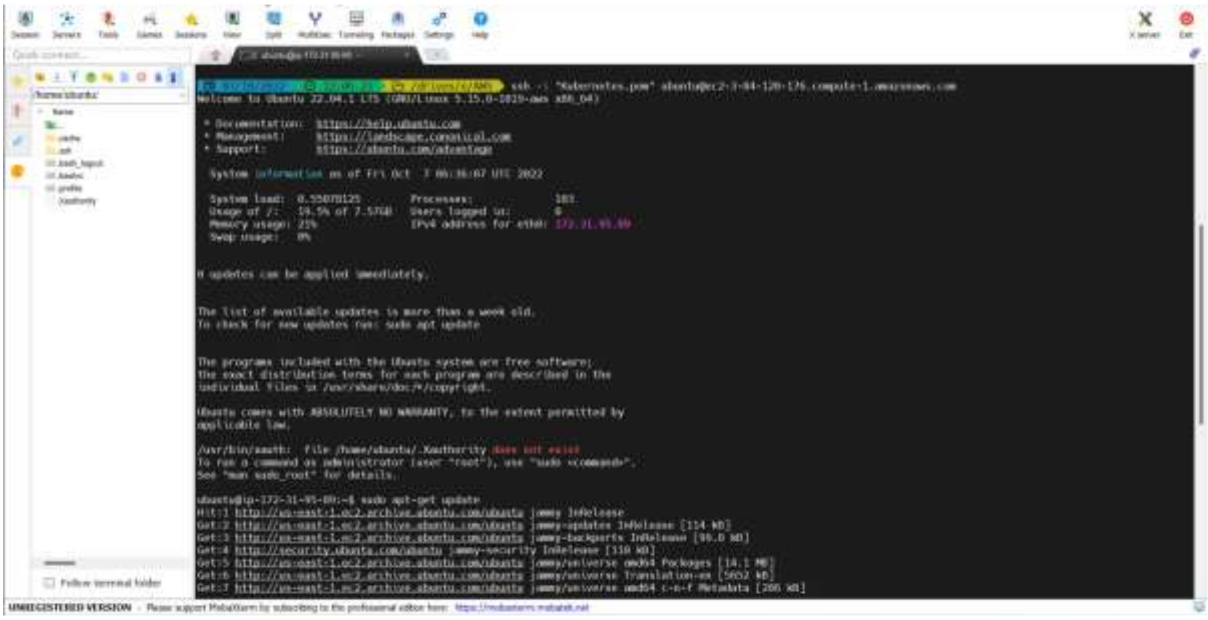
Firewall (security group): New security group

Storage (volumes): -

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier

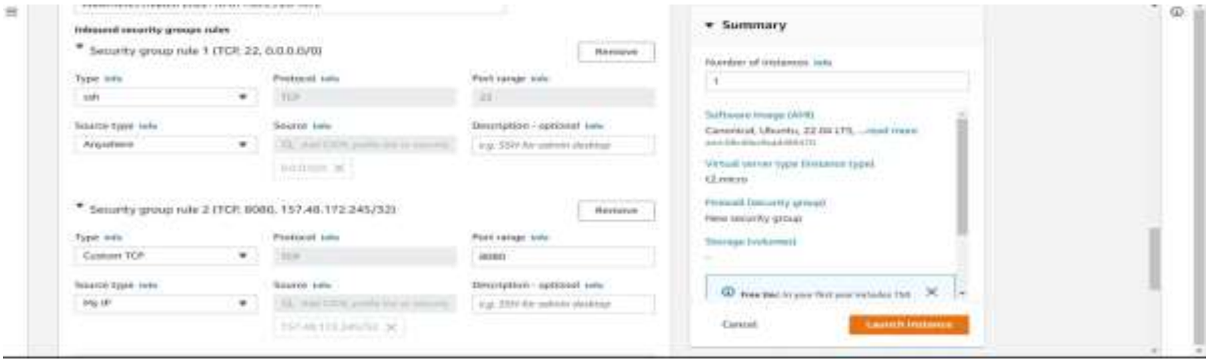
Cancel | Launch instance

Moba X-term



```
ssh -i "Kubernetes.pem" ubuntu@ec2-3-84-120-176.compute-1.amazonaws.com
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-1019-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Fri Oct  7 06:36:07 UTC 2022

  System load:  0.55078125        Processes:             103
  Usage of /:   19.5% of 7.57GB   Users logged in:       0
  Memory usage: 23%               IPv4 address for eth0: 172.31.95.89
  Swap usage:   0%


0 updates can be applied immediately.


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.

/usr/bin/xauth:  file /home/ubuntu/.Xauthority does not exist
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-95-89:~$ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [114 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [99.8 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]
```

Port number



Inbound security groups rules

Security group rule 1 (TCP, 22, 0.0.0.0/0)

Type: ssh
Protocol: TCP
Port range: 22
Source type: Anywhere
Source: 0.0.0.0/0
Description - optional: e.g. SSH for admin desktop

Security group rule 2 (TCP, 8080, 157.48.172.245/32)

Type: Custom TCP
Protocol: TCP
Port range: 8080
Source type: My IP
Source: 157.48.172.245/32
Description - optional: e.g. SSH for admin desktop

Summary

Number of instances: 1

Software Image (AMI): Canonical, Ubuntu, 22.04 LTS, ...read more

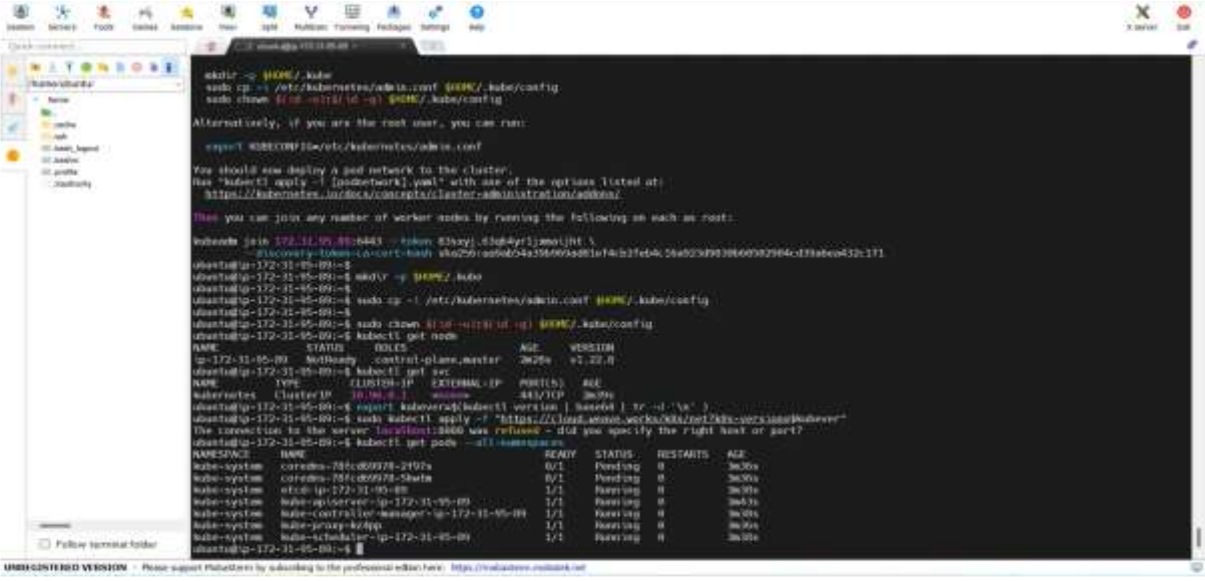
Virtual server type (instance type): t2.micro

Firewall (security group): New security group

Storage (volumes)

Remove
Remove
Info
CIDR, prefix list or security group
Free tier: In your first year includes 750
Cancel
Launch instance

Validate running pods



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

Alternatively, if you are the root user, you can run:

  export KUBECONFIG=/etc/kubernetes/admin.conf

You should now deploy a pod network to the cluster.
Run "kubectl apply -f [podnetwork].yaml" with one of the options listed at:
  https://kubernetes.io/docs/concepts/cluster-administration/addons/

Then you can join any number of worker nodes by running the following on each as root:

kubeadm join 172.31.95.89:6443 --token 83xxyj.63gb4yrljmmoljht \
        --discovery-token-ca-cert-hash sha256:aa6ab54a398993a081ef4cb2feb4c56a923d9839860938298 4cd39abea432c171
ubuntu@ip-172-31-95-89:~$
ubuntu@ip-172-31-95-89:~$ mkdir -p $HOME/.kube
ubuntu@ip-172-31-95-89:~$
ubuntu@ip-172-31-95-89:~$ sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
ubuntu@ip-172-31-95-89:~$
ubuntu@ip-172-31-95-89:~$ sudo chown $(id -u):$(id -g) $HOME/.kube/config
ubuntu@ip-172-31-95-89:~$ kubectl get node
NAME              STATUS     ROLES                  AGE     VERSION
ip-172-31-95-89   NotReady   control-plane,master   3m28s   v1.22.0
ubuntu@ip-172-31-95-89:~$ kubectl get svc
NAME         TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes   ClusterIP   10.96.0.1    <none>        443/TCP   3m39s
ubuntu@ip-172-31-95-89:~$ export kubever=$(kubectl version | base64 | tr -d '\n' )
ubuntu@ip-172-31-95-89:~$ sudo kubectl apply -f "https://cloud.weave.works/k8s/net?k8s-version=$kubever"
The connection to the server localhost:8080 was refused - did you specify the right host or port?
ubuntu@ip-172-31-95-89:~$ kubectl get pods --all-namespaces
NAMESPACE     NAME                                      READY   STATUS    RESTARTS   AGE
kube-system   coredns-78fcd69978-2f92s                  0/1     Pending   0          3m36s
kube-system   coredns-78fcd69978-5hwtm                  0/1     Pending   0          3m36s
kube-system   etcd-ip-172-31-95-89                      1/1     Running   0          3m39s
kube-system   kube-apiserver-ip-172-31-95-89            1/1     Running   0          3m43s
kube-system   kube-controller-manager-ip-172-31-95-89   1/1     Running   0          3m39s
kube-system   kube-proxy-kz4pp                          1/1     Running   0          3m36s
kube-system   kube-scheduler-ip-172-31-95-89            1/1     Running   0          3m39s
ubuntu@ip-172-31-95-89:~$
```