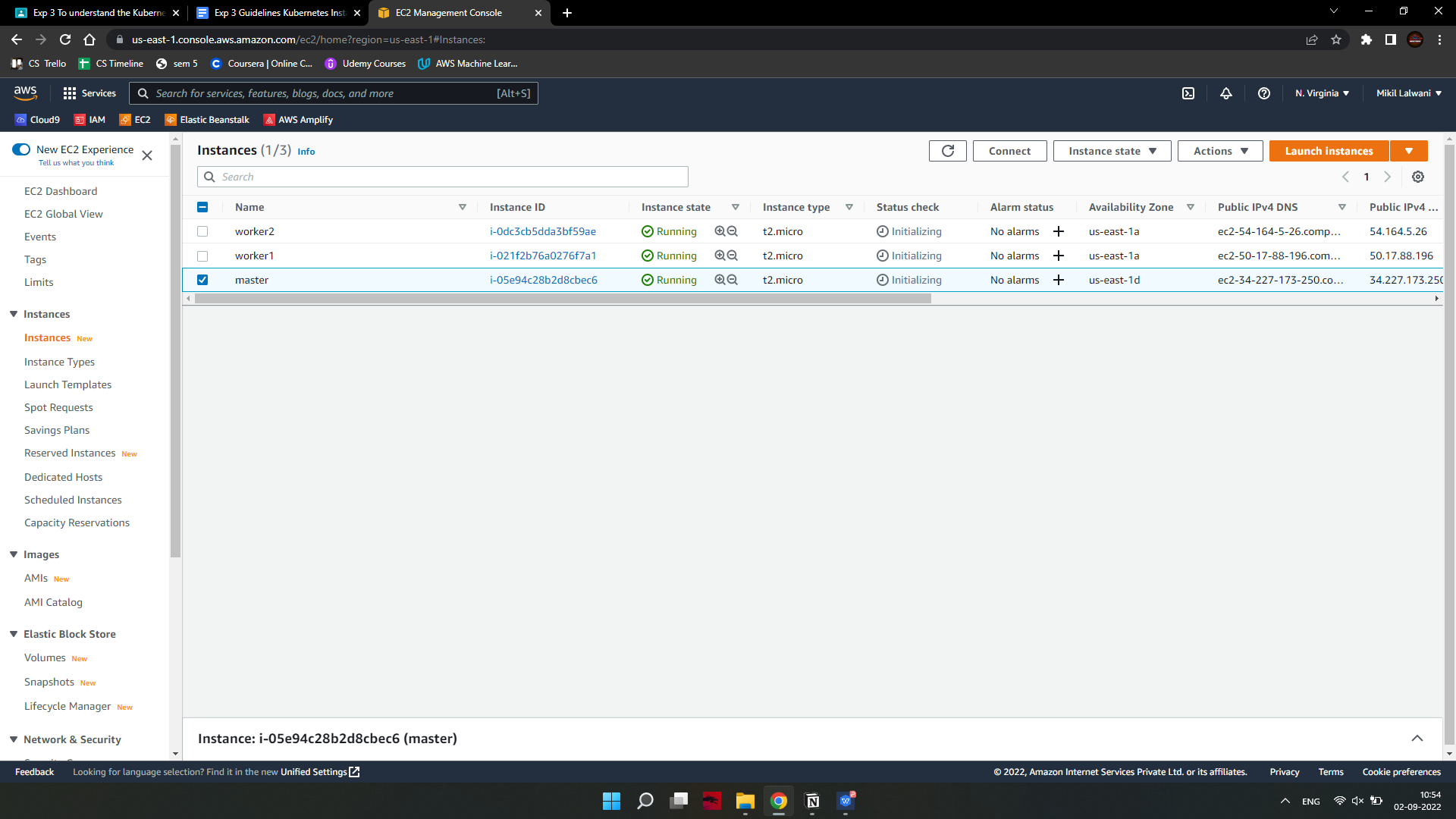
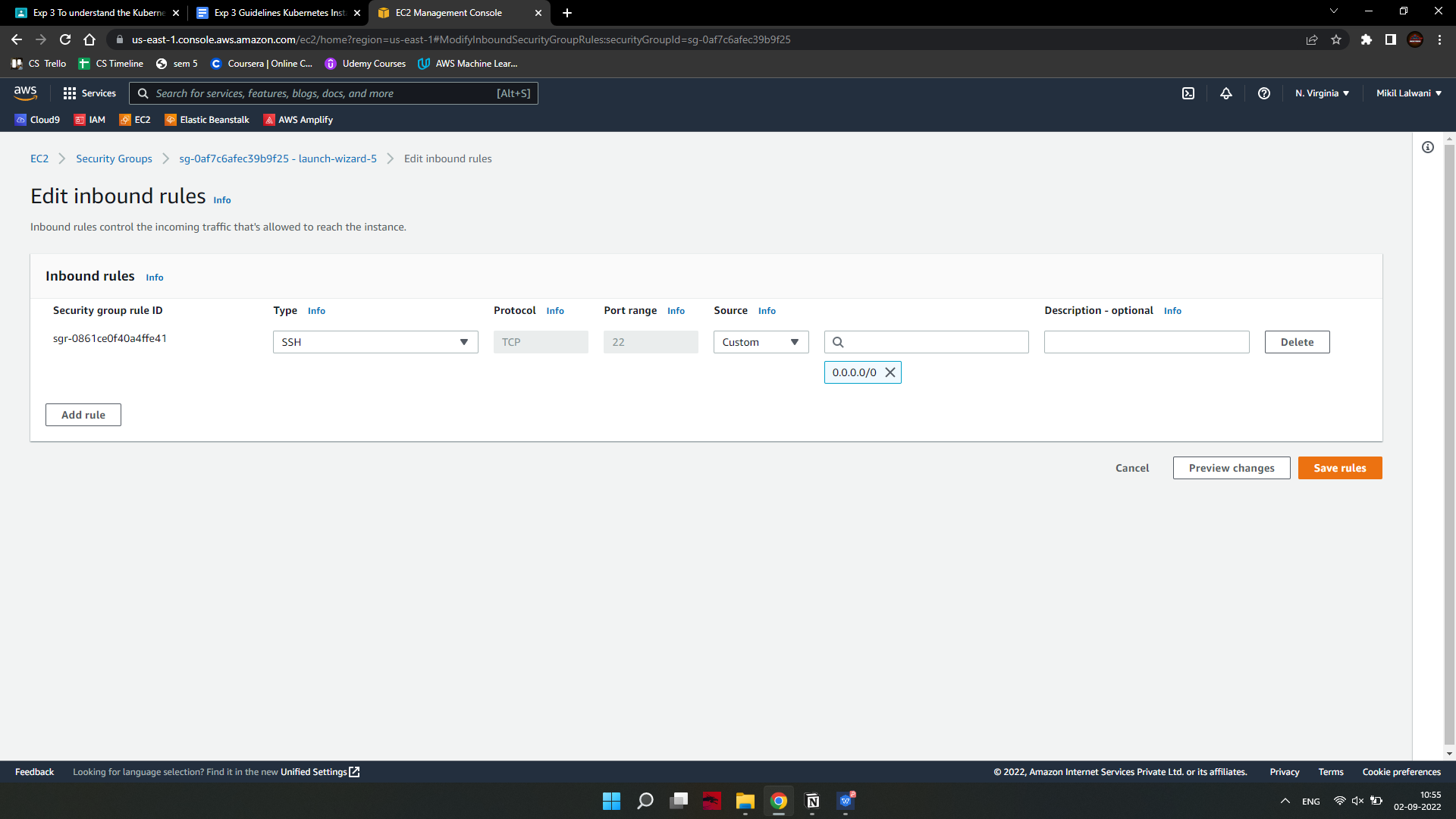
1. Create 3 EC2 instances i.e master, worker1, and worker2.



1. Change network setting.



1. Set up Docker

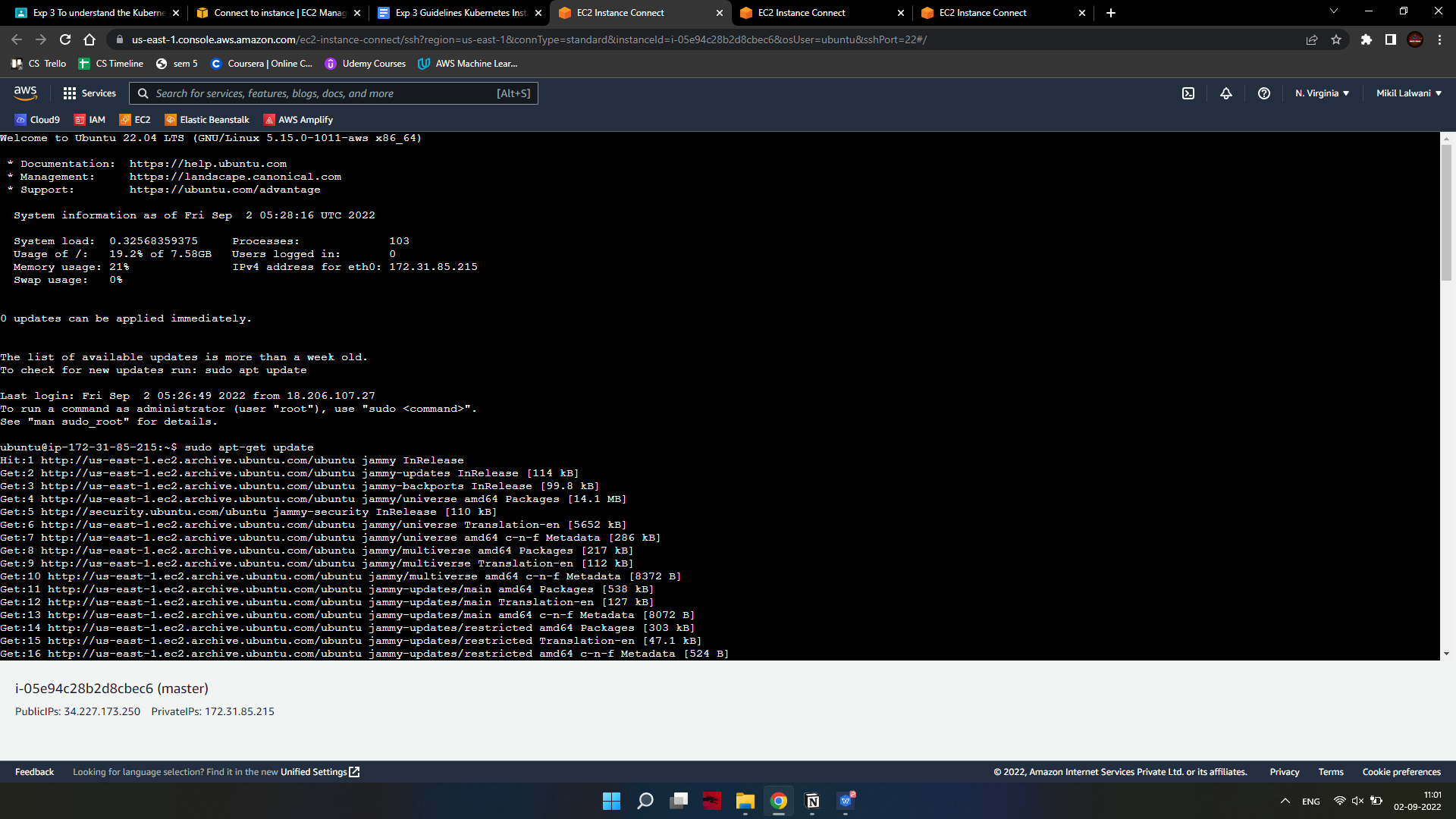
Install Docker

Kubernetes requires an existing Docker installation. If you already have Docker installed, skip ahead to Step 4.

If you do not have Kubernetes, install it by following these steps:

Update the package list with the command:

on-master&slave$sudo apt-get update



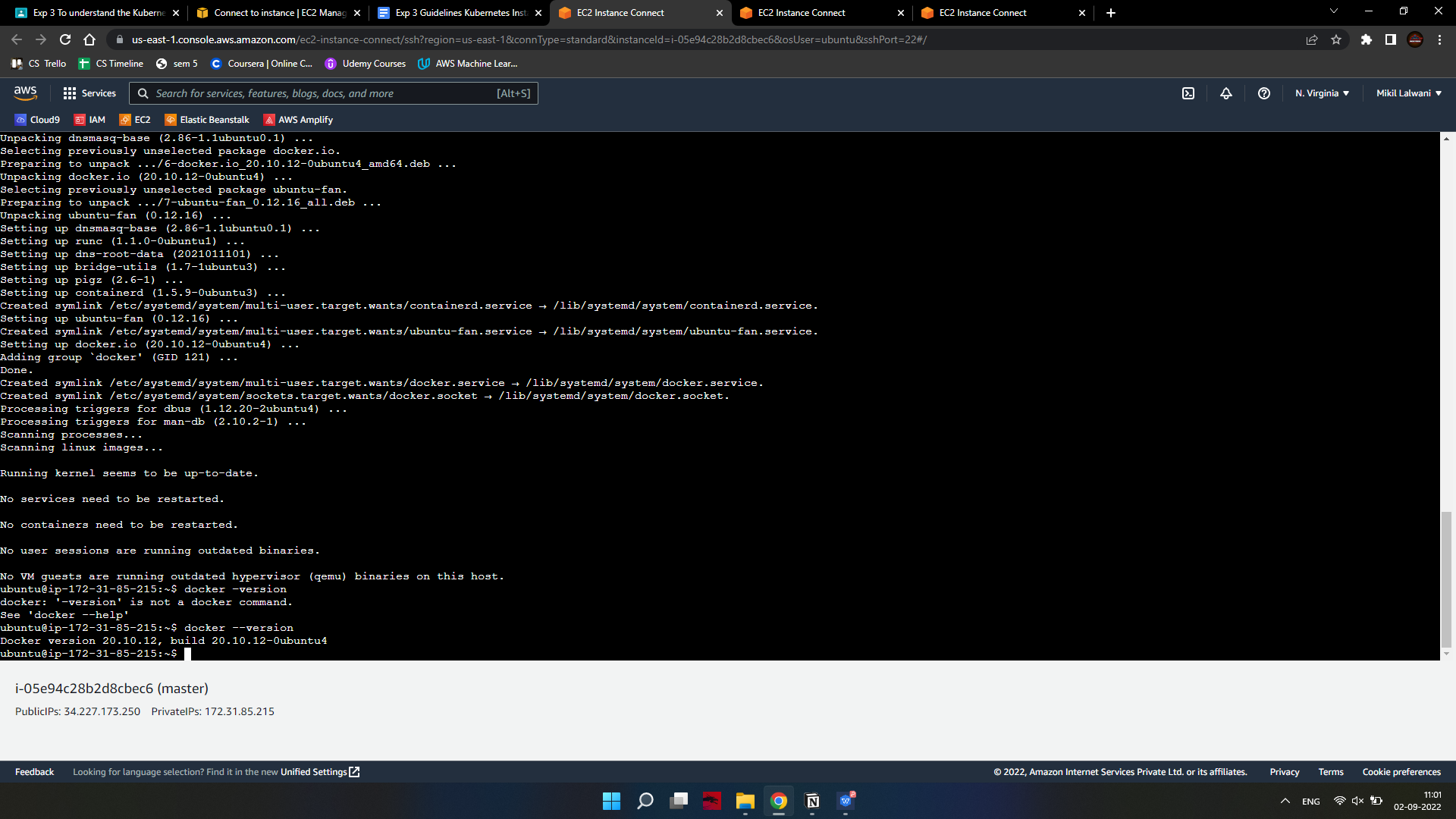
Next, install Docker with the command:

on-master&slave$sudo apt-get install docker.io

Repeat the process on each server that will act as a node.

Check the installation (and version) by entering the following:

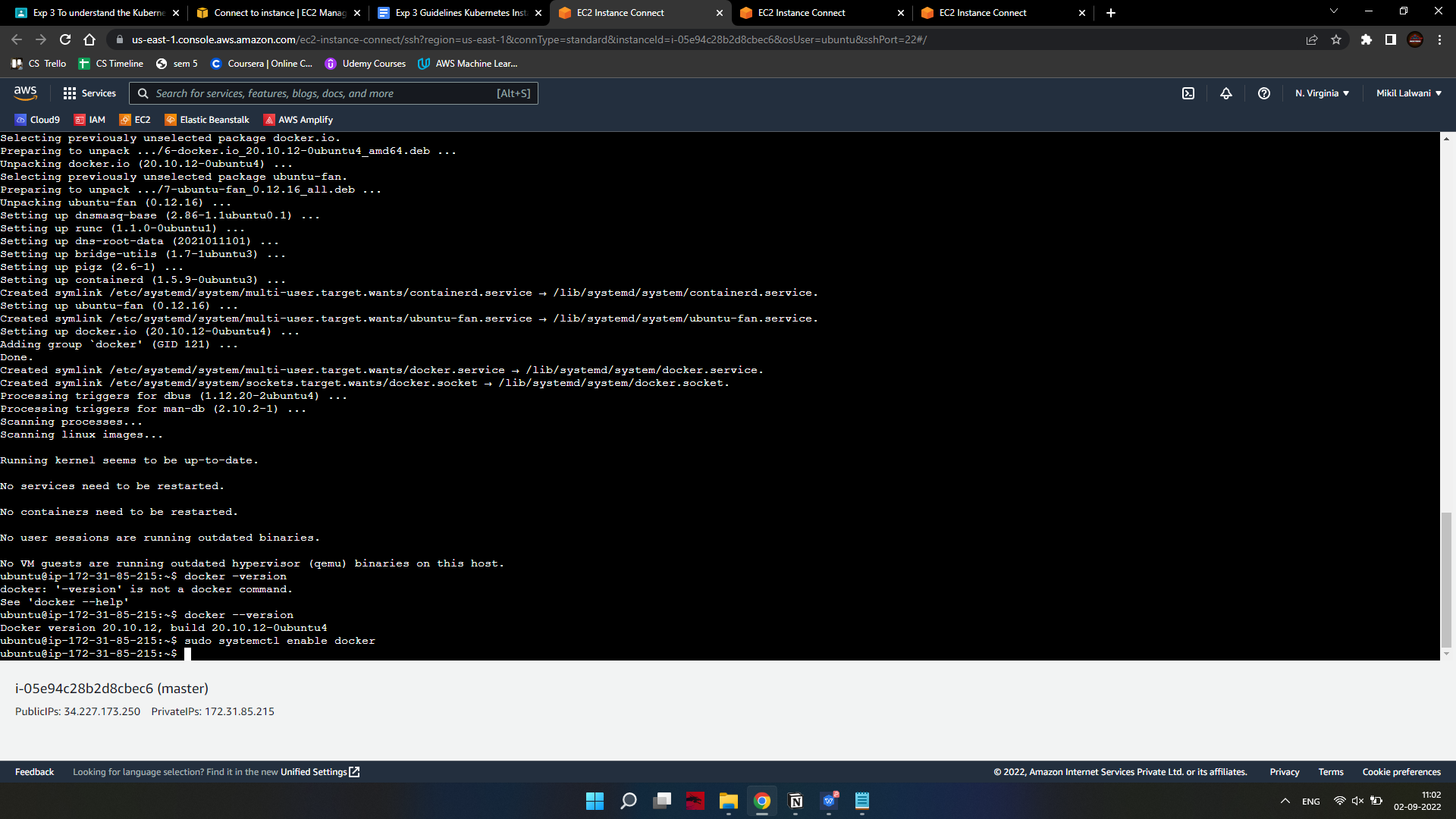
on-master&slave$docker --version



1. Start and Enable Docker

Set Docker to launch at boot by entering the following:

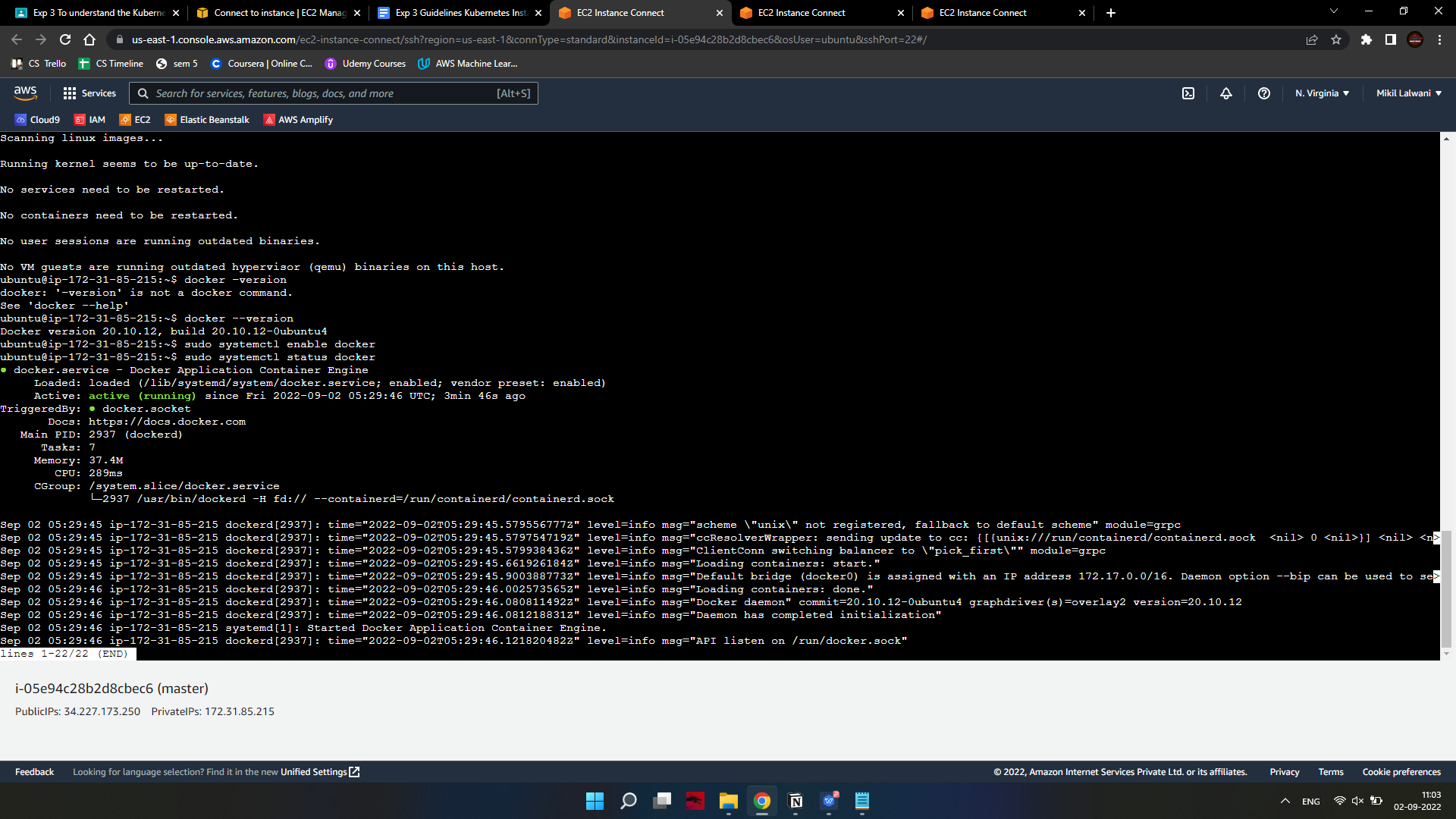
on-master&slave$sudo systemctl enable docker



Verify Docker is running:

on-master&slave$sudo systemctl status docker To start Docker if it’s not running:

on-master&slave$sudo systemctl start docker



Repeat on all the other nodes.

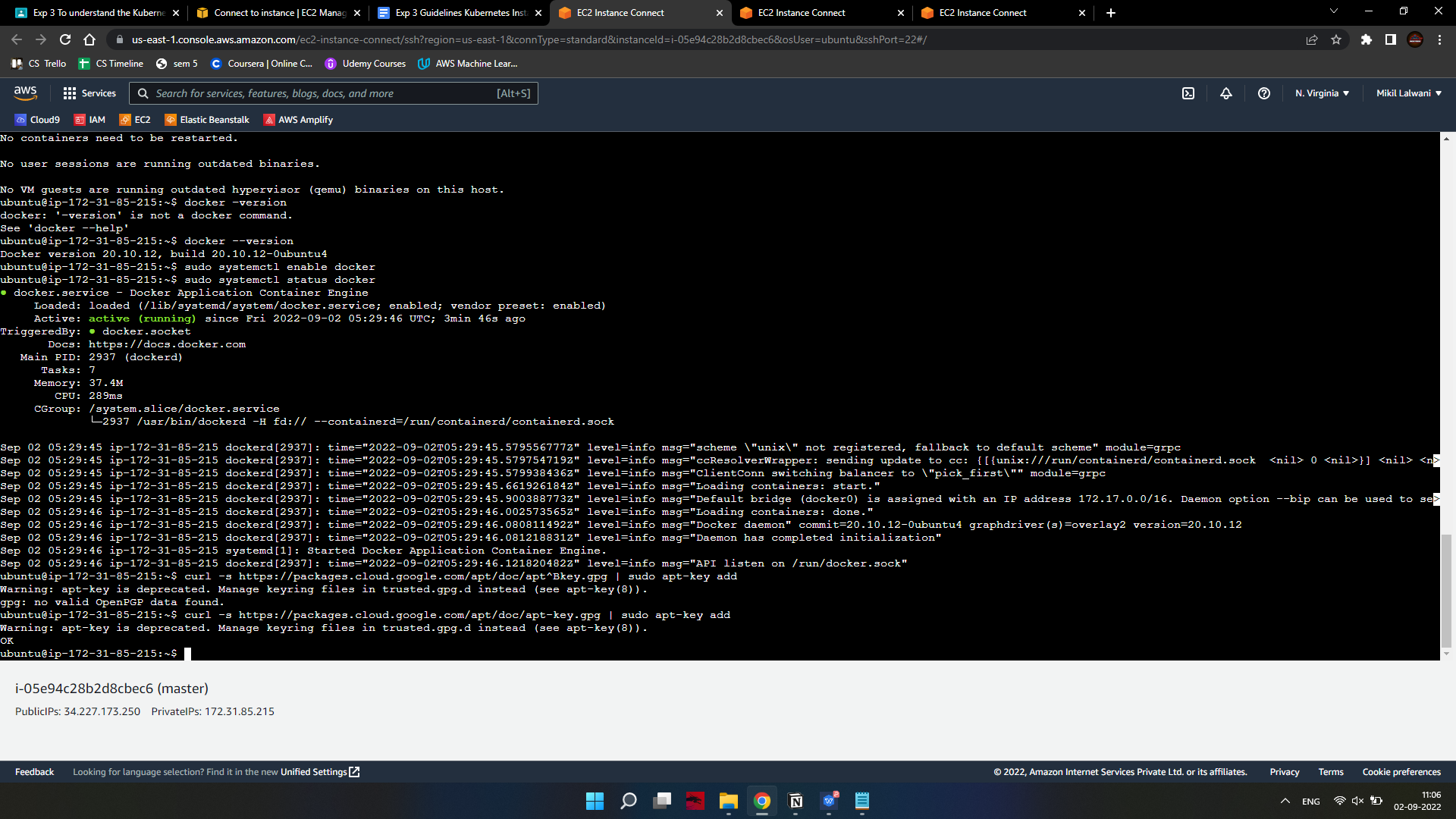
Install Kubernetes

1. Add Kubernetes Signing Key

Since you are downloading Kubernetes from a non-standard repository, it is essential to ensure that the software is authentic. This is done by adding a signing key.

Enter the following to add a signing key:

on-master&slave$curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add



If you get an error that curl is not installed, install it with:

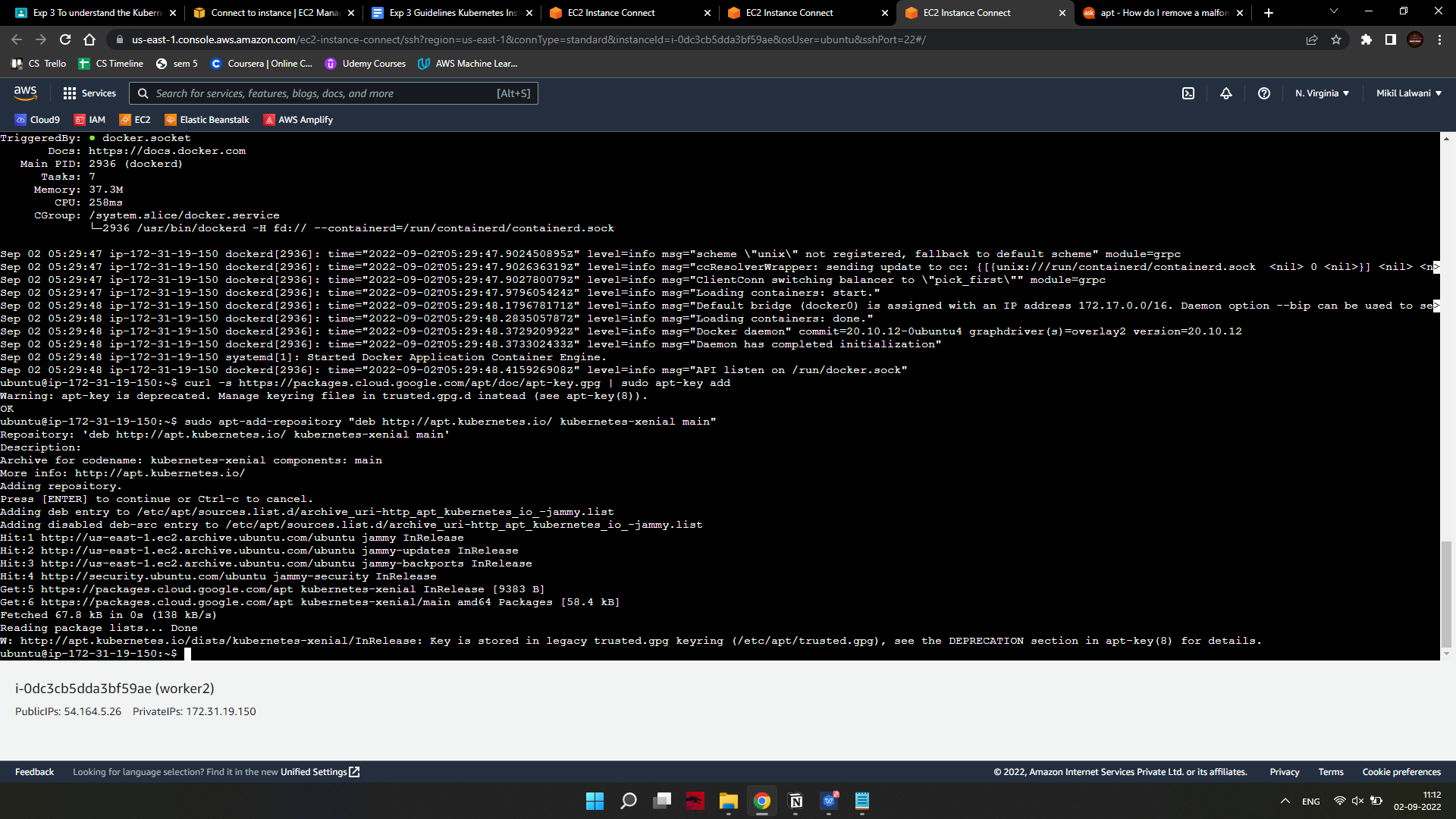
on-master&slave$sudo apt-get install curl

Then repeat the previous command to install the signing keys. Repeat for each server node.

1. Add Software Repositories

Kubernetes is not included in the default repositories. To add them, enter the following:

on-master&slave$sudo apt-add-repository "deb http://apt.kubernetes.io/ kubernetes-xenial main"

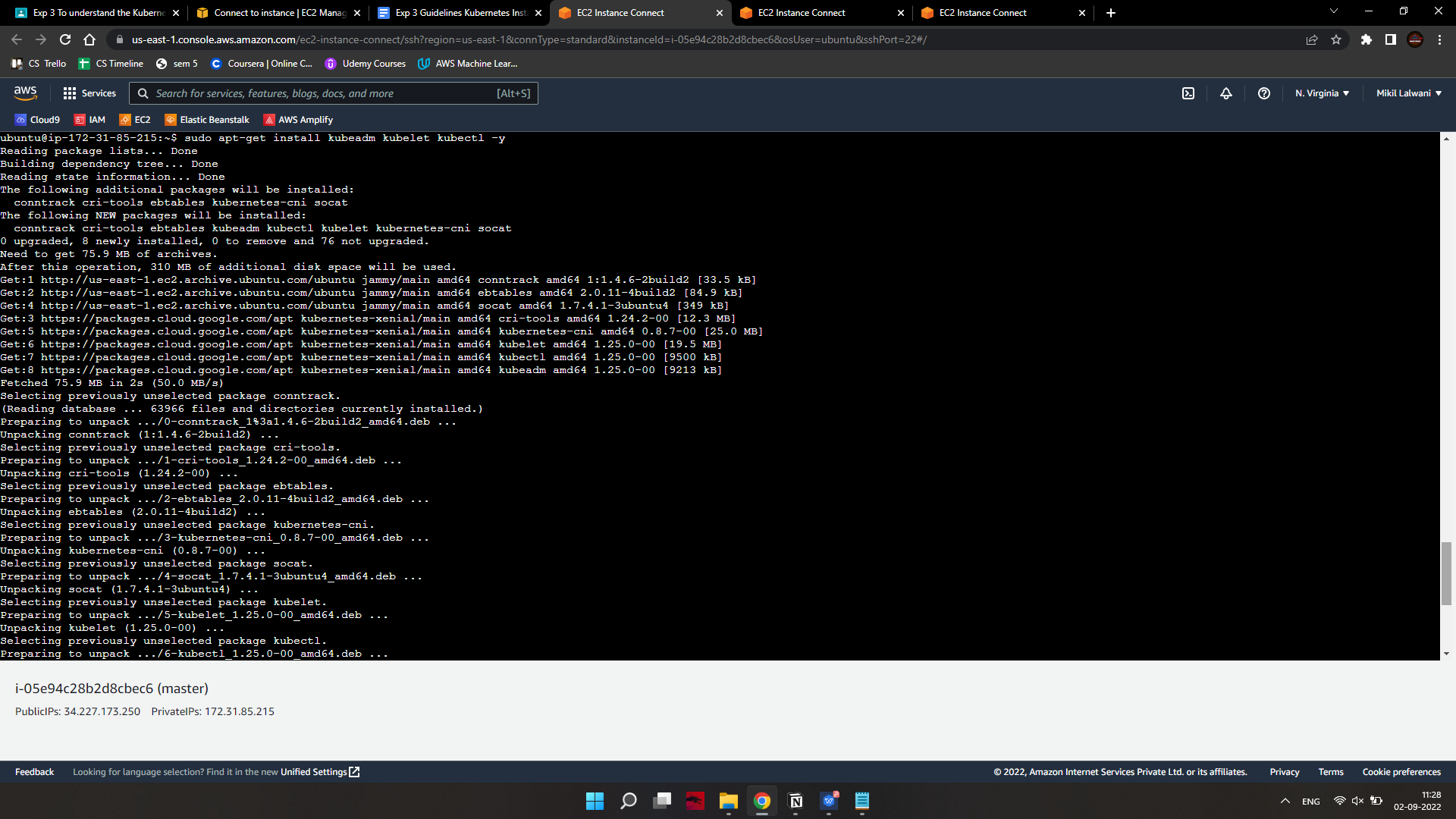


Repeat on each server node.

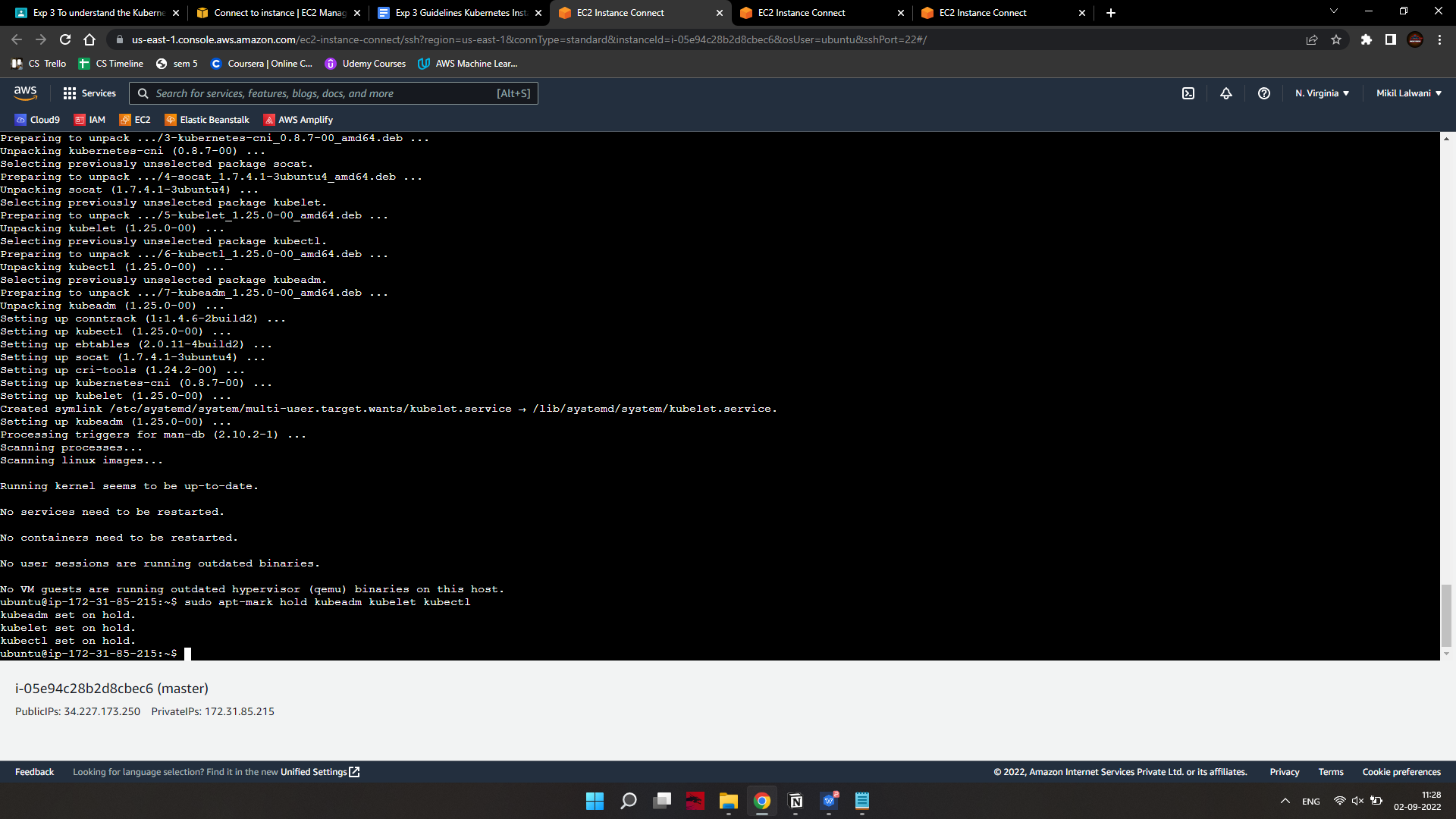
1. Kubernetes Installation Tools

Kubeadm (Kubernetes Admin) is a tool that helps initialize a cluster. It fast-tracks setup by using community-sourced best practices. Kubelet is the work package, which runs on every node and starts containers. The tool gives you command-line access to clusters.

Install Kubernetes tools with the command:

on-master&slave$sudo apt-get install kubeadm kubelet kubectl -y 

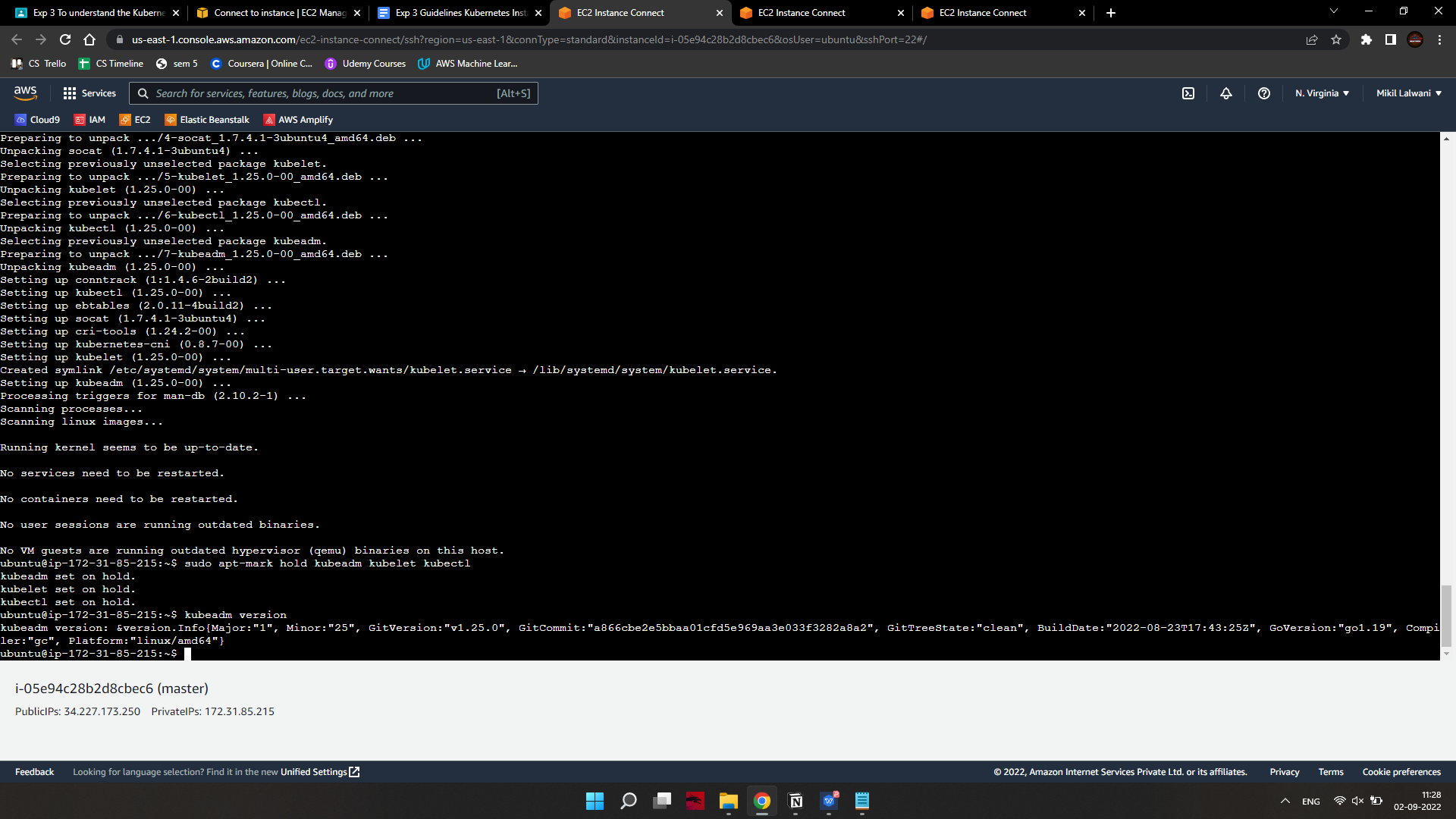
on-master&slave$sudo apt-mark hold kubeadm kubelet kubectl



Allow the process to complete.

Verify the installation with:

on-master&slave$kubeadm version



Repeat for each server node.

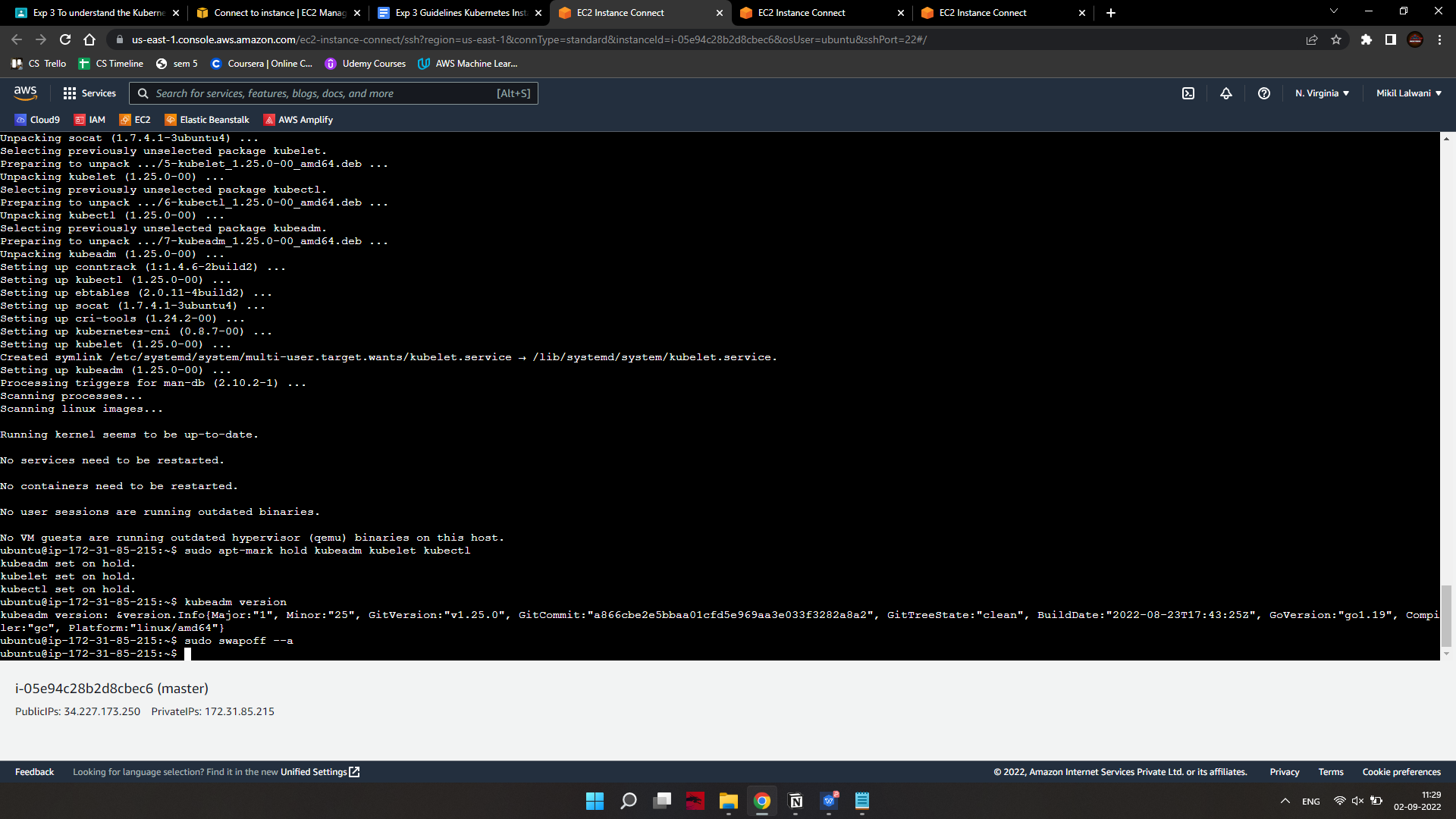
Note: Make sure you install the same version of each package on each machine. Different versions can create instability. Also, this process prevents apt from automatically updating Kubernetes. For update instructions, please see the developers’ instructions.

Kubernetes Deployment

1. Begin Kubernetes Deployment

Start by disabling the swap memory on each server:

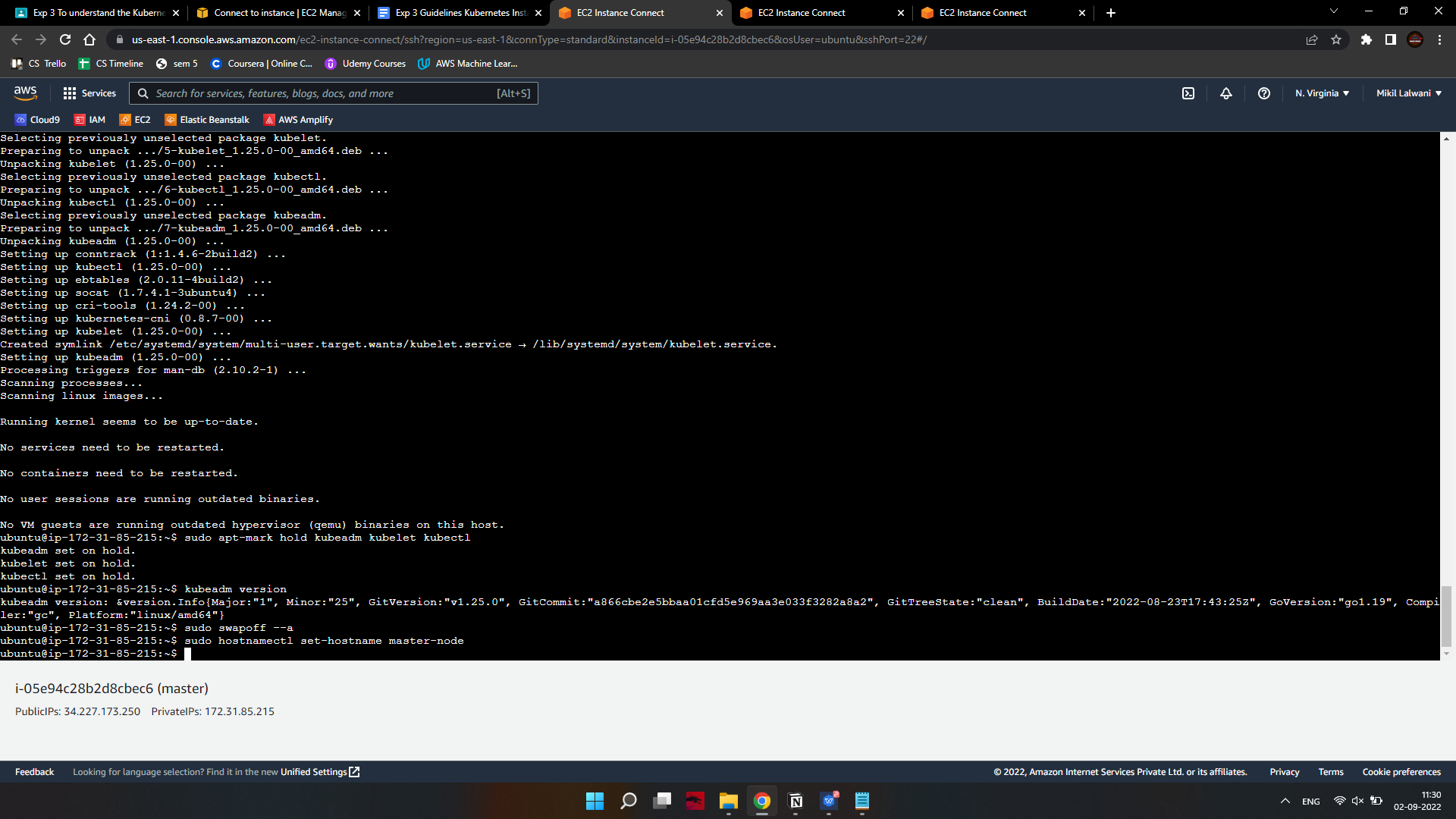
on-master&slave$sudo swapoff --a



1. Assign Unique Hostname for Each Server Node

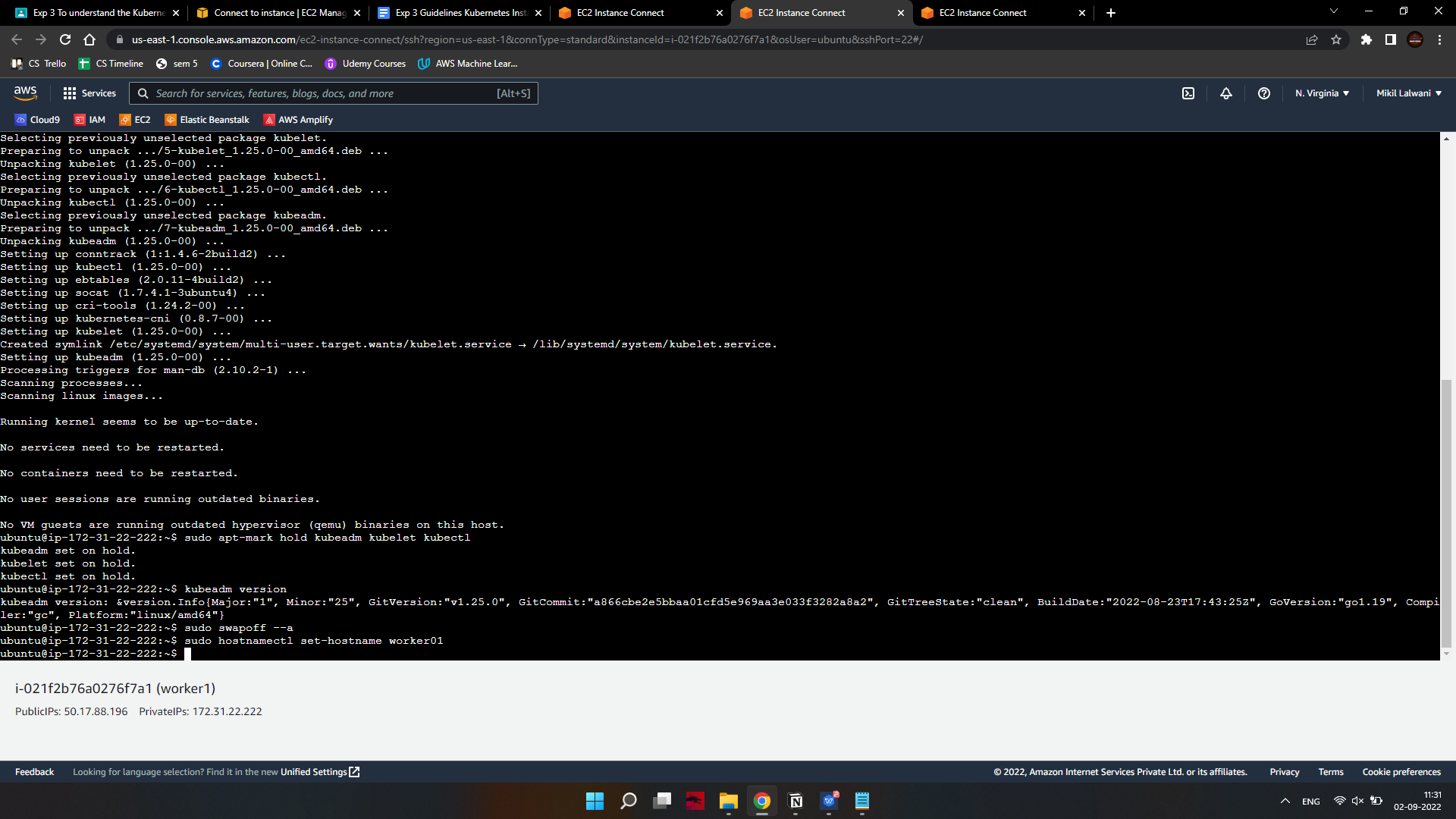
Decide which server to set as the master node. Then enter the command:

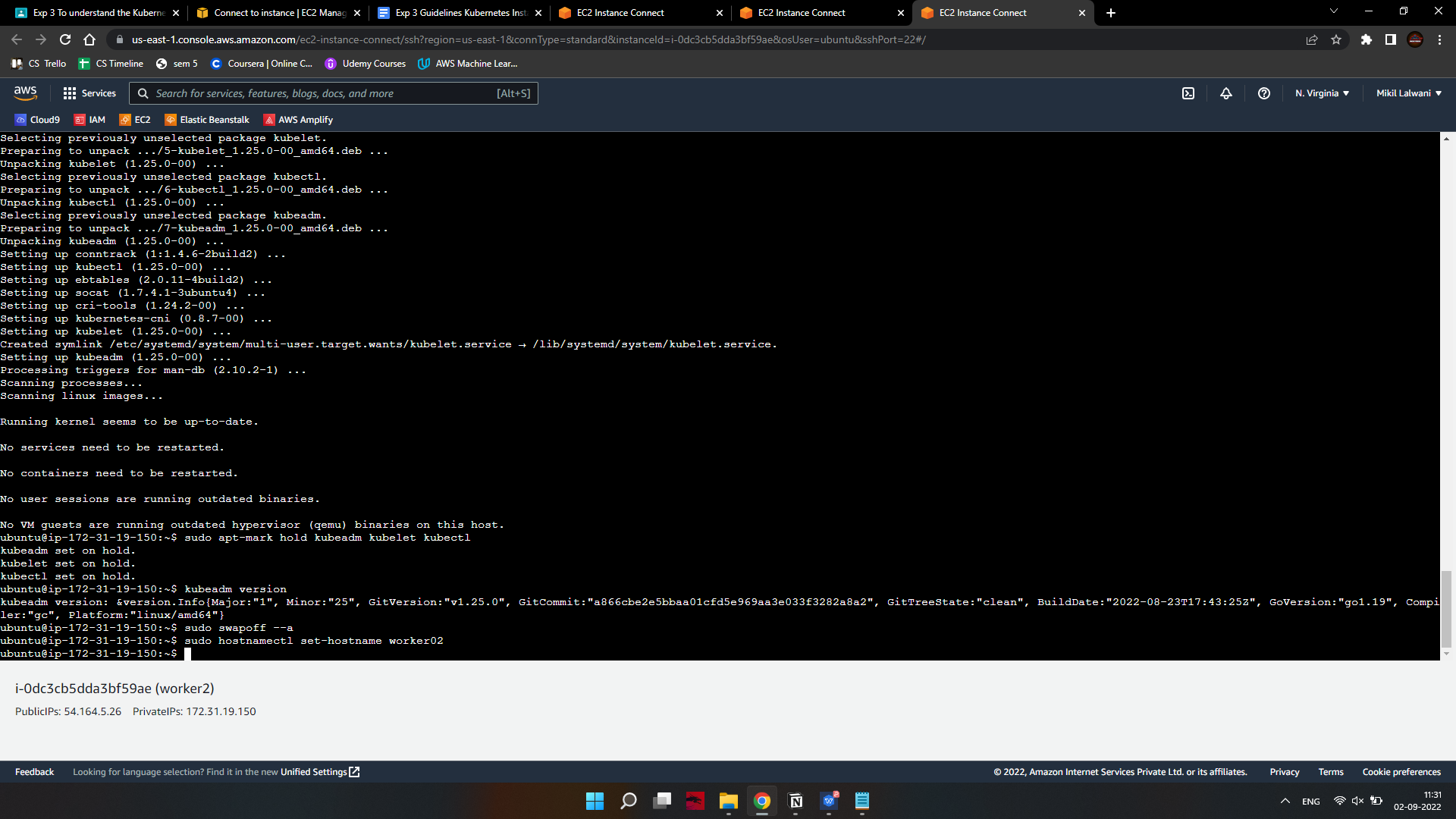
on-master$sudo hostnamectl set-hostname master-node



Next, set a worker node hostname by entering the following on the worker server:

on-slave$sudo hostnamectl set-hostname worker01\





If you have additional worker nodes, use this process to set a unique hostnsame on each.

1. Initialize Kubernetes on Master Node

Switch to the master server node, and enter the following:

on-master$sudo kubeadm init --pod-network-cidr=10.244.0.0/16

If you are trying to run this on EC2 you’ll get an error message saying less cpu and memory to override the error run the above command with --ignore-preflight-errors=all

For eg: on-master$sudo kubeadm init --pod-network-cidr=10.244.0.0/16 --ignore-preflight-errors=all

Once this command finishes, it will display a kubeadm join message at the end. Make a note of the whole entry. This will be used to join the worker nodes to the cluster.

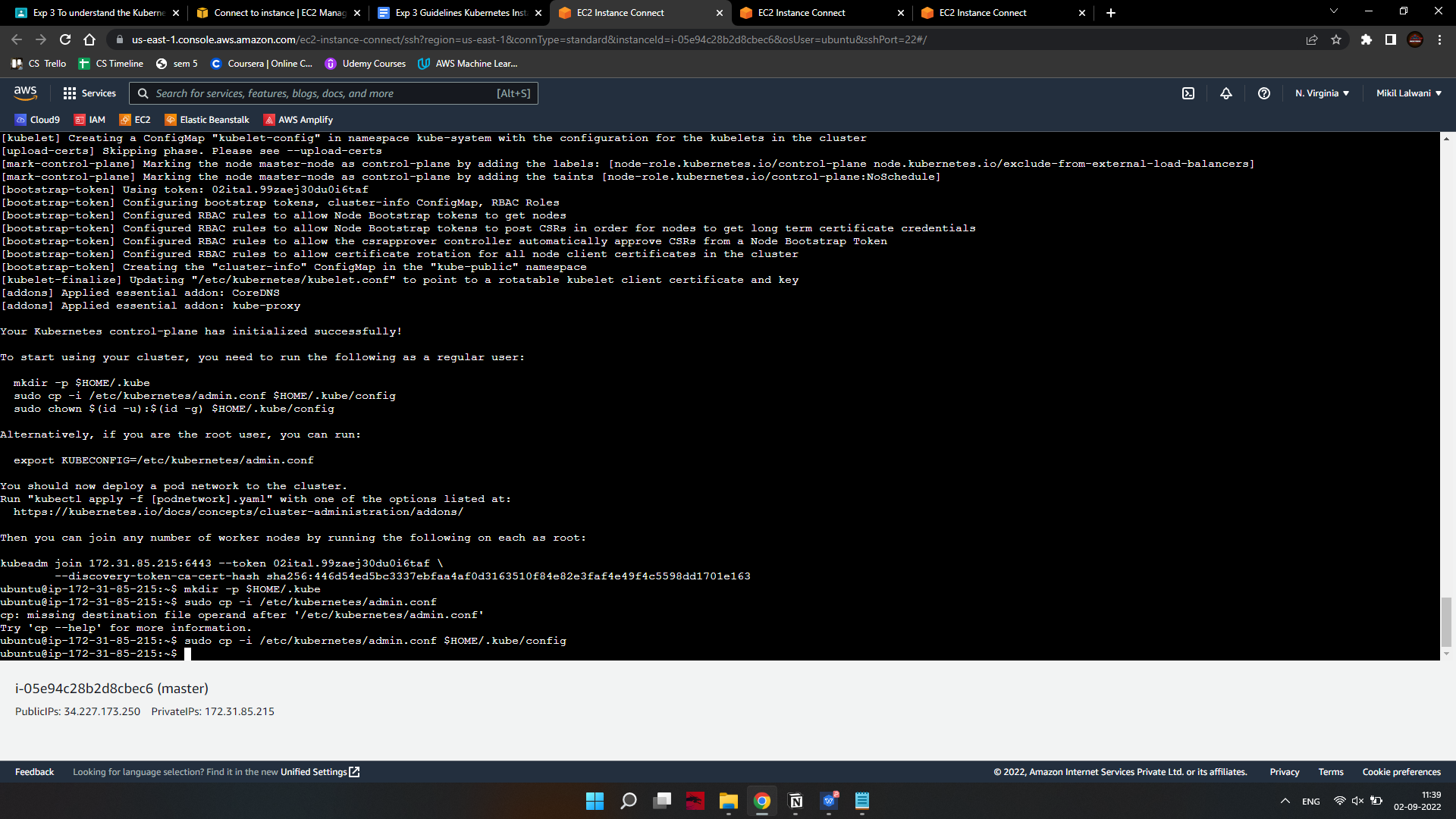
kubeadm join 172.31.38.116:6443 --token lhu178.98f4hafj9qdn4pve \

--discovery-token-ca-cert-hash sha256:8f55371cd3d53a5d72faa368e93e0a99d2953016706de871fcdb27c83e5b995c

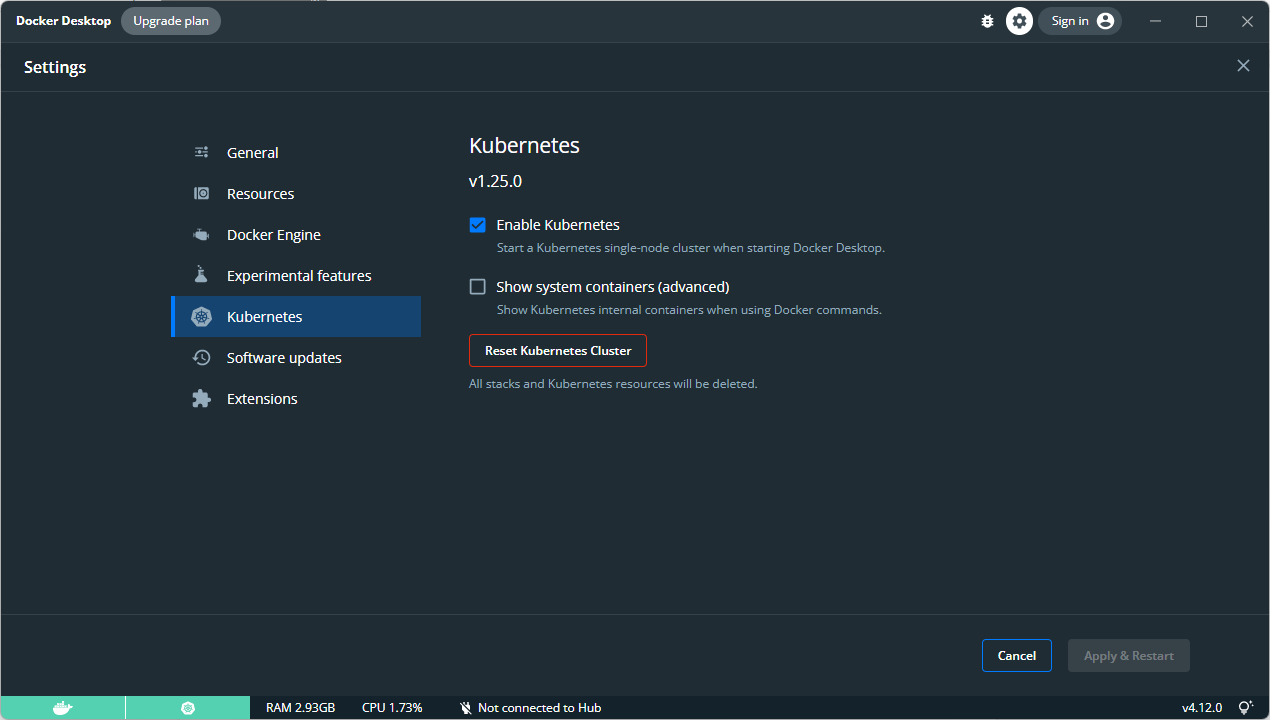
Next, enter the following to create a directory for the cluster:

kubernetes-master:~$ mkdir -p $HOME/.kube

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



1. Install Docker Desktop.
2. Enable Kubernetes in settings.



1. Run this commands

kubectl create deployment httpenv --image=bretfisher/httpenv

kubectl scale deployment/httpenv --replicas=3

kubectl expose deployment/httpenv --port 8888

kubectl get service

kubectl run tmp-shell --rm -it --image bretfisher/netshoot -- bash

1. To delete pods run kubectl delete pod <pod-name>