ANSIBLE LAB 2

ADHOC COMMANDS

First login to your user

1. In ansible server

ansible demo -a “ls”

a => argument

demo => group

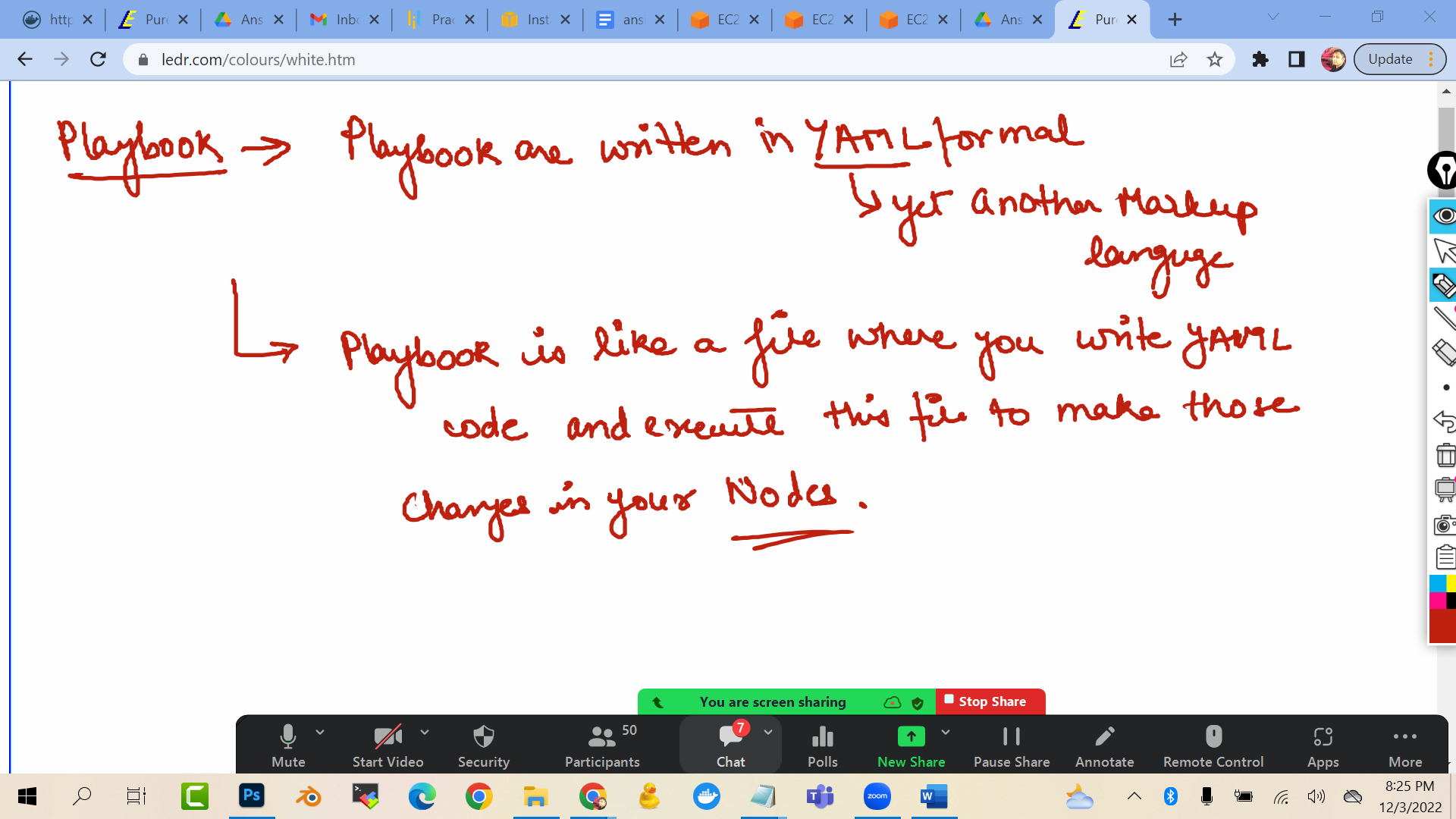
(this will list all the files in node1 and node2)

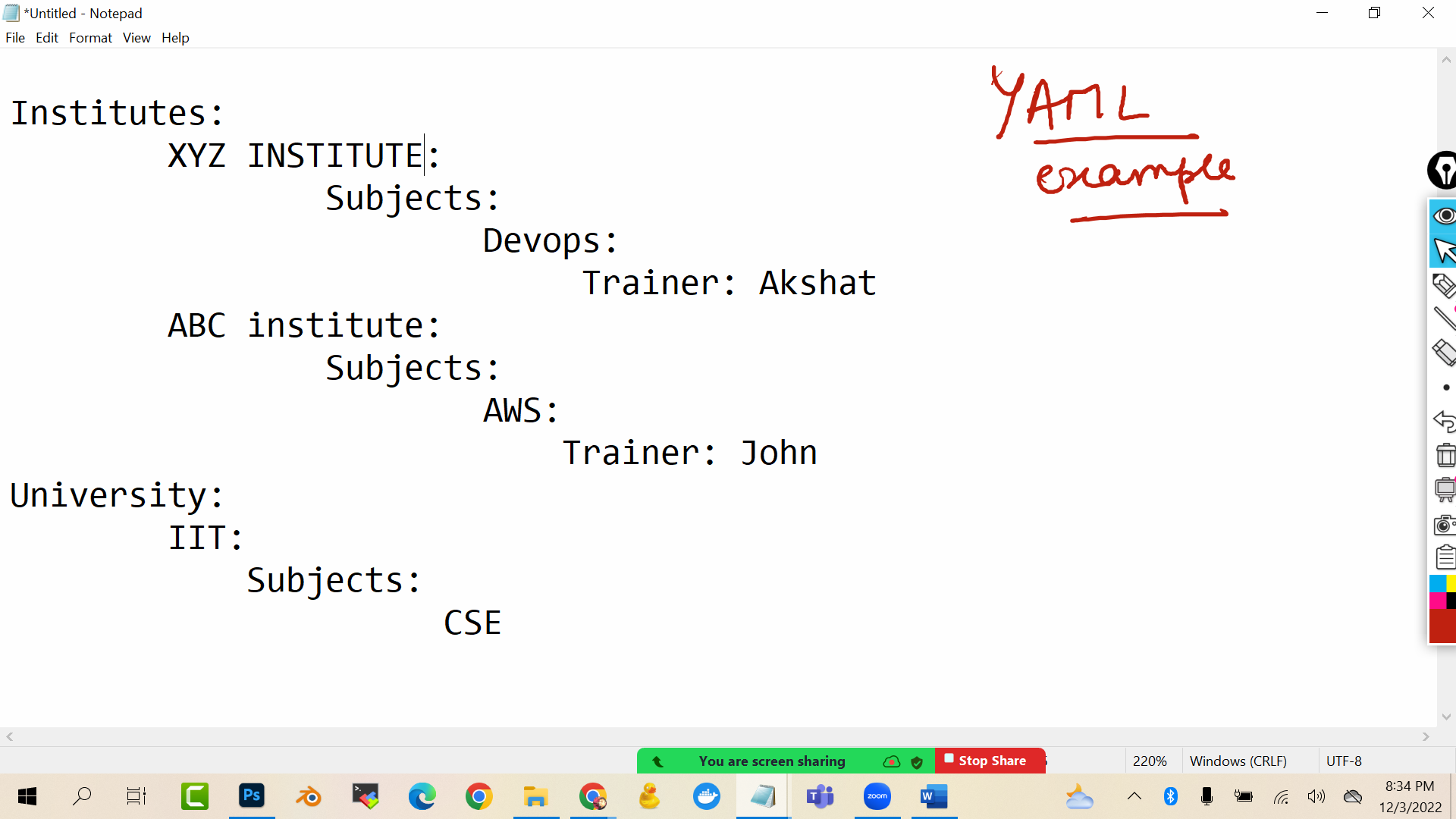
ansible demo -a “touch file1”

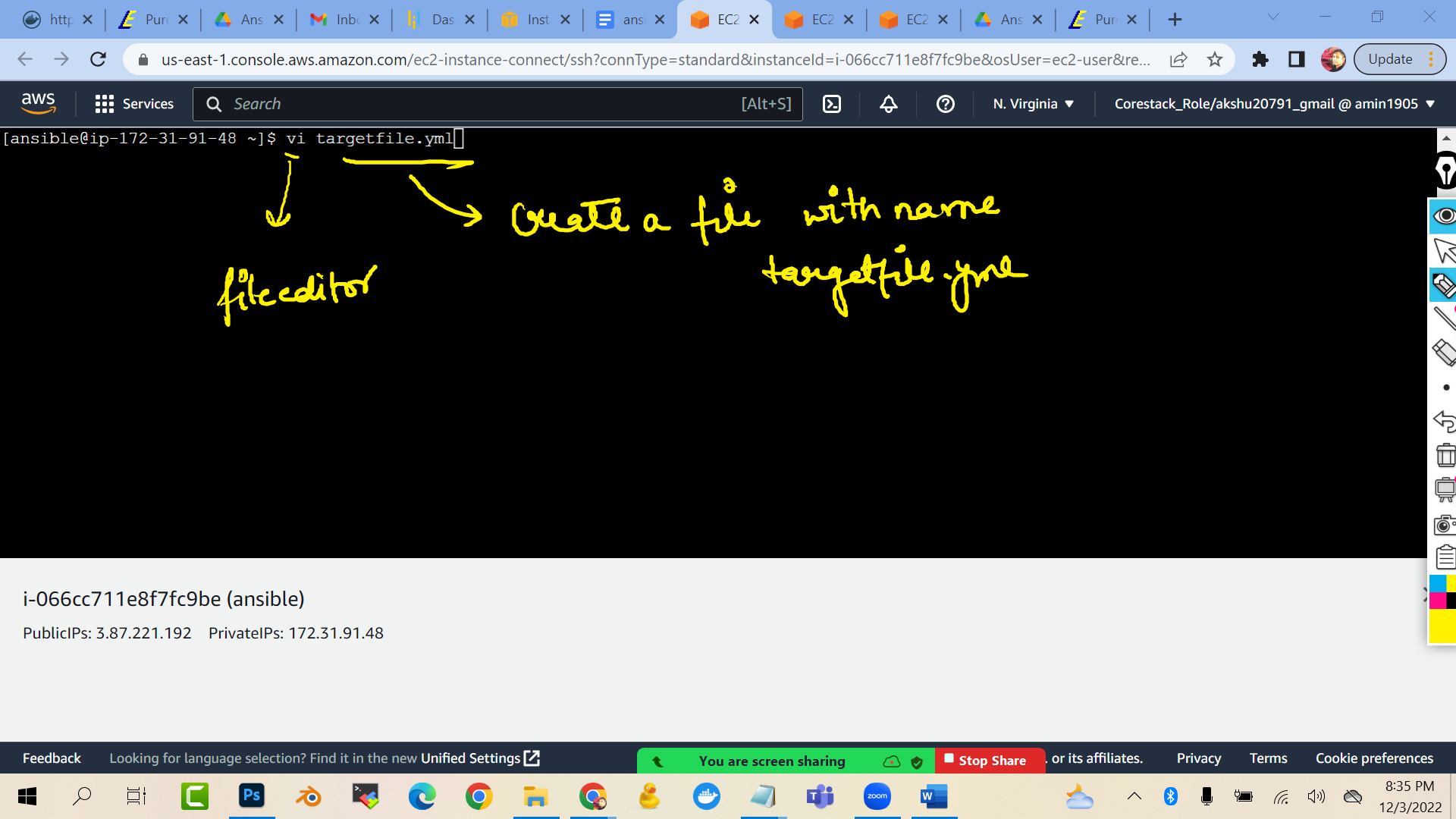
(it will create file1 in nodes)

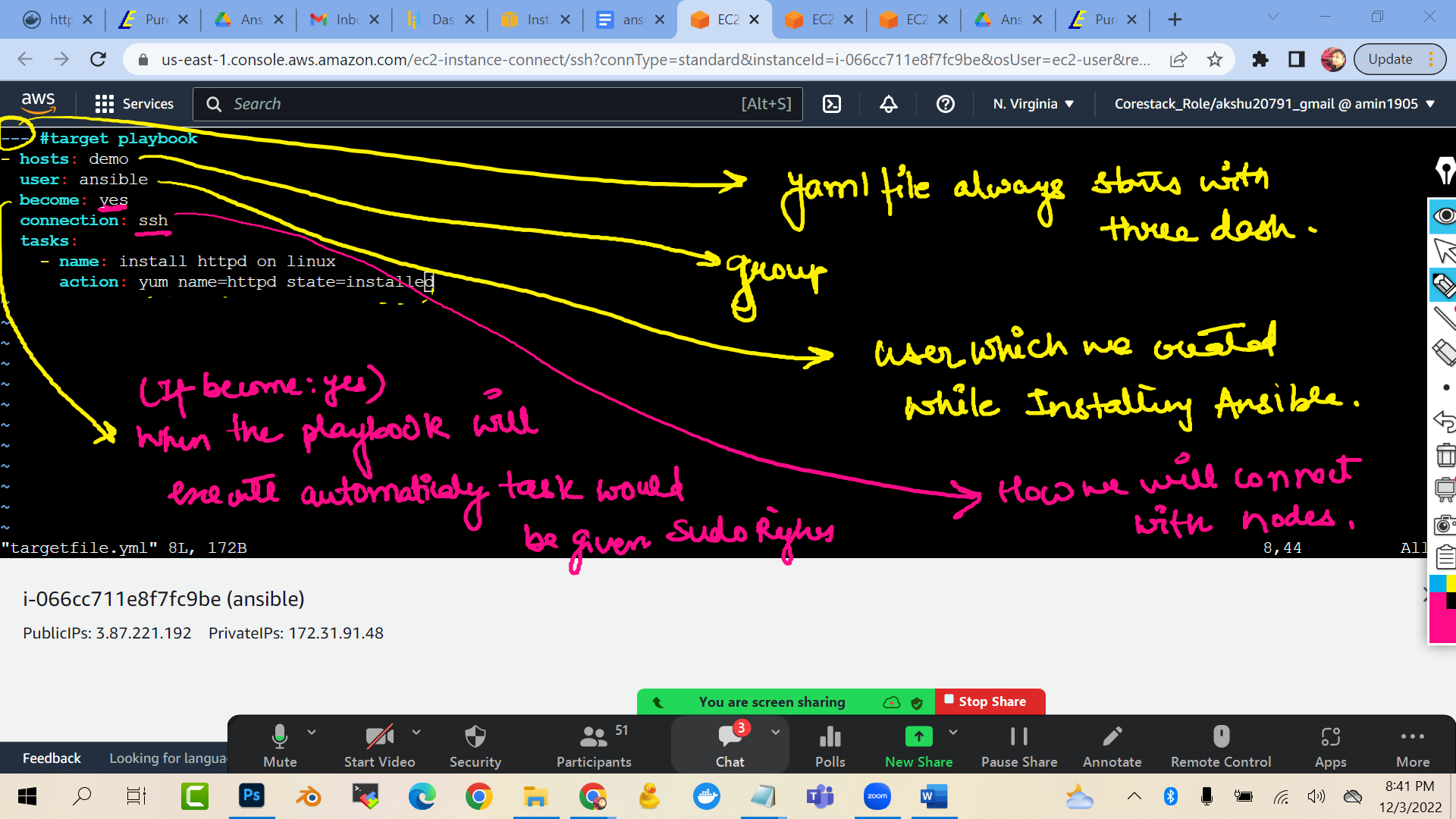
ansible demo -a “sudo yum install httpd -y”

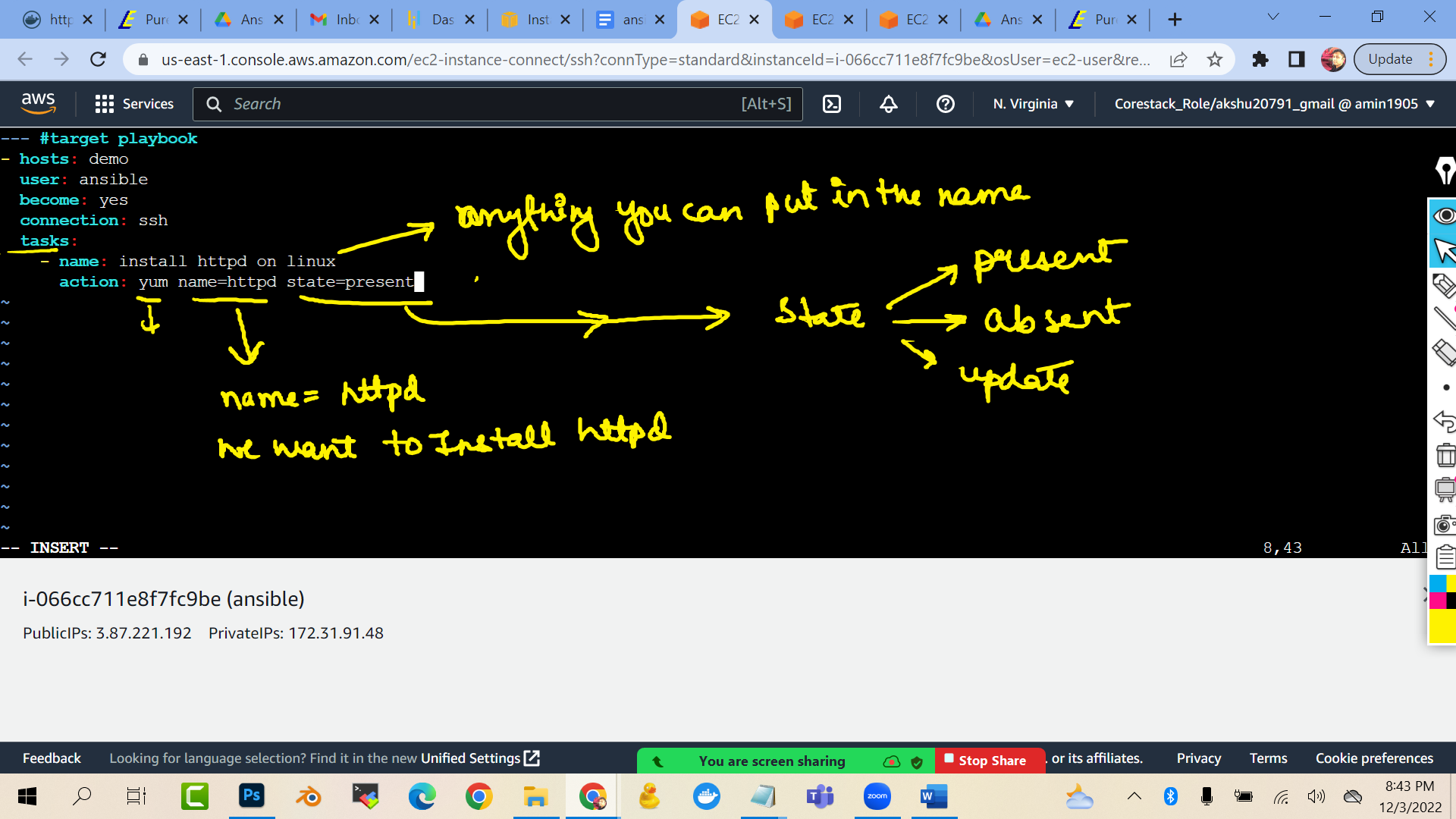
(this will install httpd in your nodes)

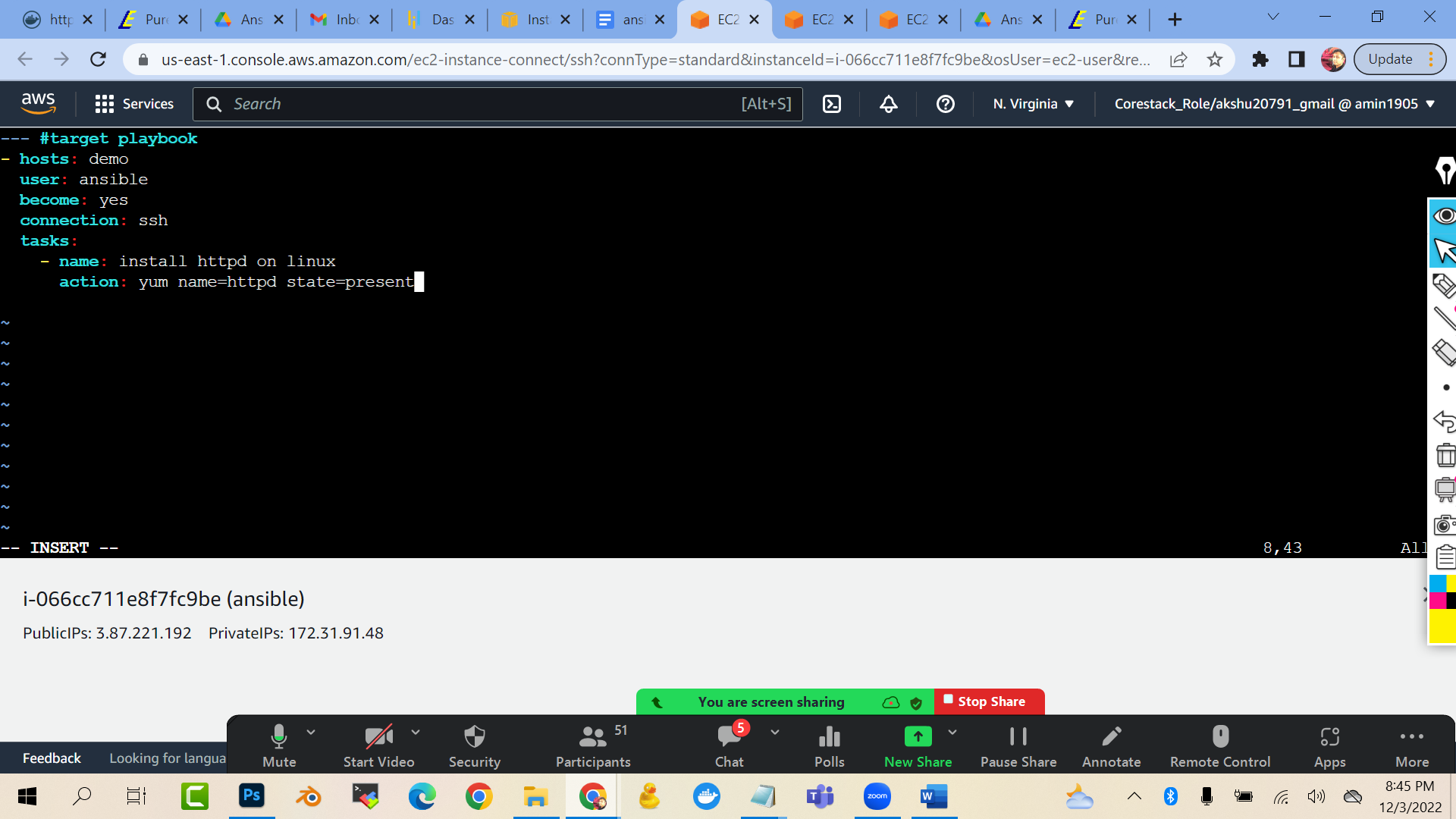












--- #target playbook

- hosts: demo

user: ansible

become: yes

connection: ssh

tasks:

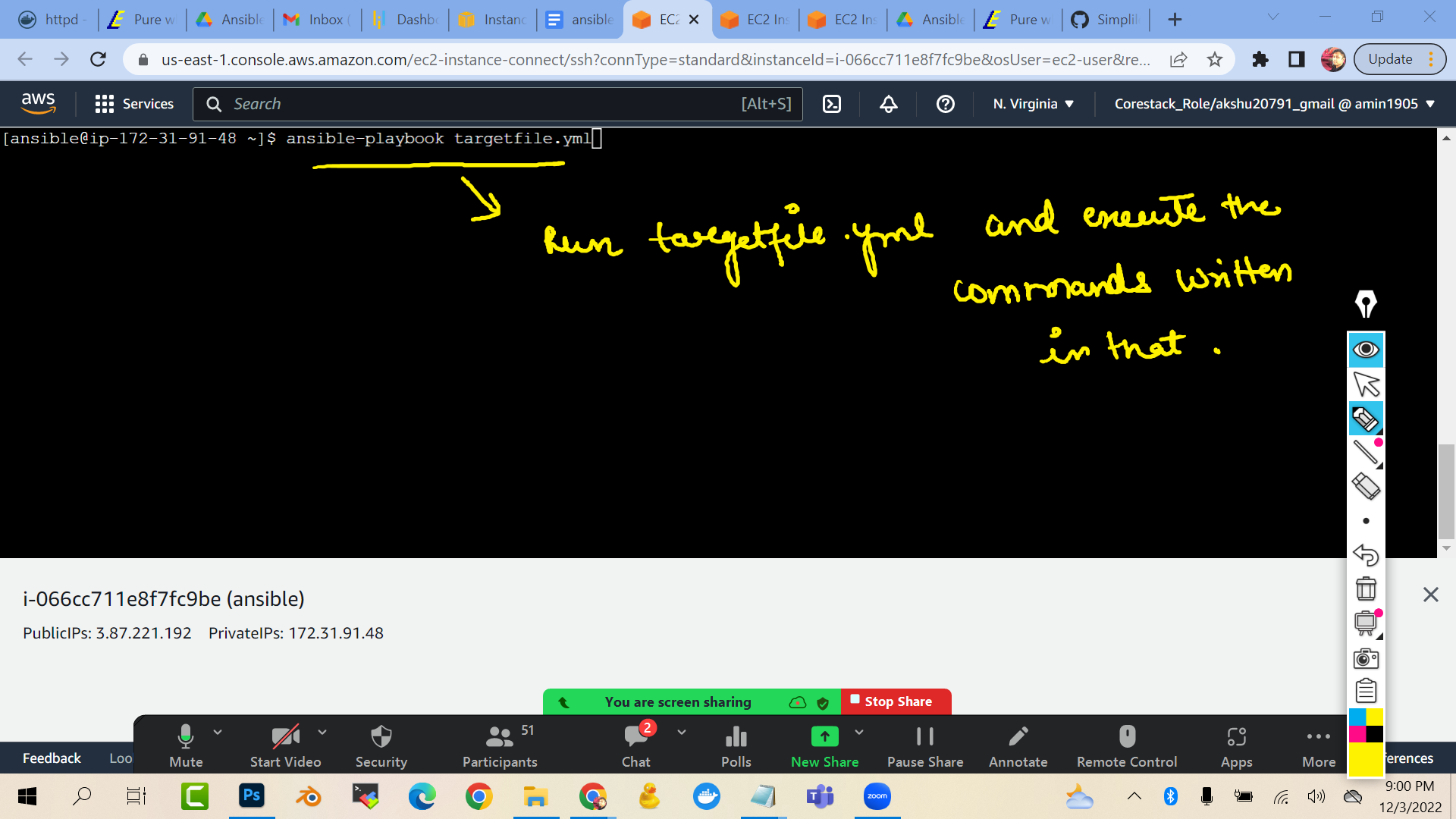
- name: install httpd on linux

action: yum name=httpd state=present

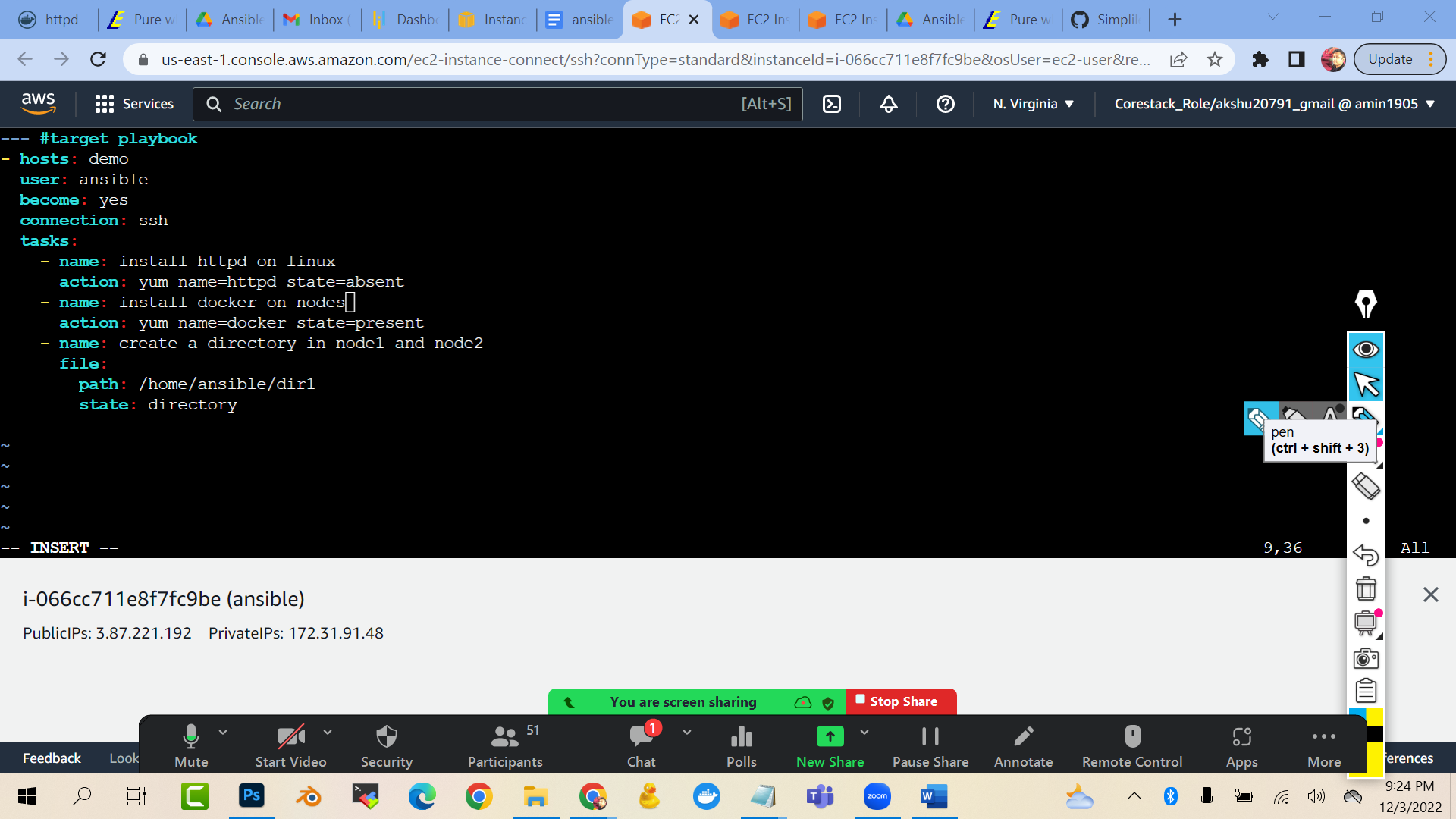
(press esc :wq to exit)

To execute this anisble playbook:

ansible-playbook targetfile.yml



Now lets do one more thing,.,,,remove httpd , install docker, create directory



--- #target playbook

- hosts: demo

user: ansible

become: yes

connection: ssh

tasks:

- name: install httpd on linux

action: yum name=httpd state=absent

- name: install docker on nodes

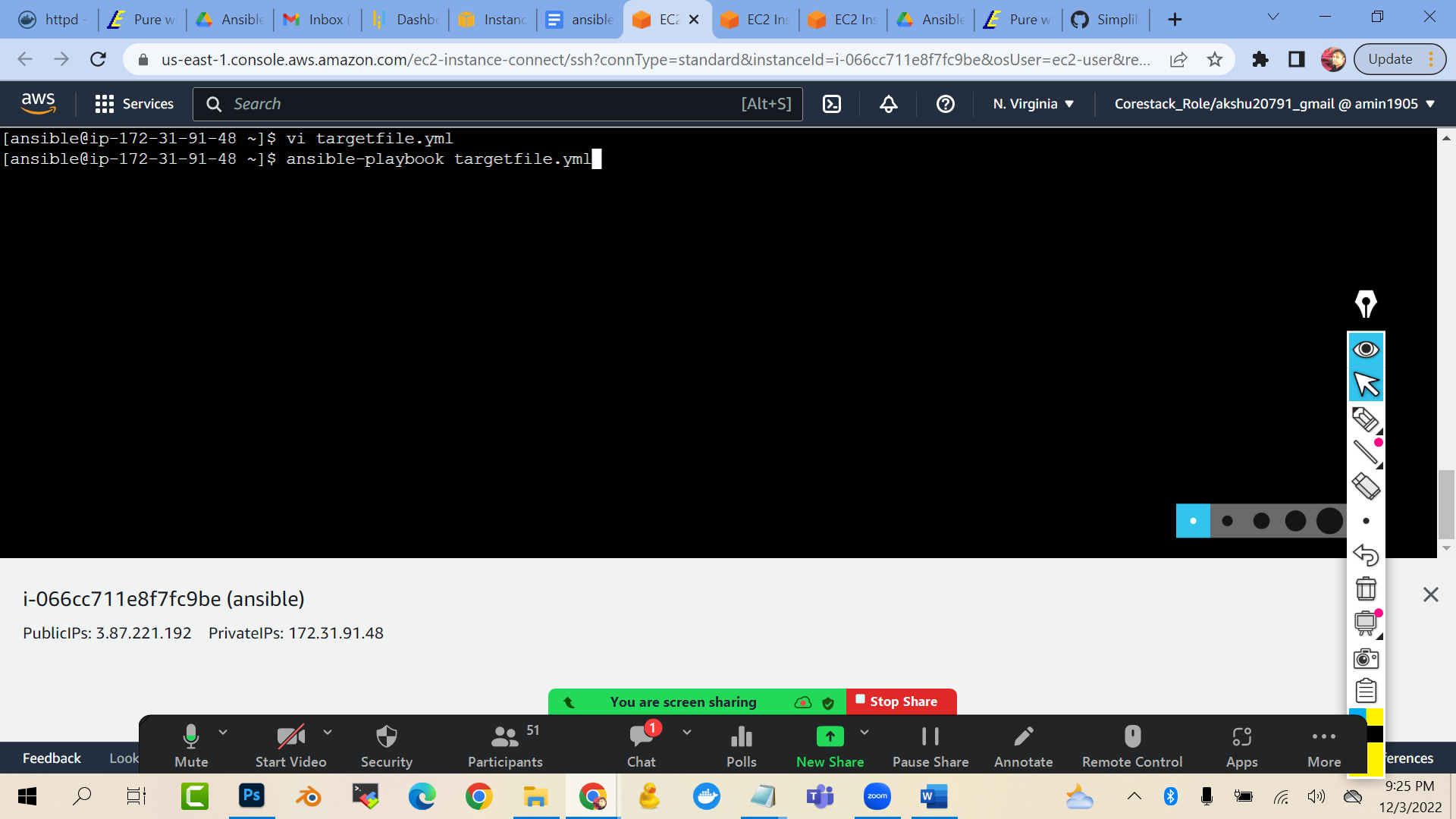
action: yum name=docker state=present

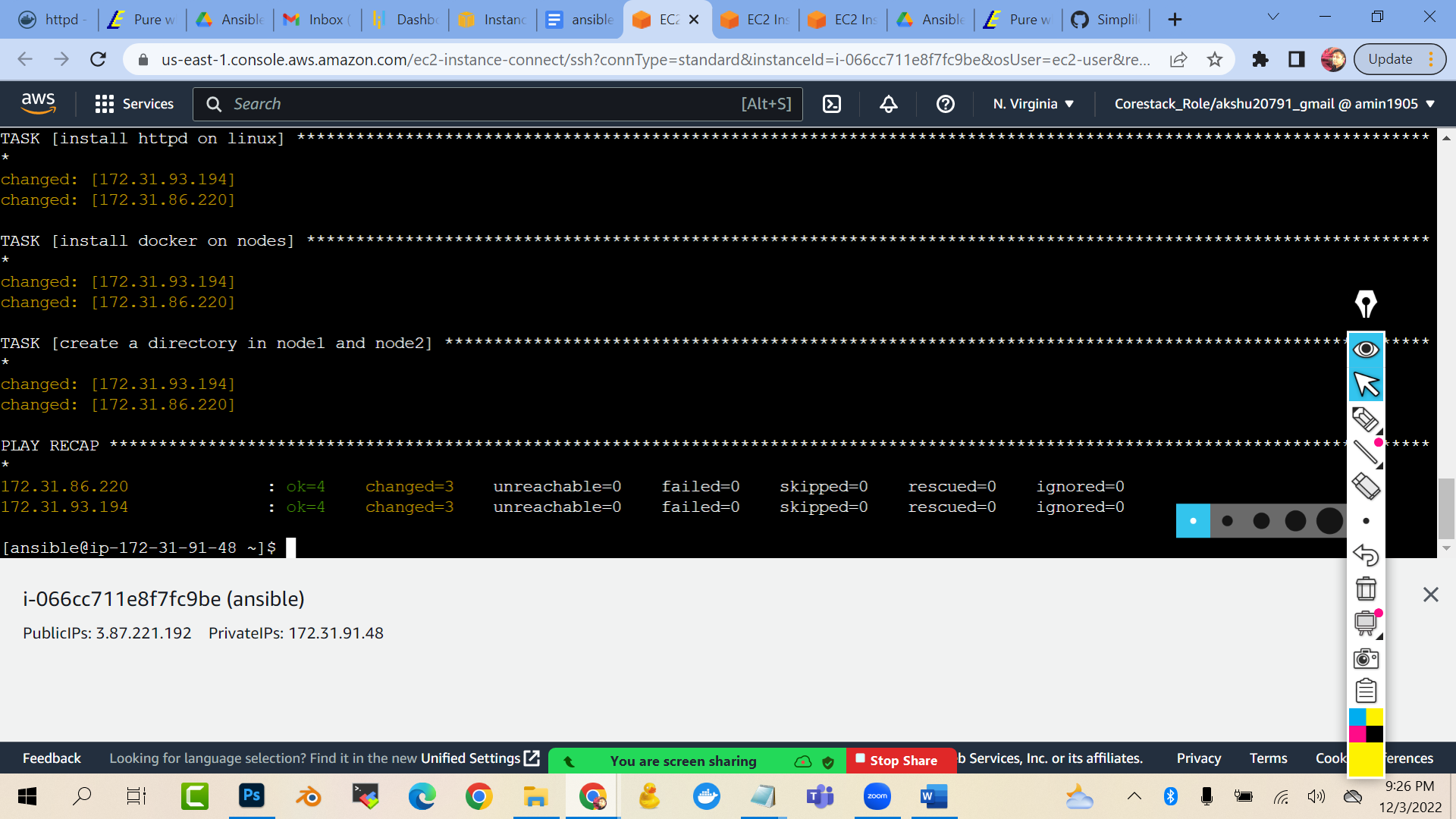
- name: create a directory in node1 and node2

file:

path: /home/ansible/dir1

state: directory

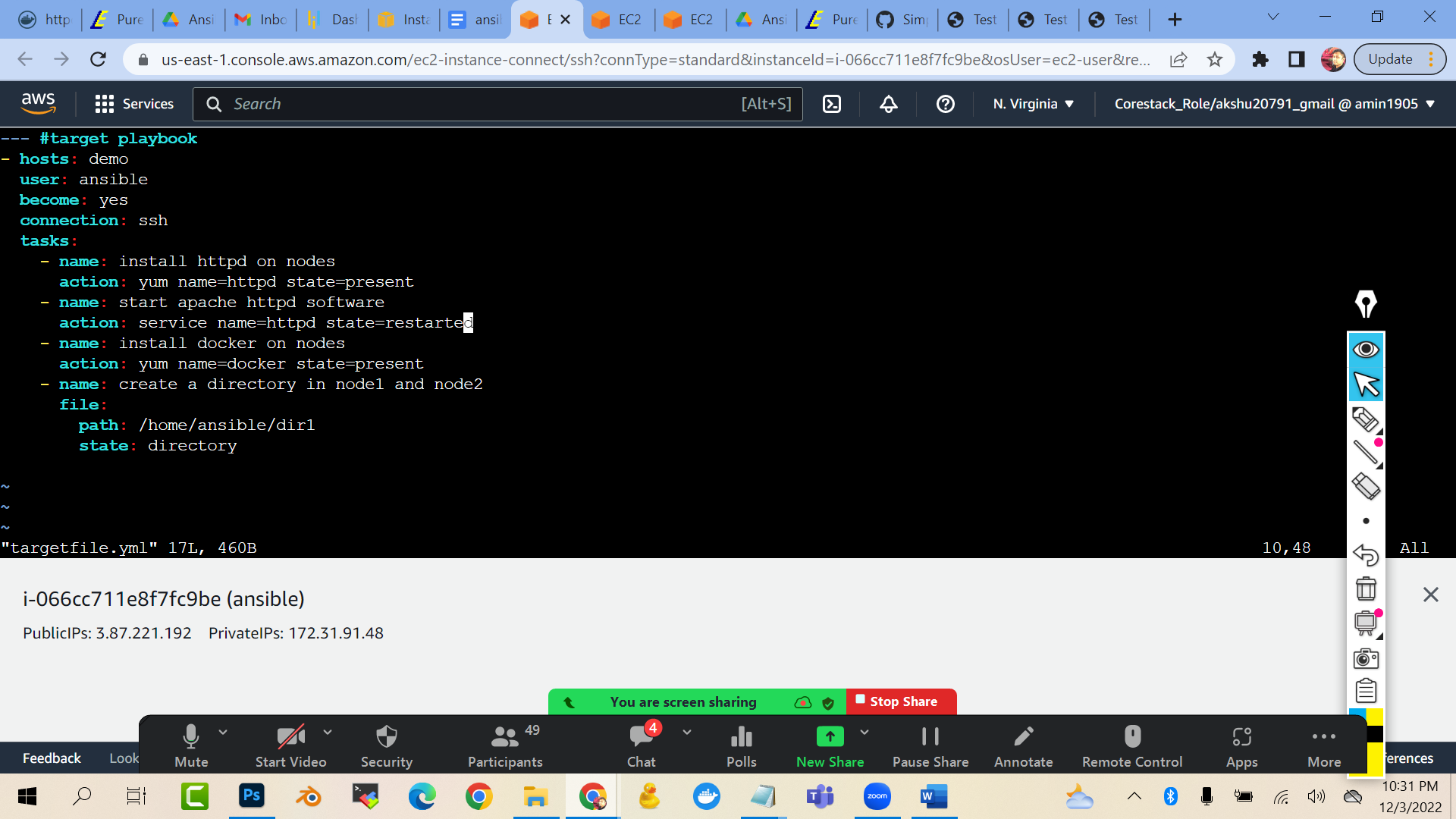




Now if you go to the nodes and check if directory is created you will see that directory with name dir1 is created

NOW , lets we install httpd …then we need to start httpd also to run the webserver(httpd apache server)

vi targetfile.yml



--- #target playbook

- hosts: demo

user: ansible

become: yes

connection: ssh

tasks:

- name: install httpd on nodes

action: yum name=httpd state=present

- name: start apache httpd software

action: service name=httpd state=restarted

- name: install docker on nodes

action: yum name=docker state=present

- name: create a directory in node1 and node2

file:

path: /home/ansible/dir1

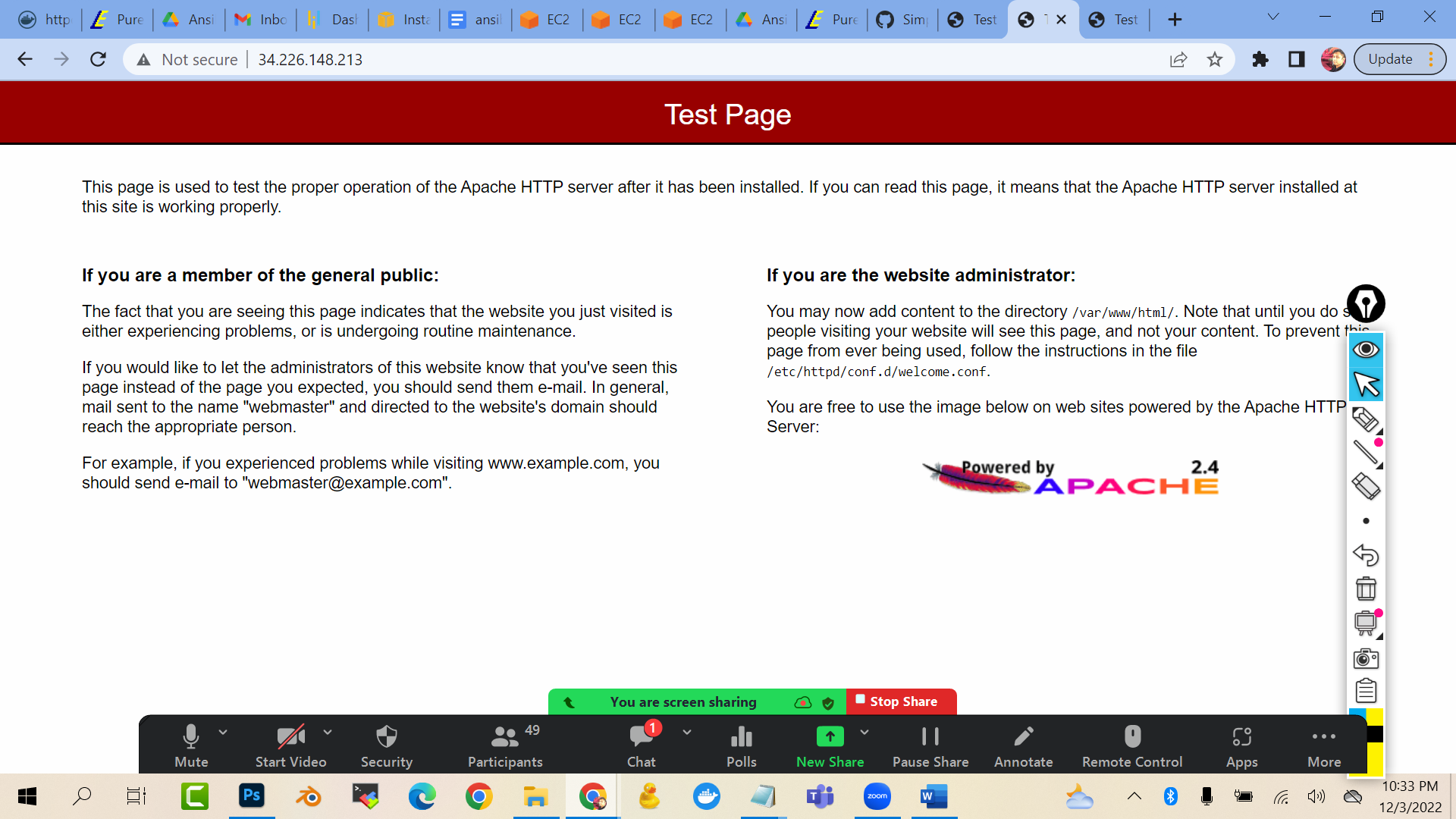
state: directory

(prwss esc :wq to quit)

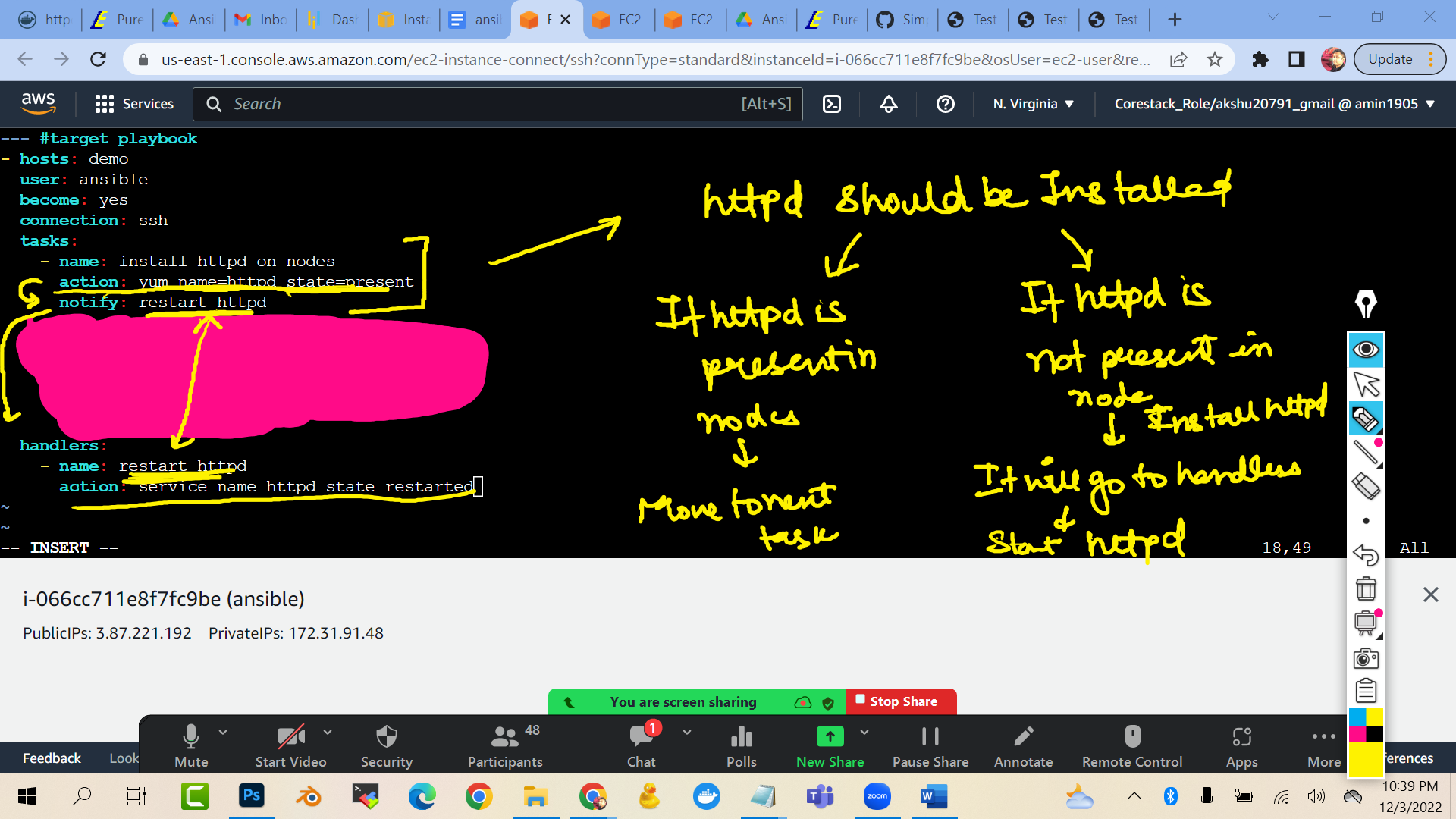
ansible-playbook targetfile.yml

(now you will see that when you run this command again and again …everytime your httpd will restart …but there should be some condition in which if httpd is installed then only httpd should be started otherwise it should not be started…this can be done handlers)

Also if you check the public ip of node1 or node 2



HANDLERS :



--- #target playbook

- hosts: demo

user: ansible

become: yes

connection: ssh

tasks:

- name: install httpd on nodes

action: yum name=httpd state=present

notify: restart httpd

- name: install docker on nodes

action: yum name=docker state=present

- name: create a directory in node1 and node2

file:

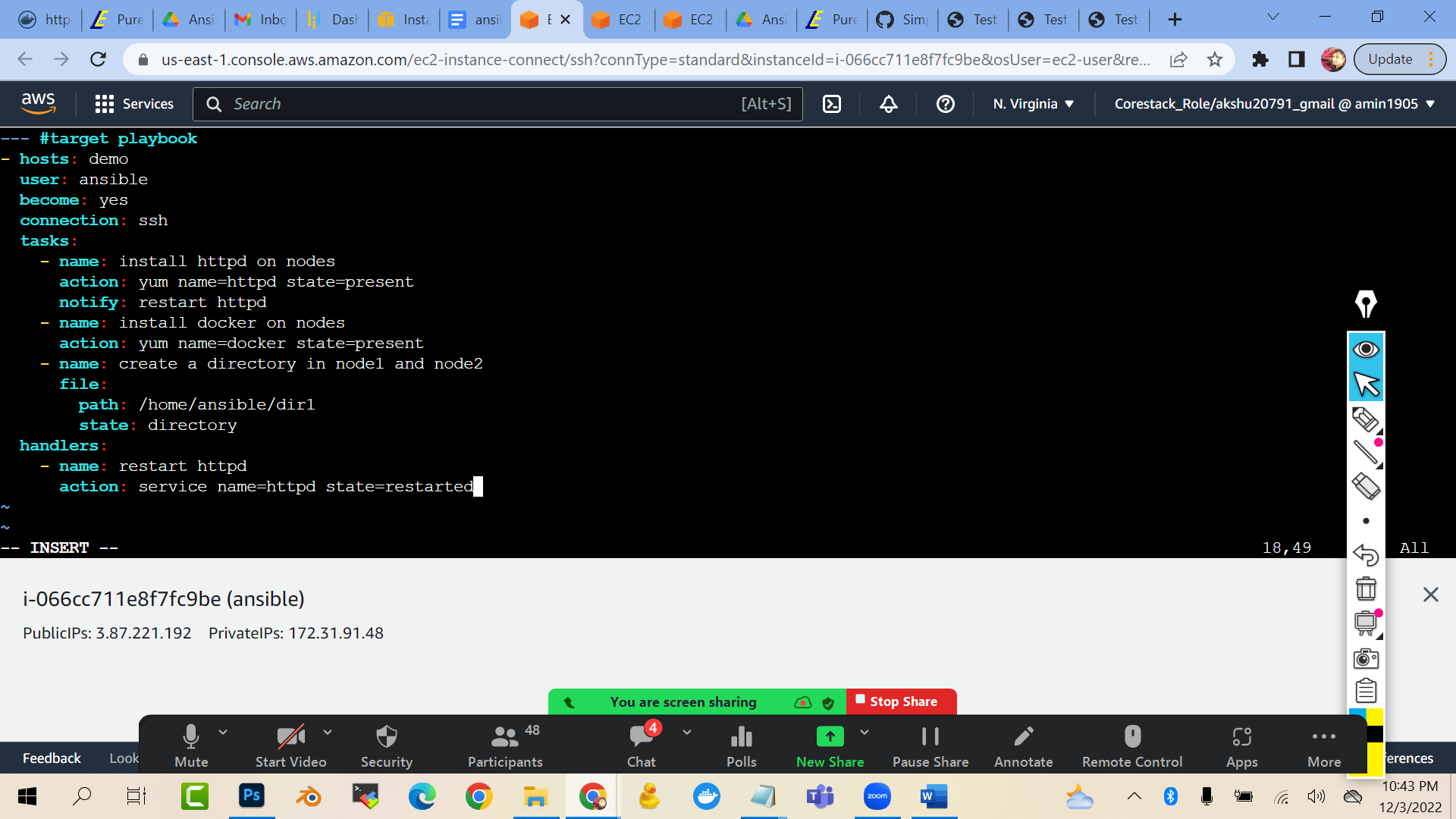
path: /home/ansible/dir1

state: directory

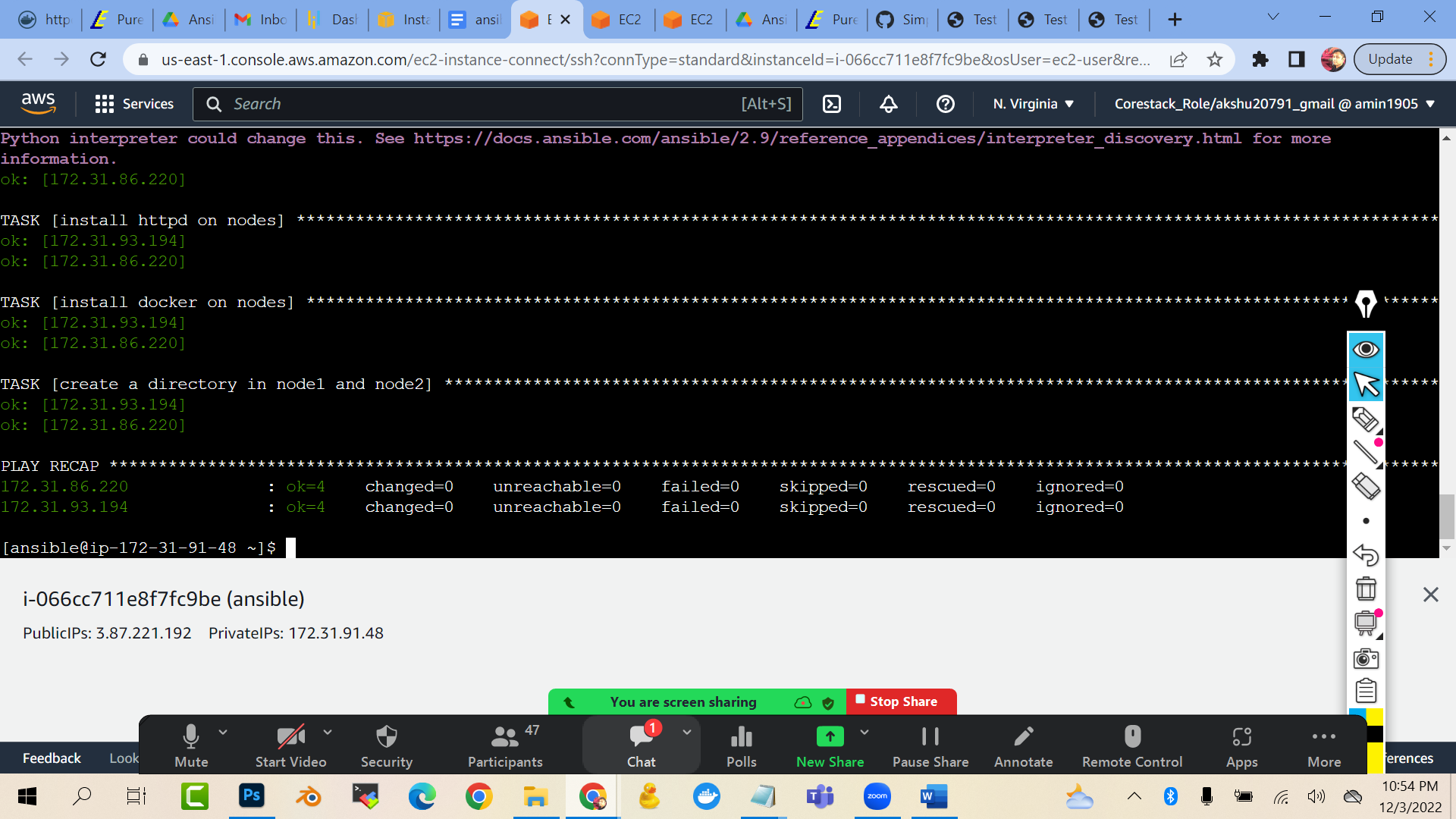
handlers:

- name: restart httpd

action: service name=httpd state=restarted



This will put a conditional statement that only when the httpd in installed then only handler will start the httpd in the nodes

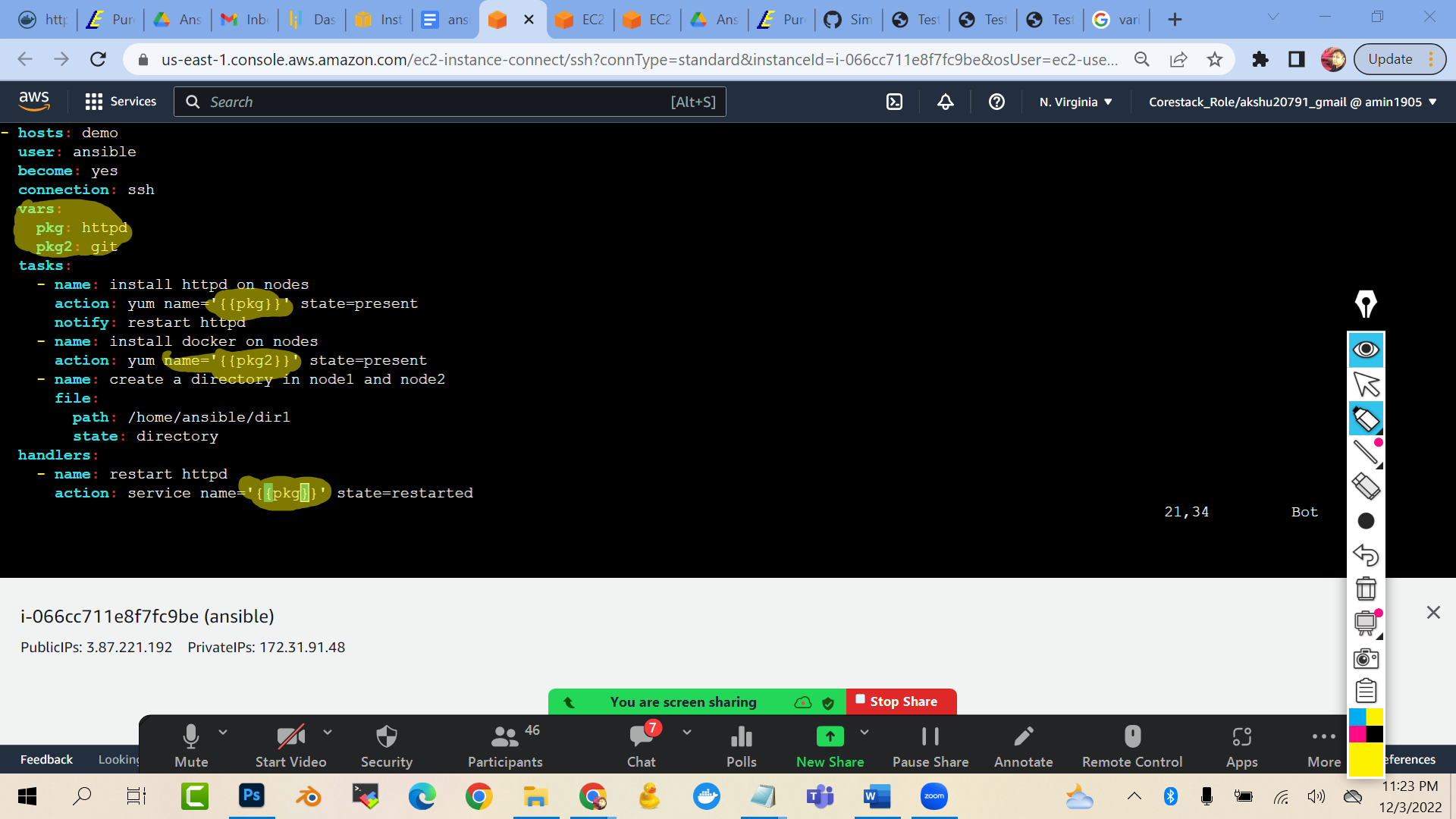


IF WE WANT THE PLAYBOOK TO TEST IF EVERYTHING IS FINE ..AND THE CONTENT OF MY PLAYBOOK IS FINE WITHOUT EXECUTING THE STEPS ACTUALLY IN NODES…WE CAN DO A DRY RUN

ansible-playbook task.yml --check

Concept of variables :

Ansible uses variables **to manage differences between systems**. With Ansible, you can execute tasks and playbooks on multiple different systems with a single command. To represent the variations among those different systems, you can create variables with standard YAML syntax, including lists and dictionaries.



--- #target playbook

- hosts: demo

user: ansible

become: yes

connection: ssh

vars:

pkg: httpd

pkg2: git

tasks:

- name: install httpd on nodes

action: yum name='{{pkg}}' state=present

notify: restart httpd

- name: install docker on nodes

action: yum name='{{pkg2}}' state=present

- name: create a directory in node1 and node2

file:

path: /home/ansible/dir1

state: directory

handlers:

- name: restart httpd

action: service name='{{pkg}}' state=restarted

ansible-playbook targetfile.yml