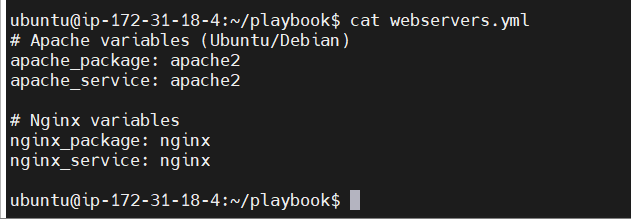
1. Write a single ansible playbook which will install apache and nginx.  
   Note: Playbook should not be hardcoded and pass the variables from different file.

Variable:webservers.yml



Playbook:webserver\_install.yml

---

- name: Install Apache and Nginx on Ubuntu

hosts: all

become: yes

vars\_files:

- webservers.yml

tasks:

- name: Install Apache

ansible.builtin.package:

name: "{{ apache\_package }}"

state: present

update\_cache: yes

- name: Enable and start Apache

ansible.builtin.service:

name: "{{ apache\_service }}"

state: started

enabled: yes

- name: Install Nginx

ansible.builtin.package:

name: "{{ nginx\_package }}"

state: present

update\_cache: yes

- name: Change Nginx to listen on port 8080 (IPv4)

ansible.builtin.replace:

path: /etc/nginx/sites-available/default

regexp: 'listen 80 default\_server;'

replace: 'listen 8080 default\_server;'

backup: yes

notify: Validate nginx

- name: Change Nginx to listen on port 8080 (IPv6)

ansible.builtin.replace:

path: /etc/nginx/sites-available/default

regexp: 'listen \[::\]:80 default\_server;'

replace: 'listen [::]:8080 default\_server;'

backup: yes

notify: Validate nginx

- name: Enable Nginx service (won't start yet)

ansible.builtin.service:

name: "{{ nginx\_service }}"

enabled: yes

handlers:

- name: Validate nginx

ansible.builtin.command: nginx -t

notify: Restart nginx

failed\_when: "'successful' not in nginx\_check.stdout"

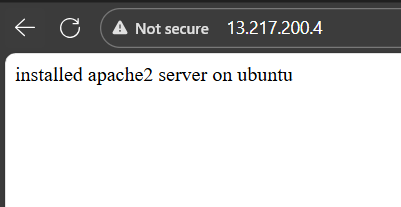
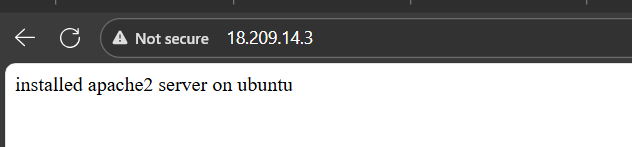
register: nginx\_check

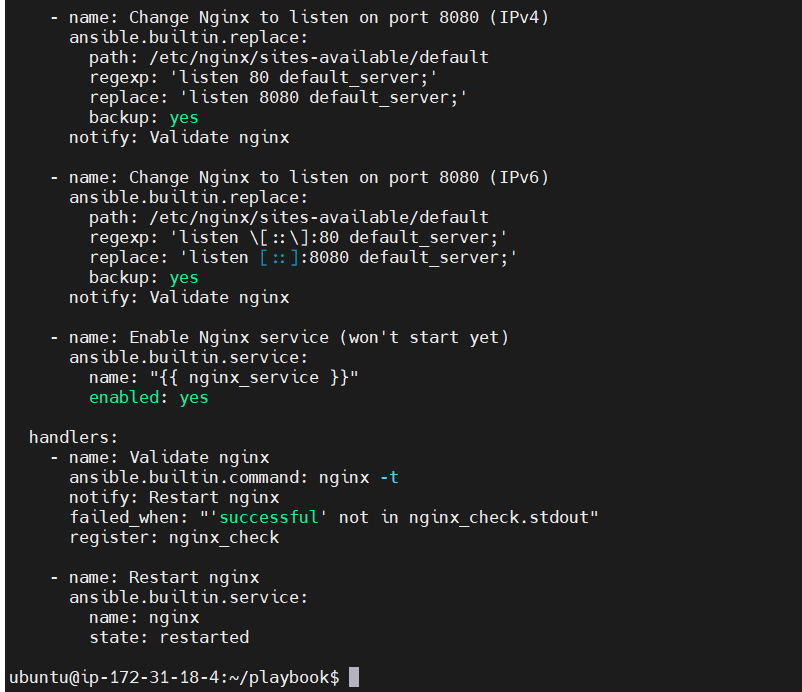
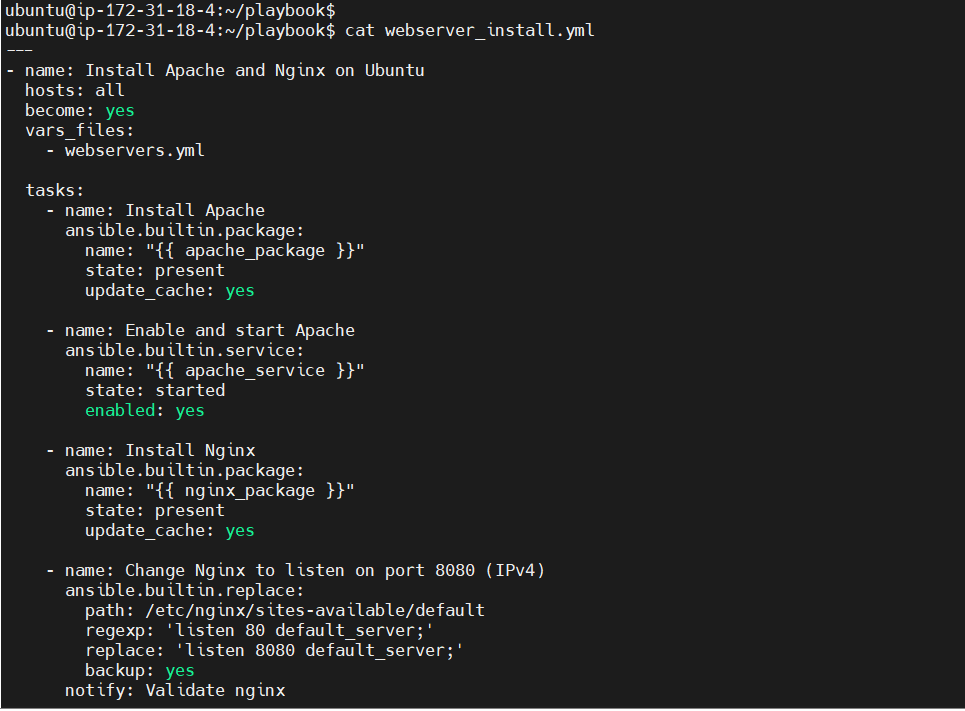
- name: Restart nginx

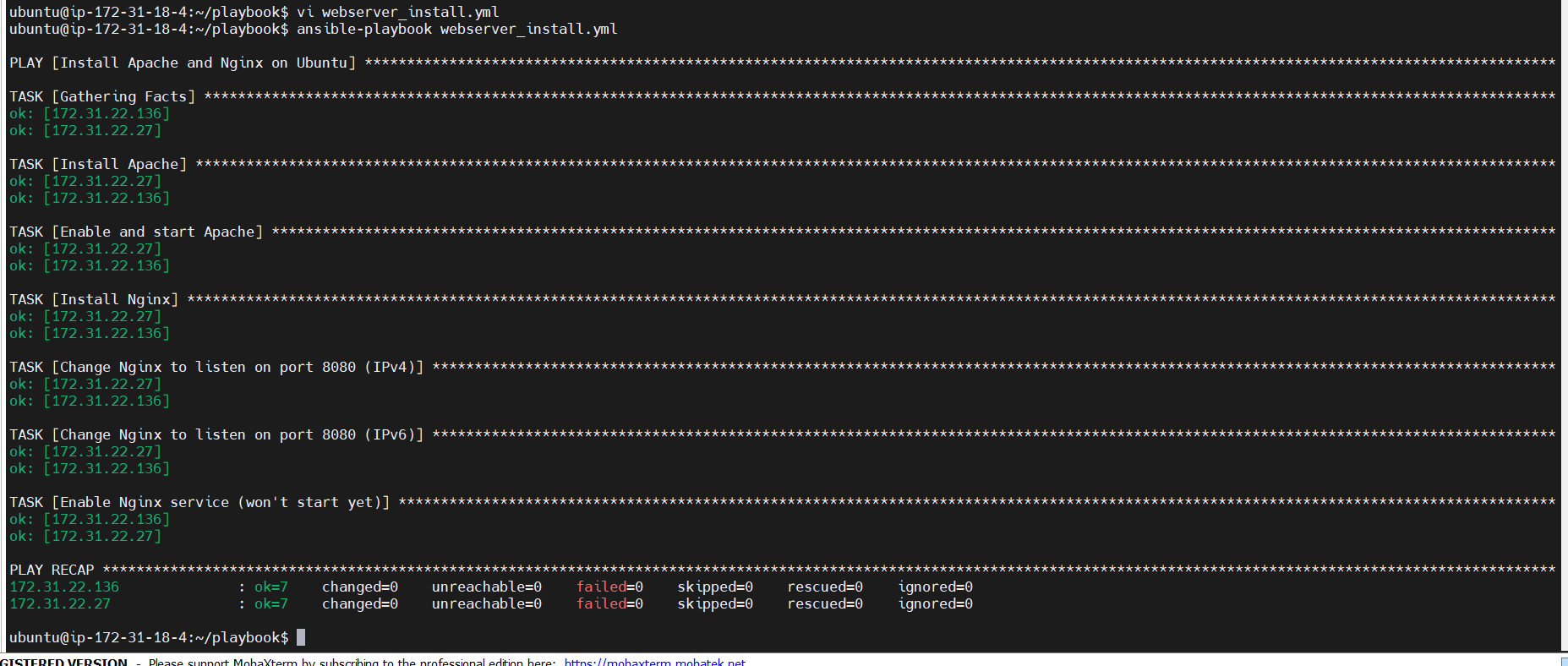
ansible.builtin.service:

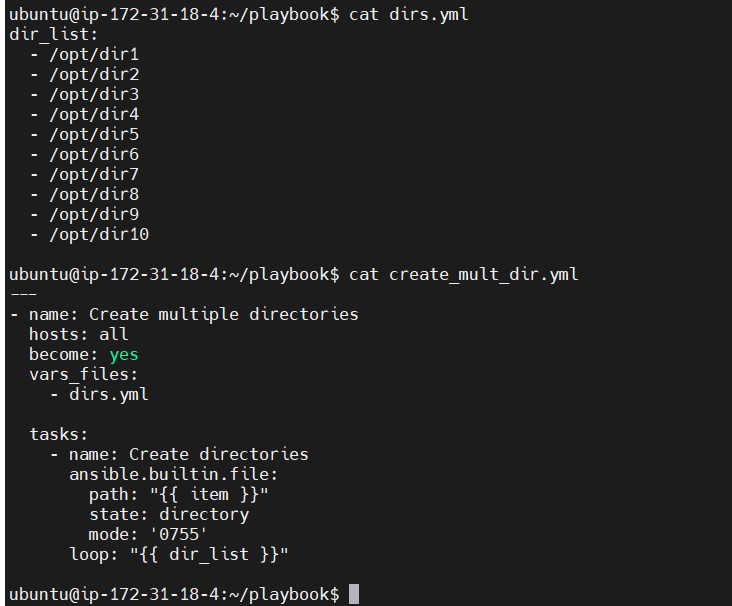
name: nginx

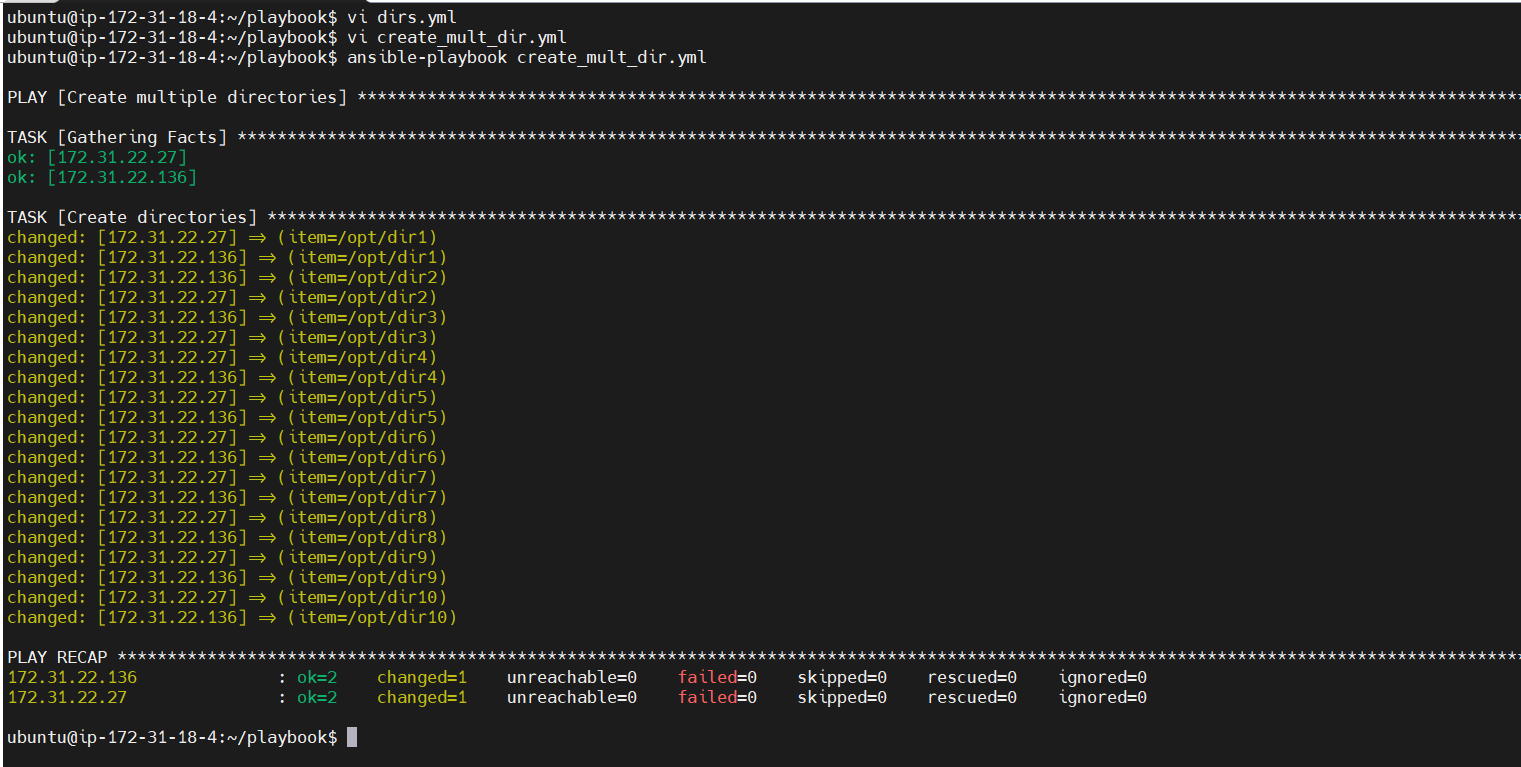
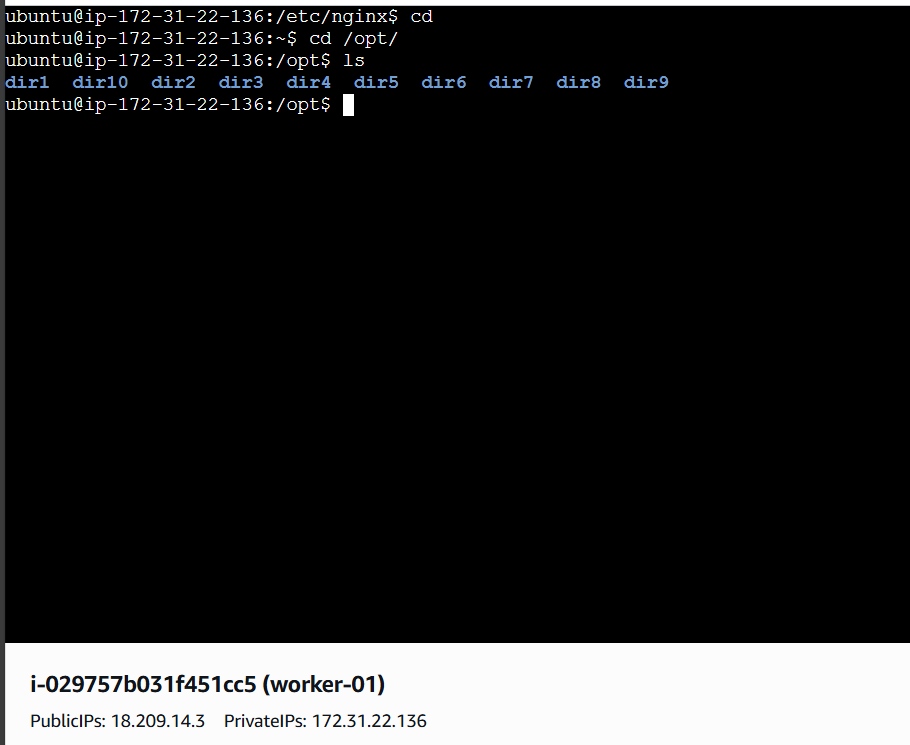
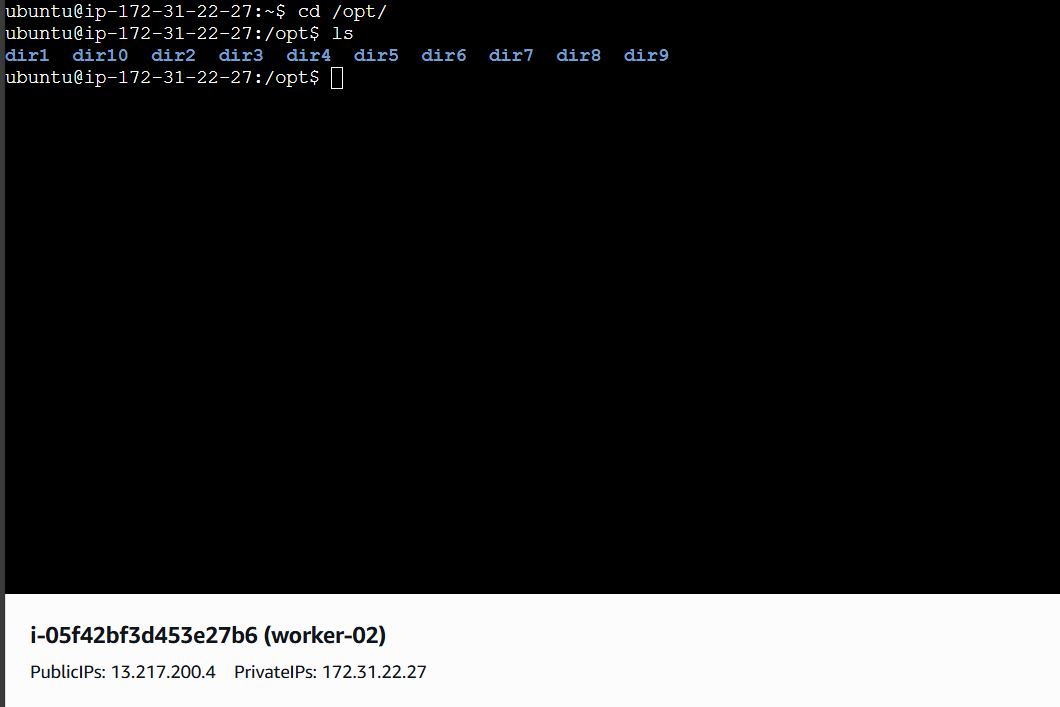
state: restarted



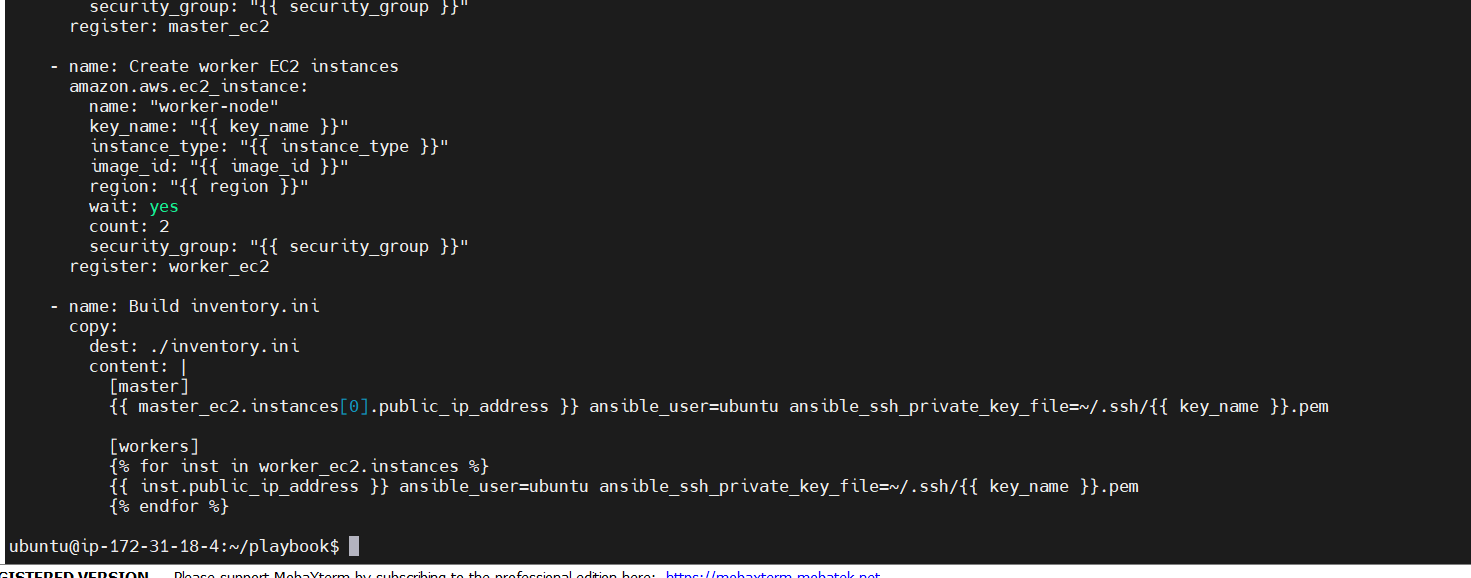
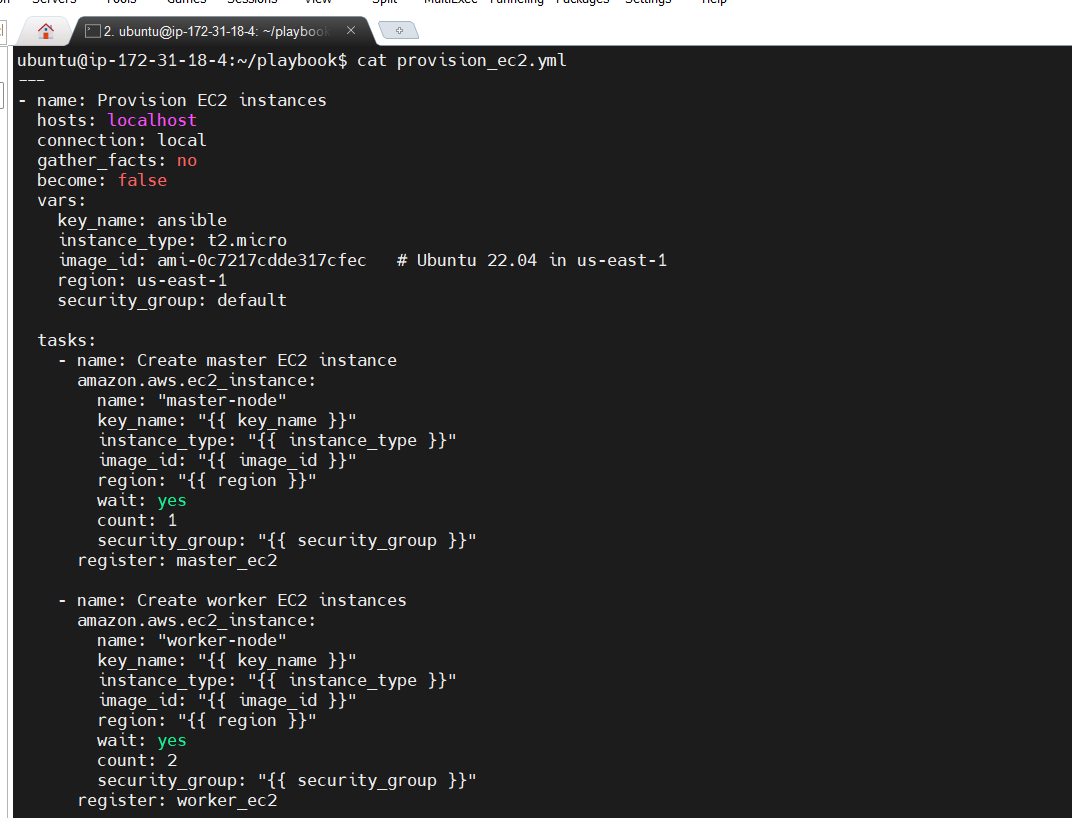


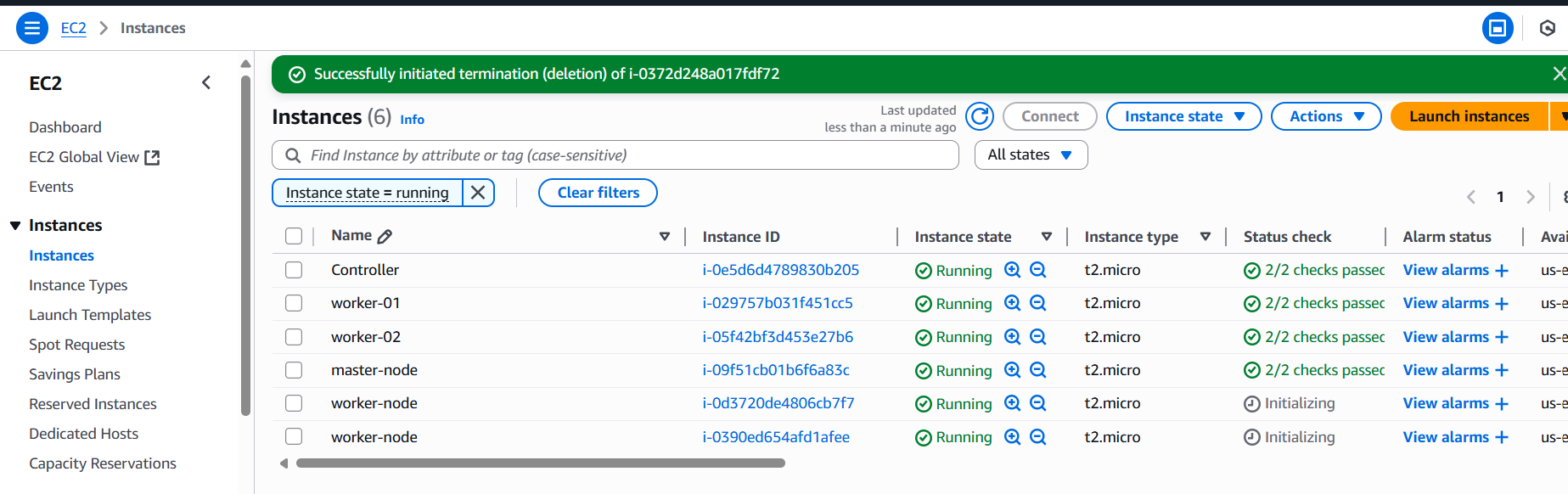
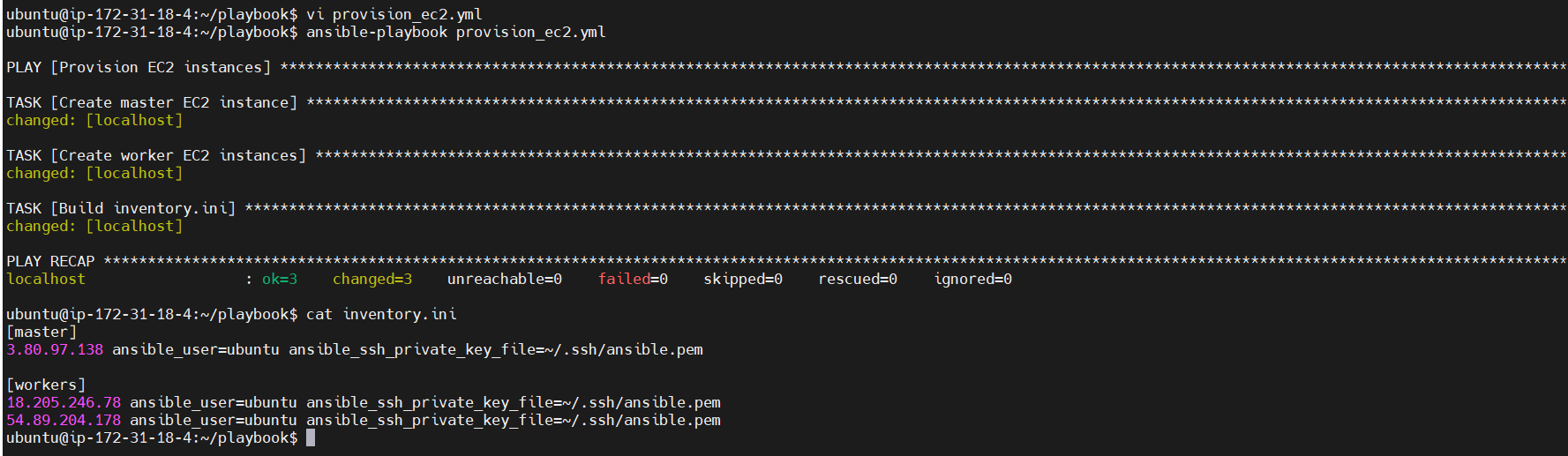
  
2) Ansible playbook to create 10 different directories with minimal code and directory names should be  
passed as variables.



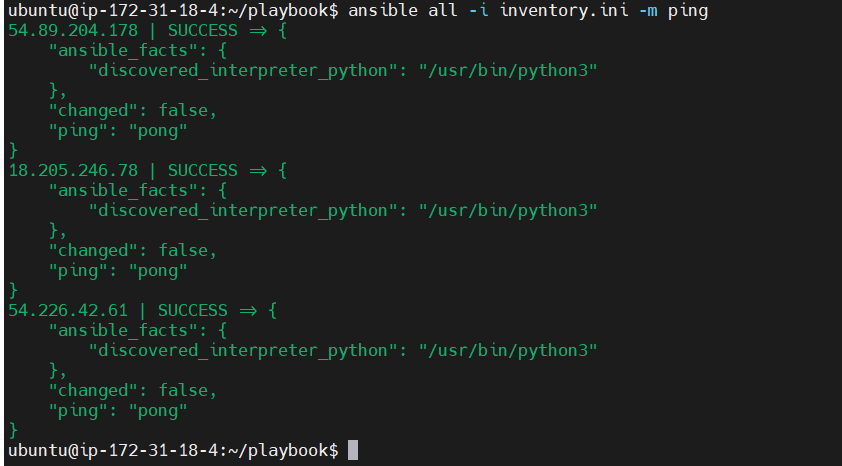
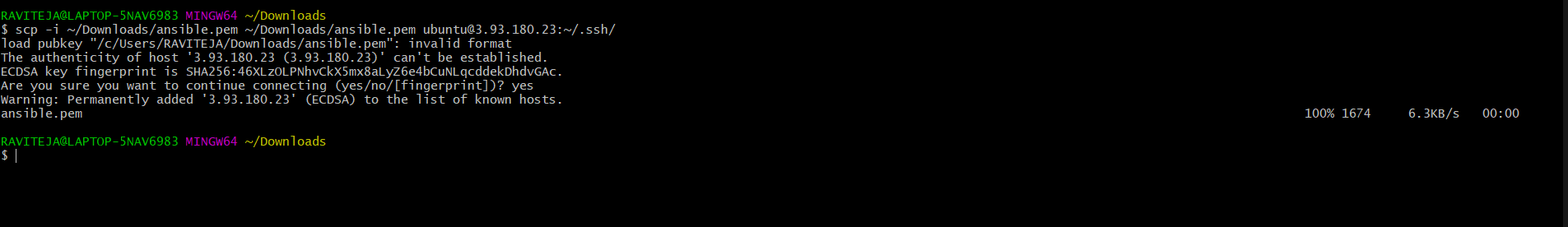
  

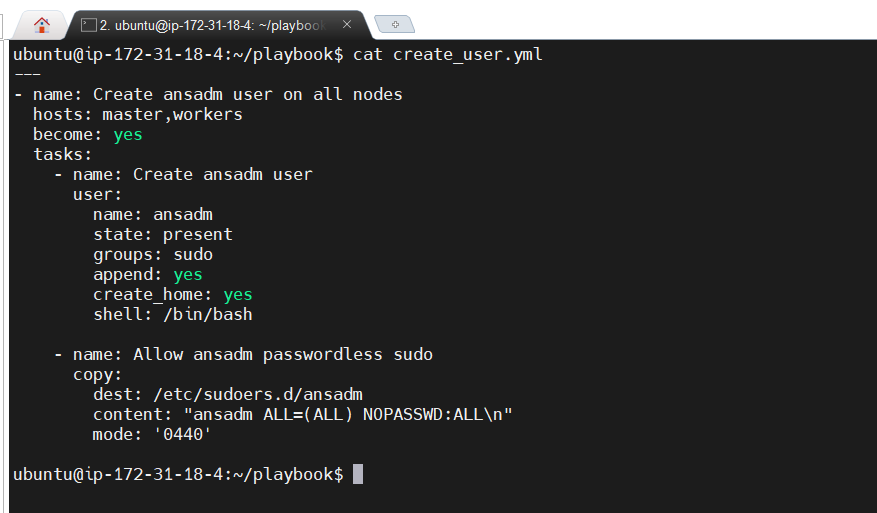
3) Ansible playbook to copy ssh-keygen from master to worker nodes.  
Note:  
a)Provision new 3 ec2 machines, one master and two worker nodes.

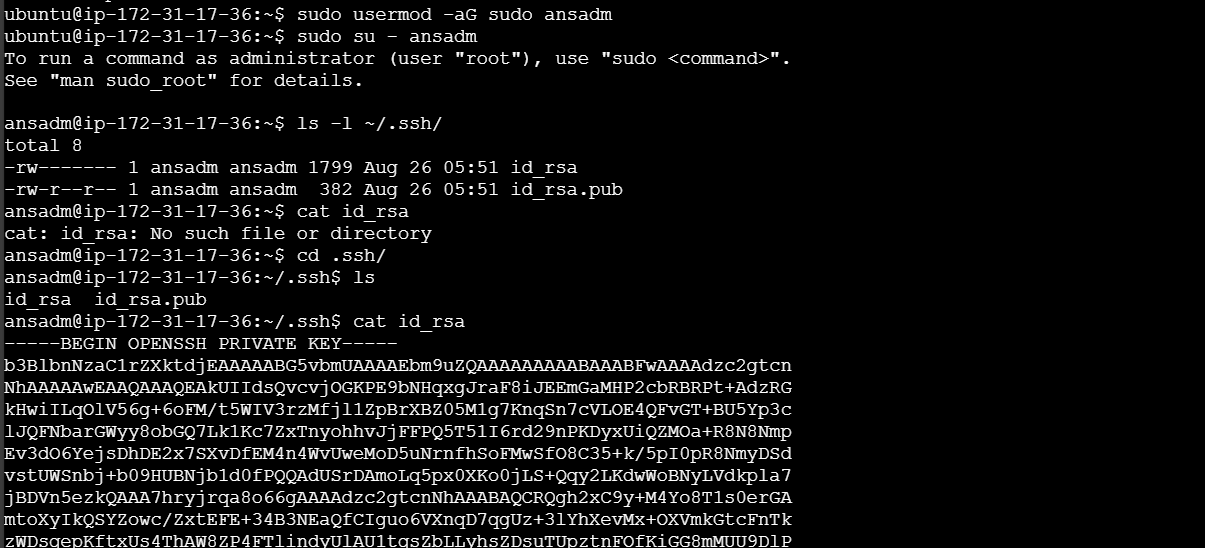
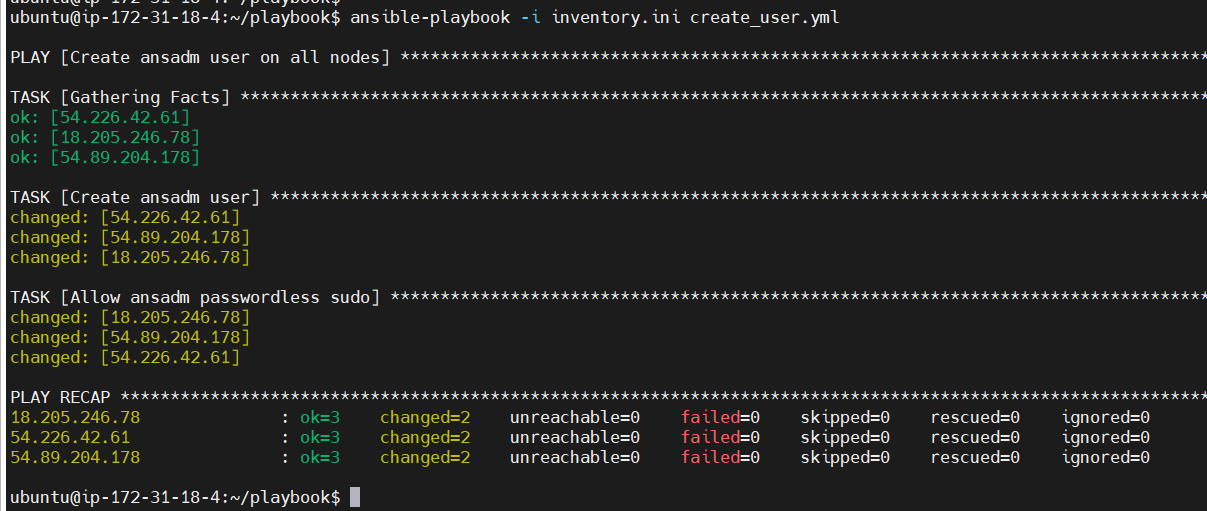




Copied ansible.pem key to controller\_node (.ssh folder)

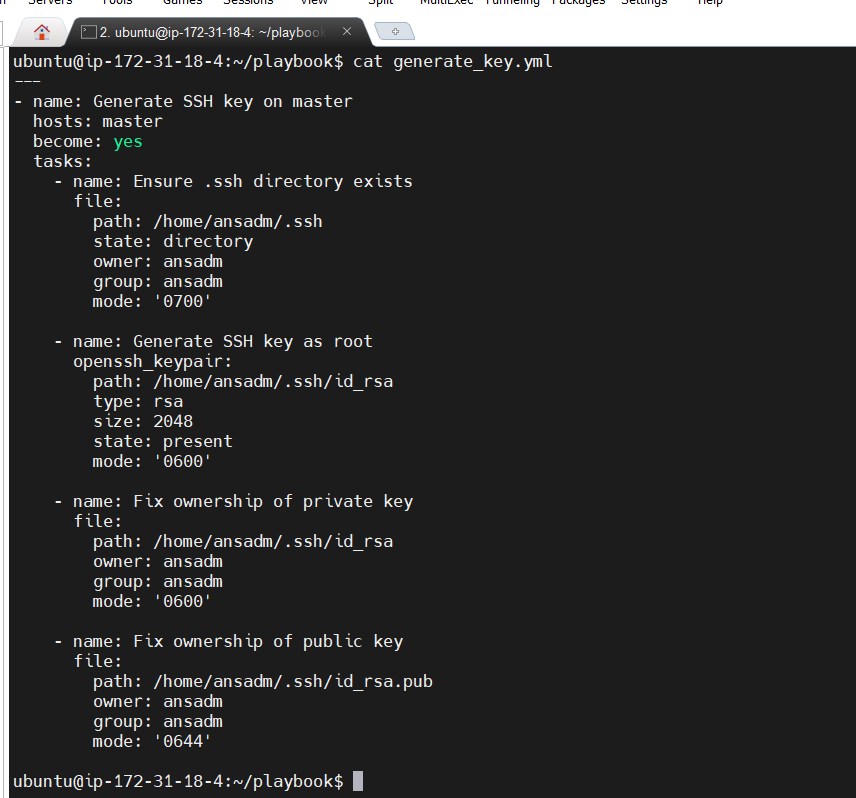
  
b)Create common user called ansadm and provide sudo priviliges on 3 ec2 instances.

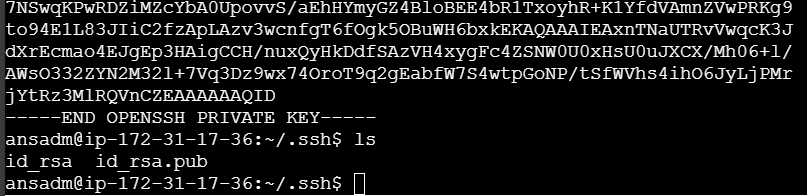
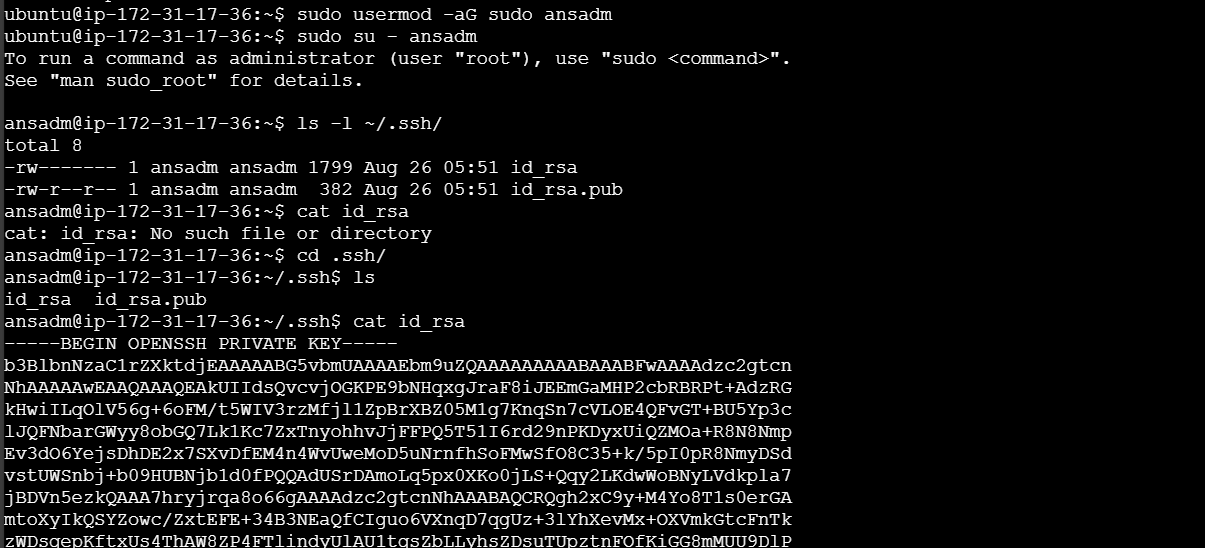
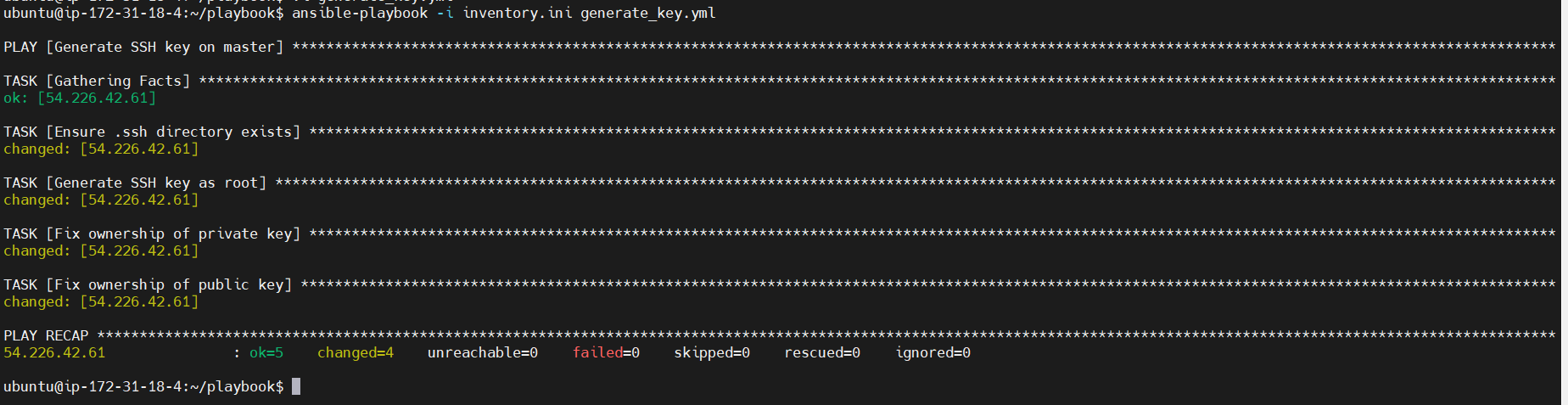




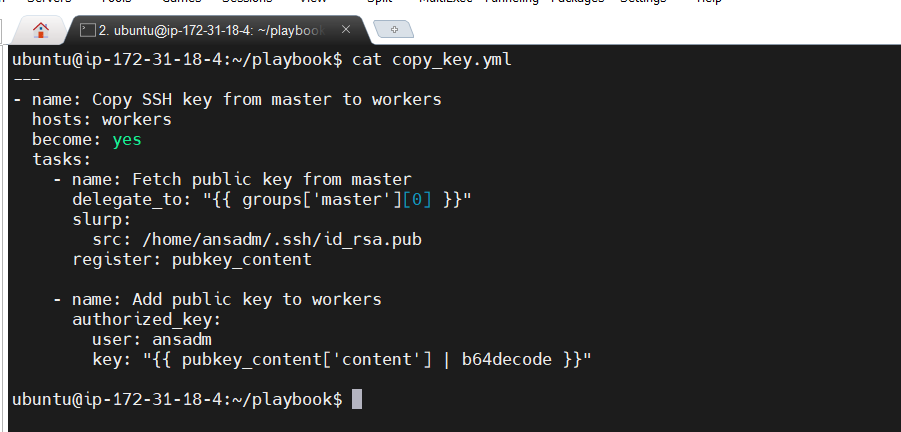
c)Create ssh-keygen in master and your playbook should copy the keygen making it password less  
authentication.

Generate ssh-keygen:

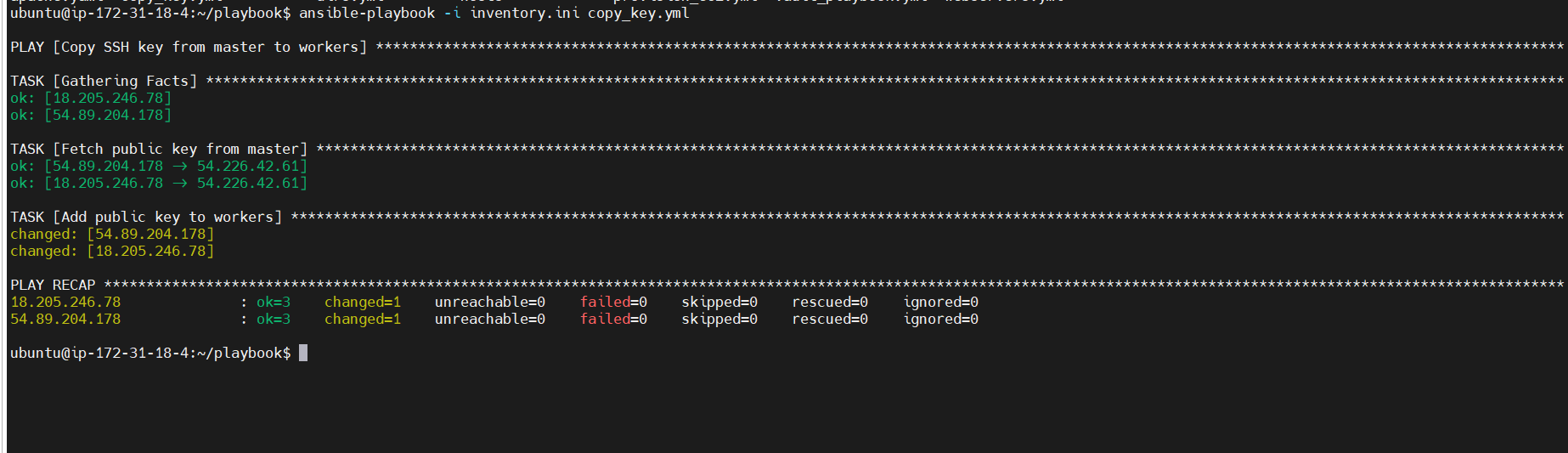




Copied pub\_key on workers



 to make password less authentication



4) Ansible playbook to inject ansible vault variables.

Vault.yml

---

db\_user: mydbuser

db\_password: mysecretpassword

