**Ansible-task-4**

1. **Create ansible playbook to create ansible vault.**
2. **Write a ansible playbook to install apache in linux and ubuntu machine by using when condition.**
3. **Create ansible playbook using roles to configure LAMP stack**
4. **Setup ansible AWX and explore the options.**

**1.Create**

**ansible-vault create secret.yml**

**---**

**- name: Create Ansible Vault file**

**hosts: localhost**

**gather\_facts: false**

**vars:**

**vault\_file\_path: "/home/ubuntu/my\_vault.yml"**

**vault\_content: |**

**secret\_key: "my\_secret\_value"**

**tasks:**

**- name: Create vault file with secret**

**copy:**

**content: "{{ vault\_content }}"**

**dest: "/tmp/temp\_secret.yml"**

**- name: Encrypt the vault file**

**command: >**

**ansible-vault encrypt /tmp/temp\_secret.yml**

**--output {{ vault\_file\_path }}**

**--vault-password-file /home/ubuntu/.vault\_pass.txt**

**- name: Remove temporary unencrypted file**

**file:**

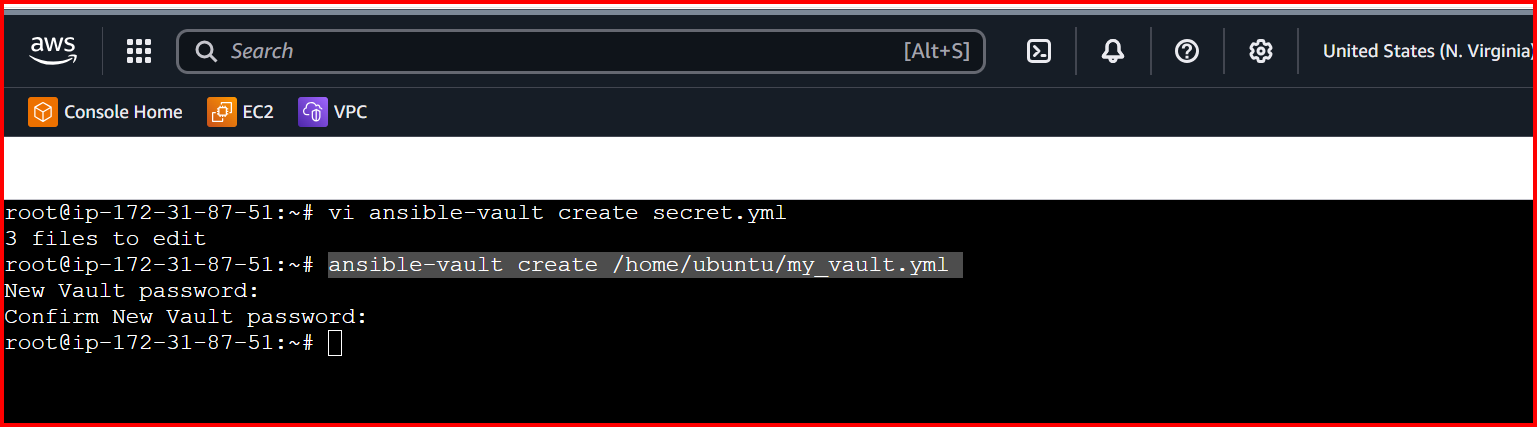
**path: /tmp/temp\_secret.yml**

**state: absent**

**Execute this command :**

**ansible-vault create /home/ubuntu/my\_vault.yml**

**create password**

****

**2.**

**---**

**- name: Install Apache web server**

**hosts: all**

**become: yes**

**gather\_facts: yes**

**tasks:**

**- name: Install Apache on Ubuntu/Debian**

**apt:**

**name: apache2**

**state: present**

**update\_cache: yes**

**when: ansible\_facts['os\_family'] == "Debian"**

**- name: Ensure Apache service is running on Ubuntu/Debian**

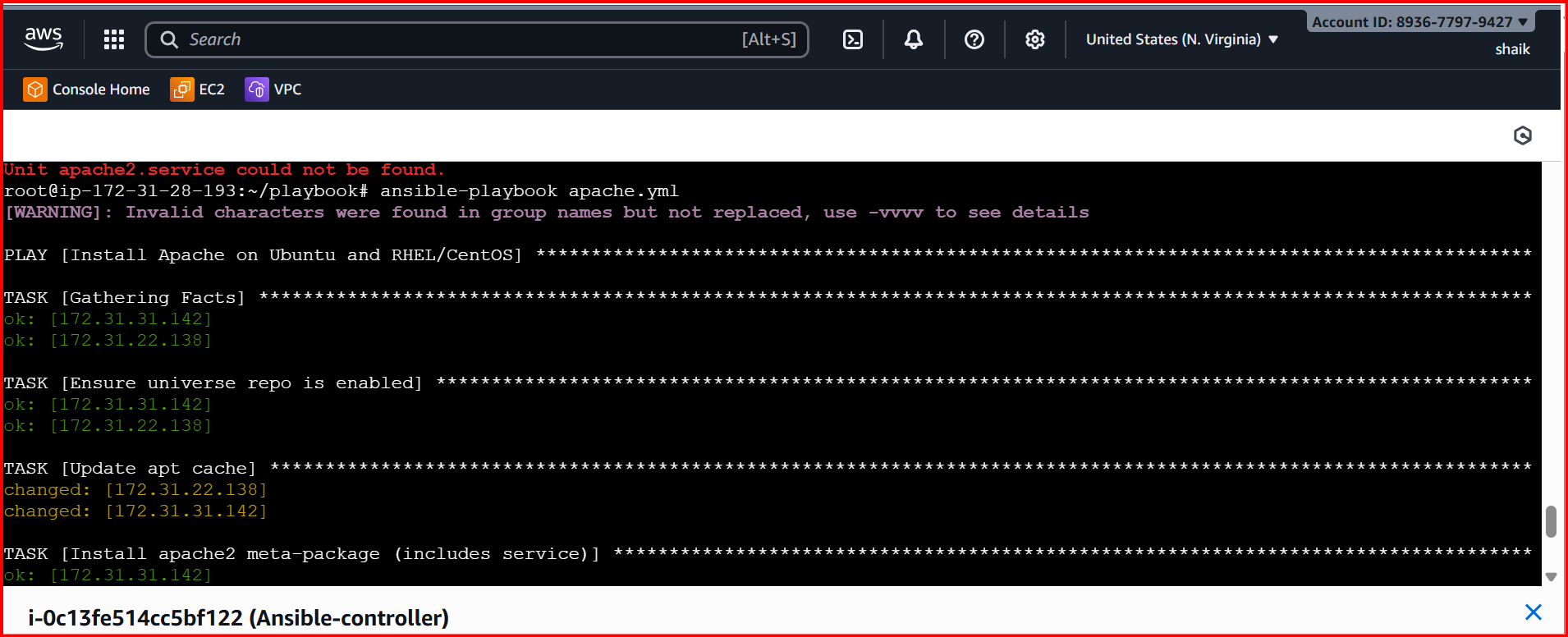
**service:**

**name: apache2**

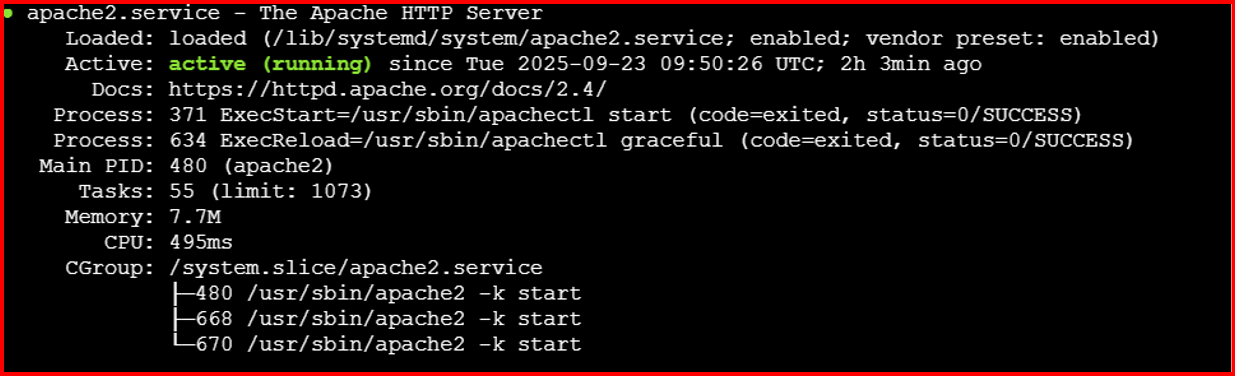
**state: started**

**enabled: yes**

**when: ansible\_facts['os\_family'] == "Debian"**

****

**Worker machine-1**

****

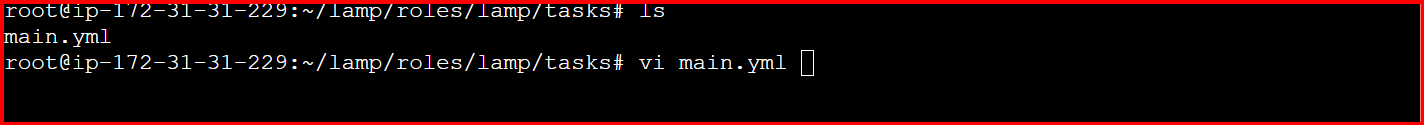
**3.**

### **Steps**

1. **Create role:**

**ansible-galaxy init lamp**

**from root : vi lamp/tasks/main.yml**

****

**---**

**# tasks file for lamp role**

**- name: Install Apache**

**apt:**

**name: apache2**

**state: present**

**update\_cache: yes**

**when: ansible\_distribution == "Ubuntu"**

**- name: Install MySQL**

**apt:**

**name: mysql-server**

**state: present**

**update\_cache: yes**

**when: ansible\_distribution == "Ubuntu"**

**- name: Install PHP and Apache module**

**apt:**

**name:**

**- php**

**- libapache2-mod-php**

**- php-mysql**

**state: present**

**update\_cache: yes**

**when: ansible\_distribution == "Ubuntu"**

**- name: Start and enable Apache**

**service:**

**name: apache2**

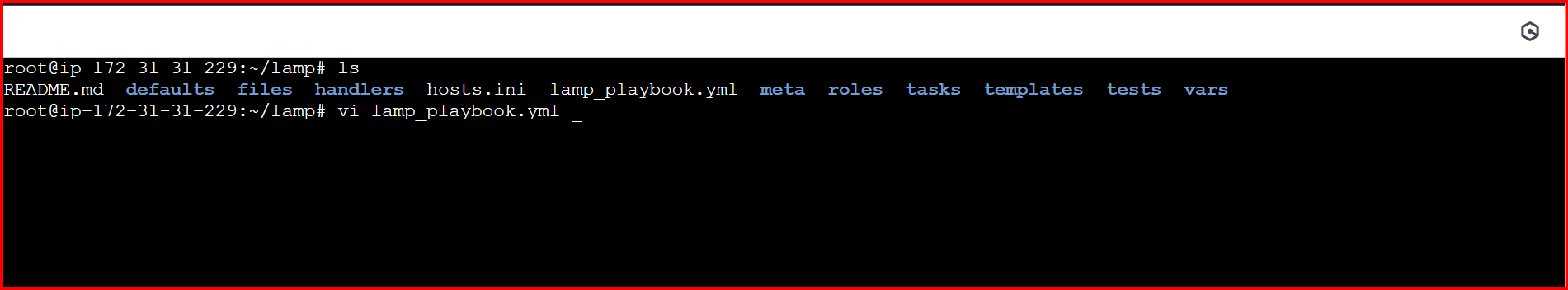
**state: started**

**enabled: yes**

**in the same path**

**create vi lamp\_playbook.yml**

**Both lamp file and lamp\_playbook.yml file are at same location**

****

**---**

**- name: Configure LAMP stack on all nodes**

**hosts: all**

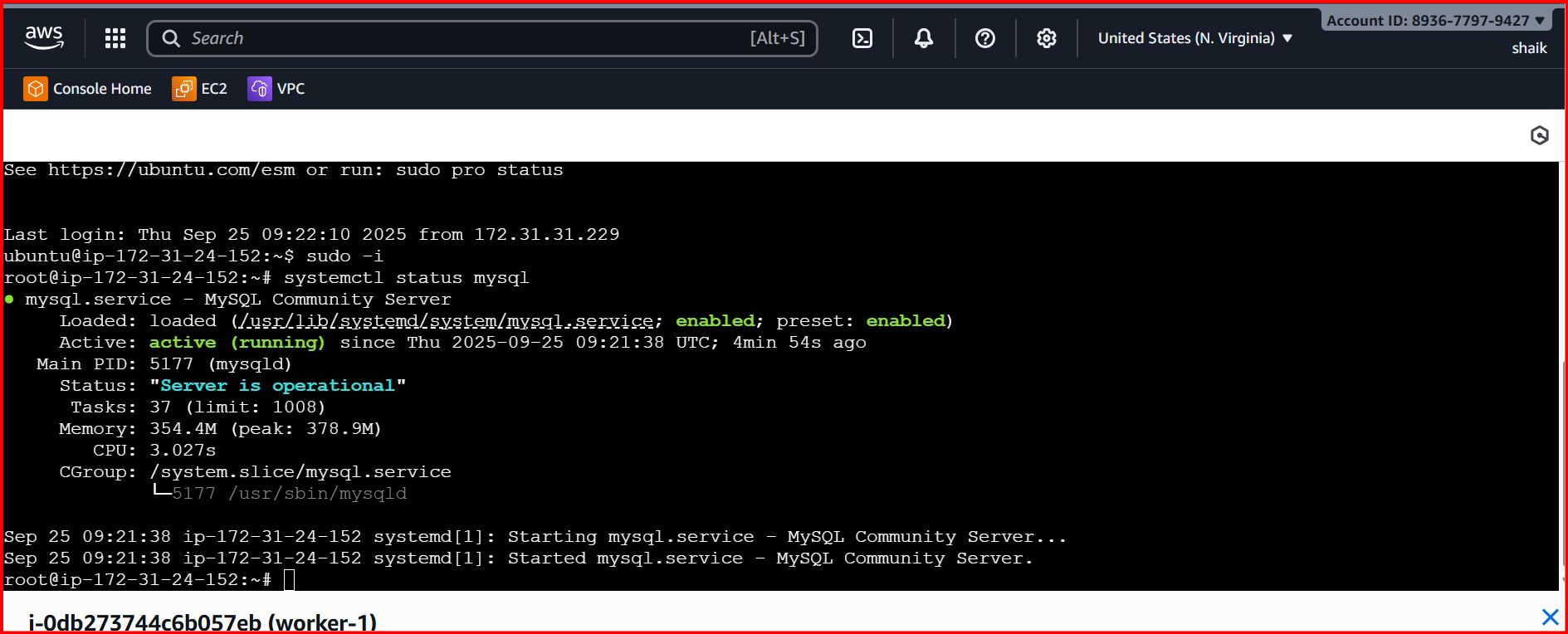
**become: yes**

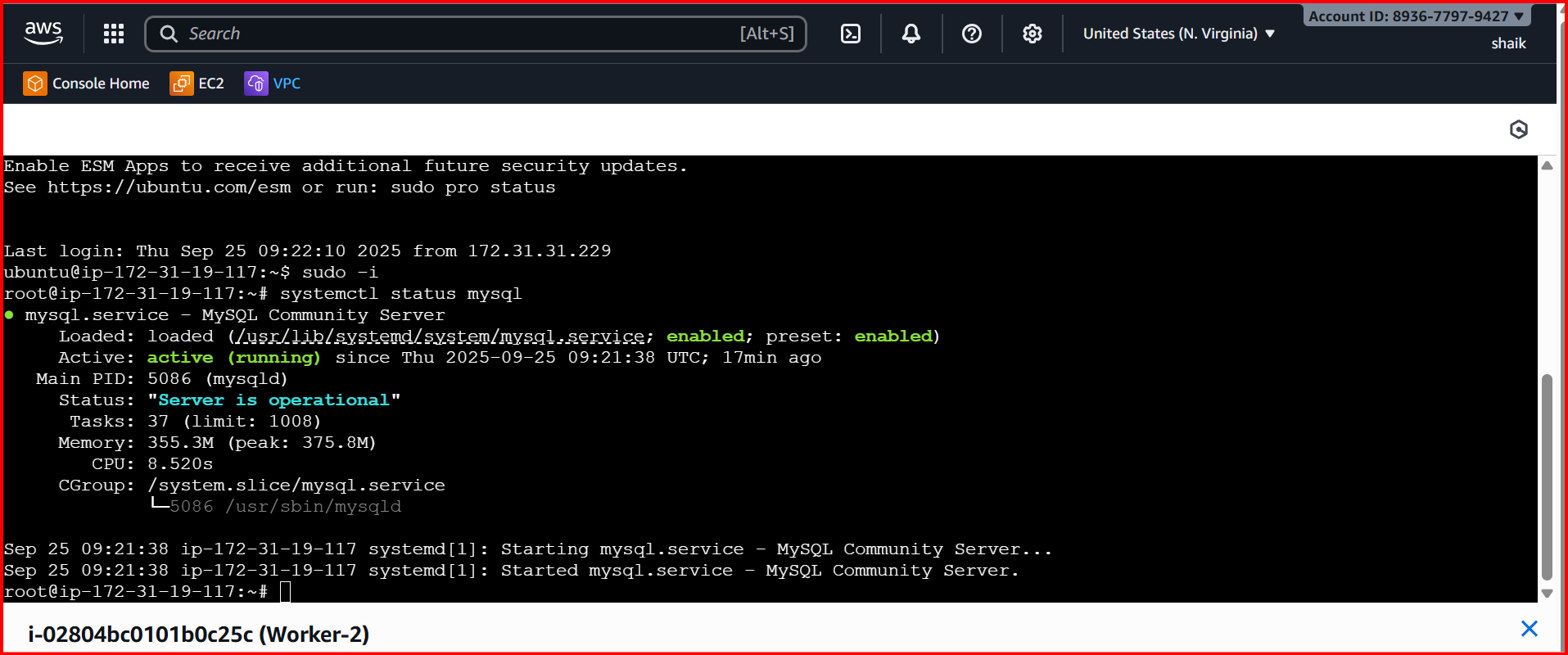
**roles:**

**- lamp**

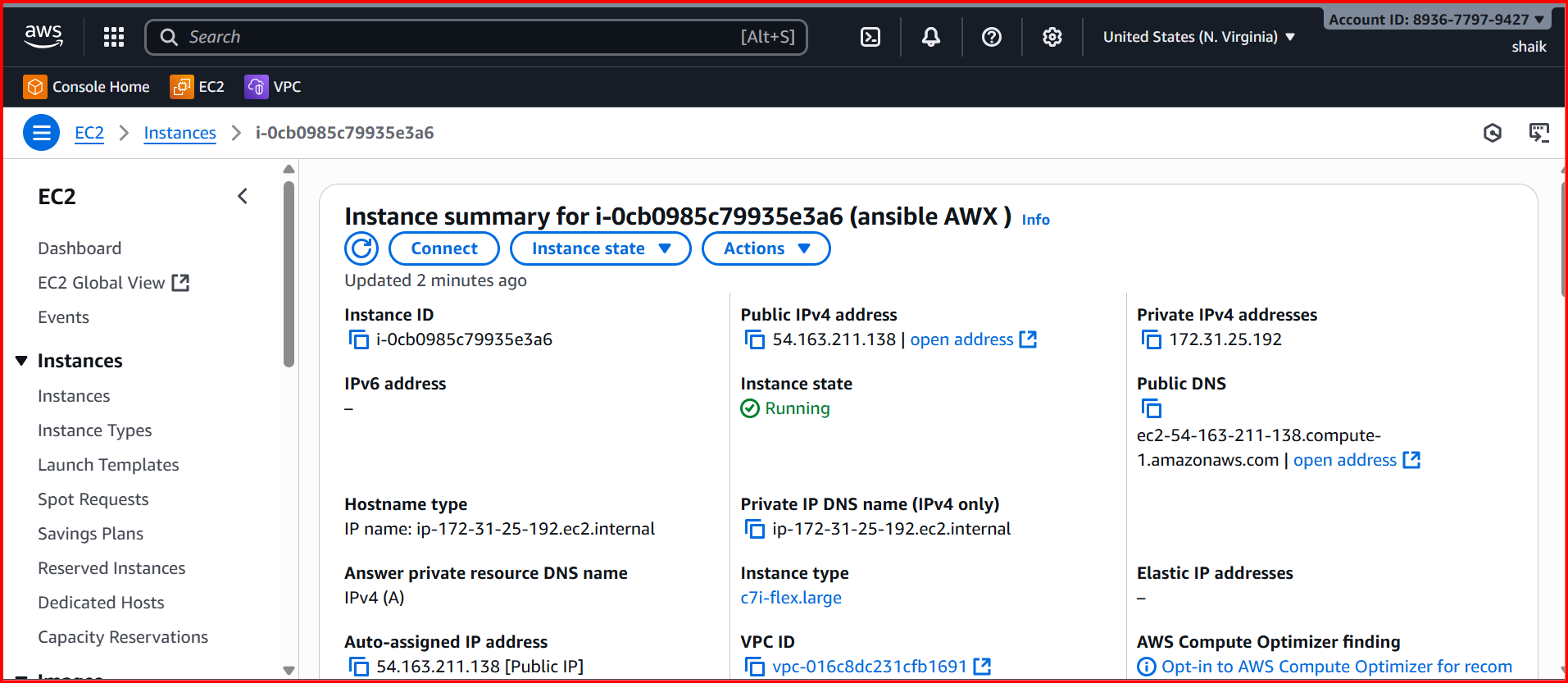
**Run this command:**

**ansible-playbook -i /root/lamp/hosts.ini /root/lamp/lamp\_playbook.yml**

****

****

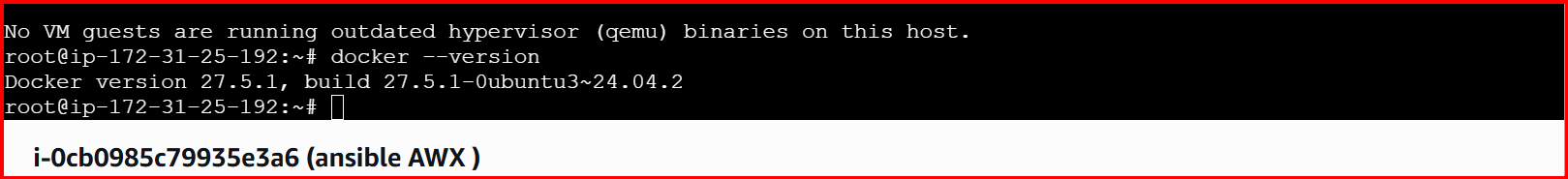
**4.Need to launch the new instabnce with instance type t2.large and storage as 20gb**

****

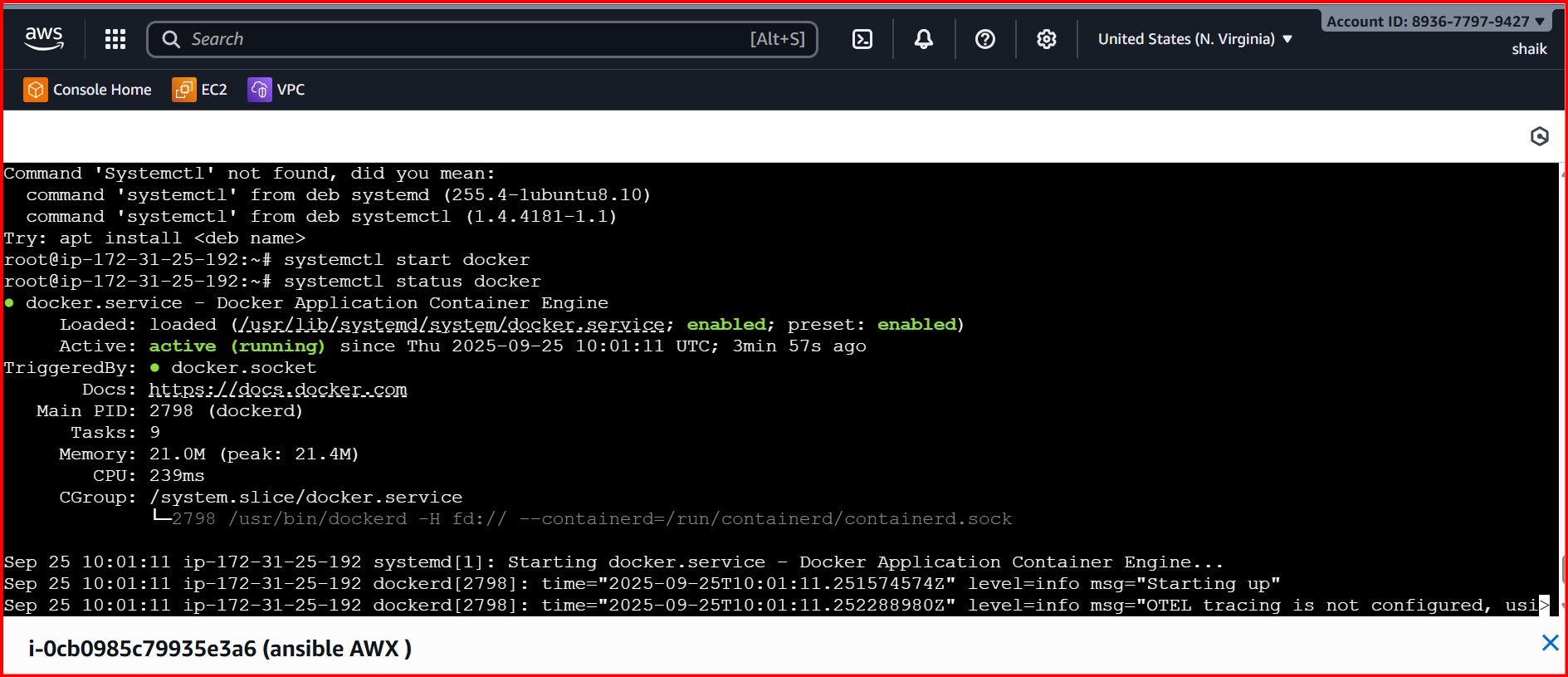
**apt update**

**apt install ansible**

**Now install docker ---> apt install docker.io**

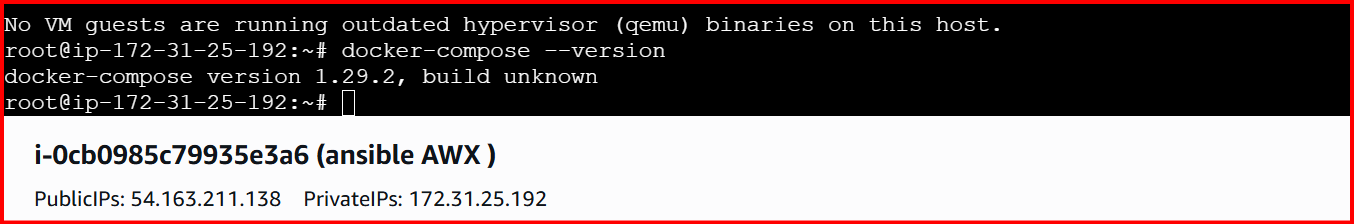
****

**Systenctl start docker**

****

**apt install docker-compose**

**docker-compose –version**

****

**git clone -b 17.1.0** [**https://github.com/ansible/awx.git**](https://github.com/ansible/awx.git)

**cd awx/**

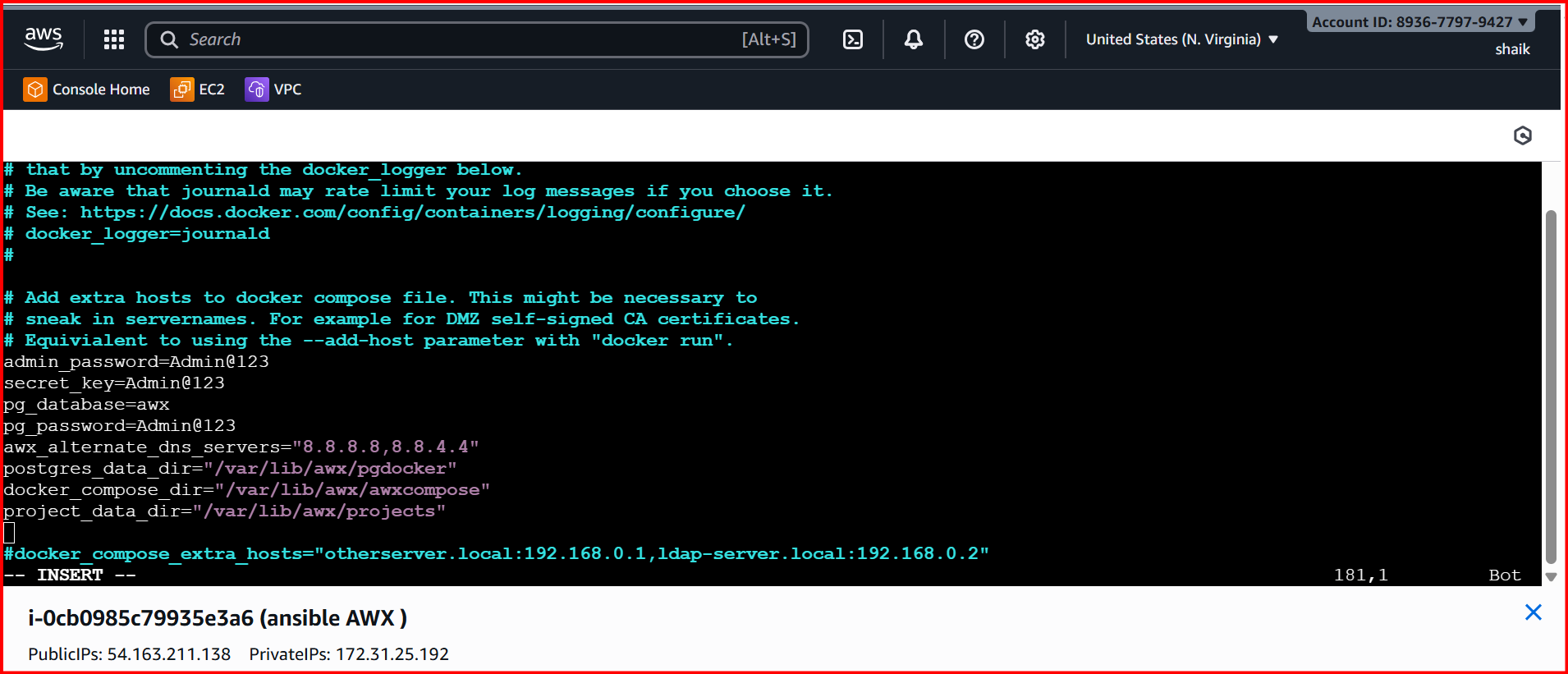
**cd installer**

**ls and go to vi inventory**

**last above line add this**

**admin\_password=Admin@123  
secret\_key=Admin@123  
pg\_database=awx  
pg\_password=Admin@123  
awx\_alternate\_dns\_servers="8.8.8.8,8.8.4.4"  
postgres\_data\_dir="/var/lib/awx/pgdocker"  
docker\_compose\_dir="/var/lib/awx/awxcompose"  
project\_data\_dir="/var/lib/awx/projects"**

**ansible-playbook -i ~/awx/installer/inventory ~/awx/installer/install.yml**

****

**Now check with public ip on browser we can config the AWX**

[**http://54.163.211.138/#/home**](http://54.163.211.138/#/home)

