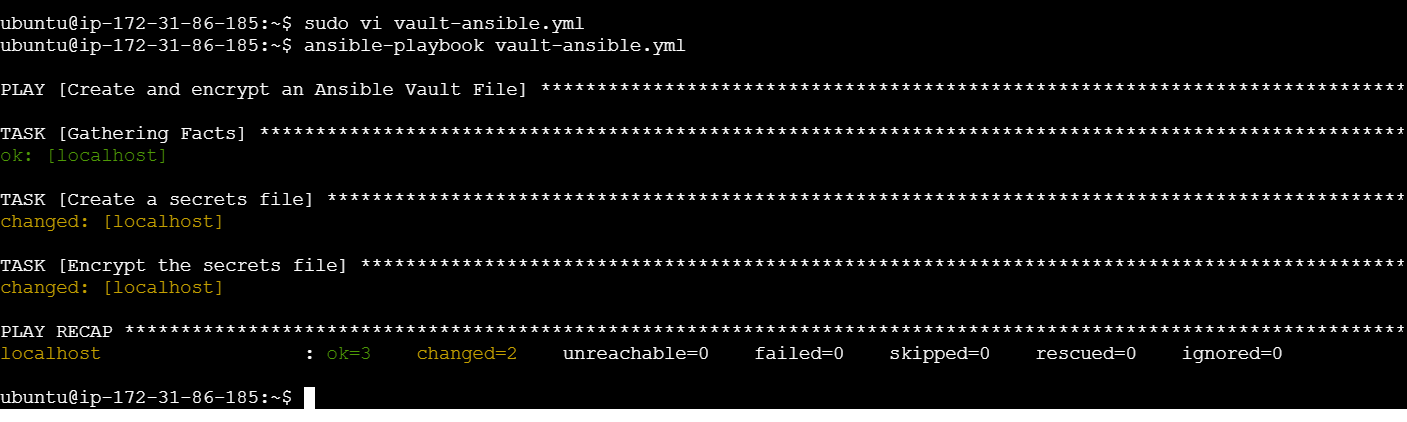
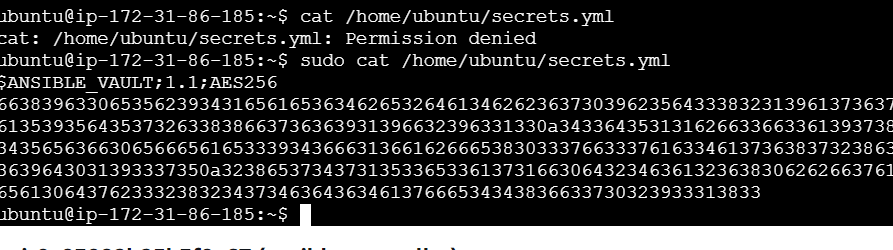
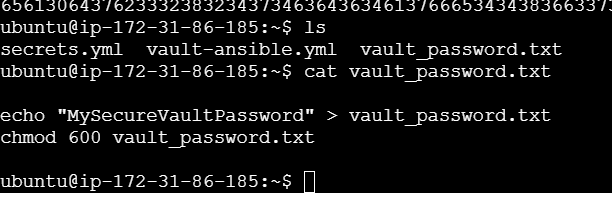
Ansible-04

1) Create an ansible playbook to create an ansible vault.







1 cd

2 ssh-keygen

3 cat /home/ubuntu/.ssh/id\_ed25519.pub

4 apt update -y

5 sudo apt update

6 apt install ansible -y

7 sudo apt install ansible -y

8 ansible --version

9 sudo mkdir /etc/ansible

10 sudo nano /etc/ansible/ansible.cfg

11 sudo nano /etc/ansible/hosts

12 ansible all -m ping

13 sudo vi vault-ansible.yml

14 cd tmp

15 sudo cd tmp

16 cd /tmp

17 ls

18 vi vault\_password.txt

19 cd

20 ansible-playbook vault-ansible.yml --syntax-check

21 ansible-playbook vault-ansible.yml --check

22 sudo apt update

23 ansible-playbook vault-ansible.yml

24 ansible-playbook vault-ansible.yml --check

25 vi vault\_password.txt

26 sudo vi vault-ansible.yml

27 ansible-playbook vault-ansible.yml --check

28 sudo vi vault-ansible.yml

29 vi vault\_password.txt

30 sudo vi vault-ansible.yml

31 sudo vi vault-password.txt

32 ansible-playbook vault-ansible.yml --check

33 ansible-playbook vault-ansible.yml

34 ls -l /home/ubuntu/vault\_password.txt

35 ls

36 sudo vi vault\_password.txt

37 ls

38 mv vault-password.txt vault\_password.txt

39 ls

40 ls -l /home/ubuntu/vault\_password.txt

41 ansible-playbook vault-ansible.yml

42 sudo vi vault-ansible.yml

43 ansible-playbook vault-ansible.yml

44 cat /home/ubuntu/secrets.yml

45 sudo cat /home/ubuntu/secrets.yml

46 ls

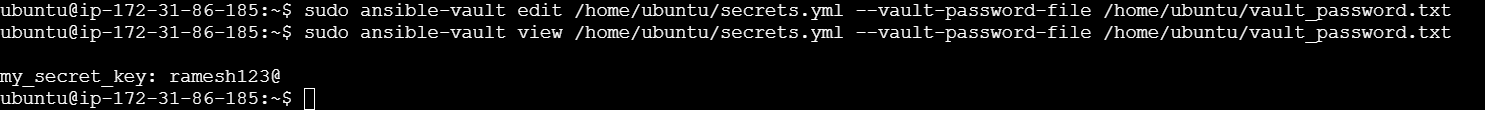
47 cat vault\_password.txt

48 ansible-vault view /home/ubuntu/secrets.yml --vault-password-file /home/ubuntu/vault\_password.txt

49 sudo ansible-vault view /home/ubuntu/secrets.yml --vault-password-file /home/ubuntu/vault\_password.txt

50 cat /home/ubuntu/secrets.yml

51 history



---

- name: Create and encrypt an Ansible Vault File

hosts: localhost

become: yes

tasks:

- name: Create a secrets file

copy:

dest: "/home/ubuntu/secrets.yml"

content: |

my\_secret\_key: my\_secret\_value

mode: '0600'

- name: Encrypt the secrets file

ansible.builtin.command:

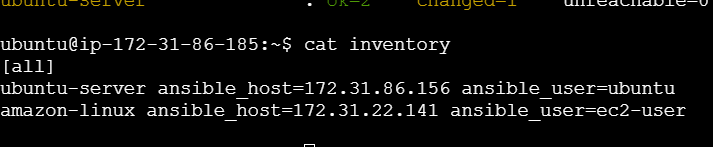
cmd: "ansible-vault encrypt /home/ubuntu/secrets.yml --vault-password-file /home/ubuntu/vault\_password.txt"

ubuntu@ip-172-31-86-185:~$ cat vault\_password.txt

echo "MySecureVaultPassword" > vault\_password.txt

chmod 600 vault\_password.txt

2) Write an ansible playbook to install apache in linux and ubuntu machine by using when condition.



---

- name: Install Apache on Linux and Ubuntu

hosts: all

become: yes

tasks:

- name: Install Apache on Ubuntu

apt:

name: apache2

state: present

when: ansible\_os\_family == "Debian"

- name: Install Apache on RedHat-based systems

package:

name: httpd

state: present

when: ansible\_os\_family == "RedHat" or ansible\_distribution == "Amazon"

- name: Start Apache on RedHat-based systems

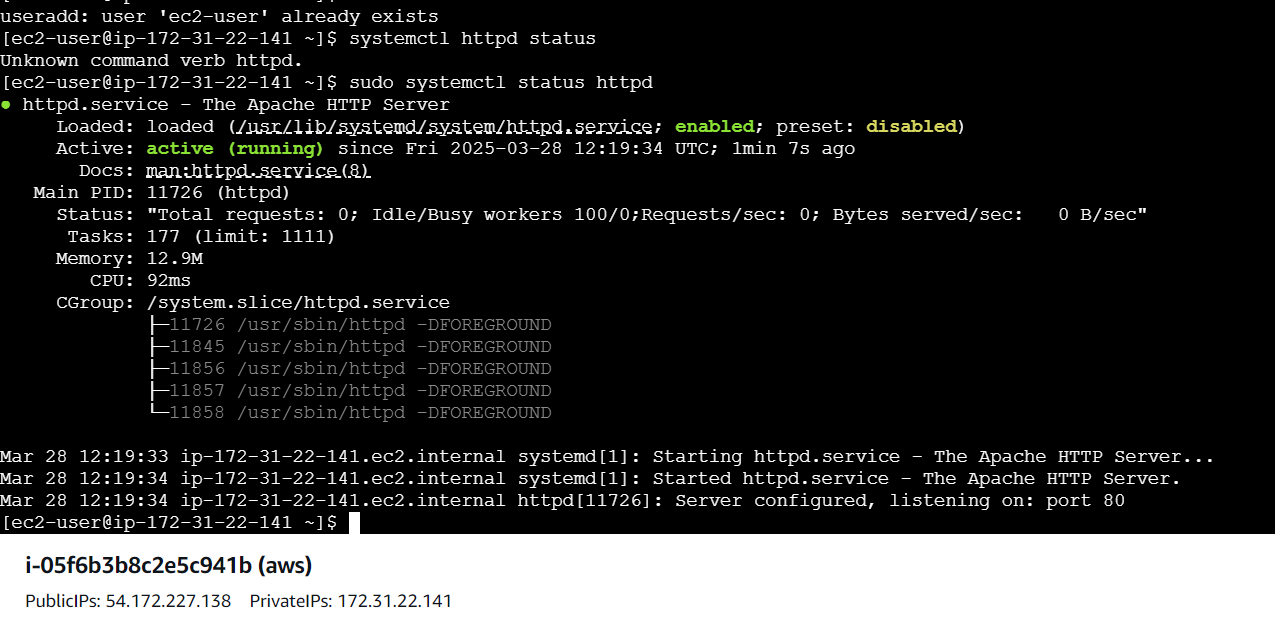
service:

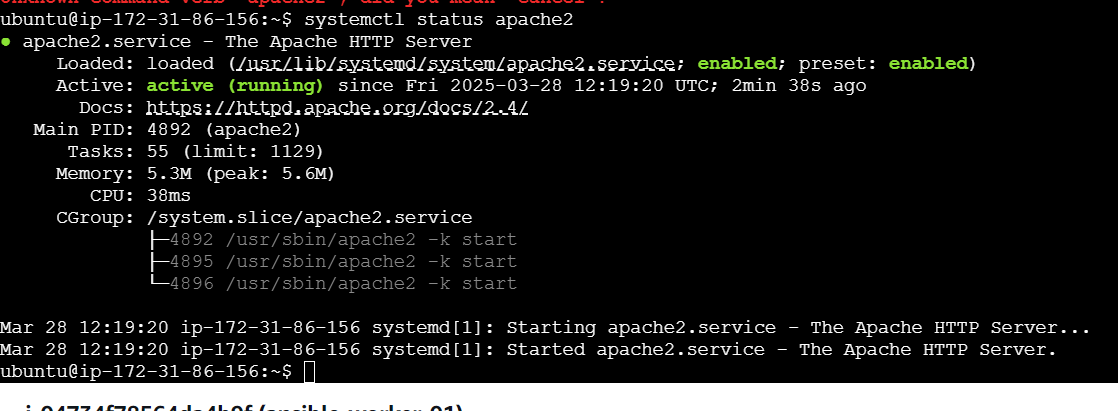
name: httpd

state: started

enabled: yes

when: ansible\_os\_family == "RedHat" or ansible\_distribution == "Amazon"





3) Create ansible playbook using roles to configure LAMP stack

mkdir -p lamp-stack/roles/{apache,mysql,php}/tasks

mkdir -p lamp-stack/inventory

lamp-stack/

├── inventory/ # Hosts inventory

│ ├── hosts.ini

├── roles/ # Roles for LAMP components

│ ├── apache/

│ │ ├── tasks/

│ │ │ ├── main.yml

│ ├── mysql/

│ │ ├── tasks/

│ │ │ ├── main.yml

│ ├── php/

│ │ ├── tasks/

│ │ │ ├── main.yml

├── site.yml # Main playbook

---

- name: Configure LAMP Stack

hosts: webservers

become: yes

roles:

- apache

- mysql

- php

roles/apache/tasks/main.yml

---

- name: Install Apache

ansible.builtin.package:

name: "{{ 'apache2' if ansible\_os\_family == 'Debian' else 'httpd' }}"

state: present

- name: Start Apache

ansible.builtin.service:

name: "{{ 'apache2' if ansible\_os\_family == 'Debian' else 'httpd' }}"

state: started

enabled: yes

roles/mysql/tasks/main.yml

---

- name: Install MySQL

ansible.builtin.package:

name: "{{ 'mysql-server' if ansible\_os\_family == 'Debian' else 'mariadb-server' }}"

state: present

- name: Start MySQL

ansible.builtin.service:

name: "{{ 'mysql' if ansible\_os\_family == 'Debian' else 'mariadb' }}"

state: started

enabled: yes

roles/php/tasks/main.yml

---

- name: Install PHP

ansible.builtin.package:

name: php

state: present

- name: Restart Apache

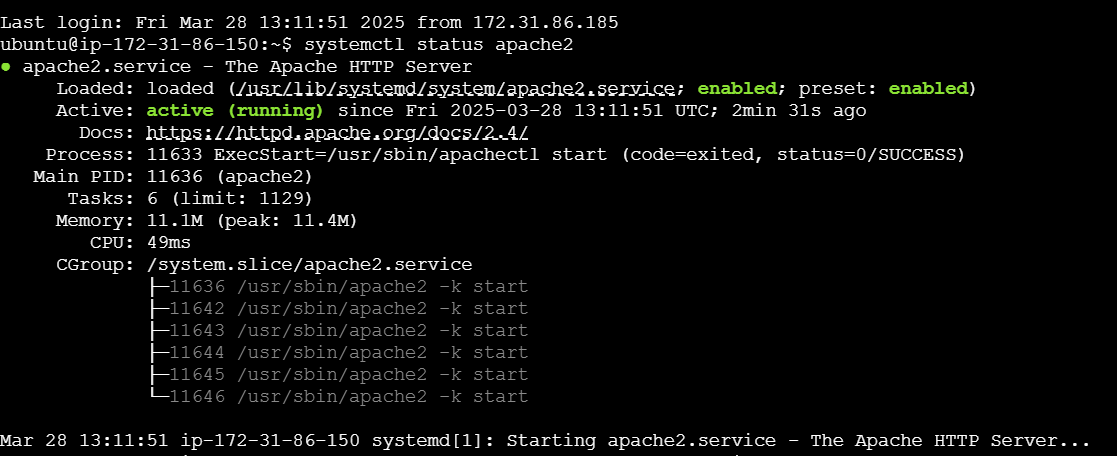
ansible.builtin.service:

name: "{{ 'apache2' if ansible\_os\_family == 'Debian' else 'httpd' }}"

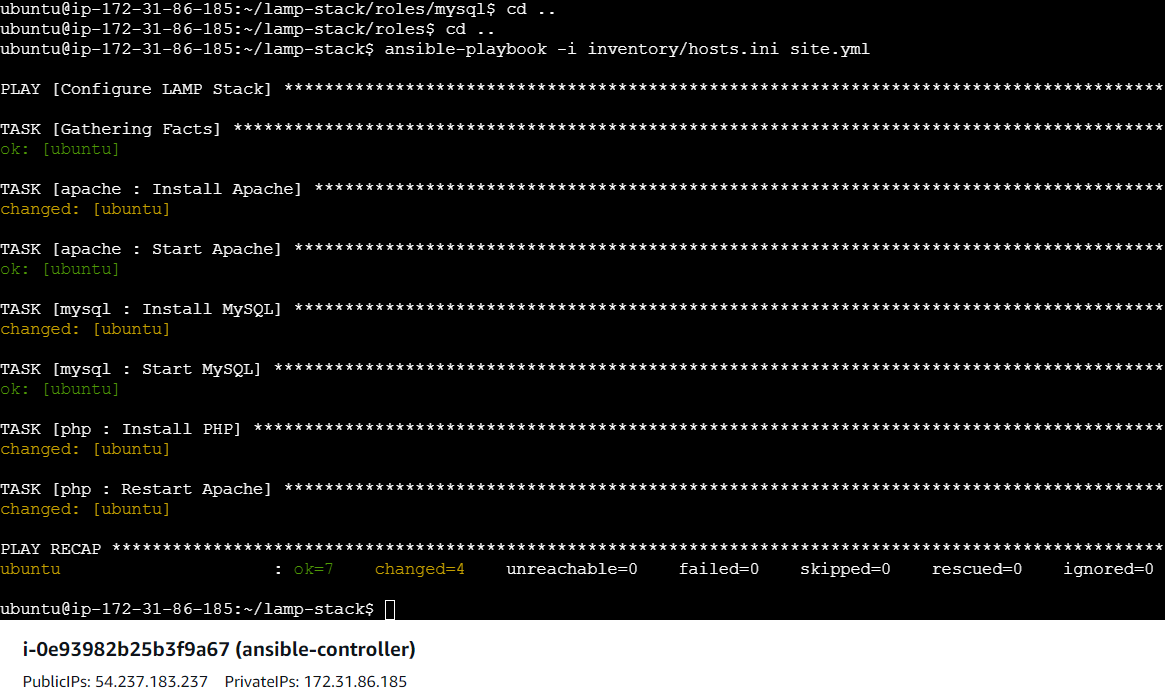
state: restarted

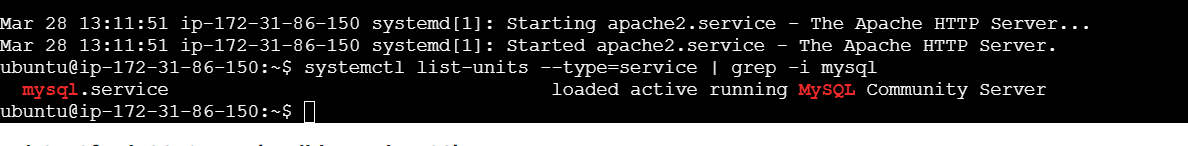
cd lamp-stack

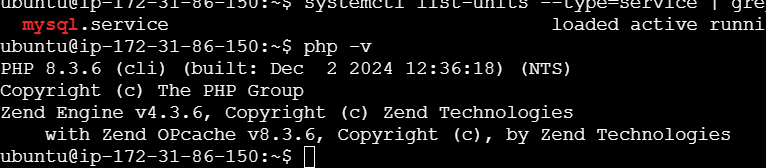
ansible-playbook -i inventory/hosts.ini site.yml









****

****

4) Setup ansible AWX and explore the options.

