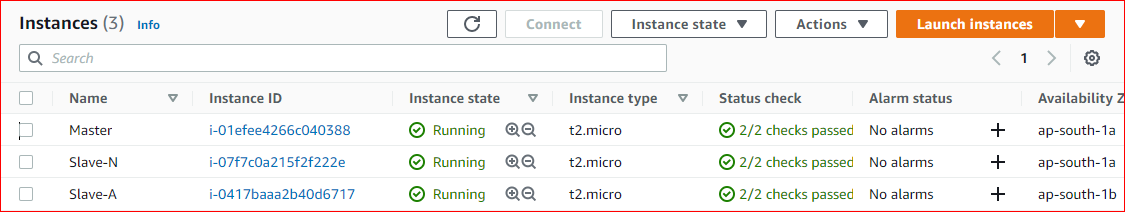
**Module-6: Ansible Assignment –1**

Setup Ansible cluster with 3 nodes

On slave 1 install java

On slave 2 install mysql-server

First I created 3 instance (1 master , slave 1, slave 2)





Vi mysql.yml

--- # comments

- hosts: apacheserv # slave2

become: yes

tasks:

- name: install mysql-server

apt: name=mysql-server state=present update\_cache=yes state=present

- name: install mysql-myclient

apt: name=mysql-client-core-8.0 update\_cache=yes state=present

- name: Start SQL Service

service: name=mysql state=started enabled=true

- hosts: nginxserv # slave 1

Become: yes

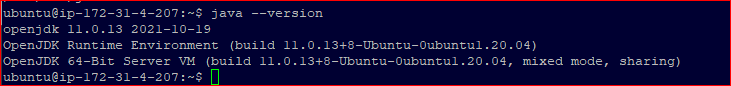
tasks:

- name: INSTALL java TASK

apt: name=openjdk-11-jdk update\_cache=yes state=latest

become: true

ansible-playbook mysql.yml





**Module- 6: Ansible Assignment – 2**

Create a script which can add text “ This text has been added by custom script” to /tmp.1.txt

Run this script using Ansible on all the hosts.

Vi script.sh

#! /bin/bash

Echo “ This text has been added by custom script” > tmp2.txt

Chmod +x script.sh

Vi runscript.yml

--- # comments

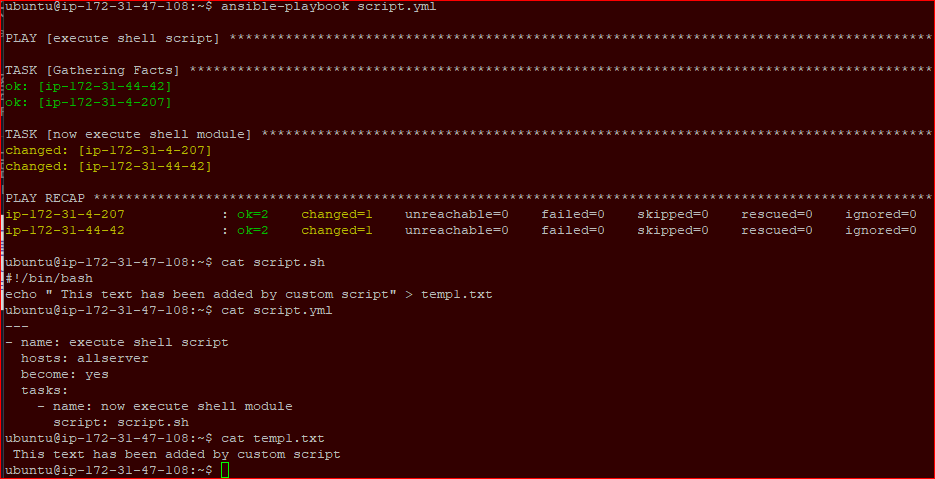
- hosts: allserver

become: yes

tasks:

- name: install mysql-server

Script: ./script.sh



**Module -6 : Ansible Assignment – 3**

Create Ansible Role called “ansible-new”

Install apach2 on slave1 and nginx on slave2

Mkdir roles; cd roles; ansible-glaxy init ansible-new

Cd ansible-new ;

mkdir -p host\_vars/ip-172-31-4-207 ; mkdir -p host\_vars/ip-172-31-44-42

Below is separate host variable defined.

Cd host\_vars/ip-172-31-4-207

Vi main.yml

Serv: apache2

Cd host\_vars/ ip-172-31-44-42

Vi main.yml

Serv: nginx

Vi ansible-new /tasks/install.yml

name: INSTALL APACHE/NGINX TASK

apt: name={{ serv }} update\_cache=yes state=latest

become: true

vi ansible-new /tasks/main.yml

----

- include: install.yml

Vi roles/playbook.yml

---

- hosts: allserver

roles:

- ansible-new

ubuntu@ip-172-31-47-108:~$ cat /etc/ansible/hosts

[apacheserv]

ip-172-31-44-42

[nginxserv]

ip-172-31-4-207

[allserver]

ip-172-31-4-207

ip-172-31-44-42

**Module-6: Ansible Assignment – 4**

Use the previous deployment of ansible cluster

Configure the files folder in “ansible-new” role with index.html which should be replaced with the original index.html

Configure the handlers folder in “ansible-new” role for restarting the nginx service.

vi ansible-new /files/index.html # edited file

vi ansible-new /tasks/install.yml

name: copy file from source to destination

copy: src=index.html dest=/var/ww/html/

notify: start server

vi ansible-new /handlers/main.yml

-name: start server

Action: service name= {{ serv }} state= restarted

**Module-6: Ansible Assignment – 5**

Create a new deployment of ansible cluster of 5 nodes

Label 2 nodes as test and other 2 as prod

Install java on test nodes

Install mysql-server on prod nodes.

Ans:

First will create 5 ec2 instances where 1 for master, 2 for test and 2 for prod. Configured ansible on respectively all servers.

Now connect from master server. mkdir roles ; cd roles. Ansible-galaxy init role1 and role2

Vi /etc/ansible/hosts

[prod]

Server1-ip

Server2-ip

[test]

Server3-ip

Server4-ip

Cd role1 ; cd task ; vi main.yml

---

- name: INSTALL java TASK

apt: name={{java-jdk}} update\_cache=yes state=latest

become: true

cd ../vars ; vi main.yml # define variable here

java-jdk: openjdk-11-jdk

cd ../../role2/tasks ; vi main.yml

- name: install database

apt: name={{item}} state=installed

with\_items:

- mysql-server

- mysql-client

- name: start my sql server

service: name={{ serv\_name}} state=started enabled=yes

cd ../vars ; vi main.yml # define variable here

serv\_name: mysql

cd ../../../

vi playbook.yml

-name: executing java

Hosts: test

Become: yes

Roles:

-role1

Hosts: prod

Become: yes

Roles:

-role2

**ANSIBLE CASE STUDY**

You are a Devops Engineer and the organization you are working on needs to set up two configuration management server groups. One for Apache another for Nginx. Being a Devops Engineer it is your task to deal with this configuration management issues.

Let us see the tasks that you need to perform using Ansible.

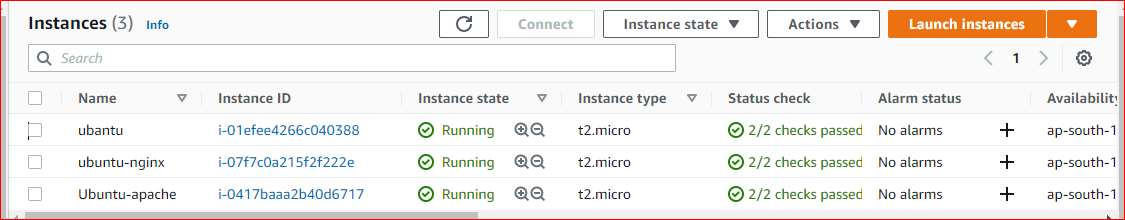
1. Create two Server Groups. One for Apache and another for Nginx.
2. Push two html files with their server information.

Make sure that you don’t forget to start the services once the installation is done. Also send post installation messages for both the server groups.

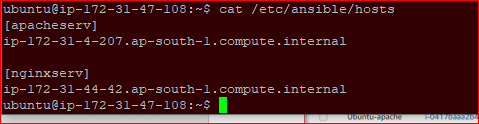
Using Ansible Roles accomplish the above the tasks.

Also, once the Apache server configuration is done you need to install java on that server group using ansible role in a playbook.

Below is ansible master and target servers.



And this is server group details.



This is installation link:

install command ansible:

https://docs.ansible.com/ansible/latest/installation\_guide/intro\_installation.html#installing-ansible-on-ubuntu

On master:

sudo apt update &&

sudo apt install software-properties-common -y &&

sudo add-apt-repository --yes --update ppa:ansible/ansible &&

sudo apt install ansible –y

SSH Connection details:

on master:

as a normal user:

cd .ssh

ls

authorized\_keys

ssh-keygen

//enter

ls

authorized\_keys id\_rsa id\_rsa.pub

cat id\_rsa.pub

//copy the key

On worker node:(both node)

cd .ssh

sudo nano authorised\_keys

Note: do not modify the existing line

on a new line paste.

to verify the connection: ssh ubuntu@<public ip of the worker>

Server group details

cd /etc/ansible

vi hosts

[apacheserver]

ip-172-31-40-134.us-east-2.compute.internal

[nginxserver]

ip-172-31-44-53.us-east-2.compute.internal

To check server connection:

ansible -m ping apacheserv

ansible -m ping nginxserv

On Master:

mkdir roles; cd roles

cd roles

sudo ansible-galaxy init role1

sudo ansible-galaxy init role2

---------------------------------------------

------------------------yaml files --------------------

cd role1

main.yml

---

# tasks file for role1

- include: install.yml

- include: config.yml

- include: service.yml

- include: java.yml

install.yml

---

- name: INSTALL APACHE TASK

apt: name=apache2 update\_cache=yes state=latest

become: true

java.yml

---

- name: INSTALL java TASK

apt: name=openjdk-11-jdk update\_cache=yes state=latest

become: true

config.yml

---

- name: copying files from the source to the target machine w1

copy: src=site dest=/var/www/html

become: true

- name: message task

copy: src=message.txt dest=/tmp

become: true

service.yml

---

- name: service task:>startin\_the service

service: name=apache2 state=started

become: true

=====================================

cd role2

main.yml

---

# tasks file for role2

- include: install.yml

- include: config.yml

- include: service.yml

install.yml

---

- name: INSTALL NGINX TASK

apt: name=nginx update\_cache=yes state=latest

become: true

config.yml

---

- name: copying files from the source to the target machine

copy: src=site dest=/var/www/html

become: true

- name: message task

copy: src=message.txt dest=/tmp

become: true

service.yml

---

- name: service task:>startin\_the service

service: name=nginx state=started

become: true

-------------------------------------------------------------------------

cd roles

vi message.txt # write any message

playbook.yml

---

- hosts: apacheserver

roles:

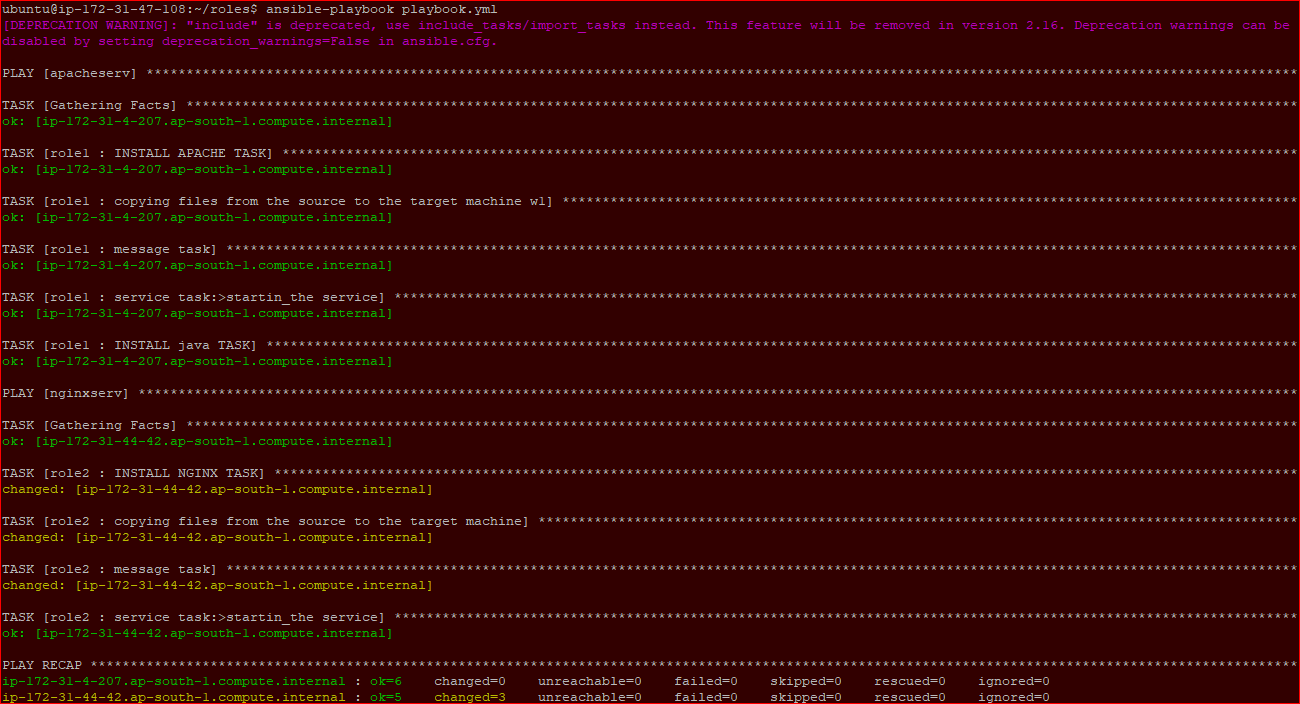
- role1

- hosts: nginxserver

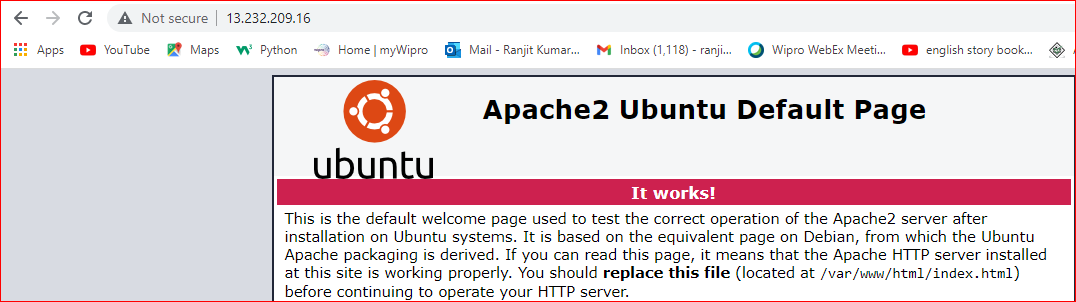
roles:

- role2

Ansible-playbook playbook.yml

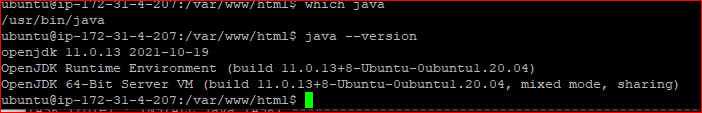


Now default Apache page is working fine as below screen

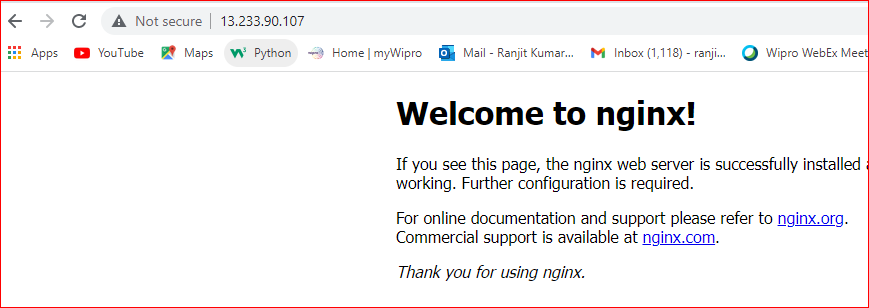


Below is customize apache index.html





Below is Nginx installed on server details:



And now nginx page customized

