ANISIBLE

We’ll be creating one Anisible Server and 2 host. By creating instance.

Step 1: To install anisible on your virtual machine :

Switch to root user

sudo apt update

sudo apt upgrade

sudo apt install software-properties-common

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

sudo apt install ansible

Step 2: To check if its install :

cd /etc/ansible

ansible --version

ls -la

Step 3: To add host IP in Anisible Master: (Host Updation )

nano /etc/ansible/hosts

Add IP address by grouping , if u have more than 1 host/instance

Ex: [group name]

10.0.2.3

10.0.200.36

[frontend]

frontend\_private\_ip

[backend]

Backend\_private\_ip

Step 4: Configuration on anisible

nano /etc/ansible/ansible.cfg

Allow : inventory = /etc/ansible/hosts

sudo\_user = root

by removing # infront.

Step 5: Create user with same name as ansible server in frontend and backend and set password for that user and set password for ansible server as well ( You can give same password that will be easy to remember)

sudo passwd yourusername(here is your ansibleServer)

Also give permission to user (You have to do this in ansibleServer, frontendServer as well in backedServer also)

sudo visudo

User privilege specification

root ALL=(ALL:ALL) ALL

ansibleServer ALL=(ALL:ALL) NOPASSWD: ALL (this need to be updated with the username which you created for

3 servers)

Step 6: To give permission to SSH ( this needs to be done on ansible server, backend server and frontend server)

nano /etc/ssh/sshd\_config

# Authentication:

#LoginGraceTime 2m

#PermitRootLogin prohibit-password --- > (Remove #)

#StrictModes yes

#MaxAuthTries 6

#MaxSessions 10

# To disable tunneled clear text passwords, change to no here!

PasswordAuthentication no ---( make PasswordAuthentication as yes instead of no)

#PermitEmptyPasswords no

Step 7: Restart SSH ( this needs to be done on ansible server, backend server and frontend server)

service sshd restart

service sshd status

Step 8: For passwordless login(switch to your user if ansibleServer)

ssh-keygen

ls -a

cd .ssh/

ls

ssh-copy-id yourusername@ privateIP (you have to run this command twice first replacing userName and ip of frontend and then replacing with backend)

ssh-copy-id frontendServer@ frontendPrivateIP

ssh-copy-id backendServer@ backendPrivateIP

Step 9: switch to /home/ansibleServer ie your ansible user

create folder config and create gunicorn.service file for backend

\*\*\*\*\*\*\* Gunicorn.service \*\*\*\*\*\*\*\*\*\*

[Unit]

Description=gunicorn daemon

After=network.target

[Service]

User=ansibleSever

Group=www-data

WorkingDirectory=/home/ansibleServer/new\_chatapp/fundoo

ExecStart=/bin/bash -c 'source /home/ansibleServer/new\_chatapp/venv/bin/activate && gunicorn --bind 0.0.0.0:8000 fundoo.wsgi:application'

[Install]

WantedBy=multi-user.target

NOTE :- Here ansibleServer means the user created in backendServer

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

in config folder create chatngx.conf for frontend

\*\*\*\*\*\*\* chatngx.conf \*\*\*\*\*\*\*\*\*\*

server {

listen 80;

server\_name \_default;

#location = /favicon.ico { access\_log off; log\_not\_found off; }

location /static/ {

root /home/ansibleServer/new\_chatapp/fundoo;

}

location / {

include proxy\_params;

proxy\_pass http://10.1.1.4:8000;

}

}

NOTE :- Here ansibleServer means the user created in frontendServer

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Step 9: Create playbook

sudo vi backend.yml

sudo vi frontend.yml

ansible-playbook backend.yml

ansible-playbook frontend.yml