1-3 empty VMs

3VMs networking is bridged

1-1xansible controller

Playbook to configure the other 2 VMS,each one

1-Rename each server

2-apache2

3-configure 2 virtual hosts on each web server

wp1.ahmed.com

wp2.ahmed.com

4-Upload wordpress to each server

2-2xEmpty VMS

**VM1**

Install network manager:

$ sudo apt install network-manager

$ sudo nano /etc/default/grub

GRUB\_CMDLINE\_LINUX="", net.ifnames=0 biosdevname=0

$ sudo grub-mkconfig -o /boot/grub/grub.cfg

$ sudo touch /etc/NetworkManager/conf.d/10-globally-managed-d.conf

$ sudo reboot

Firewall:

$ sudo apt install firewalld

$ sudo systemctl enable --now firewalld

$ sudo firewall-cmd --add-service=http

$ sudo firewall-cmd --permanent --add-service=http

$ sudo firewall-cmd --reload

$ ip a

clone **VM2** & **VM3**

**VM2**

$ ip a

$ sudo nmcli con add con-name eth1-new type ethernet ifname eth1 ip4 192.168.10.3/24 gw4 192.168.10.1

$ sudo nmcli con mod eth1-new ipv4.dns 4.2.2.2

$ sudo nmcli con up eth1-new



**VM3**

$ ip a

$ sudo nmcli con add con-name eth0-new type ethernet ifname eth0 ip4 192.168.10.2/24 gw4 192.168.10.1

$ sudo nmcli con mod eth0-new ipv4.dns 4.2.2.2

$ sudo nmcli con up eth0-new

**VM1**

Install pip

$ sudo apt install curl

$ curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py

$ python get-pip.py --user

Install ansible

$ sudo apt install software-properties-common

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

$ sudo apt install ansible

$ nano hosts

client1 ansible\_host=192.168.1.11 ansible\_user=controller ansible\_password=1234 ansible\_become\_password=1234 hostname=guest1 domain=wp1.ahmed.com

client2 ansible\_host=192.168.1.12 ansible\_user=controller ansible\_password=1234 ansible\_become\_password=1234 hostname=guest2 domain=wp2.ahmed.com

$ ansible -i hosts client1 -m ping //to test

$ ansible -i hosts all -m ping

$ nano rename.yml

---

- name: Rename hosts

become: yes

hosts: all

gather\_facts: no

tasks:

- name: Rename a host

ansible.builtin.hostname:

name: "{{ hostname }}"

- name: Install apache2

ansible.builtin.package:

name: apache2

state: present

$ ansible-playbook -i hosts rename.yml

**VM2 & VM3**

$ sudo systemctl status apache2

**VM1**

$ nano config.yml

---

- name: Rename hosts

become: yes

hosts: all

gather\_facts: no

tasks:

- name: Rename a host

ansible.builtin.hostname:

name: "{{ hostname }}"

- name: installing apache

ansible.builtin.package:

name: apache2

state: present

- name: enabled mod\_rewrite

apache2\_module:

name=rewrite

state=present

notify:

- restart apache2

- name: create virtual host file

ansible.builtin.template:

src: virtualhost.conf

dest: /etc/apache2/sites-available/{{ domain }}.conf

notify:

- restart apache2

- name: a2ensite "{{ domain }}"

command: a2ensite "{{ domain }}"

notify:

- restart apache2

- name: Install dependencies for wordpress

apt:

name={{ item }}

state=present

with\_items:

- python3-pip

- mariadb-server

- php

- php-mysql

- name: Install Required pip modules

pip:

name:

- PyMySQL

state: present

executable: pip3

- name: Ensure MariaDB is running (and enable it at boot)

service: name=mariadb state=started enabled=yes

- name: Copy ~/.my.cnf to nodes

copy: src=~/.my.cnf dest=/root/.my.cnf

- name: Create MariaDB user

community.mysql.mysql\_user: name=wpuser password=1234 state=present login\_user=root login\_password=1234 priv=\*.\*:ALL

- name: Create MariaDB database

mysql\_db: name=wpdatabase state=present

- name: Download WordPress

get\_url: url=https://www.wordpress.org/latest.tar.gz dest=/var/www/html/wordpress.tar.gz validate\_certs=no

- name: Extract WordPress

command: "tar xzf /var/www/html/wordpress.tar.gz -C /var/www/html"

- name: copy configuration file of wordpress

copy: src=~/wp-config.php dest=/var/www/html/wordpress/wp-config.php mode=0644

handlers:

- name: restart apache2

service: name=apache2 state=restarted

$ ansible-playbook -i hosts config.yml