

Assignment - Ansible

Github : https://github.com/Priygupt13/CMPE272-Assignment_Ansible

Create three Instances(taking ubuntu as O.S) on EC2, in which one VM will act as primary or control machine on which ansible is installed and other VMs will acts as secondary machines on which through ansible web servers will be deployed.

Step1.

Use Elastic IP addresses to make the static IPs for all three VMs. In this these static IPs, will not changed after restarting the machines.

A screenshot of the AWS Elastic IP Addresses page. The top navigation bar shows 'Search for services, features, blogs, docs, and more' and '[Option+S]'. On the right, it shows 'Actions' with a dropdown, 'N. California', and a user 'priyagupta'. Below the header, there's a search bar with 'Filter Elastic IP addresses' and a table with three rows. The table columns are: Name, Allocated IPv4 add..., Type, Allocation ID, and Reverse DNS record. The rows are: VM1 (52.8.28.9, Public IP, eipalloc-02c5db71dc5e09689, -), VM2 (52.8.65.36, Public IP, eipalloc-03e5dc72e6b049d21, -), and AnsibleVM (54.151.40.77, Public IP, eipalloc-0e3d4a80fdf199450, -).

<input type="checkbox"/>	Name	Allocated IPv4 add...	Type	Allocation ID	Reverse DNS record
<input type="checkbox"/>	VM1	52.8.28.9	Public IP	eipalloc-02c5db71dc5e09689	-
<input type="checkbox"/>	VM2	52.8.65.36	Public IP	eipalloc-03e5dc72e6b049d21	-
<input type="checkbox"/>	AnsibleVM	54.151.40.77	Public IP	eipalloc-0e3d4a80fdf199450	-

Step 2: Need to edit inbounds role for open TCP port 8080

Security details

IAM Role -	Owner ID 230802745283	Launch time Tue Sep 06 2022 14:54:48 GMT-0700 (Pacific Daylight Time)
Security groups sg-0a54f7382f8f3f51a (launch-wizard-4)		

Inbound rules

Security group rule ID	Port range	Protocol	Source	Security groups
sgr-092a3e5739ff5fe97	8080	TCP	0.0.0.0/0	launch-wizard-4
sgr-065464829f2ce7b69	443	TCP	0.0.0.0/0	launch-wizard-4
sgr-0130dd5da202246c8	22	TCP	0.0.0.0/0	launch-wizard-4
sgr-06b9b6bc3739ce00a	All	All	::/0	launch-wizard-4
sgr-0c87e3811d1487459	80	TCP	0.0.0.0/0	launch-wizard-4

Outbound rules

Filter rules

Step 3: After launching VMs, connect one VM(Primary VM) and install ansible using command

Sudo apt-get install ansible

```

Welcome to Ubuntu 22.04 LTS (GNU/Linux 5.15.0-1011-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

 System information as of Thu Sep  8 21:29:22 UTC 2022

 System load:  0.13671875      Processes:          102
 Usage of /:   35.9% of 7.58GB   Users logged in:    0
 Memory usage: 34%              IPv4 address for eth0: 172.31.27.45
 Swap usage:   0%

 * Ubuntu Pro delivers the most comprehensive open source security and
 compliance features.

 https://ubuntu.com/aws/pro

37 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

*** System restart required ***
Last login: Thu Sep  8 21:29:23 2022 from 13.52.6.115
ubuntu@ip-172-31-27-45:~$ sudo apt-get install ansible
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ansible is already the newest version (2.10.7+merged+base+2.10.8+dfsg-1).
0 upgraded, 0 newly installed, 0 to remove and 37 not upgraded.
ubuntu@ip-172-31-27-45:~$ 
```

Step 4: Open the SSH connection between machines by generation random public key by using command ssh-keygen on VM1(ansible machine) and copy this public key in authorized file(location of this file is under cd .ssh) of other VMs.

Do this cmd in ansible machine :

a. Cd .ssh

```
ubuntu@ip-172-31-27-45:~$ cd .ssh  
ubuntu@ip-172-31-27-45:~/ssh$  
ubuntu@ip-172-31-27-45:~/ssh$  
ubuntu@ip-172-31-27-45:~/ssh$
```

b. II

```
ubuntu@ip-172-31-27-45:~/.ssh$ ll
total 28
drwx--- 2 ubuntu ubuntu 4096 Sep 7 22:28 .
drwxr-x--- 6 ubuntu ubuntu 4096 Sep 8 23:02 ..
-rw---- 1 ubuntu ubuntu 394 Sep 4 00:47 authorized_keys
-rw---- 1 ubuntu ubuntu 2610 Sep 6 23:08 id_rsa
-rw-r--r-- 1 ubuntu ubuntu 576 Sep 6 23:08 id_rsa.pub
-rw---- 1 ubuntu ubuntu 2240 Sep 7 22:28 known_hosts
-rw---- 1 ubuntu ubuntu 1404 Sep 6 23:18 known_hosts.old
ubuntu@ip-172-31-27-45:~/.ssh$
```

c. Vim id rsa.pub

```
total 28
drwx----- 2 ubuntu ubuntu 4096 Sep  7 22:28 .
drwxr-x--- 6 ubuntu ubuntu 4096 Sep  8 23:02 ..
-rw----- 1 ubuntu ubuntu  394 Sep  4 00:47 authorized_keys
-rw----- 1 ubuntu ubuntu 2610 Sep  6 23:08 id_rsa
-rw-r--r-- 1 ubuntu ubuntu  576 Sep  6 23:08 id_rsa.pub
-rw----- 1 ubuntu ubuntu 2240 Sep  7 22:28 known_hosts
-rw----- 1 ubuntu ubuntu 1404 Sep  6 23:18 hosts.old
ubuntu@ip-172-31-27-45:~$ ssh vim id_rsa.pub
ubuntu@ip-172-31-27-45:~$ ssh
```

d. Copy the public key of VM1(Ansible machine) and paste it inside authorized file in hosts machines

```
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQgOCvUJ8nw7f5SuqSArVfpF1C8skDtJSUG3WV5ReTeAYikpNRSLefdzC1kEDeuXQh7voPgmVzL3pRBByYY+g6w/G1Q0RMQe2GbMhXyrHMWeo17d0+GmKWB  
4DFVFC1LEYo/Z9zFEP1sU5Ox5xFMrnsrdyxxyBgt0SkndzWfoZakkh0W8v19QxqV4z78oFyN19aBtfFTDN4eMywdsGT13xG0m1XuUBDaa8jFRs9oRBETdj7u94XECq+EcGEcjjHaeypjZXELsxZdtACUHLSG04r  
B5pWMxFMxIwmTTrv4WzvKkThzUofEsrhjFS/40QVweGas8L4zDB9/BaSpfaa2SRv03IxvM6_NtT6DjBHvnYjwyP5soZMMBlToivns6tvu48u+Gla8BT9c0Fisx1FJZfaOzRbhXr4QfyHlfB0ffeyjsMvb1Ksb  
HkOnAYPszIrl0oflc5Agixlqo6gtstzByjNDKCNcjjJ9R/H4BH3H7Wml6DHd6yjAWDntDqd0VGys0= ubuntu@ip-172-31-27-45
```

e. Go to authorized file inside **hosts machine**

```
curl: (7) Failed to connect to ec2-52-8-28-9.us-west-1.compute.amazonaws.com port 80 after 5 ms: Connection refused
ubuntu@ip-172-31-22-82:~$ curl http://ec2-52-8-28-9.us-west-1.compute.amazonaws.com
curl: (6) Could not resolve host: http
ubuntu@ip-172-31-22-82:~$ cd .ssh
ubuntu@ip-172-31-22-82:~/ssh$ ll
total 12
drwx----- 2 ubuntu ubuntu 4096 Sep  6 23:09 .
drwxr-x--- 6 ubuntu ubuntu 4096 Sep  7 20:38 ../
-rw----- 1 ubuntu ubuntu  964 Sep  6 23:09 authorized_keys
ubuntu@ip-172-31-22-82:~/ssh$
```

i-05d2b16e01147e56b (VM1)

f . vim authorized_keys

And copy the above public key from ansible machine which is primary machine into other two VMs.

```
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQgQCvUJ8nw7f5SuqSArvFVfP1C8skDtJSUG3WVw5ReTeAYikpNRSLefdzC1kEDeuXQhhv7oPGmVzL3pRByYYx+g6w/GiQ0RMQe2GbMhXyRHMWeoi7d0+GmKwGb4DFYClEYo/Z9zFEIFsU5Oo5XFMrnsrdyxxgBgU08skndZwfcaZakkH0W8g19QxgV4zT8oFyN19aBtFTDn4eMywdsGI13xGom1XuUBDaa8jFRs9oRBEtdj7u94KECq+EoGGEcjjhAeypjZXElSxEzdtaCUHLGSG04rB5pwFMxiwmTlrV4W2vkkKThzUofEsrhjFS/4Q0VweGas8I4zbB9/BaSPaa25Rv03lxvM6/Nt6DJBHVNjwyP5soZ2MMB1Toinv6tvu48u+GlaBT9cOfisx1FJZfa0zrBhXr4QfyHlffB0offeyjSmvb1KsbHkOnAYPszZrl1Of1c5Agixlqo6gtstzByjNDKKn0jjj9R/4B4BH3H7Wml16HD6yjAWDntDQd0VGys0= ubuntu@ip-172-31-27-45
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQaC1fIEHXDGhuQd+BP8XTrCOKElns9atynhgsp8+3hq485U15dpIGNp2g7sT2XcgRNX1SQXmt6X8v3pNxWFAYFJKXKk+7r1MMCUTBiYpKRUFOWnE/7Ss/3VnabV4yTzPNWK+DhfifhrIGjA/ZS0b767z9WC2TlQNkypqtrdS6QnyJSD8F92Mk/wjpR3wQemehL0YZta0JiJdrDxJeW42M4VI1i091jcAKk9LHQ549qyew2WnRyBZuZjG1JLTGvUmBquCgLHaJ6s6dUHHxnYM9XhjZdJycV6FGp1L64hHseMjjHHRBkERd8Vme2Sy2OE0uDTHQj/NFrFVZguY60yzM/37 VM1Key
```

Step 4: Update the **host file** inside ansible directory and mention client VM2 and VM3 ip address and along with dynamic message value which here it is set under `machine_index` for 1 and 2 for VM2 and VM3 respectively.

AWS Services Search for services, features, blogs, docs, and more [Option+S] ⓘ ⓘ N. California ▾

```
[web_servers]
VM1 ansible_host=52.8.28.9 machine_index=1
VM2 ansible_host=52.8.65.36 machine_index=2
```

-- INSERT -- 2, 42

```
i-0e5ebbd8d995eb736 (UbuntuEC2)
```

Step 5: Verify that ansible inventory is updated to reflect newly added server groups and make sure that all the hosts in [web_servers] group are pingable.

```
ubuntu@ip-172-31-27-45:~/priya/ansible$ ansible-inventory --list -y -i inventory
all:
  children:
    ungrouped: {}
    web_servers:
      hosts:
        VM1:
          ansible_host: 52.8.28.9
          machine_index: 1
        VM2:
          ansible_host: 52.8.65.36
          machine_index: 2
ubuntu@ip-172-31-27-45:~/priya/ansible$
```

Step 6: Deploy the apache web servers on both host VMs by executing the playbook (file name used here is apache.yml).

<https://github.com/Priyugupt13/CMPE272/blob/main/apache.yml>

```

- hosts: web_servers
  tasks:
    - name: install apache2
      apt: name=apache2 update_cache=yes state=latest
      become: 'yes'
    - name: enabled mod_rewrite
      apache2_module: name=rewrite state=present
      become: 'yes'
      notify:
        - restart apache2
    - name: copy index page
      ansible.builtin.copy:
        src: index.html
        dest: /var/www/html/index.html
        owner: ubuntu
        group: ubuntu
        mode: '0644'
      become: yes
    - name: update machine index in index.html
      lineinfile: >-
        dest=/var/www/html/index.html
        regexp=" _machine_index_ "
        line="{{ machine_index }}"
        state=present
      become: yes
    - name: configure apache2 to listen on port 8080
      lineinfile: >-
        dest=/etc/apache2/ports.conf
        regexp="Listen 80"
        line="Listen 8080"

```

1.7 Top

Step 7 : execute apache.yml file by using command

Command is : **ansible-playbook apache.yml -i inventory**

```

PLAY [web_servers] ****
TASK [Gathering Facts] ****
ok: [VM1]
ok: [VM2]

TASK [install apache2] ****
ok: [VM1]
ok: [VM2]

TASK [enabled mod_rewrite] ****
ok: [VM2]
ok: [VM1]

TASK [copy index page] ****
changed: [VM1]
changed: [VM2]

TASK [update machine index in index.html] ****
[WARNING]: The value "1" (type int) was converted to "'1'" (type string). If this does not look like what you expect, quote the entire value to ensure it
does not change.
changed: [VM1]
[WARNING]: The value "2" (type int) was converted to "'2'" (type string). If this does not look like what you expect, quote the entire value to ensure it
does not change.
changed: [VM2]

TASK [configure apache2 to listen on port 8080] ****
ok: [VM1]
ok: [VM2]

```

```

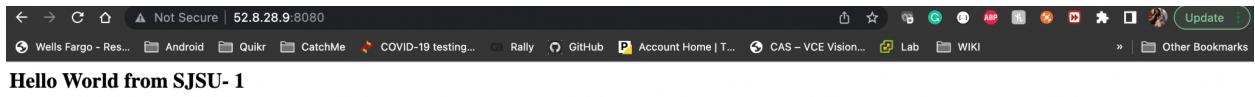
TASK [configure apache2 default config to listen on port 8080] ****
ok: [VM1]
ok: [VM2]

PLAY RECAP ****
VM1 : ok=7    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
VM2 : ok=7    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

```

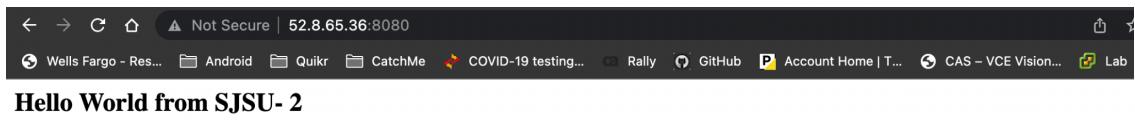
ubuntu@ip-172-31-27-45:~/oriva/ansible\$

Step 8 : Verify deployed web servers up and running on VM2



Hello World from SJSU- 1

Step 9 : Verify deployed web servers up and running on VM3



Hello World from SJSU- 2

Step 10 : Undeploy the apache web servers by executing the another undeploy_apache.yml script which will undeploy the web servers from both hosts mentioned in yml file.

Command : **ansible-playbook undeploy_apache.yml -i inventory**

```
ubuntu@ip-172-31-27-45:/priya/ansible$ ansible-playbook undeploy_apache.yml -i inventory
PLAY [web_servers] ****
TASK [Gathering Facts] ****
ok: [VM1]
ok: [VM2]

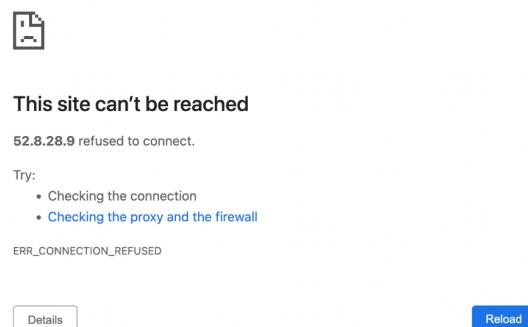
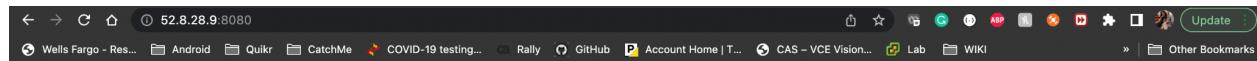
TASK [Uninstall Apache server] ****
changed: [VM1]
changed: [VM2]

TASK [Remove leftover Apache2 packages] ****
changed: [VM1]
changed: [VM2]

PLAY RECAP ****
VM1              : ok=3    changed=2      unreachable=0      failed=0      skipped=0      rescued=0      ignored=0
VM2              : ok=3    changed=2      unreachable=0      failed=0      skipped=0      rescued=0      ignored=0

ubuntu@ip-172-31-27-45:/priya/ansible$ [REDACTED]
i-0e5ebbd8d995eb736 (UbuntuEC2)
PublicIPs: 54.151.40.77 PrivateIPs: 172.31.27.45
```

Step :11 Verify web servers undeployed and not running on VM3



Step : 12 Verify web servers undeployed and not running on VM3

