

Opentext Challenge- By Shivam Namdeo

I. Ansible Playbooks

- 1) Create an Ubuntu EC2 instance (16.04 version)
- 2) Allow inbound traffic by allowing SSH rules
- 3) Install Ansible in our host Ubuntu VM

```
sudo apt-add-repository ppa:ansible/ansible
sudo apt-get update
sudo apt-get install ansible
```

- 4) Install docker using ansible playbook-

➤ Vim hosts

```
[group A]
all
```

```
[all:vars]
ansible_connection=local
```

➤ Vim docker.yml

```
---
- hosts: group A
  tasks:
    - name: Install prerequisites
      apt: name={{item}} update_cache=yes
      with_items:
        - apt-transport-https
        - ca-certificates
        - curl
        - software-properties-common
    - name: Add Docker GPG key
      apt_key: url=https://download.docker.com/linux/ubuntu/gpg
    - name: Add Docker APT repository
      apt_repository:
        repo: deb [arch=amd64] https://download.docker.com/linux/ubuntu
        {{ansible_distribution_release}} stable
    - name: Install Docker
      apt: name=docker-ce
```

5) Check if docker is installed

```
sudo docker run hello-world
```

6) Create 3 docker containers

```
sudo docker run -itd --name ansible_master ubuntu /bin/bash
```

```
sudo docker run -itd --name target2 ubuntu /bin/bash
```

```
sudo docker run -itd --name target3 ubuntu /bin/bash
```

7) `sudo docker ps` //it will show all the docker containers

8) `sudo docker exec -it ansible_master bash` //it will start bash process in ansible_master

9) `apt update` //update ubuntu

10) `apt install python ansible vim iputils-ping openssh-client -y`

11) `ansible -version` //check ansible version

12) `exit` //exit ansible_master

13) `sudo docker exec -it target1 bash` //open target1 docker container

14) `apt update` //update it

15) `apt install openssh-server` /install open ssh server in this container

16) `apt install python ansible vim iputils-ping openssh-client -y`

17) `cd /etc/ssh` //go to ssh

18) `vim sshd_config` //open sshd_config and update below value

```
PermitRootLogin- Yes
```

```
:wq!
```

19) `passwd root` //change password

20) `service ssh restart` //restart ssh service

21) `service ssh status` //check ssh status

```
root@e38f2cf2e0df:/etc/ssh# vim sshd_config
root@e38f2cf2e0df:/etc/ssh# passwd root
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
root@e38f2cf2e0df:/etc/ssh# service ssh restart
* Restarting OpenBSD Secure Shell server sshd
root@e38f2cf2e0df:/etc/ssh#
```

* sshd is running

22) Login to target 2 now and repeat the steps

```
sudo docker exec -it target2 bash
apt update
apt install openssh-server
apt install python ansible vim iputils-ping openssh-client -y
```

```
cd /etc/ssh
```

```
vim sshd_config
```

```
PermitRootLogin- Yes
:wq!
```

```
passwd root
service ssh restart
service ssh status
exit //exit target 2
```

23) Sudo docker inspect target1 // Check ip of target1

24) Sudo docker inspect target2 // Check ip of target2

25) Sudo docker ps //check docker processes

26) Sudo docker exec -it ansible_master bash //go to master again

27) cd /etc/ansible //go to ansible directory

28) ls -ltr

29) vi hosts //mention both the target ip addresses in this file

172.17.0.3

172.17.0.4

:wq!

30) Try to ping now

ping 172.17.0.3

ping 172.17.0.4

31) ssh-keygen //generate public rsa key

32) ssh-copy-id root@172.17.0.3 //copy rsa pub key to targert 1

33) ssh-copy-id root@172.17.0.4 // copy rsa pub key to targert 2

34) Try to ssh into your target container

ssh root@172.17.0.3 //successful

exit

ssh root@172.17.0.4 //successful

exit

sudo docker ps //check docker processes

35) Go to home directory in ansible_master

cd

36) Vim install_nginx.yml //create install_nginx.yml and input below data

-hosts:all

tasks:

-name:ensure nginx is at the latest version

apt: name=nginx state=latest

-name: start nginx

service:

name:nginx

state:started

```
---
- hosts: all

tasks:
  - name: ensure nginx is at the latest version
    apt: name=nginx state=latest

  - name: start nginx
    service:
      name: nginx
      state: started
```

37) Open two more terminals and login to EC2 instance

38) Check sudo docker ps on both the new terminals

```
* Support: https://ubuntu.com/advantage
Get cloud support with Ubuntu Advantage Cloud Guest:
http://www.ubuntu.com/business/services/cloud

63 packages can be updated.
44 updates are security updates.

New release '18.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Mon Feb 11 11:42:45 2019 from 73.106.74.182
ubuntu@ip-172-31-91-15:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED
STATUS        PORTS    NAMES
34078d38ee6a   ubuntu   "/bin/bash"             About an
hour ago      Up       target2
e38f2cf2e0df   ubuntu   "/bin/bash"             About an
hour ago      Up       target1
fbed1df68fc70  ubuntu   "/bin/bash"             About an
hour ago      Up       ansible_master
ubuntu@ip-172-31-91-15:~$
```

```
* Support: https://ubuntu.com/advantage
Get cloud support with Ubuntu Advantage Cloud Guest:
http://www.ubuntu.com/business/services/cloud

63 packages can be updated.
44 updates are security updates.

New release '18.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Mon Feb 11 11:25:12 2019 from 73.106.74.182
ubuntu@ip-172-31-91-15:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED
STATUS        PORTS    NAMES
34078d38ee6a   ubuntu   "/bin/bash"             About an
hour ago      Up       target2
e38f2cf2e0df   ubuntu   "/bin/bash"             About an
hour ago      Up       target1
fbed1df68fc70  ubuntu   "/bin/bash"             About an
hour ago      Up       ansible_master
ubuntu@ip-172-31-91-15:~$
```

On the second terminal

```
sudo docker exec -it target1 bash
top //no nginx running
```

On the third terminal

```
sudo docker exec -it target2 bash
top //no nginx running
```

39) ansible-playbook install_nginx.yml //run the playbook on the main terminal

```
root@fbeldf68fc70:~# vim install_nginx.yml
root@fbeldf68fc70:~# ansible-playbook install_nginx.yml

PLAY [all] *****

TASK [Gathering Facts] *****
ok: [172.17.0.4]
ok: [172.17.0.3]

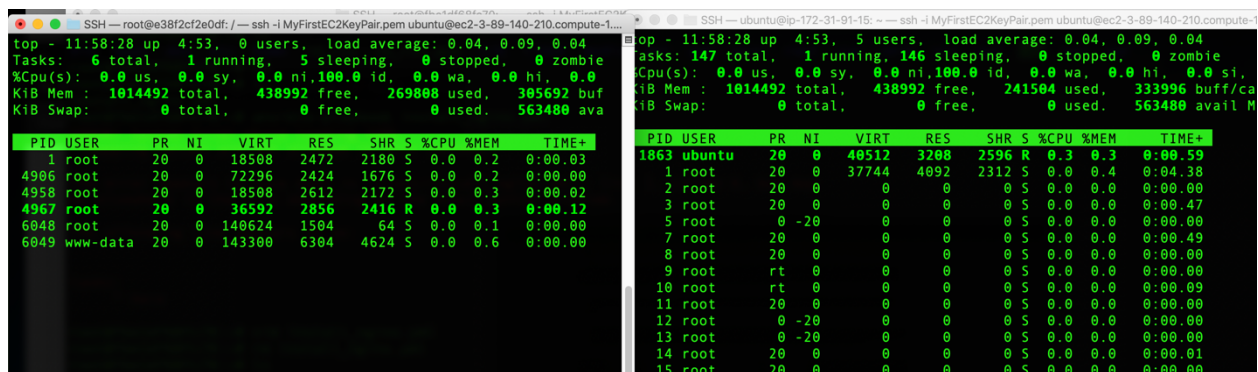
TASK [ensure nginx is at the latest version] *****
changed: [172.17.0.3]
changed: [172.17.0.4]

TASK [start nginx] *****
changed: [172.17.0.4]
changed: [172.17.0.3]

PLAY RECAP *****
172.17.0.3      : ok=3    changed=2    unreachable=0    failed=0
172.17.0.4      : ok=3    changed=2    unreachable=0    failed=0

root@fbeldf68fc70:~#
```

40) nginx will start in the **target containers** as well



The image shows two terminal windows side-by-side. The left window displays the output of the 'top' command, showing system statistics and a list of processes. The right window displays the output of the 'top' command, showing system statistics and a list of processes.

Left terminal output (top):

```
top - 11:58:28 up 4:53, 0 users, load average: 0.04, 0.09, 0.04
Tasks: 6 total, 1 running, 5 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem : 1014492 total, 438992 free, 269808 used, 305692 buff/cache
KiB Swap: 0 total, 0 free, 0 used, 563480 avail
```

| PID | USER | PR | NI | VIRT | RES | SHR | S | %CPU | %MEM | TIME+ |
|------|----------|----|----|--------|------|------|---|------|------|---------|
| 1 | root | 20 | 0 | 18508 | 2472 | 2180 | S | 0.0 | 0.2 | 0:00.03 |
| 4906 | root | 20 | 0 | 72296 | 2424 | 1676 | S | 0.0 | 0.2 | 0:00.00 |
| 4958 | root | 20 | 0 | 18508 | 2612 | 2172 | S | 0.0 | 0.3 | 0:00.02 |
| 4967 | root | 20 | 0 | 36592 | 2856 | 2416 | R | 0.0 | 0.3 | 0:00.12 |
| 6048 | root | 20 | 0 | 140624 | 1504 | 64 | S | 0.0 | 0.1 | 0:00.00 |
| 6049 | www-data | 20 | 0 | 143300 | 6304 | 4624 | S | 0.0 | 0.6 | 0:00.00 |

Right terminal output (top):

```
top - 11:58:28 up 4:53, 5 users, load average: 0.04, 0.09, 0.04
Tasks: 147 total, 1 running, 146 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem : 1014492 total, 438992 free, 241504 used, 333996 buff/cache
KiB Swap: 0 total, 0 free, 0 used, 563480 avail
```

| PID | USER | PR | NI | VIRT | RES | SHR | S | %CPU | %MEM | TIME+ |
|------|--------|----|-----|-------|------|------|---|------|------|---------|
| 1863 | ubuntu | 20 | 0 | 40512 | 3208 | 2596 | R | 0.3 | 0.3 | 0:00.59 |
| 1 | root | 20 | 0 | 37744 | 4092 | 2312 | S | 0.0 | 0.4 | 0:04.38 |
| 2 | root | 20 | 0 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.00 |
| 3 | root | 20 | 0 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.47 |
| 5 | root | 0 | -20 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.00 |
| 7 | root | 20 | 0 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.49 |
| 8 | root | 20 | 0 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.00 |
| 9 | root | rt | 0 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.00 |
| 10 | root | rt | 0 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.09 |
| 11 | root | 20 | 0 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.00 |
| 12 | root | 0 | -20 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.00 |
| 13 | root | 0 | -20 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.00 |
| 14 | root | 20 | 0 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.01 |
| 15 | root | 20 | 0 | 0 | 0 | 0 | S | 0.0 | 0.0 | 0:00.00 |

II. Static website on AWS

Create a new directory in our ubuntu vm

mkdir mysite //create a new doirectory

cd mysite //change directory to mysite

vim index.html //change the content of index.html

```
<html>
<header><title>Hello World !! </title></header>
<body>
Hello world by Shivam Namdeo
</body>
</html>
```

```
docker run -d -P -v $HOME/mysite:/usr/share/nginx/html \
//run a new docker container using nginx
```

A new prompt will open

- --name mysite nginx
It will give you info about your ports

```
sudo docker inspect <containername>
```

Open browser and type the url with the port number //That's it

```
docker stop mysite    //to stop mysite
```

```
docker ps -a            //list all docker containers
```

```
docker start mysite    //start again
```

```
docker port mysite    //port number changes every time so make sure you start it again
```

III. Since my above approach didn't work out on AWS instance, I did this challenge with an alternate approach

In the main Ubuntu terminal perform below steps-

| | |
|----------------------------|-------------------------------------|
| sudo apt update | //Update Ubuntu |
| sudo apt-get install nginx | //Install nginx |
| cd /etc | //go to etc directory |
| ls -ltr | //list files and directories |
| cd nginx/ | //go to nginx directory |
| ls -ltr | //list files and directories |
| cat nginx.conf | //Check the content of nginx.conf |
| sudo vim nginx.conf | //update nginx.conf with below data |

```

events{

}

http {

    server {
        location / {
            root /data/www;
        }
    }
}

```

We have changed the root in this so that our nginx won't redirect to default html

```
ls -ltr //list files and directories
```

```
cd sites-enabled/ //go to sites-enabled directory
```

```
ls -ltr //list files and directories
```

cat default //check default file. root /var/www/html; By default nginx points to this default html path

```
mkdir -p /data /www //create data and www directories
```

```
cd /data/www //go to /data/www
```

```
sudo vim index.html //create index.html
```

```

<!DOCTYPE html>
<html>
<body>

<h1>Hello World by Shivam Namdeo !!!</h1>
<p id="demo">Try below button !! </p>
<button type="button" onclick="myFunction()">Try it</button>

<script>
function myFunction() {
    document.getElementById("demo").innerHTML = "Do you like it ? Let me know ASAP ";
}
</script>

</body>
</html>

```



```
sudo service nginx reload          //reload nginx service
```

Now try to access the public ip address of your ubuntu VM by its public ip

<http://3.89.140.210/>