Configuring Web Server in AWS using ANSIBLE

In this article I am going to deploy a web server on the top of **AWS** cloud with the help of **ANSIBLE.**

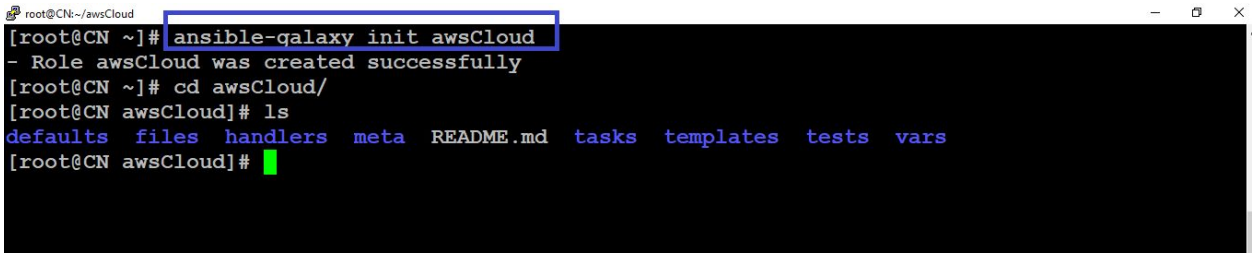
Task Description..

- ➢ Provision EC2 instance through ansible.
- ➢ Retrieve the IP Address of instance using a dynamic inventory concept.
- ➢ Configure the web server through ansible!
- ➢ Create a role for the web server to customize the Instance and deploy the web page to the root directory.

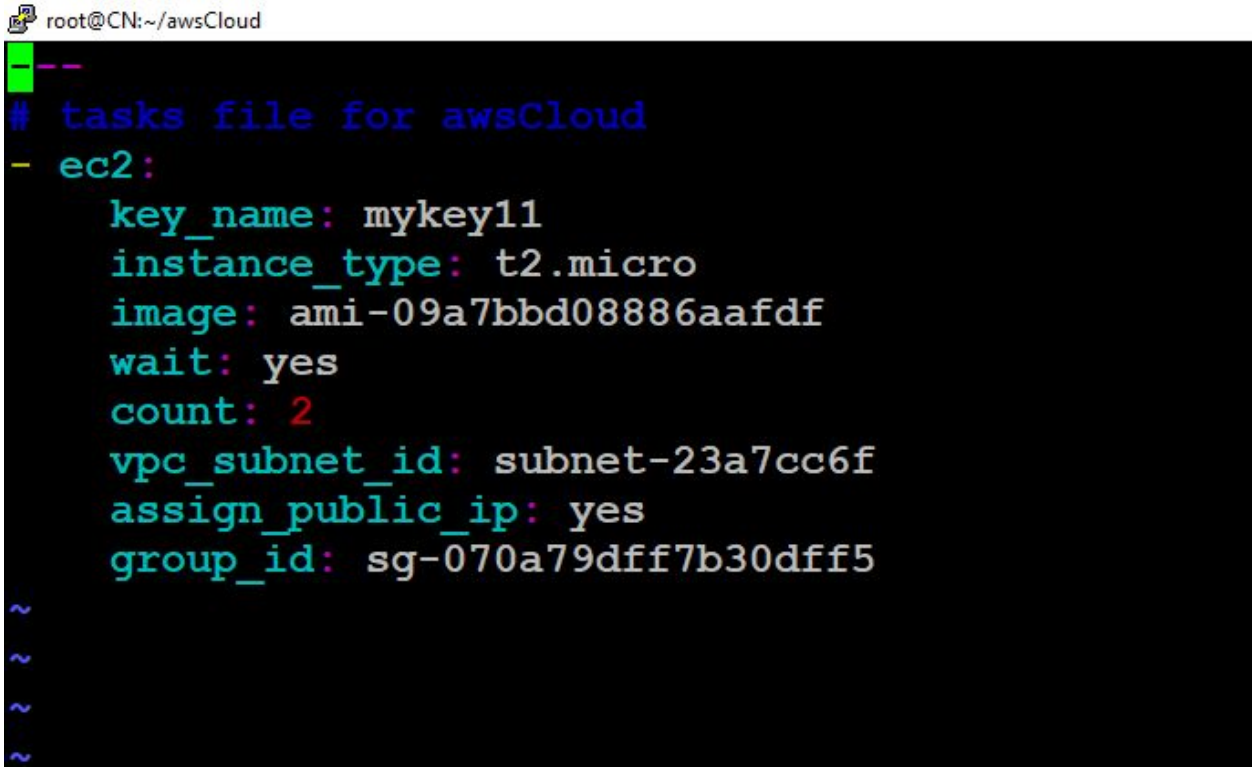
Now Let's start doing the task.

First I will install **boto** and **boto3** libraries.For installing these two libraries i will use these two commands "**pip3 install boto**" and **pip3 install boto3**" I have installed these two libraries so that ANSIBLE will be able to go to the AWS cloud..

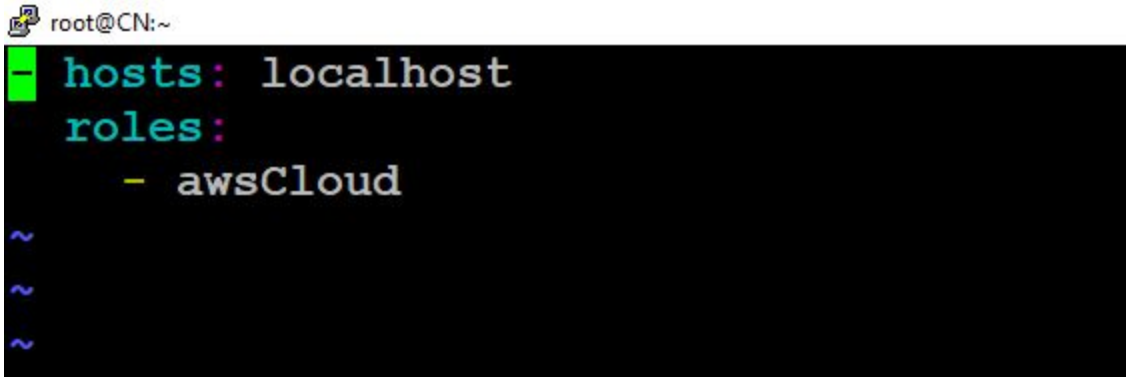
Now i will create a role for launching instances on AWS cloud. I will create a role by using "**ansible-galaxy init** [**role name**]" .



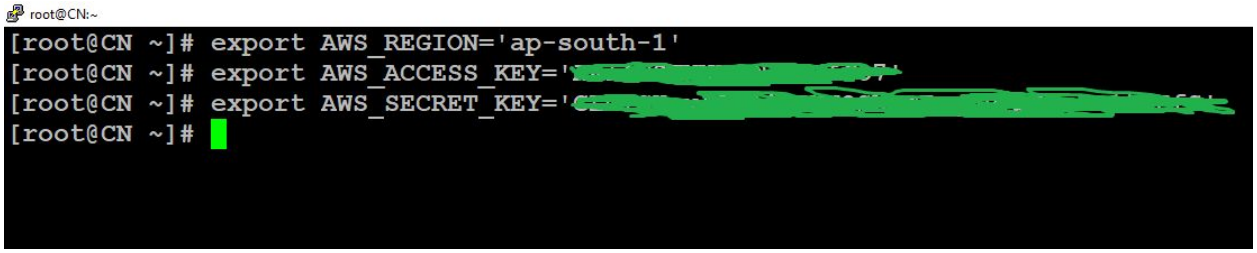
Inside the "**tasks/main.yml**" folder I will write my ansible code that will launch instances on aws cloud.



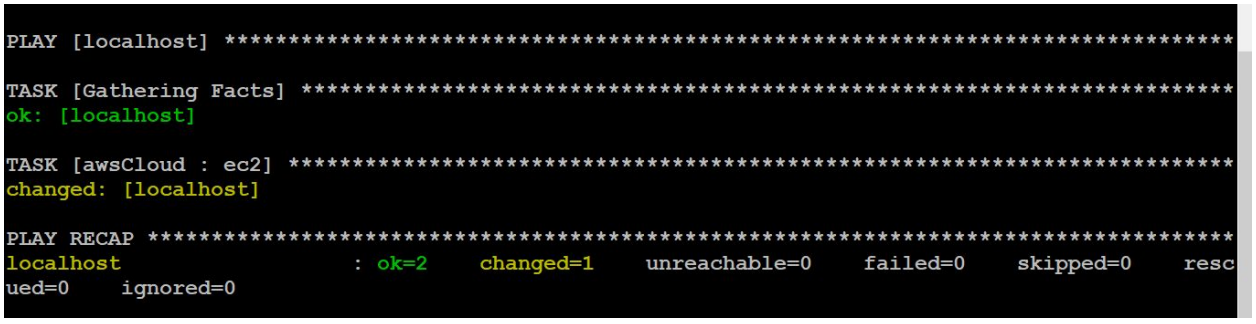
Now I will create a playbook for launching an OS.



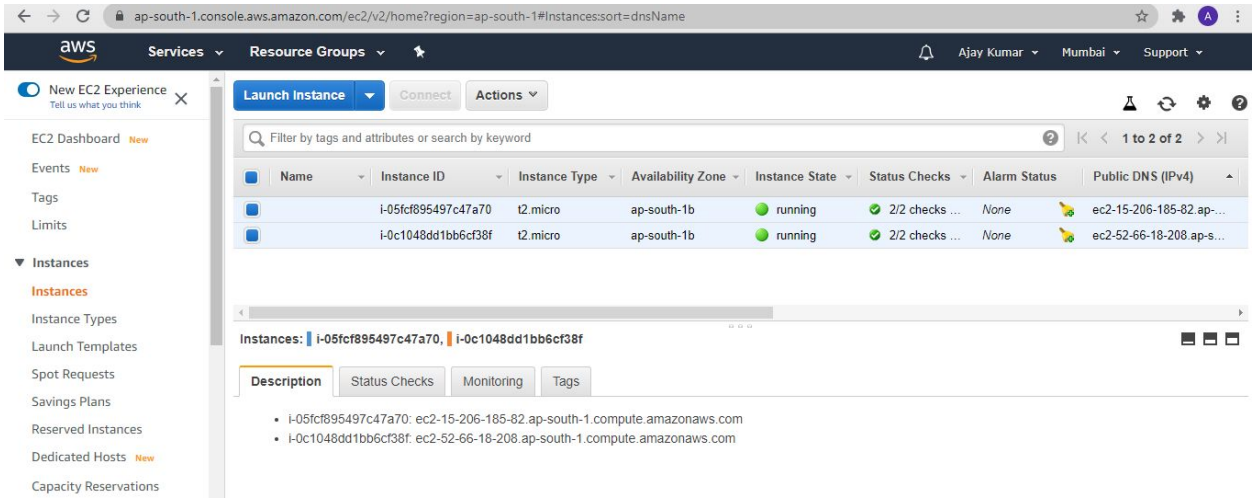
Before running this playbook I will give the access key , secret key and region where ansible will launch the OS..



Now I will run the playbook by using the command "**ansible-playbook** [**your playbook name**]".



EC2 Instances has been launched Successfully.



Now i will create a dynamic inventory that will automatically fetch the instances IP from the AWS cloud.

For this I will download a Script that is written in python language.

You will get this Script from the link below.
https://github.com/ansible/ansible/blob/stable-2.9/contrib/inventory/ec2.py

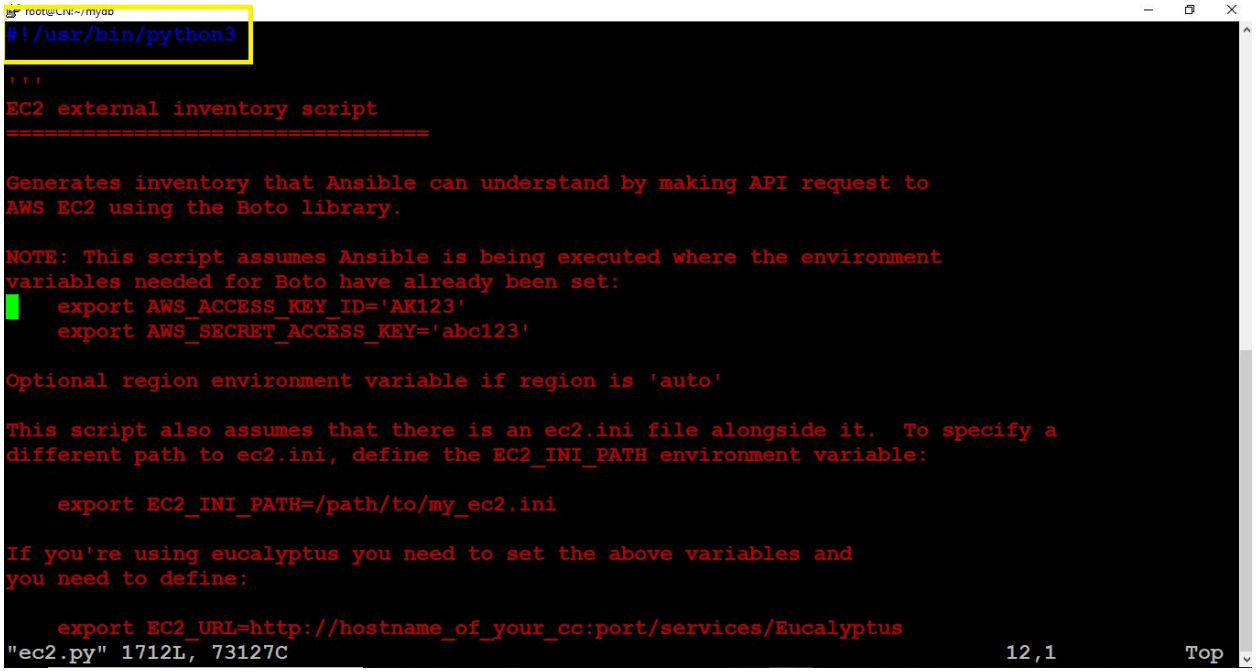
Let's download the script.

```
root@CN:~/mydb
[root@CN mydb]# wget https://raw.githubusercontent.com/ansible/ansible/stable-2.9/contrib/invent
ory/ec2.py
--2020-09-08 15:01:57--  https://raw.githubusercontent.com/ansible/ansible/stable-2.9/contrib/in
ventory/ec2.py
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 151.101.0.133, 151.101.64.133
, 151.101.128.133, ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|151.101.0.133|:443... connec
ted.
HTTP request sent, awaiting response... 200 OK
Length: 73130 (71K) [text/plain]
Saving to: 'ec2.py'

ec2.py                  100%[=============================>]  71.42K   132KB/s    in 0.5s

2020-09-08 15:02:00 (132 KB/s) - 'ec2.py' saved [73130/73130]

[root@CN mydb]# ls
ec2.py
[root@CN mydb]# chmod +x ec2.py
[root@CN mydb]# ls
ec2.py
[root@CN mydb]# vim ec2.py
[root@CN mydb]#
```

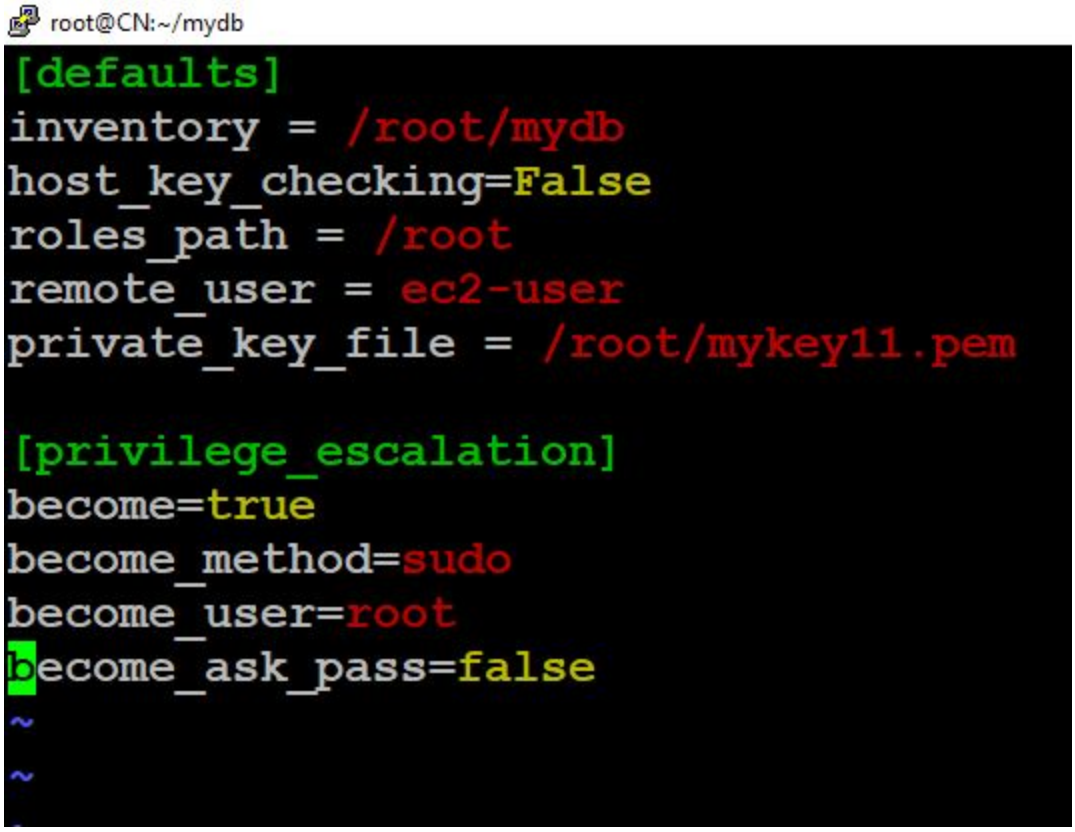
After downloading the script I have made the script executable by using "**chmod +x ec2.py**" command.

This script is written for the python 2 version and my system has python 3 . I have done some change on the script so that at the time of running the script I will not face any error.

Highlighted text is the change that I have made.

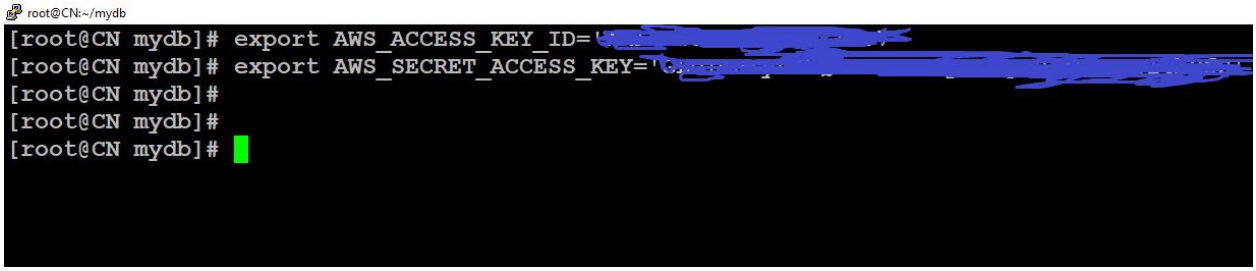


Now i will set the path of the inventory in the ansible configuration file along with i will also set the path of the private key , privilege_esclation and remote_user.

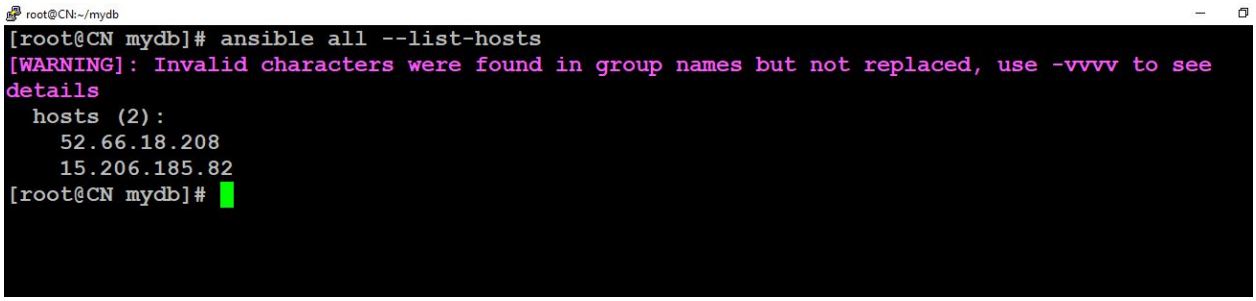


I have sent the private key from my windows to linux OS using **WinSCP** software**.**
After sending the key I have made the key executable using "**chmod 400** [**key_name**]" command.

Now i will set the Access key and Private key ..

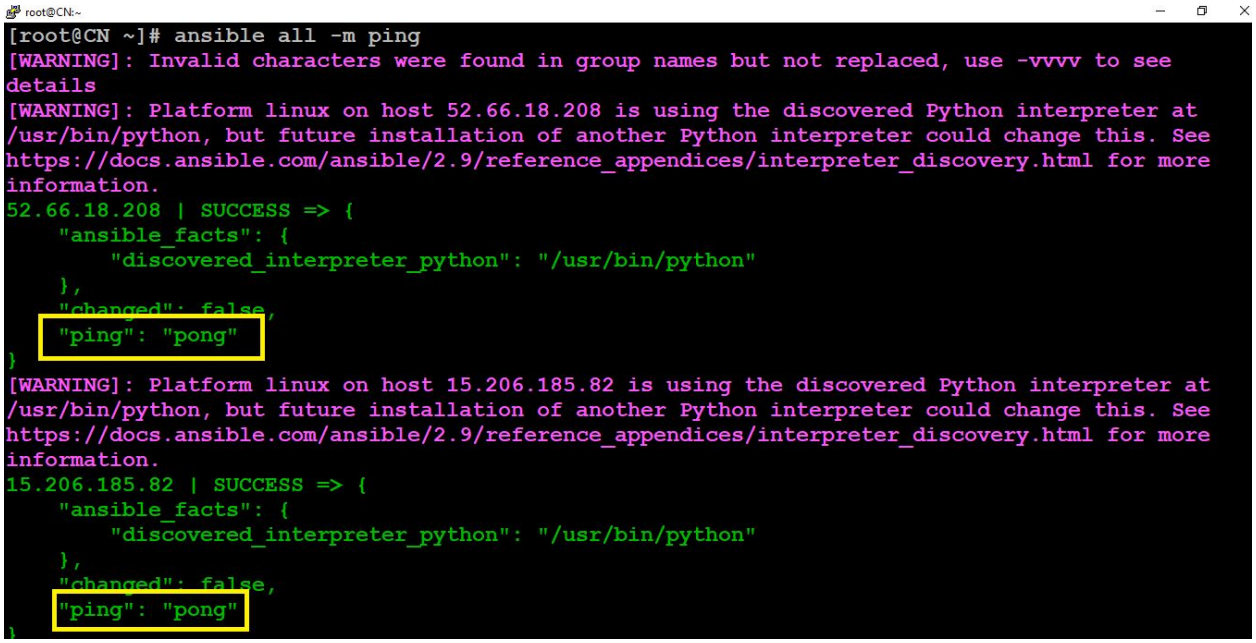


Now i will retrieve the IP of the instances using the “**ansible all --list-hosts**” command.



Finally I have retrieved the IP dynamically.

Let's check the connectivity using “**ansible all -m ping**”



Let’s create a role for setting up the web server in the cloud.

Creating the role.

root@CN:~/cloudhttpd

```
[root@CN ~]# ansible-galaxy init cloudhttpd
- Role cloudhttpd was created successfully
[root@CN ~]# cd cloudhttpd/
[root@CN cloudhttpd]# ls
defaults  files  handlers  meta  README.md  tasks  templates  tests  vars
[root@CN cloudhttpd]#
```

Code for setting up the web server.

root@CN:~/cloudhttpd

```
---
# tasks file for cloudhttpd
- name: "installing software htppd"
  package:
          name: "httpd"
          state: present
- name: "copying webpage to the web server"
  copy:
          content: "Ansible task 2 successfully done"
          dest: "/var/www/html/index.html"
- name: "starting the service of webserver"
  service:
          name: "httpd"
          state: started
          enabled: yes
~
~
```

Creating a Playbook.

root@CN:~

```
- hosts: ec2
  roles:
  - role: cloudhttpd
~
```

Now I will run the playbook using the "**ansible-playbook [your_playbook_name]**" command.

```
[root@CN ~]# ansible-playbook deployweb.yml
[WARNING]: Invalid characters were found in group names but not replaced, use -vvvv to see
details

PLAY [ec2] *********************************************************************************

TASK [Gathering Facts] *********************************************************************
[WARNING]: Platform linux on host 15.206.185.82 is using the discovered Python interpreter at
/usr/bin/python, but future installation of another Python interpreter could change this. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more
information.
ok: [15.206.185.82]
[WARNING]: Platform linux on host 52.66.18.208 is using the discovered Python interpreter at
/usr/bin/python, but future installation of another Python interpreter could change this. See
```

```
ok: [15.206.185.82]
[WARNING]: Platform linux on host 52.66.18.208 is using the discovered Python interpreter at
/usr/bin/python, but future installation of another Python interpreter could change this. See
https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more
information.
ok: [52.66.18.208]

TASK [cloudhttpd : installing software htppd] **********************************************
changed: [52.66.18.208]
changed: [15.206.185.82]

TASK [cloudhttpd : copying webpage to the web server] **************************************
changed: [52.66.18.208]
changed: [15.206.185.82]

TASK [cloudhttpd : starting the service of webserver] **************************************
changed: [15.206.185.82]
changed: [52.66.18.208]

PLAY RECAP *********************************************************************************
15.206.185.82              : ok=4    changed=3    unreachable=0    failed=0    skipped=0    resc
ued=0    ignored=0
52.66.18.208               : ok=4    changed=3    unreachable=0    failed=0    skipped=0    resc
ued=0    ignored=0
```

Playbook has been run Successfully.

Accessing the site from the browser.

Ansible task 2 successfully done

Ansible task 2 successfully done

Finally I have done the task.

Thank You guys for reading this document.

Have a great day.