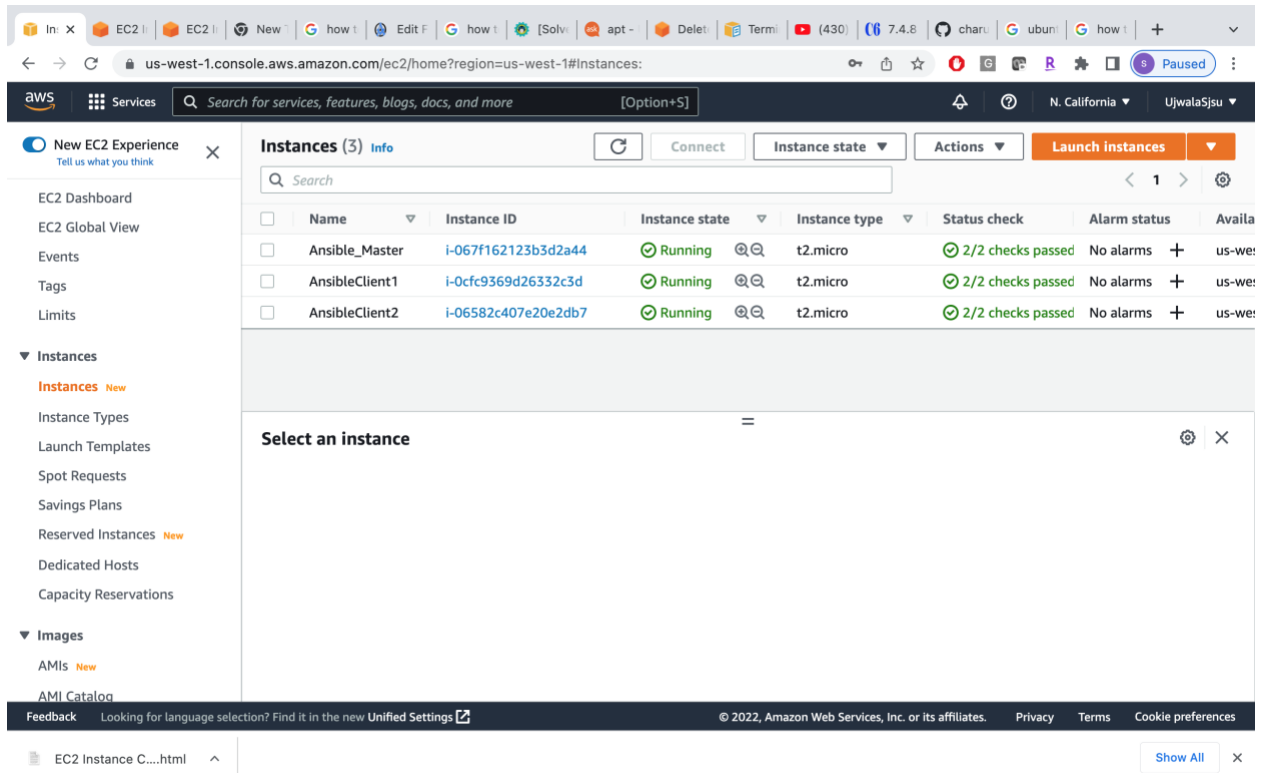


Name : UJWALA MOTE

Student Id : 016711396

Github link : <https://github.com/moteujjwala19/Ansible-ENT-272>

1. Created 3 EC2 instances in AWS named as Ansible_Master which is ansible server and AnsibleClient1 and AnsibleClient2 are hosts as shown below.



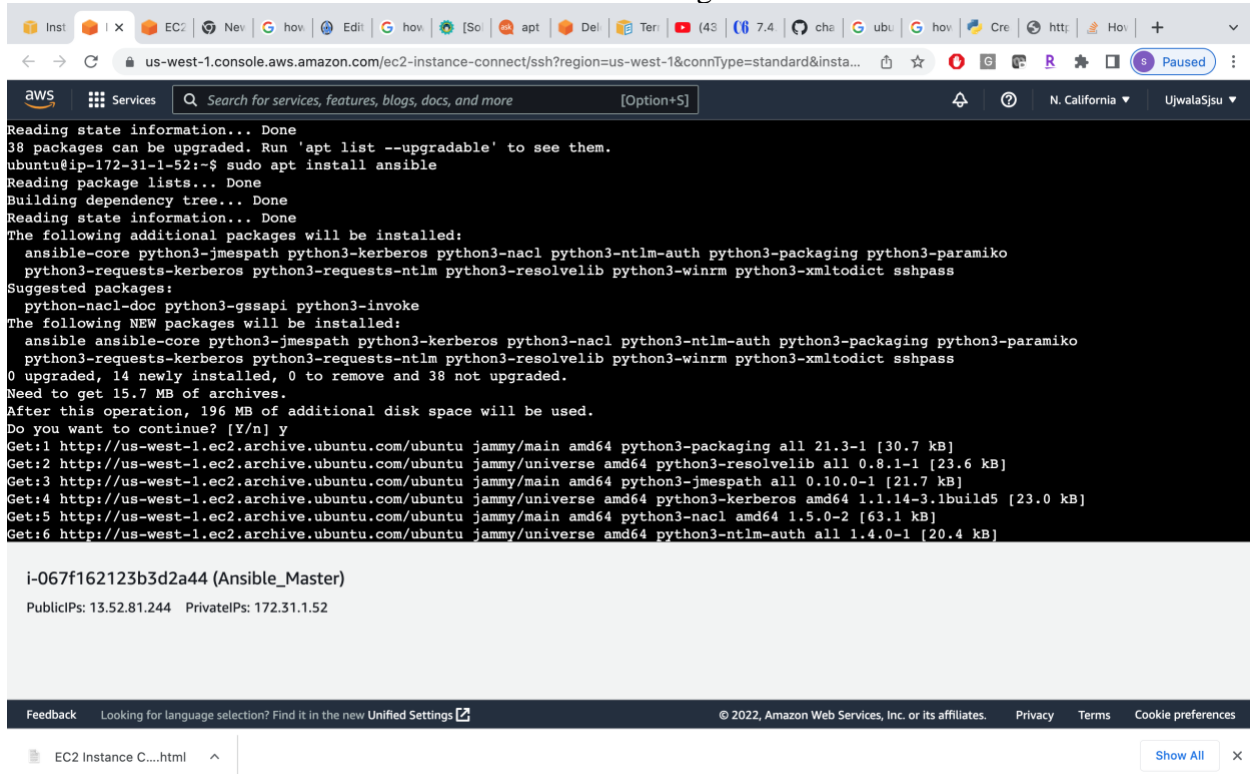
The screenshot displays the AWS Management Console interface for the us-west-1 region. The left sidebar shows navigation options like EC2 Dashboard, Events, Tags, Limits, and a list of services including Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, and AMIs. The main content area is titled 'Instances (3)' and shows a table with three instances:

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
<input type="checkbox"/>	Ansible_Master	i-067f162123b3d2a44	Running	t2.micro	2/2 checks passed	No alarms	us-we
<input type="checkbox"/>	AnsibleClient1	i-0cfc9369d26332c3d	Running	t2.micro	2/2 checks passed	No alarms	us-we
<input type="checkbox"/>	AnsibleClient2	i-06582c407e20e2db7	Running	t2.micro	2/2 checks passed	No alarms	us-we

Below the table, there is a 'Select an instance' modal window. The footer of the console shows the year 2022, copyright information for Amazon Web Services, and links for Privacy, Terms, and Cookie preferences.

2. Updated all machines created in above step by using update command.

3. install ansible on ansible master machine using ansible install command.



The screenshot shows a terminal window within the AWS Management Console. The terminal output displays the command `sudo apt install ansible` being executed on an Ubuntu instance. The output shows that 38 packages can be upgraded and lists the additional packages that will be installed along with their sizes. The installation is successful, and the terminal shows the download progress for various packages from the Ubuntu archive.

```
Reading state information... Done
38 packages can be upgraded. Run 'apt list --upgradable' to see them.
ubuntu@ip-172-31-1-52:~$ sudo apt install ansible
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ansible-core python3-jmespath python3-kerberos python3-nacl python3-ntlm-auth python3-packaging python3-paramiko
  python3-requests-kerberos python3-requests-ntlm python3-resolvelib python3-winrm python3-xlrd python3-xlsxwriter sshpass
Suggested packages:
  python-nacl-doc python3-gssapi python3-invoke
The following NEW packages will be installed:
  ansible ansible-core python3-jmespath python3-kerberos python3-nacl python3-ntlm-auth python3-packaging python3-paramiko
  python3-requests-kerberos python3-requests-ntlm python3-resolvelib python3-winrm python3-xlrd python3-xlsxwriter sshpass
0 upgraded, 14 newly installed, 0 to remove and 38 not upgraded.
Need to get 15.7 MB of archives.
After this operation, 196 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 python3-packaging all 21.3-1 [30.7 kB]
Get:2 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 python3-resolvelib all 0.8.1-1 [23.6 kB]
Get:3 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 python3-jmespath all 0.10.0-1 [21.7 kB]
Get:4 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 python3-kerberos amd64 1.1.14-3.1build5 [23.0 kB]
Get:5 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 python3-nacl amd64 1.5.0-2 [63.1 kB]
Get:6 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 python3-ntlm-auth all 1.4.0-1 [20.4 kB]
```

i-067f162123b3d2a44 (Ansible_Master)

PublicIPs: 13.52.81.244 PrivateIPs: 172.31.1.52

4. Verified if ansible installation is successful using command : ansible --version

The screenshot shows the AWS Management Console interface. The top navigation bar includes the AWS logo, a search bar, and various service icons. The main content area displays the output of an Ansible installation on an Ubuntu EC2 instance. The output shows that the kernel is up-to-date, no services need to be restarted, and no containers need to be restarted. The Ansible version is 2.13.4rc1, and the configuration file is /etc/ansible/ansible.cfg. The output also shows the Ansible configuration details, including the module search path, python module location, collection location, executable location, python version, Jinja version, and libyaml status. The instance ID is i-067f162123b3d2a44 (Ansible_Master) and the public IP is 13.52.81.244.

```
Processing triggers for man-db (2.10.2-1) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-1-52:~$ ansible --version
ansible [core 2.13.4rc1]
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/home/ubuntu/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /home/ubuntu/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.10.4 (main, Jun 29 2022, 12:14:53) [GCC 11.2.0]
  Jinja version = 3.0.3
  libyaml = True
ubuntu@ip-172-31-1-52:~$
```

i-067f162123b3d2a44 (Ansible_Master)
PublicIPs: 13.52.81.244 PrivateIPs: 172.31.1.52

Feedback Looking for language selection? Find it in the new Unified Settings [↗](#) © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

EC2 Instance C...html [Show All](#) ×

5. Installed python on both client VMs as python will be used for server deployment.

The screenshot shows the AWS Management Console interface. The top navigation bar includes the AWS logo, a search bar, and various service icons. The main content area displays the output of a python installation on an Ubuntu EC2 instance. The output shows that the package python is not available, but it is referred to by another package. The output also shows the following packages that replace it: python2-minimal, python2, dh-python, 2to3, python-is-python3. The instance ID is i-0cfc9369d26332c3d (AnsibleClient1) and the public IP is 18.144.170.33.

```
ubuntu@ip-172-31-12-14:~$ sudo apt-get update
Hit:1 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [114 kB]
Get:3 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [99.8 kB]
Hit:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Fetched 214 kB in 0s (650 kB/s)
Reading package lists... Done
ubuntu@ip-172-31-12-14:~$ sudo apt-get install python
Command 'sudod' not found, did you mean:
  command 'sudo' from deb sudo (1.9.9-1ubuntu2)
  command 'sudo' from deb sudo-ldap (1.9.9-1ubuntu2)
Try: sudo apt install <deb name>
ubuntu@ip-172-31-12-14:~$ sudo apt-get install python
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Package python is not available, but is referred to by another package.
This may mean that the package is missing, has been obsoleted, or
is only available from another source
However the following packages replace it:
  python2-minimal python2 dh-python 2to3 python-is-python3

E: Package 'python' has no installation candidate
ubuntu@ip-172-31-12-14:~$
```

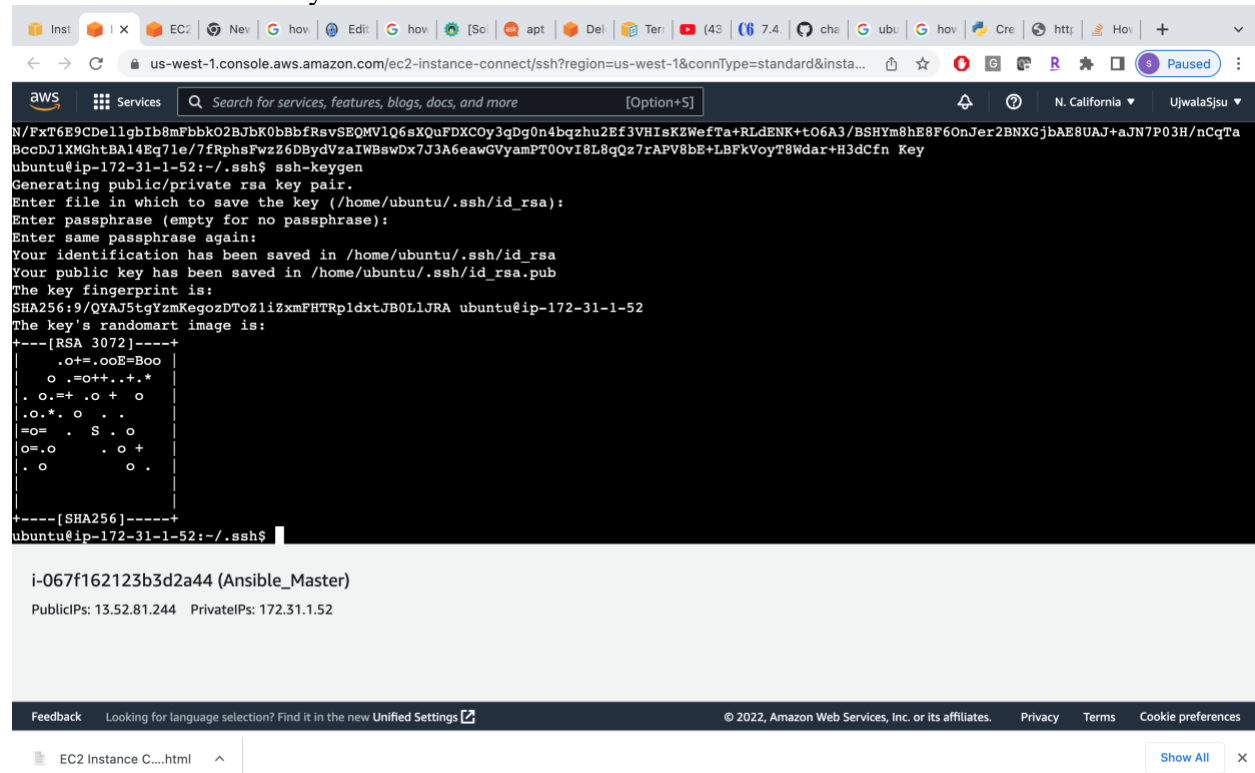
i-0cfc9369d26332c3d (AnsibleClient1)
PublicIPs: 18.144.170.33 PrivateIPs: 172.31.12.14

Feedback Looking for language selection? Find it in the new Unified Settings [↗](#) © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

EC2 Instance C...html [Show All](#) ×

6. Next step is configuring SSH to establish a connection between clients and master VM

First, we need to generate a key on master as shown below which we will put in both client so that we don't need a key to communicate with clients.



The screenshot shows the AWS Management Console interface for an EC2 instance. The terminal window displays the following commands and output:

```
ubuntu@ip-172-31-1-52:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ubuntu/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ubuntu/.ssh/id_rsa
Your public key has been saved in /home/ubuntu/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:9/QYAJ5tgYzmKegozDT0Z1iZxmFHTRpIdxtJB0LLJRA ubuntu@ip-172-31-1-52
The key's randomart image is:
+---[RSA 3072]-----+
|  .o+=.ooE=Boo      |
|  o.=o++..+.*       |
|  .o.=+ .o + o       |
|  .o.*.o . .         |
|  =o= . S . o         |
|  o=.o . o +         |
|  . o . o .          |
|  +-----[SHA256]-----+
ubuntu@ip-172-31-1-52:~$
```

Below the terminal output, the instance details are shown:

```
i-067f162123b3d2a44 (Ansible_Master)
PublicIPs: 13.52.81.244 PrivateIPs: 172.31.1.52
```

The bottom of the screenshot shows the AWS console navigation bar with the "EC2 Instance C...html" link and a "Show All" button.

7. Added above-generated key on both client VM in authorized_key file which is located in .ssh folder using vi authorized_key command as shown below. And verified if key is saved using cat command.

us-west-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=us-west-1&connType=standard&...

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQIDgGTPfSt0QdKqPxsxjgcw9/HERbQ0tXR01/5jLIEmuiP+IADabC1HnjeOddwRAmRP6KQeTdYgJe4z2u+bcO/B7VxP6JVyrLqAR5vN/FxT6E9CDellgbIb8mFbbk02BjBk0bBfRsvSEQMV1Q6sXQuFDXCOy3qDg0n4bgzhu2Ef3VHIsKZWefTa+RLdENK+tO6A3/BSHYm8hE8F6OnJer2BNXGjbaE8UAJ+aJN7P03H/nCqTaBccDJ1XMgHtBA14Eq7le/7fRphsFwzZ6DBydVzaIWBswDx7J3A6eawGVyampFT0OvI8L8qQz7rAPV8bE+LBfKvVoyT8Wdar+H3dCfn Key
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQIDgGTPfSt0QdKqPxsxjgcw9/HERbQ0tXR01/5jLIEmuiP+IADabC1HnjeOddwRAmRP6KQeTdYgJe4z2u+bcO/B7VxP6JVyrLqAR5vN/FxT6E9CDellgbIb8mFbbk02BjBk0bBfRsvSEQMV1Q6sXQuFDXCOy3qDg0n4bgzhu2Ef3VHIsKZWefTa+RLdENK+tO6A3/BSHYm8hE8F6OnJer2BNXGjbaE8UAJ+aJN7P03H/nCqTaBccDJ1XMgHtBA14Eq7le/7fRphsFwzZ6DBydVzaIWBswDx7J3A6eawGVyampFT0OvI8L8qQz7rAPV8bE+LBfKvVoyT8Wdar+H3dCfn Key
MnRVCGv3srxSS2OdQpIgmI2aK4IXyL2Cwn4+69Vea5IhBRUKP+Jm9Kck5abd/12M4LFgfGLAcOSXMeYVbzhKfep49PPsoVnzZaQ3aCoHEIgmmt/0U5TuiKrVBYtkglrFQY5brkO3yHhijNcPlnrHVNdVpFjbloZnal/1HT7dDBR0ViobAP0VgBiKH0gPzeHiMIBBdlu9dby9Qyi+K5UfmK8Mu6UQBSNstoguc14cC98GMVkcX5SOG5jO7i1R2x1wQqckBTAB6vI87Az9xdRii8eR2CkvDaxCQHJiaKtCnYdJ8WJTBaxHAnVmd+YaJB2pCPrdh43RfE/67Qd/71114KoFgE8+f/Qs0= ubuntu@ip-172-31-1-52
```

"authorized_keys" 2Lr, 960B

2,574 All

i-06582c407e20e2db7 (AnsibleClient2)

PublicIPs: 54.177.99.49 PrivateIPs: 172.31.14.10

Feedback Looking for language selection? Find it in the new Unified Settings

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

EC2 Instance C...html Show All

us-west-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=us-west-1&connType=standard&insta...

```
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Package python is not available, but is referred to by another package.
This may mean that the package is missing, has been obsoleted, or
is only available from another source
However the following packages replace it:
  python2-minimal python2 dh-python 2to3 python-is-python3

E: Package 'python' has no installation candidate
ubuntu@ip-172-31-12-14:~$ cd .ssh
ubuntu@ip-172-31-12-14:~/.ssh$ ls
authorized_keys
ubuntu@ip-172-31-12-14:~/.ssh$ sudo nano authorized_keys
ubuntu@ip-172-31-12-14:~/.ssh$ cat authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQIDgGTPfSt0QdKqPxsxjgcw9/HERbQ0tXR01/5jLIEmuiP+IADabC1HnjeOddwRAmRP6KQeTdYgJe4z2u+bcO/B7VxP6JVyrLqAR5vN/FxT6E9CDellgbIb8mFbbk02BjBk0bBfRsvSEQMV1Q6sXQuFDXCOy3qDg0n4bgzhu2Ef3VHIsKZWefTa+RLdENK+tO6A3/BSHYm8hE8F6OnJer2BNXGjbaE8UAJ+aJN7P03H/nCqTaBccDJ1XMgHtBA14Eq7le/7fRphsFwzZ6DBydVzaIWBswDx7J3A6eawGVyampFT0OvI8L8qQz7rAPV8bE+LBfKvVoyT8Wdar+H3dCfn Key
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQIDgGTPfSt0QdKqPxsxjgcw9/HERbQ0tXR01/5jLIEmuiP+IADabC1HnjeOddwRAmRP6KQeTdYgJe4z2u+bcO/B7VxP6JVyrLqAR5vN/FxT6E9CDellgbIb8mFbbk02BjBk0bBfRsvSEQMV1Q6sXQuFDXCOy3qDg0n4bgzhu2Ef3VHIsKZWefTa+RLdENK+tO6A3/BSHYm8hE8F6OnJer2BNXGjbaE8UAJ+aJN7P03H/nCqTaBccDJ1XMgHtBA14Eq7le/7fRphsFwzZ6DBydVzaIWBswDx7J3A6eawGVyampFT0OvI8L8qQz7rAPV8bE+LBfKvVoyT8Wdar+H3dCfn Key
3+DjS0ymmmct3x1CjVrvPrRiH3mYQ1uY/PRSeac6wCmjJEaheVM9fPTMnRVCGv3srxSS2OdQpIgmI2aK4IXyL2Cwn4+69Vea5IhBRUKP+Jm9Kck5abd/12M4LFgfGLAcOSXMeYVbzhKfep49PPsoVnzZaQ3aCoHEIgmmt/0U5TuiKrVBYtkglrFQY5brkO3yHhijNcPlnrHVNdVpFjbloZnal/1HT7dDBR0ViobAP0VgBiKH0gPzeHiMIBBdlu9dby9Qyi+K5UfmK8Mu6UQBSNstoguc14cC98GMVkcX5SOG5jO7i1R2x1wQqckBTAB6vI87Az9xdRii8eR2CkvDaxCQHJiaKtCnYdJ8WJTBaxHAnVmd+YaJB2pCPrdh43RfE/67Qd/71114KoFgE8+f/Qs0= ubuntu@ip-172-31-1-52
ubuntu@ip-172-31-12-14:~/.ssh$
```

i-0cfc9369d26332c3d (AnsibleClient1)

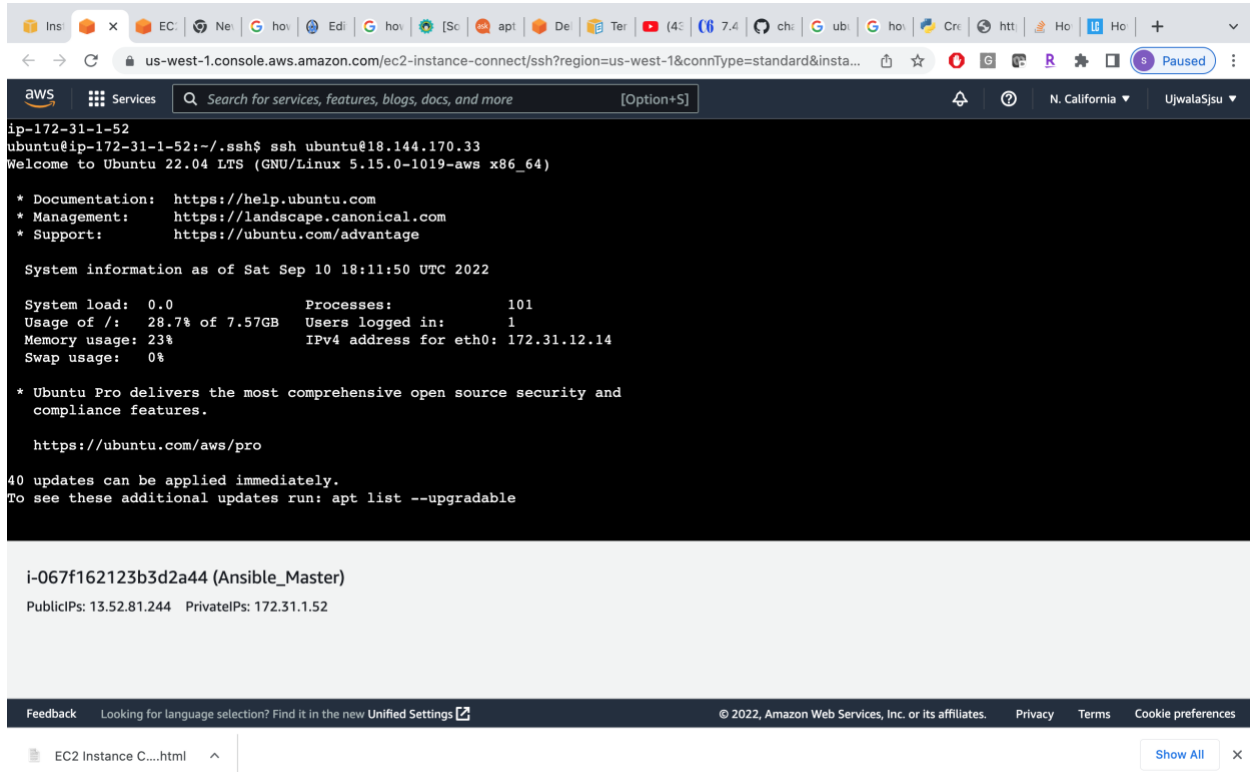
PublicIPs: 18.144.170.33 PrivateIPs: 172.31.12.14

Feedback Looking for language selection? Find it in the new Unified Settings

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

EC2 Instance C...html Show All

8. Verified if SSH successful as shown below.



The screenshot shows the AWS Management Console interface. The top navigation bar includes the AWS logo, a 'Services' menu, a search bar, and a 'Paused' button. The main content area displays the terminal output of an SSH session. The terminal shows the user 'ubuntu' logging into an instance with IP 'ip-172-31-1-52'. The output includes the Ubuntu 22.04 LTS welcome message, system information (Sat Sep 10 18:11:50 UTC 2022), and a list of updates that can be applied immediately. Below the terminal output, the instance ID 'i-067f162123b3d2a44 (Ansible_Master)' and its public and private IP addresses are listed. The bottom of the console shows a feedback bar and a 'Show All' button.

```
ip-172-31-1-52
ubuntu@ip-172-31-1-52:~/.ssh$ ssh ubuntu@18.144.170.33
Welcome to Ubuntu 22.04 LTS (GNU/Linux 5.15.0-1019-aws x86_64)

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

System information as of Sat Sep 10 18:11:50 UTC 2022

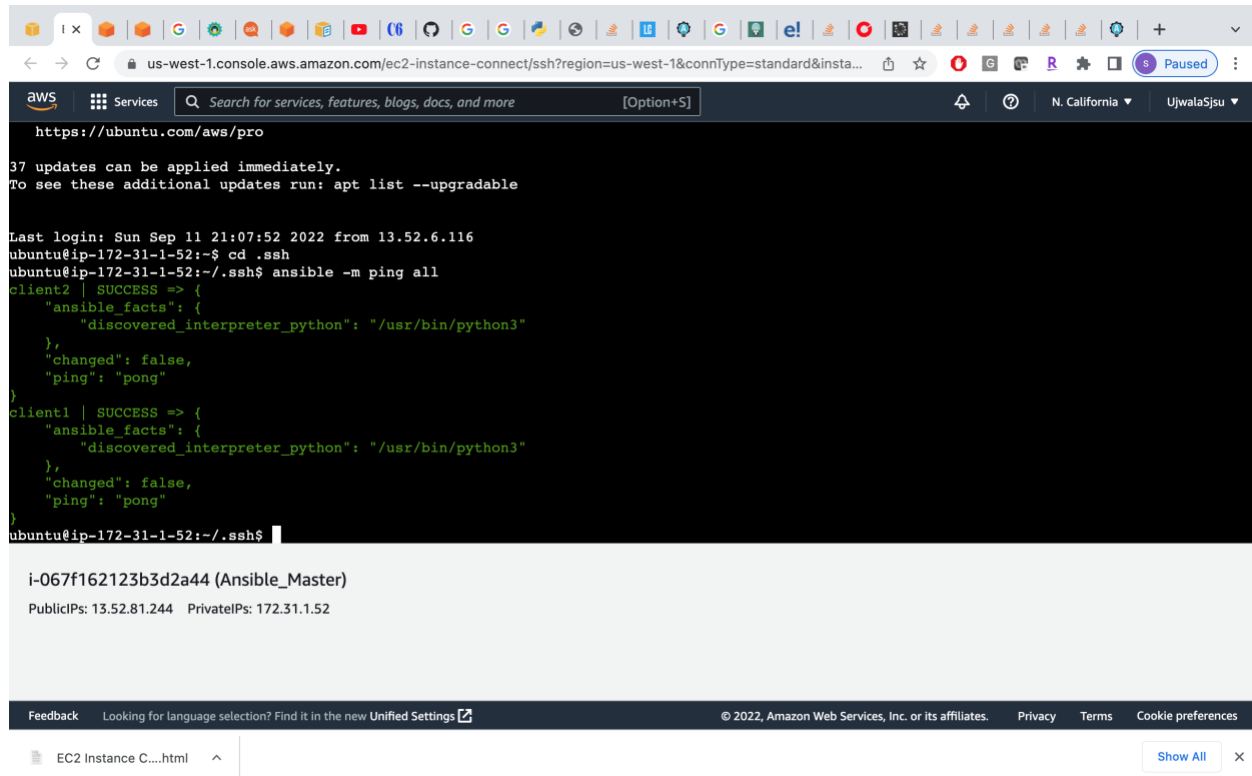
System load:  0.0               Processes:           101
Usage of /:   28.7% of 7.57GB    Users logged in:    1
Memory usage: 23%              IPv4 address for eth0: 172.31.12.14
Swap usage:   0%

 * Ubuntu Pro delivers the most comprehensive open source security and
 * compliance features.
 *
 * https://ubuntu.com/aws/pro

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

i-067f162123b3d2a44 (Ansible_Master)
PublicIPs: 13.52.81.244 PrivateIPs: 172.31.1.52

9. Also, check the connection by using ping command. Success response from both clients as shown below after running the ping command.



The screenshot shows a web browser window with the AWS Management Console interface. The terminal output is as follows:

```
https://ubuntu.com/aws/pro
37 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Last login: Sun Sep 11 21:07:52 2022 from 13.52.6.116
ubuntu@ip-172-31-1-52:~$ cd .ssh
ubuntu@ip-172-31-1-52:~/.ssh$ ansible -m ping all
client2 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
client1 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
ubuntu@ip-172-31-1-52:~/.ssh$
```

Below the terminal output, the instance details are shown:

i-067f162123b3d2a44 (Ansible_Master)
PublicIPs: 13.52.81.244 PrivateIPs: 172.31.1.52

The bottom of the screenshot shows the AWS console footer with links for Feedback, language selection, Unified Settings, and copyright information for 2022.

10. To set up an ansible host added host details in server's host file located under /etc/ansible repo.

The screenshot shows a web browser window with the AWS Management Console interface. The terminal window is running GNU nano 6.2 and editing the file /etc/ansible/hosts. The content of the file is as follows:

```
## www[001:006].example.com

# Ex 3: A collection of database servers in the 'dbservers' group:

## [dbservers]
##
## db01.intranet.mydomain.net
## db02.intranet.mydomain.net
## 10.25.1.56
## 10.25.1.57

# Here's another example of host ranges, this time there are no
# leading 0s:

## db-[99:101]-node.example.com

[hosts]
client1 ansible_ssh_host=54.177.99.49
client2 ansible_ssh_host=18.144.170.33
```

Below the editor, the instance details are shown: i-067f162123b3d2a44 (Ansible_Master), with PublicIPs: 13.52.81.244 and PrivateIPs: 172.31.1.52. The bottom of the console shows a feedback bar and a file explorer with 'EC2 Instance C...html'.

11. created below python file on both webserver to display msg on screen when it gets a request from ansible server via playbook.

Client1:deploy.py

us-west-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=us-west-1&connType=standard&i...

```
from http.server import BaseHTTPRequestHandler, HTTPServer
import time

hostName = "localhost"
serverPort = 8080

class MyServer(BaseHTTPRequestHandler):
    def do_GET(self):
        self.send_response(200)
        self.send_header("Content-type", "text/html")
        self.end_headers()
        self.wfile.write(bytes("Hello World from SJSU-1", "utf-8"))

if __name__ == "__main__":
    webServer = HTTPServer((hostName, serverPort), MyServer)
    print("Server started http://%s:%s" % (hostName, serverPort))

    try:
        webServer.serve_forever()
    except KeyboardInterrupt:
        pass

    webServer.server_close()
    print("Server stopped.")
```

"deploy.py" 24L, 666B

i-0cf9369d26332c3d (AnsibleClient1)

PublicIPs: 18.144.170.33 PrivateIPs: 172.31.12.14

Feedback Looking for language selection? Find it in the new Unified Settings

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

EC2 Instance C....html Show All

client2 : deploy.py

us-west-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=us-west-1&connType=standard&i...

```
from http.server import BaseHTTPRequestHandler, HTTPServer
import time

hostName = "localhost"
serverPort = 8080

class MyServer(BaseHTTPRequestHandler):
    def do_GET(self):
        self.send_response(200)
        self.send_header("Content-type", "text/html")
        self.end_headers()
        self.wfile.write(bytes("Hello World from SJSU-2", "utf-8"))

if __name__ == "__main__":
    webServer = HTTPServer((hostName, serverPort), MyServer)
    print("Server started http://%s:%s" % (hostName, serverPort))

    try:
        webServer.serve_forever()
    except KeyboardInterrupt:
        pass

    webServer.server_close()
```

"deploy.py" [readonly] 23L, 629B

i-06582c407e20e2db7 (AnsibleClient2)

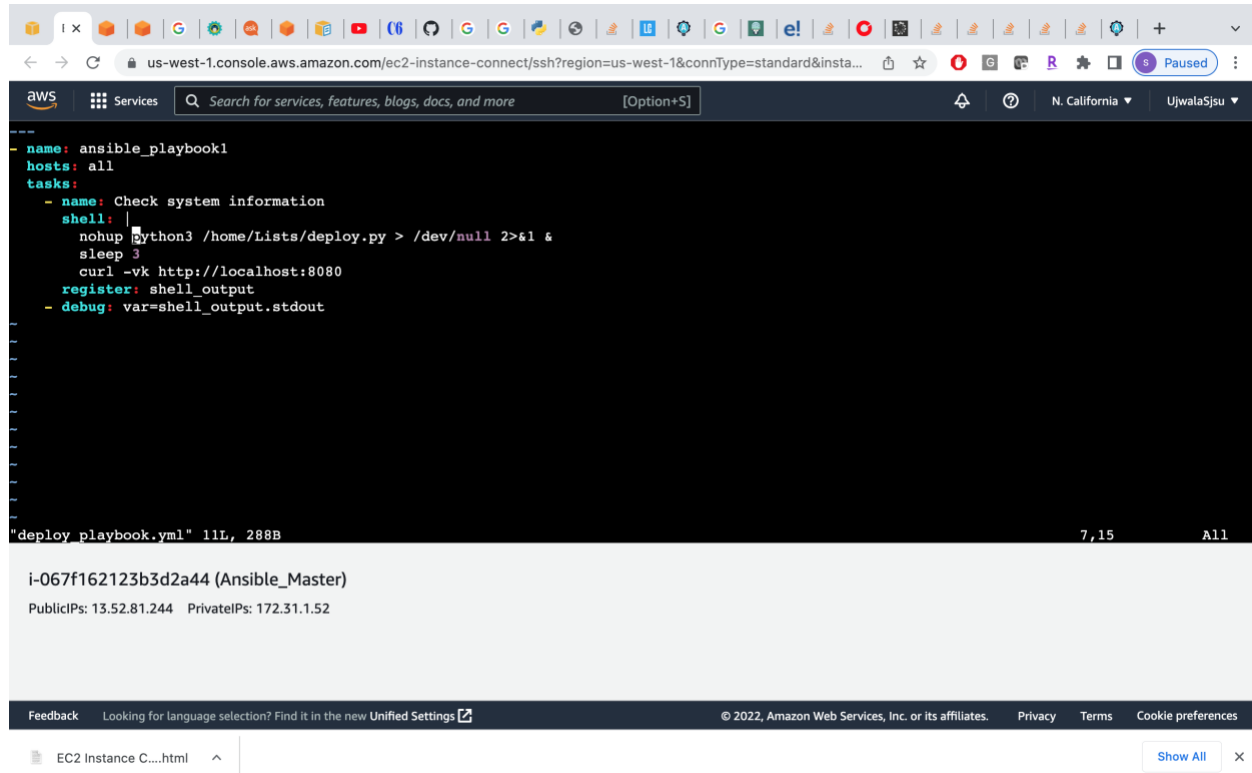
PublicIPs: 54.177.99.49 PrivateIPs: 172.31.14.10

Feedback Looking for language selection? Find it in the new Unified Settings

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

EC2 Instance C....html Show All

12. Created following deploy_playbook.yml which has a set of commands to run the python file on both the clients. As hosts value is 'all' in yml file it will include all the hosts added in ansible host config file which was done in earlier step.

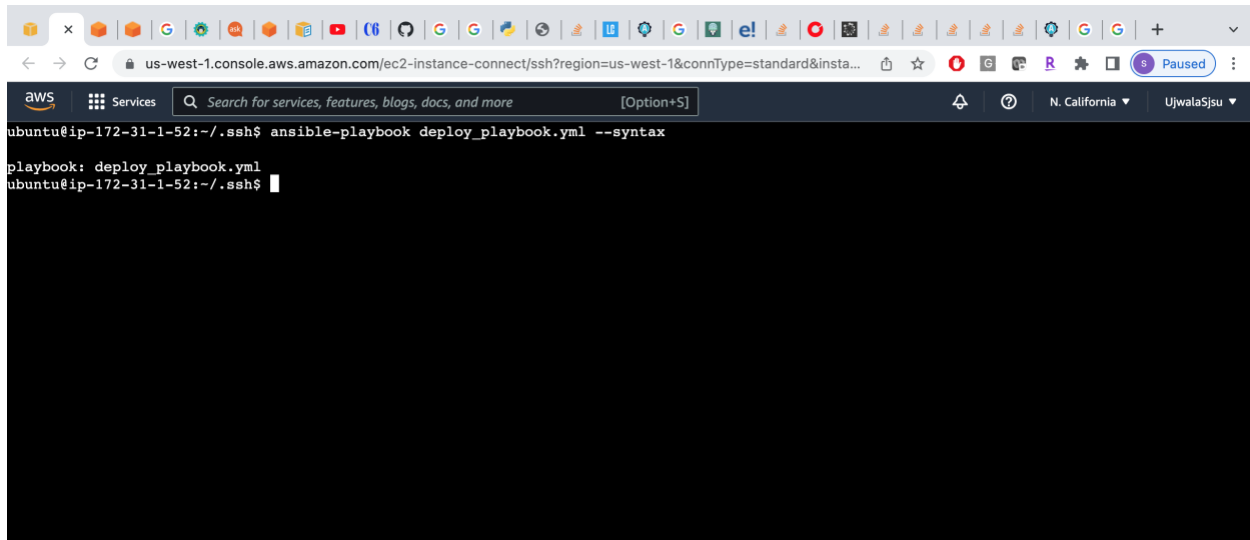


The screenshot shows the AWS Management Console interface for an EC2 instance. The top navigation bar includes the AWS logo, a search bar, and the user's location (N. California) and name (UjwalaSjsu). The main content area displays the configuration for an Ansible playbook named 'ansible_playbook1'. The configuration is as follows:

```
- name: ansible_playbook1
  hosts: all
  tasks:
    - name: Check system information
      shell: |
        nohup python3 /home/Lists/deploy.py > /dev/null 2>&1 &
        sleep 3
        curl -vk http://localhost:8080
      register: shell_output
    - debug: var=shell_output.stdout
```

Below the configuration, the instance ID is shown as 'i-067f162123b3d2a44 (Ansible_Master)'. The public IP address is '13.52.81.244' and the private IP address is '172.31.1.52'. The bottom of the console shows a feedback bar and a tab for 'EC2 Instance C....html'.

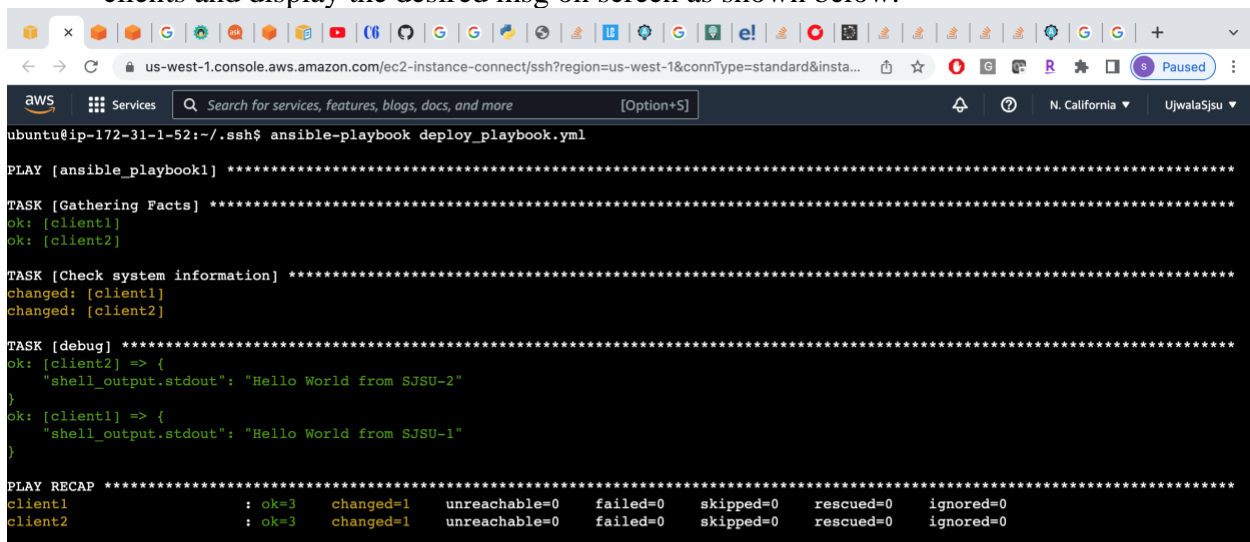
13. check if syntax is correct by executing below command. If nothing comes on screen but the name that means there are no syntactical issues in file.



```
ubuntu@ip-172-31-1-52:~/.ssh$ ansible-playbook deploy_playbook.yml --syntax
playbook: deploy_playbook.yml
ubuntu@ip-172-31-1-52:~/.ssh$
```

i-067f162123b3d2a44 (Ansible_Master)
PublicIPs: 13.52.81.244 PrivateIPs: 172.31.1.52

14. Next step is to execute `deploy_playbook.yml` which will deploy webserver on both clients and display the desired msg on screen as shown below.



```
ubuntu@ip-172-31-1-52:~/.ssh$ ansible-playbook deploy_playbook.yml
PLAY [ansible_playbook1] *****

TASK [Gathering Facts] *****
ok: [client1]
ok: [client2]

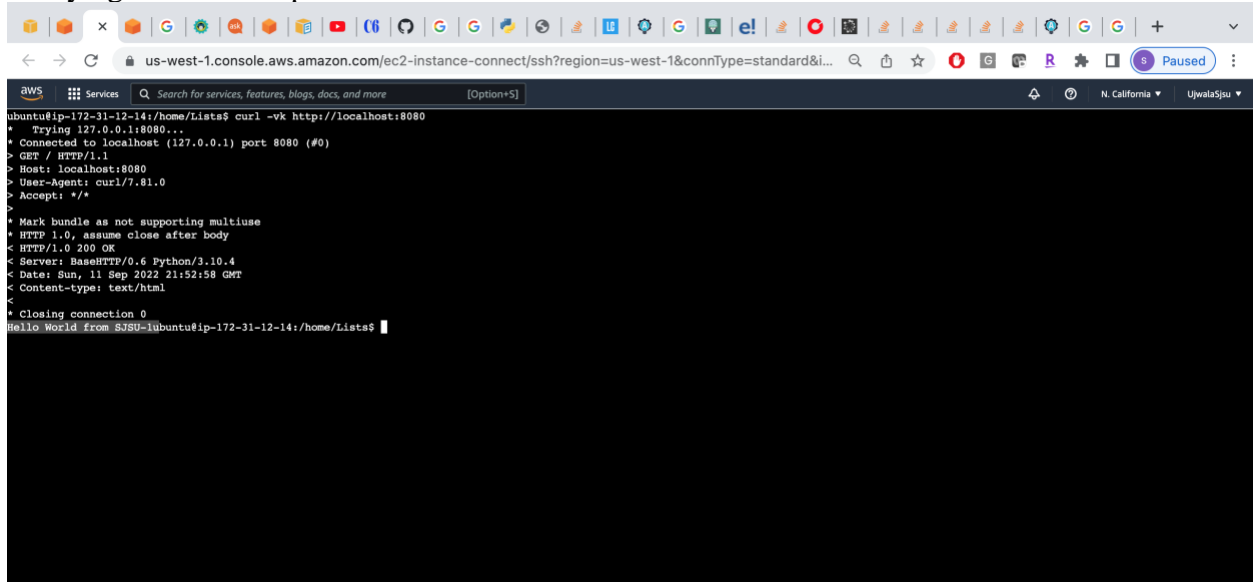
TASK [Check system information] *****
changed: [client1]
changed: [client2]

TASK [debug] *****
ok: [client2] => {
  "shell_output.stdout": "Hello World from SJSU-2"
}
ok: [client1] => {
  "shell_output.stdout": "Hello World from SJSU-1"
}

PLAY RECAP *****
client1      : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
client2      : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

i-067f162123b3d2a44 (Ansible_Master)
PublicIPs: 13.52.81.244 PrivateIPs: 172.31.1.52

verifying if server is up on individual clients

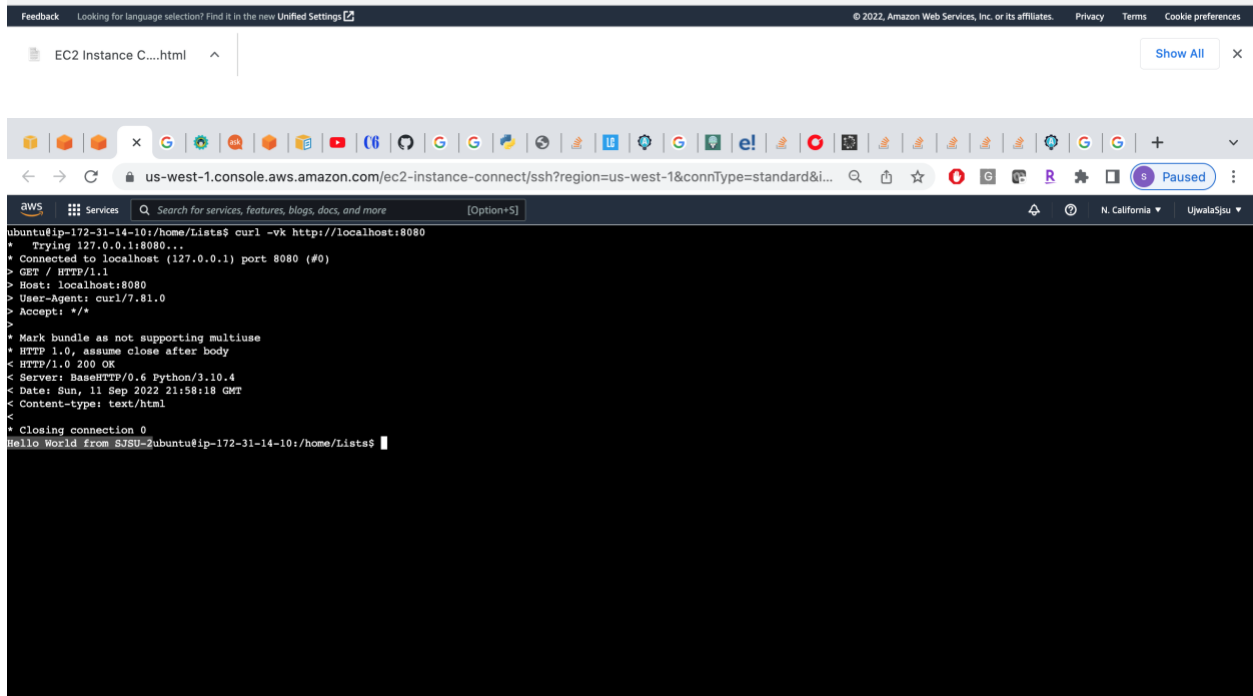


The screenshot shows the AWS Management Console terminal for an EC2 instance. The terminal output shows a successful curl command executed on the instance, connecting to localhost:8080 and receiving a 'Hello World' response from the application.

```
ubuntu@ip-172-31-12-14:/home/Lists$ curl -vk http://localhost:8080
* Trying 127.0.0.1:8080...
* Connected to localhost (127.0.0.1) port 8080 (#0)
> GET / HTTP/1.1
> Host: localhost:8080
> User-Agent: curl/7.81.0
> Accept: */*
* Mark bundle as not supporting multiuse
* HTTP/1.0, assume close after body
< HTTP/1.0 200 OK
< Server: BaseHTTP/0.6 Python/3.10.4
< Date: Sun, 11 Sep 2022 21:52:58 GMT
< Content-type: text/html
* Closing connection 0
Hello World from 83BU-1ubuntu@ip-172-31-12-14:/home/Lists$
```

i-0fc9369d26332c3d (AnsibleClient1)

PublicIPs: 18.144.170.33 PrivateIPs: 172.31.12.14



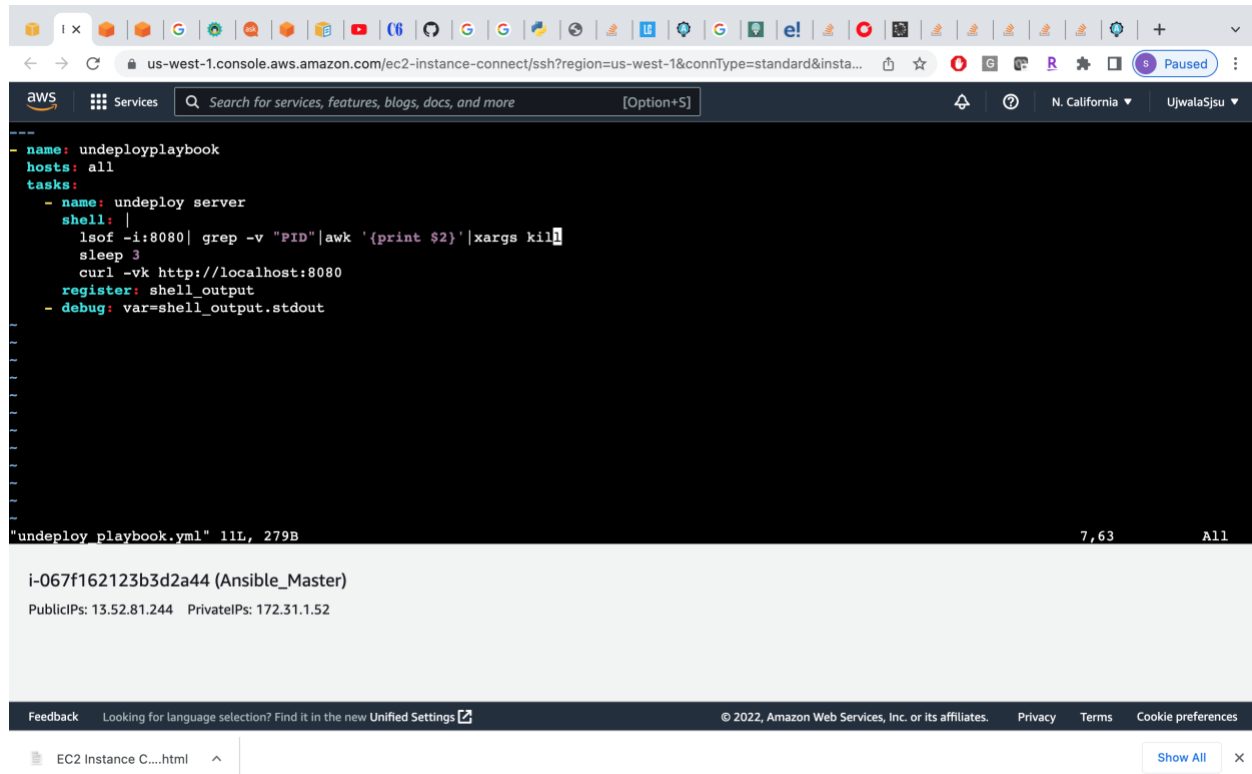
The screenshot shows the AWS Management Console terminal for an EC2 instance. The terminal output shows a successful curl command executed on the instance, connecting to localhost:8080 and receiving a 'Hello World' response from the application.

```
ubuntu@ip-172-31-14-10:/home/Lists$ curl -vk http://localhost:8080
* Trying 127.0.0.1:8080...
* Connected to localhost (127.0.0.1) port 8080 (#0)
> GET / HTTP/1.1
> Host: localhost:8080
> User-Agent: curl/7.81.0
> Accept: */*
* Mark bundle as not supporting multiuse
* HTTP/1.0, assume close after body
< HTTP/1.0 200 OK
< Server: BaseHTTP/0.6 Python/3.10.4
< Date: Sun, 11 Sep 2022 21:58:18 GMT
< Content-type: text/html
* Closing connection 0
Hello World from 83BU-2ubuntu@ip-172-31-14-10:/home/Lists$
```

i-06582c407e20e2db7 (AnsibleClient2)

PublicIPs: 54.177.99.49 PrivateIPs: 172.31.14.10

15. created undeploy_playbook.yml with commands to undeploy server as shown below

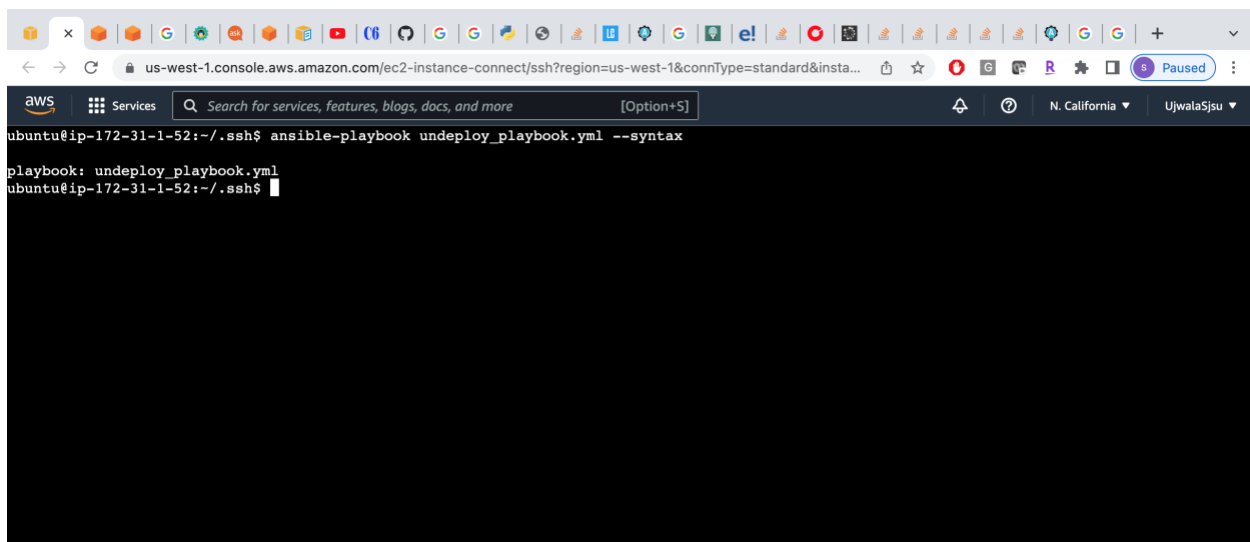


The screenshot shows the AWS Management Console interface. The top navigation bar includes the AWS logo, a 'Services' menu, a search bar, and a '[Option+S]' button. The main content area displays the content of the 'undeploy_playbook.yml' file. The file content is as follows:

```
- name: undeployplaybook
  hosts: all
  tasks:
    - name: undeploy server
      shell: |
        lsof -i:8080 | grep -v "PID"|awk '{print $2}'|xargs kill
        sleep 3
        curl -vk http://localhost:8080
      register: shell_output
    - debug: var=shell_output.stdout
```

Below the file content, the instance details are shown: 'i-067f162123b3d2a44 (Ansible_Master)' with PublicIPs: 13.52.81.244 and PrivateIPs: 172.31.1.52. The bottom of the console shows a footer with 'Feedback', a language selection prompt, copyright information '© 2022, Amazon Web Services, Inc. or its affiliates.', and links for 'Privacy', 'Terms', and 'Cookie preferences'. A tab at the bottom indicates the current view is 'EC2 Instance C....html'.

check syntax of undeploy_playbook.yml file using ansible syntax command

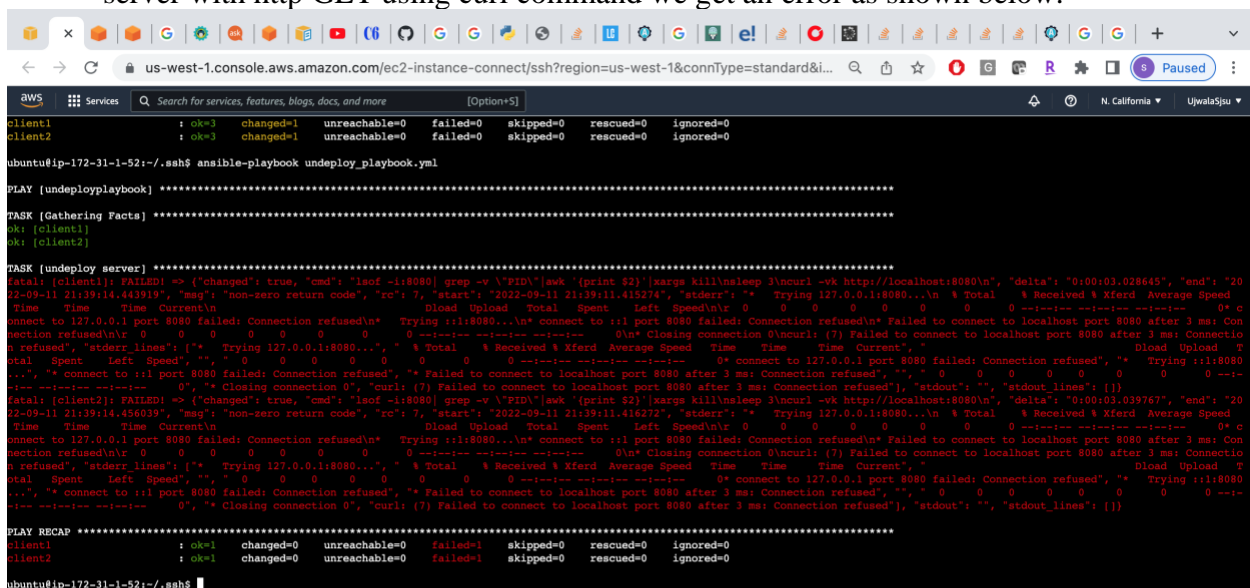


i-067f162123b3d2a44 (Ansible_Master)

PublicIPs: 13.52.81.244 PrivateIPs: 172.31.1.52



16. Once we run `undeploy_playbook.yml` it will kill the webserver and when we try to reach server with http GET using curl command we get an error as shown below.

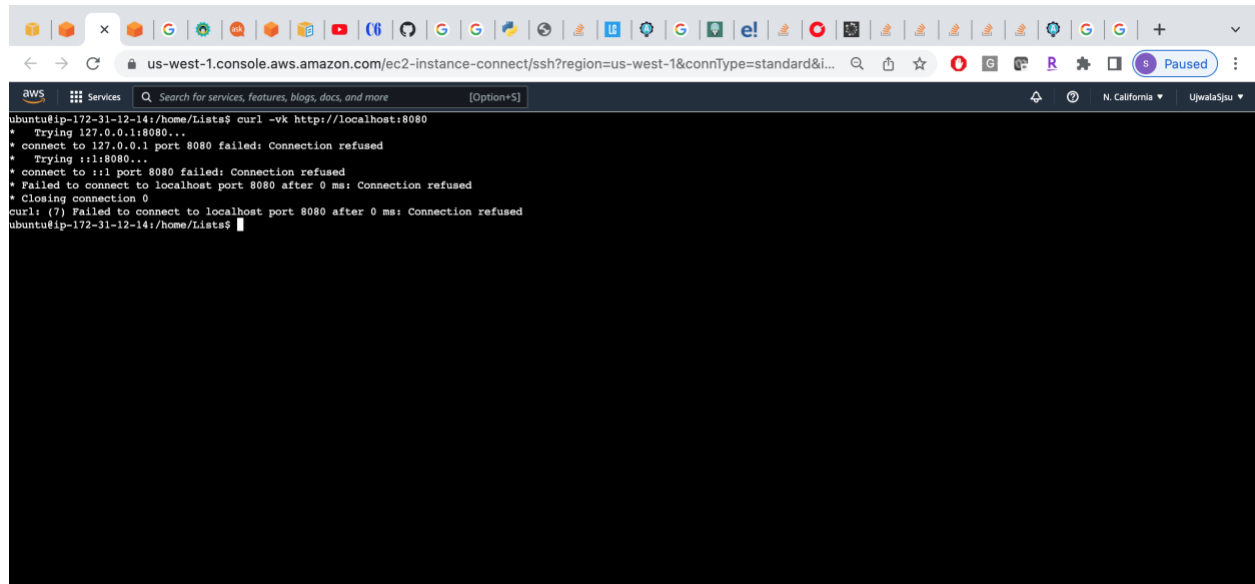


i-067f162123b3d2a44 (Ansible_Master)

PublicIPs: 13.52.81.244 PrivateIPs: 172.31.1.52



screenshots from client machines showing connection refused after undeployment playbook execution



The screenshot shows a terminal window within the AWS Management Console. The terminal output is as follows:

```
ubuntu@ip-172-31-12-14:/home/Lists$ curl -vk http://localhost:8080
* Trying 127.0.0.1:8080...
* connect to 127.0.0.1 port 8080 failed: Connection refused
* Trying ::1:8080...
* connect to ::1 port 8080 failed: Connection refused
* Failed to connect to localhost port 8080 after 0 ms: Connection refused
* Closing connection 0
curl: (7) Failed to connect to localhost port 8080 after 0 ms: Connection refused
ubuntu@ip-172-31-12-14:/home/Lists$
```

Below the terminal window, the instance details are shown:

i-0cf9369d26332c3d (AnsibleClient1)
PublicIPs: 18.144.170.33 PrivateIPs: 172.31.12.14

The bottom of the screenshot shows the AWS console footer with a feedback link, copyright information (© 2022, Amazon Web Services, Inc. or its affiliates), and a cookie preferences link. A notification bar at the bottom right says "Show All" with a close button.

