CMPE 272 Ansible Assignment Documentation

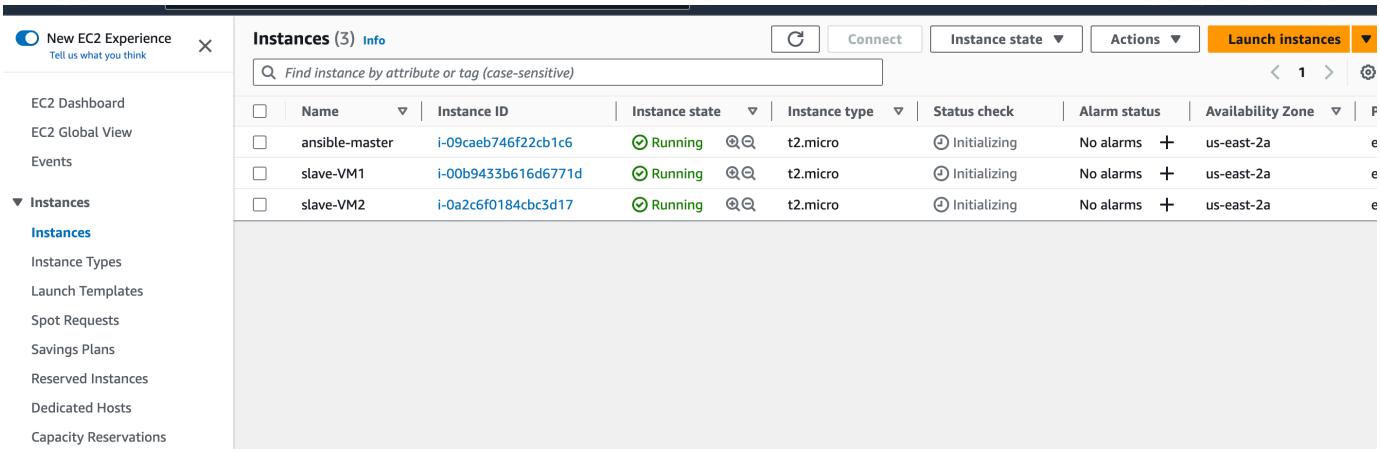
Github: https://github.com/shiva-vardhineedi/CMPE_272_Ansible_CodeHeist

Cloud Provider Used: Amazon Web Services

Team Members:

1. **Siva Swaroop Vardhineedi**
2. **Sri Charan Reddy Mallu**
3. **Sai Kiran Kammari**
4. **Mahek Virani**

1. Create three AWS EC2 instances one is master and other two are slaves.



2.Install Ansible in master VM.

```
[ec2-user@ip-172-31-8-101 ~]$ sudo su -
[root@ip-172-31-8-101 ~]# amazon-linux-extras install ansible2
Topic ansible2 has end-of-support date of 2023-09-30
Installing ansible
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Cleaning repos: amzn2-core amzn2extra-ansible2 amzn2extra-docker amzn2extra-kernel-5.10
17 metadata files removed
```

```
[root@ip-172-31-8-101 ~]# ansible --version
ansible 2.9.23
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/root/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/site-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.18 (default, Feb 28 2023, 02:51:06) [GCC 7.3.1 20180712 (Red Hat 7.3.1-15)]
```

3.Generate SSH key for the master instance

```
[root@ip-172-31-8-101 ~]# cd .ssh
[root@ip-172-31-8-101 .ssh]# ls -la
total 4
drwx------ 2 root root  29 Sep 10 21:15 .
dr-xr-x--- 4 root root 119 Sep 10 21:19 ..
-rw------- 1 root root 548 Sep 10 21:15 authorized_keys
[root@ip-172-31-8-101 .ssh]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:Nww+Xh/iM9Ad5jtqV22ehpQdqZTPXAsly7rvYE/3Ytg root@ip-172-31-8-101.us-east-2.compute.internal
The key's randomart image is:
+---[RSA 2048]----+
|                 |
|             . . |
|        .   + = .|
|       . + + B o.|
|        S B * O.+|
|       . * = *.*o|
|        . +o*=.+.|
|          o=*oEoo|
|         ...o=...|
+----[SHA256]-----+
```

4. **Create a Folder and add hosts file in the /etc directory in the master instance with the below content.**

- **Slave1—> 18.117.172.203**
- **Slave2—> 18.222.109.3**

```
# This is the default ansible 'hosts' file.
#
# It should live in /etc/ansible/hosts
#
#   - Comments begin with the '#' character
#   - Blank lines are ignored
#   - Groups of hosts are delimited by [header] elements
#   - You can enter hostnames or ip addresses
#   - A hostname/ip can be a member of multiple groups

# Ex 1: Ungrouped hosts, specify before any group headers.

## green.example.com
## blue.example.com
## 192.168.100.1
## 192.168.100.10

# Ex 2: A collection of hosts belonging to the 'webservers' group
[VM1]
18.117.172.203
[VM2]
18.222.109.3
## [webservers]
## alpha.example.org
```

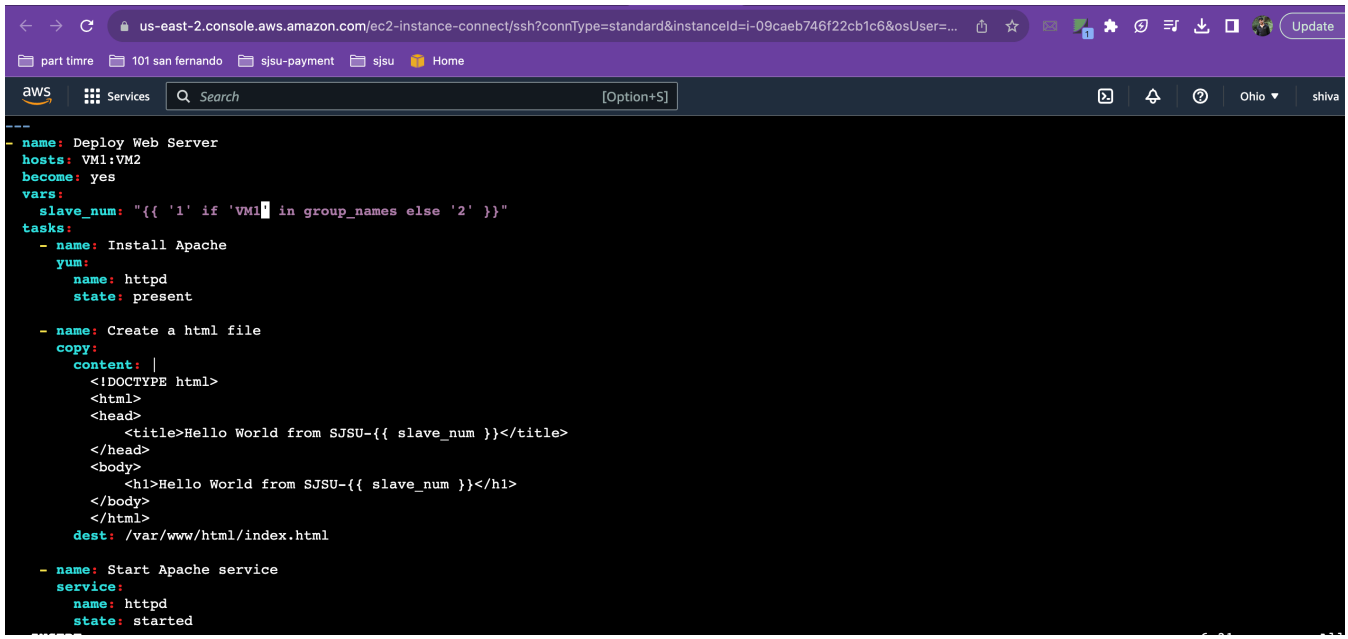
5. **Add the generated ssh key of master in both the slave instances in /.ssh/authorized_keys file**

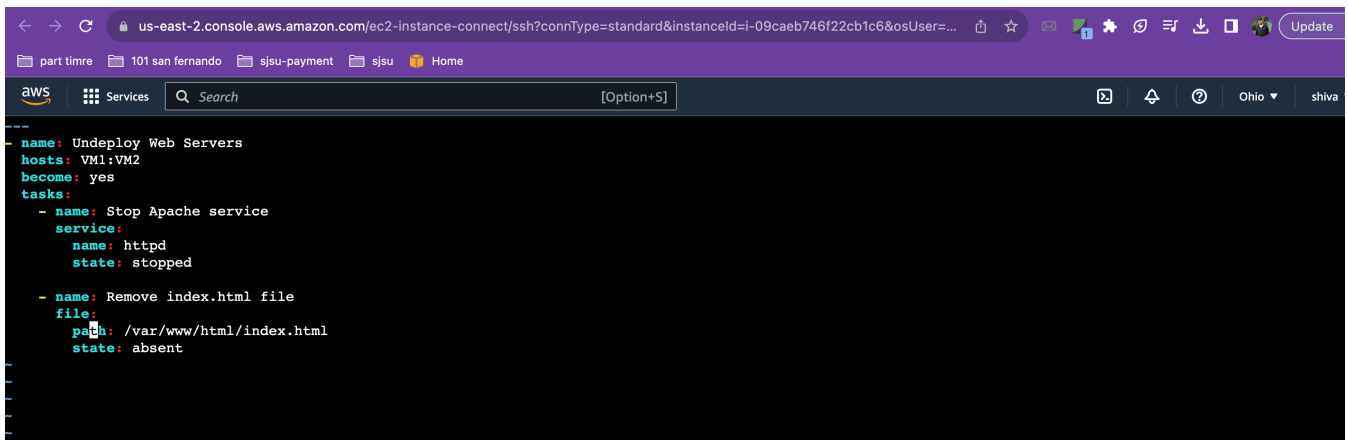
```
|         ...o=...|
+----[SHA256]-----+
[root@ip-172-31-8-101 .ssh]# ls -la
total 12
drwx------ 2 root root   61 Sep 10 21:21 .
dr-xr-x--- 4 root root  119 Sep 10 21:19 ..
-rw------- 1 root root  548 Sep 10 21:15 authorized_keys
-rw------- 1 root root 1675 Sep 10 21:21 id_rsa
-rw-r--r-- 1 root root  429 Sep 10 21:21 id_rsa.pub
[root@ip-172-31-8-101 .ssh]# cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQC/SsPlvUZSsaNLY9pSKF8kLfxjj6soVH1kZyG0jtpM+RS7Sdgv01q9ii0OPk9k3lIFlF3KGCeAFUVqDD+hDKjpmGS2dICG1j69kJd
HQvUsNvPgWP/RimX2cc4FT/7NfY/Lvd+VWkqjcomvVXhZr7JWO93ADsnMYZRlUpOH62V4dw+CtHL4MyblTSbZbk3ryy8MXPkZXMGyo1+FOfF/k8fCw1BPh126GyP7Dh1hK2220WQ/Cn
Dojy/Z2uaBSSEMmkuqd0XbIPcrBwiTmazowDPlPiZAB7/Sx4XvOrYeH4Kj5gKxgR root@ip-172-31-8-101.us-east-2.compute.internal
[root@ip-172-31-8-101 .ssh]# ssh 18.117.172.203
The authenticity of host '18.117.172.203 (18.117.172.203)' can't be established.
ECDSA key fingerprint is SHA256:9QYy0qtafVX9CwJWir3E/7IQ1WWGnZ6d9A24CNwaNxI.
ECDSA key fingerprint is MD5:0c:8a:de:48:90:8b:a5:37:13:ed:d3:77:07:fc:ed:8a.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '18.117.172.203' (ECDSA) to the list of known hosts.
Last login: Sun Sep 10 21:23:05 2023
```

6. **Ping the slave vms from master and check whether we are able to ping with success message as below.**

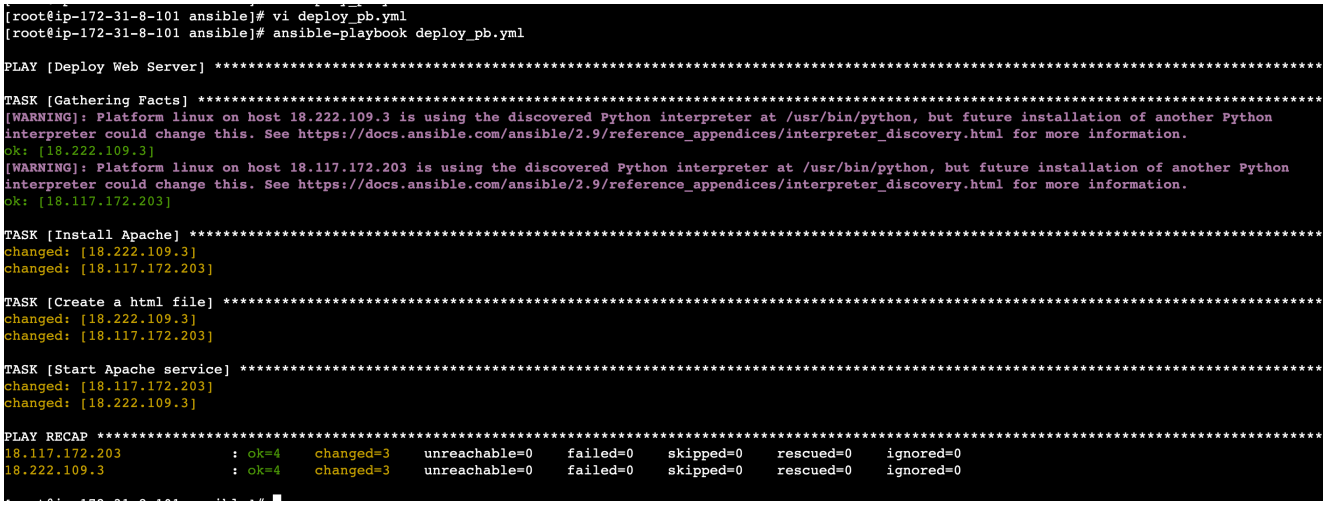
```
5 package(s) needed for security, out of 22 available
Run "sudo yum update" to apply all updates.
[root@ip-172-31-6-115 ~]# exit
logout
Connection to 18.222.109.3 closed.
[root@ip-172-31-8-101 ansible]# ansible -m ping all
[WARNING]: Platform linux on host 18.117.172.203 is using the discovered Python 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.
18.117.172.203 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
[WARNING]: Platform linux on host 18.222.109.3 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.
18.222.109.3 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
```

7. Create a directory and add the files for deploying server1 in slave one and deploying server 2 in slave2, default file for server validation, undeploy yml file to stop the deployed servers and html files for showing message from server1 and server2 from slave instances.



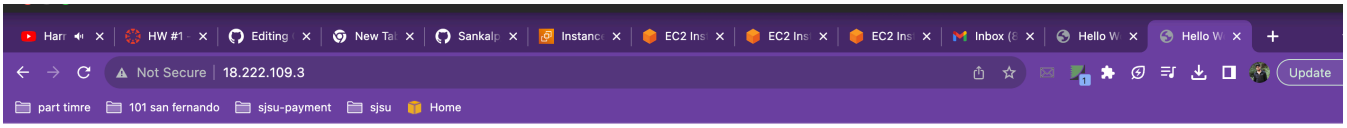


8. Run the deploy playbook yaml files to deploy servers in slave1, slave2 from master instance.



9. Open the servers of slave1 and slave2 in browsers to check the messages from slave1 and slave2

Hello World from SJSU-1



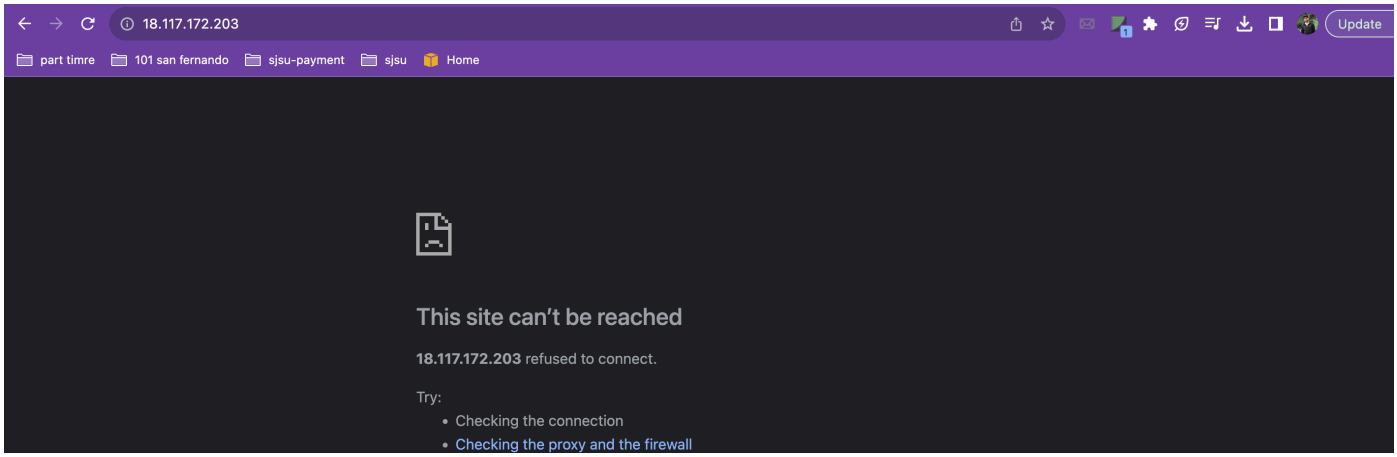
Hello World from SJSU-2

10.Run the Undeploy playbook yaml file on master instance to undeploy the deployed servers in slave1 and slave2.

```
[root@ip-172-31-8-101 ansible]# vi undeploy_pb.yml
[root@ip-172-31-8-101 ansible]# ansible-playbook undeploy_pb.yml

PLAY [Undeploy Web Servers] ****************************************************************************************************************************************

TASK [Gathering Facts] *********************************************************************************************************************************************
[WARNING]: Platform linux on host 18.222.109.3 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: [18.222.109.3]
[WARNING]: Platform linux on host 18.117.172.203 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: [18.117.172.203]

TASK [Stop Apache service] *****************************************************************************************************************************************
changed: [18.222.109.3]
changed: [18.117.172.203]

TASK [Remove index.html file] **************************************************************************************************************************************
changed: [18.222.109.3]
changed: [18.117.172.203]

PLAY RECAP *********************************************************************************************************************************************************
18.117.172.203             : ok=3    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
18.222.109.3               : ok=3    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

11.Below are the Images of accessing the servers of slave1 and slave2 from the browser after un-deploying them.

Slave1 server from browser:



Slave2 server from browser: