

Module 5: Ansible Assignment – 5

1. Create a new deployment of Ansible cluster of 5 nodes
2. Label 2 nodes as test and other 2 as prod
3. Install Java on test nodes
4. Install MySQL server on prod nodes Use Ansible roles for the above and group the hosts under test and prod.

5 instances naming ansible 1 to 5 are created in aws with ubuntu machine image

Instance Name	AMI ID	State	Instance Type	Progress	Alarms
Ansible2	i-07ac3067599861097	Running	t2.micro	Initializing	No alarms
Ansible3	i-0f338d72209a53f8c	Running	t2.micro	Initializing	No alarms
Ansible4	i-0d9421e4712b96829	Running	t2.micro	Initializing	No alarms
Ansible5	i-07b0f2745fcd0971	Running	t2.micro	Initializing	No alarms

- Navigated into ansible1 instance and set hostname as ansible master. Ssh key is generated and copied for further use

```
ubuntu@ip-172-31-30-200:~$ 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:lCy9yqTrO/4V4EUpolEg0QoEL9fyfFxX6eL5uEQhNlc ubuntu@ip-172-31-30-200
The key's randomart image is:
+---[RSA 3072]-----+
|*..o.  ..  ..|
|.oo.. .+.  ..|
|o.+o...o.E +.|
|.o.+ o B B...|
|  o = S..o    |
|  = . oo      |
|  . o . .o    |
|  .. . . . .  |
|  o=+.  ..    |
+---[SHA256]-----+
```

i-00fe8d736ab70cd38 (Ansible1)

PublicIPs: 54.221.158.23 PrivateIPs: 172.31.30.200

```
ubuntu@ip-172-31-30-200:~$ sudo cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCA2p7a4hwYd1YMKqclYzZwa7oYJ2jephsp6/Mek0Cwh7Euj+ZP30c3OQ1bCzAbt0E+nGQylEKVH1BwORS1xH8fbFUhHdVxKTD2yq
hB2mNHeFgIX15XVqk1SrX0t+R1HHwdnKnpGbGVmkwGajkWkvThPhVXHOSj3YU4it4PR3LZRzc1f/W+FYMS3dlHDyvdO361OLKy4LI91hNOR3mHv/Boc6vTfp9p22SZYVepEclJaOt0rz
eqCciBLqb0f2FGTK7UmeJsoDo7R0Mmp6uRYJsN9gQQ45hVslIR5dPz0cR1k1kDPSEhV+wp140TAUhP6801JXFYSlcA/zZf43DLpGgRRvOkirruMp0cuwoFIvIvuVibS5j7OTWAFynypjP
W5uMyPcchoUW0XxdzK30iHLziqCTRHszlXgy+7S/ppOXaK5mv+cR2L2JOYQZdkORci2IrP/dBD2tClWfAGzGBXEXzXGHyExwJctXK/WD15Rd62Kmm+Vooa0uAZuxmM2U= ubuntu@ip-1
72-31-30-200
ubuntu@ip-172-31-30-200:~$
```

- Navigated into ansible2 instance and set hostname as test1

```
ubuntu@ip-172-31-16-89:~$ sudo hostnamectl set-hostname test1
ubuntu@ip-172-31-16-89:~$ sudo nano ~/.ssh/authorized_keys
ubuntu@ip-172-31-16-89:~$ sudo nano ~/.ssh/authorized_keys
ubuntu@ip-172-31-16-89:~$
```

i-07ac3067599861097 (Ansible2)

PublicIPs: 18.207.117.213 PrivateIPs: 172.31.16.89

- In test1 navigated to ~/.ssh/authorized_keys and pasted the ssh key generated in instace 1.

```
GNU nano 6.2 ~/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCA2p7a4hwYd1YMKqclYzZwa7oYJ2jephsp6/Mek0Cwh7Euj+ZP30c3OQ1bCzAbt0E+nGQylEKVH1BwORS1xH8fbFUhHdVxKTD2yq
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCA2p7a4hwYd1YMKqclYzZwa7oYJ2jephsp6/Mek0Cwh7Euj+ZP30c3OQ1bCzAbt0E+nGQylEKVH1BwORS1xH8fbFUhHdVxKTD2yq
^G Help      ^O Write Out  ^W Where Is   ^R Cut         ^T Execute    ^C Location   ^U Undo       ^M Set Mark
^X Exit      ^R Read File  ^\ Replace    ^V Paste       ^J Justify    ^Y Go To Line ^B Redo       ^_ Copy
[ Read 2 lines ]
i-07ac3067599861097 (Ansible2)
PublicIPs: 18.207.117.213 PrivateIPs: 172.31.16.89
```

- Same procedure followed for ansible3-test2, ansible4-prod1 and ansible5-prod2
- Sshkey generated in ansible master is copied in ~/.ssh/authorized_keys in test1&2 and prod1 &prod2
 - A shellscript file is created with ansible installation commands in ansible master
 - Shellscript file is executed to install ansible in ansible master.

```
sudo apt update
sudo apt install software-properties-common
sudo apt-add-repository --yes --update ppa:ansible/ansible
sudo apt install ansible
~
~
~
~
~
```

- Then navigated into etc/ansible/hosts.
- Ip addresses of test1&2 and prod1&2 are added into groups naming test and prod

```
GNU nano 6.2 /etc/ansible/hosts *
# Ex4: Multiple hosts arranged into groups such as 'Debian' and 'openSUSE':

## [Debian]
## alpha.example.org
## beta.example.org

## [openSUSE]
## green.example.com
## blue.example.com

[test]
test1 ansible_host=172.31.16.89
test2 ansible_host=172.31.25.122
[prod]
prod1 ansible_host=172.31.26.63
prod2 ansible_host=172.31.17.245
]
```

i-00fe8d736ab70cd38 (Ansible1)

- To check the ansible cluster ansible -m ping all command is executed

```
ubuntu@ansible-master:~$ ansible -m ping all
prod1 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
test2 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
test1 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
prod2 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
```

Ansible cluster is successfully created with 5 nodes

- In ansible master a playbook javainstall.yml is created with script to install java on test servers

```
ubuntu@ansible-master:~$ cat javainstall.yml
---
- hosts: test
  become: yes
  tasks:
    - name: Update APT cache
      ansible.builtin.apt:
        update_cache: yes

    - name: Install Java OpenJDK
      ansible.builtin.package:
        name: openjdk-11-jdk # Modify this to the desired Java package
        state: present
```

- Javainstall.yml playbook is executed to successfully install java on test servers

```
ubuntu@ansible-master:~$ vi javainstall.yml
ubuntu@ansible-master:~$ ansible-playbook javainstall.yml

PLAY [test] *****

TASK [Gathering Facts] *****
ok: [test1]
ok: [test2]

TASK [Update APT cache] *****
changed: [test1]
changed: [test2]

TASK [Install Java OpenJDK] *****
changed: [test2]
changed: [test1]

PLAY RECAP *****
test1      : ok=3    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
test2      : ok=3    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

- Similarly sqlinstall.yml playbook is created in ansible master to install sql in prod servers

```
ubuntu@ansible-master:~$ cat sqlinstall.yml
---
- hosts: prod
  become: yes
  tasks:
    - name: mysql installation
      apt:
        name: mysql-server
        state: present

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

- Sqlinstall.yml playbook is executed. Successfully sql is installed in prod servers

```
ubuntu@ansible-master:~$ vi sqlinstall.yml
ubuntu@ansible-master:~$ ansible-playbook sqlinstall.yml --syntax-check

playbook: sqlinstall.yml
ubuntu@ansible-master:~$ ansible-playbook sqlinstall.yml

PLAY [prod] *****

TASK [Gathering Facts] *****
ok: [prod2]
ok: [prod1]

TASK [mysql installation] *****
ok: [prod1]
ok: [prod2]

TASK [start mysql service] *****
ok: [prod2]
ok: [prod1]

PLAY RECAP *****
prod1      : ok=3    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
prod2      : ok=3    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```