Lab for Ansible

1) Create 3 Ec2 machine name them as(amazon linux 2) : Ansibleserver

Node1 and node2

2) Go to software Mobaxterm and there login into all the three machines

3) **In Ansible machine/ansible server only :**

sudo su

yum install <https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm>

or

rpm -Uvh https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm

yum update -y

yum install git python python-level python-pip openssl ansible -y

ansible –version

OR

sudo amazon-linux-extras install ansible2

ansible –version

vi /etc/ansible/hosts

press i

## Add the below content in the hosts file…make sure you copy paste the correct ip

[demo]

<<private ip of the machine1>>

<<private ip of the machine2>>

press esc :wq

vi /etc/ansible/ansible.cfg

press i

Remove # from inventory (line 14)

Remove # from sudo-user (line 22)

esc and :wq

4) >>>>>>>do this step in all the nodes and ansible machine <<<<<<<

adduser ansible

# create a user with the name ansible

passwd ansible

#setting up the password for the user ansible

(now set the password)

## Now we need to provide the authentication rights to the ansible user. So that the ansible user can perform sudo commands …

visudo

press i

scroll down and find #All root to run any command ....

now there add

ansible ALL=(ALL) NOPASSWD: ALL (Add this below root)

(line number: 101)

5) **Do this step in ansible server only:**

su - ansible

*## su ansible means that you want to login as ansible user*

sudo yum install httpd -y

*# we are checking that are we able to install a software httpd via ansible user*

sudo yum remove httpd -y

exit

#*exit will exit from the ansible user*

6) **Do this step in ansible and node servers**

vi /etc/ssh/sshd\_config

press i

in authentication,

uncomment Permit root login yes

(remove #) (line number 38)

scroll down and uncomment password authentication yes(by removing #) (line61)

and comment password authentication no

(by putting #)(line 63)

7) **do this for nodes and ansible server**

service sshd restart

8)**In ansible server**:

##*Login as a ansible user*

su - ansible

#*# Trying to connect with node1 . you can copy paste the private ip of node1 from aws management console*

ssh <<private ip of node1>>

touch fileinnode1 file1innode1

exit

**Now go to node 1**

su - ansible

ls

check if fileinnode1 and file1innode1 is there

Again come back to ansible server

su - ansible

ssh <<private ip of node2>>

touch fileinnode2 file1innode2

exit

**Now go to node 2**

su - ansible

ls

check if fileinnode2 and file1innode2 is there

###########################REMOVE PASSWORD AUTHENTICATION USING TRUST RELATIONSHIP#########################

9) Go to ansible server (you need to ensure that you are logged in as ansible user)

ssh-keygen (press enter - enter - enter)

ls -a

cd .ssh/

ls

ssh-copy-id ansible@<<private ip of node1>>

password : last time it will ask for pass

ssh-copy-id ansible@<<private ip of node2>>

password : last time it will ask for pass

cd ..

ssh <<private ip of node1>>

exit

ssh <<private ip of node 2>>

exit

##########################################HOST PATTERN###########################################

10) Do this in ansible machine

ansible all --list-hosts

ansible demo --list-hosts

#show me the hosts within the demo group

################################################ADHOC COMMANDS ########################################

11) Go to ansible server

ansible demo -a "ls"

ansible all -a "touch fileinallnodes"

ansible demo -a "sudo yum install httpd -y"

ansible all -m ping

ansible all -m copy -a “src=abc.txt dest=/tmp”

ansible all -m file -a “path=/tmp/abc.txt state=touch”

ansible all -m package -a “name=nginx state=present” -b -----------download nginx with taking sudo permission with -b

ansible-inventory --list -y

ansible all --list-host

above both command used to check the hosts

############ Ansible module #############

-m ping -----Trivial test program

-m file -----file and directories

-m copy -----copied

-m setup -----To gather information about the remote server(hardware/software/config)

-m user -----To manage user

-m package -----To package installation ---nginx

-m service ----Services

-m git -----for Git

$ ansible-doc package -------get help for package module

$ ansible all/Demo/dbserver/webserver -m package -a "name=nginx state=present/latest" -b(--become sudo)

Playbook

---

- hosts: 172.31.91.55

become: true

tasks:

- name: Install the Apache Package

package: name={{ item }} state=latest

loop: ['apache2', 'vim', 'screen']

tags: ['setup']

- name: Copy Index Page

copy: src=index.html dest=/var/www/html owner=root group=root mode='0664'

tags: ['update', 'sync']

- name: Start the Apache Services

service: name=apache2 state=started enabled=yes

tags: ['service']

- name: Changing port number

lineinfile:

path: /etc/apache2/ports.conf

regexp: '^Listen'

insertafter: '^Listen'

line: Listen 9090

notify: Restart httpd

tags: ['update', 'sync']

handlers:

- name: Restart httpd

service: name=httpd state=restarted

…

####### FAILED! => {"msg": "Missing sudo password"} ###

To resolve this issue. Go to remote machine

$ sudo su

$ vim /etc/sudoers.d/ansible -----here ansible is username

Add the below code

ansible ALL=(ALL) NOPASSWD: ALL

########Command to check port ####

sudo netstat -antup | grep 9090

##### Ansible Roles #####

-------Decluttering a playbook or breaking the playbook in more organized way

---sharaibility

---organisability

---reusuability

$ ansible-galaxy init httpd

---- it will create a directory called httpd and structure of roles

Write the code in the httpd roles

Like in tasks, handlers, meta, vars etc as per required.

Create a new yaml file ex. site.yaml

---

- hosts: Demo

become: true

roles:

- httpd

$ ansible-playbook site.yaml