

to install ansible on Amazon Linux or to setup ansible lab in aws we need two or three ec2 instances. one is ansible master ec2 instance remaining ec2 instances are clients. in the master ec2 instance only we will install ansible.

### Launch three or two ubuntu 16.04 instances

give Name one ubuntu ec2 instances as ansible-master

give remaining ec2 instances names as client1, client2

in both ansible master and clients security groups

open ssh port no 22 from anywhere

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Anywhere	0.0.0.0/0, ::/0 e.g. SSH for Admin Desktop

**Add Rule**

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

**Cancel Save**

ansible built on python so install python in all machines

## install python in ansible master and clients instances

ansible and its modules are built on python, so we have to install python in all master and client machines.

to install python execute below commands as root user

```
sudo -i
```

```
apt-get install python-minimal
```

```
apt-get install python3
```

check python version with

```
python --version  
Python 2.7.12
```

## installing ansible in ansible master instance

run below commands as root user

```
sudo -i

apt-get update

apt-get install software-properties-common

apt-add-repository ppa:ansible/ansible

apt-get update

apt-get install ansible
```

check ansible with

```
ansible --version
ansible 2.6.3
  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/dist-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.12 (default, Dec  4 2017, 14:50:18) [GCC 5.4.0 20160609]
```

## establish ssh connection between ansible master and clients

to establish a connection between master and clients we have to generate the id\_rsa.pub key in master and paste this key in authorized\_keys file of client machines. This file exists in the .ssh directory. So if the .ssh directory has not existed in client ec2 instances, We have to create the .ssh directory and inside that, we have to create the authorized\_keys file.

## generating id\_rsa.pub public key in ansible master instance

in master, ec2 instance execute below commands

```
sudo -i

ssh-keygen -t rsa
```

```

root@ip-10-0-0-11: ~
root@ip-10-0-0-11:~# sudo -i
root@ip-10-0-0-11:~# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
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:RHS+KKID6t+mo8XVNV+6/qVTd0QYZLQSFX4QoC/9w4M root@ip-10-0-0-11
The key's randomart image is:
+---[RSA 2048]-----+
|      .o . o=X= |
|      . o . +.o. |
|      .oo ..o.. |
|      ....o+o. .. |
|.      ....S ooo . |
|..... .      ..+ .o|
|. oo      .E *.o|
|. .o..      . .oo |
| ooo+.      ..o. |
+-----[SHA256]-----+
root@ip-10-0-0-11:~#

```

It will create the id\_rsa.pub key in the .ssh directory

```
cd .ssh
```

```
ls
```

```
id_rsa id_rsa.pub known_hosts
```

```
cat id_rsa.pub
```

```

root@ip-10-0-0-11: ~/.ssh
root@ip-10-0-0-11:~# cd .ssh/
root@ip-10-0-0-11:~/.ssh# ls
id_rsa id_rsa.pub known_hosts
root@ip-10-0-0-11:~/.ssh# cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDQ9CbfNOF0GQI2T9C363Y7RWQkVH2QhBy4qLVNgyWUailzDquRi3
Pg4N5oFYKfLI3tle5fP+a5sBLCQrxnrSDkEogPLwUCxNgDDLkPHk19XaALWEUW1lsg2NhkbPaEwDkLM6kTthJQ78w7
N+CLx3UPy0IuuPcH06Idv53SD1joNUL//8tV6IUiti63 root@ip-10-0-0-11
root@ip-10-0-0-11:~/.ssh#

```

Copy this id\_rsa.pub key

## In All Client Ec2 Instances

```
Sudo -i
```

```
cd .ssh
```

```
ls
```

Here you can see the authorized\_keys file. [If the file is not existed here create the file with touch command

**touch authorized\_keys** and paste the id\_rsa.pub key in this file

Vi authorized\_keys

```
root@ip-10-0-0-14: ~/.ssh
root@ip-10-0-0-14:~# cd .ssh/
root@ip-10-0-0-14:~/.ssh# ls
authorized_keys
root@ip-10-0-0-14:~/.ssh# vi authorized_keys
```

Paste id\_rsa.pub key of the master here

```
root@ip-10-0-0-14: ~/.ssh
no-port-forwarding,no-agent-forwarding,no-X11-forwarding,command="echo 'Please login as th
3NzaC1yc2EAAAADAQABAAQACppHLyQsneRfLAQaUgCXc7D9BjbHk+HMaqL92UjCWloQWpE+0/ai5dBqHtIfQJ6nE
ICVf6C/fjRRcc163/ORU9+zyAqdkQzUw1JS8BudOW821PCuOWzt/2eaPggttvKkFiQsX5JCCRnDDTUJzCH7g8O3gMN
34Cm2usmZLgRQfx/nydKeYXWDCM10NB linux

ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQAD09CbfnOf0GQI2T9C363Y7RWQkVH2QhBy4qLVNgyWUai1zDquRiS
Pg4N5oFYKfLI3tle5fP+a5sBLCQrxnrSDkEogPLwUCxNgDDLkPHk19XaALWEUWllsg2NhkbPaEwDkLM6kTthJQ78w
N+CLx3UPy0IuuPcH06Idv53SD1joNUL//8tV6IUiti63 root@ip-10-0-0-11
```

now we have shared ssh keys between master and clients

## Adding clients to ansible master

to add clients to ansible master machine, we need to add all IP's of clients in master machine /etc/ansible/hosts file

now go to ansible master machine

## Ansible AWS Inventory

cd /etc/ansible

vi hosts

add like below

[web]

10.0.0.14

```
root@ip-10-0-0-11: ~  
This is the default ansible 'hosts' file.  
[web]  
10.0.0.14  
# 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.
```

here 10.0.0.14 is private IP of the client1 machine

here you can mention all client machines private IP's

```
root@ip-10-0-0-11: ~  
root@ip-10-0-0-11:~# ansible -m ping all  
The authenticity of host '10.0.0.14 (10.0.0.14)' can't be established.  
ECDSA key fingerprint is SHA256:d/T+7RLCZcTrVsB9oqFwzfyM4yiOIDvoBN+n3cn6uFA.  
Are you sure you want to continue connecting (yes/no)? yes  
10.0.0.14 | SUCCESS => {  
    "changed": false,  
    "ping": "pong"  
}  
root@ip-10-0-0-11:~#
```

the first time it will ask are you sure you want to continue connecting yes/no

write yes and click on enter

you can see the output in green color.

now we have successfully configured ansible practice lab in aws.

now you can run your playbooks and roles in clients.

## IMP POINTS

- allow ssh port forwarding between clients and master by opening port no 22
- install python in all master and client ec2 instances
- install ansible in master ec2 instance
- generate the public key in ansible master instance
- copy id\_rsa.pub key and paste in all clients instances authorized\_keys file
- enter all client IPs in master /etc/ansible/hosts file
- now you can run your playbooks