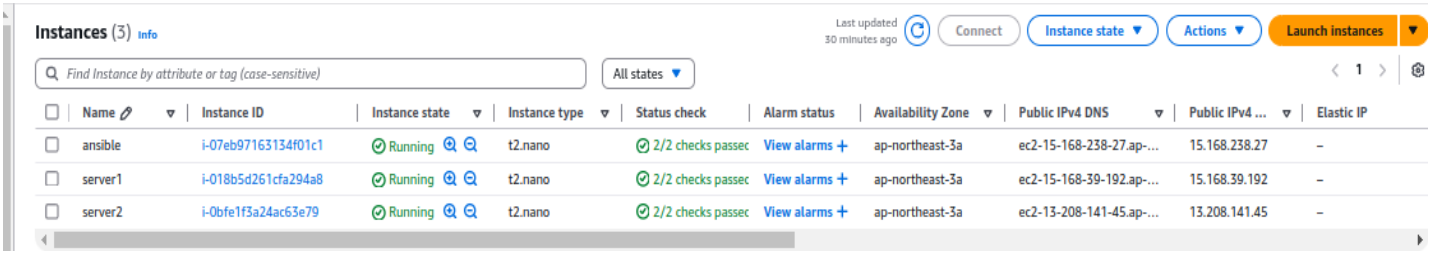
How to setup Ansible and SSH keys in AWS

1) Create 3 AWS ec2 instance in ubuntu



Instances (3) Info

Last updated 30 minutes ago | Connect | Instance state | Actions | Launch instances

Find Instance by attribute or tag (case-sensitive) | All states

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
ansible	i-07eb97163134f01c1	Running	t2.nano	2/2 checks passed	View alarms +	ap-northeast-3a	ec2-15-168-238-27.ap-...	15.168.238.27	–
server1	i-018b5d261cfa294a8	Running	t2.nano	2/2 checks passed	View alarms +	ap-northeast-3a	ec2-15-168-39-192.ap-...	15.168.39.192	–
server2	i-0bfe1f3a24ac63e79	Running	t2.nano	2/2 checks passed	View alarms +	ap-northeast-3a	ec2-13-208-141-45.ap-...	13.208.141.45	–

First instance - Ansible

Two instance - Server1 and server2

2) Login in Ansible EC2 instance and use these commands

→ switch as root

sudo su -

→ update packages

apt update -y

→ run the following command to include the official project's PPA (personal package archive) in your system's list of source

apt-add-repository ppa:ansible/ansible

→ Next, refresh your system's package index so that it is aware of the packages available in the newly included PPA:

apt update -y

→ Following this update, you can install the Ansible software with:

apt install ansible -y

→ Check ansible version

ansible --version

→ Go the hosts and add your server1 and server2

nano /etc/hosts

```
  GNU nano 6.2
127.0.0.1 localhost

15.168.39.192 server1
13.208.141.45 server2



# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
```

Add : **public IP server 1**

public IP server 2

→ Generate ssh key from ansible server

And

Press - Enter → Enter → Enter

```
root@ip-172-31-37-136:~# cd .ssh/
root@ip-172-31-37-136:~/.ssh# ls
authorized_keys  id_rsa  id_rsa.pub  known_hosts  known_hosts.old
root@ip-172-31-37-136:~/.ssh#
```

Your you can see ssh keys of public key and private key

→ Copy the public key (id_rsa.pub) and paste it in athuorized_key on server1 and server2

cat id_rsa.pub

```
root@ip-172-31-37-136:~/.ssh# cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQCcOXHg7gtY+ppKXeAkuHKW2rJj7PIG5ga5TzBVkPckciieH8FYYarLO4YO3qCfoXZzvknoKpU3Cs4lsr5Na15BYh/Pq96HJvkQ3O20rMsfxOMOOoCnNRgo7o4VXjmTK5pTD3vKTOkT/LdwsFCBw2u5Jp4zAxoqbyQg93Gu+5DIkBfCx
LERIKfsPBzSri7YThwDME80/mWhMXag/B5T5RvaUT07uu2Zh+m8IezaWJZnRv+srwrtaRjxgt9fEhAx7Ub2+rs01/ztFOCpoWCk+Vf4rCldTgfjsDhtkNNCQZMqIcPn6iBZvSDoZEM7z+MErBvdtaoJ35U8JmOu1TA5PeSQJ5tvVFiSg1lZ3NCkdEm6UBdm6en3PG5GNjtHTC6WRGX0Y
23/eqU8PJBeURQVCvsrXRUFE2Q9Gd+rtkzCUbl2++zBxW0mVLB+PDdfSCMjaBnAepQdvUL7pqkcq0etEaQYqUXMV5yj0r70q2iv0Qf8x3KzusiiX8aM5MVr6UQ4gHuM= root@ip-172-31-37-136
root@ip-172-31-37-136:~/.ssh#
```

→ Go to the server 1 and server 2

→ Login server1 and paste this public key in .ssh/athuorized_keys

nano .ssh/authorized_keys

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQCcOXHg7gtY+ppKXeAkuHKW2rJj7PIG5ga5TzBVKPckc1eH0FYYarLO4YO3qCfOXZzVknoKpU3Cs4lsr5Na15BYh/Pq96HJvkQ3O28rMsfxOMOOoCnNRgo7o4VXjmTK5pTD3vKtOkT/LdwsFcBw2u5Jp4zAxoqbyQg93Gu+5DlkBfCp
```

Save it and come out from the shell

→ Login server1 and paste this public key in .ssh/athuorized_keys

nano .ssh/authorized_keys


```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQCcOXHg7gtY+ppKXeAkuHKW2rJj7PIG5ga5TzBVKPckc1eH0FYYarLO4YO3qCfOXZzVknoKpU3Cs4lsr5Na15BYh/Pq96HJvkQ3O28rMsfxOMOOoCnNRgo7o4VXjmTK5pTD3vKtOkT/LdwsFcBw2u5Jp4zAxoqbyQg93Gu+5DlkBfCp
```


Save it and come out from the shell

→ Return to the Ansible server and check if the ping is working on server1 and server2.

ping server1

```
root@ip-172-31-37-136:~# ping server1
PING server1 (15.168.39.192) 56(84) bytes of data.
64 bytes from server1 (15.168.39.192): icmp_seq=1 ttl=63 time=1.01 ms
64 bytes from server1 (15.168.39.192): icmp_seq=2 ttl=63 time=0.769 ms
64 bytes from server1 (15.168.39.192): icmp_seq=3 ttl=63 time=0.873 ms
64 bytes from server1 (15.168.39.192): icmp_seq=4 ttl=63 time=1.39 ms
^C
--- server1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3032ms
rtt min/avg/max/mdev = 0.769/1.011/1.391/0.235 ms
root@ip-172-31-37-136:~#
```

ping server2

```
root@ip-172-31-37-136:~# ping server2
PING server2 (13.208.141.45) 56(84) bytes of data.
64 bytes from server2 (13.208.141.45): icmp_seq=1 ttl=63 time=0.570 ms
64 bytes from server2 (13.208.141.45): icmp_seq=2 ttl=63 time=1.91 ms
64 bytes from server2 (13.208.141.45): icmp_seq=3 ttl=63 time=1.05 ms
^C
--- server2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2040ms
rtt min/avg/max/mdev = 0.570/1.175/1.911/0.555 ms
root@ip-172-31-37-136:~#
```

It is working fine in Ansible server

→ Create a directory in the name of Ansbile

mkdir ansible

```
root@ip-172-31-37-136:~# ls
ansible  snap
root@ip-172-31-37-136:~#
```

→ Get in the Ansible directory

cd ansible

→ Create a inventory file and add these hosts

nano inventory

[webservers]

server1

server2 (save it and come out from the shell)

cat inventory

→ Create ansible.cfg file and these lines

nano ansible.cfg

[defaults]

inventory=/root/ansible/inventory

remote_user=ubuntu

ask_pass=false (save it and come out from the shell)

→ For testing purpose, we need to install nginx in server 1 and apache in server2 from ansible server

→ Create yaml file for install nginx and apache in server1 and server2

nano install_webservers.yml

```
---
- name: Install Web Servers
  hosts: webservers
  become: true
  tasks:
    - name: Install Nginx on server1
      apt:
        name: nginx
        state: present
      when: inventory_hostname == 'server1'

    - name: Install Apache on server2
      apt:
        name: apache2
        state: present
      when: inventory_hostname == 'server2'

    - name: Ensure Nginx is started and enabled on server1
      service:
        name: nginx
        state: started
        enabled: yes
      when: inventory_hostname == 'server1'

    - name: Ensure Apache is started and enabled on server2
      service:
        name: apache2
        state: started
        enabled: yes
      when: inventory_hostname == 'server2'
```

(save it and come out from the shell)

→run ansible yml file following this command

ansible-playbook -i /root/ansible/inventory install_webservers.yml

```
root@ip-172-31-37-136:~/ansible# ansible-playbook -i /root/ansible/inventory install_webservers.yml

PLAY [Install Web Servers] ************************************************************************************************************************************************************************************************************************************

TASK [Gathering Facts] ****************************************************************************************************************************************************************************************************************************************
[WARNING]: Platform linux on host server1 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [server1]
[WARNING]: Platform linux on host server2 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [server2]

TASK [Install Nginx on server1] *******************************************************************************************************************************************************************************************************************************
skipping: [server2]
changed: [server1]

TASK [Install Apache on server2] ******************************************************************************************************************************************************************************************************************************
skipping: [server1]
changed: [server2]

TASK [Ensure Nginx is started and enabled on server1] *********************************************************************************************************************************************************************************************************
skipping: [server2]
ok: [server1]

TASK [Ensure Apache is started and enabled on server2] ********************************************************************************************************************************************************************************************************
skipping: [server1]
ok: [server2]

PLAY RECAP ****************************************************************************************************************************************************************************************************************************************************
server1                    : ok=3    changed=1    unreachable=0    failed=0    skipped=2    rescued=0    ignored=0
server2                    : ok=3    changed=1    unreachable=0    failed=0    skipped=2    rescued=0    ignored=0
```

Here you can see installing nginx and apache each servers and you can test by copy each servers ip and paste it browser.