Date : Jan 4 , 2022
Ansible :

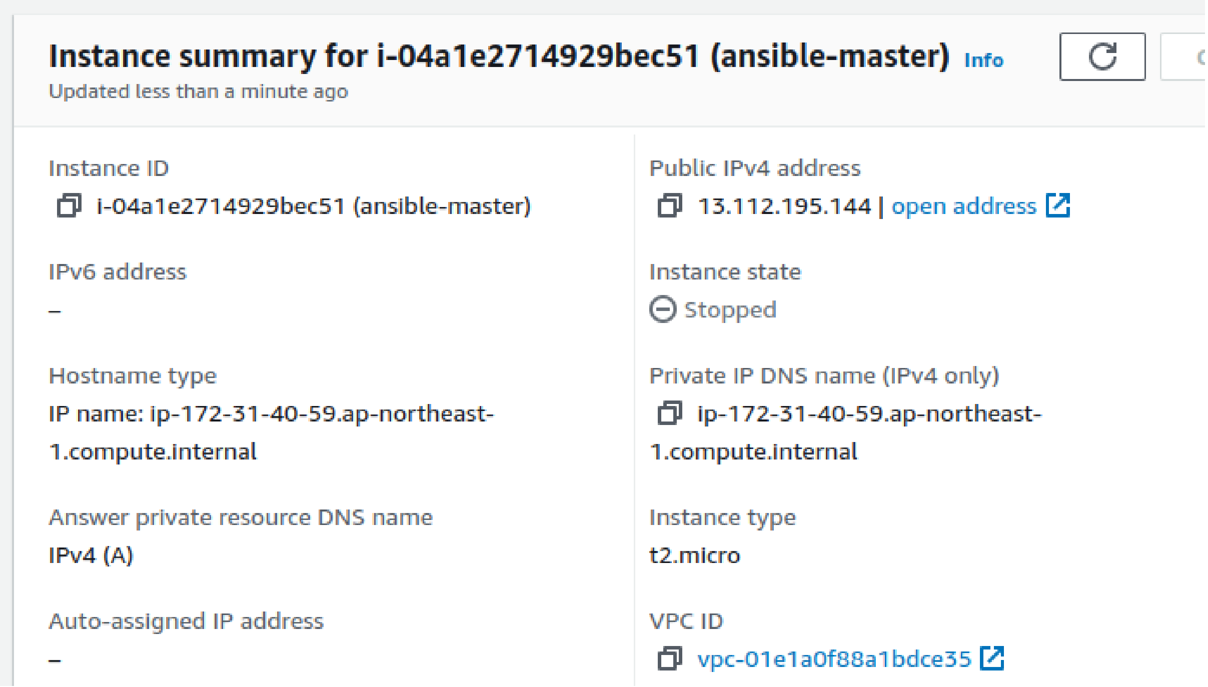
What is ansible : - configuration management tool , Uses YAML file , communicate through ssh
Main Functionality : using ansible we can make one master node and can control various servers .

Components :

- Ansible server : the machine where ansible is installed
- Module : commands or set of commands that to be executed on client side
- Inventory : File containing data about client servers
- Play : execution of playbook
- Ad hoc commands : can be performed individually to perform a quick function
- Inventory file location : /etc/ansible/hosts

Creating Master node ec2 and multiple servers and checking memory usage & uptimecheck using ansible :

Step 1 : on aws console launched an free tier instance named : ansible-master



Instance summary for i-04a1e2714929bec51 (ansible-master) Info
Updated less than a minute ago

Instance ID
i-04a1e2714929bec51 (ansible-master)

Public IPv4 address
13.112.195.144 | open address

IPv6 address
–

Instance state
Stopped

Hostname type
IP name: ip-172-31-40-59.ap-northeast-1.compute.internal

Private IP DNS name (IPv4 only)
ip-172-31-40-59.ap-northeast-1.compute.internal

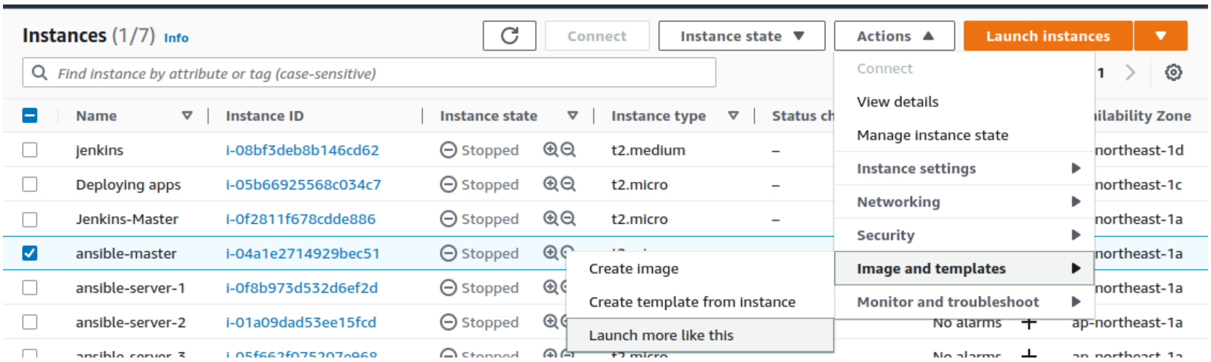
Answer private resource DNS name
IPv4 (A)

Instance type
t2.micro

Auto-assigned IP address
–

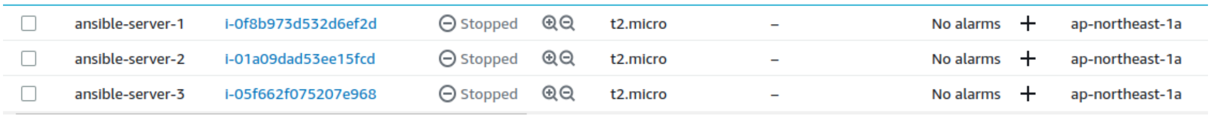
VPC ID
vpc-01e1a0f88a1bdce35

Step 2: now created multiple servers that are connected to ansible master
Using **master node > actions >image & templated> launch more instance like this**



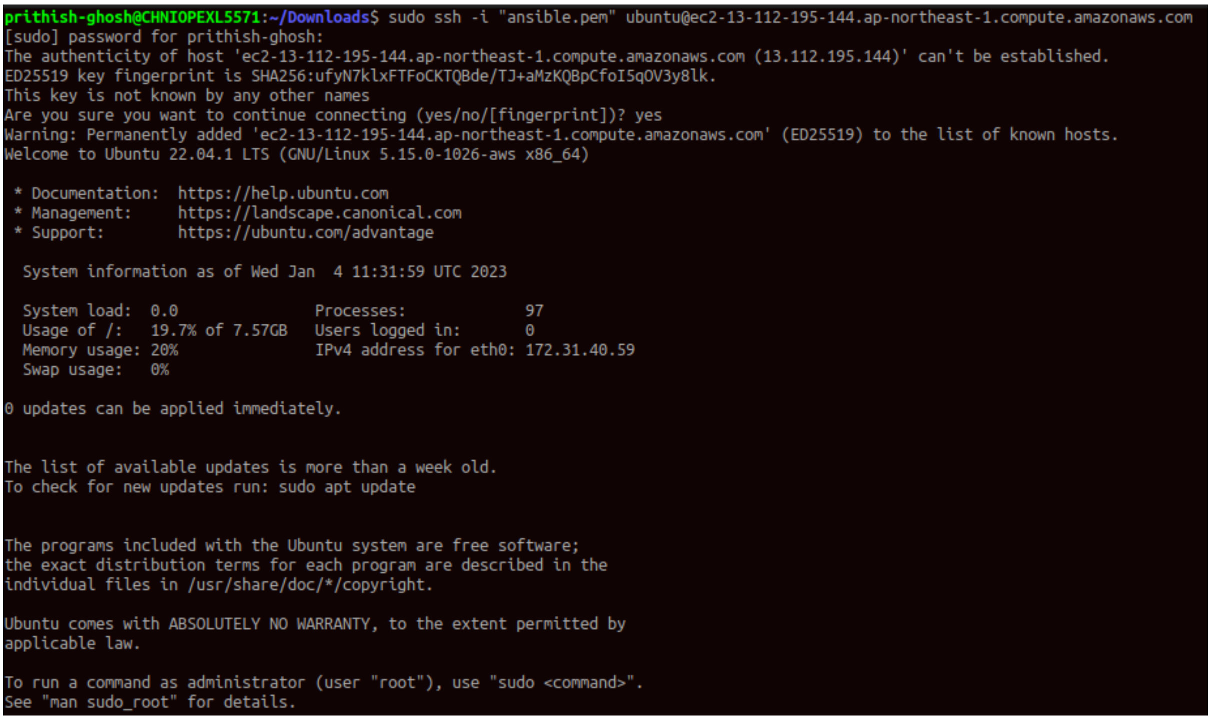
	Name	Instance ID	Instance state	Instance type	Status ch		ilability Zone
☐	jenkins	i-08bf3deb8b146cd62	⊖ Stopped	t2.medium	–		northeast-1d
☐	Deploying apps	i-05b66925568c034c7	⊖ Stopped	t2.micro	–		northeast-1c
☐	Jenkins-Master	i-0f2811f678cdde886	⊖ Stopped	t2.micro	–		northeast-1a
☑	ansible-master	i-04a1e2714929bec51	⊖ Stopped				northeast-1a
☐	ansible-server-1	i-0f8b973d532d6ef2d	⊖ Stopped				northeast-1a
☐	ansible-server-2	i-01a09dad53ee15fcd	⊖ Stopped			No alarms +	ap-northeast-1a
☐	ansible-server-3	i-05f662f075207e968	⊖ Stopped	t2.micro		No alarms +	ap-northeast-1a

After that need to choose how many servers we should create in my case i have chosen 3



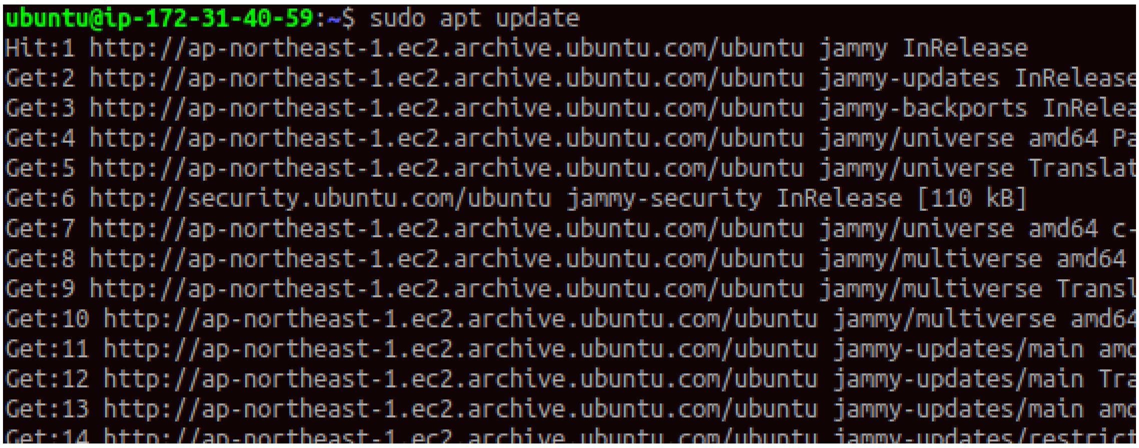
☐	ansible-server-1	i-0f8b973d532d6ef2d	⊖ Stopped	t2.micro	–	No alarms +	ap-northeast-1a
☐	ansible-server-2	i-01a09dad53ee15fcd	⊖ Stopped	t2.micro	–	No alarms +	ap-northeast-1a
☐	ansible-server-3	i-05f662f075207e968	⊖ Stopped	t2.micro	–	No alarms +	ap-northeast-1a

Step 3 : Now i have connected to master node which is ansible-master using ssh



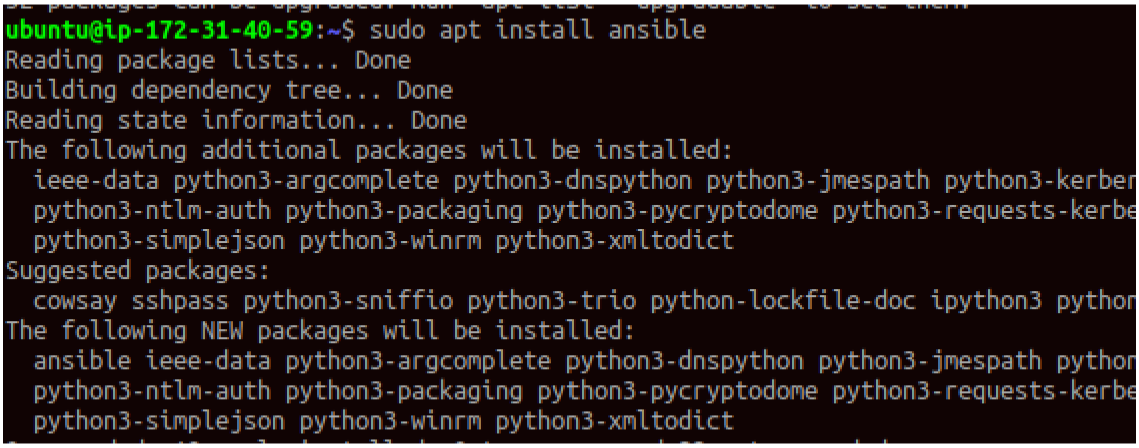
```
prithish-ghosh@CHNIOPEXL5571:~/Downloads$ sudo ssh -i "ansible.pem" ubuntu@ec2-13-112-195-144.ap-northeast-1.compute.amazonaws.com
[sudo] password for prithish-ghosh:
The authenticity of host 'ec2-13-112-195-144.ap-northeast-1.compute.amazonaws.com (13.112.195.144)' can't be established.
ED25519 key fingerprint is SHA256:ufyN7klxFTFoCKTQBde/TJ+aMzKQBpCfoI5qOV3y8lk.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-13-112-195-144.ap-northeast-1.compute.amazonaws.com' (ED25519) to the list of known hosts.
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-1026-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Wed Jan  4 11:31:59 UTC 2023

  System load:  0.0               Processes:             97
  Usage of /:   19.7% of 7.57GB   Users logged in:       0
  Memory usage: 20%               IPv4 address for eth0: 172.31.40.59
  Swap usage:   0%

0 updates can be applied immediately.


The list of available updates is more than a week old.
To check for new updates run: sudo apt update


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
```

Step 4 : updating my ubuntu system using **sudo apt update**



```
ubuntu@ip-172-31-40-59:~$ sudo apt update
Hit:1 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Get:3 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelea
Get:4 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Pa
Get:5 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translat
Get:6 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:7 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 c-
Get:8 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64
Get:9 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse Transl
Get:10 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64
Get:11 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amc
Get:12 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Tra
Get:13 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amc
Get:14 http://ap-northeast-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restrict
```

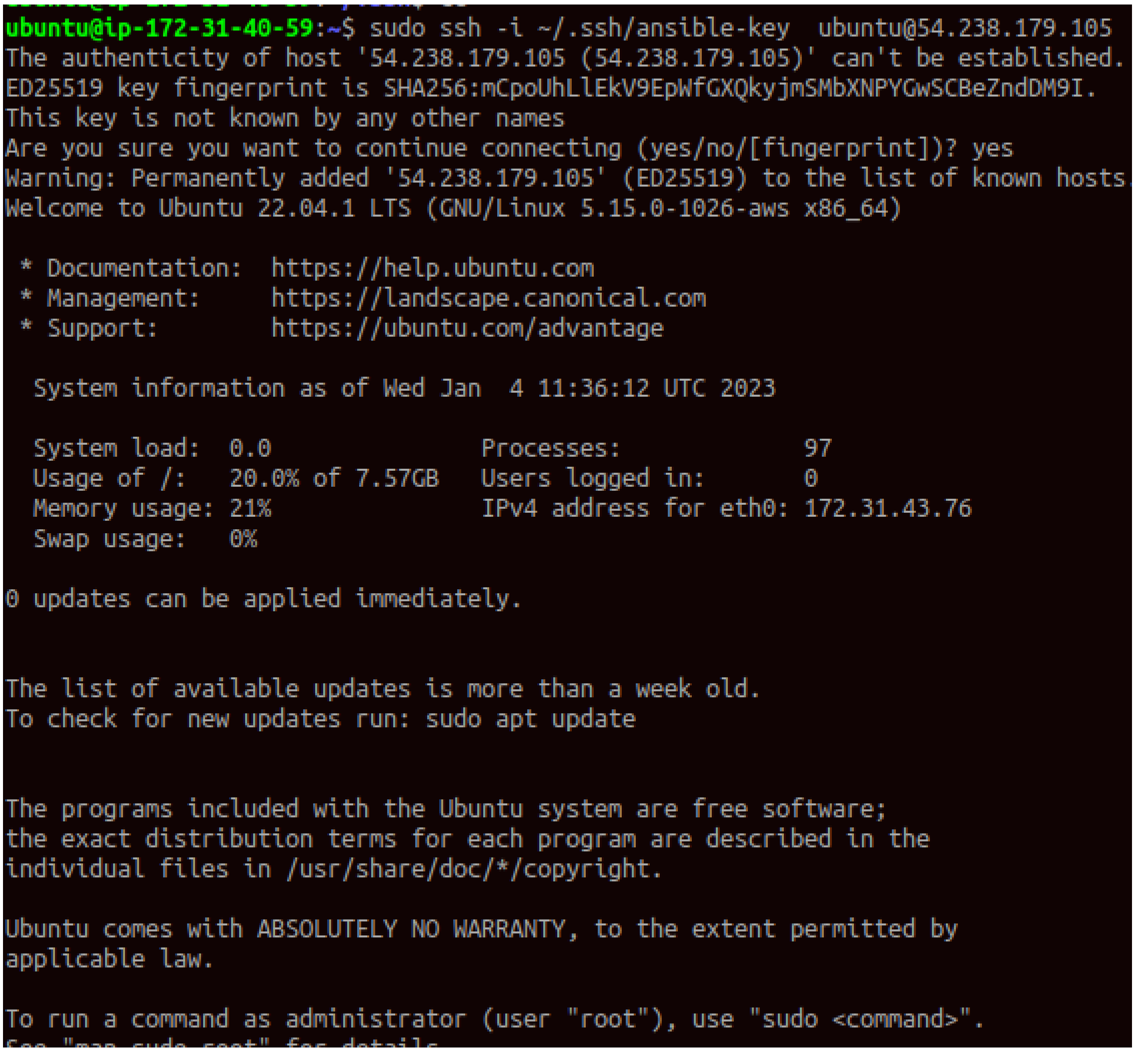
Step 5 : installing ansible in master node



```
ubuntu@ip-172-31-40-59:~$ sudo apt install ansible
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ieee-data python3-argcomplete python3-dnspython python3-jmespath python3-kerber
  python3-ntlm-auth python3-packaging python3-pycryptodome python3-requests-kerbe
  python3-simplejson python3-winrm python3-xmltodict
Suggested packages:
  cowsay sshpass python3-sniffio python3-trio python-lockfile-doc ipython3 python
The following NEW packages will be installed:
  ansible ieee-data python3-argcomplete python3-dnspython python3-jmespath python
  python3-ntlm-auth python3-packaging python3-pycryptodome python3-requests-kerbe
  python3-simplejson python3-winrm python3-xmltodict
```

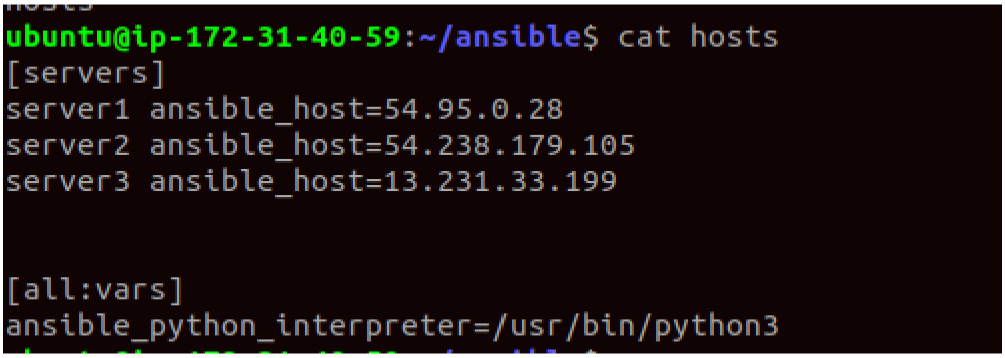
Step 6 :

In the ssh directory I made a ansible-key file where have defined my ssh key **(/home/ubuntu/.ssh)** with this key i will be able to connect any of the servers .



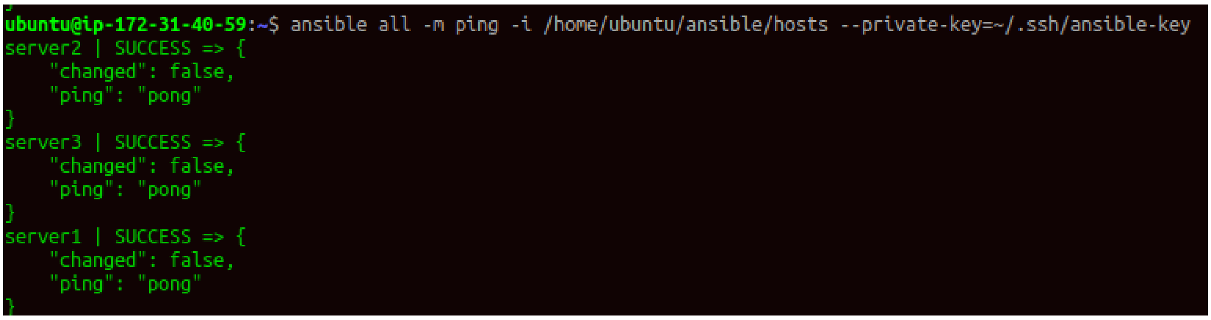
```
ubuntu@ip-172-31-40-59:~$ sudo ssh -i ~/.ssh/ansible-key  ubuntu@54.238.179.105
The authenticity of host '54.238.179.105 (54.238.179.105)' can't be established.
ED25519 key fingerprint is SHA256:mCpoUhLlEkV9EpWfGXQkyjmSMbXNPYGwSCBeZndDM9I.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '54.238.179.105' (ED25519) to the list of known hosts.
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-1026-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Wed Jan  4 11:36:12 UTC 2023

  System load:  0.0               Processes:             97
  Usage of /:   20.0% of 7.57GB   Users logged in:       0
  Memory usage: 21%               IPv4 address for eth0: 172.31.43.76
  Swap usage:   0%

0 updates can be applied immediately.


The list of available updates is more than a week old.
To check for new updates run: sudo apt update


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
```

Step 7: added servers information and made inventory with 3 servers and by default added python packages (added in all servers in variable)



```
ubuntu@ip-172-31-40-59:~/ansible$ cat hosts
[servers]
server1 ansible_host=54.95.0.28
server2 ansible_host=54.238.179.105
server3 ansible_host=13.231.33.199


[all:vars]
ansible_python_interpreter=/usr/bin/python3
```

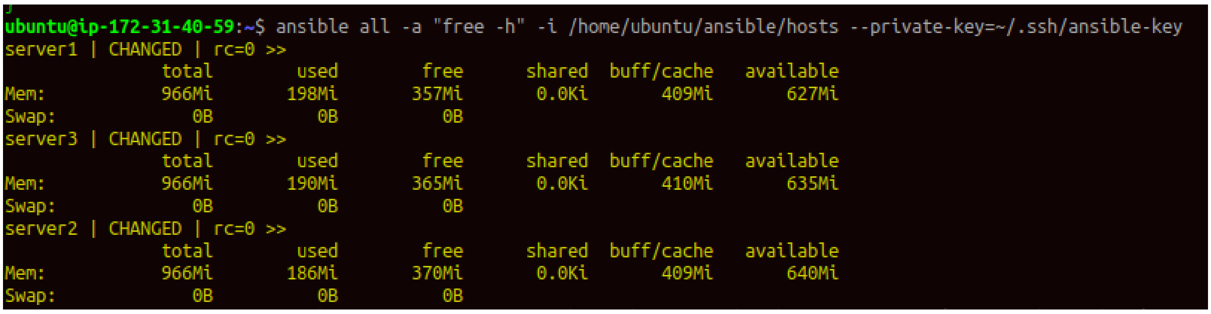
Step 8 : now for testing all 3 servers from master node am using this command :
ansible all -m ping -i /home/ubuntu/ansible/hosts --private-key=~/.ssh/ansible-key



```
ubuntu@ip-172-31-40-59:~$ ansible all -m ping -i /home/ubuntu/ansible/hosts --private-key=~/.ssh/ansible-key
server2 | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
server3 | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
server1 | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
```

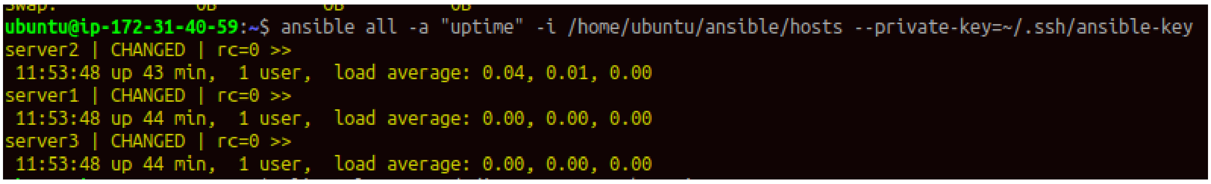
Output : coming success means all servers working fine from master node

Step 9 : checking usage and adding uptime check in 3 servers . for that these two command : **ansible all -a "free -h" -i /home/ubuntu/ansible/hosts --private-key=~/.ssh/ansible-key**



```
ubuntu@ip-172-31-40-59:~$ ansible all -a "free -h" -i /home/ubuntu/ansible/hosts --private-key=~/.ssh/ansible-key
server1 | CHANGED | rc=0 >>
               total        used        free      shared  buff/cache   available
Mem:           966Mi       198Mi       357Mi       0.0Ki       409Mi       627Mi
Swap:             0B          0B          0B
server3 | CHANGED | rc=0 >>
               total        used        free      shared  buff/cache   available
Mem:           966Mi       190Mi       365Mi       0.0Ki       410Mi       635Mi
Swap:             0B          0B          0B
server2 | CHANGED | rc=0 >>
               total        used        free      shared  buff/cache   available
Mem:           966Mi       186Mi       370Mi       0.0Ki       409Mi       640Mi
Swap:             0B          0B          0B
```

For uptime check : ansible all -a "uptime" -i /home/ubuntu/ansible/hosts --private-key=~/.ssh/ansible-key

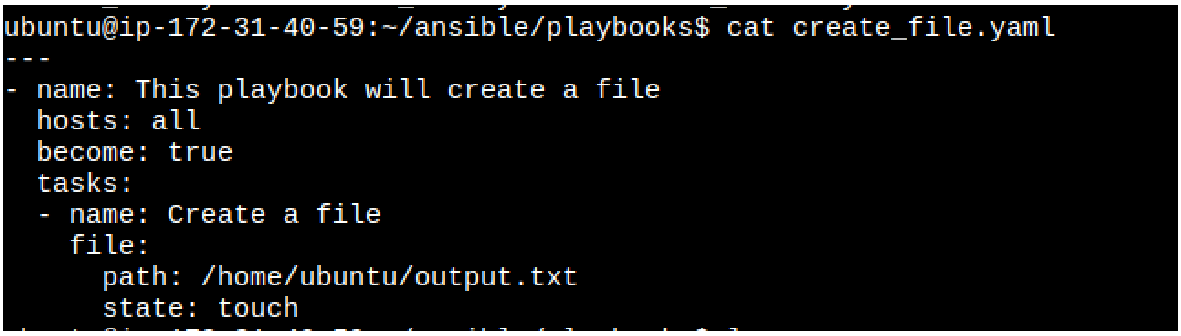


```
ubuntu@ip-172-31-40-59:~$ ansible all -a "uptime" -i /home/ubuntu/ansible/hosts --private-key=~/.ssh/ansible-key
server2 | CHANGED | rc=0 >>
 11:53:48 up 43 min,  1 user,  load average: 0.04, 0.01, 0.00
server1 | CHANGED | rc=0 >>
 11:53:48 up 44 min,  1 user,  load average: 0.00, 0.00, 0.00
server3 | CHANGED | rc=0 >>
 11:53:48 up 44 min,  1 user,  load average: 0.00, 0.00, 0.00
```

Date Jan 5, 2022

Deploying ansible playbooks and creating a file , users & installing docker from single master node to different servers

Step 1 : on the master node of ec2 i have made one playbooks which is **create_file.yaml** file



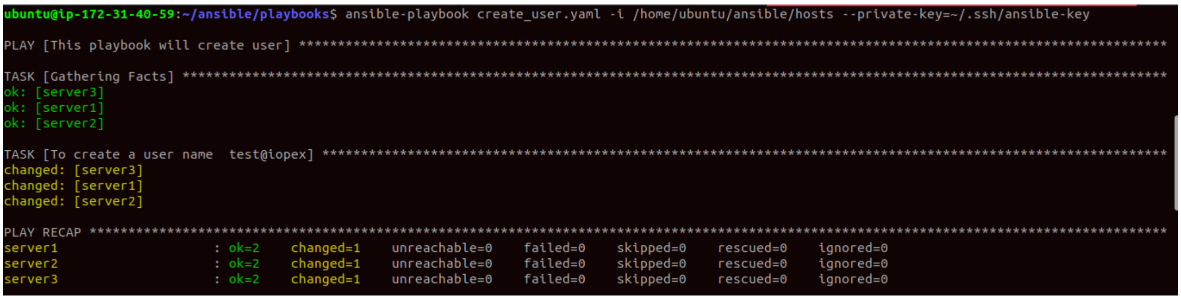
```
ubuntu@ip-172-31-40-59:~/ansible/playbooks$ cat create_file.yaml
---
- name: This playbook will create a file
  hosts: all
  become: true
  tasks:
  - name: Create a file
    file:
      path: /home/ubuntu/output.txt
      state: touch
```

Here I have defined hosts as global as root user along with a path where I wanted to store .

Step 2 : After creation of file i have deployed create_user.yaml with this command :

ansible-playbook create_user.yaml -i /home/ubuntu/ansible/hosts --private-key=~/.ssh/ansible-key

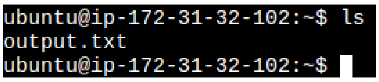
Output :



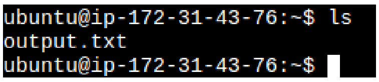
```
ubuntu@ip-172-31-40-59:~/ansible/playbooks$ ansible-playbook create_user.yaml -i /home/ubuntu/ansible/hosts --private-key=~/.ssh/ansible-key

PLAY [This playbook will create user] ************************************************************************************************

TASK [Gathering Facts] ***************************************************************************************************************
ok: [server3]
ok: [server1]
ok: [server2]

TASK [To create a user name  test@iopex] *********************************************************************************************
changed: [server3]
changed: [server1]
changed: [server2]

PLAY RECAP ***************************************************************************************************************************
server1                    : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
server2                    : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
server3                    : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

It shows changed status as 1 with all 3 servers .

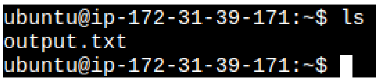
And output.txt file are accessible in all 3 servers :



```
ubuntu@ip-172-31-32-102:~$ ls
output.txt
ubuntu@ip-172-31-32-102:~$
```

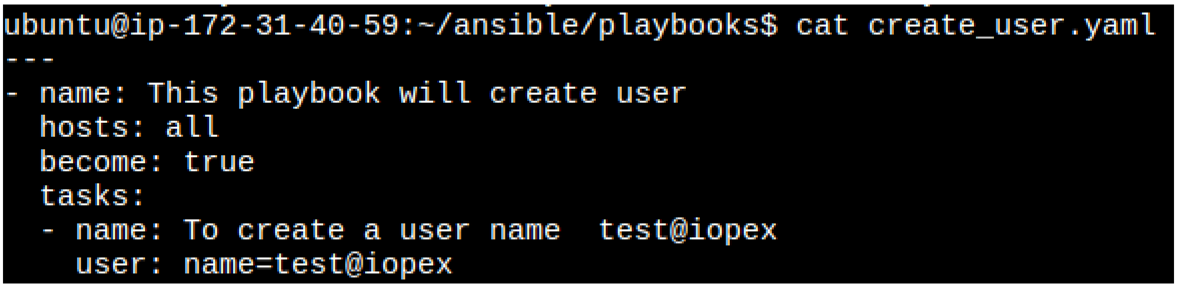


```
ubuntu@ip-172-31-43-76:~$ ls
output.txt
ubuntu@ip-172-31-43-76:~$
```



```
ubuntu@ip-172-31-39-171:~$ ls
output.txt
ubuntu@ip-172-31-39-171:~$
```

Step 3 : for creating a user from master node to different server i have created a yaml file which is a playbook again :

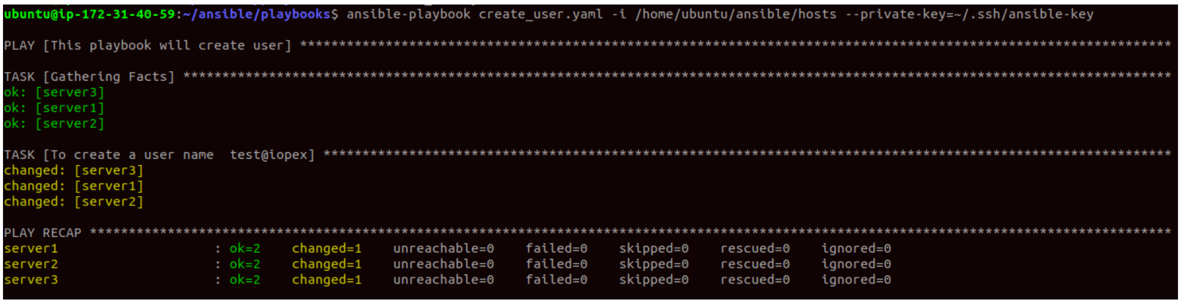


```
ubuntu@ip-172-31-40-59:~/ansible/playbooks$ cat create_user.yaml
---
- name: This playbook will create user
  hosts: all
  become: true
  tasks:
  - name: To create a user name  test@iopex
    user: name=test@iopex
```

Here i have defined name as test@iopex and made it global along with root user

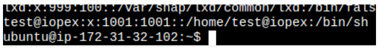
Step 4 : deploying my command with : **ansible-playbook create_user.yaml -i /home/ubuntu/ansible/hosts**

--private-key=~/.ssh/ansible-key

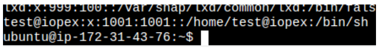


```
ubuntu@ip-172-31-40-59:~/ansible/playbooks$ ansible-playbook create_user.yaml -i /home/ubuntu/ansible/hosts --private-key=~/.ssh/ansible-key

PLAY [This playbook will create user] ******************************************************************************************

TASK [Gathering Facts] *********************************************************************************************************
ok: [server3]
ok: [server1]
ok: [server2]

TASK [To create a user name  test@iopex] ***************************************************************************************
changed: [server3]
changed: [server1]
changed: [server2]

PLAY RECAP *********************************************************************************************************************
server1                    : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
server2                    : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
server3                    : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

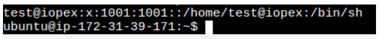
Output : using **cat etc/passwd** we can able to check users list in ubuntu system



```
lxd:x:999:100::/var/snap/lxd/common/lxd:/bin/fals
test@iopex:x:1001:1001::/home/test@iopex:/bin/sh
ubuntu@ip-172-31-32-102:~$
```

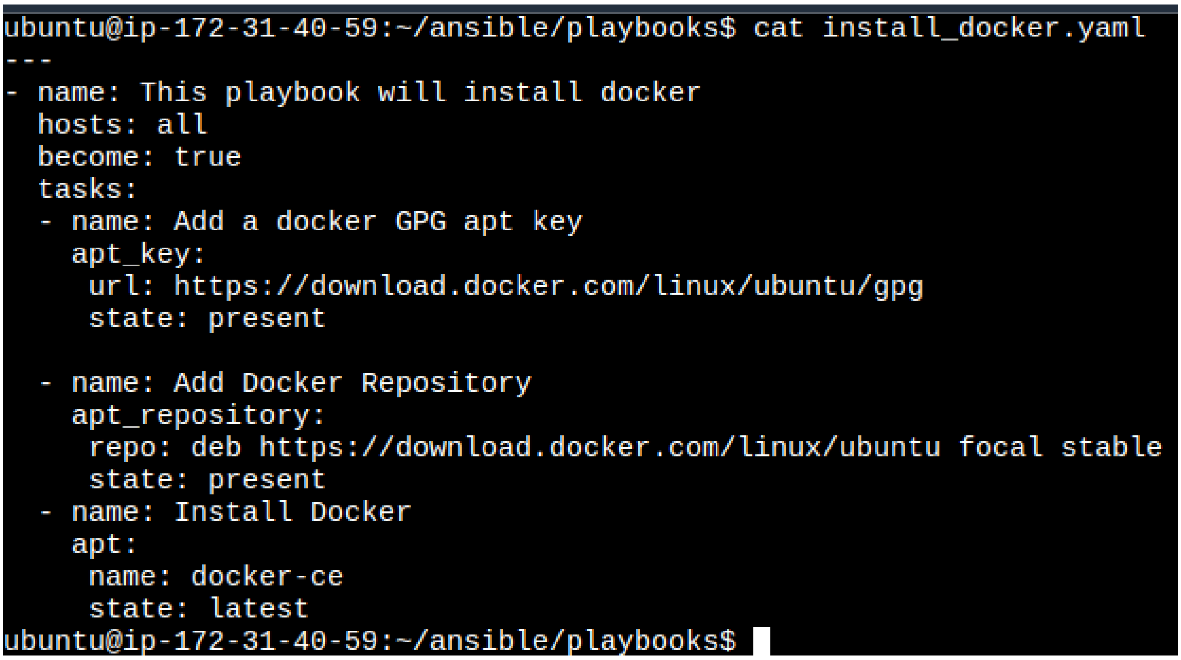


```
lxd:x:999:100::/var/snap/lxd/common/lxd:/bin/fals
test@iopex:x:1001:1001::/home/test@iopex:/bin/sh
ubuntu@ip-172-31-43-76:~$
```



```
test@iopex:x:1001:1001::/home/test@iopex:/bin/sh
ubuntu@ip-172-31-39-171:~$
```

Step 5 : now for my docker installation i have defined docker yaml playbook as well :

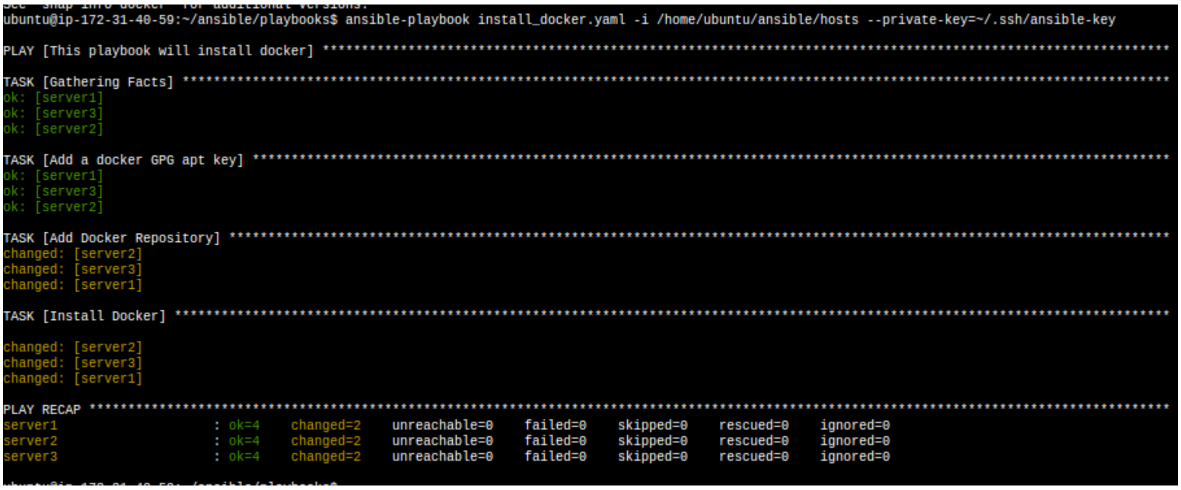


```
ubuntu@ip-172-31-40-59:~/ansible/playbooks$ cat install_docker.yaml
---
- name: This playbook will install docker
  hosts: all
  become: true
  tasks:
  - name: Add a docker GPG apt key
    apt_key:
     url: https://download.docker.com/linux/ubuntu/gpg
     state: present

  - name: Add Docker Repository
    apt_repository:
     repo: deb https://download.docker.com/linux/ubuntu focal stable
     state: present
  - name: Install Docker
    apt:
     name: docker-ce
     state: latest
ubuntu@ip-172-31-40-59:~/ansible/playbooks$
```

Here I have defined different conditions : 1. Storing GPG key from apt 2. Docker repository 3. From apt installing docker-ce with latest version

Step 6 : for deploying docker playbook with command :

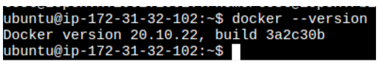
ansible-playbook install_docker.yaml -i /home/ubuntu/ansible/hosts --private-key=~/.ssh/ansible-key



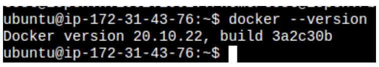
```
See 'snap info docker' for additional versions.
ubuntu@ip-172-31-40-59:~/ansible/playbooks$ ansible-playbook install_docker.yaml -i /home/ubuntu/ansible/hosts --private-key=~/.ssh/ansible-key

PLAY [This playbook will install docker] ***************************************************************************************

TASK [Gathering Facts] *********************************************************************************************************
ok: [server1]
ok: [server3]
ok: [server2]

TASK [Add a docker GPG apt key] ************************************************************************************************
ok: [server1]
ok: [server3]
ok: [server2]

TASK [Add Docker Repository] ***************************************************************************************************
changed: [server2]
changed: [server3]
changed: [server1]

TASK [Install Docker] **********************************************************************************************************

changed: [server2]
changed: [server3]
changed: [server1]

PLAY RECAP *********************************************************************************************************************
server1                    : ok=4    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
server2                    : ok=4    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
server3                    : ok=4    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ubuntu@ip-172-31-40-59:~/ansible/playbooks$
```

Installation done successfully as my change state is 2

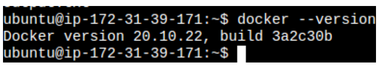
,checking servers whether docker have installed :



```
ubuntu@ip-172-31-32-102:~$ docker --version
Docker version 20.10.22, build 3a2c30b
ubuntu@ip-172-31-32-102:~$
```



```
ubuntu@ip-172-31-43-76:~$ docker --version
Docker version 20.10.22, build 3a2c30b
ubuntu@ip-172-31-43-76:~$
```



```
ubuntu@ip-172-31-39-171:~$ docker --version
Docker version 20.10.22, build 3a2c30b
ubuntu@ip-172-31-39-171:~$
```

Successfully installed from master node to different servers using ansible .