ANSIBLE

Ansible is an open-source automation tool used for **IT configuration management, deployment, orchestration, and automation**. It is designed to simplify complex processes and reduce manual effort by automating repetitive tasks.

Ansible Used in DevOps

Ansible is widely used in **DevOps** due to the following reasons:

1. **Infrastructure as Code (IaC):**
    - Ansible allows you to define your infrastructure and configurations in code, enabling version control and repeatability.

2. **Agentless Architecture:**
    - Unlike other configuration management tools, Ansible does not require an agent to be installed on the target systems. It communicates over **SSH** or **WinRM**.

3. **Simple YAML Syntax:**
    - Ansible uses **YAML** for its playbooks, which is human-readable and easy to learn, even for non-programmers.

4. **Cross-Platform:**
    - Ansible supports multiple operating systems, including Linux, Windows, and macOS.

5. **Automation Across the Lifecycle:**
    - It automates tasks like configuration management, application deployment, and orchestration of services.

How Ansible Works

1. **Key Components:**
    - **Control Node:**
        - The machine where Ansible is installed and executed. It manages the target systems.
    - **Managed Nodes:**
        - The systems (servers, VMs, etc.) that Ansible manages. They are typically referred to as hosts.
    - **Inventory:**
        - A file that lists the managed nodes (hosts). It can be static or dynamic.
    - **Modules:**
        - Predefined scripts that Ansible uses to perform tasks like installing packages, copying files, and managing services.
    - **Playbooks:**
        - YAML files containing a series of tasks to be executed on the managed nodes.
    - **Ad-hoc Commands:**
        - One-time commands to quickly execute a task on a managed node.
2. **Workflow:**

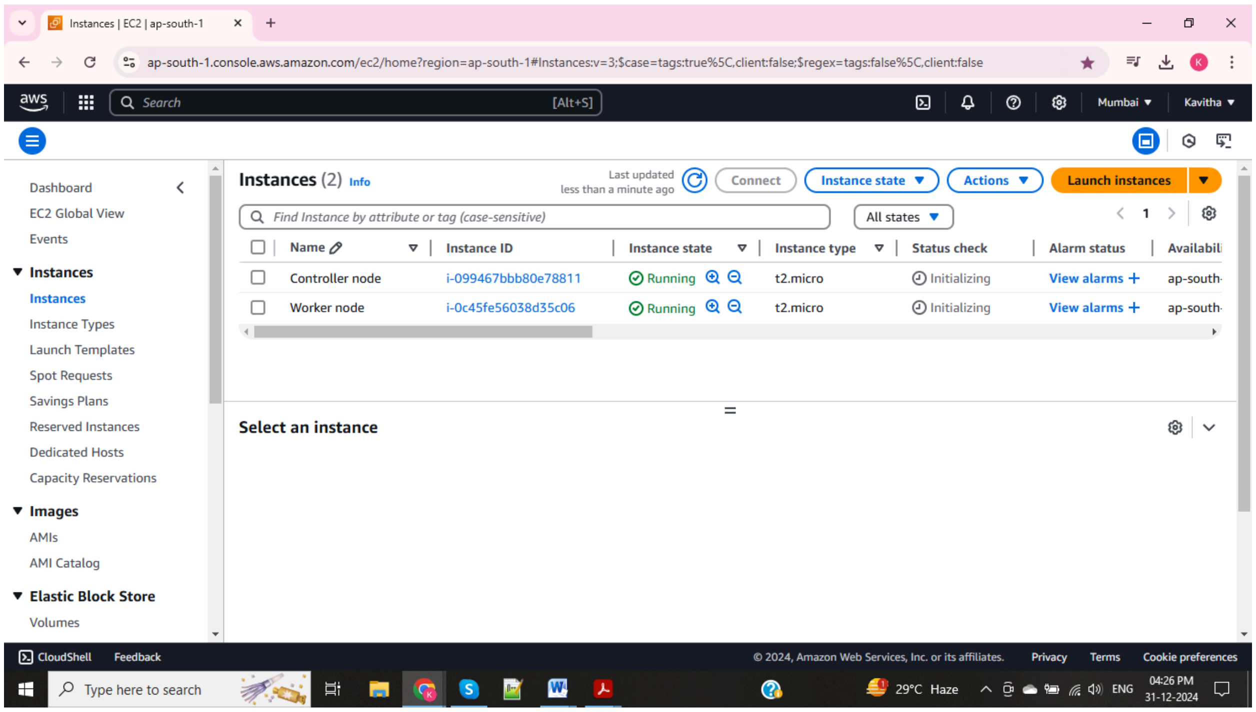
- **Step 1: Define Inventory**
    - List the IPs or hostnames of target systems in an inventory file.

- **Step 2: Write a Playbook**
    - Create a YAML file defining the tasks to perform.

- **Step 3: Execute**
    - Run the playbook using the `ansible-playbook` command.

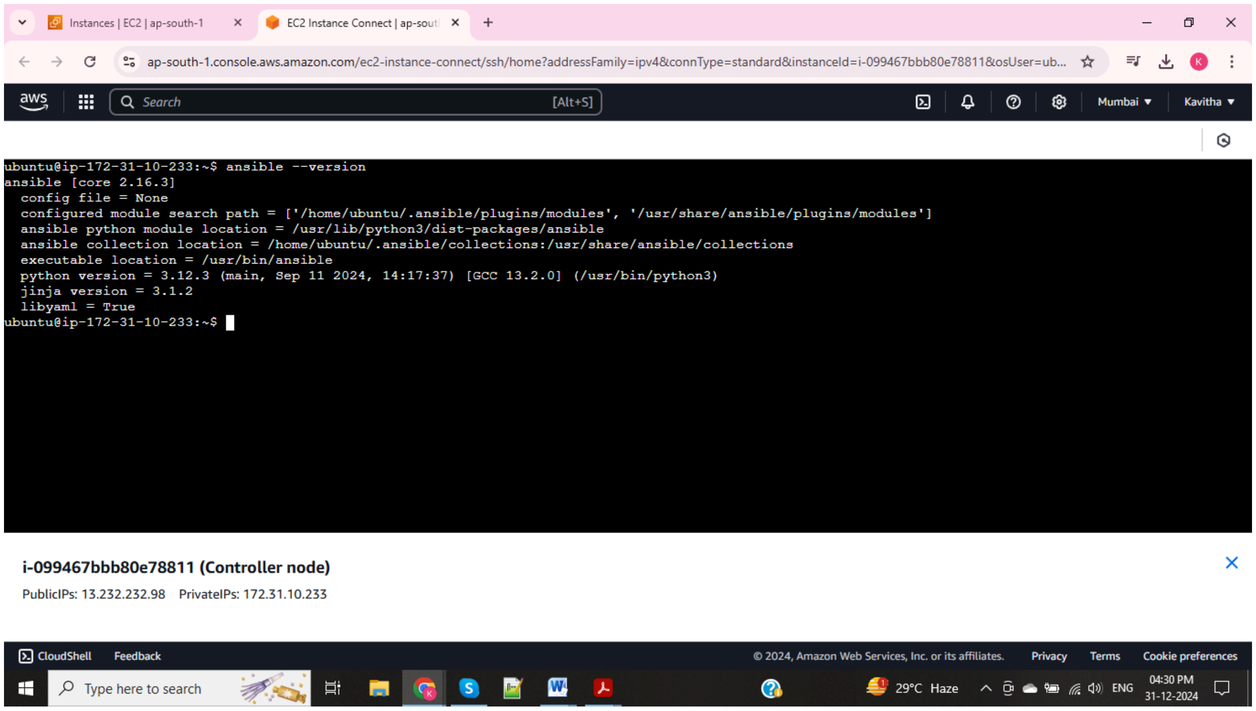
Deploying NGINX on a slave server using an Ansible playbook:

Created a EC2 instance named as Controller node and worker node



Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availabil
Controller node	i-099467bbb80e78811	Running	t2.micro	Initializing	View alarms +	ap-south-
Worker node	i-0c45fe56038d35c06	Running	t2.micro	Initializing	View alarms +	ap-south-

In Controller node installed ansible

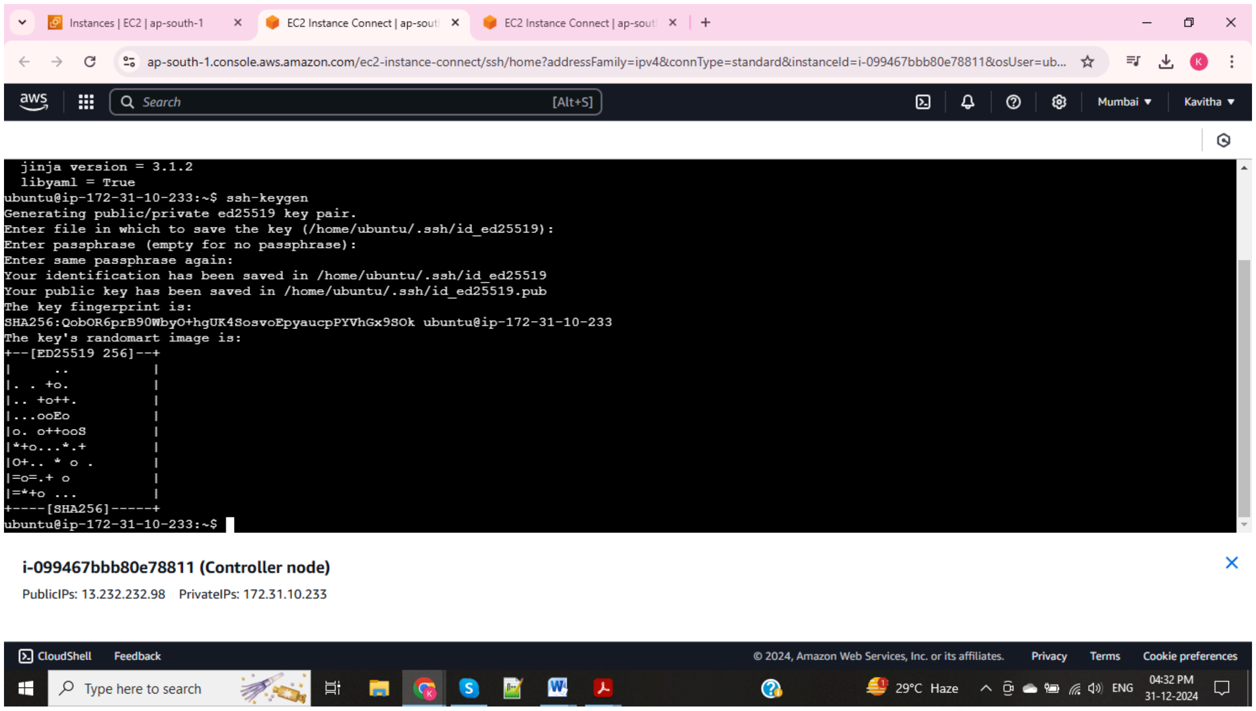

Instances | EC2 | ap-south-1
EC2 Instance Connect | ap-sout
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?addressFamily=ipv4&connType=standard&instanceId=i-099467bbb80e78811&osUser=ub...
Search
Mumbai
Kavitha
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
29°C Haze
04:30 PM
31-12-2024

```
ubuntu@ip-172-31-10-233:~$ ansible --version
ansible [core 2.16.3]
  config file = None
  configured module search path = ['/home/ubuntu/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /home/ubuntu/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.12.3 (main, Sep 11 2024, 14:17:37) [GCC 13.2.0] (/usr/bin/python3)
  jinja version = 3.1.2
  libyaml = True
ubuntu@ip-172-31-10-233:~$
```

i-099467bbb80e78811 (Controller node)

PublicIPs: 13.232.232.98 PrivateIPs: 172.31.10.233

Created a ssh-keygen in ansible server for Passwordless Authentication

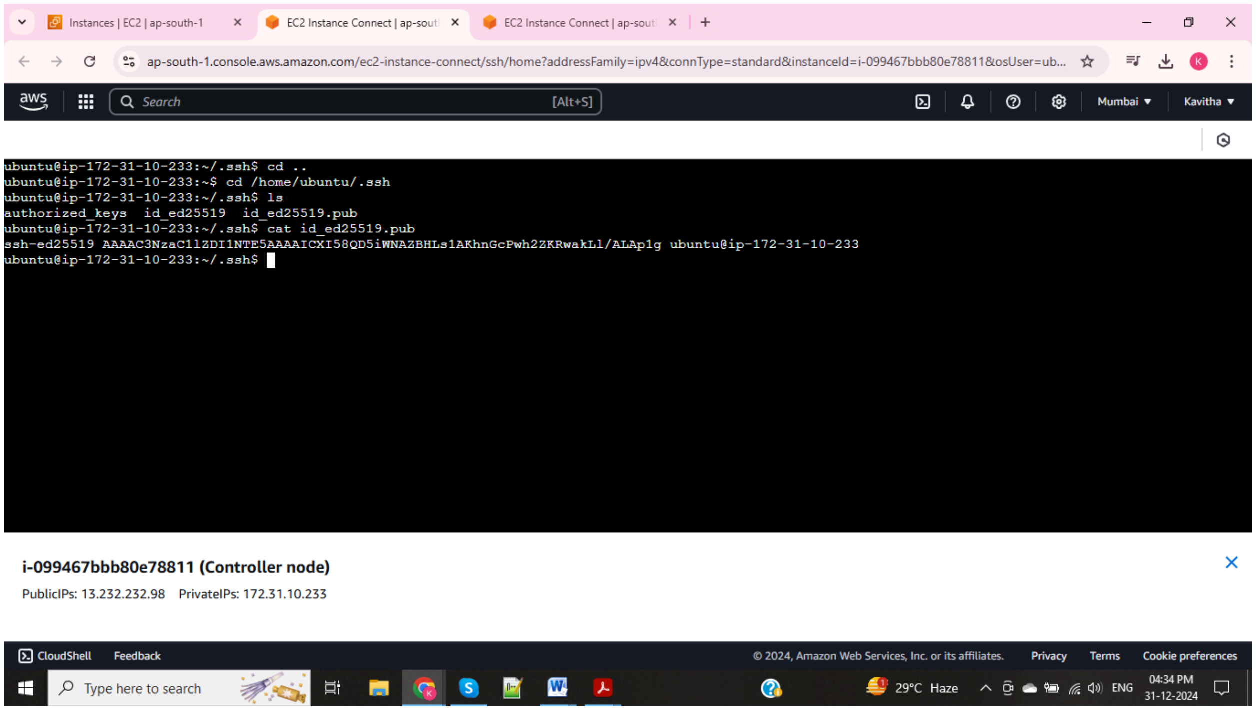


```
  jinja version = 3.1.2
  libyaml = True
ubuntu@ip-172-31-10-233:~$ ssh-keygen
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/ubuntu/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ubuntu/.ssh/id_ed25519
Your public key has been saved in /home/ubuntu/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:QobOR6prB90WbyO+hgUK4SosvoEpyaucpPYVhGx9SOk ubuntu@ip-172-31-10-233
The key's randomart image is:
+--[ED25519 256]--+
|      ..         |
|. . +o.          |
|.. +o++.         |
|...ooEo          |
|o. o++ooS        |
|*+o...*.+        |
|O+.. * o .       |
|=o=.+ o          |
|=*+o ...         |
+----[SHA256]-----+
ubuntu@ip-172-31-10-233:~$
```

i-099467bbb80e78811 (Controller node)

PublicIPs: 13.232.232.98 PrivateIPs: 172.31.10.233

To view the public key use “cat” --> Copy the key and it has to be pasted in slave server

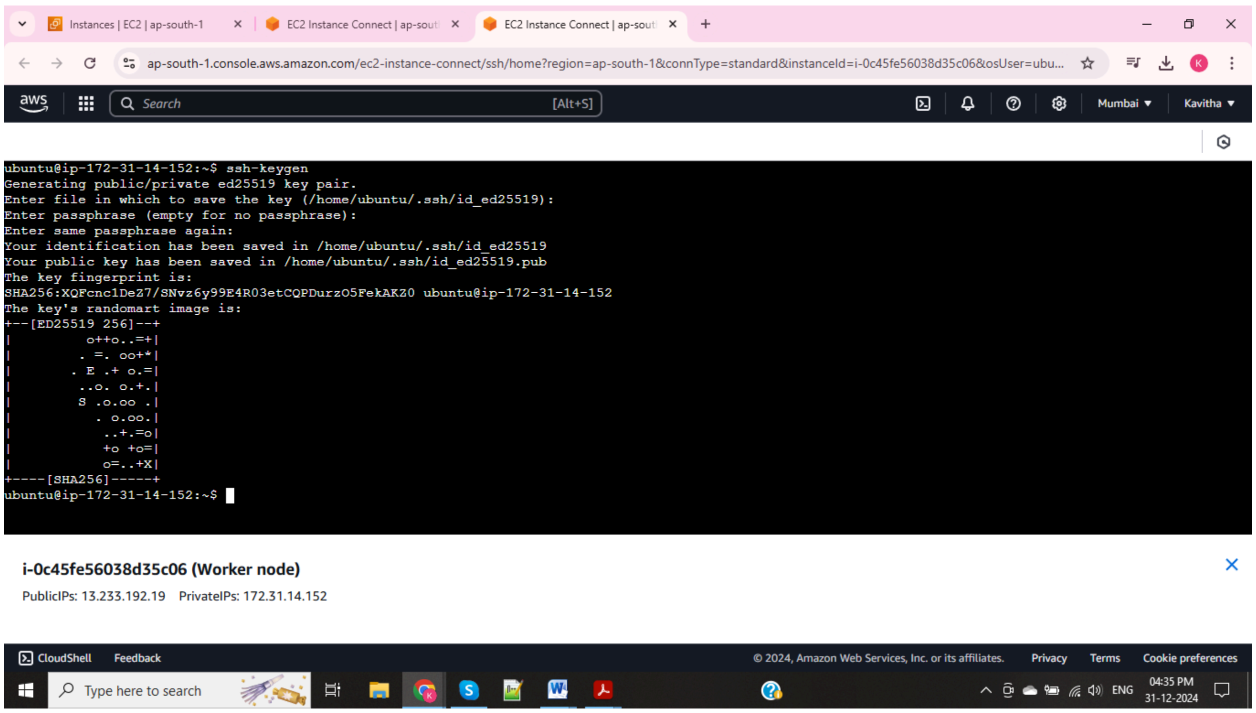


```
ubuntu@ip-172-31-10-233:~/.ssh$ cd ..
ubuntu@ip-172-31-10-233:~$ cd /home/ubuntu/.ssh
ubuntu@ip-172-31-10-233:~/.ssh$ ls
authorized_keys  id_ed25519  id_ed25519.pub
ubuntu@ip-172-31-10-233:~/.ssh$ cat id_ed25519.pub
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAICXI58QD5iWNAZBHLs1AKhnGcPwh2ZKRwakLl/ALAp1g ubuntu@ip-172-31-10-233
ubuntu@ip-172-31-10-233:~/.ssh$
```

i-099467bbb80e78811 (Controller node)

PublicIPs: 13.232.232.98 PrivateIPs: 172.31.10.233

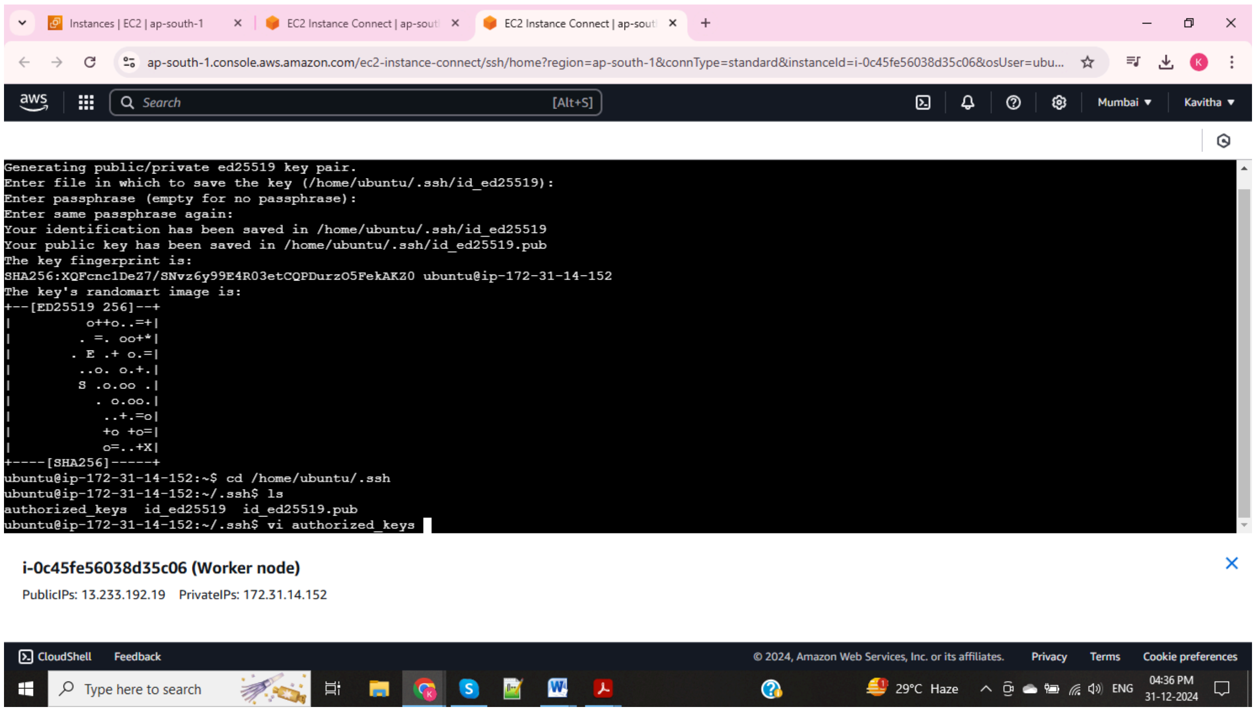
Connected the slave server and key created in this server also



```
ubuntu@ip-172-31-14-152:~$ ssh-keygen
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/ubuntu/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ubuntu/.ssh/id_ed25519
Your public key has been saved in /home/ubuntu/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:XQFcnc1DeZ7/SNvz6y99E4R03etCQPDurzO5FekAKZ0 ubuntu@ip-172-31-14-152
The key's randomart image is:
+--[ED25519 256]--+
|          o++o..=+|
|         . =. oo+*|
|        . E .+ o.=|
|         ..o. o.+.|
|         S .o.oo .|
|           . o.oo.|
|            ..+.=o|
|            +o +o=|
|            o=..+X|
+----[SHA256]-----+
ubuntu@ip-172-31-14-152:~$
```

i-0c45fe56038d35c06 (Worker node)
PublicIPs: 13.233.192.19 PrivateIPs: 172.31.14.152

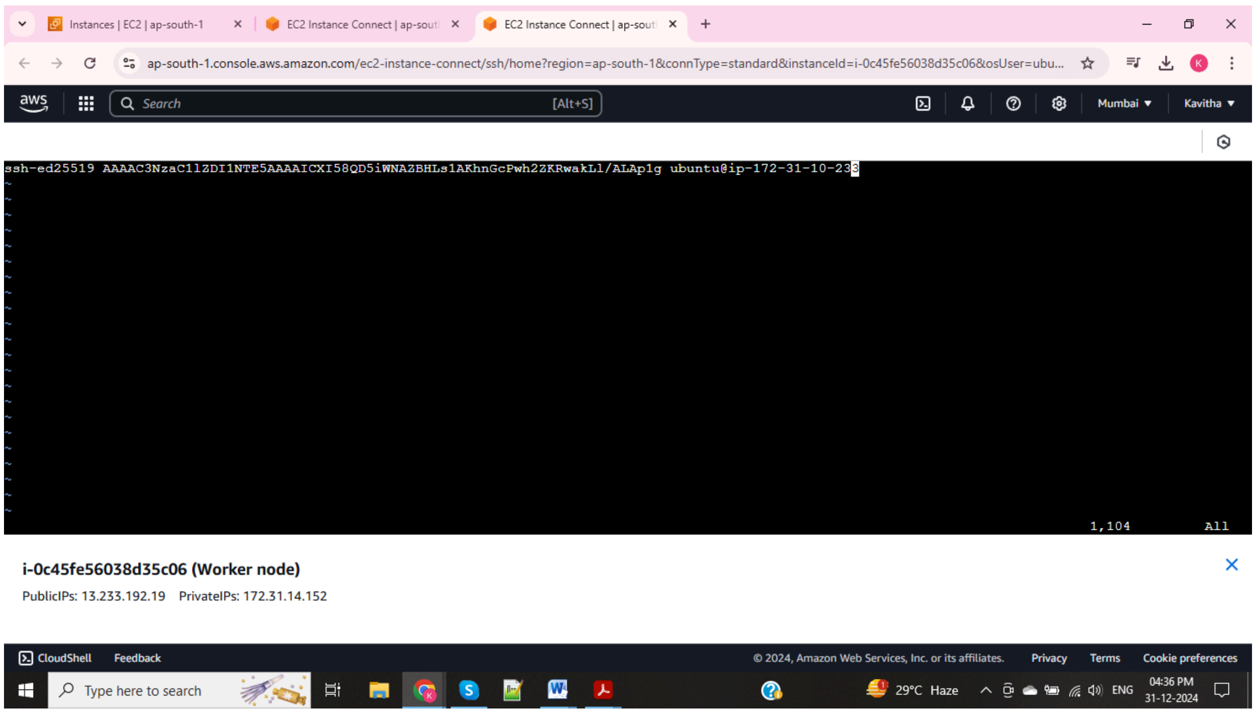
Editing the **authorized_keys** file on a server is typically done to configure **passwordless Authentication SSH access**.



```
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/ubuntu/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ubuntu/.ssh/id_ed25519
Your public key has been saved in /home/ubuntu/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:XQFcnc1DeZ7/SNvz6y99E4R03etCQPDurzO5FekAKZ0 ubuntu@ip-172-31-14-152
The key's randomart image is:
+--[ED25519 256]--+
|          o++o..=+|
|         . =. oo+*|
|        . E .+ o.=|
|         ..o. o.+.|
|         S .o.oo .|
|           . o.oo.|
|            ..+.=o|
|            +o +o=|
|            o=..+X|
+----[SHA256]-----+
ubuntu@ip-172-31-14-152:~$ cd /home/ubuntu/.ssh
ubuntu@ip-172-31-14-152:~/.ssh$ ls
authorized_keys  id_ed25519  id_ed25519.pub
ubuntu@ip-172-31-14-152:~/.ssh$ vi authorized_keys
```

i-0c45fe56038d35c06 (Worker node)
PublicIPs: 13.233.192.19 PrivateIPs: 172.31.14.152

Enable Passwordless SSH for Automation.

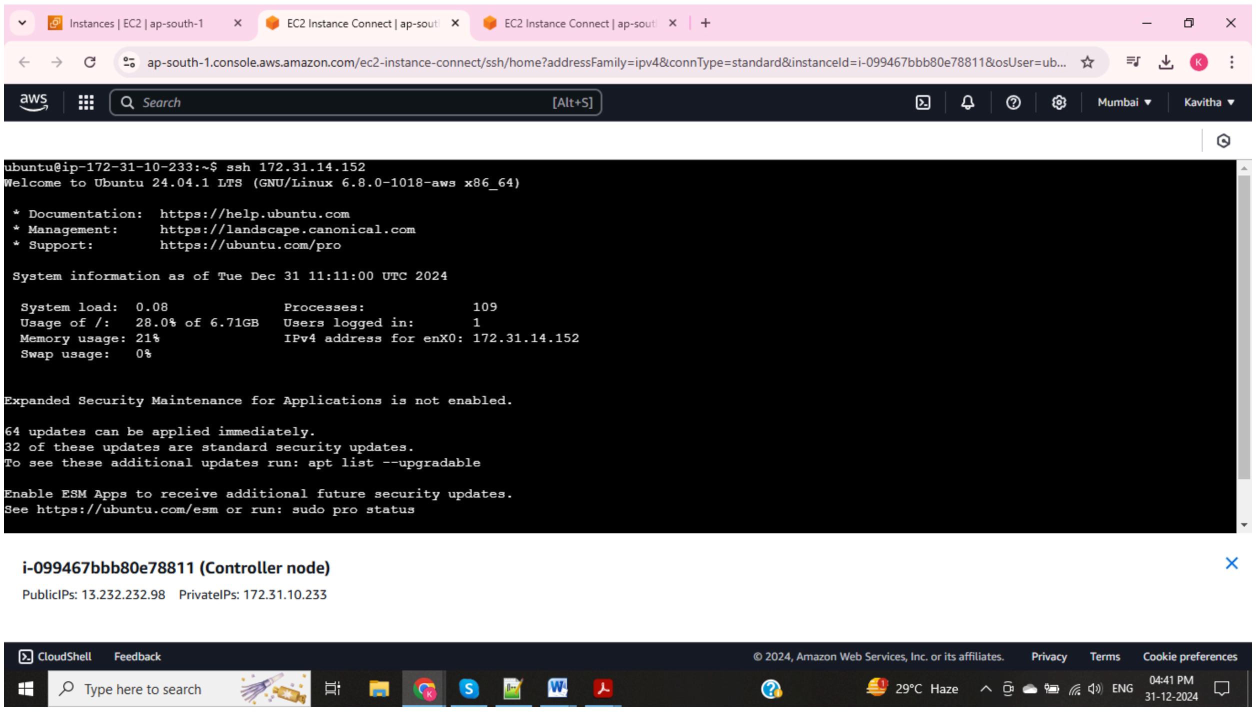


```
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAICXI58QD5iWNAZBHLs1AKhnGcPwh2ZKRwakLl/ALAp1g ubuntu@ip-172-31-10-233
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
                                                                                          1,104         All
```

i-0c45fe56038d35c06 (Worker node)

PublicIPs: 13.233.192.19 PrivateIPs: 172.31.14.152

Logged into ansible server and checking the slave server was working: ssh 172.31.14.152

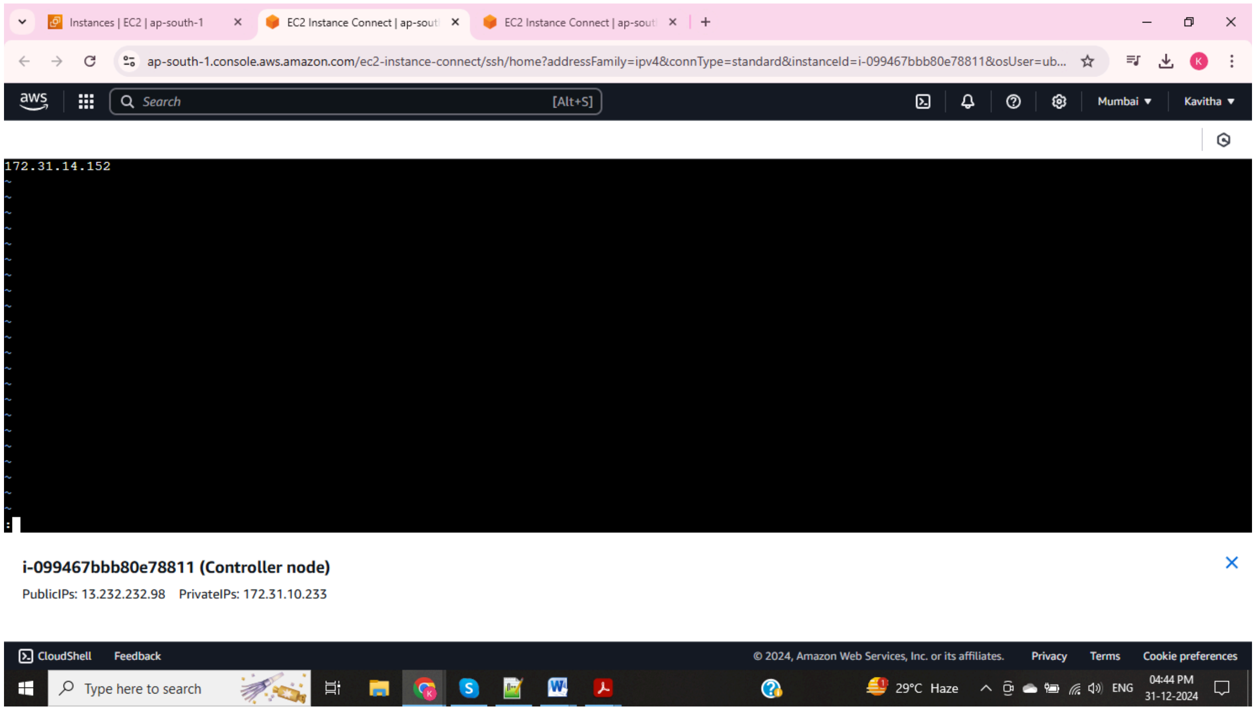


```
ubuntu@ip-172-31-10-233:~$ ssh 172.31.14.152
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1018-aws x86_64)

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

 System information as of Tue Dec 31 11:11:00 UTC 2024

  System load:  0.08              Processes:             109
  Usage of /:   28.0% of 6.71GB   Users logged in:       1
  Memory usage: 21%               IPv4 address for enX0: 172.31.14.152
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

64 updates can be applied immediately.
32 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status
```

i-099467bbb80e78811 (Controller node)

PublicIPs: 13.232.232.98 PrivateIPs: 172.31.10.233

Creating a INVENTORY FILE for Ansible will interact with slave servers

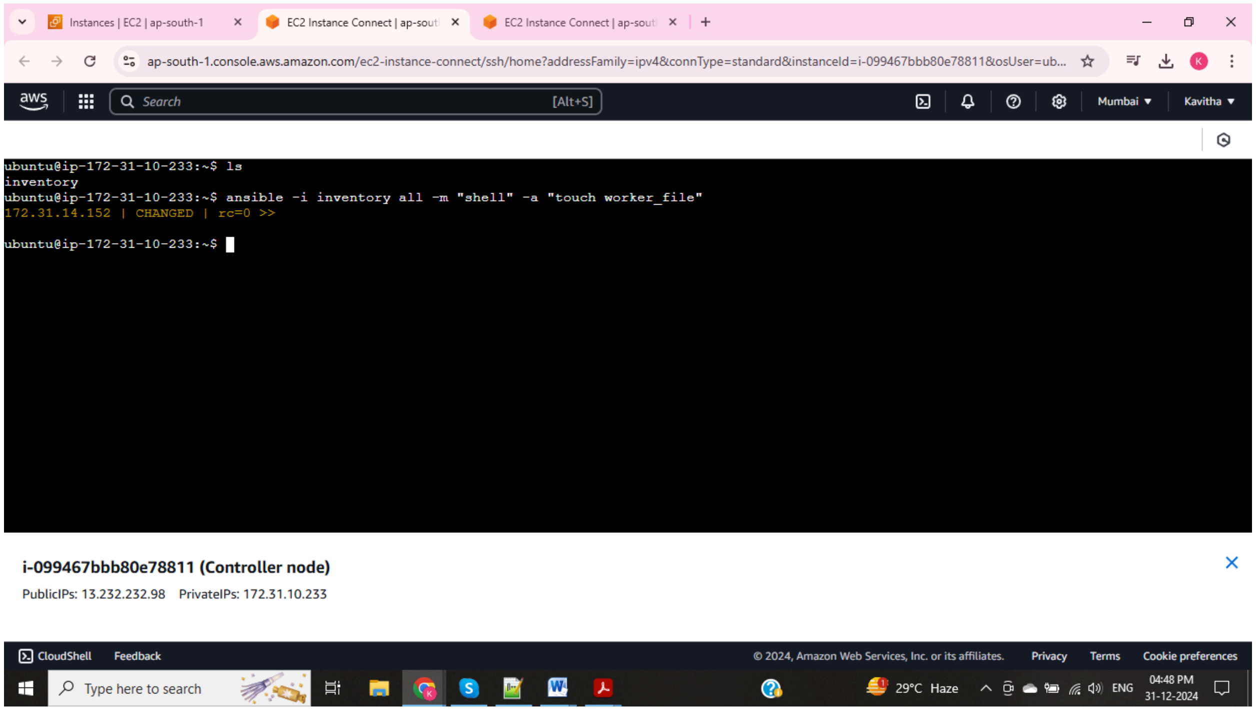


```
172.31.14.152
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:
```

i-099467bbb80e78811 (Controller node)

PublicIPs: 13.232.232.98 PrivateIPs: 172.31.10.233

In **Ansible**, **ad-hoc commands** are simple, one-line commands used to quickly perform tasks on remote systems without creating a full **playbook**. These commands are executed from the command line and are ideal for short, repetitive tasks or one-off operations.

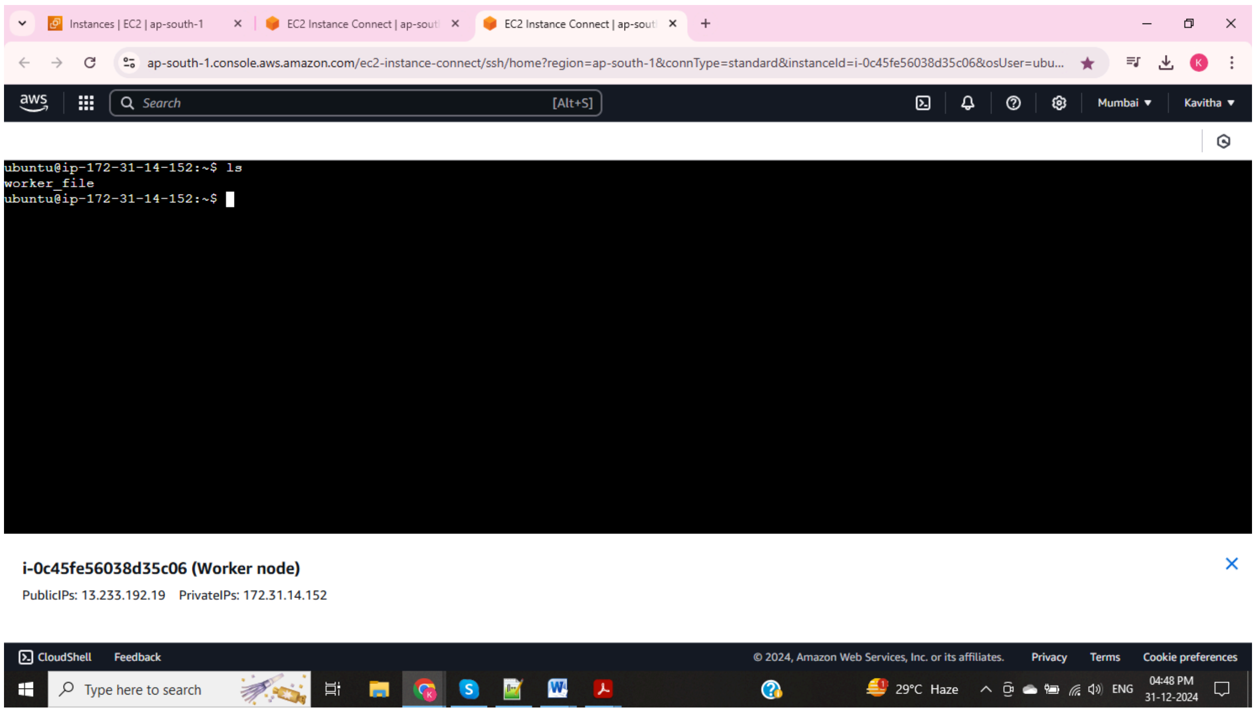


```
ubuntu@ip-172-31-10-233:~$ ls
inventory
ubuntu@ip-172-31-10-233:~$ ansible -i inventory all -m "shell" -a "touch worker_file"
172.31.14.152 | CHANGED | rc=0 >>

ubuntu@ip-172-31-10-233:~$
```

i-099467bbb80e78811 (Controller node)

PublicIPs: 13.232.232.98 PrivateIPs: 172.31.10.233

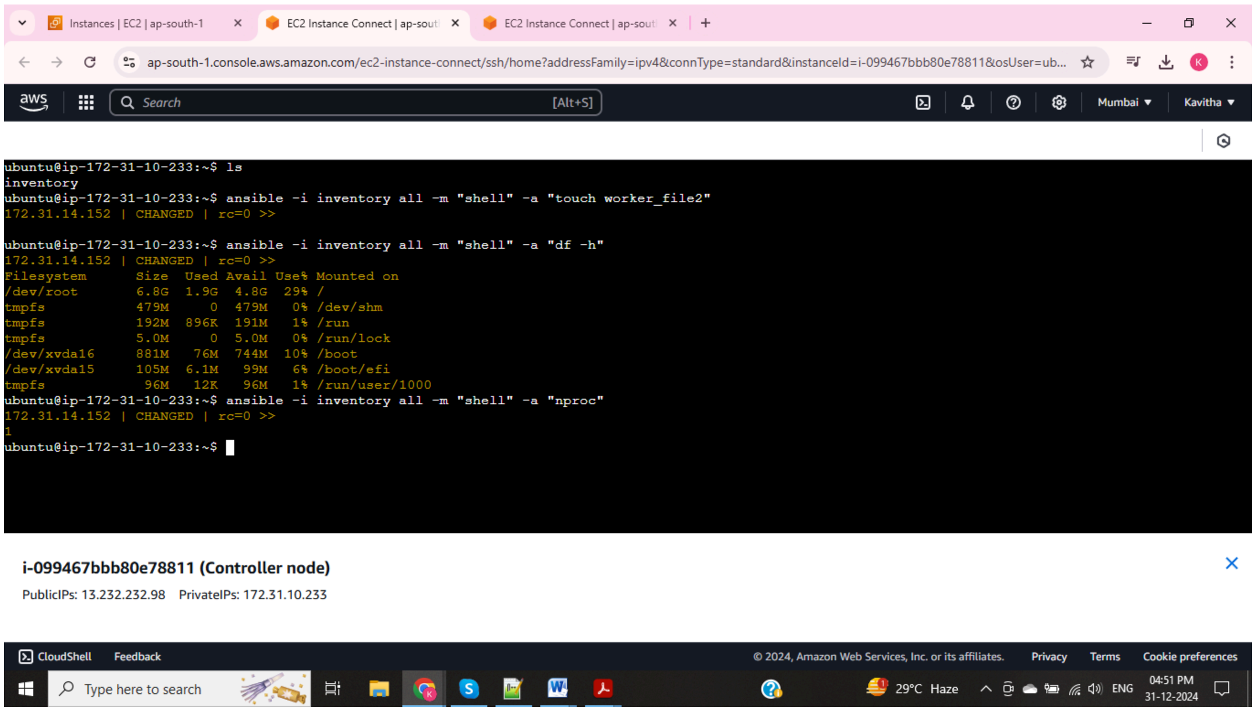
Checking adhoc commands are working and the output shows in slave server



```
ubuntu@ip-172-31-14-152:~$ ls
worker_file
ubuntu@ip-172-31-14-152:~$
```

i-0c45fe56038d35c06 (Worker node)
PublicIPs: 13.233.192.19 PrivateIPs: 172.31.14.152

Execute some adhoc commands

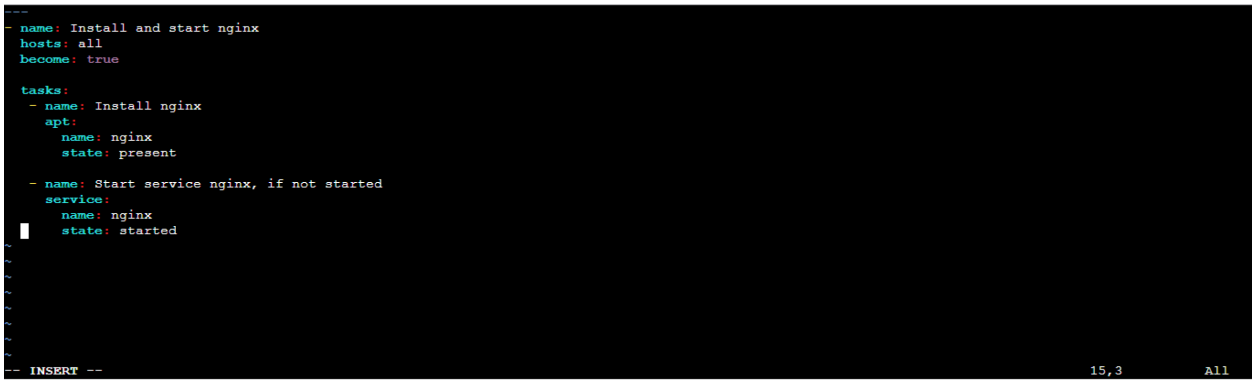


```
ubuntu@ip-172-31-10-233:~$ ls
inventory
ubuntu@ip-172-31-10-233:~$ ansible -i inventory all -m "shell" -a "touch worker_file2"
172.31.14.152 | CHANGED | rc=0 >>

ubuntu@ip-172-31-10-233:~$ ansible -i inventory all -m "shell" -a "df -h"
172.31.14.152 | CHANGED | rc=0 >>
Filesystem      Size  Used Avail Use% Mounted on
/dev/root       6.8G  1.9G  4.8G  29% /
tmpfs           479M     0  479M   0% /dev/shm
tmpfs           192M  896K  191M   1% /run
tmpfs           5.0M     0  5.0M   0% /run/lock
/dev/xvda16     881M   76M  744M  10% /boot
/dev/xvda15     105M  6.1M   99M   6% /boot/efi
tmpfs            96M   12K   96M   1% /run/user/1000
ubuntu@ip-172-31-10-233:~$ ansible -i inventory all -m "shell" -a "nproc"
172.31.14.152 | CHANGED | rc=0 >>
1
ubuntu@ip-172-31-10-233:~$
```

i-099467bbb80e78811 (Controller node)
PublicIPs: 13.232.232.98 PrivateIPs: 172.31.10.233

In **Ansible**, a **playbook** is a YAML file that defines a series of tasks to be executed on managed hosts. It provides a more structured, repeatable, and reusable way to automate configuration management, application deployment, and other IT tasks compared to ad-hoc commands.

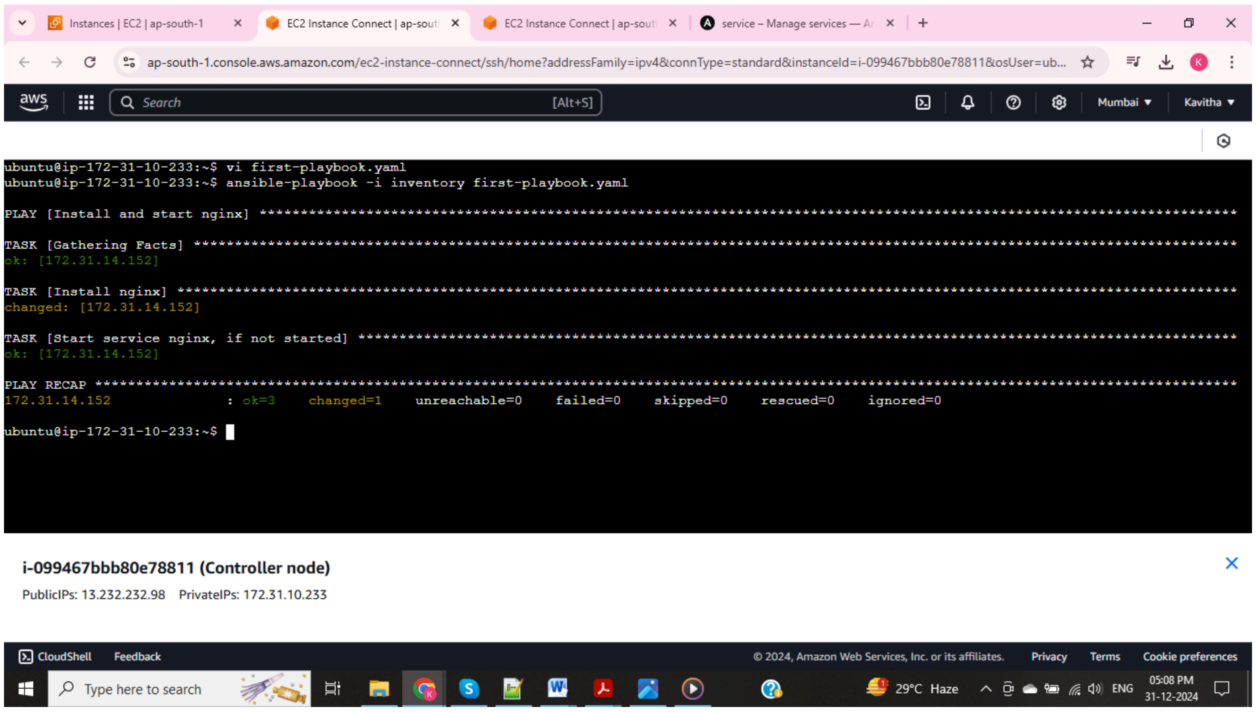
Using ansible playbook YAML file created to install nginx and start nginx service



```
---
- name: Install and start nginx
  hosts: all
  become: true

  tasks:
    - name: Install nginx
      apt:
        name: nginx
        state: present

    - name: Start service nginx, if not started
      service:
        name: nginx
        state: started
~
~
~
~
~
~
~
~
-- INSERT --                                                                15,3          All
```

To run and check the playbook with the help of this command:

ansible-playbook -i inventory first-playbook.yaml and success output



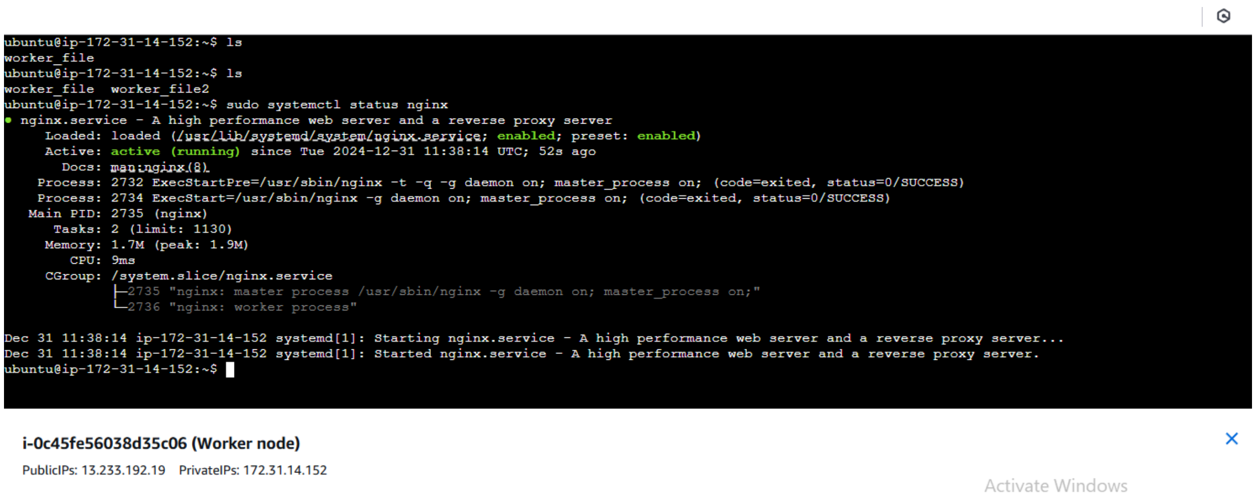
Instances | EC2 | ap-south-1
EC2 Instance Connect | ap-sout
EC2 Instance Connect | ap-sout
service – Manage services — A
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?addressFamily=ipv4&connType=standard&instanceId=i-099467bbb80e78811&osUser=ub...
aws
Search
[Alt+S]
Mumbai
Kavitha

```
ubuntu@ip-172-31-10-233:~$ vi first-playbook.yaml
ubuntu@ip-172-31-10-233:~$ ansible-playbook -i inventory first-playbook.yaml

PLAY [Install and start nginx] ***************************************************************************

TASK [Gathering Facts] ***********************************************************************************
ok: [172.31.14.152]

TASK [Install nginx] *************************************************************************************
changed: [172.31.14.152]

TASK [Start service nginx, if not started] ***************************************************************
ok: [172.31.14.152]

PLAY RECAP ***********************************************************************************************
172.31.14.152              : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ubuntu@ip-172-31-10-233:~$
```

i-099467bbb80e78811 (Controller node)

PublicIPs: 13.232.232.98   PrivateIPs: 172.31.10.233

For checking the output of playbook in slave server use: sudo systemctl status nginx

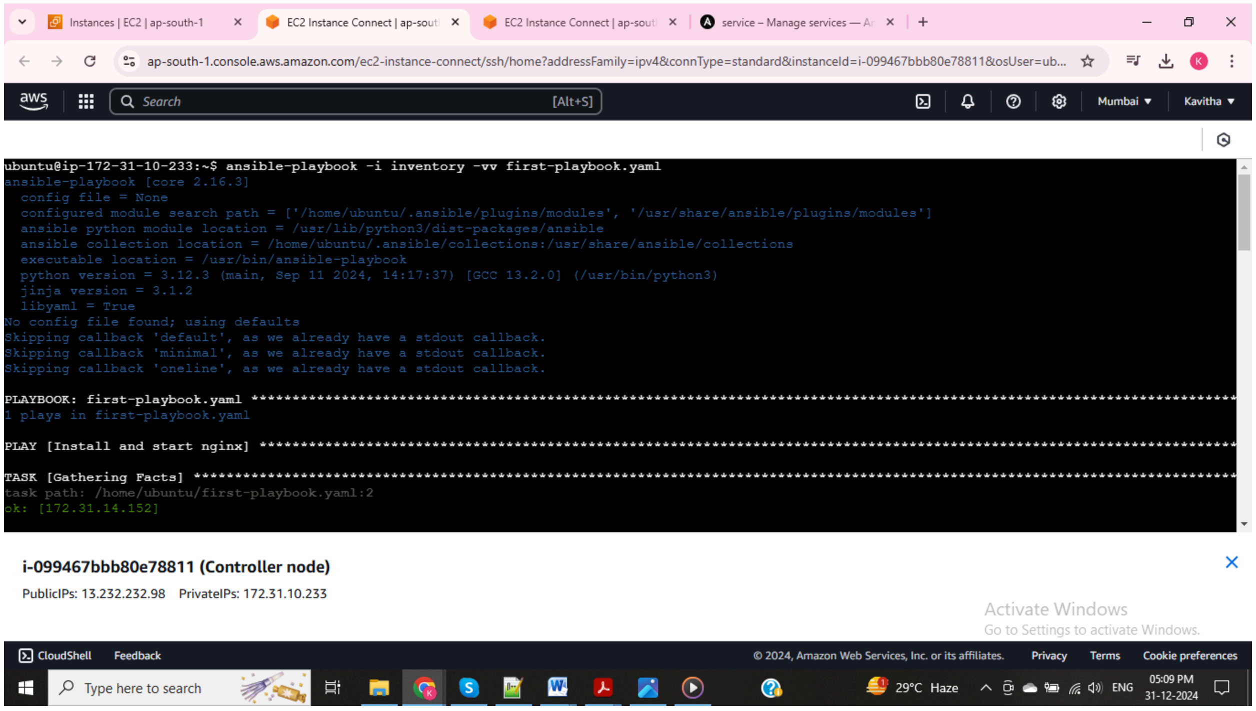


```
ubuntu@ip-172-31-14-152:~$ ls
worker_file
ubuntu@ip-172-31-14-152:~$ ls
worker_file  worker_file2
ubuntu@ip-172-31-14-152:~$ sudo systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
     Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: enabled)
     Active: active (running) since Tue 2024-12-31 11:38:14 UTC; 52s ago
       Docs: man:nginx(8)
    Process: 2732 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
    Process: 2734 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Main PID: 2735 (nginx)
      Tasks: 2 (limit: 1130)
     Memory: 1.7M (peak: 1.9M)
        CPU: 9ms
     CGroup: /system.slice/nginx.service
             ├─2735 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             └─2736 "nginx: worker process"

Dec 31 11:38:14 ip-172-31-14-152 systemd[1]: Starting nginx.service - A high performance web server and a reverse proxy server...
Dec 31 11:38:14 ip-172-31-14-152 systemd[1]: Started nginx.service - A high performance web server and a reverse proxy server.
ubuntu@ip-172-31-14-152:~$
```

i-0c45fe56038d35c06 (Worker node)

PublicIPs: 13.233.192.19 PrivateIPs: 172.31.14.152

Verbosity refers to the level of detail provided in the output logs when running a playbook or command. It controls how much information Ansible displays during execution, helping users to debug or understand the process better. Verbosity can be set using the `-v`, `-vv`, `-vvv`, and `-vvvv` flags, with each level providing more detailed information.



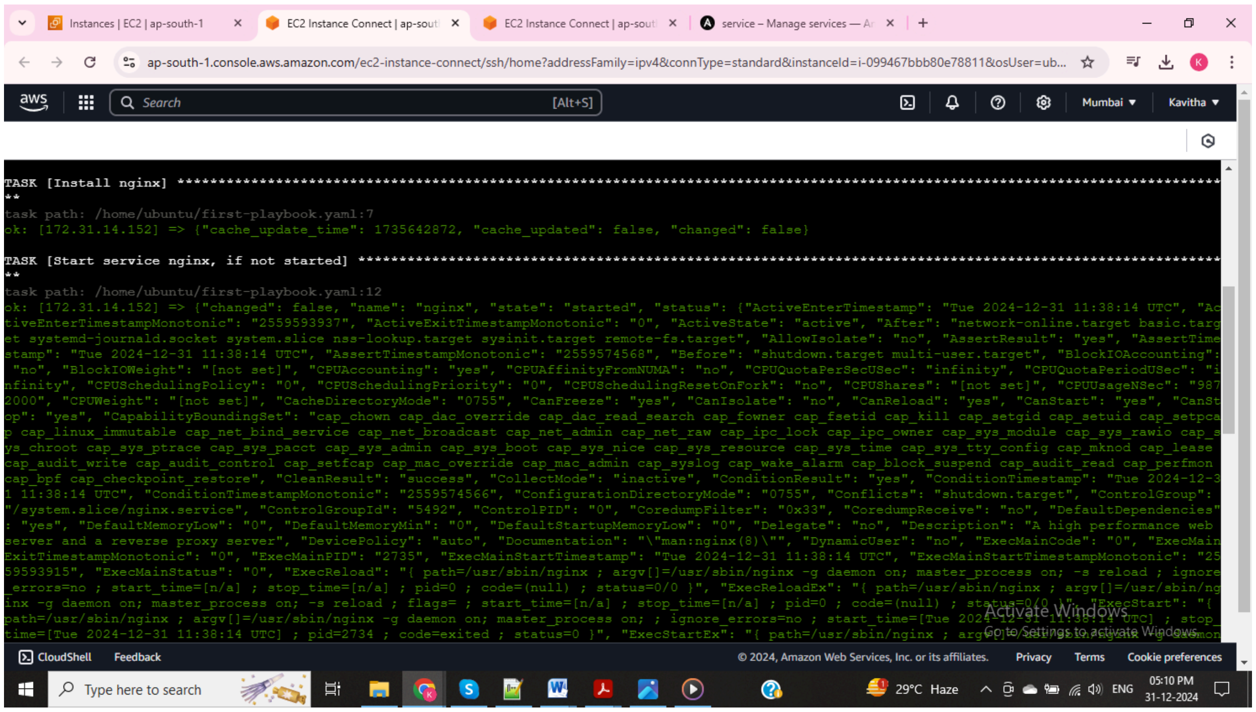
Instances | EC2 | ap-south-1
EC2 Instance Connect | ap-sout
EC2 Instance Connect | ap-sout
service – Manage services — A
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?addressFamily=ipv4&connType=standard&instanceId=i-099467bbb80e78811&osUser=ub...
Search
[Alt+S]
Mumbai
Kavitha

```
ubuntu@ip-172-31-10-233:~$ ansible-playbook -i inventory -vv first-playbook.yaml
ansible-playbook [core 2.16.3]
  config file = None
  configured module search path = ['/home/ubuntu/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /home/ubuntu/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible-playbook
  python version = 3.12.3 (main, Sep 11 2024, 14:17:37) [GCC 13.2.0] (/usr/bin/python3)
  jinja version = 3.1.2
  libyaml = True
No config file found; using defaults
Skipping callback 'default', as we already have a stdout callback.
Skipping callback 'minimal', as we already have a stdout callback.
Skipping callback 'oneline', as we already have a stdout callback.

PLAYBOOK: first-playbook.yaml *****************************************************************************************
1 plays in first-playbook.yaml

PLAY [Install and start nginx] ****************************************************************************************

TASK [Gathering Facts] ************************************************************************************************
task path: /home/ubuntu/first-playbook.yaml:2
ok: [172.31.14.152]
```

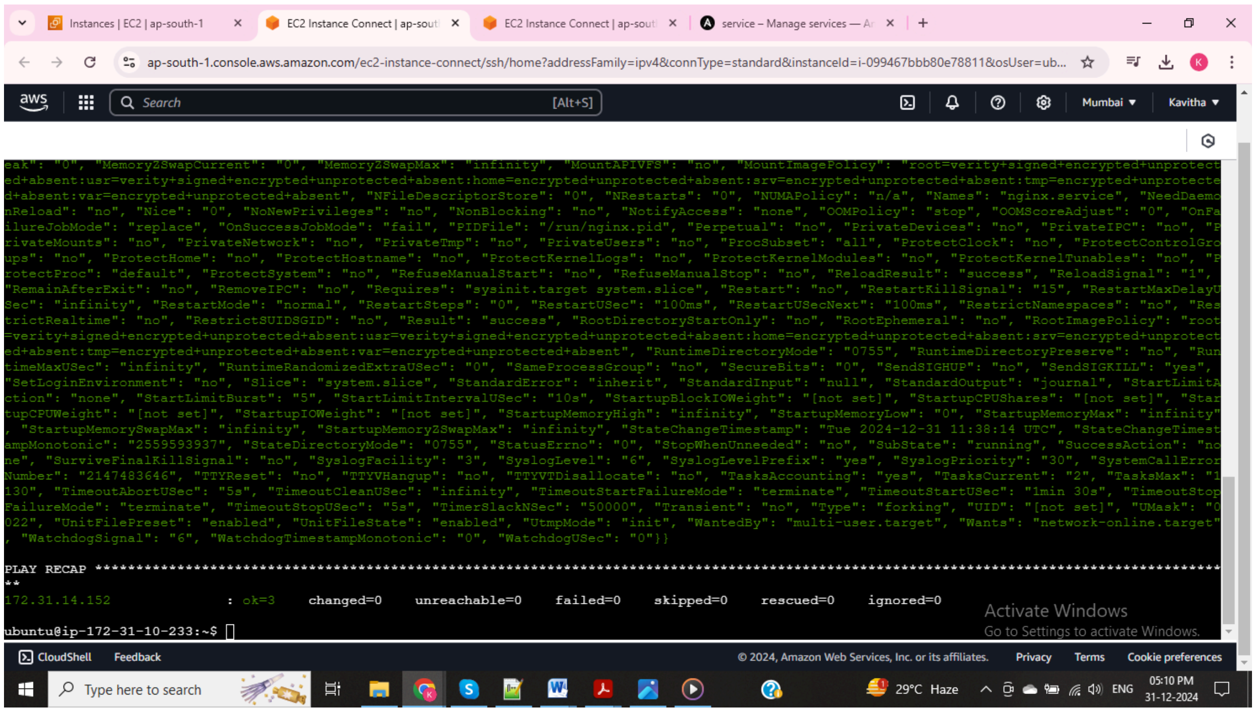
i-099467bbb80e78811 (Controller node)

PublicIPs: 13.232.232.98 PrivateIPs: 172.31.10.233

I used **-vv (More verbose)**: In addition to the basic output, shows additional details, such as the parameters passed to tasks.

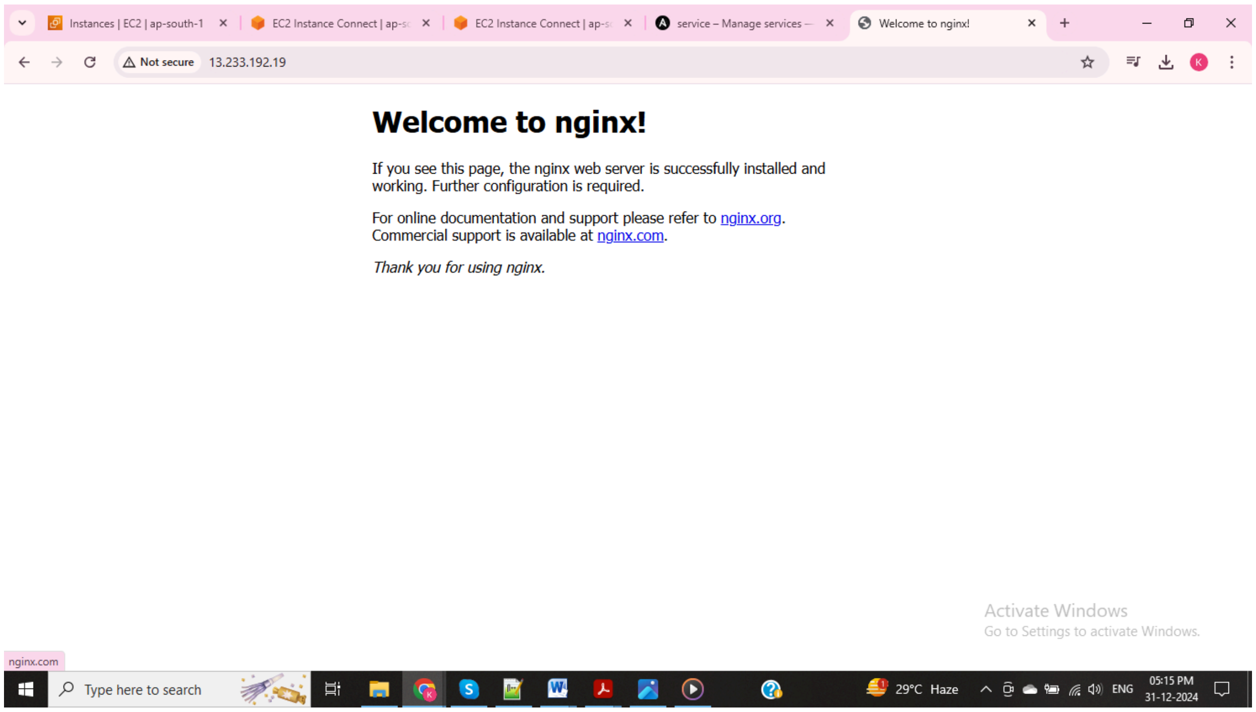


```
TASK [Install nginx] ****************************************************************************************
**
task path: /home/ubuntu/first-playbook.yaml:7
ok: [172.31.14.152] => {"cache_update_time": 1735642872, "cache_updated": false, "changed": false}

TASK [Start service nginx, if not started] ******************************************************************
**
task path: /home/ubuntu/first-playbook.yaml:12
ok: [172.31.14.152] => {"changed": false, "name": "nginx", "state": "started", "status": {"ActiveEnterTimestamp": "Tue 2024-12-31 11:38:14 UTC", "Ac
tiveEnterTimestampMonotonic": "2559593937", "ActiveExitTimestampMonotonic": "0", "ActiveState": "active", "After": "network-online.target basic.targ
et systemd-journald.socket system.slice nss-lookup.target sysinit.target remote-fs.target", "AllowIsolate": "no", "AssertResult": "yes", "AssertTime
stamp": "Tue 2024-12-31 11:38:14 UTC", "AssertTimestampMonotonic": "2559574568", "Before": "shutdown.target multi-user.target", "BlockIOAccounting":
 "no", "BlockIOWeight": "[not set]", "CPUAccounting": "yes", "CPUAffinityFromNUMA": "no", "CPUQuotaPerSecUSec": "infinity", "CPUQuotaPeriodUSec": "i
nfinity", "CPUSchedulingPolicy": "0", "CPUSchedulingPriority": "0", "CPUSchedulingResetOnFork": "no", "CPUShares": "[not set]", "CPUUsageNSec": "987
2000", "CPUWeight": "[not set]", "CacheDirectoryMode": "0755", "CanFreeze": "yes", "CanIsolate": "no", "CanReload": "yes", "CanStart": "yes", "CanSt
op": "yes", "CapabilityBoundingSet": "cap_chown cap_dac_override cap_dac_read_search cap_fowner cap_fsetid cap_kill cap_setgid cap_setuid cap_setpca
p cap_linux_immutable cap_net_bind_service cap_net_broadcast cap_net_admin cap_net_raw cap_ipc_lock cap_ipc_owner cap_sys_module cap_sys_rawio cap_s
ys_chroot cap_sys_ptrace cap_sys_pacct cap_sys_admin cap_sys_boot cap_sys_nice cap_sys_resource cap_sys_time cap_sys_tty_config cap_mknod cap_lease
cap_audit_write cap_audit_control cap_setfcap cap_mac_override cap_mac_admin cap_syslog cap_wake_alarm cap_block_suspend cap_audit_read cap_perfmon
cap_bpf cap_checkpoint_restore", "CleanResult": "success", "CollectMode": "inactive", "ConditionResult": "yes", "ConditionTimestamp": "Tue 2024-12-3
1 11:38:14 UTC", "ConditionTimestampMonotonic": "2559574566", "ConfigurationDirectoryMode": "0755", "Conflicts": "shutdown.target", "ControlGroup":
"/system.slice/nginx.service", "ControlGroupId": "5492", "ControlPID": "0", "CoredumpFilter": "0x33", "CoredumpReceive": "no", "DefaultDependencies"
: "yes", "DefaultMemoryLow": "0", "DefaultMemoryMin": "0", "DefaultStartupMemoryLow": "0", "Delegate": "no", "Description": "A high performance web
server and a reverse proxy server", "DevicePolicy": "auto", "Documentation": "\"man:nginx(8)\"", "DynamicUser": "no", "ExecMainCode": "0", "ExecMain
ExitTimestampMonotonic": "0", "ExecMainPID": "2735", "ExecMainStartTimestamp": "Tue 2024-12-31 11:38:14 UTC", "ExecMainStartTimestampMonotonic": "25
59593915", "ExecMainStatus": "0", "ExecReload": "{ path=/usr/sbin/nginx ; argv[]=/usr/sbin/nginx -g daemon on; master_process on; -s reload ; ignore
_errors=no ; start_time=[n/a] ; stop_time=[n/a] ; pid=0 ; code=(null) ; status=0/0 }", "ExecReloadEx": "{ path=/usr/sbin/nginx ; argv[]=/usr/sbin/ng
inx -g daemon on; master_process on; -s reload ; flags= ; start_time=[n/a] ; stop_time=[n/a] ; pid=0 ; code=(null) ; status=0/0 }", "ExecStart": "{
path=/usr/sbin/nginx ; argv[]=/usr/sbin/nginx -g daemon on; master_process on; ; ignore_errors=no ; start_time=[Tue 2024-12-31 11:38:14 UTC] ; stop_
time=[Tue 2024-12-31 11:38:14 UTC] ; pid=2734 ; code=exited ; status=0 }", "ExecStartEx": "{ path=/usr/sbin/nginx ; argv[]=/usr/sbin/nginx -g daemon
```



```
eak": "0", "MemoryZSwapCurrent": "0", "MemoryZSwapMax": "infinity", "MountAPIVFS": "no", "MountImagePolicy": "root=verity+signed+encrypted+unprotect
ed+absent:usr=verity+signed+encrypted+unprotected+absent:home=encrypted+unprotected+absent:srv=encrypted+unprotected+absent:tmp=encrypted+unprotecte
d+absent:var=encrypted+unprotected+absent", "NFileDescriptorStore": "0", "NRestarts": "0", "NUMAPolicy": "n/a", "Names": "nginx.service", "NeedDaemo
nReload": "no", "Nice": "0", "NoNewPrivileges": "no", "NonBlocking": "no", "NotifyAccess": "none", "OOMPolicy": "stop", "OOMScoreAdjust": "0", "OnFa
ilureJobMode": "replace", "OnSuccessJobMode": "fail", "PIDFile": "/run/nginx.pid", "Perpetual": "no", "PrivateDevices": "no", "PrivateIPC": "no", "P
rivateMounts": "no", "PrivateNetwork": "no", "PrivateTmp": "no", "PrivateUsers": "no", "ProcSubset": "all", "ProtectClock": "no", "ProtectControlGro
ups": "no", "ProtectHome": "no", "ProtectHostname": "no", "ProtectKernelLogs": "no", "ProtectKernelModules": "no", "ProtectKernelTunables": "no", "P
rotectProc": "default", "ProtectSystem": "no", "RefuseManualStart": "no", "RefuseManualStop": "no", "ReloadResult": "success", "ReloadSignal": "1",
"RemainAfterExit": "no", "RemoveIPC": "no", "Requires": "sysinit.target system.slice", "Restart": "no", "RestartKillSignal": "15", "RestartMaxDelayU
Sec": "infinity", "RestartMode": "normal", "RestartSteps": "0", "RestartUSec": "100ms", "RestartUSecNext": "100ms", "RestrictNamespaces": "no", "Res
trictRealtime": "no", "RestrictSUIDSGID": "no", "Result": "success", "RootDirectoryStartOnly": "no", "RootEphemeral": "no", "RootImagePolicy": "root
=verity+signed+encrypted+unprotected+absent:usr=verity+signed+encrypted+unprotected+absent:home=encrypted+unprotected+absent:srv=encrypted+unprotect
ed+absent:tmp=encrypted+unprotected+absent:var=encrypted+unprotected+absent", "RuntimeDirectoryMode": "0755", "RuntimeDirectoryPreserve": "no", "Run
timeMaxUSec": "infinity", "RuntimeRandomizedExtraUSec": "0", "SameProcessGroup": "no", "SecureBits": "0", "SendSIGHUP": "no", "SendSIGKILL": "yes",
"SetLoginEnvironment": "no", "Slice": "system.slice", "StandardError": "inherit", "StandardInput": "null", "StandardOutput": "journal", "StartLimitA
ction": "none", "StartLimitBurst": "5", "StartLimitIntervalUSec": "10s", "StartupBlockIOWeight": "[not set]", "StartupCPUShares": "[not set]", "Star
tupCPUWeight": "[not set]", "StartupIOWeight": "[not set]", "StartupMemoryHigh": "infinity", "StartupMemoryLow": "0", "StartupMemoryMax": "infinity"
, "StartupMemorySwapMax": "infinity", "StartupMemoryZSwapMax": "infinity", "StateChangeTimestamp": "Tue 2024-12-31 11:38:14 UTC", "StateChangeTimest
ampMonotonic": "2559593937", "StateDirectoryMode": "0755", "StatusErrno": "0", "StopWhenUnneeded": "no", "SubState": "running", "SuccessAction": "no
ne", "SurviveFinalKillSignal": "no", "SyslogFacility": "3", "SyslogLevel": "6", "SyslogLevelPrefix": "yes", "SyslogPriority": "30", "SystemCallError
Number": "2147483646", "TTYReset": "no", "TTYVHangup": "no", "TTYVTDisallocate": "no", "TasksAccounting": "yes", "TasksCurrent": "2", "TasksMax": "1
130", "TimeoutAbortUSec": "5s", "TimeoutCleanUSec": "infinity", "TimeoutStartFailureMode": "terminate", "TimeoutStartUSec": "1min 30s", "TimeoutStop
FailureMode": "terminate", "TimeoutStopUSec": "5s", "TimerSlackNSec": "50000", "Transient": "no", "Type": "forking", "UID": "[not set]", "UMask": "0
022", "UnitFilePreset": "enabled", "UnitFileState": "enabled", "UtmpMode": "init", "WantedBy": "multi-user.target", "Wants": "network-online.target"
, "WatchdogSignal": "6", "WatchdogTimestampMonotonic": "0", "WatchdogUSec": "0"}}

PLAY RECAP **************************************************************************************************
**
172.31.14.152              : ok=3    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ubuntu@ip-172-31-10-233:~$
```

Nginx is running in slave server to view the output use its public IP address



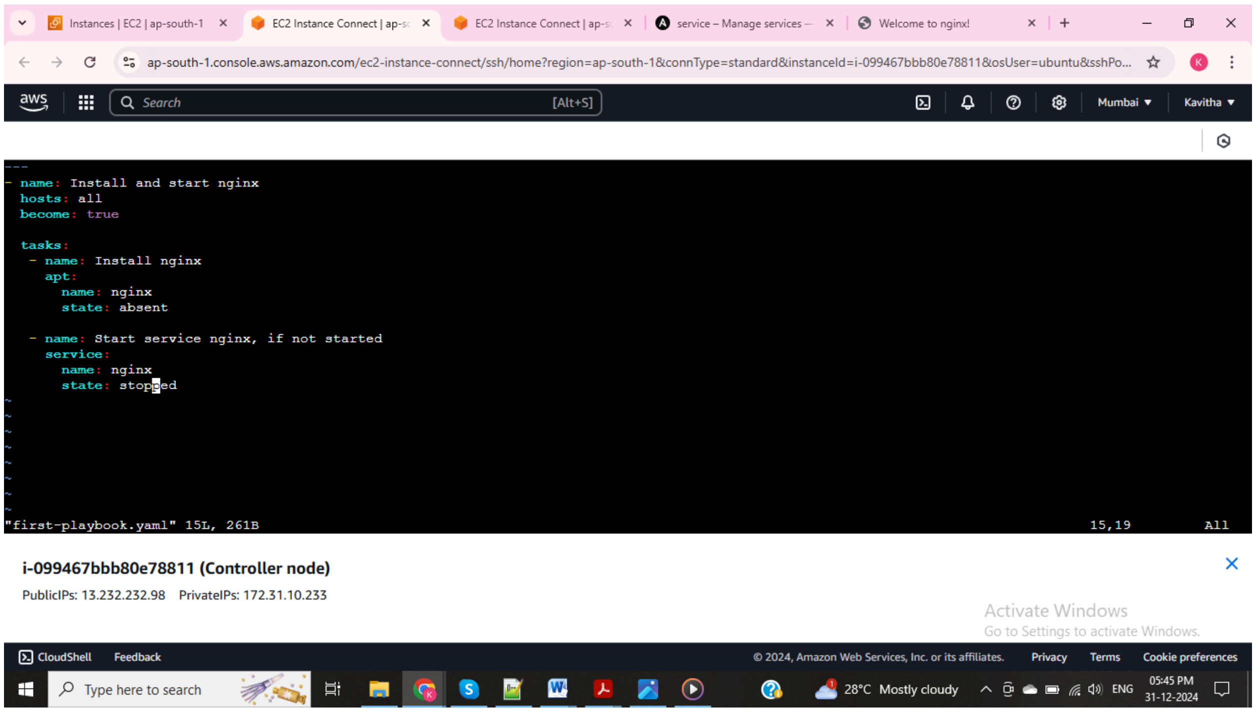
Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org. Commercial support is available at nginx.com.

Thank you for using nginx.

Now, Edited the first playbook to stop the nginx application

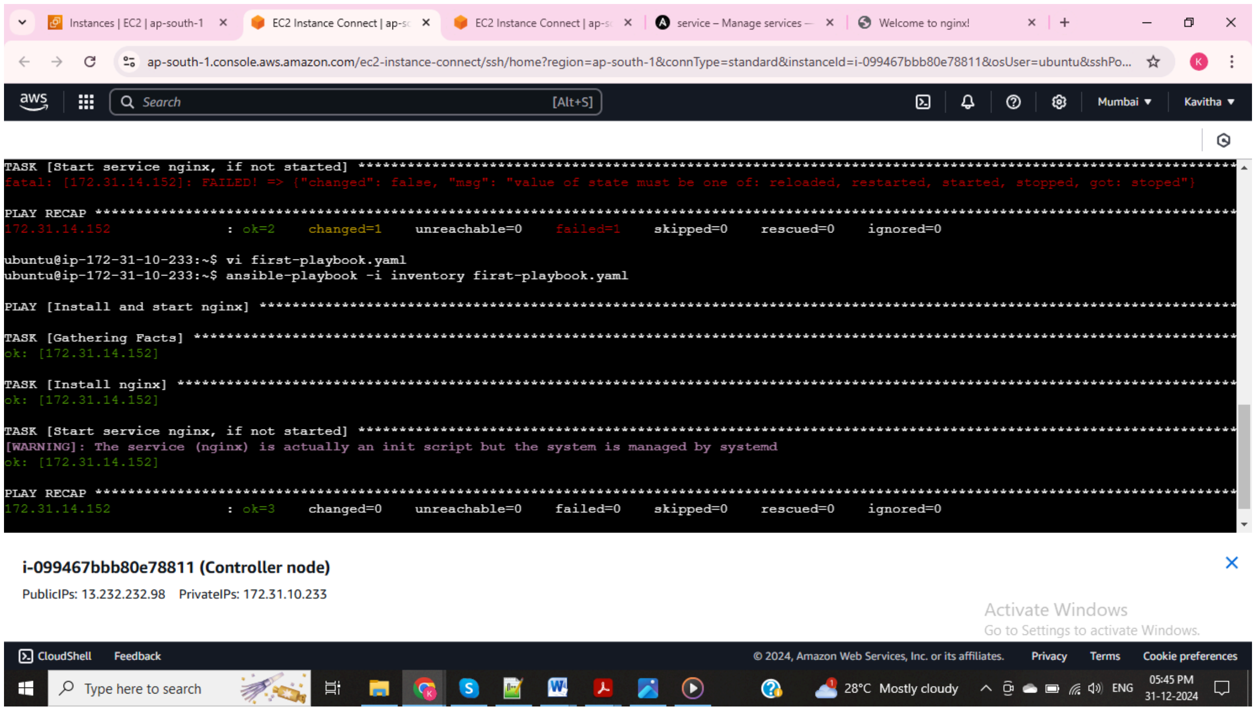


```
---
- name: Install and start nginx
  hosts: all
  become: true

  tasks:
   - name: Install nginx
     apt:
       name: nginx
       state: absent

   - name: Start service nginx, if not started
     service:
       name: nginx
       state: stopped
~
~
~
~
~
~
~
~
"first-playbook.yaml" 15L, 261B                                                  15,19          All
```

i-099467bbb80e78811 (Controller node)

PublicIPs: 13.232.232.98 PrivateIPs: 172.31.10.233

Ansible-playbook is running successfully and nginx application is stopped

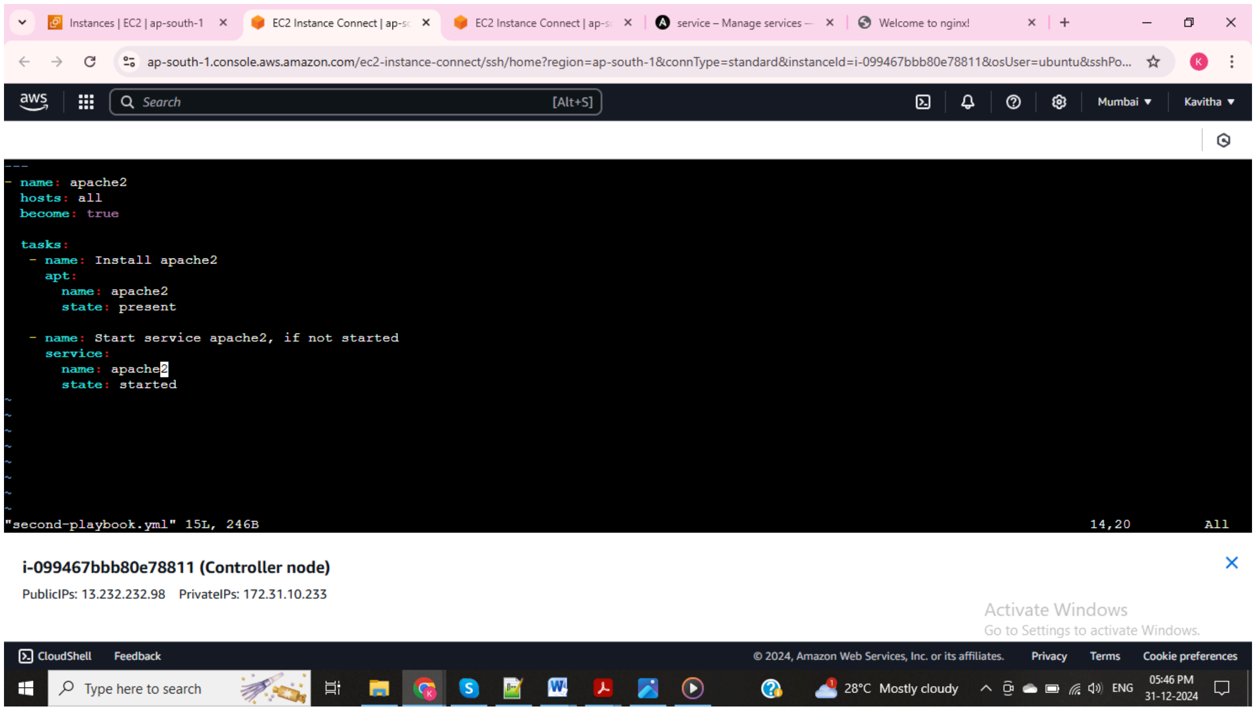

```
TASK [Start service nginx, if not started] *************************************************************************************
fatal: [172.31.14.152]: FAILED! => {"changed": false, "msg": "value of state must be one of: reloaded, restarted, started, stopped, got: stoped"}

PLAY RECAP *********************************************************************************************************************
172.31.14.152              : ok=2    changed=1    unreachable=0    failed=1    skipped=0    rescued=0    ignored=0

ubuntu@ip-172-31-10-233:~$ vi first-playbook.yaml
ubuntu@ip-172-31-10-233:~$ ansible-playbook -i inventory first-playbook.yaml

PLAY [Install and start nginx] *************************************************************************************************

TASK [Gathering Facts] *********************************************************************************************************
ok: [172.31.14.152]

TASK [Install nginx] ***********************************************************************************************************
ok: [172.31.14.152]

TASK [Start service nginx, if not started] *************************************************************************************
[WARNING]: The service (nginx) is actually an init script but the system is managed by systemd
ok: [172.31.14.152]

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

i-099467bbb80e78811 (Controller node)

PublicIPs: 13.232.232.98 PrivateIPs: 172.31.10.233

Instances | EC2 | ap-south-1
EC2 Instance Connect | ap-s
EC2 Instance Connect | ap-s
service – Manage services –
Welcome to nginx!
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?region=ap-south-1&connType=standard&instanceId=i-099467bbb80e78811&osUser=ubuntu&sshPo...
Search [Alt+S]
Mumbai
Kavitha
Activate Windows
Go to Settings to activate Windows.
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
28°C Mostly cloudy
ENG
05:45 PM
31-12-2024

Creating the another playbook to check second application is working in same slave server

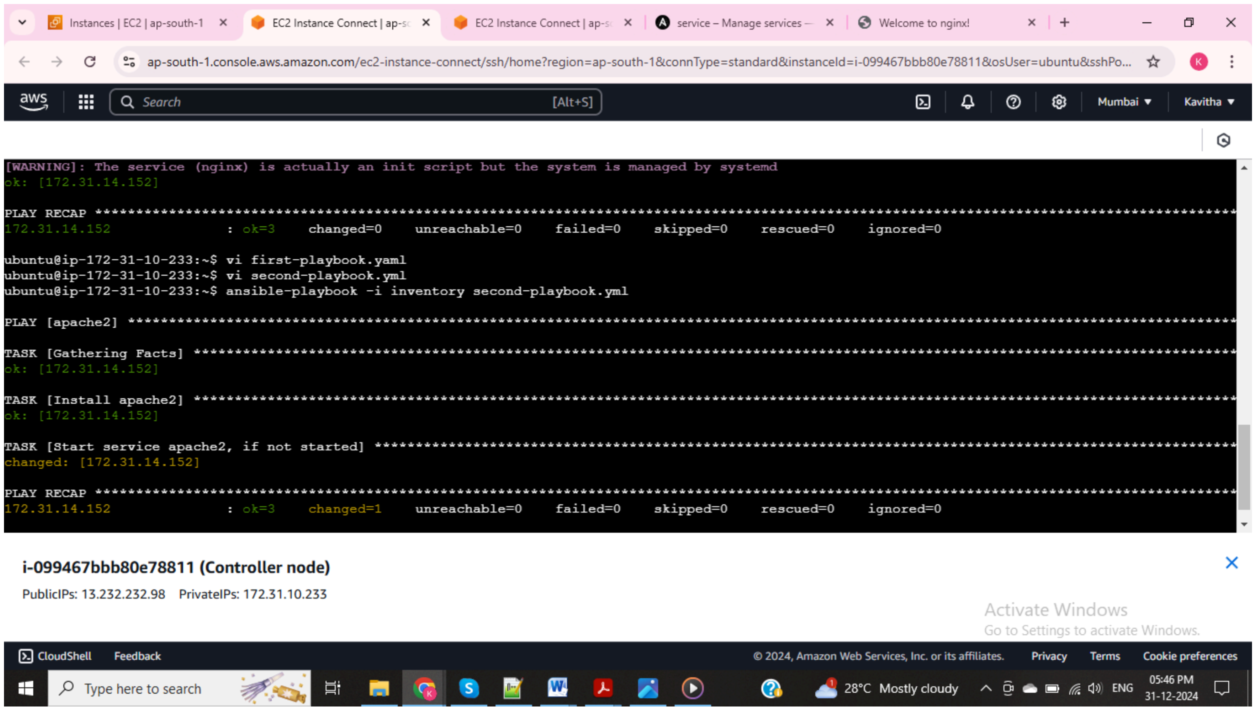

```
---
- name: apache2
  hosts: all
  become: true

  tasks:
   - name: Install apache2
     apt:
       name: apache2
       state: present

   - name: Start service apache2, if not started
     service:
       name: apache2
       state: started
~
~
~
~
~
~
~
~
"second-playbook.yml" 15L, 246B                                                                                14,20         All
```

i-099467bbb80e78811 (Controller node)

PublicIPs: 13.232.232.98 PrivateIPs: 172.31.10.233

Instances | EC2 | ap-south-1
EC2 Instance Connect | ap-s
EC2 Instance Connect | ap-s
service – Manage services –
Welcome to nginx!
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh/home?region=ap-south-1&connType=standard&instanceId=i-099467bbb80e78811&osUser=ubuntu&sshPo...
Search [Alt+S]
Mumbai
Kavitha
Activate Windows
Go to Settings to activate Windows.
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
28°C Mostly cloudy
ENG
05:46 PM
31-12-2024

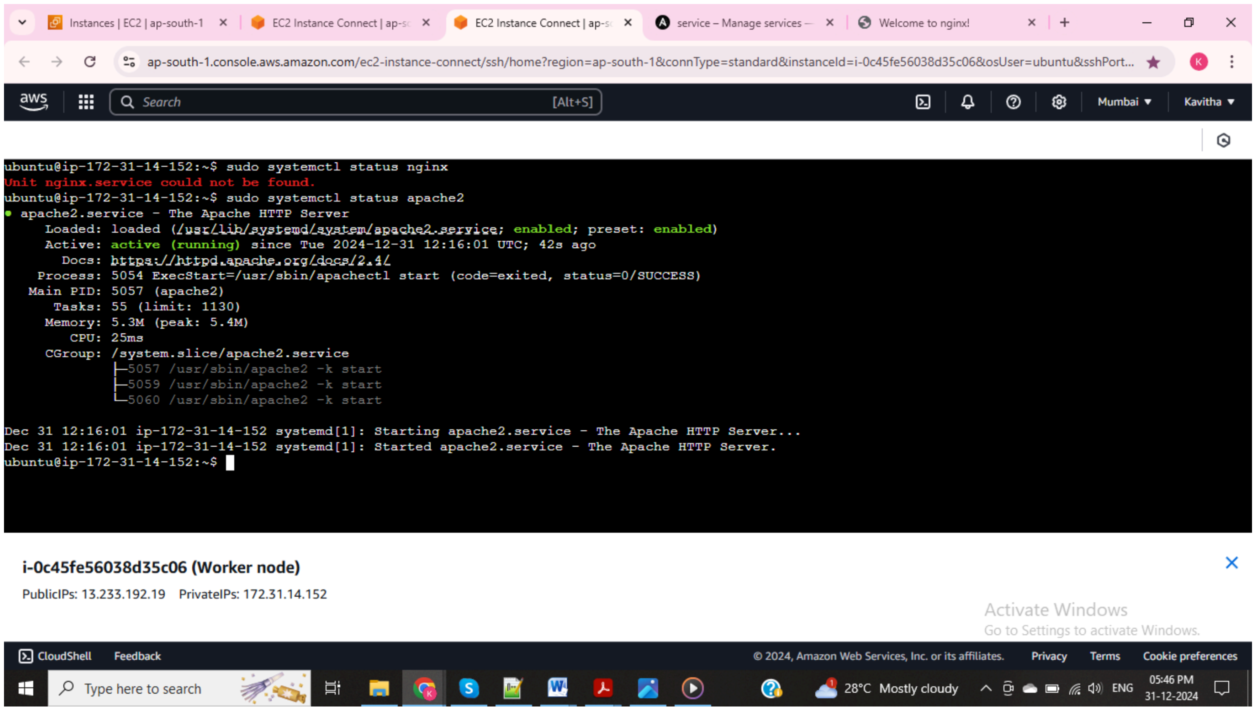
Ansible-playbook -i inventory second-playbook.yaml is success



```
[WARNING]: The service (nginx) is actually an init script but the system is managed by systemd
ok: [172.31.14.152]

PLAY RECAP ***************************************************************************************************************
172.31.14.152              : ok=3    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ubuntu@ip-172-31-10-233:~$ vi first-playbook.yaml
ubuntu@ip-172-31-10-233:~$ vi second-playbook.yml
ubuntu@ip-172-31-10-233:~$ ansible-playbook -i inventory second-playbook.yml

PLAY [apache2] ***********************************************************************************************************

TASK [Gathering Facts] ***************************************************************************************************
ok: [172.31.14.152]

TASK [Install apache2] ***************************************************************************************************
ok: [172.31.14.152]

TASK [Start service apache2, if not started] *****************************************************************************
changed: [172.31.14.152]

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

i-099467bbb80e78811 (Controller node)
PublicIPs: 13.232.232.98 PrivateIPs: 172.31.10.233

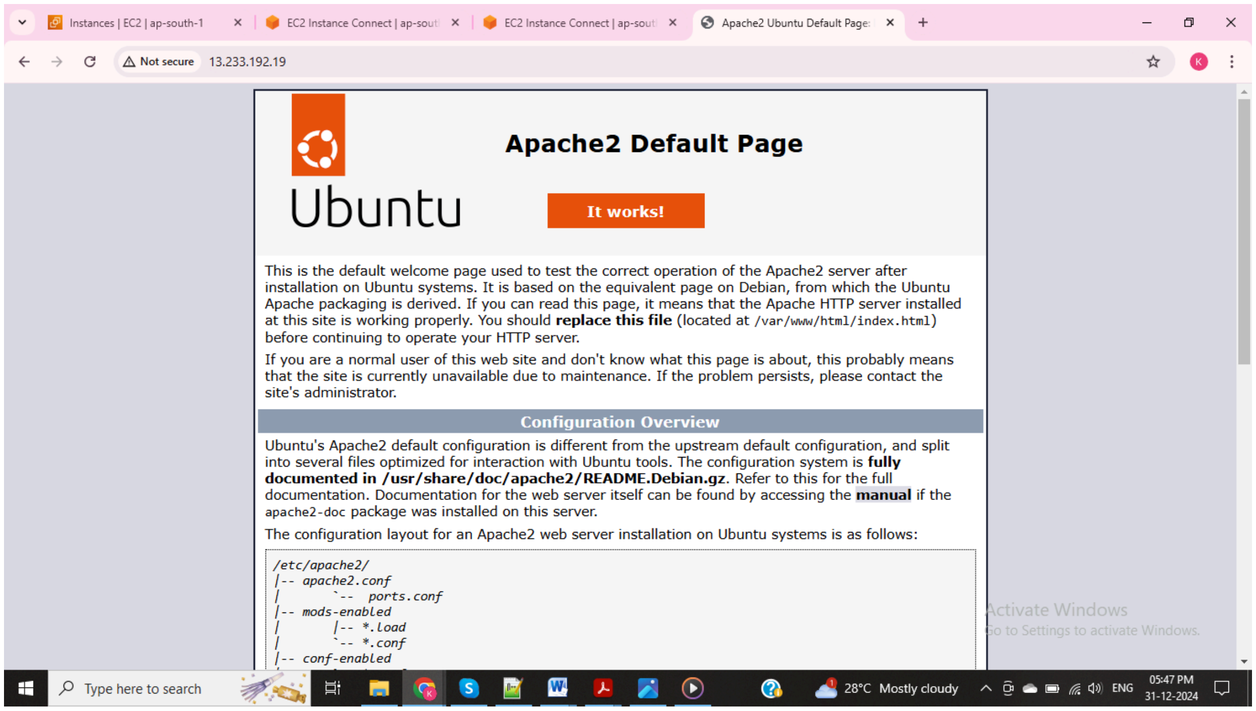
The output for stopped the previous application NGINX and running the current application APACHE2 and its status is active



```
ubuntu@ip-172-31-14-152:~$ sudo systemctl status nginx
Unit nginx.service could not be found.
ubuntu@ip-172-31-14-152:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
     Active: active (running) since Tue 2024-12-31 12:16:01 UTC; 42s ago
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 5054 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 5057 (apache2)
      Tasks: 55 (limit: 1130)
     Memory: 5.3M (peak: 5.4M)
        CPU: 25ms
     CGroup: /system.slice/apache2.service
             ├─5057 /usr/sbin/apache2 -k start
             ├─5059 /usr/sbin/apache2 -k start
             └─5060 /usr/sbin/apache2 -k start

Dec 31 12:16:01 ip-172-31-14-152 systemd[1]: Starting apache2.service - The Apache HTTP Server...
Dec 31 12:16:01 ip-172-31-14-152 systemd[1]: Started apache2.service - The Apache HTTP Server.
ubuntu@ip-172-31-14-152:~$
```

i-0c45fe56038d35c06 (Worker node)
PublicIPs: 13.233.192.19 PrivateIPs: 172.31.14.152

OUTPUT: Apache2 is running in Slave server


Instances | EC2 | ap-south-1
EC2 Instance Connect | ap-sout
EC2 Instance Connect | ap-sout
Apache2 Ubuntu Default Page:
Not secure
13.233.192.19
Type here to search
28°C Mostly cloudy
ENG
05:47 PM
31-12-2024
Activate Windows
Go to Settings to activate Windows.

Ubuntu

Apache2 Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|       `--  ports.conf
|-- mods-enabled
|       |-- *.load
|       `-- *.conf
|-- conf-enabled
```

Commands used in Ansible server:

1. sudo apt update
2. sudo apt install ansible
3. Ansible --version
4. ssh-keygen
5. cd /home/ubuntu/.ssh
6. cat (public key number)
7. ssh (slave server private IP address)
8. exit (from slave server)

Ad-hoc commands:

1. ansible -i inventory all -m “shell” -a “touch slave file”
2. ansible -i inventory all -m “shell” -a “df -h”
3. ansible -i inventory all -m “shell” -a “nproc”

Ansible playbook commands:

1. vi first-playbook.yaml
2. ansible-playbook -i inventory first-playbook.yaml
3. Ansible-playbook -i inventory -v first-playbook.yaml