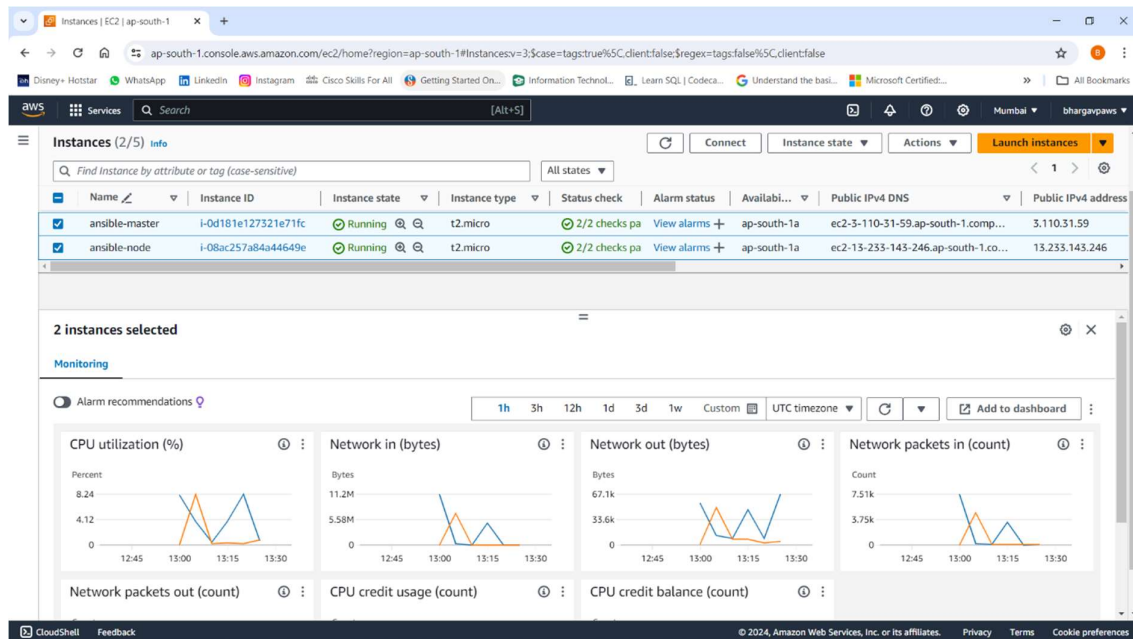


Step 1: Launching Instances

1. **Sign into your AWS account.**
2. **Launch two Ubuntu instances:**
 - Name one instance as "Ansible Master".
 - Name the other as "Ansible Node".



Step 2: Initial Setup

1. **SSH into both instances** using their respective public IP addresses. (Using MobaXterm)
2. Run the following commands on both instances to switch to the root user and update the package lists:

```
sudo su
```

```
apt update -y
```

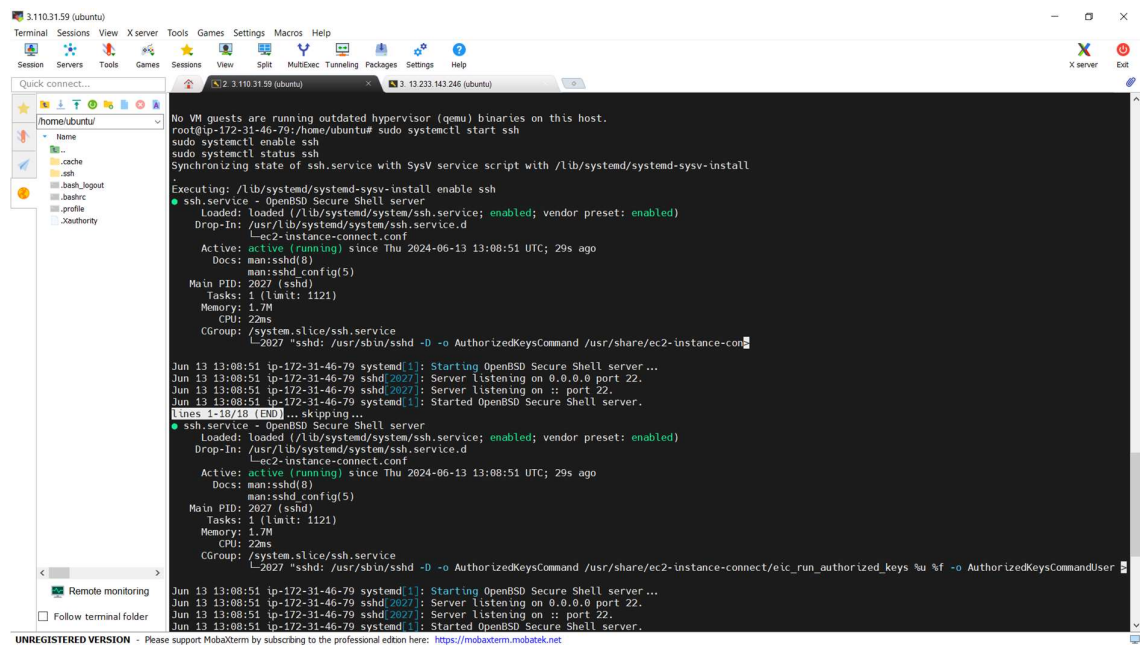
Step 3: Install and Configure SSH

1. Install OpenSSH Server on both instances:

```
sudo apt install openssh-server -y
```

2. Start and enable the SSH service:

```
sudo systemctl start ssh
sudo systemctl enable ssh
sudo systemctl status ssh
```



The screenshot shows a terminal window with the following output:

```
31103159 (ubuntu)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split Multitask Tunneling Packages Settings Help
Quick connect...
home/ubuntu/
Name
.cache
.ssh
.bash_logout
.bashrc
.profile
.sshd
No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-46-79:/home/ubuntu# sudo systemctl enable ssh
sudo systemctl start ssh
sudo systemctl status ssh
Synchronizing state of ssh.service with SysV service script with /lib/systemd/systemd-sysv-install
Executing: /lib/systemd/systemd-sysv-install enable ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
   Drop-In: /usr/lib/systemd/system/ssh.service.d
            └─ec2-instance-connect.conf
   Active: active (running) since Thu 2024-06-13 13:08:51 UTC; 29s ago
     Docs: man:ssh(8)
           man:ssh_config(5)
   Main PID: 2027 (sshd)
     Tasks: 1 (limit: 1121)
    Memory: 1.7M
         CPU: 22ms
   CGroup: /system.slice/ssh.service
           └─2027 "sshd: /usr/sbin/sshd -D -o AuthorizedKeysCommand /usr/share/ec2-instance-con

Jun 13 13:08:51 ip-172-31-46-79 systemd[1]: Starting OpenBSD Secure Shell server ...
Jun 13 13:08:51 ip-172-31-46-79 sshd[2027]: Server listening on 0.0.0.0 port 22.
Jun 13 13:08:51 ip-172-31-46-79 sshd[2027]: Server listening on :: port 22.
Jun 13 13:08:51 ip-172-31-46-79 systemd[1]: Started OpenBSD Secure Shell server.
Times: 1s 10/15 (END) vs. skipping...
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
   Drop-In: /usr/lib/systemd/system/ssh.service.d
            └─ec2-instance-connect.conf
   Active: active (running) since Thu 2024-06-13 13:08:51 UTC; 29s ago
     Docs: man:ssh(8)
           man:ssh_config(5)
   Main PID: 2027 (sshd)
     Tasks: 1 (limit: 1121)
    Memory: 1.7M
         CPU: 22ms
   CGroup: /system.slice/ssh.service
           └─2027 "sshd: /usr/sbin/sshd -D -o AuthorizedKeysCommand /usr/share/ec2-instance-connect/eic_run_authorized_keys %u %f -o AuthorizedKeysCommandUser

Jun 13 13:08:51 ip-172-31-46-79 systemd[1]: Starting OpenBSD Secure Shell server ...
Jun 13 13:08:51 ip-172-31-46-79 sshd[2027]: Server listening on 0.0.0.0 port 22.
Jun 13 13:08:51 ip-172-31-46-79 sshd[2027]: Server listening on :: port 22.
Jun 13 13:08:51 ip-172-31-46-79 systemd[1]: Started OpenBSD Secure Shell server.
UNREGISTERED VERSION - Please support MobaxTerm by subscribing to the professional edition here: https://mobaxterm.mobatek.net
```

Step 4: Create and Configure Ansible User

1. Create a user named **ansible** on both machines:

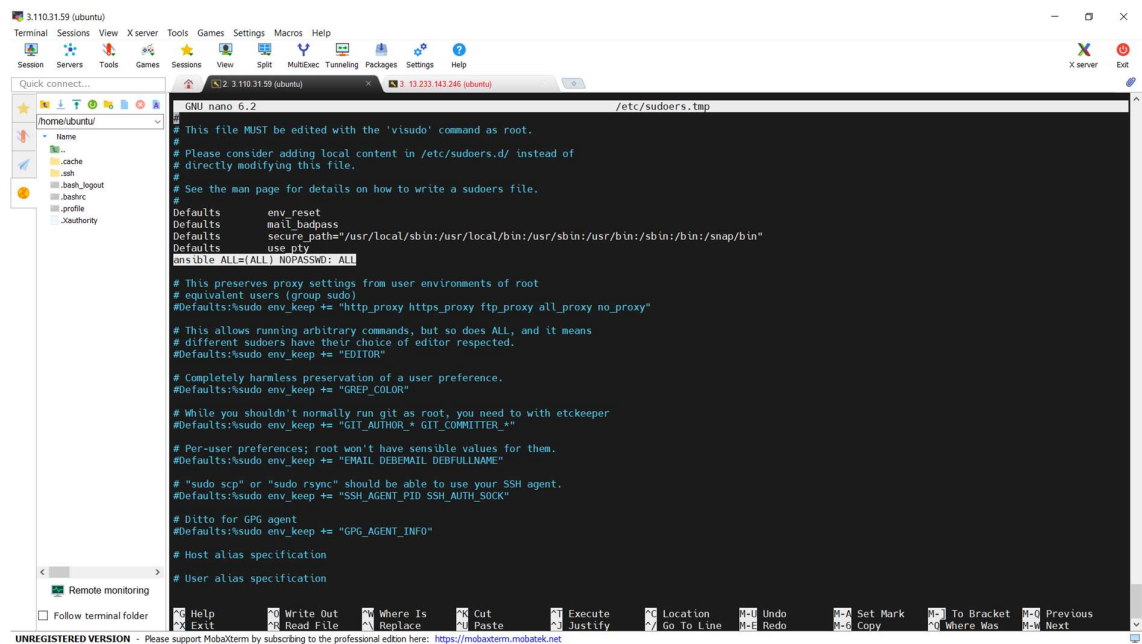
`adduser ansible`

2. Grant **sudo** privileges to **ansible**:

`visudo`

Add the following line to the file:

`ansible ALL=(ALL) NOPASSWD: ALL`



```
GNU nano 6.2 /etc/sudoers.tmp
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"
Defaults        use_pty
ansible ALL=(ALL) NOPASSWD: ALL

# This preserves proxy settings from user environments of root
# equivalent users (group sudo)
#Defaults:sudo env_keep += "http_proxy https_proxy ftp_proxy all_proxy no_proxy"

# This allows running arbitrary commands, but so does ALL, and it means
# different sudoers have their choice of editor respected.
#Defaults:sudo env_keep += "EDITOR"

# Completely harmless preservation of a user preference.
#Defaults:sudo env_keep += "GREP_COLOR"

# While you shouldn't normally run git as root, you need to with etckeeper
#Defaults:sudo env_keep += "GIT_AUTHOR_* GIT_COMMITTER_*"

# Per-user preferences; root won't have sensible values for them.
#Defaults:sudo env_keep += "EMAIL DEBEMAIL DEBFULLNAME"

# "sudo scp" or "sudo rsync" should be able to use your SSH agent.
#Defaults:sudo env_keep += "SSH_AGENT_PID SSH_AUTH_SOCK"

# Ditto for GPG agent
#Defaults:sudo env_keep += "GPG_AGENT_INFO"

# Host alias specification

# User alias specification
```

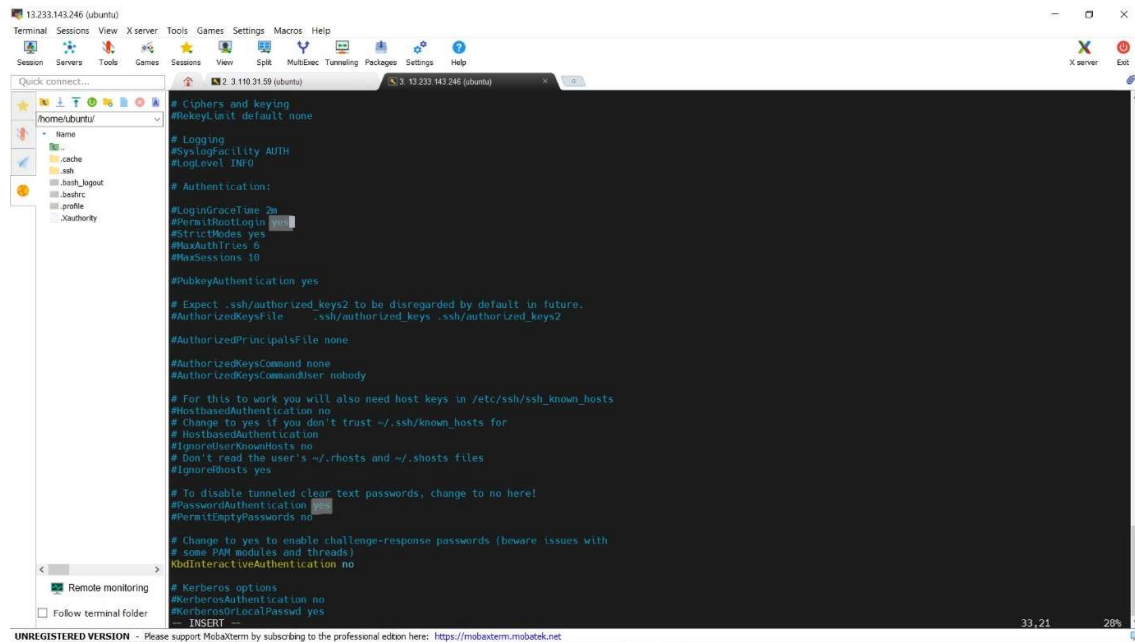
Save and exit the file by pressing **Ctrl+X** & **Ctrl+Y** & **Enter**.

Paste the copied public key into the file, save, and exit.(:wq & Enter)

Step 6: Adjust SSH Configuration

1. Edit the SSH configuration file on both instances:

```
sudo vim /etc/ssh/sshd_config
```



```
# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin yes
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile .ssh/authorized_keys .ssh/authorized_keys2

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

# To disable tunneled clear text passwords, change to no here!
#PasswordAuthentication yes
#PermitEmptyPasswords no

# Change to yes to enable challenge-response passwords (beware issues with
# some PAM modules and threads)
#KbdInteractiveAuthentication no

# Kerberos options
#KerberosAuthentication no
#KerberosOrLocalPasswd yes
#KerberosTicketCleanup yes
#KerberosLogLevel none
```

2. Uncomment and set the following parameters:

```
PermitRootLogin yes
PasswordAuthentication yes
```

Save and exit the file.

3. Restart the SSH service:

```
sudo service ssh restart
```

Step 7: Test SSH Connection

1. Switch to the ansible user on the Ansible Server:

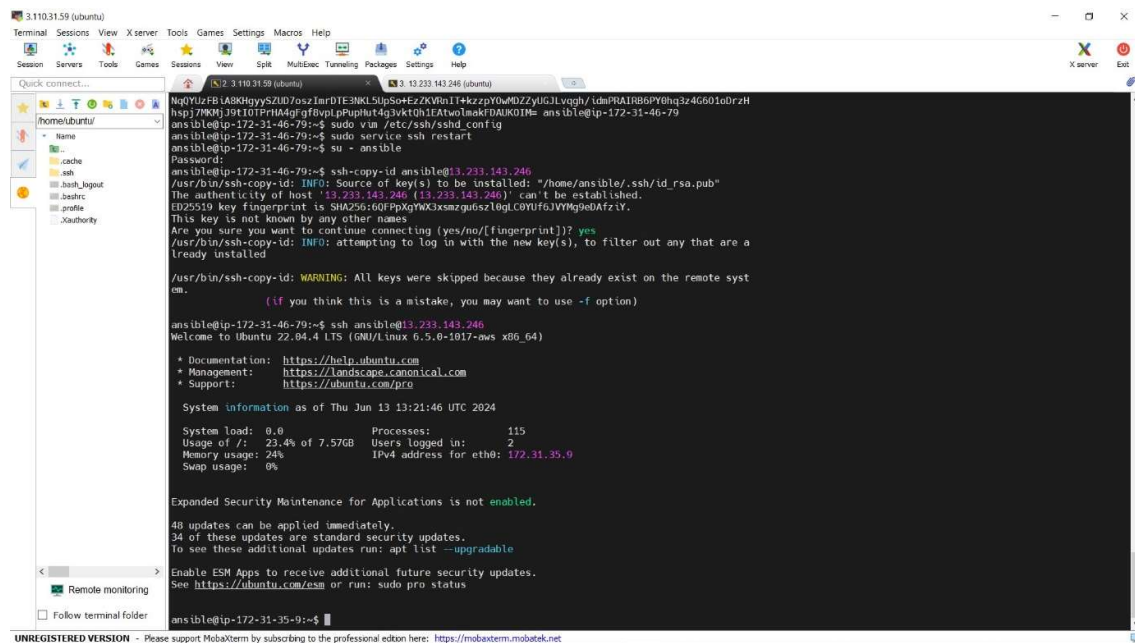
```
su - ansible
```

2. Copy the SSH key to the Ansible Node:

```
ssh-copy-id ansible@<Ansible_Node_IP>  
ssh-copy-id ansible@13.233.143.246
```

3. Test the SSH connection:

```
ssh ansible@13.233.143.246
```



```
3.110.31.59 (ubuntu)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split Multiterm Tunneling Packages Settings Help

Quick connect...
/home/ubuntu/
  Name
  cache
  ash
  ash_logout
  ashrc
  profile
  .authority

3.110.31.59 (ubuntu) 3.13.233.143.246 (ubuntu)
Ng0YUzFBIA8KHgyYSZUD7oszImrDIE3NKL5UpSo+eZKvWnIT+krzpy0wMDZZyUGJLvqgh/ldmPRAIRB6PY6hq3z4G601oDrZH
hspj7MR0J3H10TFPMAGrGfovpLpPupht4q3Wkth1EAtwoIeakRFAUK0IH= ansible@ip-172-31-46-79
ansible@ip-172-31-46-79:~$ sudo vim /etc/ssh/sshd_config
ansible@ip-172-31-46-79:~$ sudo service ssh restart
ansible@ip-172-31-46-79:~$ su - ansible
ansible@ip-172-31-46-79:~$ ssh-copy-id ansible@13.233.143.246
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/ansible/.ssh/id_rsa.pub"
The authenticity of host '13.233.143.246 (13.233.143.246)' can't be established.
1022519 key fingerprint is SHA256:60lPpXgYhKXsmzgu6szlbgLC0YUfoJVMqJebAfz1Y.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are a
lready installed
/usr/bin/ssh-copy-id: WARNING: All keys were skipped because they already exist on the remote syst
em.
(if you think this is a mistake, you may want to use -f option)
ansible@ip-172-31-46-79:~$ ssh ansible@13.233.143.246
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 6.5.0-1017-aws x86_64)

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

System information as of Thu Jun 13 13:21:46 UTC 2024

System load: 0.0          Processes:    115
Usage of /: 23.4% of 7.57GB Users logged in:    2
Memory usage: 24%        IPv4 address for eth0: 172.31.35.9
Swap usage: 0%

Expanded Security Maintenance for Applications is not enabled.

48 updates can be applied immediately.
34 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

ansible@ip-172-31-35-9:~$
```

You should now be able to SSH into the Ansible Node without a password.

Type exit to log out.

Step 8: Install and Configure Ansible

1. Install Ansible on the Ansible Server:

```
sudo apt install ansible -y
```

2. Create the Ansible inventory directory:

```
sudo mkdir -p /etc/ansible
```

3. Create the inventory file:

```
sudo vim /etc/ansible/inventory.ini
```

Add the following content:

```
[ubuntu_servers]
machine1 ansible_host=<Ansible_Node_IP> # Enter ansible node ip here.
```

Save and exit the file.

Step 9: Test Ansible Connection

1. Ping the Ansible Node from the Ansible Server:

```
ansible -i /etc/ansible/inventory.ini -u ansible -m ping
```

Step 10: Install Git Using Ansible

1. **Create a playbook file** named `install_git.yml`:

```
vim install_git.yml
```

2. **Add the following content** to the `install_git.yml` playbook:

```
---  
- hosts: ubuntu_servers  
  become: yes  
  tasks:  
    - name: Ensure Git is installed  
      apt:  
        name: git  
        state: present
```

3. **Run the Ansible playbook:**

```
ansible-playbook -i /etc/ansible/inventory.ini -u ansible install_git.yml
```