# Automating Infrastructure using Terraform

## Step 1: Install and set up Terraform on your local system.

1.1 Create a folder

***mkdir ec2pro***

***cd ec2pro***

1.2 Run the following command to download the appropriate package (make sure to get the latest version from [Terraform Versions | HashiCorp Releases](https://releases.hashicorp.com/terraform/))

wget https://releases.hashicorp.com/terraform/1.1.8/terraform_1.1.8_linux_amd64.zip

## Step 2: Add the binary file into the bin directory

2.1 Run the below set of commands to download, unzip, and move the terraform binary file to the **bin**directory:

***sudo apt-get install unzip***

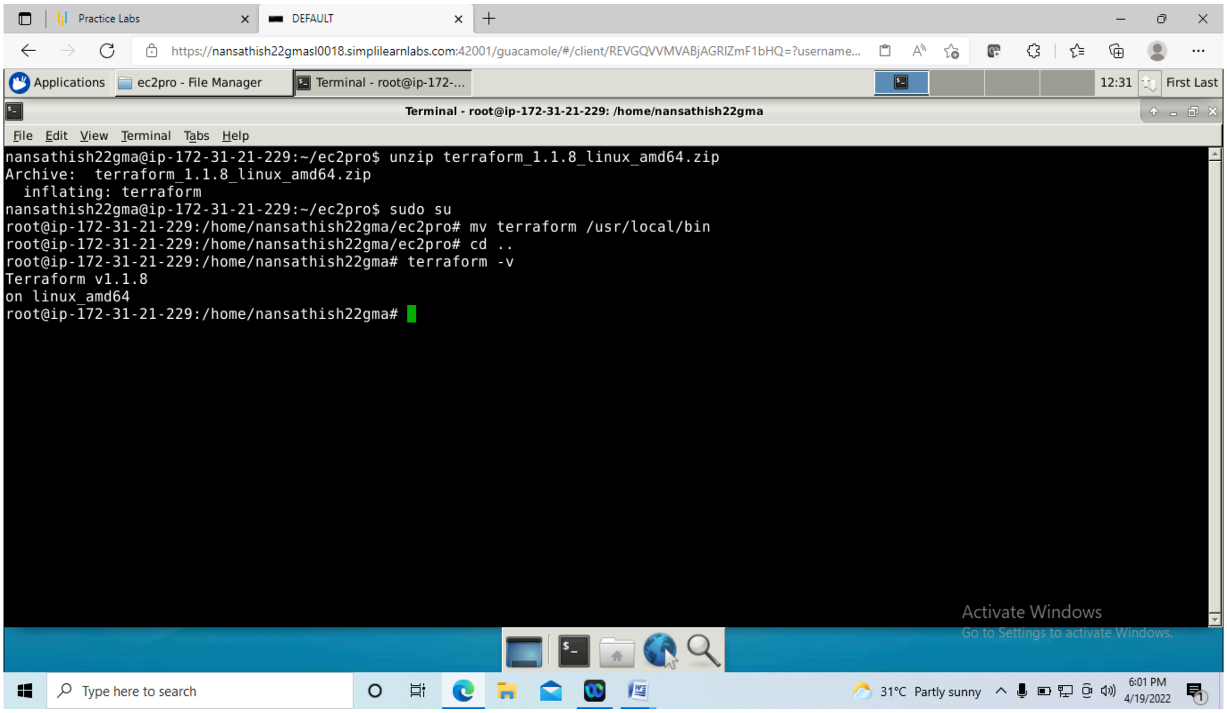
***unzip*** **terraform_1.1.8_linux_amd64.zip**

***sudo su***

***mv terraform /usr/local/bin***

***cd ..***

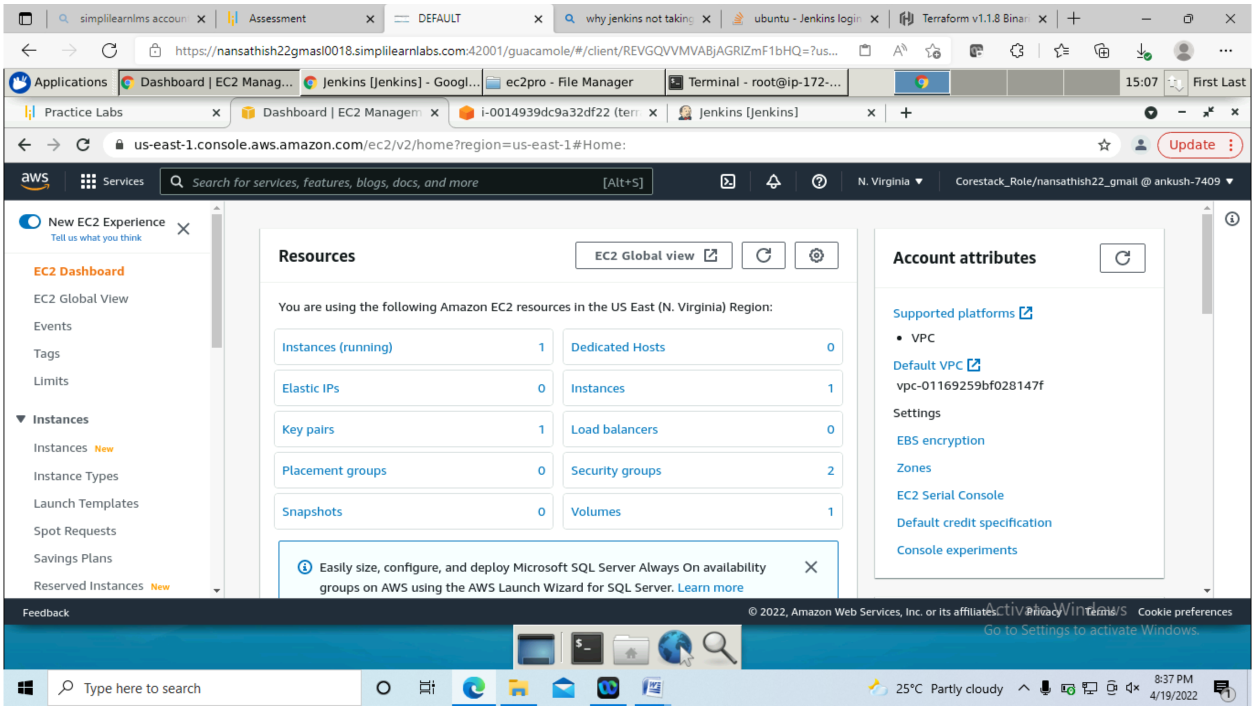
***terraform –v***


Practice Labs
DEFAULT
https://nansathish22gmasl0018.simplilearnlabs.com:42001/guacamole/#/client/REVGQVVMVABjAGRIZmF1bHQ=?username...
Applications
ec2pro - File Manager
Terminal - root@ip-172-...
12:31
First Last
Terminal - root@ip-172-31-21-229: /home/nansathish22gma
File
Edit
View
Terminal
Tabs
Help
Activate Windows
Go to Settings to activate Windows.
Type here to search
31°C Partly sunny
6:01 PM
4/19/2022

```
nansathish22gma@ip-172-31-21-229:~/ec2pro$ unzip terraform_1.1.8_linux_amd64.zip
Archive:  terraform_1.1.8_linux_amd64.zip
  inflating: terraform
nansathish22gma@ip-172-31-21-229:~/ec2pro$ sudo su
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# mv terraform /usr/local/bin
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# cd ..
root@ip-172-31-21-229:/home/nansathish22gma# terraform -v
Terraform v1.1.8
on linux_amd64
root@ip-172-31-21-229:/home/nansathish22gma#
```

## Step 3: Create an AWS EC2 instance with Terraform

3.1 Create AWS Keypair. For this step, login with your profile to the AWS Management Console, navigate to All Services → EC2:



You are using the following Amazon EC2 resources in the US East (N. Virginia) Region:

| | | | |
|---|---|---|---|
| Instances (running) | 1 | Dedicated Hosts | 0 |
| Elastic IPs | 0 | Instances | 1 |
| Key pairs | 1 | Load balancers | 0 |
| Placement groups | 0 | Security groups | 2 |
| Snapshots | 0 | Volumes | 1 |

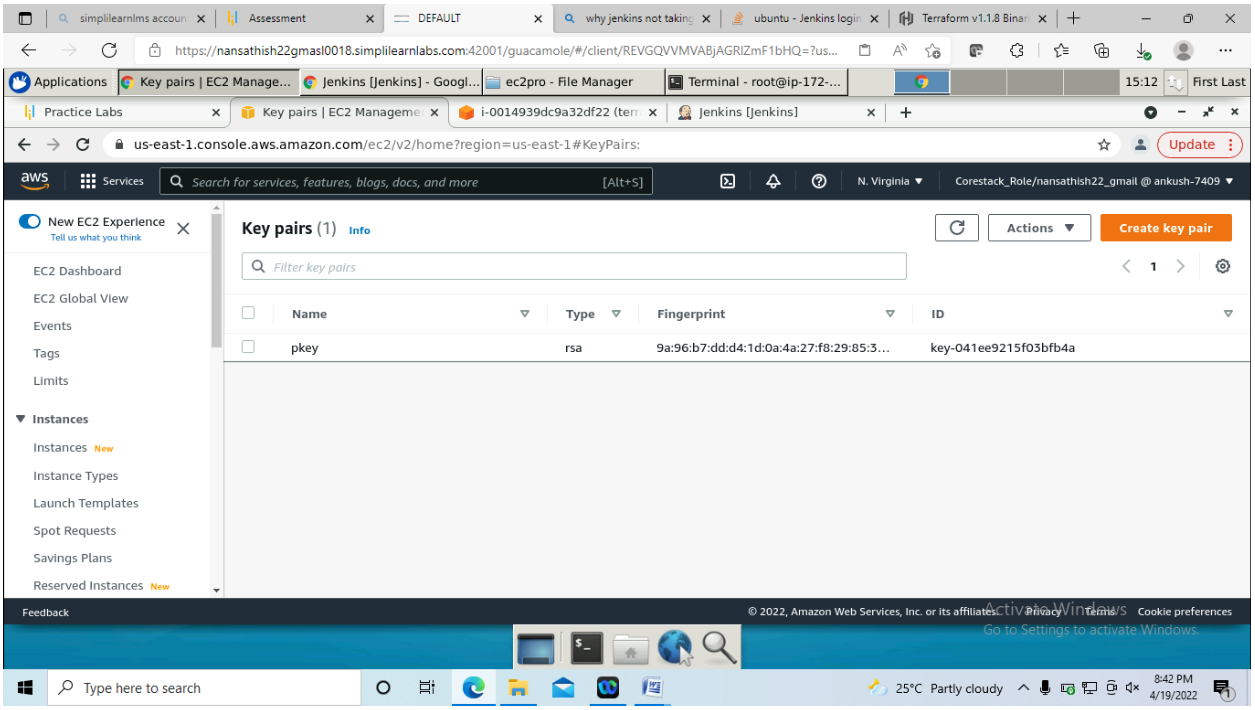
Easily size, configure, and deploy Microsoft SQL Server Always On availability groups on AWS using the AWS Launch Wizard for SQL Server. Learn more

Supported platforms
- VPC

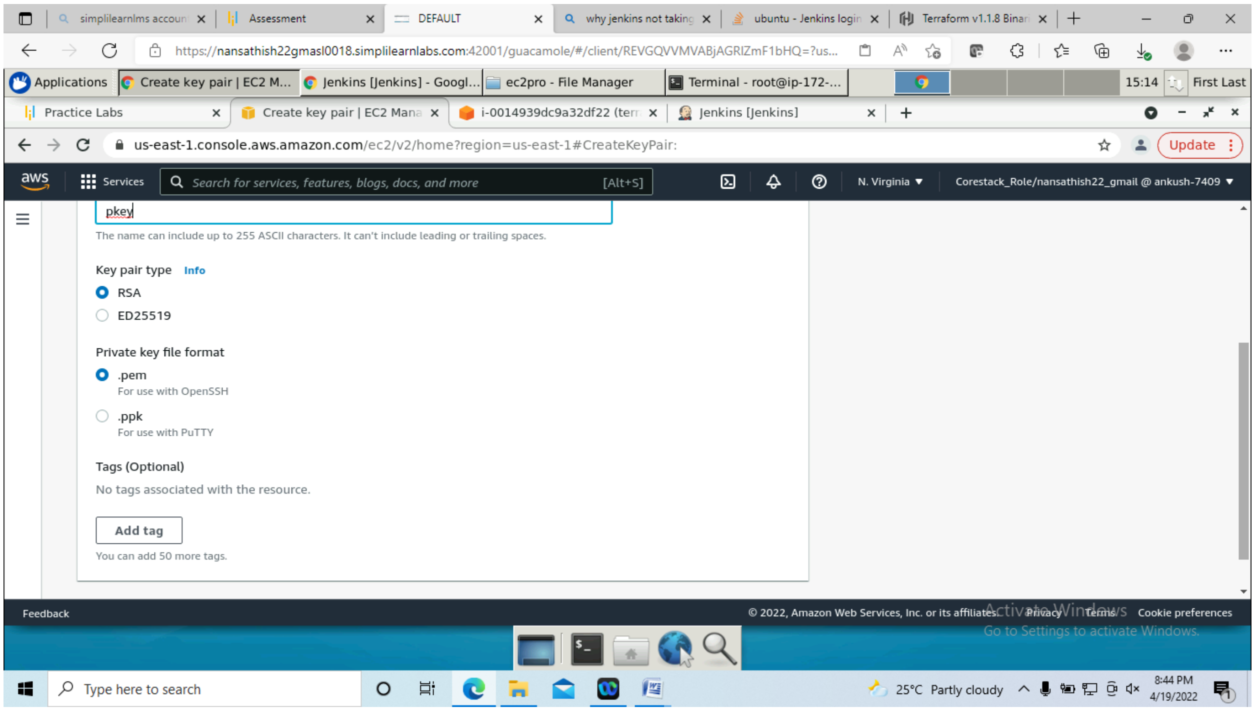
Default VPC
vpc-01169259bf028147f

Settings
EBS encryption
Zones
EC2 Serial Console
Default credit specification
Console experiments

On the next screen navigate to Network and Security ▪ Key Pairs ▪ Create key pair:



| Name | Type | Fingerprint | ID |
|---|---|---|---|
| pkey | rsa | 9a:96:b7:dd:d4:1d:0a:4a:27:f8:29:85:3... | key-041ee9215f03bfb4a |



us-east-1.console.aws.amazon.com/ec2/v2/home?region=us-east-1#CreateKeyPair:

pkey

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type Info

● RSA
○ ED25519

Private key file format

● .pem
For use with OpenSSH
○ .ppk
For use with PuTTY

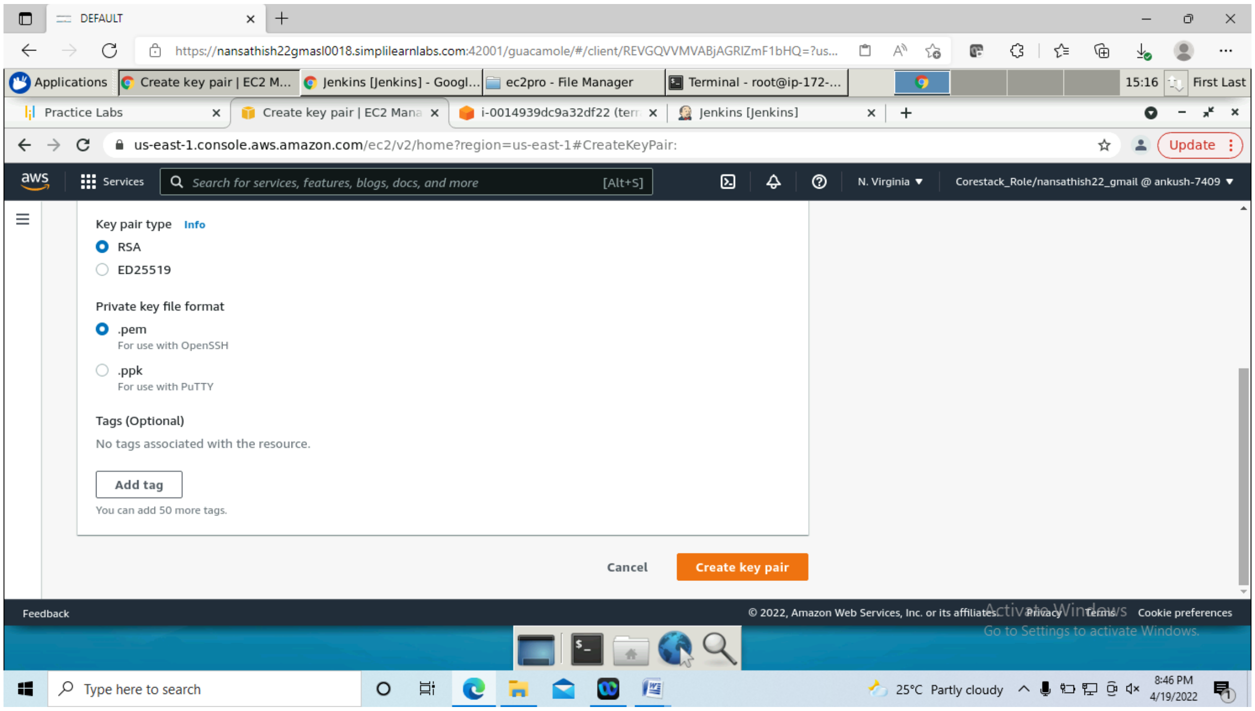
Tags (Optional)

No tags associated with the resource.

Add tag

You can add 50 more tags.

Services
Search for services, features, blogs, docs, and more [Alt+S]
N. Virginia
Corestack_Role/nansathish22_gmail @ ankush-7409
Feedback
© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
Type here to search
25°C Partly cloudy
8:44 PM
4/19/2022



us-east-1.console.aws.amazon.com/ec2/v2/home?region=us-east-1#CreateKeyPair:

Key pair type Info

● RSA
○ ED25519

Private key file format

● .pem
For use with OpenSSH
○ .ppk
For use with PuTTY

Tags (Optional)

No tags associated with the resource.

Add tag

You can add 50 more tags.

Cancel Create key pair

Services
Search for services, features, blogs, docs, and more [Alt+S]
N. Virginia
Corestack_Role/nansathish22_gmail @ ankush-7409
Feedback
© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
Type here to search
25°C Partly cloudy
8:46 PM
4/19/2022

Click on Create key pair. It will automatically download the file to your local Downloads folder.

3.2 Prepare a new terraform file for execution.

Going back to the console of your local system, navigate to your project folder and create a new Terraform file for execution:

***vim ecinput.tf***

Configure the following script:

```
terraform {
required_providers {
aws = {
      source = "hashicorp/aws"
      version = "~> 3.27"
    }
  }
required_version = ">= 0.14.9"
}
provider "aws" {
  profile = "Corestack_Role/nansathish22_gmail"
  region = "us-east-1"
access_key = "AKIA22VGYN2G2HDKC3ND"
secret_key = "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"
}
resource "aws_instance" "terraform1" {
ami                   = "ami-024fc608af8f886bc"
instance_type          = "t2.micro"
key_name               = "pkey"
vpc_security_group_ids = [aws_security_group.sg-010a62ae308d0b0af.name]
  tags = {
    Name = "terraform1"
  }
}
resource "aws_security_group" "sg-010a62ae308d0b0af" {
  name = "terra"
  ingress {
from_port   = 22
to_port     = 22
    protocol    = "tcp"
cidr_blocks = ["0.0.0.0/0"]
  }

  ingress {
from_port   = 443
to_port     = 443
    protocol    = "tcp"
cidr_blocks = ["0.0.0.0/0"]
  }

  ingress {
from_port   = 8080
to_port     = 8080
    protocol    = "tcp"
cidr_blocks = ["0.0.0.0/0"]
  }

egress {
from_port   = 0
to_port     = 0
    protocol    = -1
cidr_blocks = ["0.0.0.0/0"]
  }
  tags = {
    Name = "terra"
  }
}
```

We want to configure also an outputs file to give us the ID and public IP address of the instance, which will be used further.

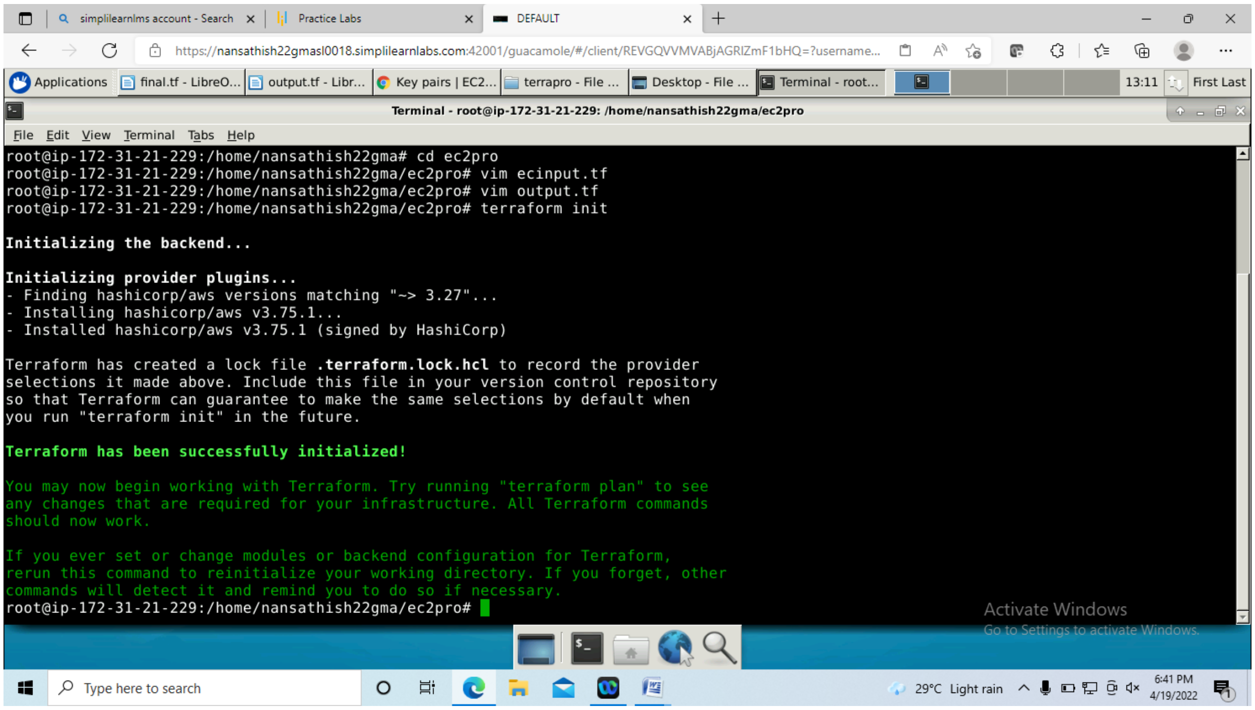
```
vim outputs.tf
```

Configure the following script:

```
output "instance_id" {
  description = "ID of the EC2 instance:"
  value       = aws_instance.terraform1.id


}
output "instance_public_ip" {
  description = "EC2 instance public IP:"
  value       = aws_instance.terraform1.public_ip
}
```

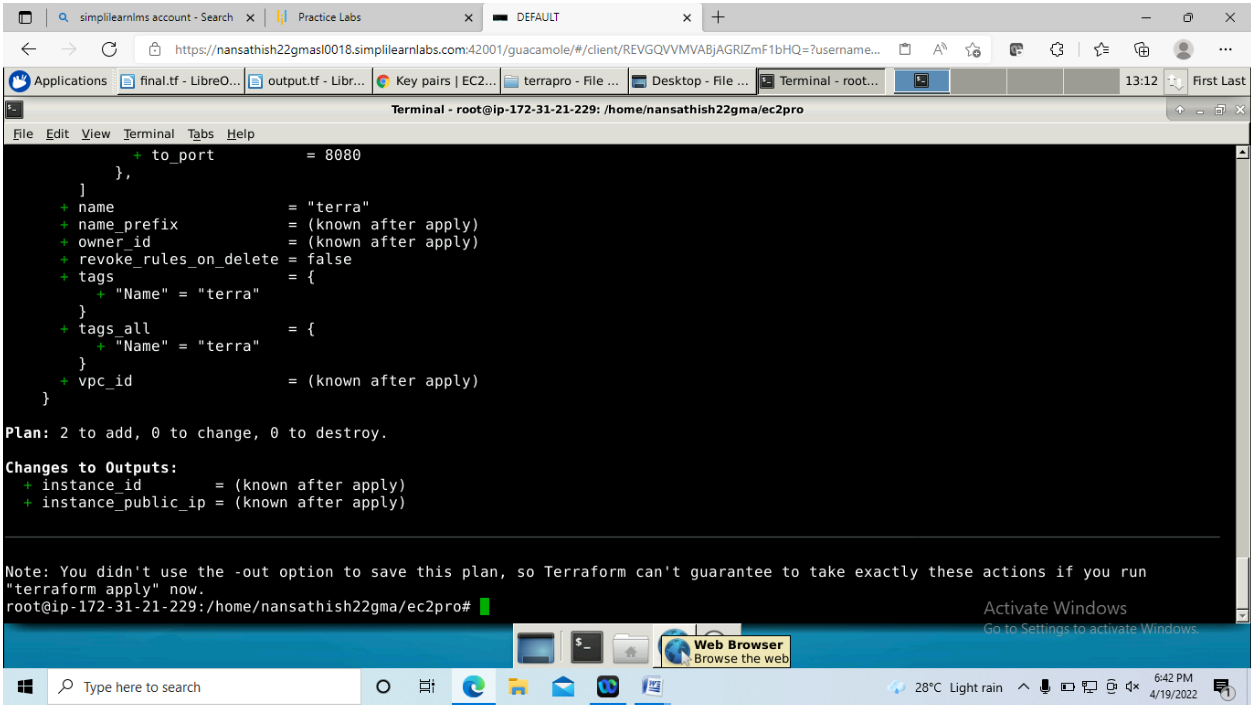
3.3 Run the next commands to create a new EC2 instance:

```
terraform init
```


simplilearnlms account - Search
Practice Labs
DEFAULT
https://nansathish22gmasl0018.simplilearnlabs.com:42001/guacamole/#/client/REVGQVVMVABjAGRlZmF1bHQ=?username...
Applications
final.tf - LibreO...
output.tf - Libr...
Key pairs | EC2...
terrapro - File ...
Desktop - File ...
Terminal - root...
13:11
First Last
Terminal - root@ip-172-31-21-229: /home/nansathish22gma/ec2pro
File Edit View Terminal Tabs Help
Activate Windows
Go to Settings to activate Windows.
Type here to search
29°C Light rain
6:41 PM
4/19/2022

```
root@ip-172-31-21-229:/home/nansathish22gma# cd ec2pro
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# vim ecinput.tf
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# vim output.tf
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# terraform init

Initializing the backend...

Initializing provider plugins...
- Finding hashicorp/aws versions matching "~> 3.27"...
- Installing hashicorp/aws v3.75.1...
- Installed hashicorp/aws v3.75.1 (signed by HashiCorp)

Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# 
```

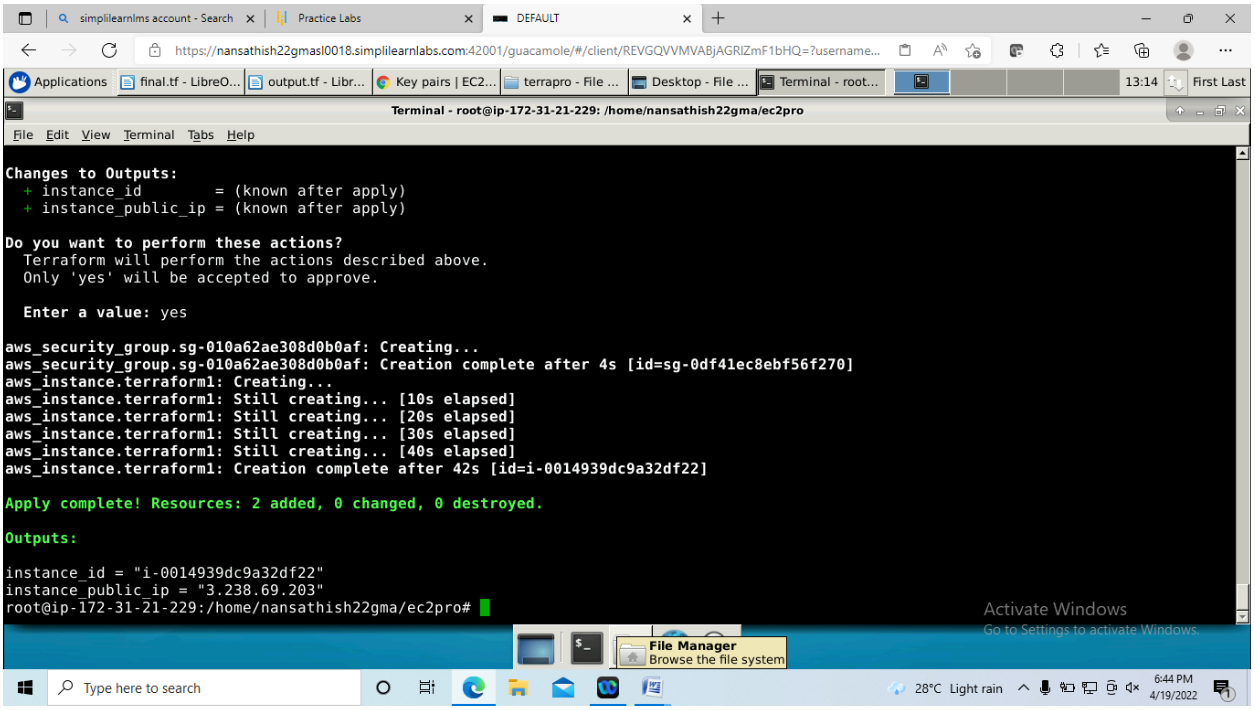
### *terraform plan*


Terminal - root@ip-172-31-21-229: /home/nansathish22gma/ec2pro
File Edit View Terminal Tabs Help
Web Browser
Browse the web
Activate Windows
Go to Settings to activate Windows.

```
              + to_port          = 8080
            },
        ]
      + name                   = "terra"
      + name_prefix            = (known after apply)
      + owner_id               = (known after apply)
      + revoke_rules_on_delete = false
      + tags                   = {
          + "Name" = "terra"
        }
      + tags_all               = {
          + "Name" = "terra"
        }
      + vpc_id                 = (known after apply)
    }

Plan: 2 to add, 0 to change, 0 to destroy.

Changes to Outputs:
  + instance_id        = (known after apply)
  + instance_public_ip = (known after apply)

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run
"terraform apply" now.
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro#
```

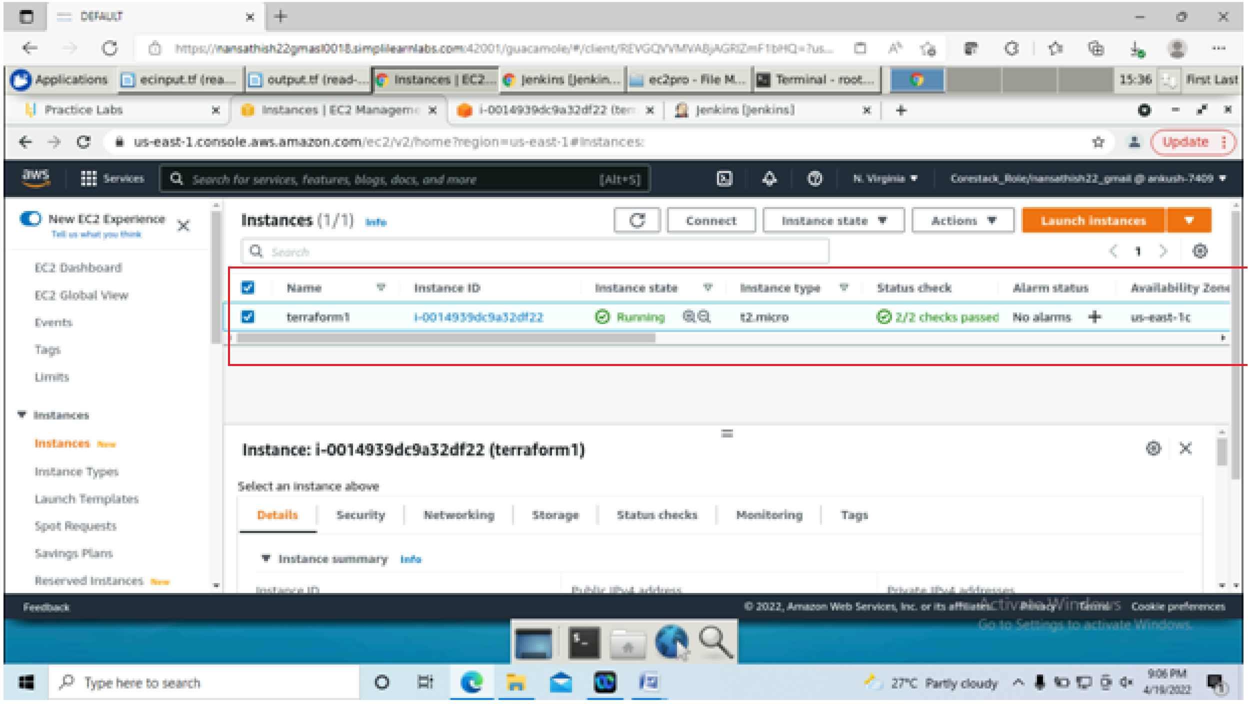
### *terraform apply*

When prompted enter a “yes” value. The result is as follows:


Terminal - root@ip-172-31-21-229: /home/nansathish22gma/ec2pro
File Edit View Terminal Tabs Help
File Manager
Browse the file system
Activate Windows
Go to Settings to activate Windows.

```
Changes to Outputs:
  + instance_id        = (known after apply)
  + instance_public_ip = (known after apply)

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_security_group.sg-010a62ae308d0b0af: Creating...
aws_security_group.sg-010a62ae308d0b0af: Creation complete after 4s [id=sg-0df41ec8ebf56f270]
aws_instance.terraform1: Creating...
aws_instance.terraform1: Still creating... [10s elapsed]
aws_instance.terraform1: Still creating... [20s elapsed]
aws_instance.terraform1: Still creating... [30s elapsed]
aws_instance.terraform1: Still creating... [40s elapsed]
aws_instance.terraform1: Creation complete after 42s [id=i-0014939dc9a32df22]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.

Outputs:

instance_id = "i-0014939dc9a32df22"
instance_public_ip = "3.238.69.203"
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro#
```

Navigate to your AWS Management Console → All services → Compute → EC2→EC2 Dashboard to review your newly created instance.



Instances (1/1)

| ☑ | Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone |
|---|---|---|---|---|---|---|---|
| ☑ | terraform1 | i-0014939dc9a32df22 | Running | t2.micro | 2/2 checks passed | No alarms | us-east-1c |

Instance: i-0014939dc9a32df22 (terraform1)

## Step 4. Establish connectivity to your AWS EC2 instance

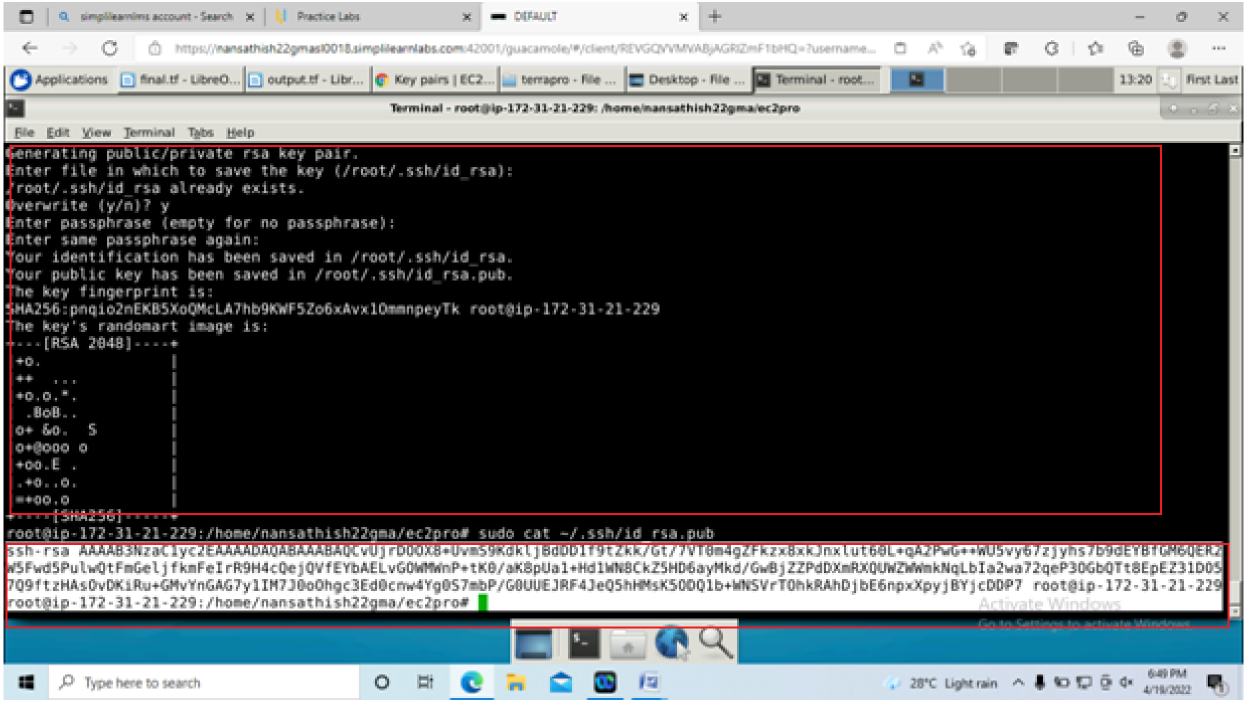
Before we proceed with Ansible execution, we want to make sure there is connectivity to our newly created EC2 instance. For this purpose, run the following command in your local system:

```
ssh-keygen
```

When prompted, push “Enter”:

And after execute the following command and copy the key:

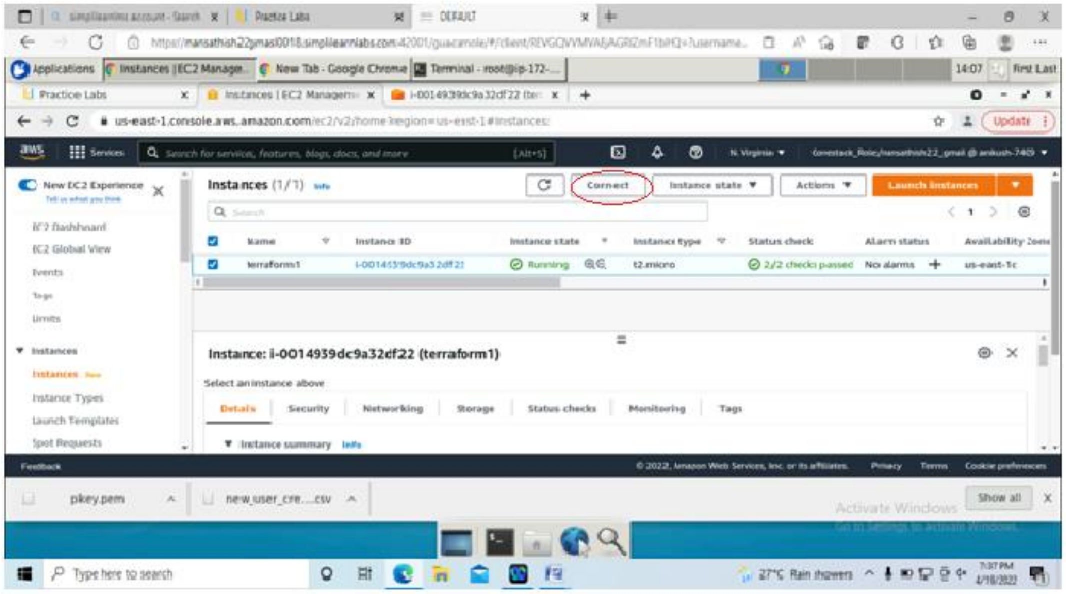
```
sudo cat ~/.ssh/id_rsa.pub
```



Terminal - root@ip-172-31-21-229: /home/nansathish22gma/ec2pro
File Edit View Terminal Tabs Help

```
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
/root/.ssh/id_rsa already exists.
Overwrite (y/n)? y
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:pnqio2nEKB5XoQMcLA7hb9KWF5Zo6xAvx1OmmnpeyTk root@ip-172-31-21-229
The key's randomart image is:
+---[RSA 2048]----+
|+o.              |
|++   ...         |
|+o.o.*.          |
| .BoB..          |
|o+ &o.   S       |
|o+@ooo o         |
|+oo.E .          |
|.+o..o.          |
|=+oo.o           |
+----[SHA256]-----+
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# sudo cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCvUjrDOOX8+UvmS9KdkljBdDD1f9tZkk/Gt/7VT0m4gZFkzx8xkJnxlut60L+qA2PwG++WU5vy67zjyhs7b9dEYBfGM6QER2
W5Fwd5PulwQtFmGeljfkmFeIrR9H4cQejQVfEYbAELvG0WMWnP+tK0/aK8pUa1+Hd1WN8CkZ5HD6ayMkd/GwBjZZPdDXmRXQUWZWWmkNqLbIa2wa72qeP30GbQTt8EpEZ31D05
7Q9ftzHAsOvDKiRu+GMvYnGAG7y1IM7J0oOhgc3Ed0cnw4Yg0S7mbP/G0UUEJRF4JeQ5hHMsK50DQ1b+WNSVrTOhkRAhDjbE6npxXpyjBYjcDDP7 root@ip-172-31-21-229
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro#
```

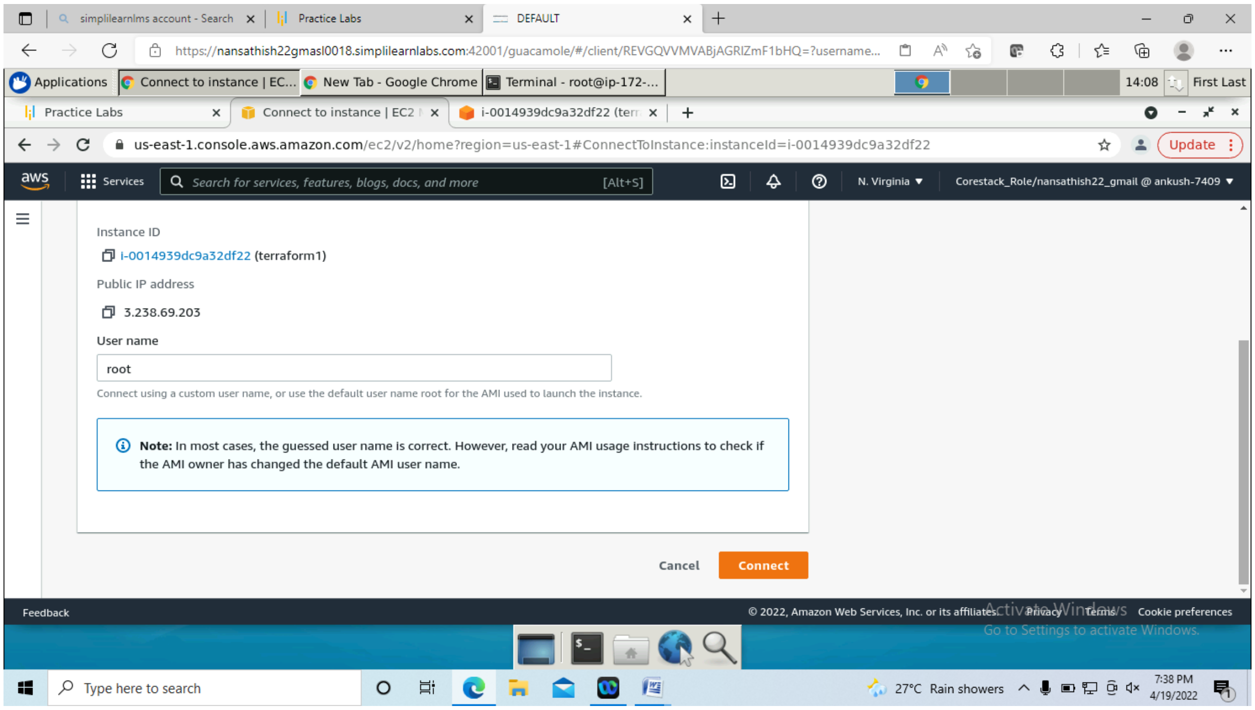
Ensure the EC2 instance allows connection from local system. For this purpose, go back to the AWS Console and connect to the instance:



| Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone |
|---|---|---|---|---|---|---|
| terraform1 | i-0014939dc9a32df22 | Running | t2.micro | 2/2 checks passed | No alarms | us-east-1c |

Instance: i-0014939dc9a32df22 (terraform1)

On the next screen, provide a user (or use the default):



Instance ID

i-0014939dc9a32df22 (terraform1)

Public IP address

3.238.69.203

User name

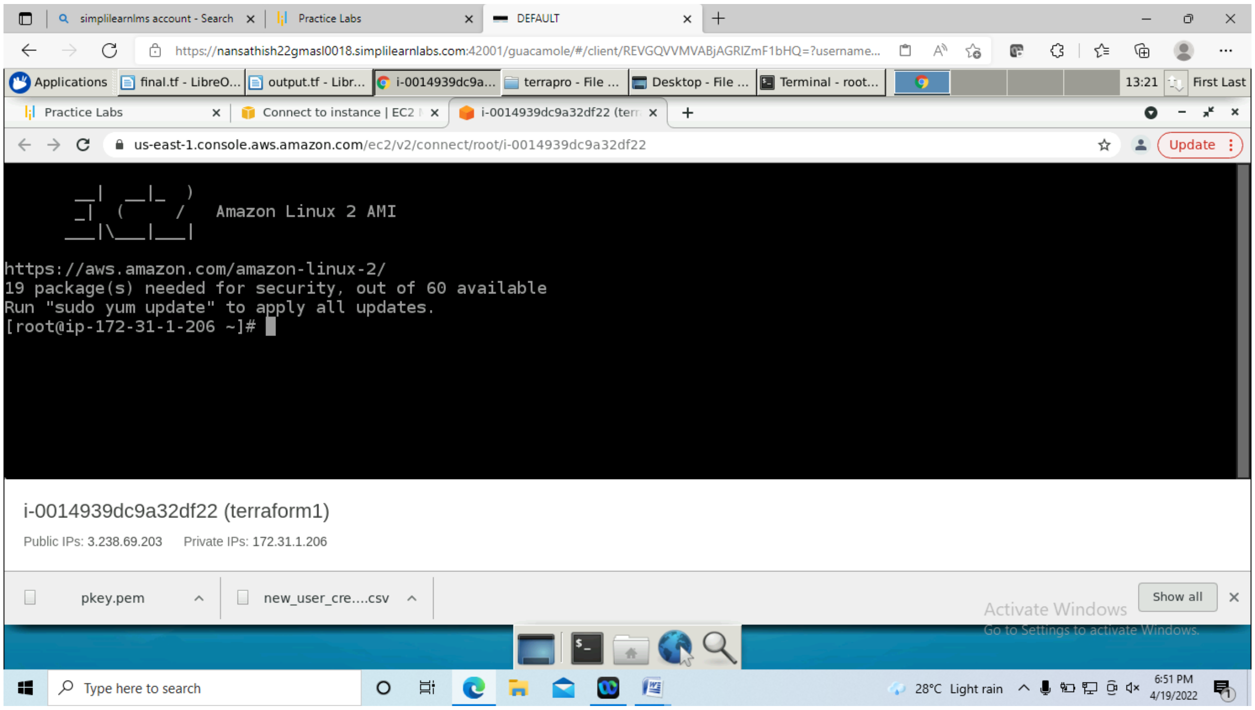
root

Connect using a custom user name, or use the default user name root for the AMI used to launch the instance.

**Note:** In most cases, the guessed user name is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.

https://nansathish22gmasl0018.simplilearnlabs.com:42001/guacamole/#/client/REVGQVVMVABjAGRIZmF1bHQ=?username...
us-east-1.console.aws.amazon.com/ec2/v2/home?region=us-east-1#ConnectToInstance:instanceId=i-0014939dc9a32df22
Cancel
Connect
Feedback
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Click “Connect”. A new console tab will be loaded with your instance:



```
       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
19 package(s) needed for security, out of 60 available
Run "sudo yum update" to apply all updates.
[root@ip-172-31-1-206 ~]# 
```

i-0014939dc9a32df22 (terraform1)

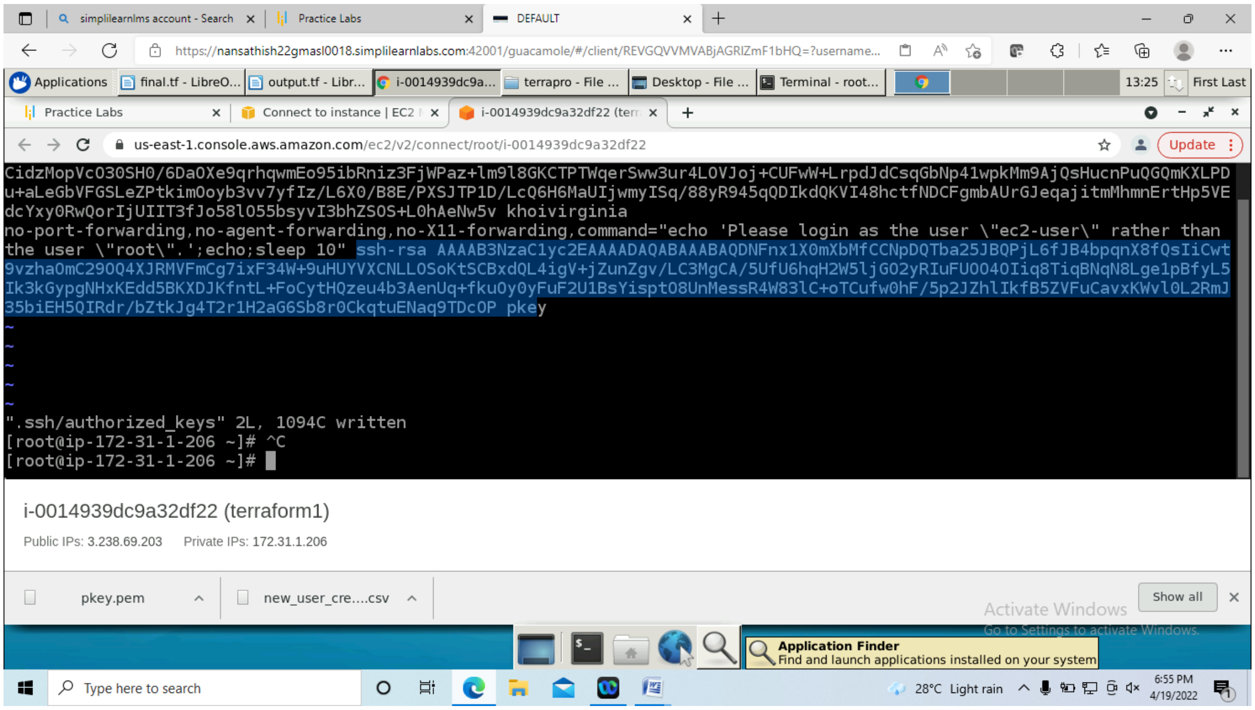
Public IPs: 3.238.69.203    Private IPs: 172.31.1.206

us-east-1.console.aws.amazon.com/ec2/v2/connect/root/i-0014939dc9a32df22
pkey.pem
new_user_cre....csv
Show all

Run the following command:

***sudo vi ~/.ssh/authorized_keys***

Copy the SSH key of your local system in this file (ESC : wq! to write and exit the file):


simplilearnlms account - Search
Practice Labs
DEFAULT
https://nansathish22gmasl0018.simplilearnlabs.com:42001/guacamole/#/client/REVGQVVMVABjAGRlZmF1bHQ=?username...
Applications
final.tf - LibreO...
output.tf - Libr...
i-0014939dc9a...
terrapro - File ...
Desktop - File ...
Terminal - root...
13:25
First Last
Practice Labs
Connect to instance | EC2
i-0014939dc9a32df22 (terr
us-east-1.console.aws.amazon.com/ec2/v2/connect/root/i-0014939dc9a32df22
Update

```
CidzMopVc030SH0/6Da0Xe9qrhqwmEo95ibRniz3FjWPaz+lm9l8GKCTPTWqerSww3ur4LOVJoj+CUFwW+LrpdJdCsqGbNp41wpkMm9AjQsHucnPuQGQmKXLPD
u+aLeGbVFGSLeZPtkimOoyb3vv7yfIz/L6X0/B8E/PXSJTP1D/LcQ6H6MaUIjwmyISq/88yR945qQDIkdQKVI48hctfNDCFgmbAUrGJeqajitmMhmnErtHp5VE
dcYxy0RwQorIjUIIT3fJo58l055bsyvI3bhZSOS+L0hAeNw5v khoivirginia
no-port-forwarding,no-agent-forwarding,no-X11-forwarding,command="echo 'Please login as the user \"ec2-user\" rather than
the user \"root\".';echo;sleep 10" ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDNFnx1X0mXbMfCCNpDQTba25JBQPjL6fJB4bpqnX8fQsIiCwt
9vzhaOmC29OQ4XJRMVFmCg7ixF34W+9uHUYVXCNLLOSoKtSCBxdQL4igV+jZunZgv/LC3MgCA/5UfU6hqH2W5ljGO2yRIuFU0O4OIiq8TiqBNqN8Lge1pBfyL5
Ik3kGypgNHxKEdd5BKXDJKfntL+FoCytHQzeu4b3AenUq+fku0y0yFuF2U1BsYispt08UnMessR4W83lC+oTCufw0hF/5p2JZhlIkfB5ZVFuCavxKWvl0L2RmJ
35biEH5QIRdr/bZtkJg4T2r1H2aG6Sb8r0CkqtuENaq9TDcOP pkey
~
~
~
~
~
".ssh/authorized_keys" 2L, 1094C written
[root@ip-172-31-1-206 ~]# ^C
[root@ip-172-31-1-206 ~]#
```

i-0014939dc9a32df22 (terraform1)

Public IPs: 3.238.69.203 Private IPs: 172.31.1.206

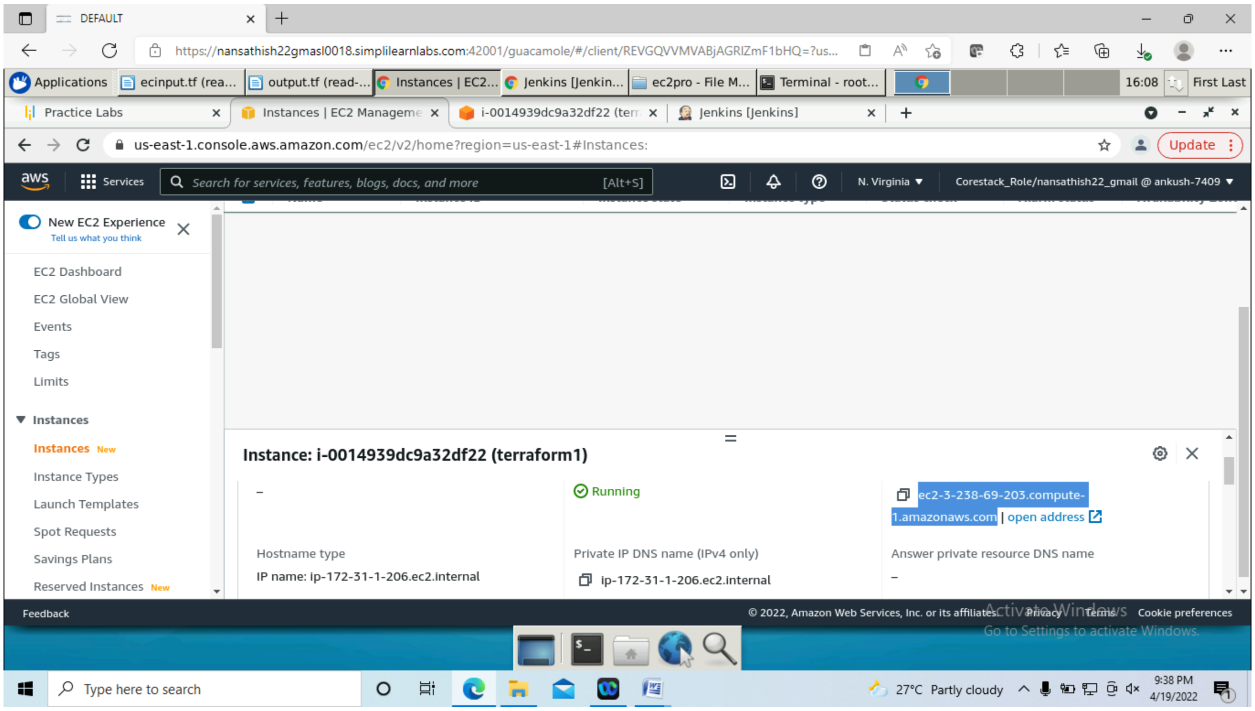
Go back to the console of your local system and run:

***ssh<EC2 user>@<EC2 public DNS>***

ssh ec2-user@ec2-3-238-69-203.compute-1.amazonaws.com

or ssh -i "pkey.pem" ec2-user@ec2-3-238-69-203.compute-1.amazonaws.com

The command looks like this "ssh ec2-user@ec2-3-238-69-203.compute-1.amazonaws.com"). When prompted, provide yes to permanently add the IP to the list of known hosts. we can copy the same from your instance's summary:



Instance: i-0014939dc9a32df22 (terraform1)

Running

ec2-3-238-69-203.compute-1.amazonaws.com | open address

Hostname type

IP name: ip-172-31-1-206.ec2.internal

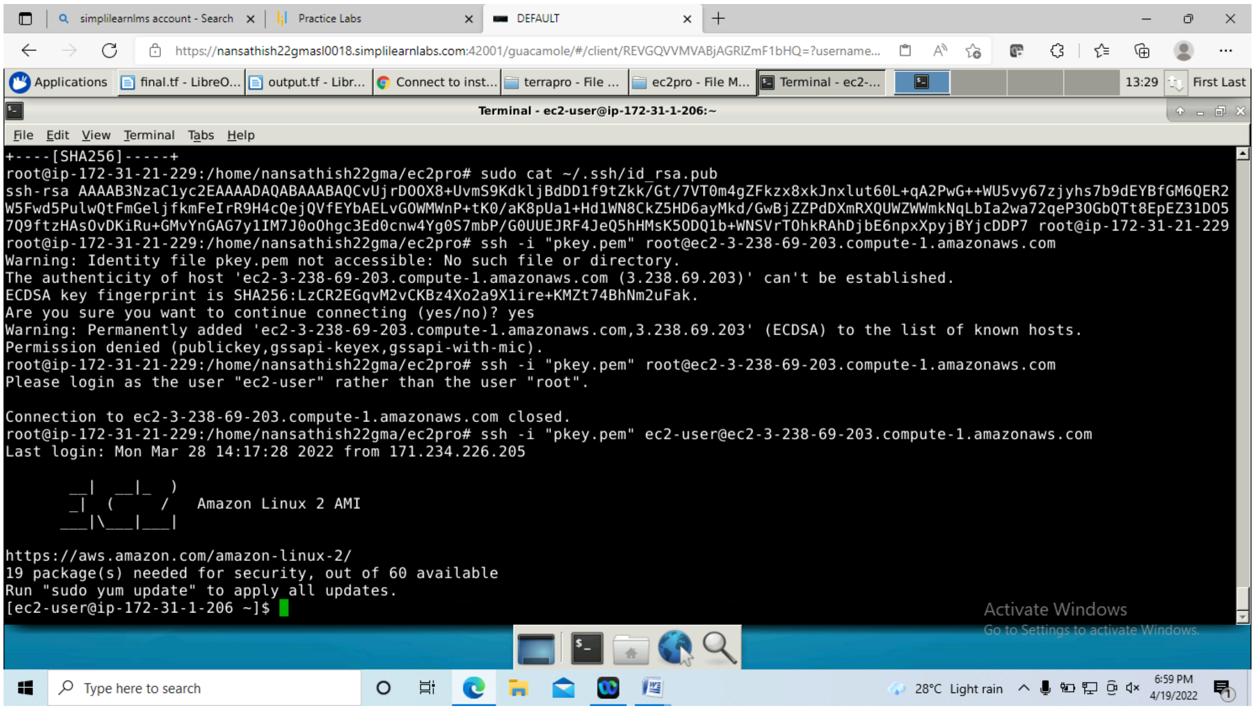
Private IP DNS name (IPv4 only)

ip-172-31-1-206.ec2.internal

Answer private resource DNS name

–

Connectivity is established between local system and the newly created EC2 instance:



```
+----[SHA256]-----+
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# sudo cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCvUjrDOOX8+UvmS9KdkljBdDD1f9tZkk/Gt/7VT0m4gZFkzx8xkJnxlut60L+qA2PwG++WU5vy67zjyhs7b9dEYBfGM6QER2
W5Fwd5PulwQtFmGeljfkmFeIrR9H4cQejQVfEYbAELvGOWMWnP+tK0/aK8pUa1+Hd1WN8CkZ5HD6ayMkd/GwBjZZPdDXmRXQUWZWWmkNqLbIa2wa72qeP3OGbQTt8EpEZ31DO5
7Q9ftzHAsOvDKiRu+GMvYnGAG7y1IM7J0oOhgc3Ed0cnw4Yg0S7mbP/G0UUEJRF4JeQ5hHMsK5ODQ1b+WNSVrTOhkRAhDjbE6npxXpyjBYjcDDP7 root@ip-172-31-21-229
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# ssh -i "pkey.pem" root@ec2-3-238-69-203.compute-1.amazonaws.com
Warning: Identity file pkey.pem not accessible: No such file or directory.
The authenticity of host 'ec2-3-238-69-203.compute-1.amazonaws.com (3.238.69.203)' can't be established.
ECDSA key fingerprint is SHA256:LzCR2EGqvM2vCKBz4Xo2a9X1ire+KMZt74BhNm2uFak.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-3-238-69-203.compute-1.amazonaws.com,3.238.69.203' (ECDSA) to the list of known hosts.
Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# ssh -i "pkey.pem" root@ec2-3-238-69-203.compute-1.amazonaws.com
Please login as the user "ec2-user" rather than the user "root".

Connection to ec2-3-238-69-203.compute-1.amazonaws.com closed.
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# ssh -i "pkey.pem" ec2-user@ec2-3-238-69-203.compute-1.amazonaws.com
Last login: Mon Mar 28 14:17:28 2022 from 171.234.226.205

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
19 package(s) needed for security, out of 60 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-1-206 ~]$ 
```

Terminal - ec2-user@ip-172-31-1-206:~
File Edit View Terminal Tabs Help

Next is to prepare the instance for the upcoming installation of Jenkins, Java and PythonusingAnsible. For this purpose, we want to execute the following commands to ensure we have the packages we need:

***sudo yum update***

When prompted enter "yes".

```
sudo amazon-linux-extras install epel –y
```

This is a package required by Jenkins. Exit the instance:

```
exit
```

**Step 5. Install Jenkins, Java and Python using Ansible.**

For this step, we want to ensure that Ansible is installed on our local system. The required steps fallows as:

# Setting up Ansible

**Objective:** To install Ansible and set up in your system

**Pre-requisites:** You need to have Python 2.7 or higher, Minimum 8 GB RAM, and SSH or SCP communicator.

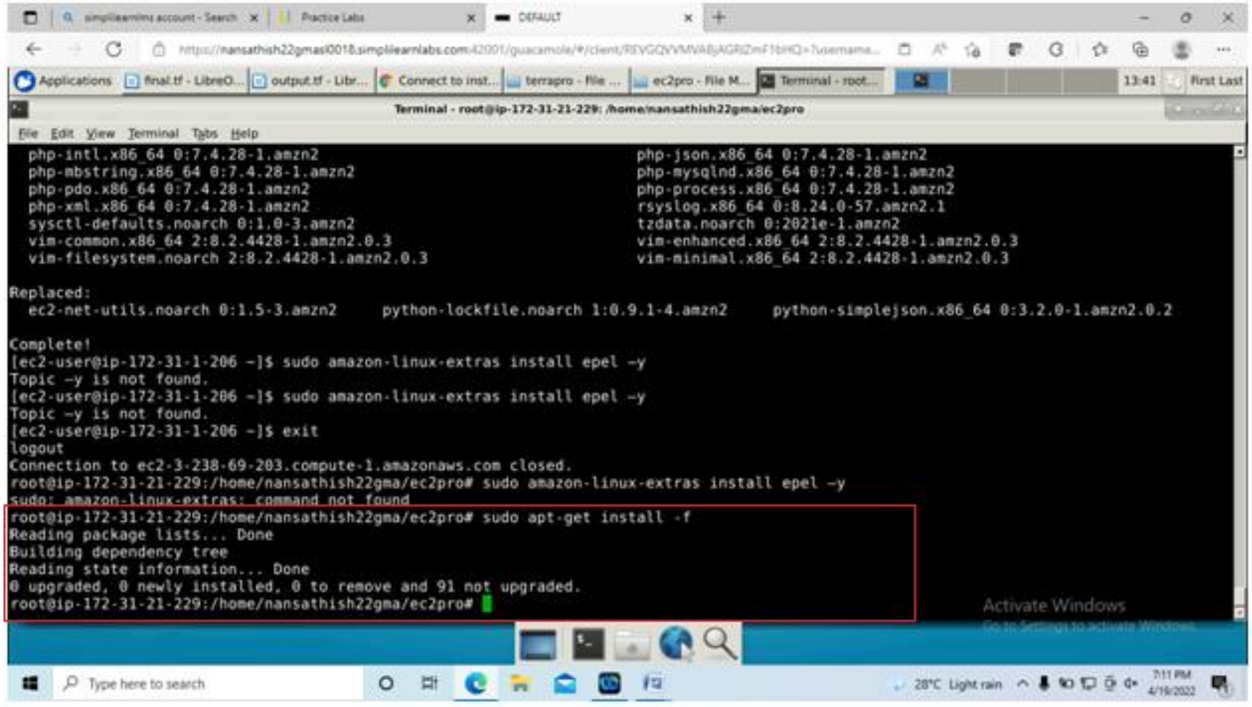
**Steps to be followed:**

1. Install Ansible

## Step 1: Install Ansible

1.1 Use the below command to check and find the dependencies of the packages you want, and install any that are needed

```
sudo apt-get install -f
```

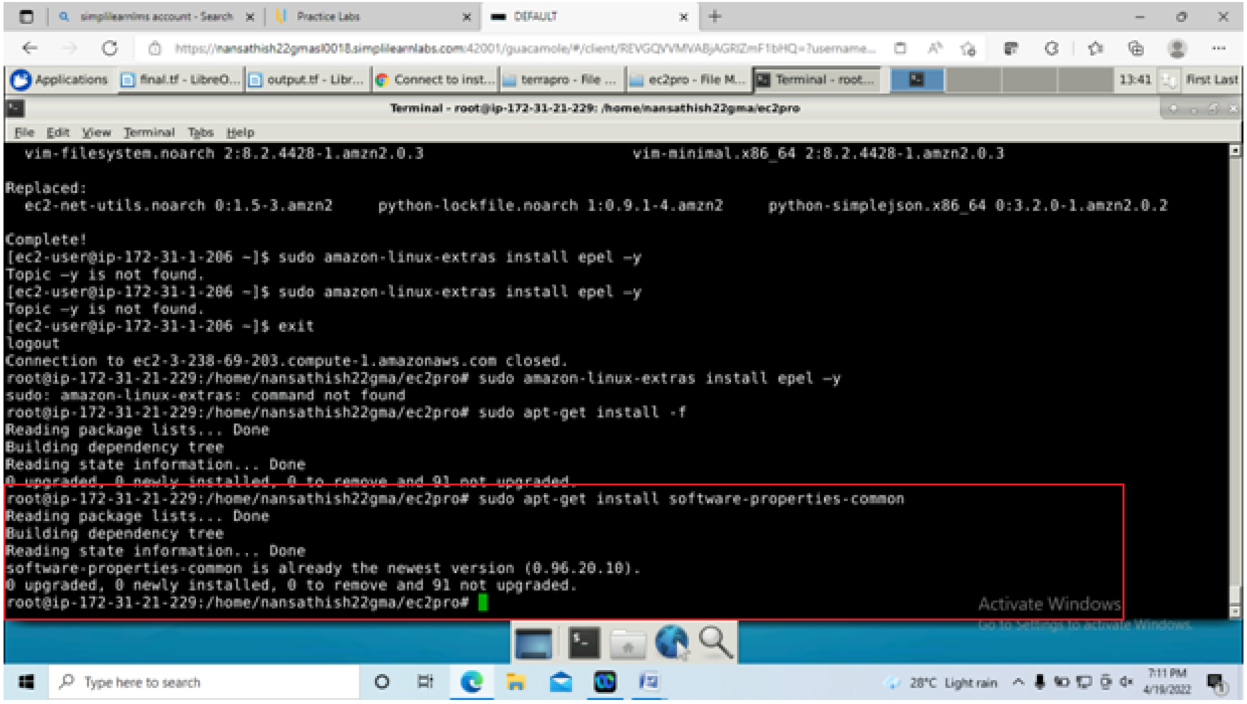


Terminal - root@ip-172-31-21-229: /home/nansathish22gma/ec2pro
File Edit View Terminal Tabs Help

```
  php-intl.x86_64 0:7.4.28-1.amzn2                                   php-json.x86_64 0:7.4.28-1.amzn2
  php-mbstring.x86_64 0:7.4.28-1.amzn2                               php-mysqlnd.x86_64 0:7.4.28-1.amzn2
  php-pdo.x86_64 0:7.4.28-1.amzn2                                    php-process.x86_64 0:7.4.28-1.amzn2
  php-xml.x86_64 0:7.4.28-1.amzn2                                    rsyslog.x86_64 0:8.24.0-57.amzn2.1
  sysctl-defaults.noarch 0:1.0-3.amzn2                               tzdata.noarch 0:2021e-1.amzn2
  vim-common.x86_64 2:8.2.4428-1.amzn2.0.3                           vim-enhanced.x86_64 2:8.2.4428-1.amzn2.0.3
  vim-filesystem.noarch 2:8.2.4428-1.amzn2.0.3                       vim-minimal.x86_64 2:8.2.4428-1.amzn2.0.3

Replaced:
  ec2-net-utils.noarch 0:1.5-3.amzn2     python-lockfile.noarch 1:0.9.1-4.amzn2     python-simplejson.x86_64 0:3.2.0-1.amzn2.0.2

Complete!
[ec2-user@ip-172-31-1-206 ~]$ sudo amazon-linux-extras install epel —y
Topic —y is not found.
[ec2-user@ip-172-31-1-206 ~]$ sudo amazon-linux-extras install epel —y
Topic —y is not found.
[ec2-user@ip-172-31-1-206 ~]$ exit
logout
Connection to ec2-3-238-69-203.compute-1.amazonaws.com closed.
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# sudo amazon-linux-extras install epel —y
sudo: amazon-linux-extras: command not found
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# sudo apt-get install -f
Reading package lists... Done
Building dependency tree
Reading state information... Done
0 upgraded, 0 newly installed, 0 to remove and 91 not upgraded.
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro#
```

1.2 Use the below command to update package repositories and get latest package information

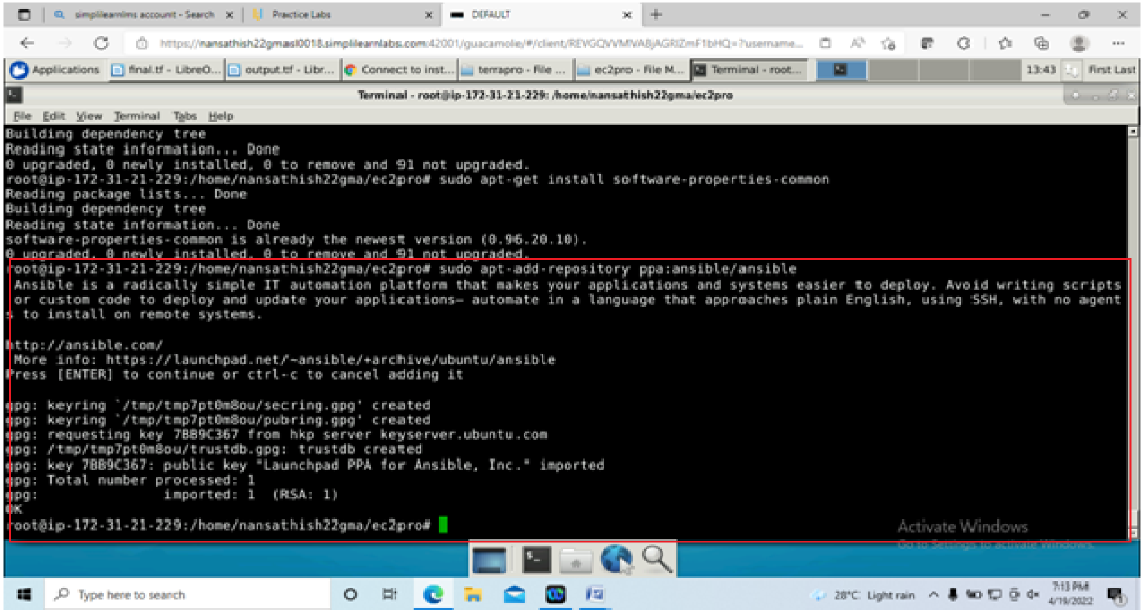
***sudo apt-get install software-properties-common***



```
  vim-filesystem.noarch 2:8.2.4428-1.amzn2.0.3                        vim-minimal.x86_64 2:8.2.4428-1.amzn2.0.3

Replaced:
  ec2-net-utils.noarch 0:1.5-3.amzn2     python-lockfile.noarch 1:0.9.1-4.amzn2     python-simplejson.x86_64 0:3.2.0-1.amzn2.0.2

Complete!
[ec2-user@ip-172-31-1-206 ~]$ sudo amazon-linux-extras install epel –y
Topic –y is not found.
[ec2-user@ip-172-31-1-206 ~]$ sudo amazon-linux-extras install epel –y
Topic –y is not found.
[ec2-user@ip-172-31-1-206 ~]$ exit
logout
Connection to ec2-3-238-69-203.compute-1.amazonaws.com closed.
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# sudo amazon-linux-extras install epel –y
sudo: amazon-linux-extras: command not found
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# sudo apt-get install -f
Reading package lists... Done
Building dependency tree
Reading state information... Done
0 upgraded, 0 newly installed, 0 to remove and 91 not upgraded.
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# sudo apt-get install software-properties-common
Reading package lists... Done
Building dependency tree
Reading state information... Done
software-properties-common is already the newest version (0.96.20.10).
0 upgraded, 0 newly installed, 0 to remove and 91 not upgraded.
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# 
```

1.3 Run these commands to update the list of available software once again and install Ansible. It also pulls down **Ansible PPA's** signing key and **adds** it to your system

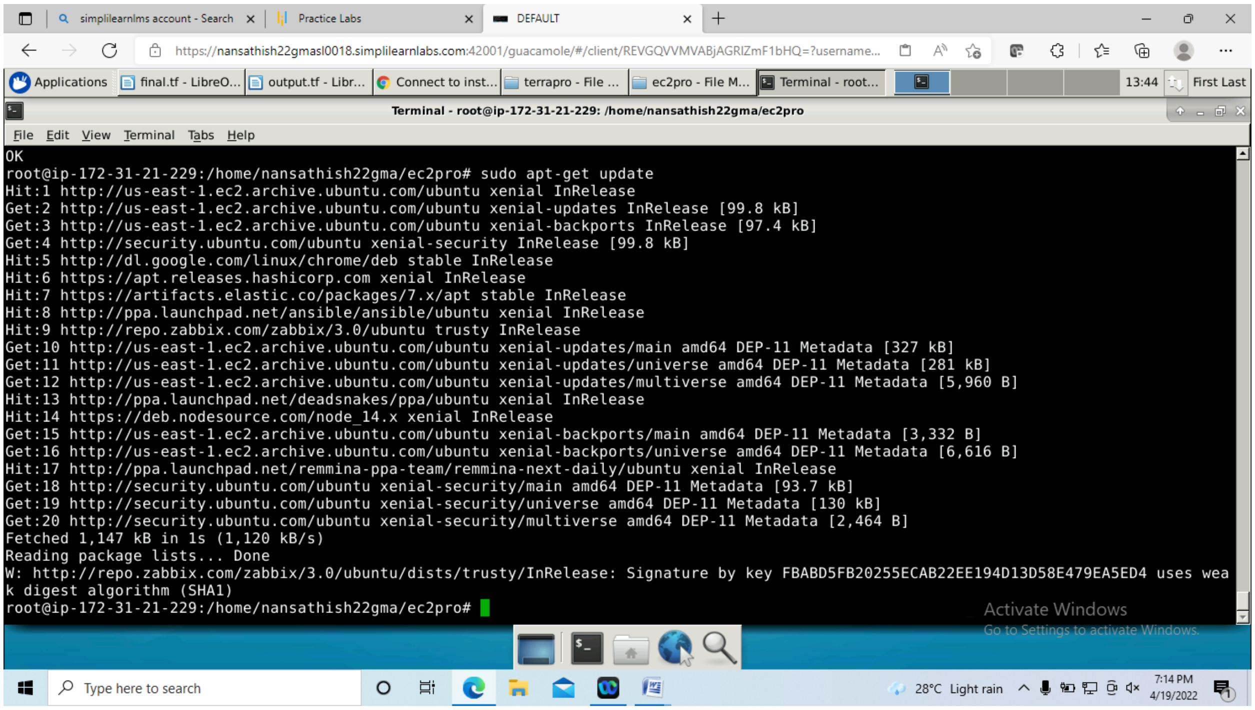
***sudo apt-add-repository ppa:ansible/ansible***


Terminal - root@ip-172-31-21-229: /home/nansathish22gma/ec2pro
File Edit View Terminal Tabs Help

```
Building dependency tree
Reading state information... Done
0 upgraded, 0 newly installed, 0 to remove and 91 not upgraded.
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# sudo apt-get install software-properties-common
Reading package lists... Done
Building dependency tree
Reading state information... Done
software-properties-common is already the newest version (0.96.20.10).
0 upgraded, 0 newly installed, 0 to remove and 91 not upgraded.
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# sudo apt-add-repository ppa:ansible/ansible
 Ansible is a radically simple IT automation platform that makes your applications and systems easier to deploy. Avoid writing scripts
 or custom code to deploy and update your applications— automate in a language that approaches plain English, using SSH, with no agent
s to install on remote systems.

http://ansible.com/
 More info: https://launchpad.net/~ansible/+archive/ubuntu/ansible
Press [ENTER] to continue or ctrl-c to cancel adding it

gpg: keyring `/tmp/tmp7pt0m8ou/secring.gpg' created
gpg: keyring `/tmp/tmp7pt0m8ou/pubring.gpg' created
gpg: requesting key 7BB9C367 from hkp server keyserver.ubuntu.com
gpg: /tmp/tmp7pt0m8ou/trustdb.gpg: trustdb created
gpg: key 7BB9C367: public key "Launchpad PPA for Ansible, Inc." imported
gpg: Total number processed: 1
gpg:               imported: 1  (RSA: 1)
OK
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# 
```

1.4 Use the below command to download package information from all configured sources

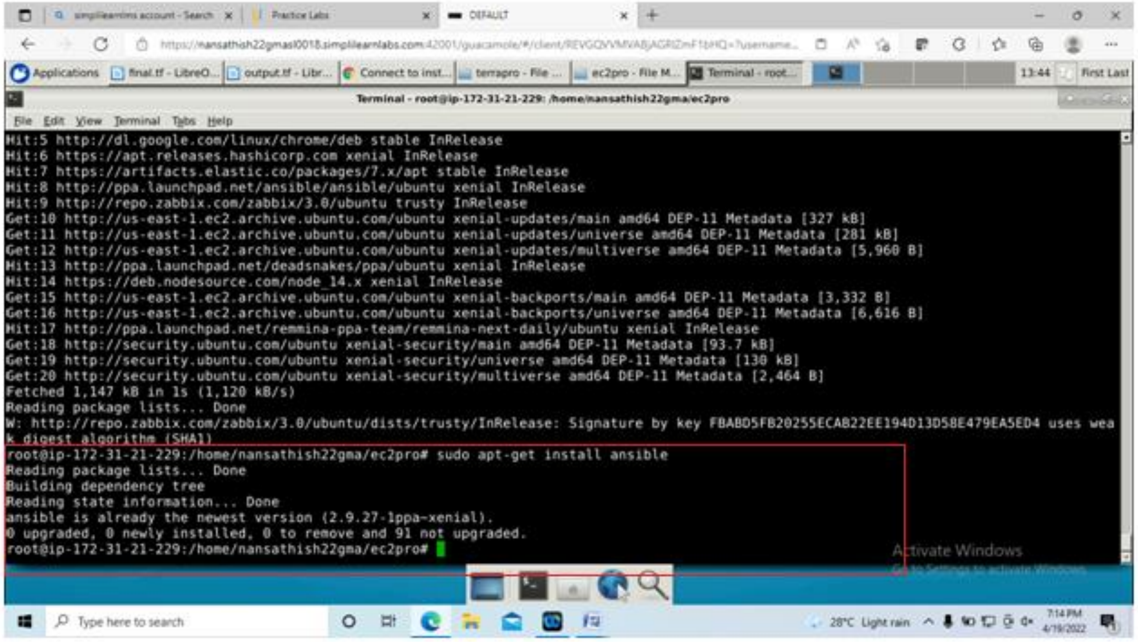
***sudo apt-get update***


Terminal - root@ip-172-31-21-229: /home/nansathish22gma/ec2pro
File Edit View Terminal Tabs Help

```
OK
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates InRelease [99.8 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-backports InRelease [97.4 kB]
Get:4 http://security.ubuntu.com/ubuntu xenial-security InRelease [99.8 kB]
Hit:5 http://dl.google.com/linux/chrome/deb stable InRelease
Hit:6 https://apt.releases.hashicorp.com xenial InRelease
Hit:7 https://artifacts.elastic.co/packages/7.x/apt stable InRelease
Hit:8 http://ppa.launchpad.net/ansible/ansible/ubuntu xenial InRelease
Hit:9 http://repo.zabbix.com/zabbix/3.0/ubuntu trusty InRelease
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates/main amd64 DEP-11 Metadata [327 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates/universe amd64 DEP-11 Metadata [281 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates/multiverse amd64 DEP-11 Metadata [5,960 B]
Hit:13 http://ppa.launchpad.net/deadsnakes/ppa/ubuntu xenial InRelease
Hit:14 https://deb.nodesource.com/node_14.x xenial InRelease
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-backports/main amd64 DEP-11 Metadata [3,332 B]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-backports/universe amd64 DEP-11 Metadata [6,616 B]
Hit:17 http://ppa.launchpad.net/remmina-ppa-team/remmina-next-daily/ubuntu xenial InRelease
Get:18 http://security.ubuntu.com/ubuntu xenial-security/main amd64 DEP-11 Metadata [93.7 kB]
Get:19 http://security.ubuntu.com/ubuntu xenial-security/universe amd64 DEP-11 Metadata [130 kB]
Get:20 http://security.ubuntu.com/ubuntu xenial-security/multiverse amd64 DEP-11 Metadata [2,464 B]
Fetched 1,147 kB in 1s (1,120 kB/s)
Reading package lists... Done
W: http://repo.zabbix.com/zabbix/3.0/ubuntu/dists/trusty/InRelease: Signature by key FBABD5FB20255ECAB22EE194D13D58E479EA5ED4 uses wea
k digest algorithm (SHA1)
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# 
```

1.5 Use the below command to install Ansible

***sudo apt-get install ansible***



```
Hit:5 http://dl.google.com/linux/chrome/deb stable InRelease
Hit:6 https://apt.releases.hashicorp.com xenial InRelease
Hit:7 https://artifacts.elastic.co/packages/7.x/apt stable InRelease
Hit:8 http://ppa.launchpad.net/ansible/ansible/ubuntu xenial InRelease
Hit:9 http://repo.zabbix.com/zabbix/3.0/ubuntu trusty InRelease
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates/main amd64 DEP-11 Metadata [327 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates/universe amd64 DEP-11 Metadata [281 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-updates/multiverse amd64 DEP-11 Metadata [5,960 B]
Hit:13 http://ppa.launchpad.net/deadsnakes/ppa/ubuntu xenial InRelease
Hit:14 https://deb.nodesource.com/node_14.x xenial InRelease
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-backports/main amd64 DEP-11 Metadata [3,332 B]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu xenial-backports/universe amd64 DEP-11 Metadata [6,616 B]
Hit:17 http://ppa.launchpad.net/remmina-ppa-team/remmina-next-daily/ubuntu xenial InRelease
Get:18 http://security.ubuntu.com/ubuntu xenial-security/main amd64 DEP-11 Metadata [93.7 kB]
Get:19 http://security.ubuntu.com/ubuntu xenial-security/universe amd64 DEP-11 Metadata [130 kB]
Get:20 http://security.ubuntu.com/ubuntu xenial-security/multiverse amd64 DEP-11 Metadata [2,464 B]
Fetched 1,147 kB in 1s (1,120 kB/s)
Reading package lists... Done
W: http://repo.zabbix.com/zabbix/3.0/ubuntu/dists/trusty/InRelease: Signature by key FBABD5FB20255ECAB22EE194D13D58E479EA5ED4 uses wea
k digest algorithm (SHA1)
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# sudo apt-get install ansible
Reading package lists... Done
Building dependency tree
Reading state information... Done
ansible is already the newest version (2.9.27-1ppa~xenial).
0 upgraded, 0 newly installed, 0 to remove and 91 not upgraded.
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro#
```

Next, we want to establish connectivity between Ansible controller and the EC2 instance:

***sudo vi /etc/ansible/hosts***

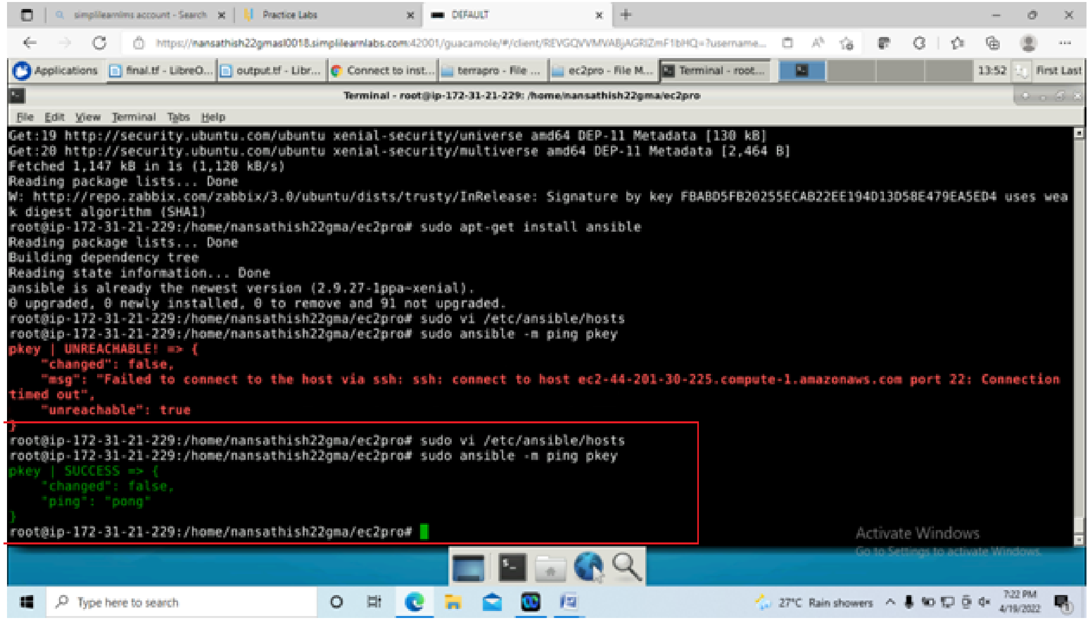
At the bottom of the file, insert the following line:

***[all]***

***pkey ansible_host=ec2-3-238-69-203.compute-1.amazonaws.com ansible_user=ec2-user ansible_ssh_private_key_file=/home/nansathish22gma/ec2pro/pkey.pem ansible_python_interpreter=/usr/bin/python2***

Now make sure the connection is working using the following command:

***sudo ansible -m ping pkey***



Terminal - root@ip-172-31-21-229: /home/nansathish22gma/ec2pro
File Edit View Terminal Tabs Help

```
Get:19 http://security.ubuntu.com/ubuntu xenial-security/universe amd64 DEP-11 Metadata [130 kB]
Get:20 http://security.ubuntu.com/ubuntu xenial-security/multiverse amd64 DEP-11 Metadata [2,464 B]
Fetched 1,147 kB in 1s (1,120 kB/s)
Reading package lists... Done
W: http://repo.zabbix.com/zabbix/3.0/ubuntu/dists/trusty/InRelease: Signature by key FBABD5FB20255ECAB22EE194D13D58E479EA5ED4 uses wea
k digest algorithm (SHA1)
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# sudo apt-get install ansible
Reading package lists... Done
Building dependency tree
Reading state information... Done
ansible is already the newest version (2.9.27-1ppa~xenial).
0 upgraded, 0 newly installed, 0 to remove and 91 not upgraded.
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# sudo vi /etc/ansible/hosts
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# sudo ansible -m ping pkey
pkey | UNREACHABLE! => {
    "changed": false,
    "msg": "Failed to connect to the host via ssh: ssh: connect to host ec2-44-201-30-225.compute-1.amazonaws.com port 22: Connection
timed out",
    "unreachable": true
}
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# sudo vi /etc/ansible/hosts
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# sudo ansible -m ping pkey
pkey | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro#
```

Finally, deploy Jenkins, Java and Python usingAnsible. For this purpose, configure the Ansible"yml" file:

***vim proansi.yml***

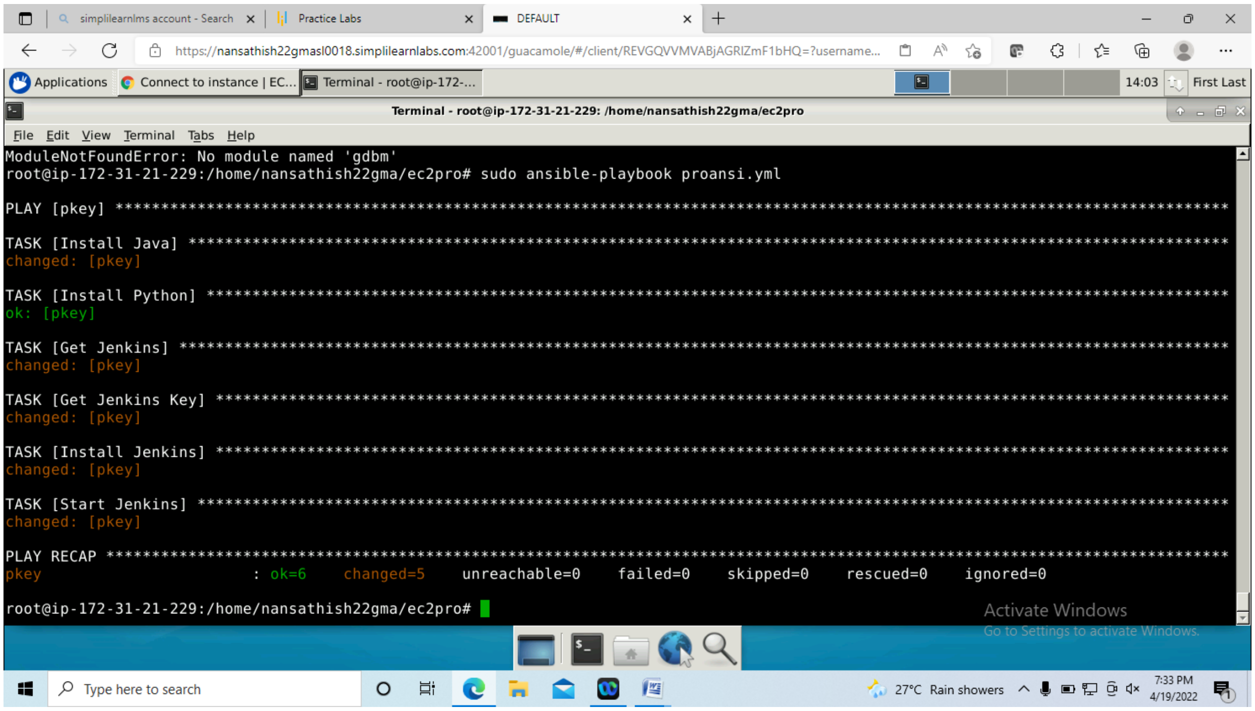
Configure the following script:

```
---

- hosts: pkey
  remote_user: ec2-user
  gather_facts: no
  become: true
  tasks:
    - name: Install Java
      yum:
        name: java-1.8.0-openjdk-devel
        state: present
        update_cache: yes
    - name: Install Python
      yum:
        name: python2
        state: present
        update_cache: yes
    - name: Get Jenkins
      get_url:
        url: http://pkg.jenkins-ci.org/redhat-stable/jenkins.repo
        dest: /etc/yum.repos.d/jenkins.repo
    - name: Get Jenkins Key
      rpm_key:
        state: present
        key: https://pkg.jenkins.io/redhat/jenkins.io.key
    - name: Install Jenkins
      yum:
        name: jenkins
        state: present
        update_cache: yes
    - name: Start Jenkins
      systemd:
        name: jenkins
        state: started
        enabled: true
```

Save the file. Execute Ansible via the following command:

```
sudo  ansible-playbook  proansi .yml
```

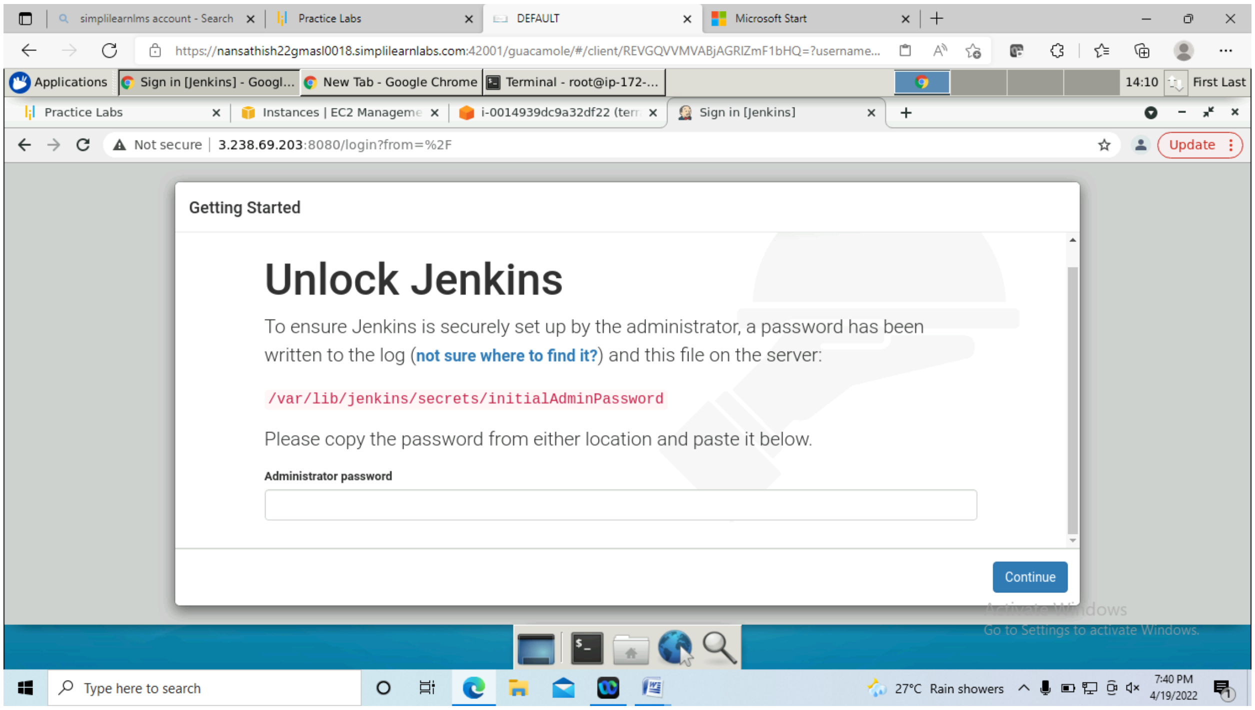
The end result is as displayed:


Terminal - root@ip-172-31-21-229: /home/nansathish22gma/ec2pro
File Edit View Terminal Tabs Help

```
ModuleNotFoundError: No module named 'gdbm'
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# sudo ansible-playbook proansi.yml

PLAY [pkey] *********************************************************************************************

TASK [Install Java] *************************************************************************************
changed: [pkey]

TASK [Install Python] ***********************************************************************************
ok: [pkey]

TASK [Get Jenkins] **************************************************************************************
changed: [pkey]

TASK [Get Jenkins Key] **********************************************************************************
changed: [pkey]

TASK [Install Jenkins] **********************************************************************************
changed: [pkey]

TASK [Start Jenkins] ************************************************************************************
changed: [pkey]

PLAY RECAP **********************************************************************************************
pkey                       : ok=6    changed=5    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

root@ip-172-31-21-229:/home/nansathish22gma/ec2pro#
```

Finally, ensure Jenkins is correctly installed. For this purpose, copy the EC2 instance public IP address and enter it in a new browser tab (<EC2 instance public IP>:8080 )

http://3.238.69.203:8080/



## Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log (not sure where to find it?) and this file on the server:

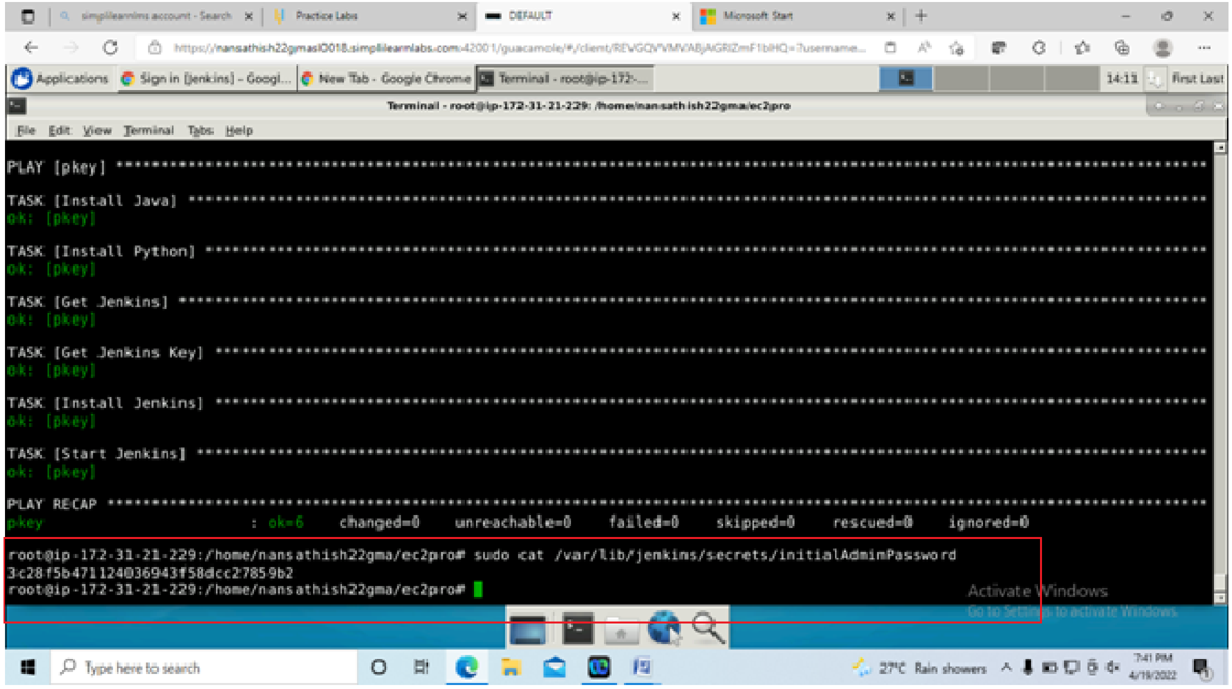
/var/lib/jenkins/secrets/initialAdminPassword

Please copy the password from either location and paste it below.

Administrator password

Get the Jenkins password via the following command:

***sudo cat /var/lib/jenkins/secrets/initialAdminPassword***



```
PLAY [pkey] ************************************************************************************************

TASK [Install Java] *****************************************************************************************
ok: [pkey]

TASK [Install Python] ***************************************************************************************
ok: [pkey]

TASK [Get Jenkins] ******************************************************************************************
ok: [pkey]

TASK [Get Jenkins Key] **************************************************************************************
ok: [pkey]

TASK [Install Jenkins] **************************************************************************************
ok: [pkey]

TASK [Start Jenkins] ****************************************************************************************
ok: [pkey]

PLAY RECAP **************************************************************************************************
pkey                       : ok=6    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

root@ip-172-31-21-229:/home/nansathish22gma/ec2pro# sudo cat /var/lib/jenkins/secrets/initialAdmimPassword
3c28f5b471124036943f58dcc27859b2
root@ip-172-31-21-229:/home/nansathish22gma/ec2pro#
```

Terminal - root@ip-172-31-21-229: /home/nansathish22gma/ec2pro
File Edit View Terminal Tabs Help
Applications
Sign in [Jenkins] - Googl...
New Tab - Google Chrome
Terminal - root@ip-172-...
14:11
First Last
Activate Windows
Go to Settings to activate Windows.
Type here to search
27°C Rain showers
7:41 PM
4/19/2022

Copy the same to unlock Jenkins.