

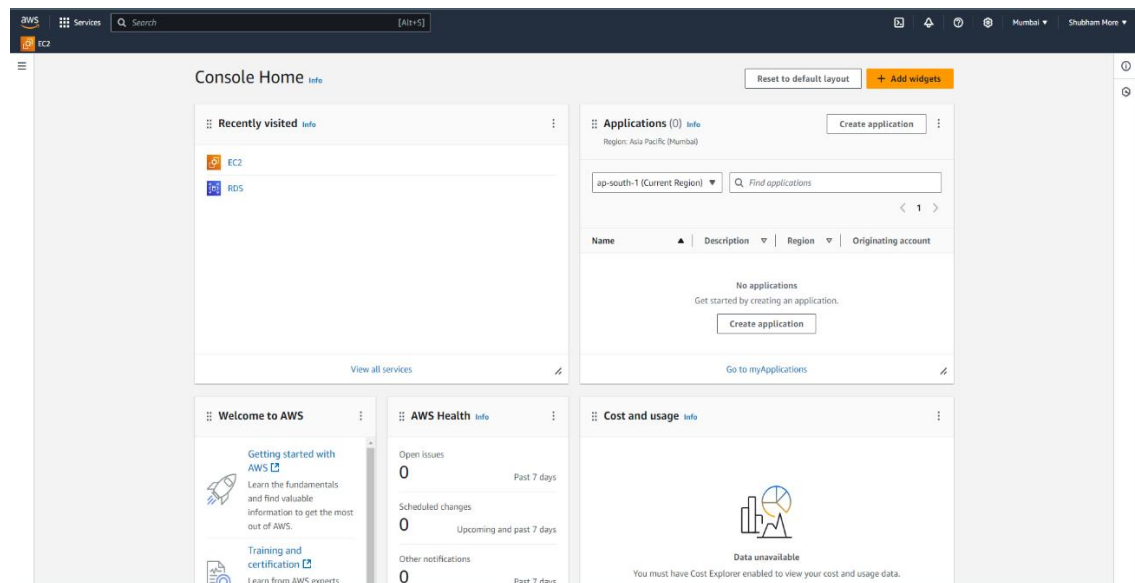
PRACTICAL - 5

Aim: Working in AWS to demonstrate provisioning and Scaling of a website.

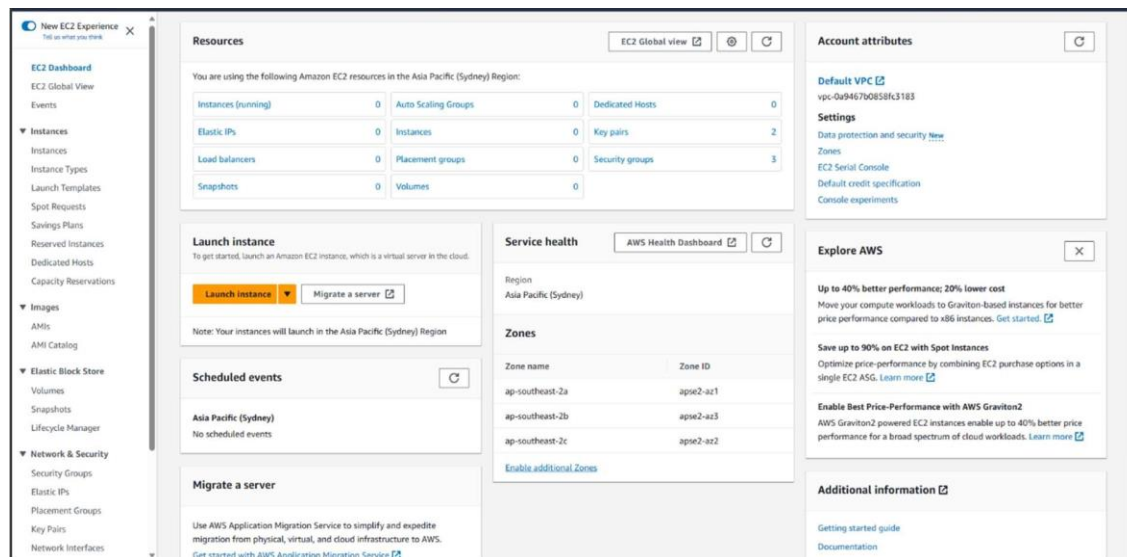
Practical Requirements:

- i) AWS (EC2)
- ii) Bitwise SSH Server (s/w)

Step 1: Open the AWS Management Console and select EC2 service.



Step 2 : Click on Launch instance button to create a new instance.



Step 3: Select the ubuntu server 22.04 LTS (HVM) to install ubuntu on instance.

Quick Start

Amazon Linux
aws


macOS
Mac

Ubuntu
ubuntu

Windows
Microsoft

Red Hat
Red Hat

SUSE Linux
SUSE


[Browse more AMIs](#)
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type Free tier eligible
ami-0310483fb2b488153 (64-bit (x86)) / ami-0370d04afa44db778 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Canonical, Ubuntu, 22.04 LTS, amd64 jammy image build on 2023-05-16

Architecture
64-bit (x86)

AMI ID
ami-0310483fb2b488153 Verified provider

Step 4: Choose the default instance type (free tier eligible only)

▼ Instance type [Info](#)

Instance type
t2.micro Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Linux base pricing: 0.0146 USD per Hour
On-Demand Windows base pricing: 0.0192 USD per Hour
On-Demand SUSE base pricing: 0.0146 USD per Hour
On-Demand RHEL base pricing: 0.0746 USD per Hour

☒ All generations
[Compare instance types](#)

Additional costs apply for AMIs with pre-installed software

Step 5: Now create key pair and select .pem file format and click on create key pair and download it.

Create key pair ✕

Key pair name
Key pairs allow you to connect to your instance securely.

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

Key pair type



☒ **RSA**
RSA encrypted private and public key pair

☐ **ED25519**
ED25519 encrypted private and public key pair

Private key file format

☒ **.pem**
For use with OpenSSH

☐ **.ppk**
For use with PuTTY


 When prompted, store the private key in a secure and accessible location on your computer. **You will need it later to connect to your instance.** [Learn more](#) 

Cancel Create key pair

Step 6: In this 8 gb storage is sufficient.

▼ Configure storage Info Advanced

1x GiB ▼ Root volume (Not encrypted)

 Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage ✕

Add new volume

The selected AMI contains more instance store volumes than the instance allows. Only the first 0 instance store volumes from the AMI will be accessible from the instance

0 x File systems Edit

Step 7: In this, select SSH, HTTPS, HTTP .

▼ Network settings Info

Edit

Network Info
vpc-0a9467b0858fc3183

Subnet Info
No preference (Default subnet in any availability zone)

Auto-assign public IP Info
Enable

Firewall (security groups) Info
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group ☐ Select existing security group

We'll create a new security group called 'launch-wizard-4' with the following rules:

☒ Allow SSH traffic from
Helps you connect to your instance
Anywhere
0.0.0.0/0

☒ Allow HTTPS traffic from the internet
To set up an endpoint, for example when creating a web server

☒ Allow HTTP traffic from the internet
To set up an endpoint, for example when creating a web server

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only. ✕

Step 8: Click on launch instance.

aws Services Search [Alt+S]

Firewall (security groups) Info
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group ☐ Select existing security group

We'll create a new security group called 'launch-wizard-3' with the following rules:

☒ Allow SSH traffic from
Helps you connect to your instance
Anywhere
0.0.0.0/0

☐ Allow HTTPS traffic from the internet
To set up an endpoint, for example when creating a web server

☐ Allow HTTP traffic from the internet
To set up an endpoint, for example when creating a web server

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only. ✕

▼ Configure storage Info Advanced

1x 8 GIB gp2 Root volume (Not encrypted)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage ✕

Add new volume

The selected AMI contains more instance store volumes than the instance allows. Only the first 0 instance store volumes from the AMI will be accessible from the instance

0 x File systems Edit

► Advanced details Info

▼ Summary

Number of instances Info
1

Software Image (AMI)
Canonical, Ubuntu, 22.04 LTS, ...read more
ami-0310483fb2b488153

Virtual server type (instance type)
t2.micro

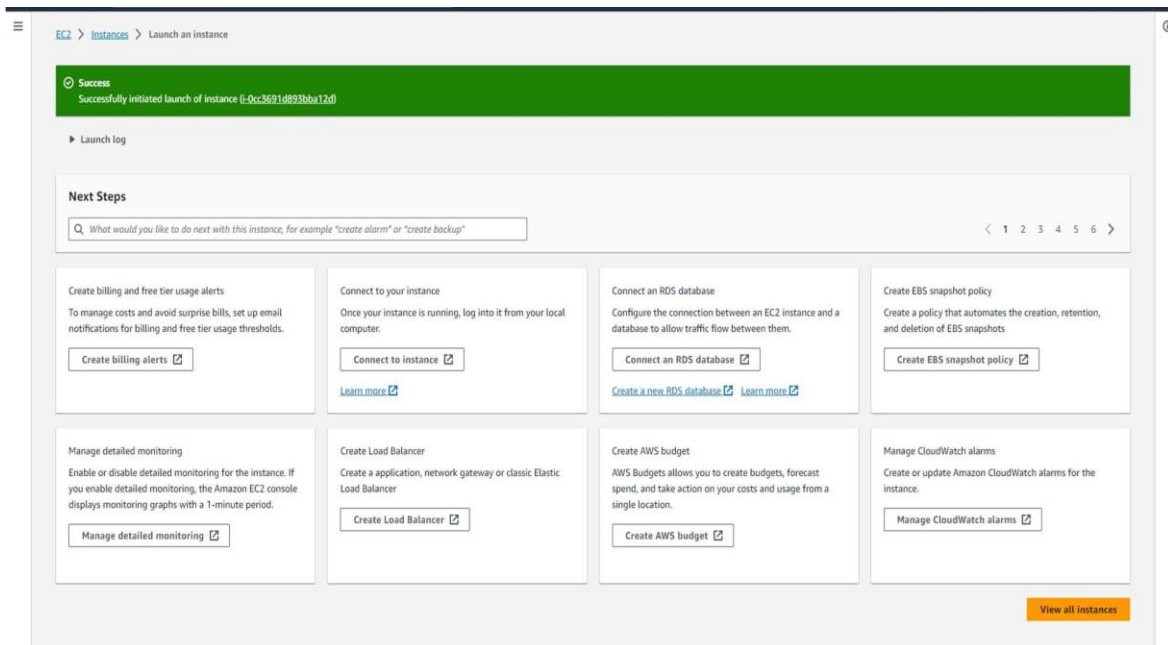
Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

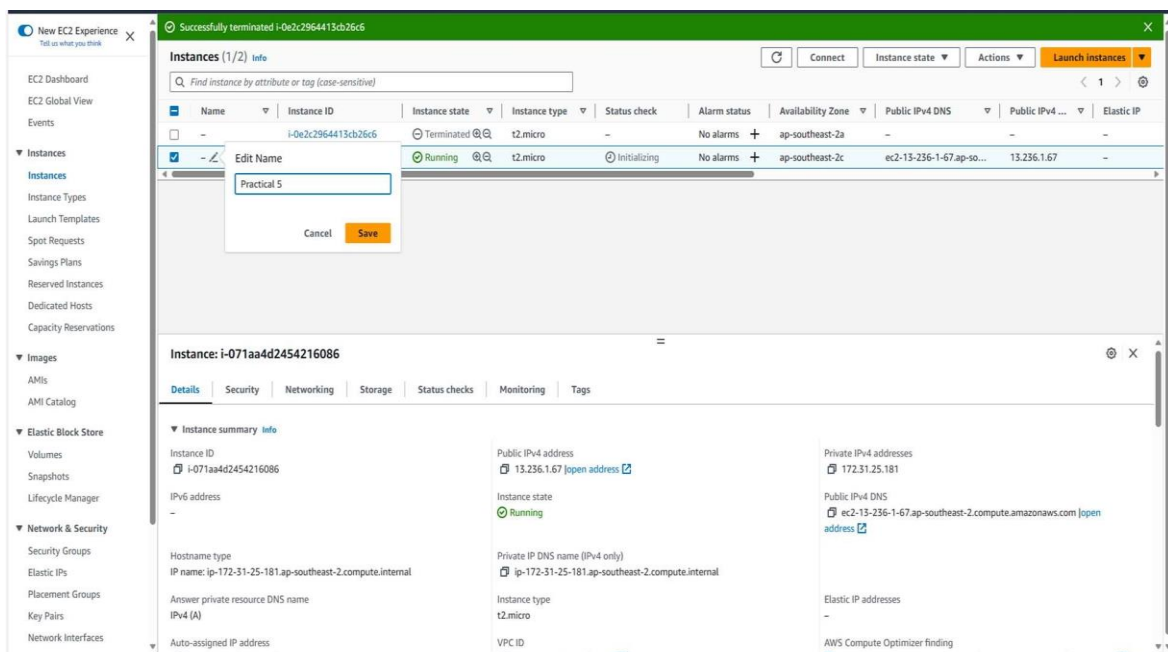
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet. ✕

Cancel Launch instance
Review commands

Step 9: Click on view all instances.



Step 10 : Then at this page your instance will be created and give any name to it to identify better be in future for example Practical 5

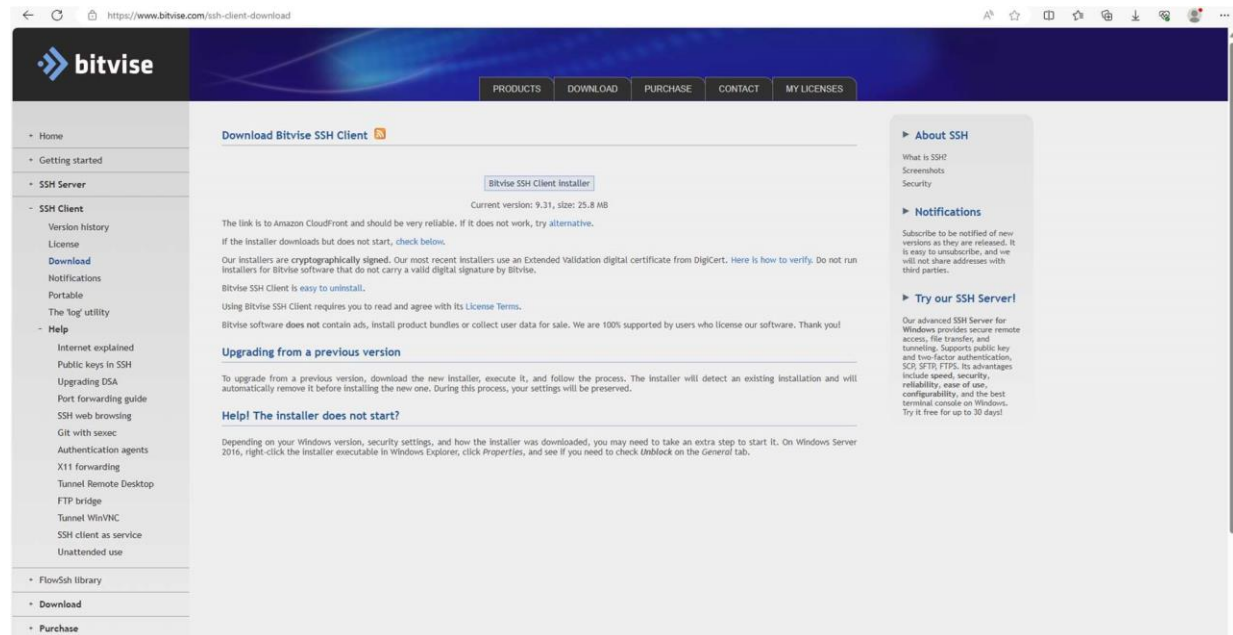


Step 11 : Then select that instance and click on connect at top to see your public IP and instance information.

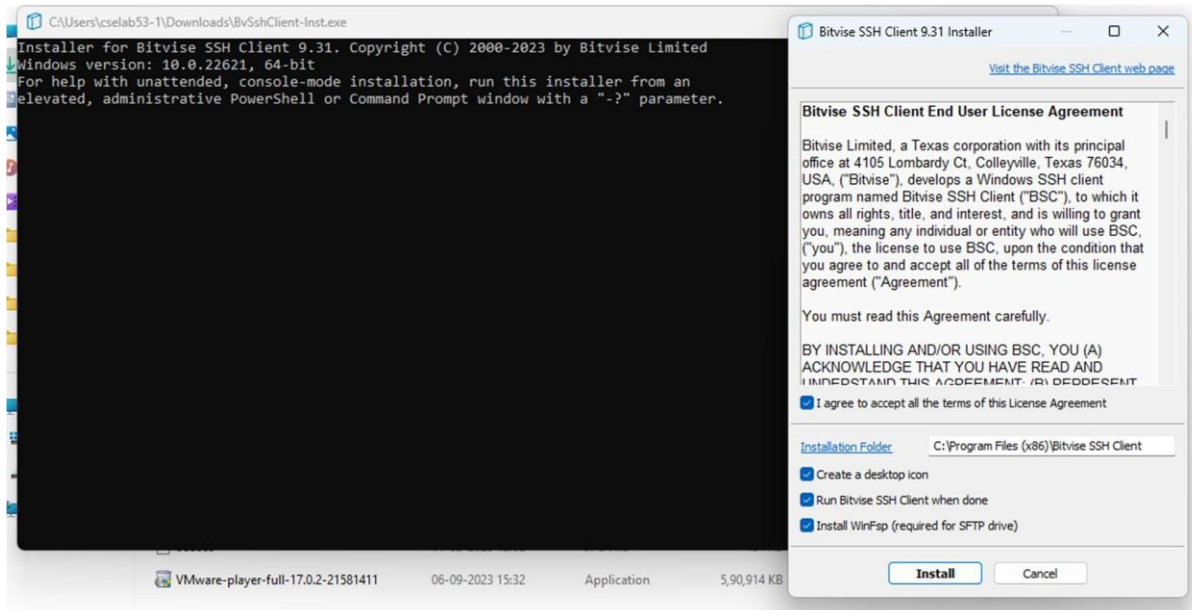
The screenshot shows the AWS Management Console interface. On the left is the navigation menu with categories like EC2 Dashboard, Images, Elastic Block Store, and Network & Security. The main area displays a table of EC2 instances. One instance, 'Practical 5' with ID 'i-071aa4d2454216086', is highlighted. Below the table, the 'Instance: i-071aa4d2454216086' details page is open, showing tabs for Details, Security, Networking, Storage, Status checks, Monitoring, and Tags. The 'Details' tab is active, showing the instance is 'Running' and its public IPv4 address is '13.236.1.67'.

The screenshot shows the 'Connect to instance' page in the AWS Management Console. The breadcrumb trail is 'EC2 > Instances > i-071aa4d2454216086 > Connect to instance'. The page title is 'Connect to instance' with an 'Info' link. Below the title, it says 'Connect to your instance i-071aa4d2454216086 using any of these options'. There are four tabs: 'EC2 Instance Connect', 'Session Manager', 'SSH client', and 'EC2 serial console'. The 'EC2 Instance Connect' tab is selected. Under this tab, there are two radio button options: 'Connect using EC2 Instance Connect' (selected) and 'Connect using EC2 Instance Connect Endpoint'. Below these options, the 'Public IP address' is shown as '13.236.1.67' and the 'User name' is 'ubuntu'. A note at the bottom states: 'Note: In most cases, the default user name, ubuntu, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.' At the bottom right, there are 'Cancel' and 'Connect' buttons.

Step 12 : Then download Bitwise SSH client 9.31 from <https://www.bitwise.com/ssh-clientdownload>



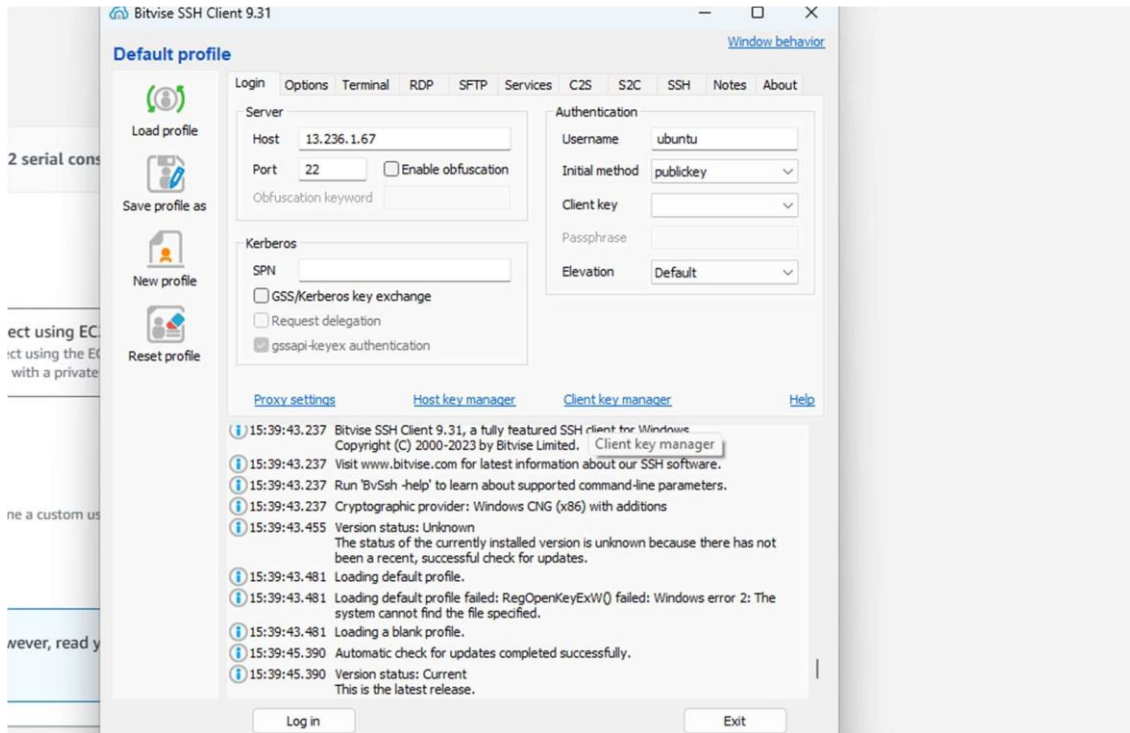
Step 13: Click on install and install it.



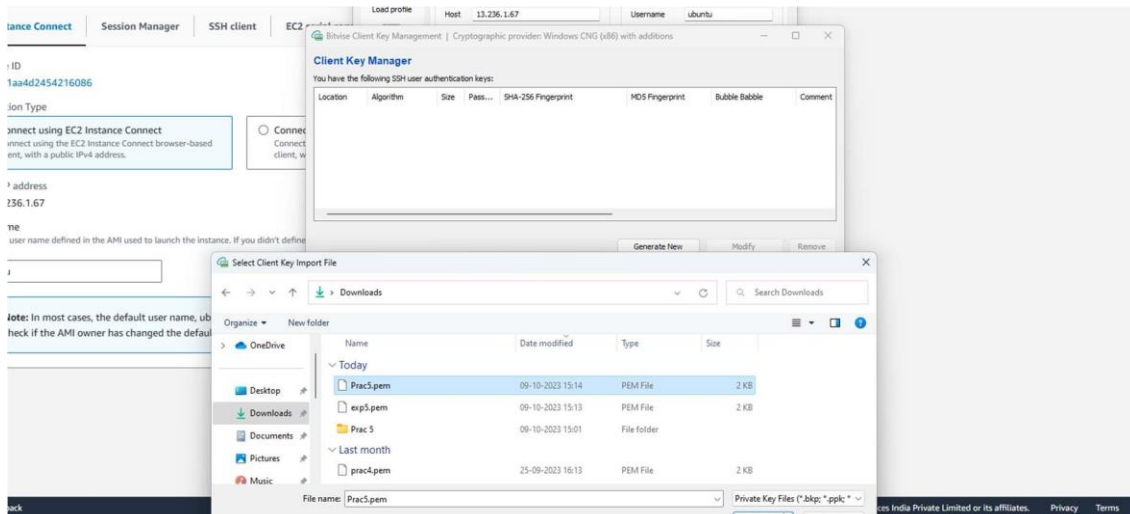
Step 14 Then open Bitvise SSH client and copy the public address of your instance from the EC2 instance connect section ex: 13.236.1.67. and paste it on the Bitvise Host text area column as shown in SS.

Then put the username as ubuntu, at initial method option select public key and elevation as default.

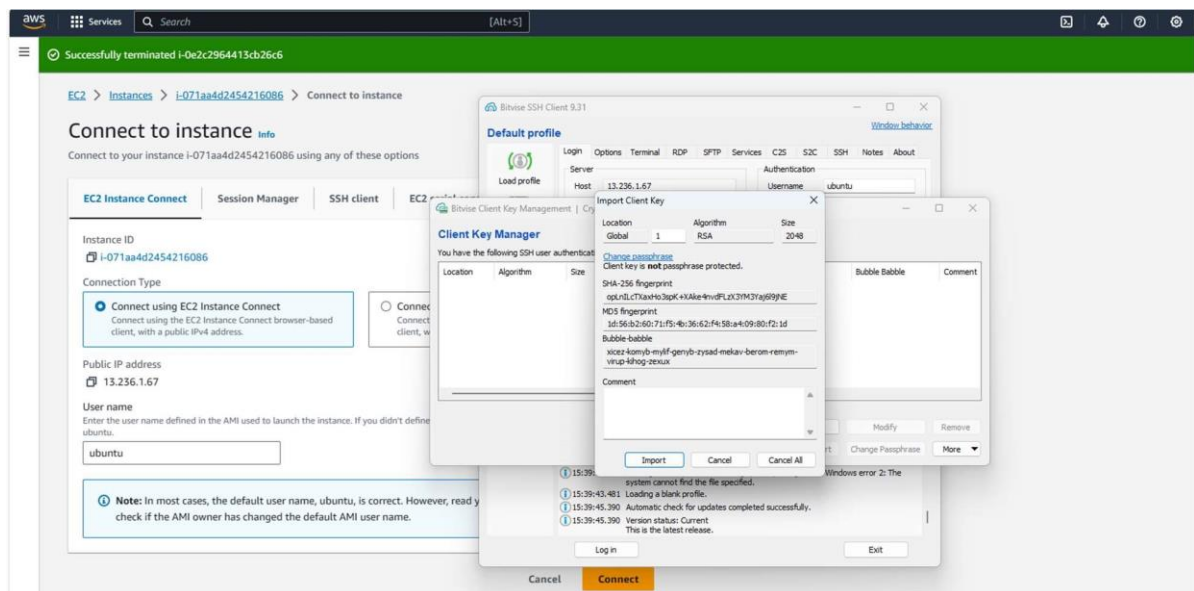
Then Click on Client key manager link.



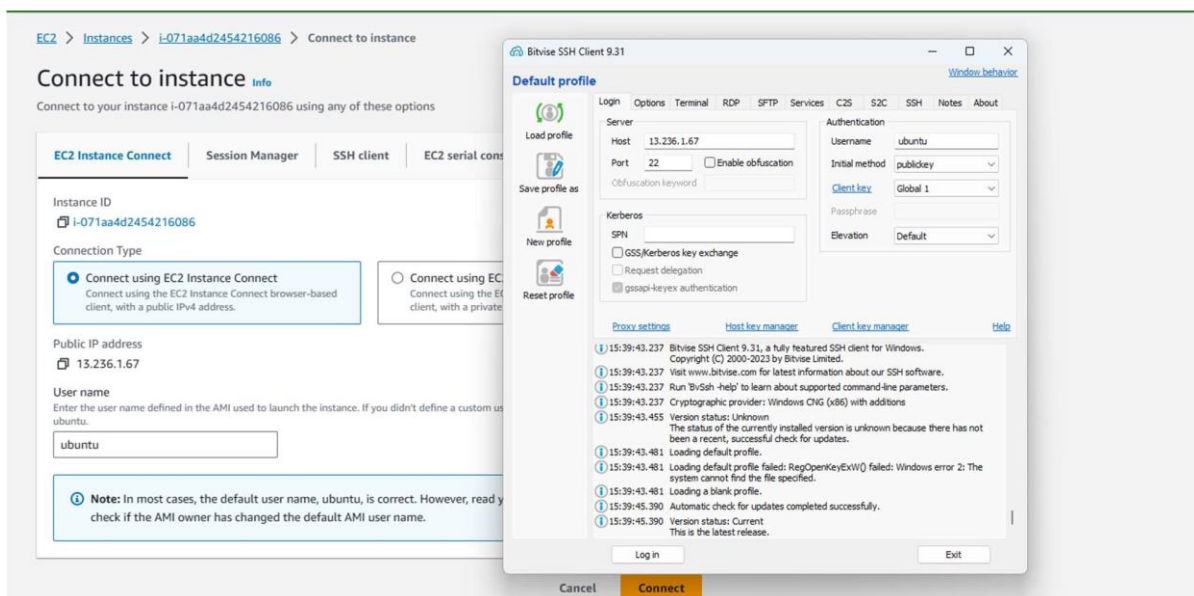
Step 15: Open the .pem file that we have downloaded at step no 5 (create key pair).



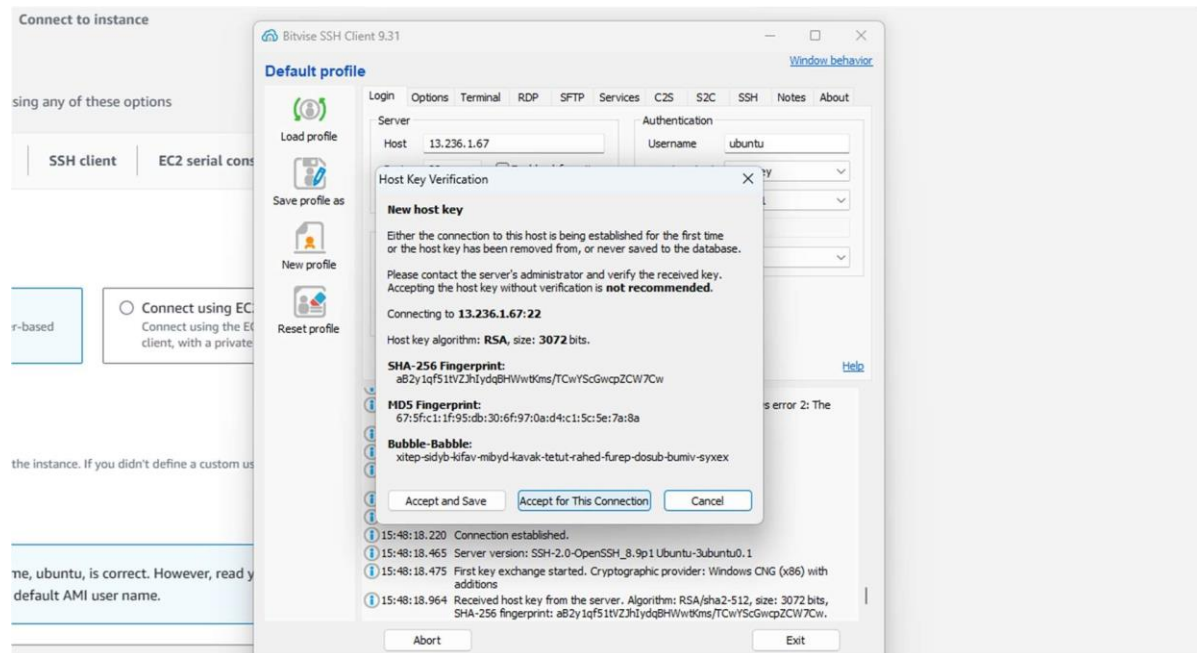
Step 16 : Then click on import button.



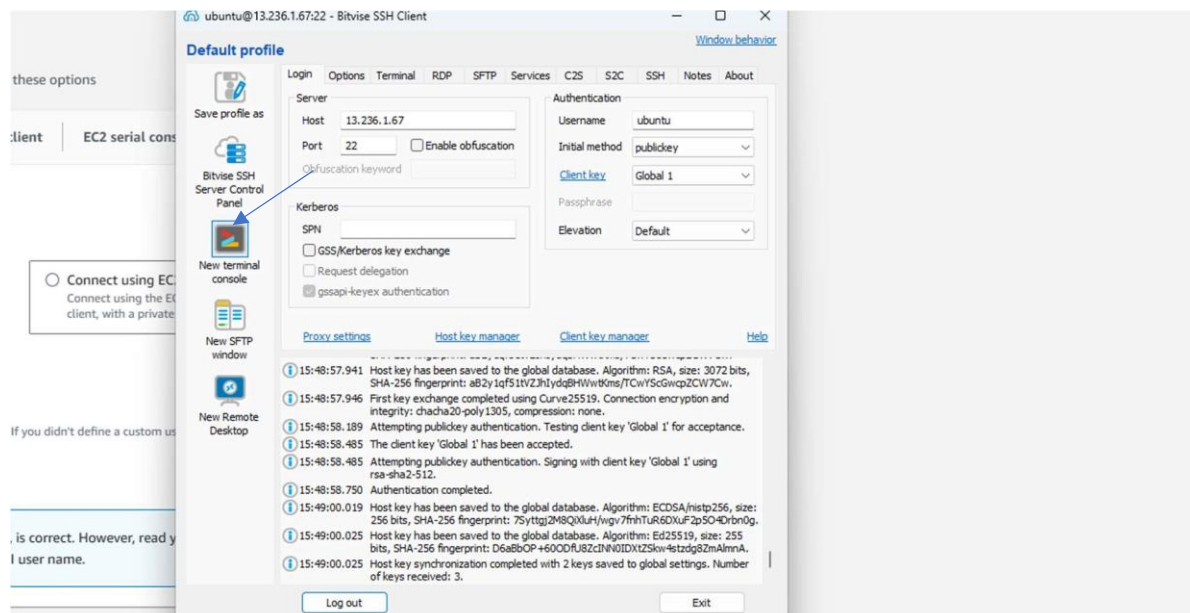
Step 17: Now at this step select the client key as Global 1 and click on log in button.



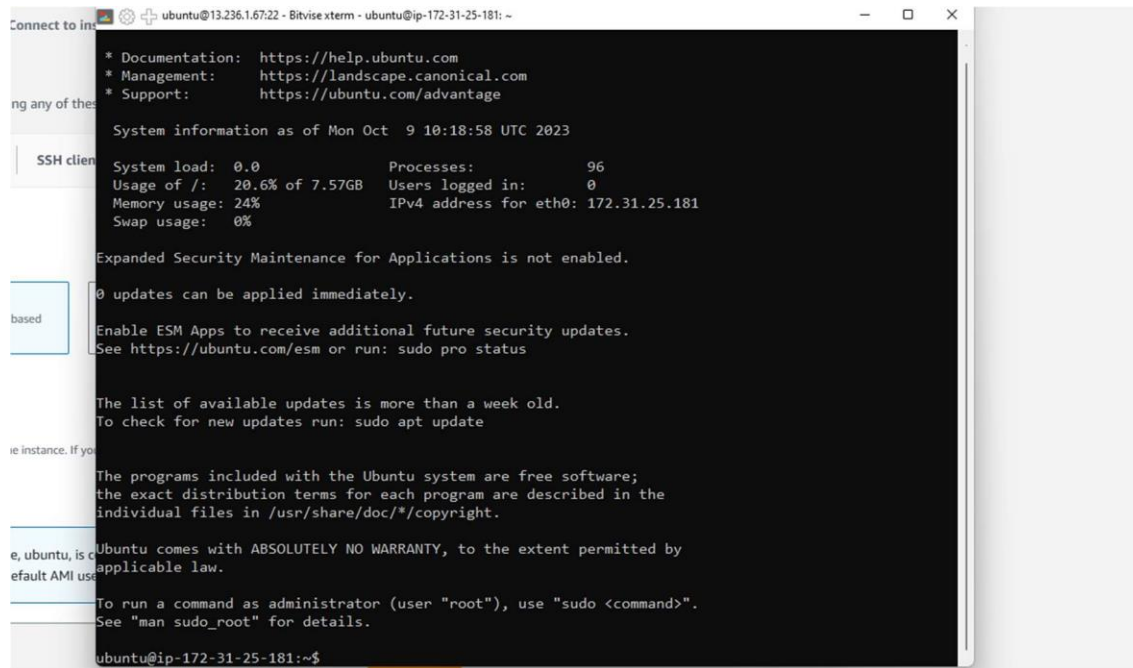
Step 18 Click on Accept and Save.



Step 19: Click on the new terminal window option to open the ubuntu OS.



Now our ubuntu OS window will Open like this.



```
Connect to ubuntu@13.236.1.67:22 - Bitvise xterm - ubuntu@ip-172-31-25-181: ~
* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/advantage

System information as of Mon Oct  9 10:18:58 UTC 2023

System load:  0.0          Processes:    96
Usage of /:   20.6% of 7.57GB Users logged in:  0
Memory usage: 24%         IPv4 address for eth0: 172.31.25.181
Swap usage:  0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

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

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

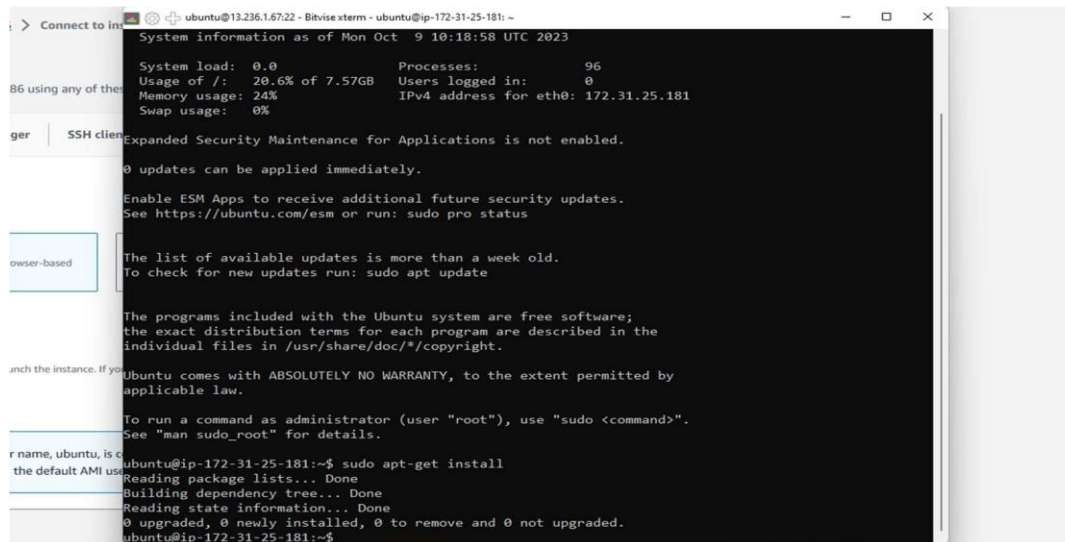
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

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.

ubuntu@ip-172-31-25-181:~$
```

Step 20: Use the following command to update your system before initiating a new installation `sudo apt-get install` or `sudo apt install` update depends on your system here the system is updated successfully.



```
> Connect to ubuntu@13.236.1.67:22 - Bitvise xterm - ubuntu@ip-172-31-25-181: ~
System information as of Mon Oct  9 10:18:58 UTC 2023

System load:  0.0          Processes:    96
Usage of /:   20.6% of 7.57GB Users logged in:  0
Memory usage: 24%         IPv4 address for eth0: 172.31.25.181
Swap usage:  0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

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

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.

ubuntu@ip-172-31-25-181:~$ sudo apt-get install
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
ubuntu@ip-172-31-25-181:~$
```

```
E: Failed to fetch http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu/pool/main/a/apache2/apache2-utls_2.4.52-1ubuntu4.5_amd64.deb 404 Not Found [IP: 13.210.201.60 80]
E: Failed to fetch http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu/pool/main/a/apache2/apache2_2.4.52-1ubuntu4.5_amd64.deb 404 Not Found [IP: 13.210.201.60 80]
E: Unable to fetch some archives, maybe run apt-get update or try with --fix-missing?
root@ip-172-31-25-181:~# apt-get update
Hit:1 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
```

Step 21: Switch to root user to install the apache server , for this , use following command.

sudo su -

```
ubuntu@ip-172-31-25-181:~$ sudo su -
root@ip-172-31-25-181:~#
```

Step 22: To install the apache2 use the following command.

Apt install apache2

```
ubuntu@ip-172-31-25-181:~$ sudo apt-get install
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
ubuntu@ip-172-31-25-181:~$ sudo su -
root@ip-172-31-25-181:~# apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
```

Step 23: Then we need to install te nginx server in your root user for this, use following command it will be installed.

Sudo apt-get install nginx

```
root@ip-172-31-25-181:~# sudo apt-get install nginx
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
```

Step 24 To switch in /var/www/html folder use the following command cd

/var/www/html

Now to list files in folder use following command

ls

```
root@ip-172-31-25-181:~# cd /var/www/html
root@ip-172-31-25-181:/var/www/html# ^C
-bash: :s^C: substitution failed
root@ip-172-31-25-181:/var/www/html# ^C
root@ip-172-31-25-181:/var/www/html# ls
index.html  index.nginx-debian.html
root@ip-172-31-25-181:/var/www/html#
```

Step 25: To check where the apache2 server is installed or not, Simply paste the public instance IP on our web browser and search it.

If the address got displayed by apache page then it is installed successfully.

Apache2 Default Page

Ubuntu **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

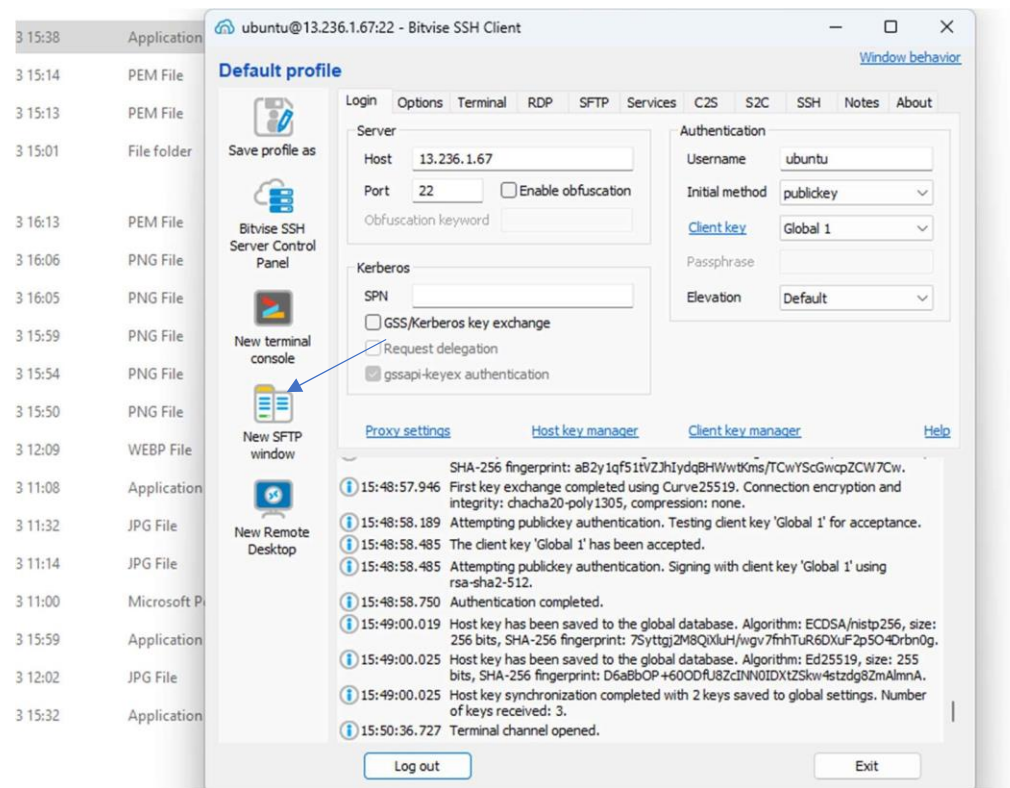
Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. 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:

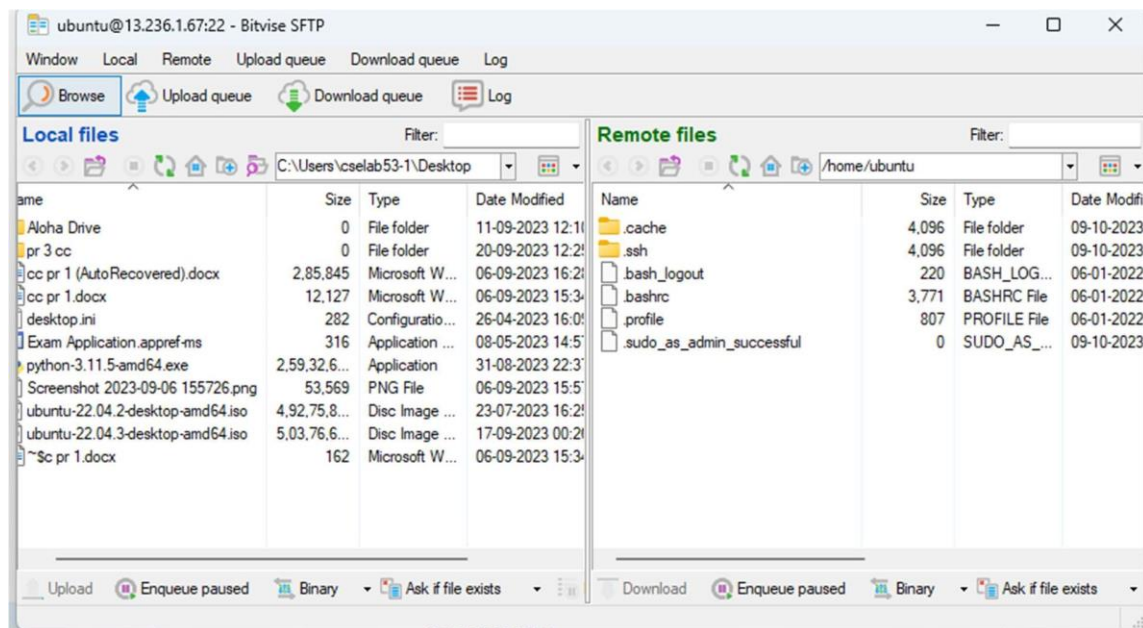
```
/etc/apache2/
|-- apache2.conf
|   |-- ports.conf
|-- mods-enabled
|   |-- *.load
|   |-- *.conf
|-- conf-enabled
|   |-- *.conf
|-- sites-enabled
|   |-- *.conf
```

- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
- Configuration files in the `mods-enabled/`, `conf-enabled/` and `sites-enabled/` directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.
- They are activated by symlinking available configuration files from their respective `*-available/` counterparts. These should be managed by using our helpers `a2enmod`, `a2dismod`, `a2ensite`, `a2dissite`, and `a2enconf`, `a2disconf`. See their respective man pages for detailed information.
- The binary is called `apache2` and is managed using `systemd` or `init` to start/stop the service.

Step 26 : Then click on the ‘New SFTP window’ button on Bitvise SSH client

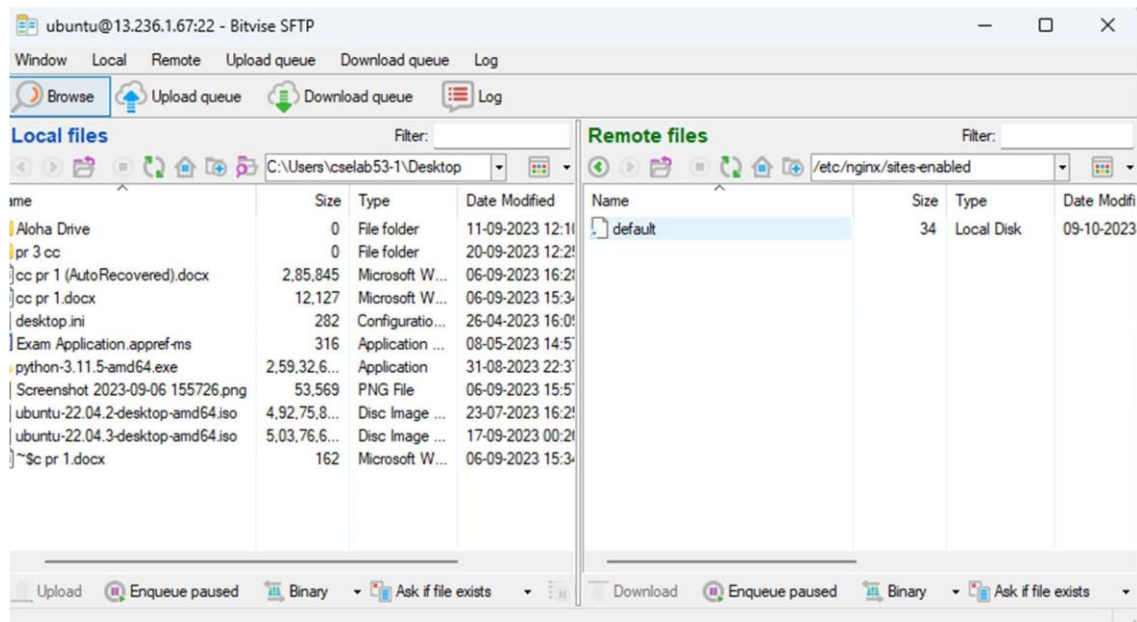


After clicking on new SFTP window the file transfer protocol will open.



Step 27 Then at search bar of SFTP open the path as:

/etc/nginx/sites-enabled



In the /etc/nginx/sites-enabled there is only one file is available that default file

So /etc/nginx/sites-enabled folder having the restriction to the user to perform read and write operations on it.

Step 28: Then to exceed restrictions open /etc/nginx directory at root user for this firstly use

use following command to go in root again `sudo`

`su -`

and then use `cd`


`/etc/nginx`

```
root@ip-172-31-25-181:/var/www/html# sudo su -
root@ip-172-31-25-181:~# cd /etc/nginx
root@ip-172-31-25-181:/etc/nginx#
```

:

Step 29 Use the following command for giving the permission to /etc/nginx.

```
sudo chmod 777
```

A terminal window screenshot showing a command being executed. The top line shows the user 'ubuntu' at IP '13.232.61.161:22' connected via 'Bitvise xterm' to a remote host 'root@ip-172-31-46-157: /etc/nginx'. The second line shows the prompt 'root@ip-172-31-46-157:/etc/nginx#' followed by the command 'sudo chmod 777 *' which is highlighted in black.

```
ubuntu@13.232.61.161:22 - Bitvise xterm - root@ip-172-31-46-157: /etc/nginx  
root@ip-172-31-46-157:/etc/nginx# sudo chmod 777 *
```

Now /etc/nginx root is allowing to perform read and write operations on it.

We can check that the read and write operations are allowed or not by creating file with .conf extension.

Step 30: To delete the index.html file from the /var/www/html folder use following command.

```
rm index.html
```

The index.html file is deleted now

To see whether the index.html file is deleted or not, use list command again

```
ls
```

```
root@ip-172-31-25-181:~# cd /var/www/html
root@ip-172-31-25-181:/var/www/html# ls
index.html  index.nginx-debian.html
root@ip-172-31-25-181:/var/www/html# rm index.html
root@ip-172-31-25-181:/var/www/html# ls
index.nginx-debian.html
root@ip-172-31-25-181:/var/www/html#
```

We can see that index.html is now deleted.

Step 31: Now create a new as “index.html” file again. For this use following command

```
sudo nano index.html
```

```
root@ip-172-31-25-181:/var/www/html# sudo nano index.html
```

Step 32: Now the blank index.html is getting created. So we can add our html content over here.

Example:

```
<HTML>

<center>

<h1> M S BIDVE ENGINEERING COLLEGE, LATUR </h1>

<br>

<br>

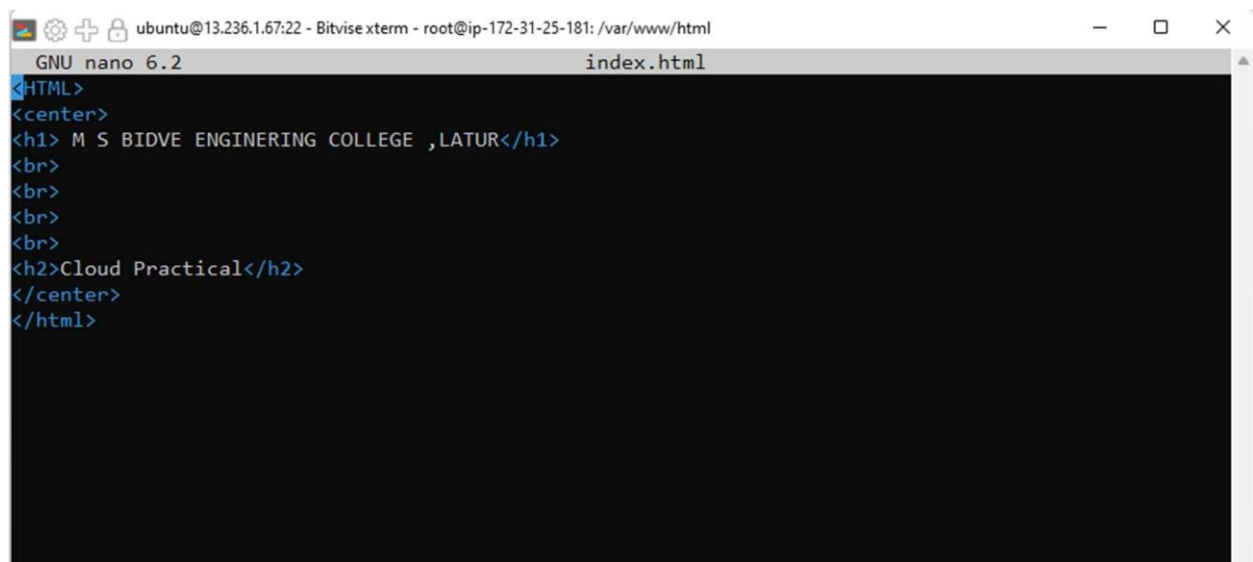
<br>

<br>

<h2>Cloud Practical </h2>

</center>

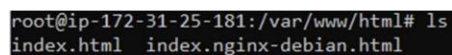
</html>
```



The screenshot shows a terminal window titled 'ubuntu@13.236.1.67:22 - Bitvise xterm - root@ip-172-31-25-181: /var/www/html'. The window contains the GNU nano 6.2 text editor editing the file 'index.html'. The code entered in the editor is: <HTML>, <center>, <h1> M S BIDVE ENGINEERING COLLEGE ,LATUR</h1>,
,
,
,
, <h2>Cloud Practical</h2>, </center>, and </html>.

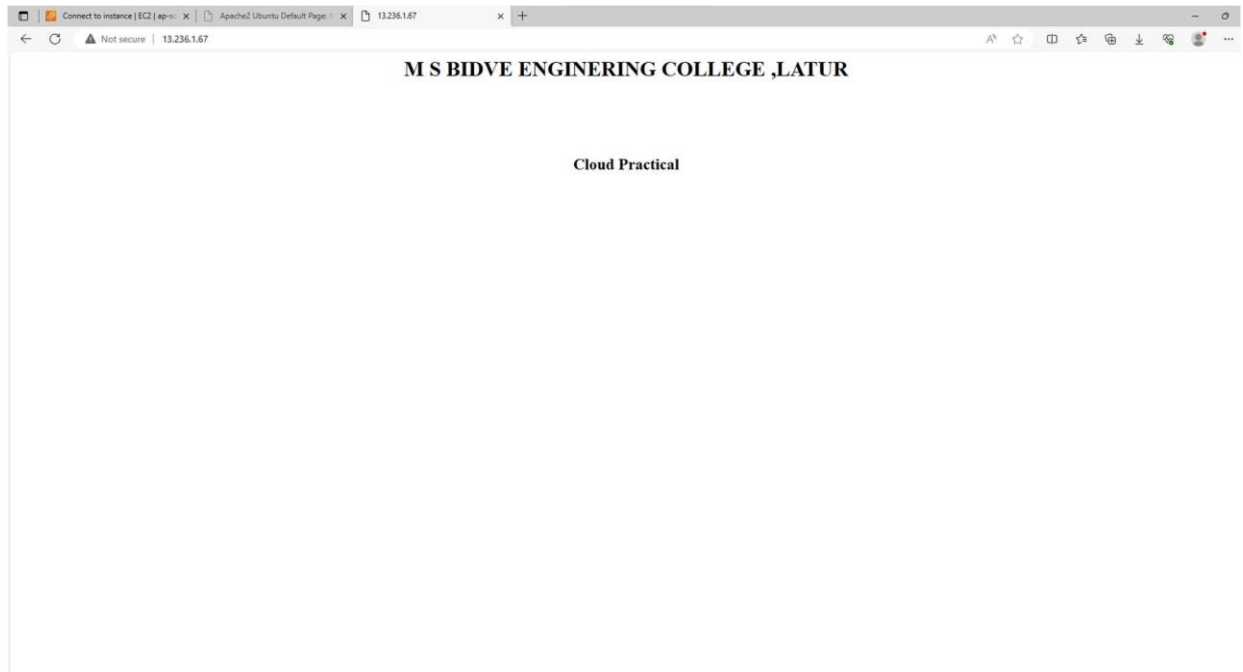
Save this code in index.html for this use ctrl + X command (^X) on the index.html

So now the index.html file is sorted.



The screenshot shows a terminal window with the command 'ls' being executed in the directory '/var/www/html'. The output of the command is 'index.html index.nginx-debian.html'.

Step 33: So to access our web site paste your instance IP address in web browser



Conclusion: We are able to perform the Provisioning and Scaling of the websites using AWS