

Hands on with Cloud Computing

11th & 12th -November-2021

—

Created by:

Ankit Velani, Sr. Consultant
Deloitte India, Bangalore

Submitted at:

Department Of MCA
Siddaganga Institute of Technology, Tumkuru

Overview

This document contains steps to set up various cloud services on the Amazon Web Service (AWS). Document divides in various sections, Create Amazon account, Setup EC-2 on Amazon, Accessing Amazon EC2 machine using FileZilla and SSH, Installing Softwares (PHP, MySQL, R) on EC-2 and last Working with Amazon RDS.

Prerequisites

- Audience should have clear understanding on Computer Networking.
- Audience should know installing and uninstalling software in Computer.
- Audience should know any programming and DBMS.

Create Amazon Account

1. Open an amazon portal : <https://aws.amazon.com/free/>

2. Click on *Create a Free account*

3. Enter **email** address and **password** for AWS account creation.
4. Enter your personal details to complete the AWS account.



Sign up for AWS

Free Tier offers

All AWS accounts can explore 5 different types of free offers, depending on the product used.

- Always free** Never expires
- 12 months free** Start from initial sign-up date
- Trials** Start from service activation date

Contact Information

How do you plan to use AWS?

- Business - for your work, school, or organization
- Personal - for your own projects

Who should we contact about this account?

Full Name:

Phone Number: Enter your country code and your phone number.
+1 222-333-4444

Country or Region: United States

Address:
Apartment, suite, unit, building, floor, etc.

City:

State, Province, or Region:

Postal Code:

I have read and agree to the terms of the AWS Customer Agreement [\[?\]](#)

Continue (step 2 of 5)

5. Click on **Create Account and Continue**
6. Payment Details. Amazon will verify your credit/debit card with minimal amount which is Rs. 2 , will be refund in 3-5 business day.

Sign up for AWS

Secure verification

We will not charge for usage below AWS Free Tier limits. We temporarily hold INR 2 as a pending transaction for 3-5 days to verify your identity.

Billing Information

Credit or Debit card number: VISA MASTERCARD

AWS accepts all major credit and debit cards. To learn more about payment options, review our [FAQ](#)

Expiration date: Month: Year:

Cardholder's name:

CVV:

Billing address:

Use my contact address
Bangalore
Bangalore Karnataka 560068
IN

Use a new address

Do you have a PAN? Permanent Account Number (PAN) is a ten-digit alphanumeric number issued by the Indian Income Tax Department. This 10-digit number is printed on the front of your PAN card.

Yes

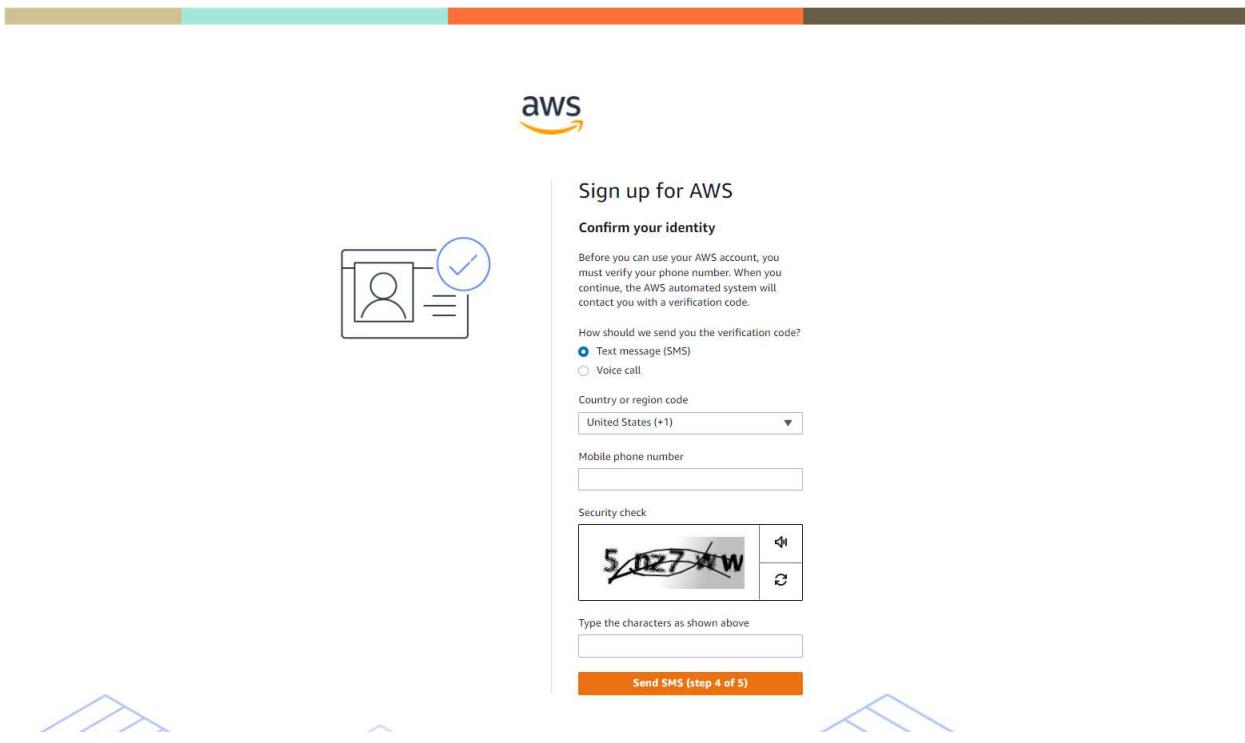
No

You can go on the Tax Settings Page on Billing and Cost Management Console to update your PAN information.

Verify and Continue (step 3 of 5)

You might be redirected to your bank's website to authorize the verification charge.

7. Enter Create/Debit card details and click on **Secure Submit** for Payment.
8. Next Step, Amazon will verify the mobile number.



Sign up for AWS

Confirm your identity

Verify code

Continue (step 4 of 5)

Having trouble? Sometimes it takes up to 10 minutes to retrieve a verification code. If it's been longer than that, [return to the previous page](#) and try again.

Account verification may done either OTP/ by Call. Please verify your account after providing your mobile number.

9. Click on **Continue**,

10. Once verification finish, Amazon allow us to **choose the plan** for Cloud Services.



Sign up for AWS

Select a support plan

Choose a support plan for your business or personal account. [Compare plans and pricing examples](#)
 You can change your plan anytime in the AWS Management Console.

<input checked="" type="radio"/> Basic support - Free <ul style="list-style-type: none"> Recommended for new users just getting started with AWS 24x7 self-service access to AWS resources For account and billing issues only Access to Personal Health Dashboard & Trusted Advisor 	<input type="radio"/> Developer support - From \$29/month <ul style="list-style-type: none"> Recommended for developers experimenting with AWS Email access to AWS Support during business hours 12 (business)-hour response times 	<input type="radio"/> Business support - From \$100/month <ul style="list-style-type: none"> Recommended for running production workloads on AWS 24x7 tech support via email, phone, and chat 1-hour response times Full set of Trusted Advisor best-practice recommendations 
---	--	--



Need Enterprise level support?

From \$15,000 a month you will receive 15-minute response times and concierge-style experience with an assigned Technical Account Manager. [Learn more](#)

[Complete sign up](#)

11. Chose the Basic plan (**Free**).

12. Click on Complete sign up.



Congratulations

Thank you for signing up for AWS.

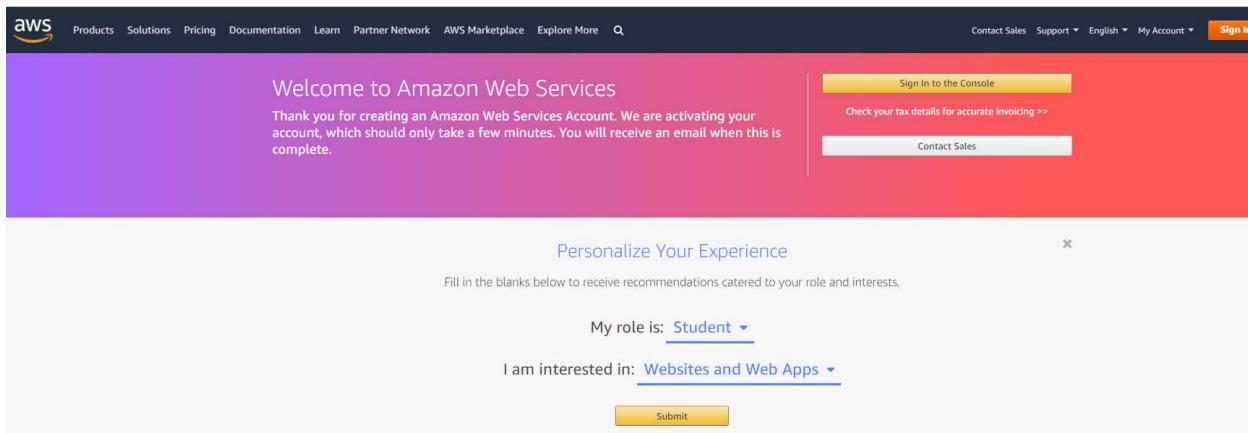
We are activating your account, which should only take a few minutes. You will receive an email when this is complete.

[Go to the AWS Management Console](#)

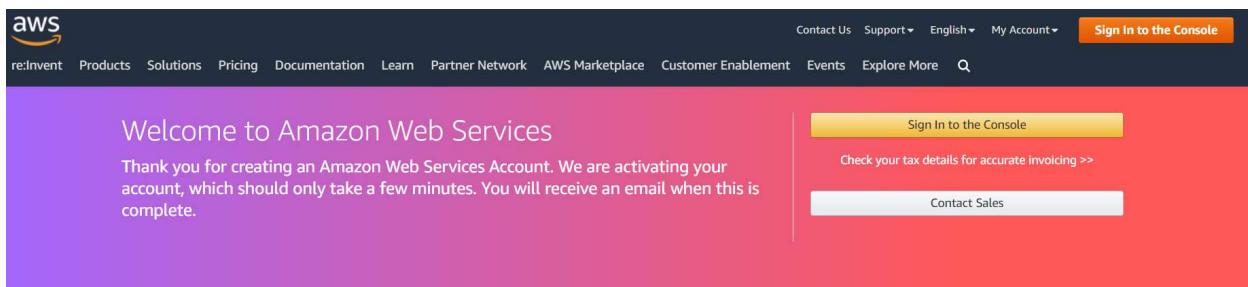
[Sign up for another account or contact sales.](#)



13. Congratulations screen will appear after successful registration.
 14. Click on Go to AWS Management Console.
 15. Next it will ask for your role for providing personalized experience.



16. Click on Submit , after selecting your role.



Thank You

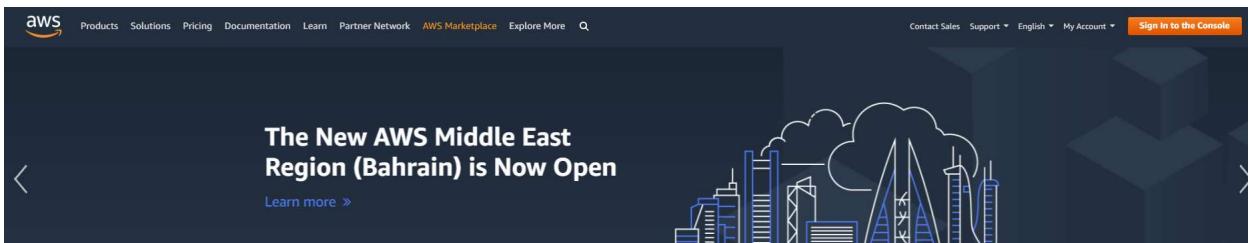
17. Congratulation, you have successfully registered with Amazon Cloud Platform.

Setup EC-2 on Amazon Cloud.

What is Elastic Compute Cloud (EC2)?

- Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers.
- Amazon EC2's simple web service interface allows you to obtain and configure capacity with minimal friction. It provides you with complete control of your computing resources and lets you run on Amazon's proven computing environment.
- Amazon EC2 reduces the time required to obtain and boot new server instances to minutes, allowing you to quickly scale capacity, both up and down, as your computing requirements change.
- Amazon EC2 changes the economics of computing by allowing you to pay only for capacity that you actually use. Amazon EC2 provides developers the tools to build failure resilient applications and isolate them from common failure scenarios.

1. Login to Amazon cloud portal. <https://aws.amazon.com/>



Click on Sign in to the Console.

2. Select Root user option and enter registered email then click on **Next**.

The screenshot shows the AWS sign-in page. It features the AWS logo at the top left. Below it, the heading "Sign in" is displayed. There are two radio button options: "Root user" (selected) and "IAM user". The "Root user" option is described as "Account owner that performs tasks requiring unrestricted access" with a "Learn more" link. The "IAM user" option is described as "User within an account that performs daily tasks" with a "Learn more" link. Below these options is a field labeled "Root user email address" containing the value "username@example.com". A blue "Next" button is positioned below the email field. At the bottom of the page, there is a note about agreeing to the AWS Customer Agreement and Privacy Notice, followed by a "New to AWS?" link and a "Create a new AWS account" button.





Security check

Type the characters seen in the image below




Root user sign in

Email: ankitkumarsvelani@gmail.com

Password	Forgot password?
<input type="password" value="*****"/>	
<input type="button" value="Sign in"/>	

[Sign in to a different account](#)

[Create a new AWS account](#)



3. Enter your password and click on **Sign in**
4. After successfully login, Console landing page will appear.

5. Click on All Services ,

6. Choose EC2 , from category " Compute "

The screenshot shows the AWS Home page with a search bar at the top. Below it, there's a section for 'Recently visited services' and 'All services'. Under 'Compute', 'EC2' is highlighted. Other services listed include Developer Tools (CodeStar, CodeCommit, CodeBuild, CodeDeploy, CodePipeline, X-Ray), Internet of Things (AWS IoT, AWS Greengrass), Contact Center (Amazon Connect), Storage (S3, EFS, Glacier, Storage Gateway), Management Tools (CloudWatch, CloudFormation, CloudTrail, Config, OpsWorks, Service Catalog, Trusted Advisor), Game Development (Amazon GameLift), Mobile Services (Mobile Hub, Cognito, Device Farm, Mobile Analytics, Pinpoint), and Database (Amazon RDS). To the right, there's a 'Helpful tips' section with links to 'Manage your costs' and 'Create an organization', and an 'Explore AWS' section with links to 'New Product Announcements' and 'Migrate from Oracle to Amazon Aurora'.

7. Let's jumped to EC2 Dashboard. Click on [Launch Instance](#)

The screenshot shows the EC2 Dashboard. On the left, there's a sidebar with navigation links like 'EC2 Dashboard', 'Instances', 'Images', and 'Elastic Block Store'. The main area has a 'Resources' section showing counts for Instances (running), Dedicated Hosts, Elastic IPs, Instances, Key pairs, Load balancers, Placement groups, Security groups, Snapshots, and Volumes. Below this is a 'Launch instance' button with the text 'To get started, launch an Amazon EC2 instance, which is a virtual server in the cloud.' To the right, there's an 'Account attributes' section with 'Supported platforms' (VPC), 'Default VPC' (vpc-0e914769f12598dde), and other settings like 'EBS encryption', 'Zones', 'EC2 Serial Console', 'Default credit specification', and 'Console experiments'. At the bottom right, there's an 'Explore AWS' section with a link to 'Enable Best Price-Performance with AWS Graviton2'.

8. Choose Amazon Machine Image (AMI), for learning choose **Free tier eligible**.

Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Quick Start

1 to 18 of 18 AMIs

Image	Name	Description	Select	Architecture
Amazon Linux icon	Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-0cf31d971a3ca20d6	Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras.	Select	64-bit
Amazon Linux icon	Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type - ami-0b59bfac6be064b78	The Amazon Linux AMI is an EBS-backed, AWS-supported image. The default image includes AWS command line tools, Python, Ruby, Perl, and Java. The repositories include Docker, PHP, MySQL, PostgreSQL, and other packages.	Select	64-bit
Red Hat icon	Red Hat Enterprise Linux 7.5 (HVM), SSD Volume Type - ami-03291866	Red Hat Enterprise Linux version 7.5 (HVM), EBS General Purpose (SSD) Volume Type	Select	64-bit

9. Select AMI type Ubuntu **Server 18.04 LTS (HVM)**, click on Select.

10. Next step, choose instance type. Select the **t2.mirco (Free tier eligible)**

Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. [Learn more](#) about Instance types and how they can meet your computing needs.

Filter by:	All instance types	Current generation	Show/Hide Columns					
Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)								
	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="checkbox"/>	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	General purpose	t2.micro	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.large	2	8	FRS only	-	Low to Moderate	Yes

Review and Launch

T2.mirco instance configuration.

Variable ECUs,

1 vCPUs

2.5 GHZ , Intel Xeon Family

1 GiB memory RAM

Check the free services

<https://aws.amazon.com/free/#legal>

11. Click on Review and Launch.

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

AMI Details

Ubuntu Server 14.04 LTS (HVM), SSD Volume Type - ami-7d132e18
Free tier eligible

Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
I2.micro	Variable	1	1	EBS only	-	Low to Moderate

Security Groups

Security group name: launch-wizard-1
Description: launch-wizard-1 created 2018-08-20T13:28:10.802+05:30

Buttons: Cancel, Previous, Launch

12. Click on Launch.

13. Next Screen, It will ask to create Key for accessing EC-2 Machine.

Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about [removing existing key pairs from a public AMI](#).

Choose an existing key pair
Select a key pair
No key pairs found

No key pairs found
You don't have any key pairs. Please create a new key pair by selecting the [Create a new key pair](#) option above to continue.

Buttons: Cancel, Launch Instances

14. Select the option to create a new key pair from select box.

15. Write KeyName and Download Key Pair

16. After Download, Click on [Launch Instances](#).

The screenshot shows the AWS CloudWatch Launch Status page. At the top, there's a green banner with the message "Your instances are now launching". Below it, a blue banner says "Get notified of estimated charges". The main content area is titled "How to connect to your instances" and includes a link to "View Instances". There are also links for "Amazon EC2: User Guide" and "Amazon EC2: Discussion Forum". At the bottom right is a "View Instances" button.

17. Click on [View Instances](#).

18. Back to EC2 Dashboard, under **INSTANCES**, recently created instance will appear.

The screenshot shows the AWS EC2 Instances dashboard. On the left sidebar, under the "INSTANCES" section, the "Instances" tab is selected. In the main content area, a table lists the recently launched instance: "i-0ad2d2f2e17281327" (t2.micro, us-east-2c, running). Below the table, a detailed view for the instance "i-0ad2d2f2e17281327" is shown, including its Public DNS (ec2-52-14-56-150.us-east-2.compute.amazonaws.com) and Private IP (52.1.2.1).

19. Successfully launched t2.micro instance. We have description of an instance to access. For the password we must use Downloaded Key files. There is no option to create user defined password.

20. Description of the instance.

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links for EC2 Dashboard, Events, Tags, Reports, Limits, Instances (selected), Spot Requests, Reserved Instances, Dedicated Hosts, Images (AMIs), Bundle Tasks, Elastic Block Store (Volumes, Snapshots), Network & Security (Security Groups, Elastic IPs, Placement Groups, Key Pairs), and Feedback. The main content area has tabs for Launch Instance, Connect, Actions, and Description (selected). A search bar at the top says "Filter by tags and attributes or search by keyword". Below it is a table with columns: Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, Public DNS (IPv4), and IPv4. One row is selected, showing details for an instance named "i-0ad2d2f2e17281327" which is a t2.micro type in us-east-2c, running, with 2/2 checks passing, no alarm status, and a public DNS of "ec2-52-14-56-150.us-east-2.compute.amazonaws.com". Below this, there are tabs for Status Checks, Monitoring, and Tags. The Status Checks tab is selected, showing detailed information about the instance's health.

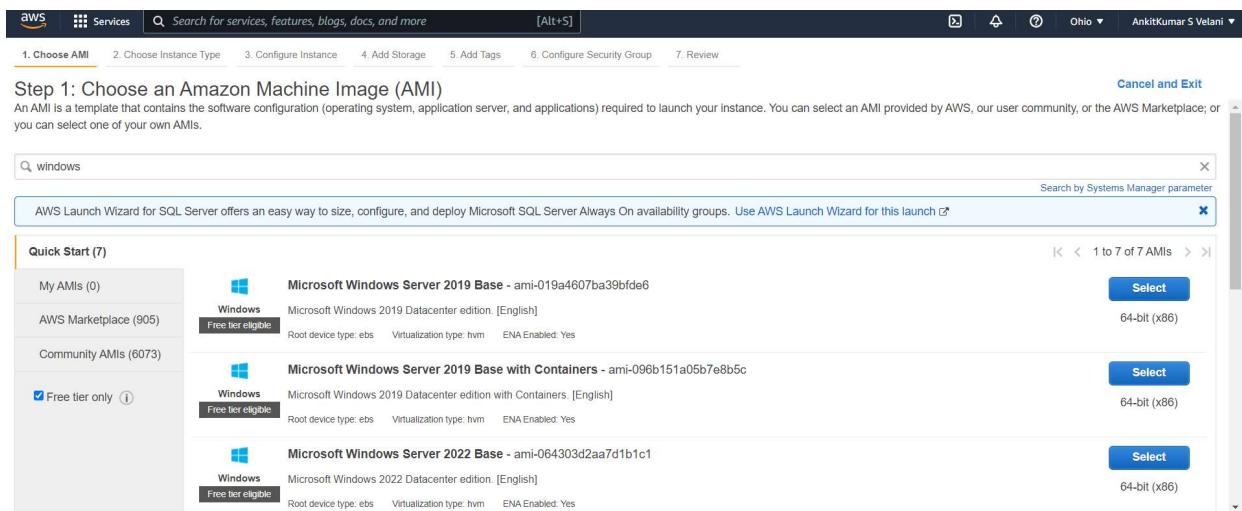
Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4
i-0ad2d2f2e17281327	t2.micro	us-east-2c	running	2/2 checks ...	None		ec2-52-14-56-150.us-e... 52.1...	

Description

Instance ID	i-0ad2d2f2e17281327	Public DNS	ec2-52-14-56-150.us-east-2.compute.amazonaws.com
Instance state	running	IPv4 Public IP	52.14.56.150
Instance type	t2.micro	IPv6 IPs	-
Elastic IPs		Private DNS	ip-172-31-32-230.us-east-2.compute.internal
Availability zone	us-east-2c	Private IPs	172.31.32.230
Security groups	launch-wizard-1, view inbound rules	Secondary private IPs	
Scheduled events	No scheduled events	VPC ID	vpc-bd2d6dd4
AMI ID	ubuntu/images/hvm-ssd/ubuntu-trusty-14.04-amd64-server-20170619 (ami-019abc64)	Subnet ID	subnet-94bd1ad9
Platform	-	Network interfaces	eth0
IAM role	-	Source/dest. check	True
Key pair name	AmazonEC2		

Setup EC2 with Windows Instance

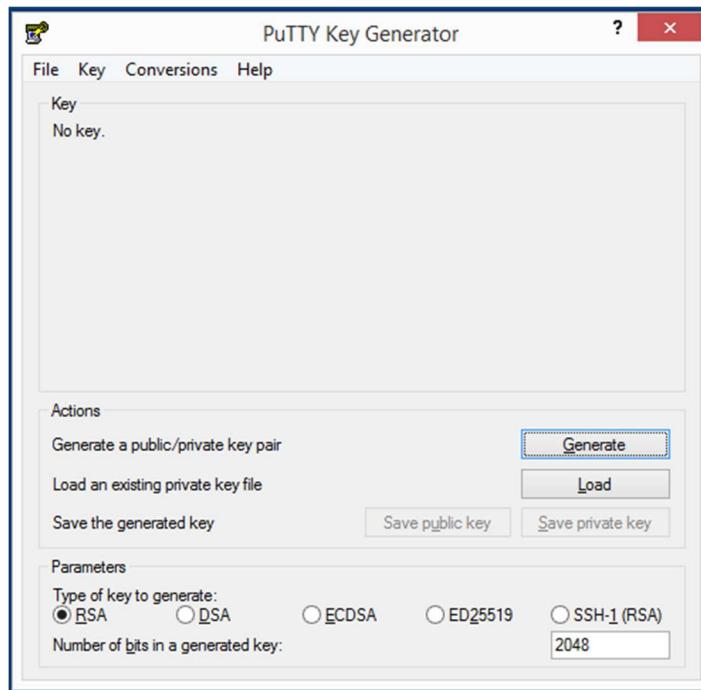
- Kindly refer the same steps which we followed for Ubuntu instance, select the Windows as instance type in **step -9**, rest steps will be the same as Ubuntu.
- Choose micro instance for learning which free-tier is.



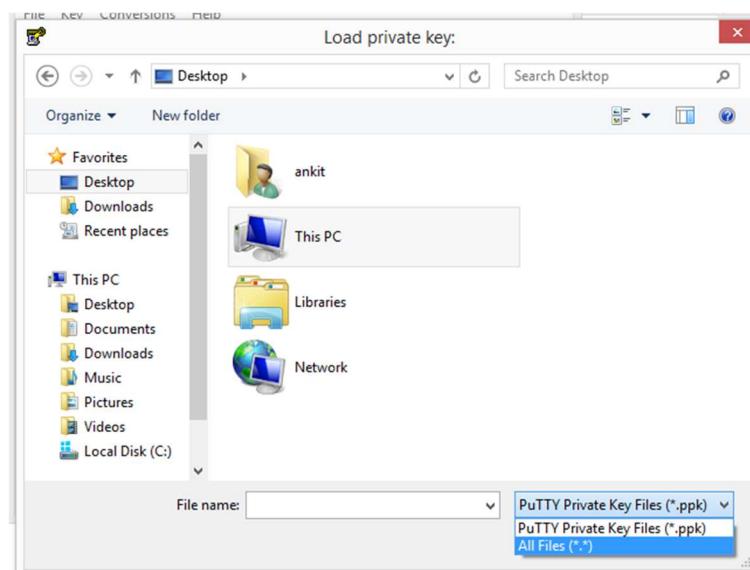
Converting Encrypted file to ppk (Private Key) to access EC2

Note: Make sure that Puttgen should be installed in your machine.

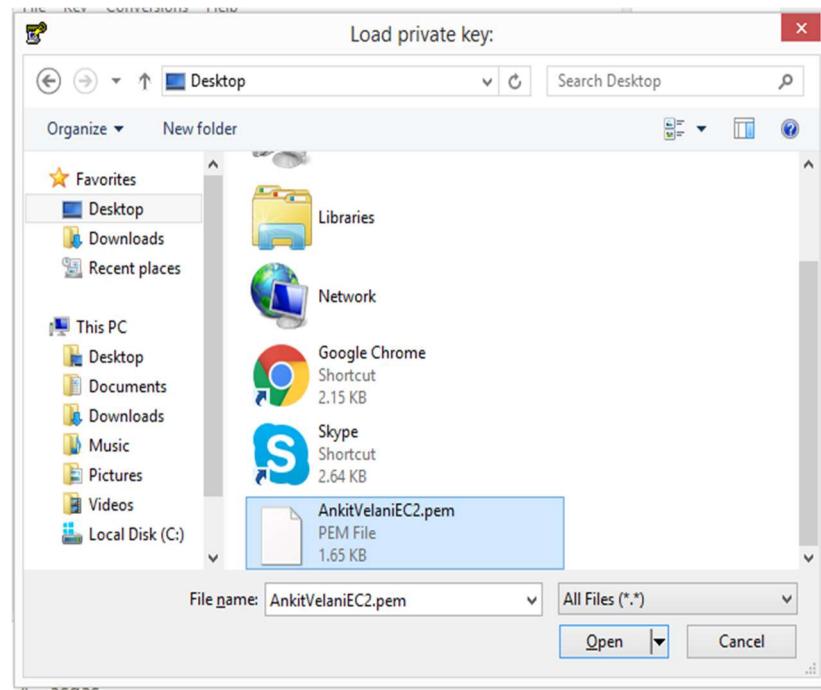
1. Open PuTTygen.



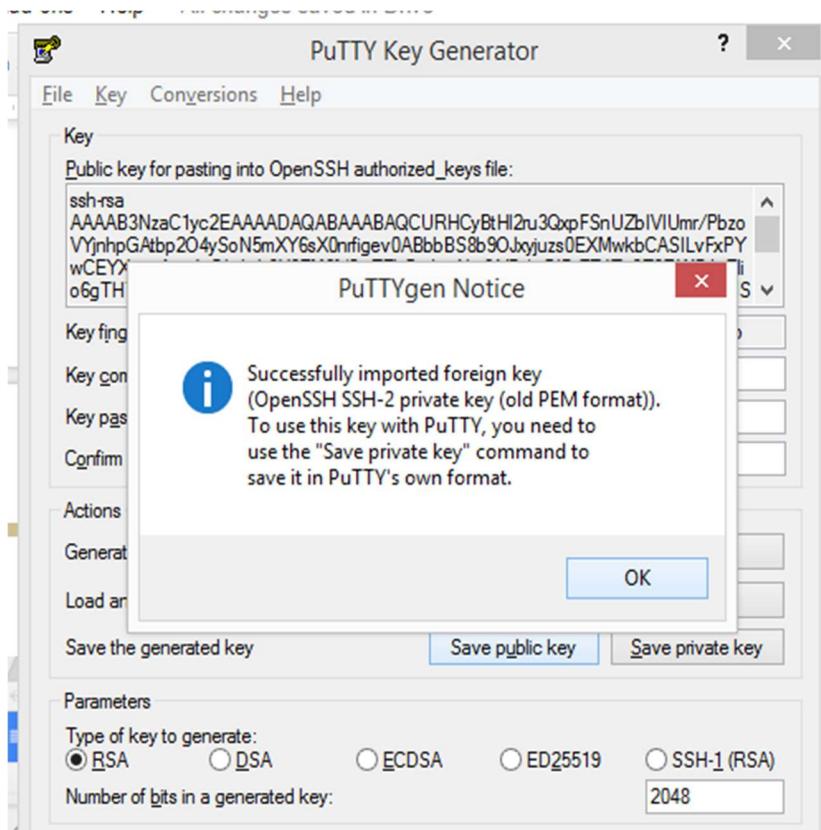
2. Click on **Load**, It gives FileBox , choose file type as **All Files(*)**



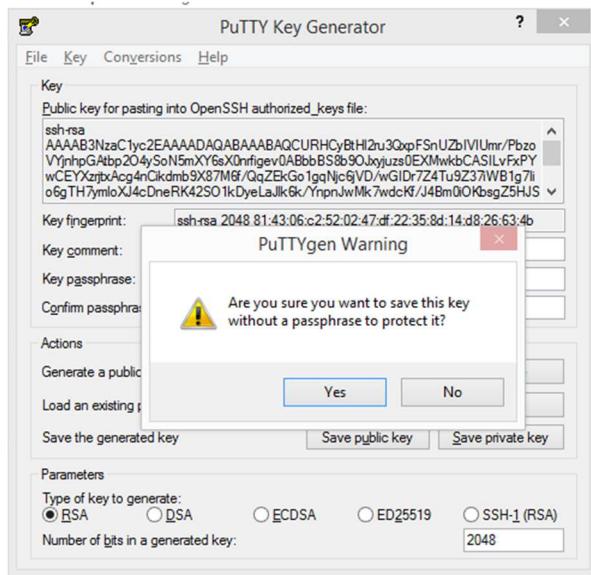
3. Select your .pem file and click on Open.



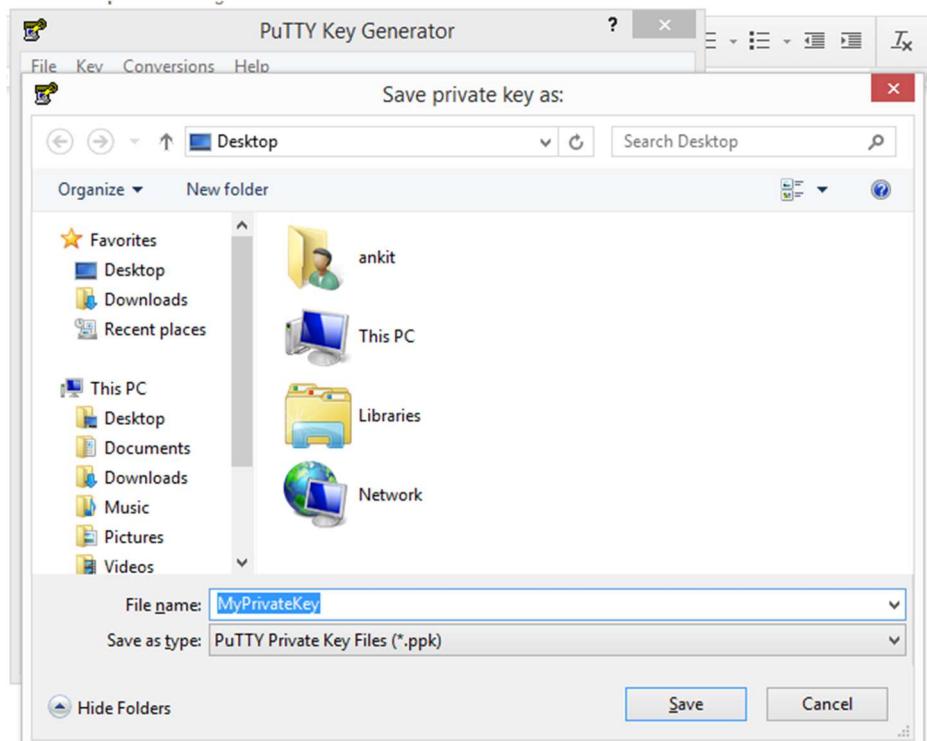
4. Click on Ok , if Putty show any notice.



5. Click on **Save private key**



6. Click on Yes.



7. File box file open and it allow to enter the filename which you want save the private key. Click on Save.
8. Close the PuttyGen software.

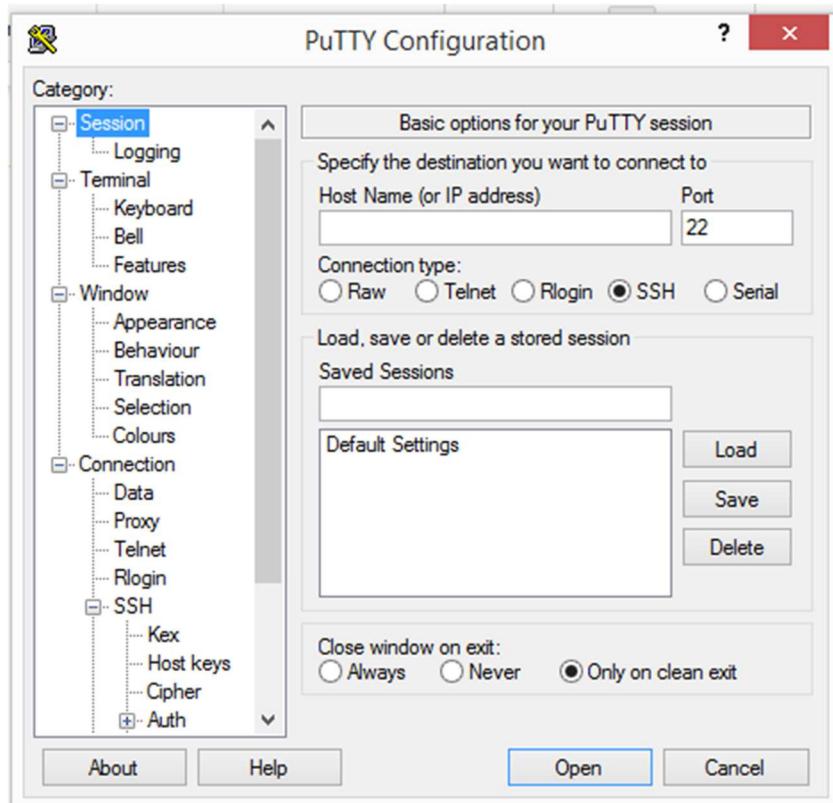
Congratulations :) Successfully created Private key(ppk) file.

Accessing Amazon EC2 Machine using SSH Client

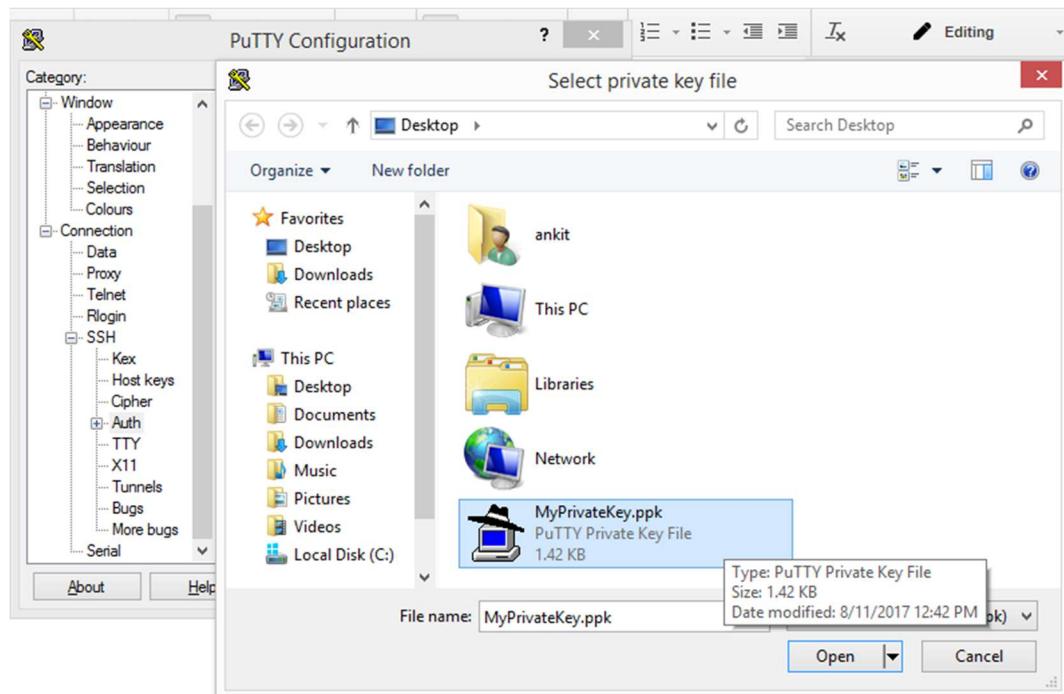
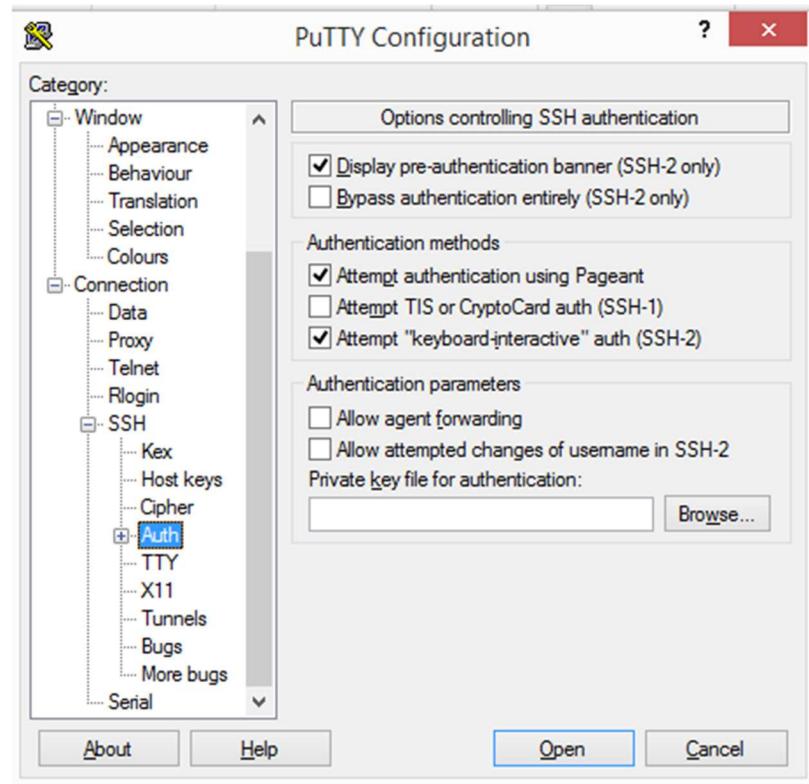
Note: Make sure that Putty ssh client should be installed in your machine.



1. Open Putty

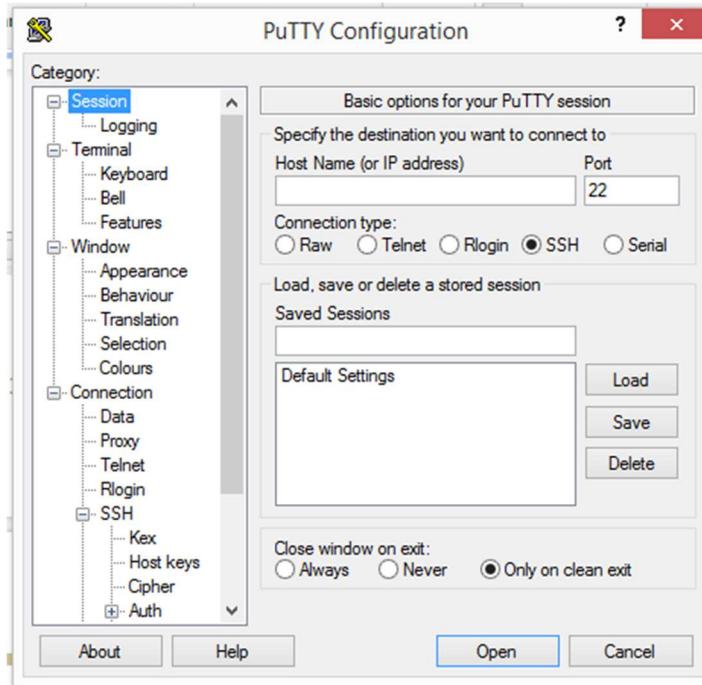


2. Go to **Connection > SSH > Auth** , Click on **Browse** , Choose private key (.ppk) file.



3. Click on open.

4. Go to Session.



5. Enter the Hostname (or IP address)

6. Open an Amazon cloud portal. <https://console.aws.amazon.com>

7. Click **EC2** under Compute Category.

8. Screen will be on EC2 Dashboard.

You are using the following Amazon EC2 resources in the US East (Ohio) region:

- 1 Running Instances
- 2 Elastic IPs
- 0 Dedicated Hosts
- 0 Snapshots
- 2 Volumes
- 0 Load Balancers
- 1 Key Pairs
- 3 Security Groups
- 0 Placement Groups

Just need a simple virtual private server? Get everything you need to jumpstart your project - compute, storage, and networking – for a low, predictable price. [Try Amazon Lightsail for free.](#)

Create Instance

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.

[Launch Instance](#)

Note: Your instances will launch in the US East (Ohio) region

Service Health

Service Status: **US East (Ohio):** This region is operating normally.

Scheduled Events

US East (Ohio): No events

AWS Marketplace

Find free software trial products in the AWS Marketplace from the [EC2 Launch Wizard](#). Or try these popular AMIs: Barracuda NextGen Firewall F-Series - PAYG

9. Click on Running Instances.

Description	
Instance ID	i-0fcfc4fc3afe2f71e
Public DNS (IPv4)	ec2-52-14-202-117.us-east-2.compute.amazonaws.com
Instance state	running
IPv4 Public IP	52.14.202.117
Instance type	t2.micro
IPv6 IPs	-
Elastic IPs	
Private DNS	ip-172-31-43-82.us-east-2.compute.internal
Availability zone	us-east-2c
Private IPs	172.31.43.82

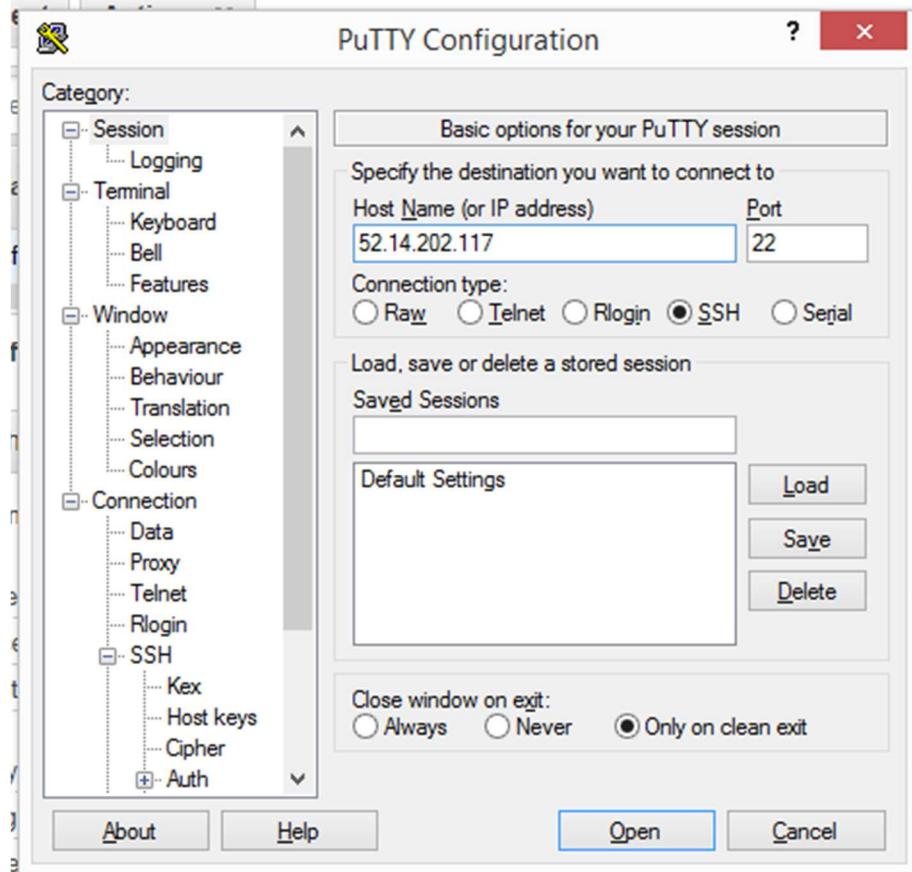
10. You will have one running instances and click on it. And you will find the Public DNS , Public IP and Private IP ...etc details.

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links like EC2 Dashboard, Events, Tags, Reports, Limits, Instances (which is selected), Spot Requests, Reserved Instances, Dedicated Hosts, Images, AMIs, Bundle Tasks, Elastic Block Store, Volumes, Snapshots, Network & Security, and Security Groups. The main content area has tabs for Launch Instance, Connect, and Actions. A search bar at the top says "Filter by tags and attributes or search by keyword". Below it is a table with columns: Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, and Public DNS. One row is visible: "i-0fcfc4fc3afe2f71e" (Instance ID), "t2.micro" (Instance Type), "us-east-2c" (Availability Zone), "running" (Instance State), "2/2 checks ... None" (Status Checks), "None" (Alarm Status), and "ec2-52-14-202-117.us-east-2.compute.amazonaws.com" (Public DNS). Below the table, there are tabs for Description, Status Checks, Monitoring, and Tags. Under the Description tab, detailed information is shown for the instance, including Instance ID, Instance state, Instance type, Elastic IPs, Availability zone, Security groups, Scheduled events, AMI ID, and various network-related fields like Public DNS (IPv4), IPv4 Public IP, IPv6 IPs, Private DNS, Private IPs, Secondary private IPs, VPC ID, and Subnet ID. At the bottom of the page, there are links for Feedback, English, and footer text: "© 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved." and "Privacy Policy Terms of Use".

11. Copy the Public IP.

Public DNS (IPv4)	ec2-52-14-202-117.us-east-2.compute.amazonaws.com
IPv4 Public IP	52.14.202.117
IPv6 IPs	-
Private DNS	ip-172-31-43-82.us-east-2.compute.internal
Private IPs	172.31.43.82
Secondary private IPs	
VPC ID	vpc-09490460
Subnet ID	subnet-2304a56e

12. Go back to Putty SSH client , enter the Hostname as a Public IP.



13. Click on Open.



14. Ask for username , login as : enter **ubuntu** as user name

15. After successfully login, will have terminal of EC2 virtual server.

```
ubuntu@ip-172-31-43-82: ~
login as: ubuntu
Authenticating with public key "imported-openssh-key"
Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 3.13.0-125-generic x86_64)

 * Documentation:  https://help.ubuntu.com/
System information as of Thu Aug 10 06:57:39 UTC 2017

System load:  0.0           Processes:      101
Usage of /:   12.4% of 7.74GB  Users logged in:    0
Memory usage: 10%
Swap usage:   0%
Graph this data and manage this system at:
https://landscape.canonical.com/

Get cloud support with Ubuntu Advantage Cloud Guest:
http://www.ubuntu.com/business/services/cloud

New release '16.04.3 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Wed Aug  9 09:48:50 2017 from 122.172.20.157
ubuntu@ip-172-31-43-82:~$
```

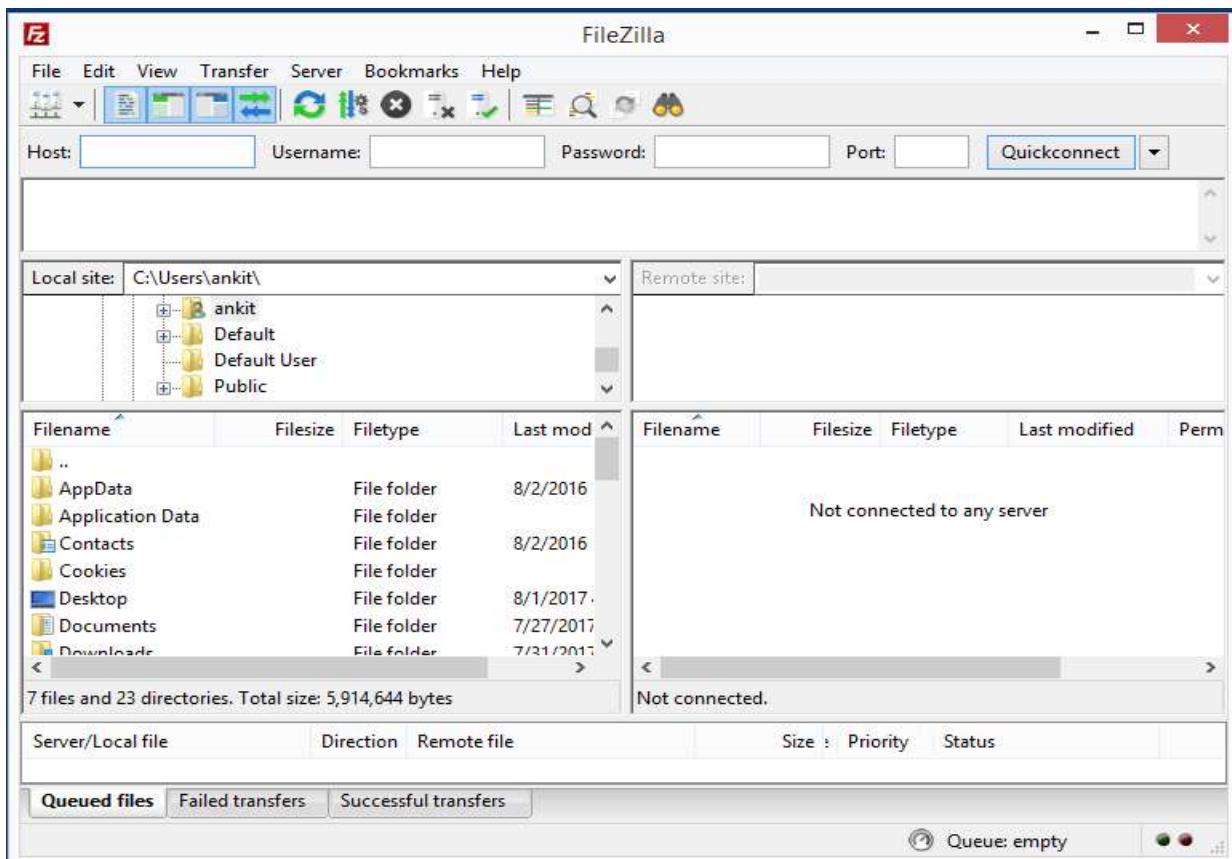
Congratulations :) you are connected to virtual machine using Putty SSH Client.

Accessing Amazon EC2 Machine using FileManager

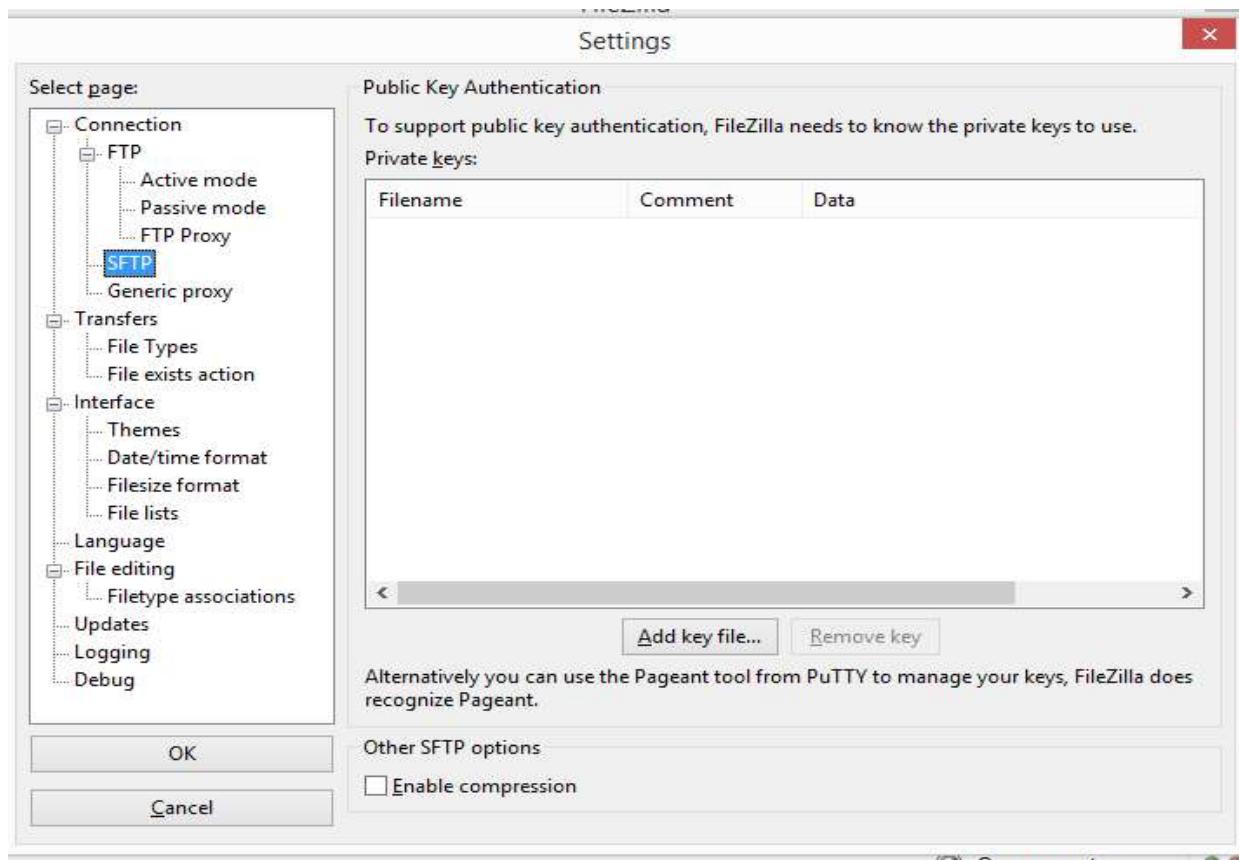
Note: Make sure that any one file manager (FileZilla, WinScp ..etc) should be installed in your machine.



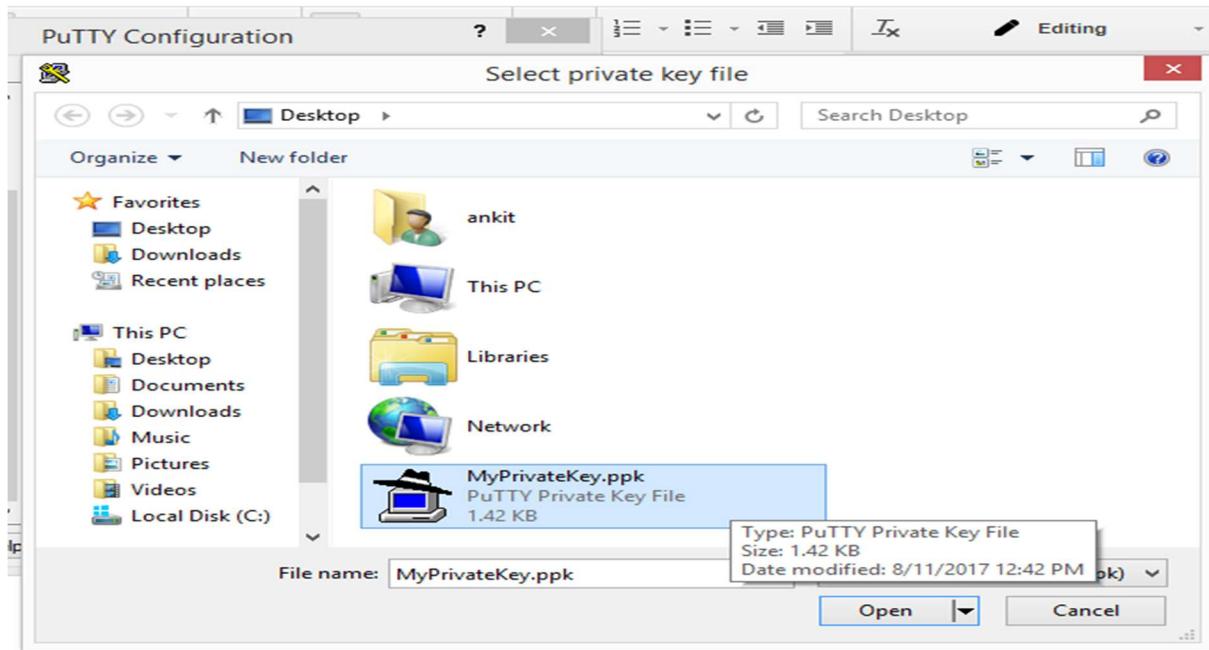
1. Open FileZilla.



2. Go to **Edit Menu > Connection > SFTP.**



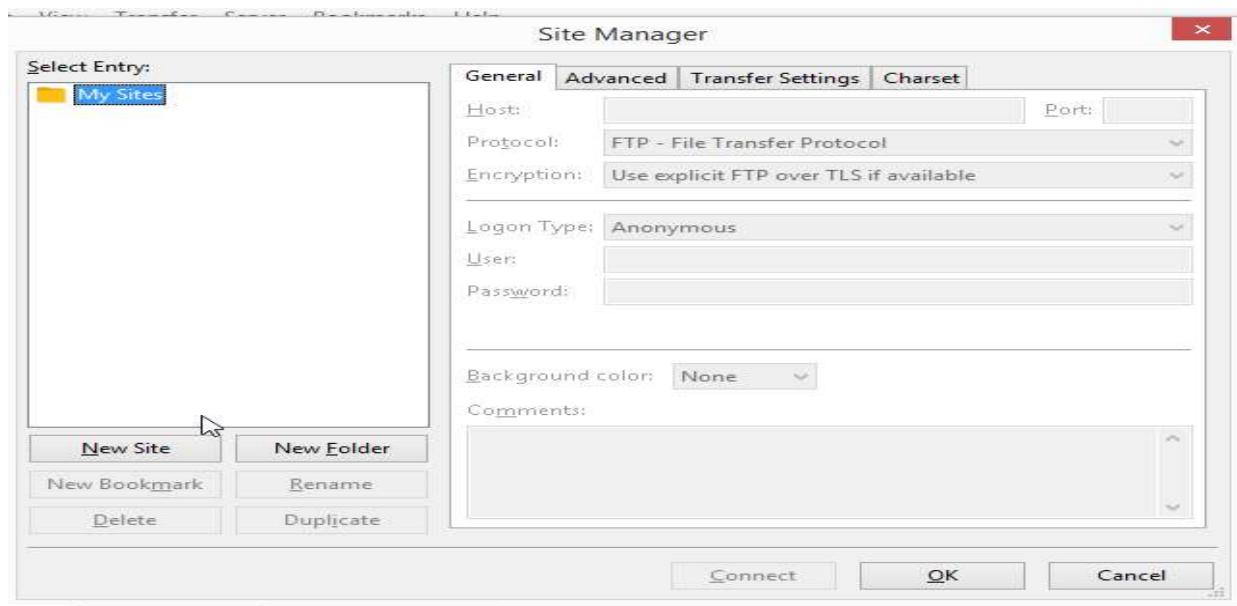
3. Click on **Add key file...**



4. Select your **.pem** or **.ppk** file and click on **Open**.

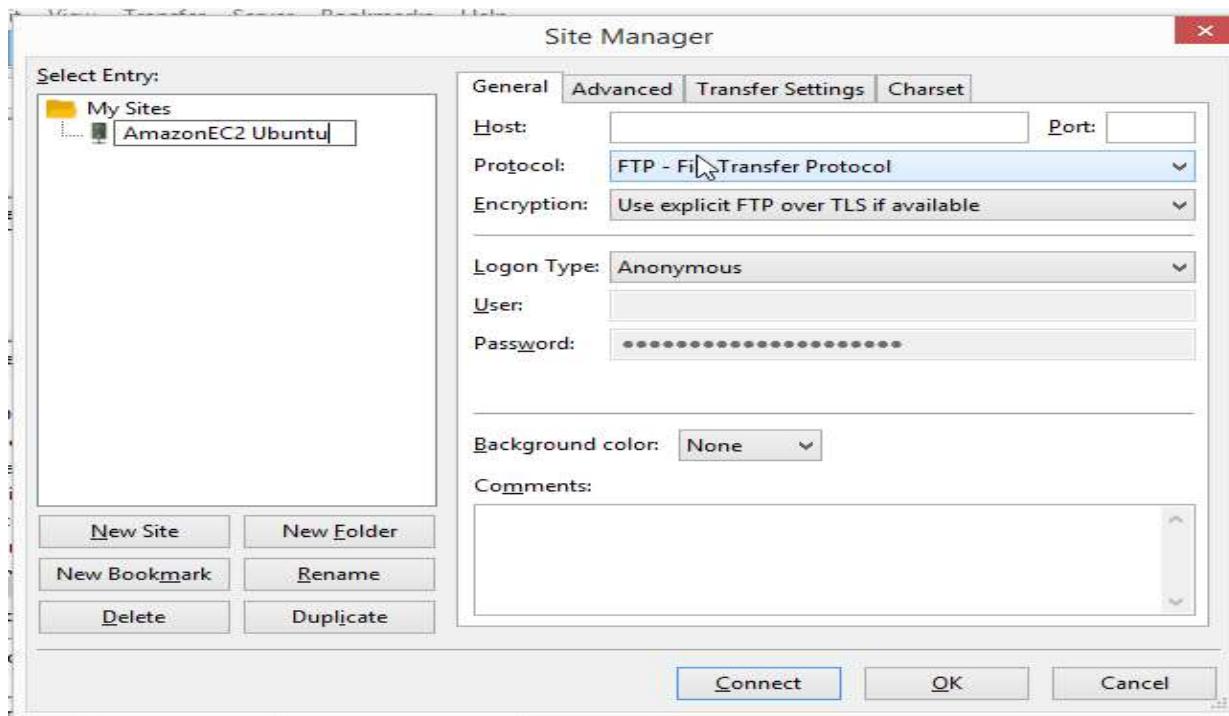
5. Click on **OK**.

6. Go to **File Menu > Site Manager**.



7. Click on **New Site**.

8. Enter Site Name as AmazonEC2 Ubuntu



9. Open an Amazon cloud portal. <https://console.aws.amazon.com>

10. Click EC2 under Compute Category.

Hands on Cloud Computing Workshop.

Dept. Of MCA – SIT, Tumkuru

11. Screen will be on EC2 Dashboard.

12. Click on Running Instances.

13. You will have one running instances and click on it. And you will find the Public DNS , Public IP and Private IP ...etc. details.

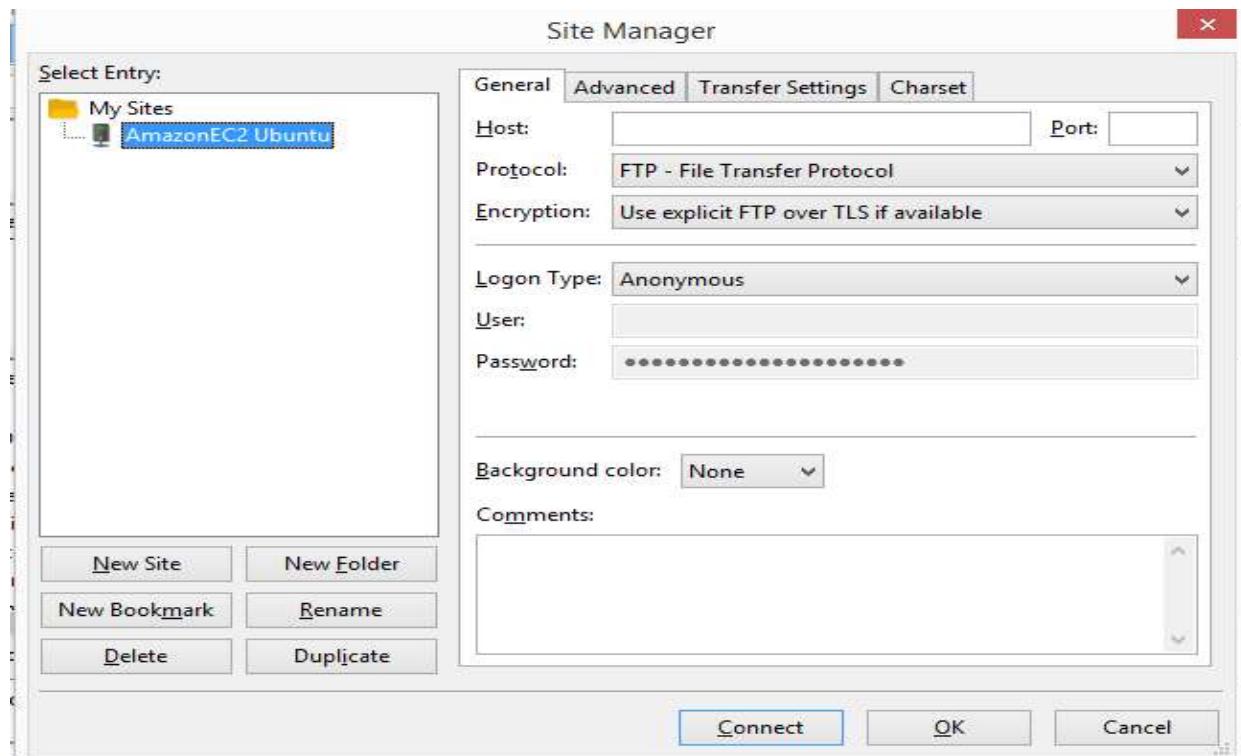
The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with links like EC2 Dashboard, Events, Tags, Reports, Limits, Instances, Spot Requests, Reserved Instances, Dedicated Hosts, Images, AMIs, Bundle Tasks, Elastic Block Store, Volumes, Snapshots, and Network & Security. The 'Instances' link is currently selected. The main content area displays a single instance named 'i-0fcfc4fc3afe2f71e'. The instance is of type 't2.micro' and is located in the 'us-east-2c' availability zone. It is currently 'running'. There are two status checks listed: '2/2 checks ...' and 'None'. The public DNS is 'ec2-52-14-202-117.us-east-2.compute.amazonaws.com'. Below the instance details, there are tabs for Description, Status Checks, Monitoring, and Tags. The 'Description' tab is active, showing detailed information such as Instance ID, Public DNS (IPv4), Instance state, IPv4 Public IP, Instance type, IPv6 IPs, Availability zone, Private DNS, Elastic IPs, Security groups, Secondary private IPs, Scheduled events, VPC ID, AMI ID, and Subnet ID.

	Value
Instance ID	i-0fcfc4fc3afe2f71e
Public DNS (IPv4)	ec2-52-14-202-117.us-east-2.compute.amazonaws.com
Instance state	running
IPv4 Public IP	52.14.202.117
Instance type	t2.micro
IPv6 IPs	-
Elastic IPs	
Availability zone	us-east-2c
Security groups	launch-wizard-1, view inbound rules
Scheduled events	No scheduled events
AMI ID	ubuntu/images/hvm-ssd/ubuntu-trusty-14.04-amd64-server-20170718 (ami-13391926)
Private DNS	ip-172-31-43-82.us-east-2.compute.internal
Private IPs	172.31.43.82
Secondary private IPs	
VPC ID	vpc-09490460
Subnet ID	subnet-2304a56e

14. Copy the Public IP.

Public DNS (IPv4)	ec2-52-14-202-117.us-east-2.compute.amazonaws.com
IPv4 Public IP	52.14.202.117
IPv6 IPs	-
Private DNS	ip-172-31-43-82.us-east-2.compute.internal
Private IPs	172.31.43.82
Secondary private IPs	
VPC ID	vpc-09490460
Subnet ID	subnet-2304a56e

15. Next , Open the FileZilla. And Go to **File Menu > SiteManager > AmazonEC2 Ubuntu**



16. Enter following details.

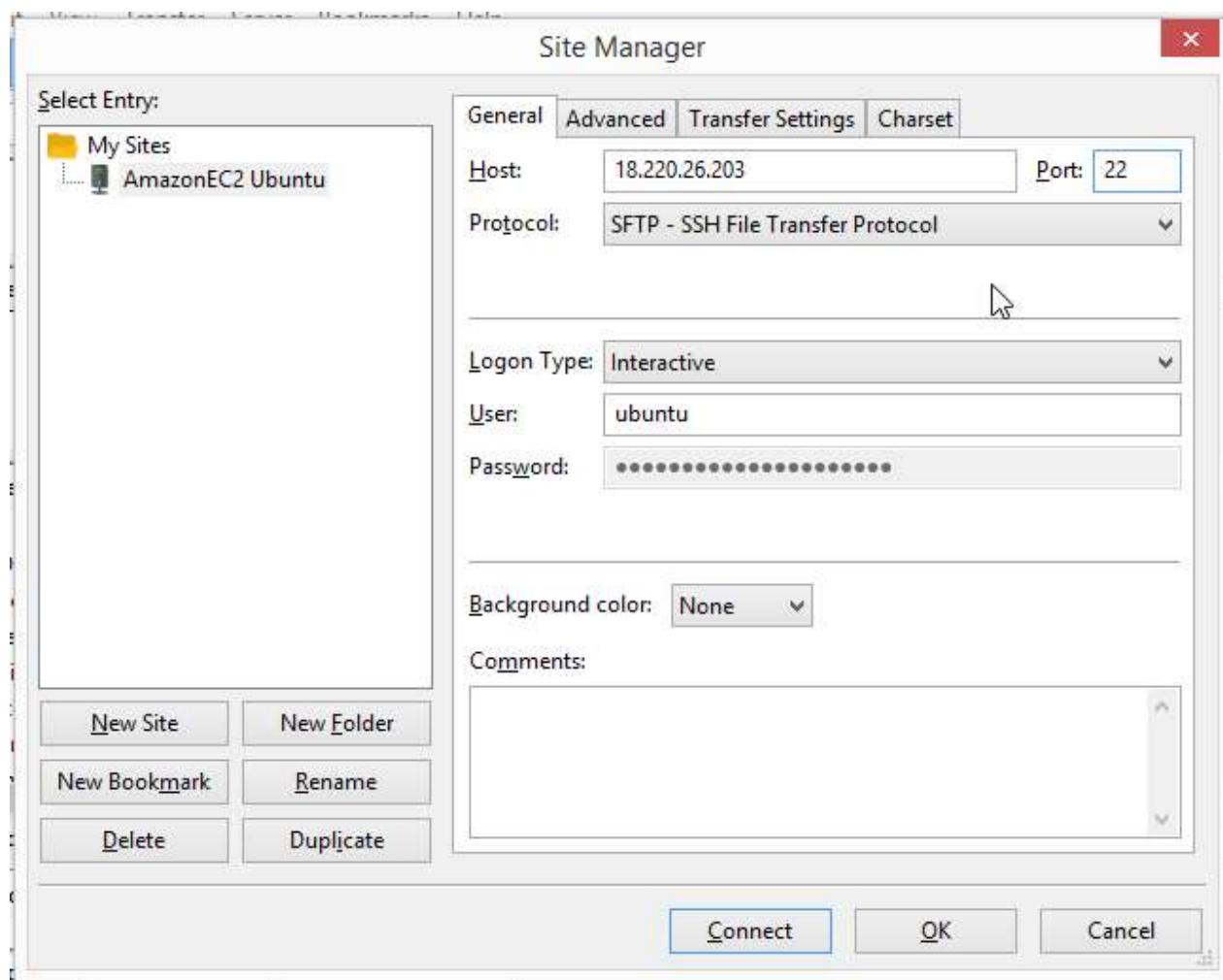
Host : Public IP

Port : 22

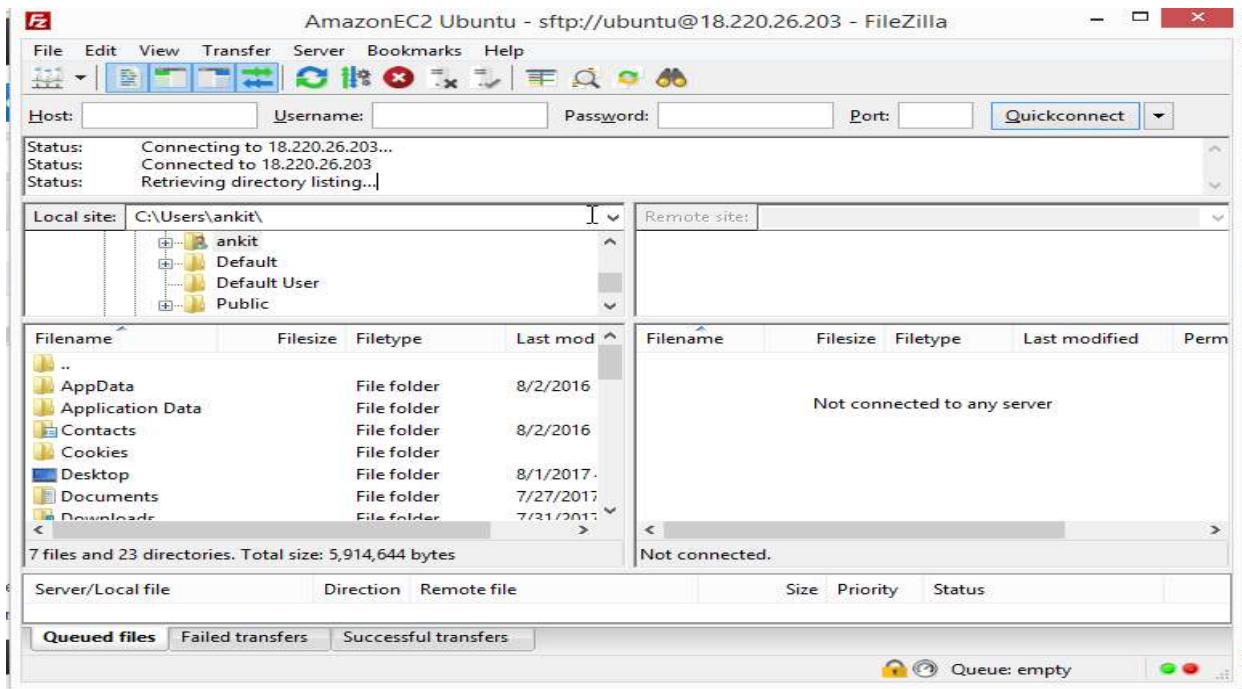
Protocol : SFTP

Logon Type : Interactive

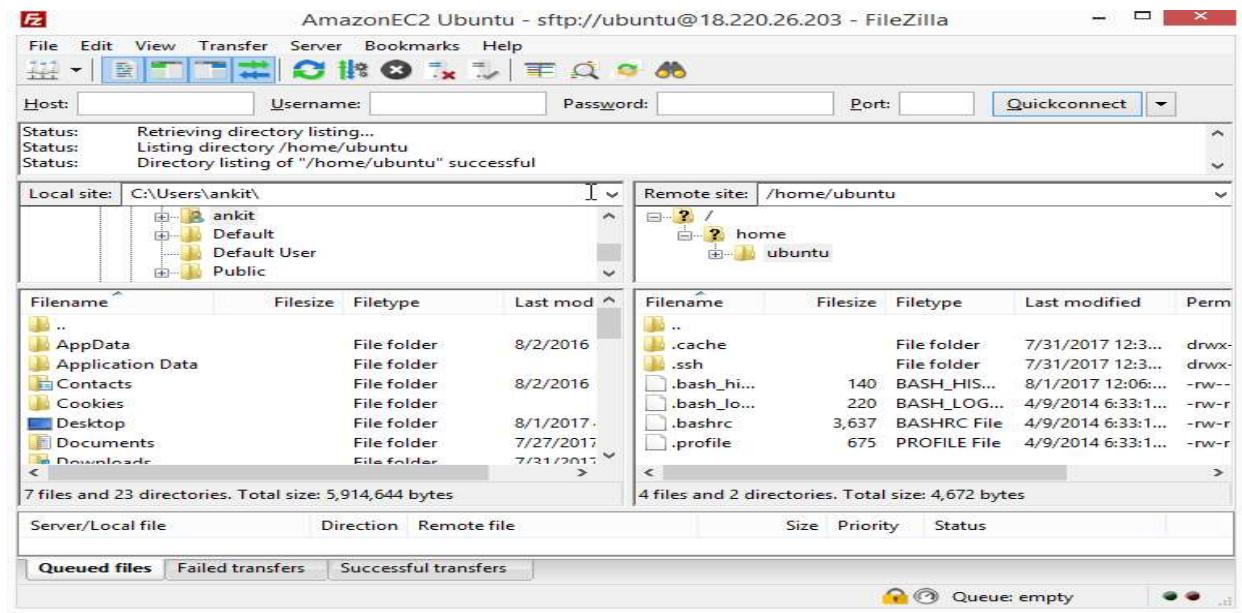
User : ubuntu



17. Click on connect and it will take you on another screen.



18. Once it successfully connected it will show the Remote Site in the right side of the box.

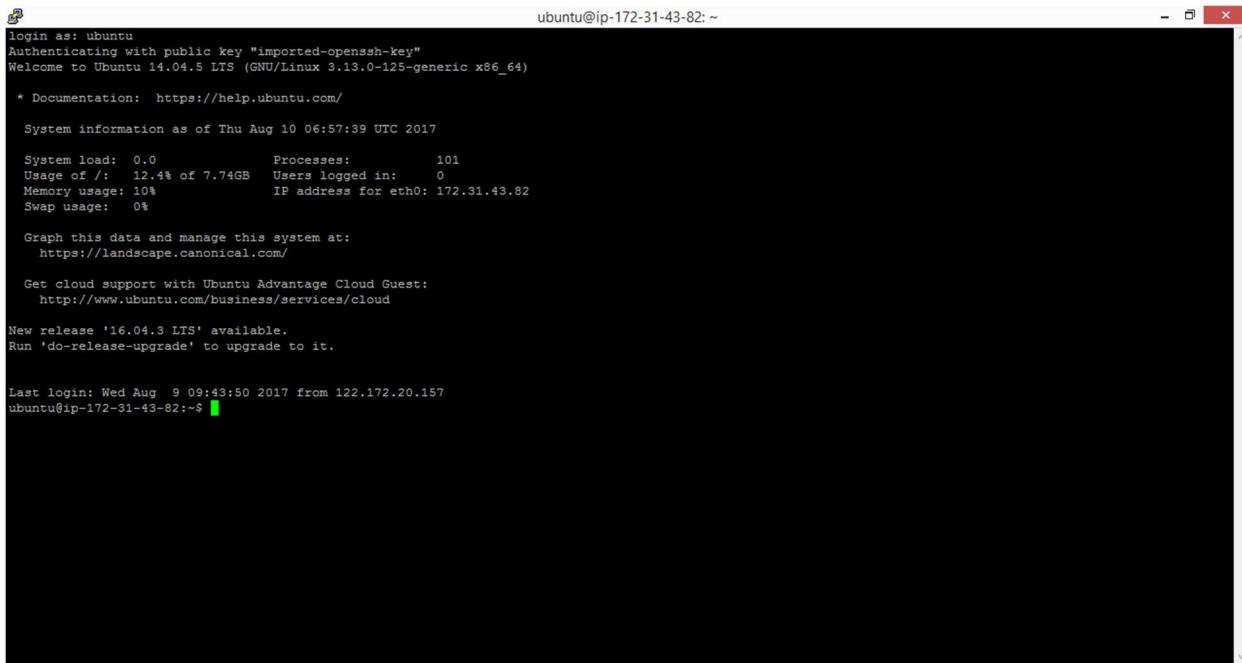


Note: You are accessing Amazon EC Virtual machine using FileZilla file manager, it's easy to download, upload and file manipulation.

Congratulations :) you are connected to virtual machine using File Manager.

Configuring Apache2 on Ubuntu 14

Note: Before you start executing command, **first connect EC-2 with Putty SSH client.**



```

login as: ubuntu
Authenticating with public key "imported-openssh-key"
Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 3.13.0-125-generic x86_64)

 * Documentation: https://help.ubuntu.com/
System information as of Thu Aug 10 06:57:39 UTC 2017

System load: 0.0          Processes:      101
Usage of /: 12.4% of 7.74GB  Users logged in:    0
Memory usage: 10%          IP address for eth0: 172.31.43.82
Swap usage:  0%

Graph this data and manage this system at:
https://landscape.canonical.com/

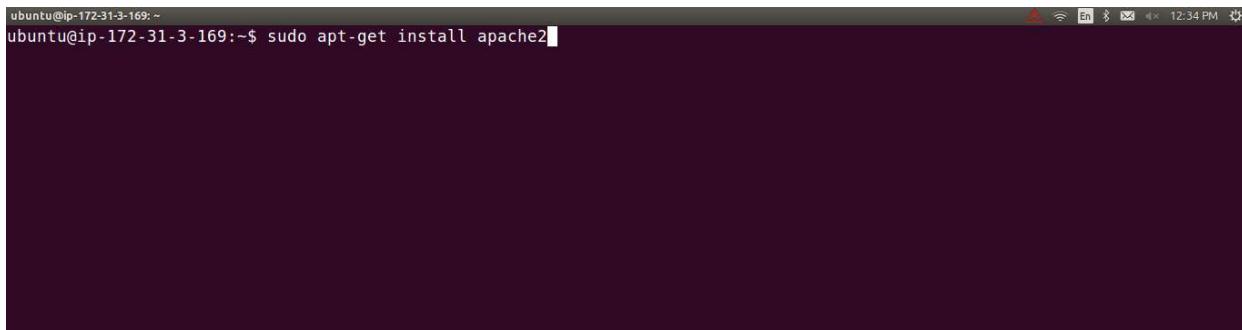
Get cloud support with Ubuntu Advantage Cloud Guest:
http://www.ubuntu.com/business/services/cloud

New release '16.04.3 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Wed Aug  9 09:43:50 2017 from 122.172.20.157
ubuntu@ip-172-31-43-82:~$ 
```

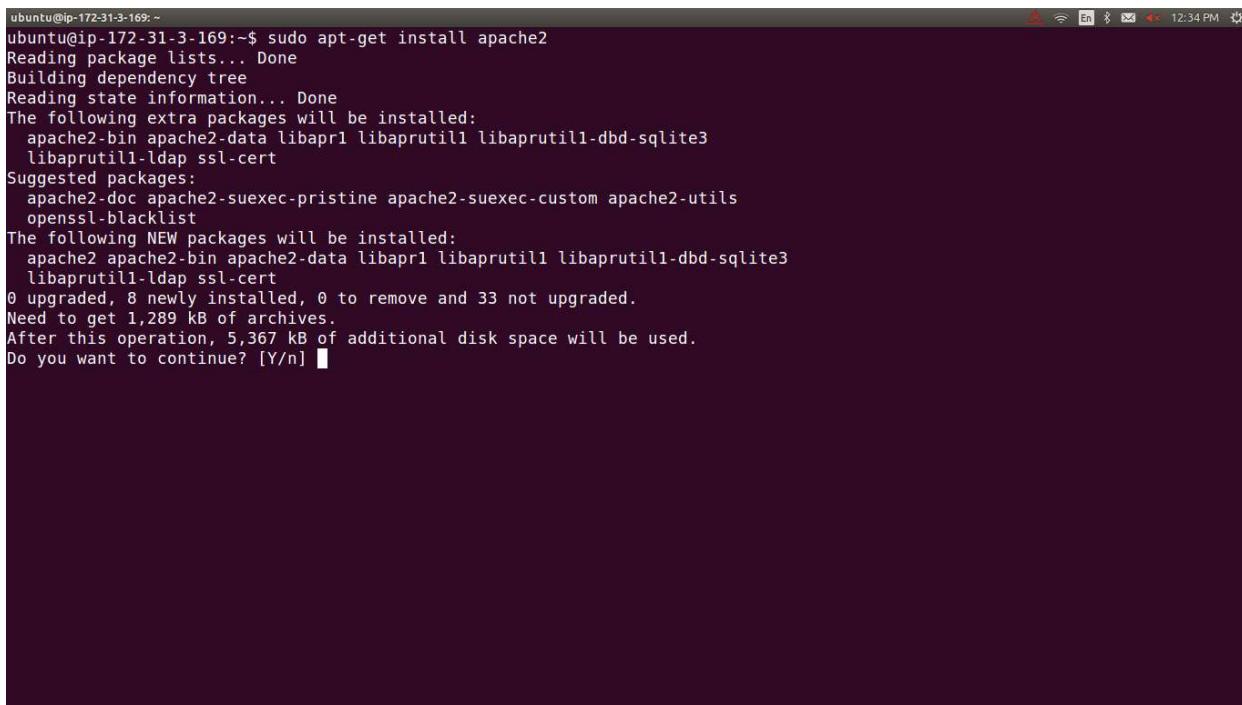
1. Execute command

```
sudo apt-get update
sudo apt-get install apache2
```



```

ubuntu@ip-172-31-3-169:~
ubuntu@ip-172-31-3-169:~$ sudo apt-get install apache2
```

A screenshot of a terminal window on an Ubuntu system. The window title is "Terminal". The command entered is "sudo apt-get install apache2". The output shows the package manager reading lists, building a dependency tree, and installing several packages. It indicates 0 upgraded, 8 newly installed, and 33 not upgraded. The total size of new files is 1,289 kB. A question "Do you want to continue? [Y/n]" is displayed at the bottom.

2. Enter [Y] for continue installation of an apache2
3. Change the permission to html directory , so outside it can be accessible.

`sudo chmod -R 777 /var/www/html`

Congratulations :) Apache service is installed on Ubuntu EC2 machine.

Configuring PHP on Ubuntu 14

1. Execute command

```
sudo apt-get update
```

```
sudo apt install -y build-essential
```

```
sudo apt install php libapache2-mod-php php-mysql libmcrypt-dev
```

A screenshot of a terminal window on an Ubuntu system. The command 'sudo apt-get install php libapache2-mod-php php-mcrypt' is being typed into the terminal. The terminal window has a dark background and light-colored text. At the top right, there are icons for battery, signal strength, and time (11:44 AM). The terminal prompt is 'ubuntu@ip-172-31-3-169:~\$'.

A screenshot of a terminal window showing the output of the 'sudo apt-get install php libapache2-mod-php php-mcrypt' command. The output includes:

Reading package lists... Done

Building dependency tree

Reading state information... Done

The following extra packages will be installed:

libmcrypt4 php5-cli php5-common php5-json php5-readline

Suggested packages:

php-pear libmcrypt-dev mcrypt php5-user-cache

The following NEW packages will be installed:

libapache2-mod-php5 libmcrypt4 php5 php5-cli php5-common php5-json

php5-mcrypt php5-readline

0 upgraded, 8 newly installed, 0 to remove and 33 not upgraded.

Need to get 4,950 kB of archives.

After this operation, 20.8 MB of additional disk space will be used.

Do you want to continue? [Y/n] ■

2. Enter [Y] for continue installation of an php
3. Next, Restart the apache2 service to recognize php is installed.

```
sudo service apache2 restart
```

Congratulations :) PHP is installed on Ubuntu EC2 machine.

Configuring MySQL on Ubuntu 14

1. Execute command

```
sudo apt-get update
sudo apt-get install mysql-server php-mysql
sudo mysql_secure_installation
```

ubuntu@ip-172-31-3-169:~\$ sudo apt-get install mysql-server php5-mysql

```
ubuntu@ip-172-31-3-169:~$ sudo apt-get install mysql-server php5-mysql
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
 libaio1 libdbd-mysql-perl libdbi-perl libhtml-template-perl libmysqlclient18
 libterm-readkey-perl mysql-client-5.5 mysql-client-core-5.5 mysql-common
 mysql-server-5.5 mysql-server-core-5.5
Suggested packages:
 libclone-perl libltdbm-perl libnet-daemon-perl libplrpc-perl
 libsql-statement-perl libipc-sharedcache-perl tinyca mailx
The following NEW packages will be installed:
 libaio1 libdbd-mysql-perl libdbi-perl libhtml-template-perl libmysqlclient18
 libterm-readkey-perl mysql-client-5.5 mysql-client-core-5.5 mysql-common
 mysql-server mysql-server-5.5 mysql-server-core-5.5 php5-mysql
0 upgraded, 13 newly installed, 0 to remove and 33 not upgraded.
0 upgraded, 13 newly installed, 0 to remove and 33 not upgraded.
Need to get 9,677 kB of archives.
After this operation, 97.5 MB of additional disk space will be used.
Do you want to continue? [Y/n] ■
```

2. Enter [Y] for continue installation of MySql

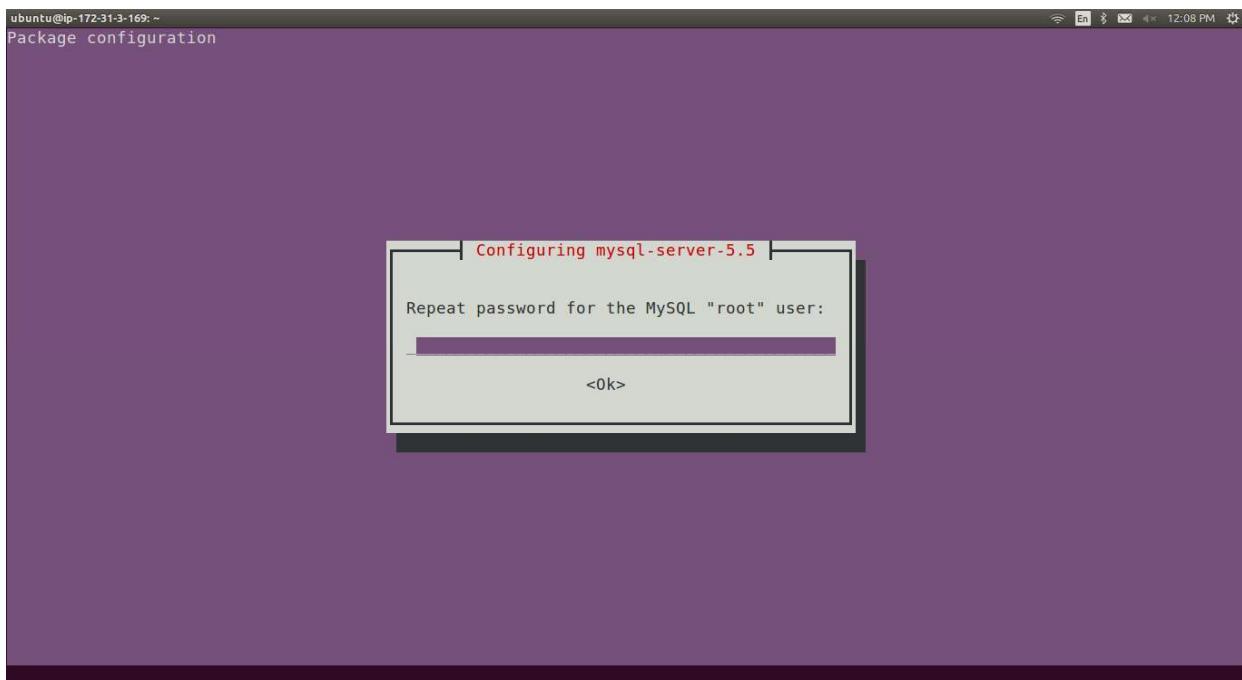
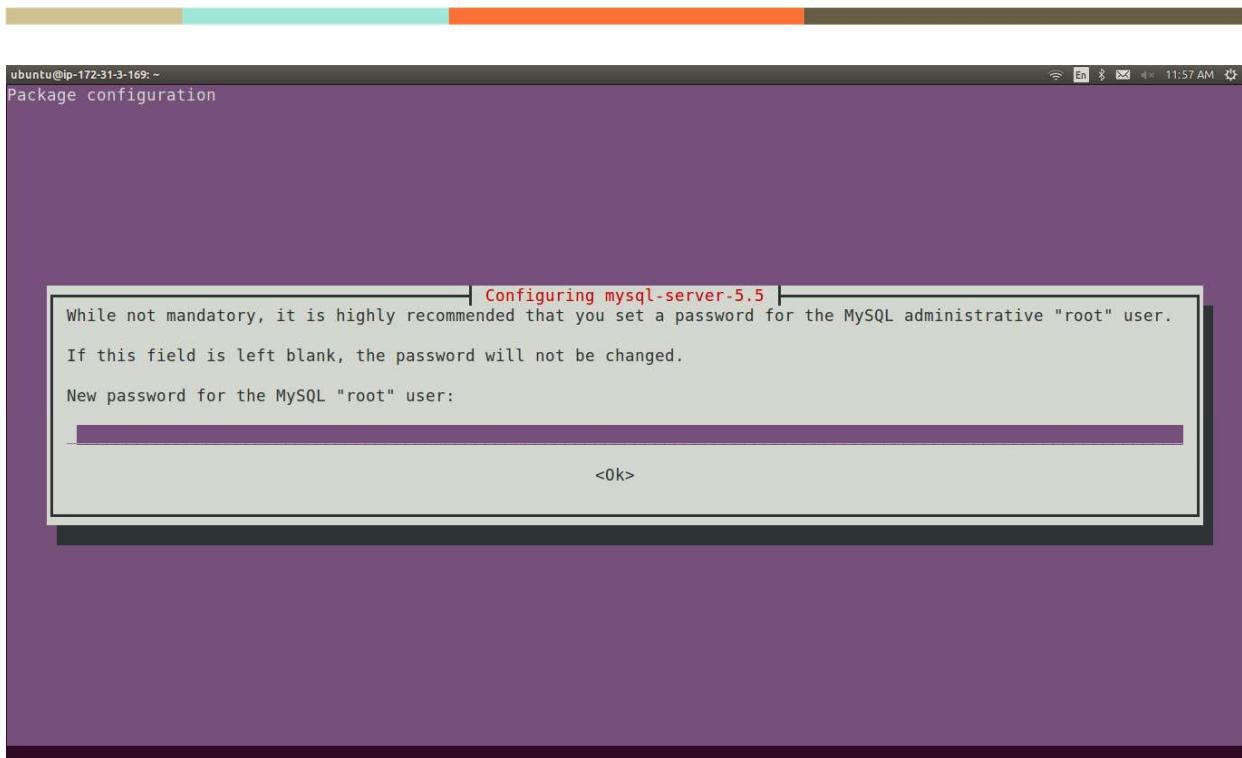
3. Next, MySQL will ask username for root user

Provide password: **root**

At the end we have user credential for MySQL database.

Username: root

Password: root



4. Next, restart the apache2 service to recognize MySQL-server is installed.

```
sudo service apache2 restart
```

Congratulations :) MySQL is installed on Ubuntu EC2 machine.

Configuring HTTP and HTTPS port on EC2 (Ubuntu)

- Open an Amazon cloud portal. <https://console.aws.amazon.com>

The screenshot shows the AWS Management Console homepage. The navigation bar at the top includes tabs for 'Inbox', 'Cloud Computing', 'Project proposal', 'AWS Management', 'Workshop Content', 'EC2 Management', and 'WinSCP - Google'. Below the navigation bar, the 'Services' and 'Resource Groups' dropdown menus are visible. The main content area is titled 'AWS services' and contains a search bar. Under 'Recently visited services', there is a link to 'EC2'. Under 'All services', the 'Compute' category is expanded, showing links to EC2, EC2 Container Service, Lightsail, Elastic Beanstalk, Lambda, and Batch. Other categories like 'Storage', 'Database', 'Cost Explorer', 'Developer Tools', 'Internet of Things', 'Contact Center', 'Management Tools', 'Game Development', 'Mobile Services', and 'Service Catalog' are also listed. To the right, a 'Helpful tips' sidebar features sections for 'Manage your costs' (with a link to start real-time billing alerts) and 'Create an organization' (with a link to use AWS Organizations for policy-based management). A 'Explore AWS' sidebar includes 'New Product Announcements' (link to view latest announcements from the AWS Summit - San Francisco), 'Migrate from Oracle to Amazon Aurora' (link to learn how to migrate with minimal downtime), and a 'Getting Started Guide'.

- Click **EC2** under Compute Category.

- Screen will be on EC2 Dashboard.

The screenshot shows the EC2 Management Console dashboard. The navigation bar at the top is identical to the previous screenshot. The main content area is titled 'EC2 Dashboard' and includes sections for 'Events', 'Tags', 'Reports', and 'Limits'. On the left, a sidebar lists 'INSTANCES' (Instances, Spot Requests, Reserved Instances, Dedicated Hosts), 'IMAGES' (AMIs, Bundle Tasks), 'ELASTIC BLOCK STORE' (Volumes, Snapshots), and 'NETWORK & SECURITY' (Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces). The central part of the dashboard displays 'Resources' used in the US East (Ohio) region, including 1 Running Instances, 2 Elastic IPs, 0 Dedicated Hosts, 2 Volumes, 1 Key Pairs, 0 Placement Groups, 0 Snapshots, 0 Load Balancers, and 3 Security Groups. A callout box suggests trying Amazon Lightsail for free. Below this, a 'Create Instance' section allows launching a new instance, noting that instances will launch in the US East (Ohio) region. It includes a 'Launch Instance' button and a note about the region. To the right, 'Account Attributes' are listed: Supported Platforms (VPC), Default VPC (vpc-bd2d6dd4), and Resource ID length management. A 'Additional Information' sidebar provides links to the Getting Started Guide, Documentation, All EC2 Resources, Forums, Pricing, and Contact Us. A 'AWS Marketplace' sidebar offers free software trial products and popular AMIs like Barracuda NextGen Firewall F-Series - PAYG.

4. Click on Running Instances.

The screenshot shows the AWS EC2 Instances page. On the left sidebar, under the 'Instances' section, 'Instances' is selected. In the main content area, there is a table with one row. The row details a single instance: Name - i-0fcfc4fc3afe2f71e, Instance ID - i-0fcfc4fc3afe2f71e, Instance Type - t2.micro, Availability Zone - us-east-2c, Instance State - running, Status Checks - 2/2 checks ... (green), Alarm Status - None, and Public DNS - ec2-52-14-202-117.us-east-2.compute.amazonaws.com. Below the table, a detailed view for the same instance is shown with tabs for Description, Status Checks, Monitoring, and Tags. The 'Description' tab is selected, displaying the following information:

Instance ID	i-0fcfc4fc3afe2f71e	Public DNS (IPv4)	ec2-52-14-202-117.us-east-2.compute.amazonaws.com
Instance state	running	IPv4 Public IP	52.14.202.117
Instance type	t2.micro	IPv6 IPs	-
Elastic IPs		Private DNS	ip-172-31-43-82.us-east-2.compute.internal
Availability zone	us-east-2c	Private IPs	172.31.43.82

5. Click on Security group for example here is **launch-wizard-1** name for security group.

The screenshot shows the AWS EC2 Instance details page for the instance i-0fcfc4fc3afe2f71e. The 'Description' tab is selected. The 'Security groups' field contains the value 'launch-wizard-1'. A link 'view inbound rules' is shown next to it. The rest of the fields include Instance ID (i-0fcfc4fc3afe2f71e), Instance state (running), Instance type (t2.micro), Availability zone (us-east-2c), Scheduled events (No scheduled events), and AMI ID (ubuntu/images/hvm-ssd/ubuntu-trusty-14.04-amd64-server-20170718 (ami-43391926)).

6. It show the security group information for the selected group name.

Secure | https://us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#SecurityGroups:groupId=sg-906393f8:sort=groupId

Services ▾ Resource Groups ▾

EC2 Dashboard Events Tags Reports Limits

INSTANCES Instances Spot Requests Reserved Instances Dedicated Hosts

IMAGES AMIs Bundle Tasks

ELASTIC BLOCK STORE Volumes Snapshots

NETWORK & SECURITY

Security Groups Elastic IPs Placement Groups Key Pairs

Description Inbound Outbound Tags

Group name: launch-wizard-1
Group ID: sg-906393f8
Group description: launch-wizard-1 created 2017-08-09T15:02:54.361+05:30
VPC ID: vpc-09490460

Feedback English

© 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

7. Click on Inbound

Services ▾ Resource Groups ▾

Create Security Group Actions

Group ID: sg-906393f8 Add filter

Name	Group ID	Group Name	VPC ID	Description
sg-906393f8	launch-wizard-1	vpc-09490460	launch-wizard-1 created 2017-08-09T15:0...	

Security Group: sg-906393f8

Description Inbound Outbound Tags

Group name: launch-wizard-1
Group ID: sg-906393f8
Group description: launch-wizard-1 created 2017-08-09T15:02:54.361+05:30
VPC ID: vpc-09490460

Edit

Type	Protocol	Port Range	Source
SSH	TCP	22	0.0.0.0/0

Feedback English

© 2008 - 2017, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

8. Click on Edit

Edit inbound rules

Type	Protocol	Port Range	Source
SSH	TCP	22	Custom 0.0.0.0/0

Add Rule

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

Cancel Save

9. Click on Add Rule

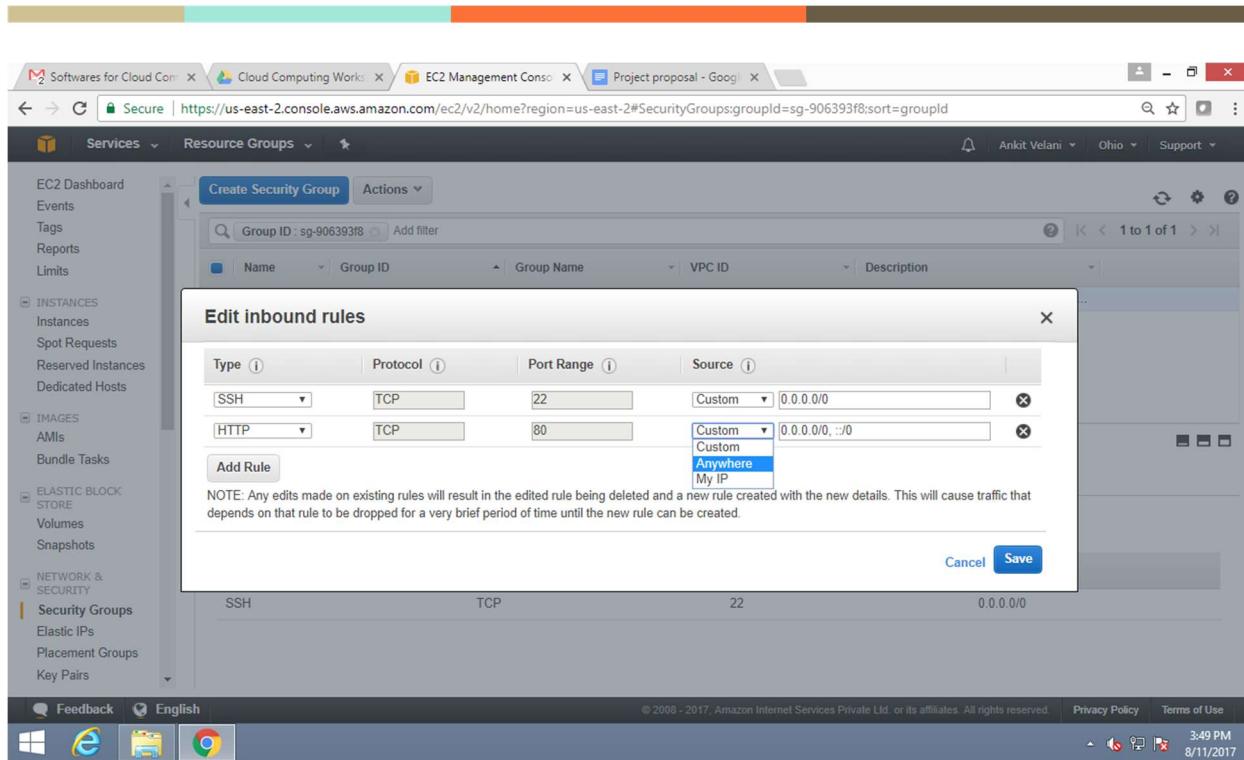
Edit inbound rules

Type	Protocol	Port Range	Source
SSH	TCP	22	0.0.0.0/0
HTTP	TCP	22	Custom 0.0.0.0/0
	TCP	0	CIDR, IP or Security Group

Add Rule

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

Cancel Save



10. Select Rule type as **HTTP** and Source as **Anywhere**.

11. Click on **Save**.

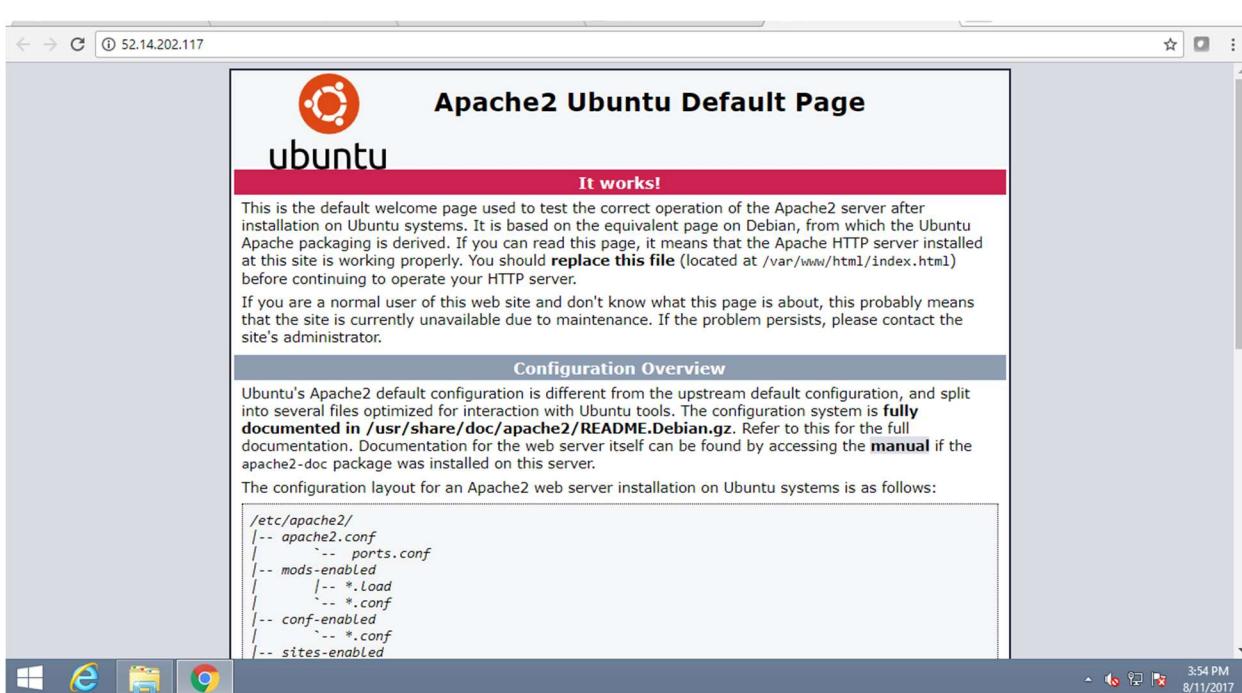
12. For adding more rules follow the same steps.

13. Copy EC-2 public IP and access from the web-browser.

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links like EC2 Dashboard, Events, Tags, Reports, Limits, Instances, Spot Requests, Reserved Instances, Dedicated Hosts, Images, AMIs, Bundle Tasks, Elastic Block Store, Volumes, Snapshots, and Network & Security. The main content area has tabs for Launch Instance, Connect, Actions, and a search bar. A table lists one instance: i-0fcfc4fc3afe2f71e, t2.micro, us-east-2c, running, with 2/2 checks passing, alarm status none, and public DNS ec2-52-14-202-117.us-east-2.compute.amazonaws.com. Below the table, a detailed view shows fields like Instance ID, Public DNS (IPv4), Instance state, IPv4 Public IP, Instance type, IPv6 IPs, Availability zone, Private DNS, Elastic IPs, Security groups, Secondary private IPs, Scheduled events, VPC ID, AMI ID, and Subnet ID.

	Value	Value	Value	Value	Value	Value	Value	Value	Value	Value	Value	Value	Value	Value	Value
Public DNS (IPv4)	ec2-52-14-202-117.us-east-2.compute.amazonaws.com														
IPv4 Public IP	52.14.202.117														
IPv6 IPs	-														
Private DNS	ip-172-31-43-82.us-east-2.compute.internal														
Private IPs	172.31.43.82														
Secondary private IPs															
VPC ID	vpc-09490460														
Subnet ID	subnet-2304a56e														

Public DNS (IPv4) ec2-52-14-202-117.us-east-2.compute.amazonaws.com
 IPv4 Public IP 52.14.202.117
 IPv6 IPs -
 Private DNS ip-172-31-43-82.us-east-2.compute.internal
 Private IPs 172.31.43.82
 Secondary private IPs
 VPC ID vpc-09490460
 Subnet ID subnet-2304a56e



14. If Web-browser is displaying Apache2 Ubuntu Default Page then we have successfully configures HTTP port on EC-2.

Congratulation :) Now EC-2 Machine can be accessible using HTTP port.

Configuring HTTP and HTTPS port on EC2 (Windows)

- Open an Amazon cloud portal. <https://console.aws.amazon.com>

The screenshot shows the AWS Management Console homepage. The navigation bar at the top includes tabs for 'Inbox', 'Cloud Computing', 'Project proposal', 'AWS Management', 'Workshop Cont...', 'EC2 Management', 'WinSCP - Google...', and 'Ankit'. Below the navigation bar, the 'Services' and 'Resource Groups' dropdown menus are visible. The main content area is titled 'AWS services' and contains a search bar. Under 'Recently visited services', there are links for EC2, Cost Explorer, and Billing. Under 'All services', the Compute category is expanded, showing EC2, EC2 Container Service, Lightsail, Elastic Beanstalk, Lambda, and Batch. Other categories like Storage, Database, Developer Tools, Internet of Things, Contact Center, Game Development, and Mobile Services are also listed. On the right side, there are 'Helpful tips' for 'Manage your costs' and 'Create an organization', and an 'Explore AWS' section with links for 'New Product Announcements' and 'Migrate from Oracle to Amazon Aurora'.

- Click **EC2** under Compute Category.

- Click on Launch Instance.

The screenshot shows the AWS EC2 Dashboard. The left sidebar includes sections for New EC2 Experience, EC2 Dashboard (Events, Tags, Limits, Instances, Images, Elastic Block Store, Network & Security), and other resources like Security Groups, Elastic IPs, and Placement Groups. The main content area features a 'Welcome to the new EC2 console' message. Below it, the 'Resources' section displays a table of EC2 resources: Instances (0), Elastic IPs (0), Dedicated Hosts (0), Snapshots (0), Volumes (0), Key pairs (1), Security groups (10), Placement groups (0), Load balancers (0), and Running instances (0). A note about launching Microsoft SQL Server Always On availability groups is shown. To the right, the 'Account attributes' section lists supported platforms (VPC, Default VPC vpc-f22c8e99), settings (EBS encryption, Zones, Default credit specification, Console experiments), and an 'Explore AWS' section with links for Apache Spark, EC2 Spot Instances, and Custom AMIs.

- Choose AMI (Amazon Machine Image) which is **free tier** and **Windows** type.

Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Quick Start (8)

- My AMIs (0)
- AWS Marketplace (734)
- Community AMIs (4331)
- Free tier only ⓘ

Image	Name	Description	Root device type	Virtualization type	ENI Enabled	Select
Windows	Microsoft Windows Server 2019 Base - ami-0354df7841220296c	Microsoft Windows 2019 Datacenter edition. [English]	ebs	hvm	Yes	64-bit (x86)
Windows	Microsoft Windows Server 2019 Base with Containers - ami-0df1c0a71406084c4	Microsoft Windows 2019 Datacenter edition with Containers. [English]	ebs	hvm	Yes	64-bit (x86)
Windows	Microsoft Windows Server 2004 Core Base - ami-007729654a972fea2	Microsoft Windows Server 2004 Semi-Annual Channel release. [English]	ebs	hvm	Yes	64-bit (x86)
Windows	Microsoft Windows Server 2016 Base - ami-0952fb5203ddacf5c	Microsoft Windows 2016 Datacenter edition. [English]	ebs	hvm	Yes	64-bit (x86)
Windows	Microsoft Windows Server 2016 Base with Containers - ami-0965403ee0aeaf79cf	Microsoft Windows 2016 Datacenter edition with Containers. [English]	ebs	hvm	Yes	64-bit (x86)

5. Select Microsoft Windows Server 2019 Base with Containers AMI type.

Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.

Filter by: All instance families ▾ Current generation ▾ Show/Hide Columns

Currently selected: t2.micro (- ECUs, 1 vCPUs, 2.5 GHz, ~ 1 GiB memory, EBS only)

Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
t2	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
t2	t2.micro <small>Free tier eligible</small>	1	1	EBS only	-	Low to Moderate	Yes
t2	t2.small	1	2	EBS only	-	Low to Moderate	Yes
t2	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
t2	t2.large	2	8	EBS only	-	Low to Moderate	Yes
t2	t2.xlarge	4	16	EBS only	-	Moderate	Yes
t2	t2.2xlarge	8	32	EBS only	-	Moderate	Yes
t3	t3.nano	2	0.5	EBS only	Yes	Up to 5 Gigabit	Yes
t3	t3.micro	2	1	EBS only	Yes	Up to 5 Gigabit	Yes
t3	t3.small	2	2	EBS only	Yes	Up to 5 Gigabit	Yes
t3	t3.medium	2	4	EBS only	Yes	Up to 5 Gigabit	Yes
t3	t3.large	2	8	EBS only	Yes	Up to 5 Gigabit	Yes

Cancel Previous Review and Launch Next: Configure Instance Details

6. Select t2.micro type instance, which is an only eligible for free tier type.

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

AMI Details

Microsoft Windows Server 2019 Base with Containers - ami-0df1c0a71408084c4

Free tier eligible Microsoft Windows 2019 Datacenter edition with Containers. [English]

Root Device Type: ebs Virtualization type: hvm

Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	-	1	1	EBS only	-	Low to Moderate

Security Groups

Security group name: launch-wizard-10
Description: launch-wizard-10 created 2020-11-04T09:31:33.737+05:30

Type	Protocol	Port Range	Source	Description
This security group has no rules				

Instance Details

Storage

Tags

Cancel Previous Launch

7. Click on Launch

8. Create or acknowledge the key file. If you wish to existing key pair file then select Choose an existing key pair option from first drop down and select the name of existing key pair file from second dropdown and then check acknowledge.
9. After selecting key pair, click on Launch instances.

10. EC-2 windows instance is configured successfully. Click on View instances to view.

The screenshot shows the AWS EC2 Management Console interface. On the left, there's a navigation sidebar with various services like EC2 Dashboard, Events, Tags, Instances, Images, Elastic Block Store, Network & Security, and more. The main content area is titled 'Instances (1) Info' and shows a table with one row. The row details are:

Name	Instance ID	Instance state	Instance type	Status check	Alarm Status	Availability zone	Public IPv4 DNS	Public IPv4 ...
-	i-077cd14fc46ca3e0e	Running	t2.micro	Initializing	No alarms	us-east-2c	ec2-18-188-76-161.us-east-2.co...	18.188.76.161

Below the table, it says 'Select an instance above'. At the bottom right of the main area, there are links for 'Feedback', 'English (US)', 'Privacy Policy', and 'Terms of Use'.

11. We can see t2.micro instance is running. Select instance to view the IP and DNS to access instance.

This screenshot is similar to the previous one but with a specific instance selected. The table now has a checked checkbox next to the first column. The instance details page below shows the following information for the selected instance (i-077cd14fc46ca3e0e):

Details

- Instance summary**

 - Instance ID: i-077cd14fc46ca3e0e
 - Public IPv4 address: 18.188.76.161
 - Private IPv4 addresses: 172.31.41.89
 - Public IPv4 DNS: ec2-18-188-76-161.us-east-2.compute.amazonaws.com
 - Private IPv4 DNS: ip-172-31-41-89.us-east-2.compute.internal
 - VPC ID: vpc-f22c8698

- Instance details**

 - Platform: windows
 - AMI ID: ami-0f6410a7140c0944
 - Monitoring: disabled

12. Access the Windows instance using Remote Desktop Connection. Remote Desktop Connection software required Public DNS/IP, username and password to access any instance.

13. To get windows instance password, select instance and click on Actions > Connect.

The screenshot shows the AWS EC2 Management Console. On the left, there's a sidebar with various services like New EC2 Experience, EC2 Dashboard, Events, Tags, Limits, Instances, Images, Elastic Block Store, Network & Security, and more. The main area shows a table of instances with one row selected. The selected instance is 'i-077cd14fc46ca3e0e', which is running and has an instance type of t2.micro. The 'Actions' dropdown menu is open, and the 'Connect' option is highlighted with a yellow box. Below the table, there's a detailed view of the selected instance, including its summary, details, and instance details.

This screenshot shows the 'Connect to instance' dialog box. It has tabs for 'Session Manager' and 'RDP client', with 'RDP client' being the active tab. It provides instructions to connect using a remote desktop client or download a shortcut file. It lists connection details: Public DNS (ec2-18-188-76-161.us-east-2.compute.amazonaws.com), User name (Administrator), and Password (Get password). A note at the bottom says, 'If you've joined your instance to a directory, you can use your directory credentials to connect to your instance.' At the bottom right, there's a 'Cancel' button.

14. Click on RDP Client, and click on Get password.

The screenshot shows the AWS EC2 interface for getting a Windows password. The URL is [EC2 > Instances > i-077cd14fc46ca3e0e > Get windows password](#). A modal window titled "Get Windows password" is open, showing a "Key pair associated with this instance" section with a file named "SIT2020.pem" selected. Below it, there's a text area for copying the contents of the key pair. At the bottom of the modal are "Cancel" and "Decrypt Password" buttons, with "Decrypt Password" being the active one.

15. Select .pem file which is key pair file for instance, and click on **Decrypt Password**.

The screenshot shows the AWS EC2 interface for connecting to an instance. The URL is [EC2 > Instances > i-077cd14fc46ca3e0e > Connect to instance](#). A modal window titled "Connect to instance" is open, showing the "RDP client" tab selected. At the bottom of the modal are "Cancel" and "Decrypt Password" buttons, with "Decrypt Password" being the active one.

16. Decrypt password will enable windows passwords, using this credential we can able to access using Remote Desktop Connection software.

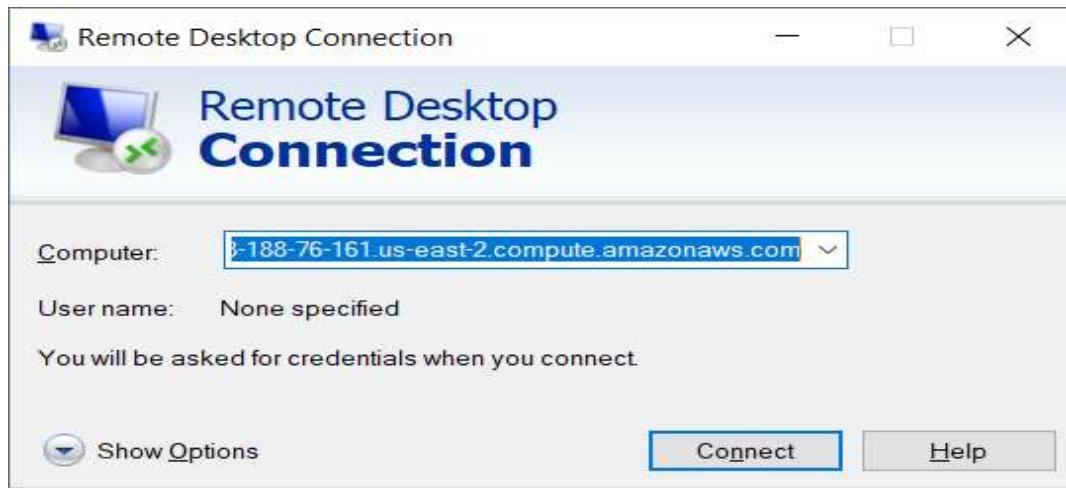
When prompted, connect to your instance using the following details:

Public DNS
 ec2-18-188-76-161.us-east-2.compute.amazonaws.com

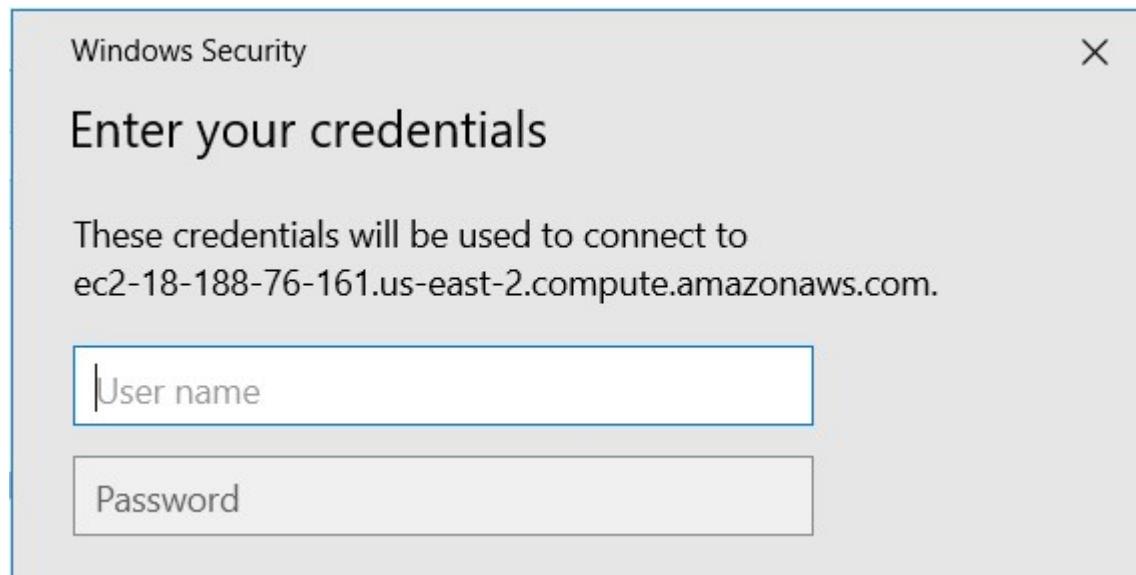
User name
 Administrator

Password
 3PVA?xu7ftpjvL8im*PLnK=Fmn2yX*3a

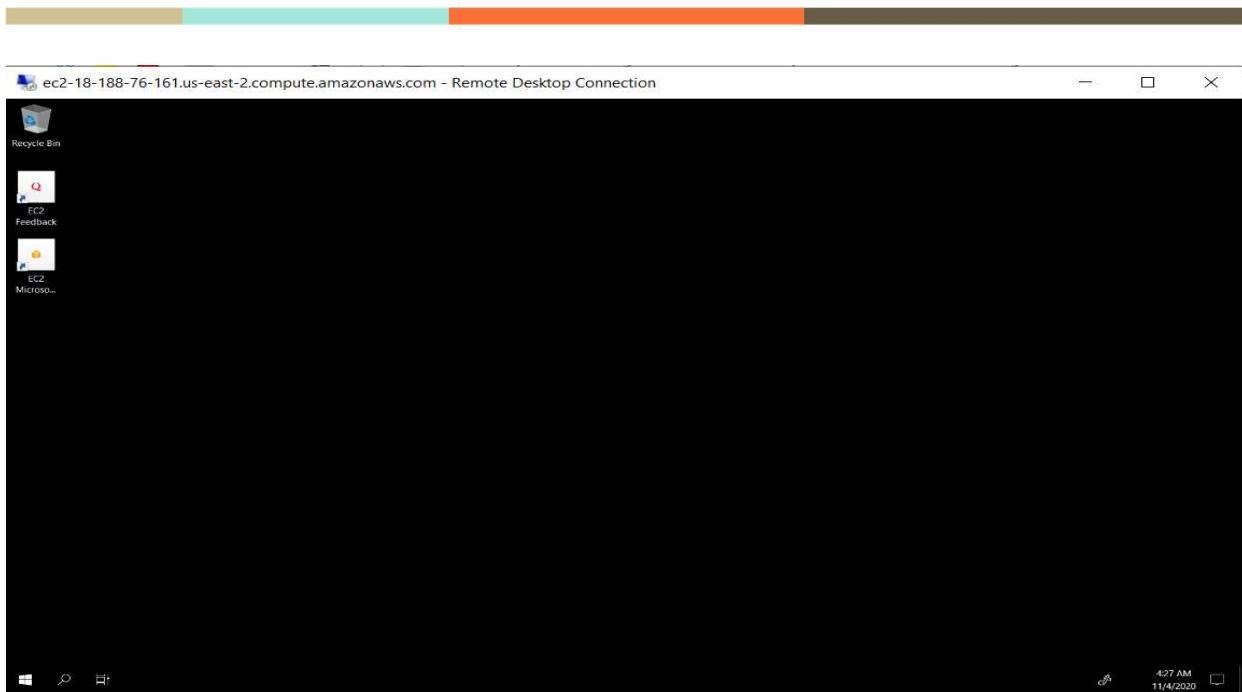
17. Open Remote Desktop Connection software and copy paste credential.



18. Copy public DNS and click on Connect

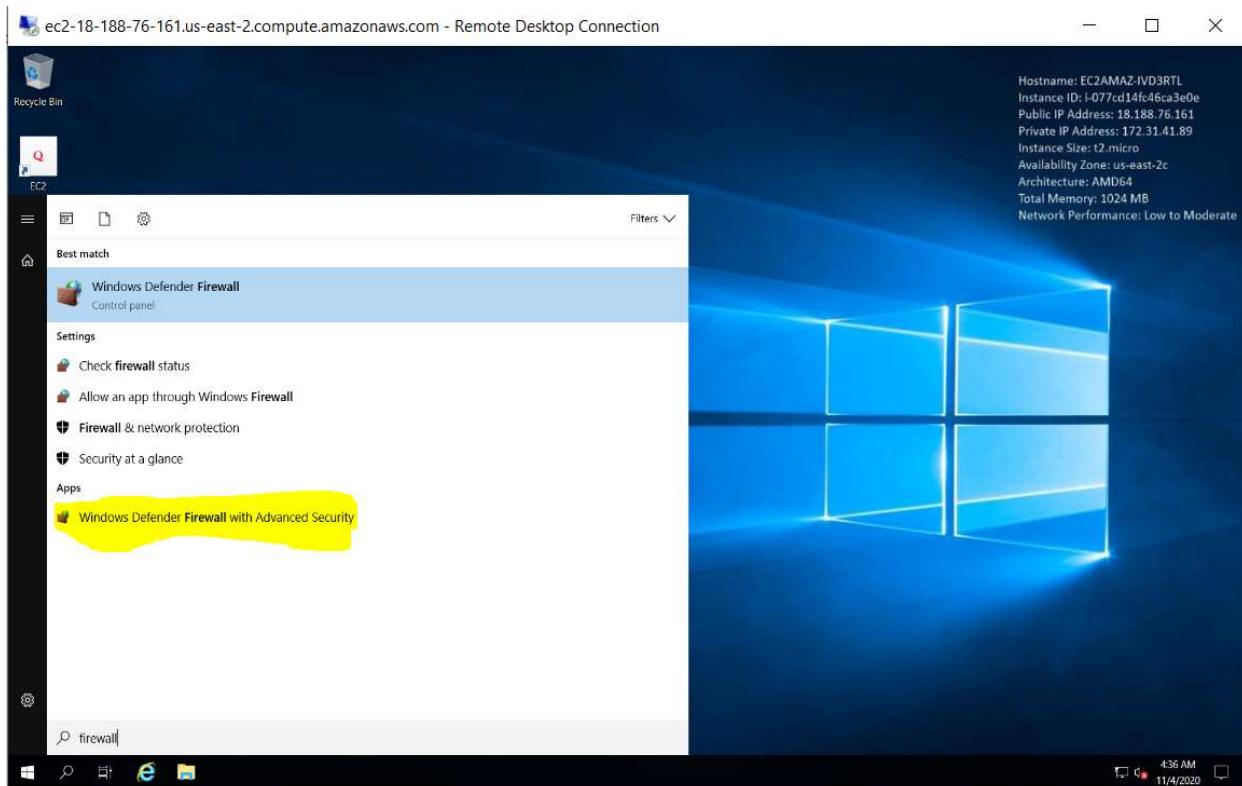


19. Enter username and password, click on connect to login into Windows EC2.



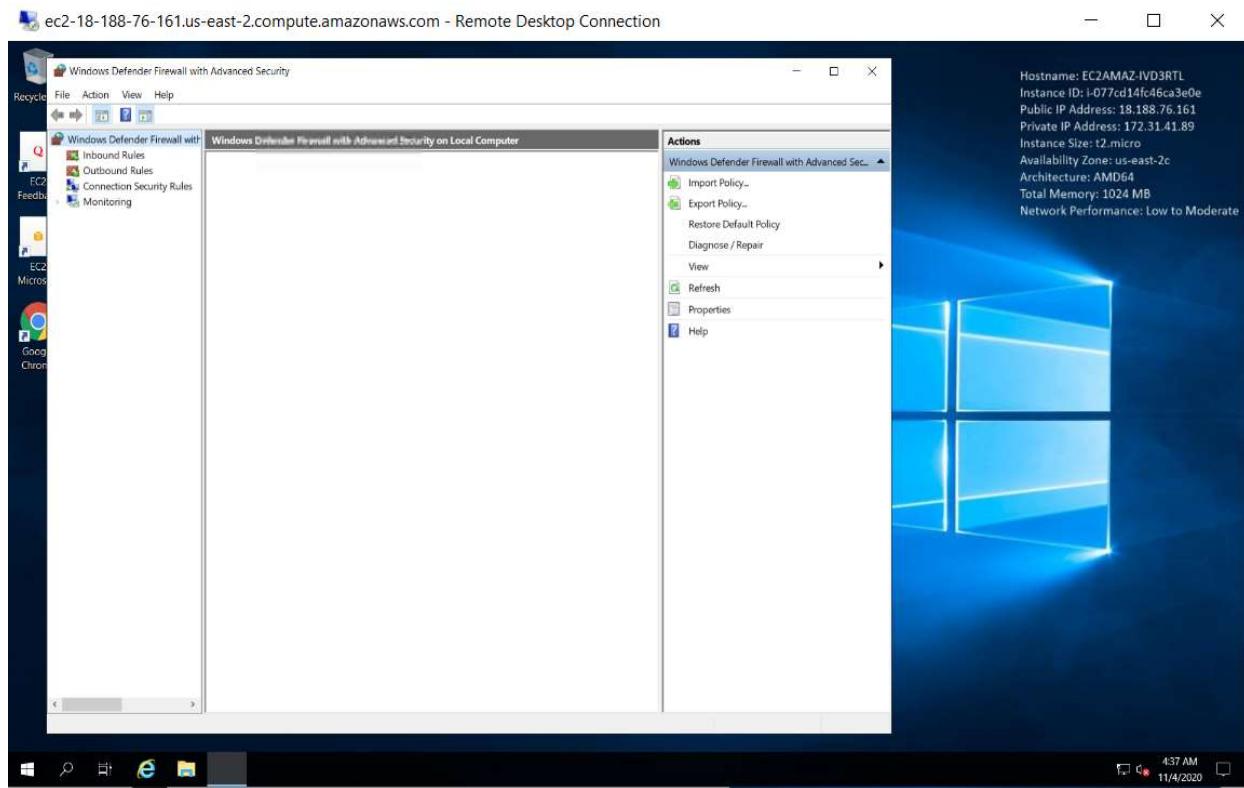
Successfully connected Windows EC2 using Remote Desktop Connection software.

20. Configure Port 80 on windows instance, open Windows Defender Firewall with advance Security features.

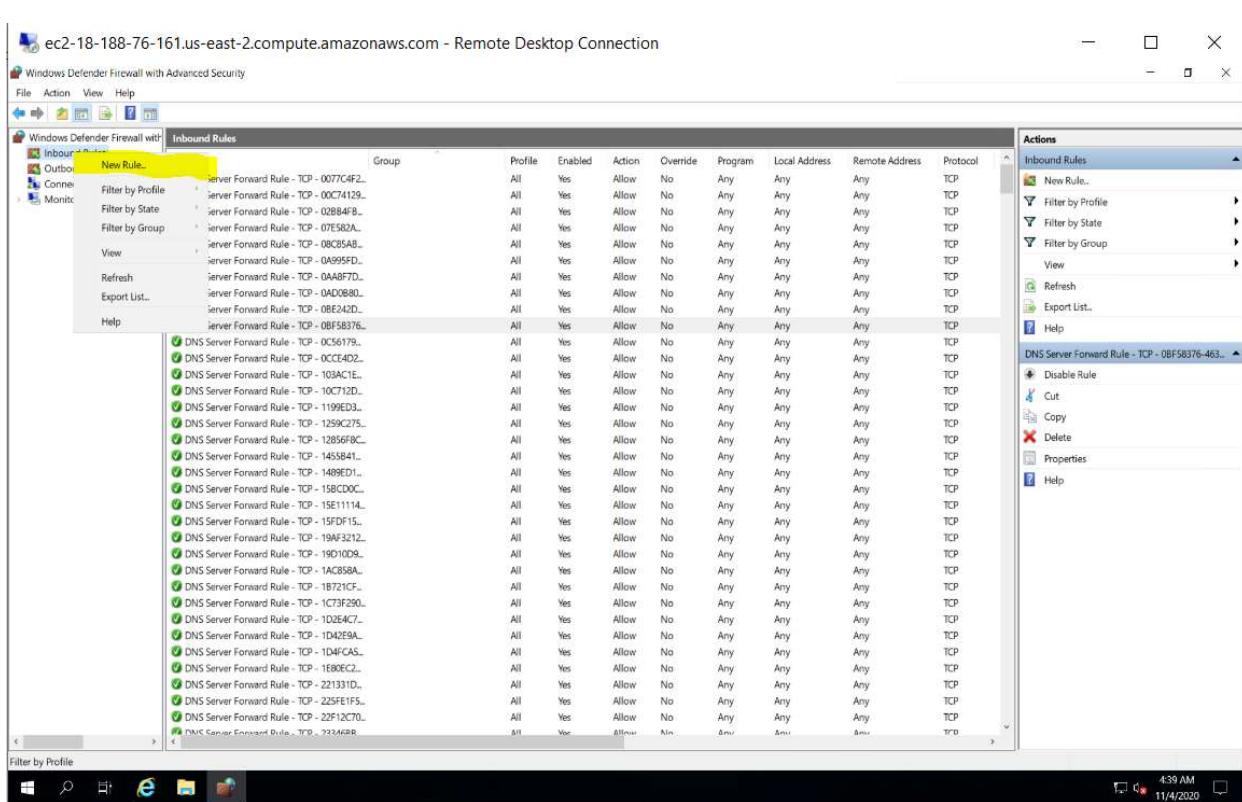




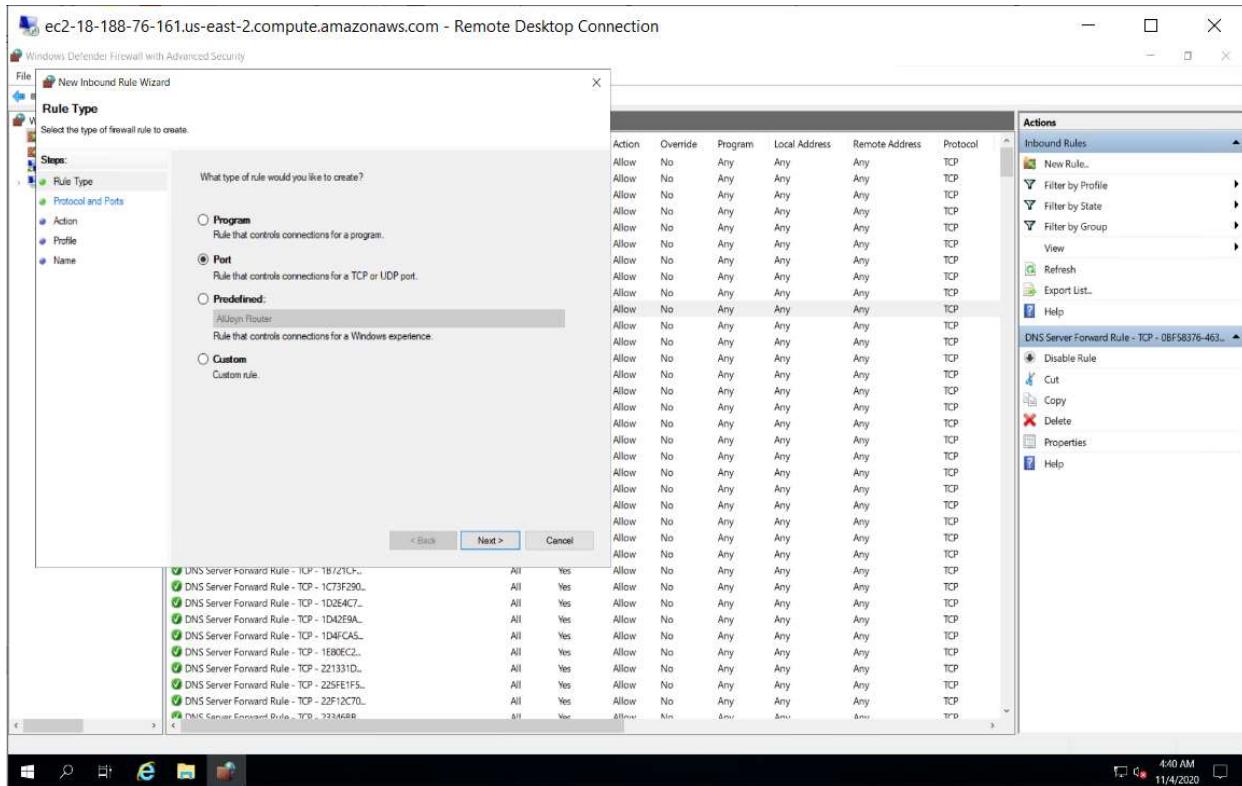
21. Select Inbound Rules tab from screen



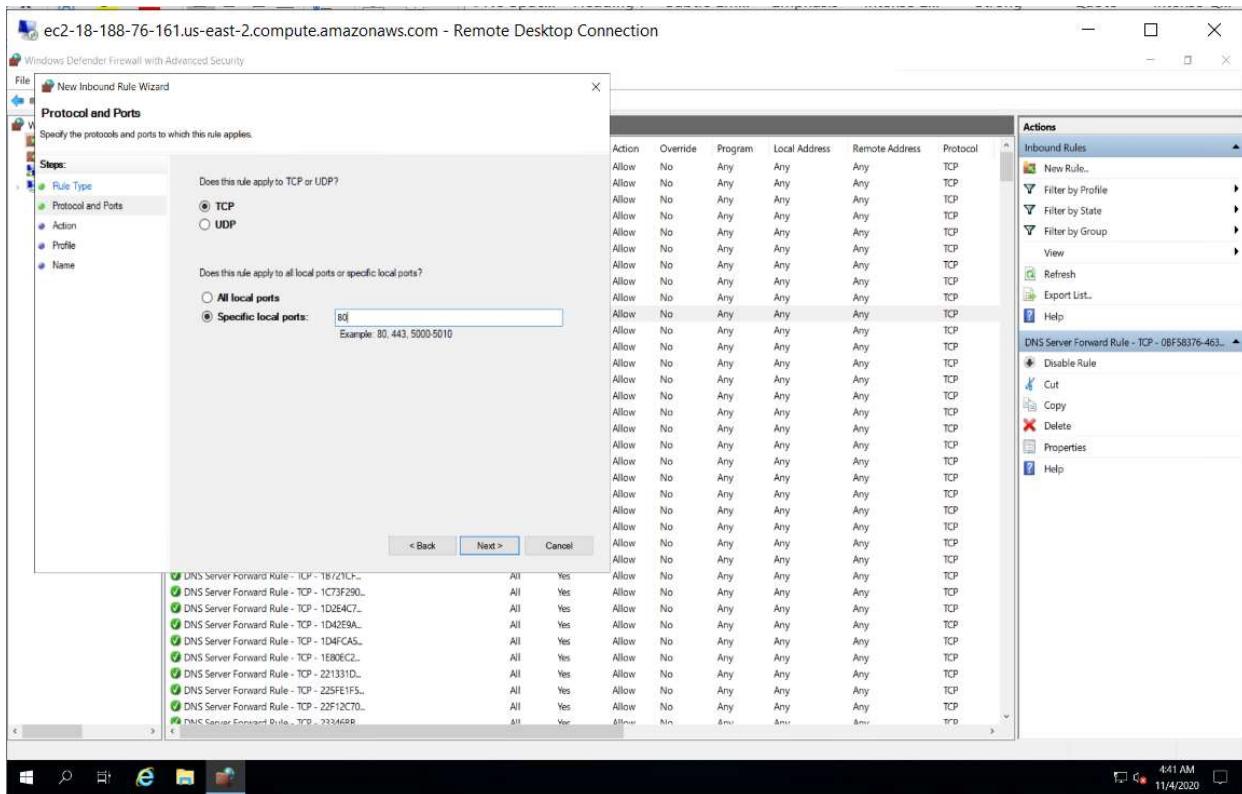
22. Right click on Inbound Rules > New Rule...



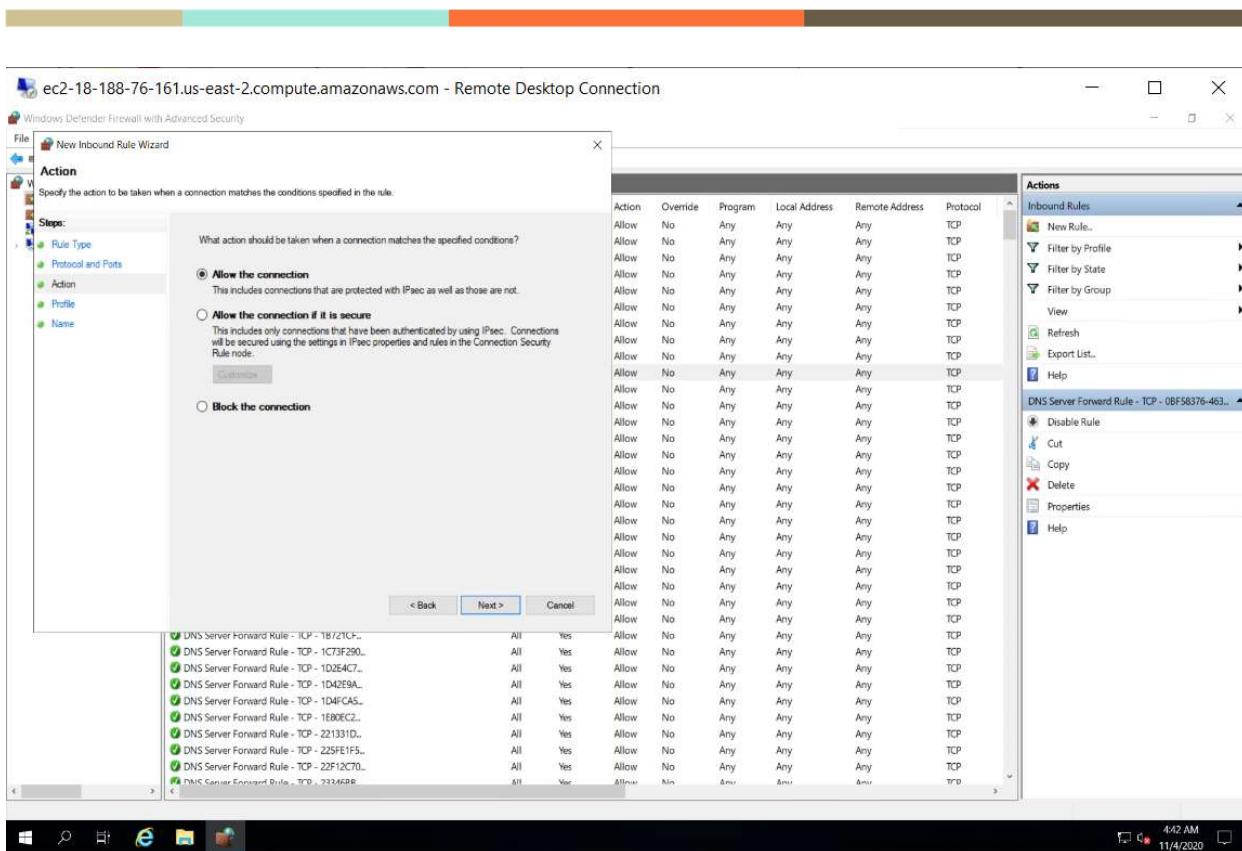
23. Select Port and click on Next



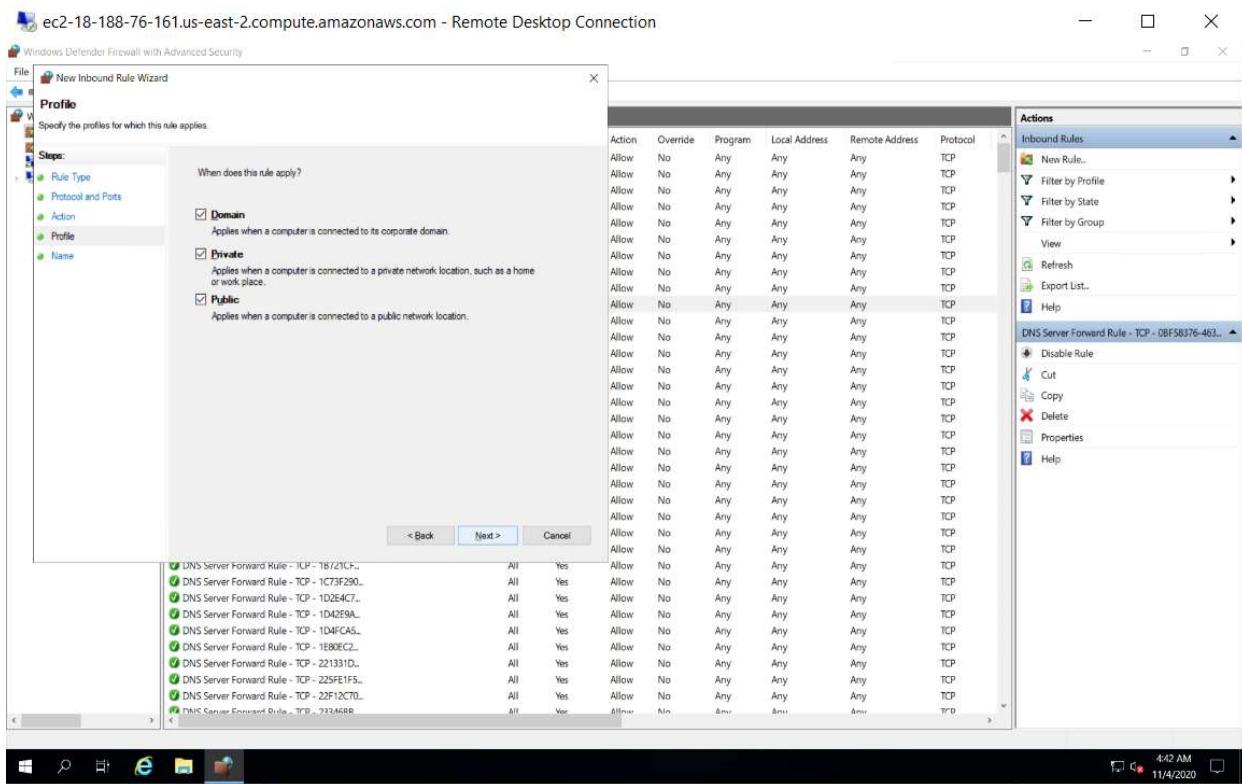
24. Type port number 80 with TCP selection and click on Next



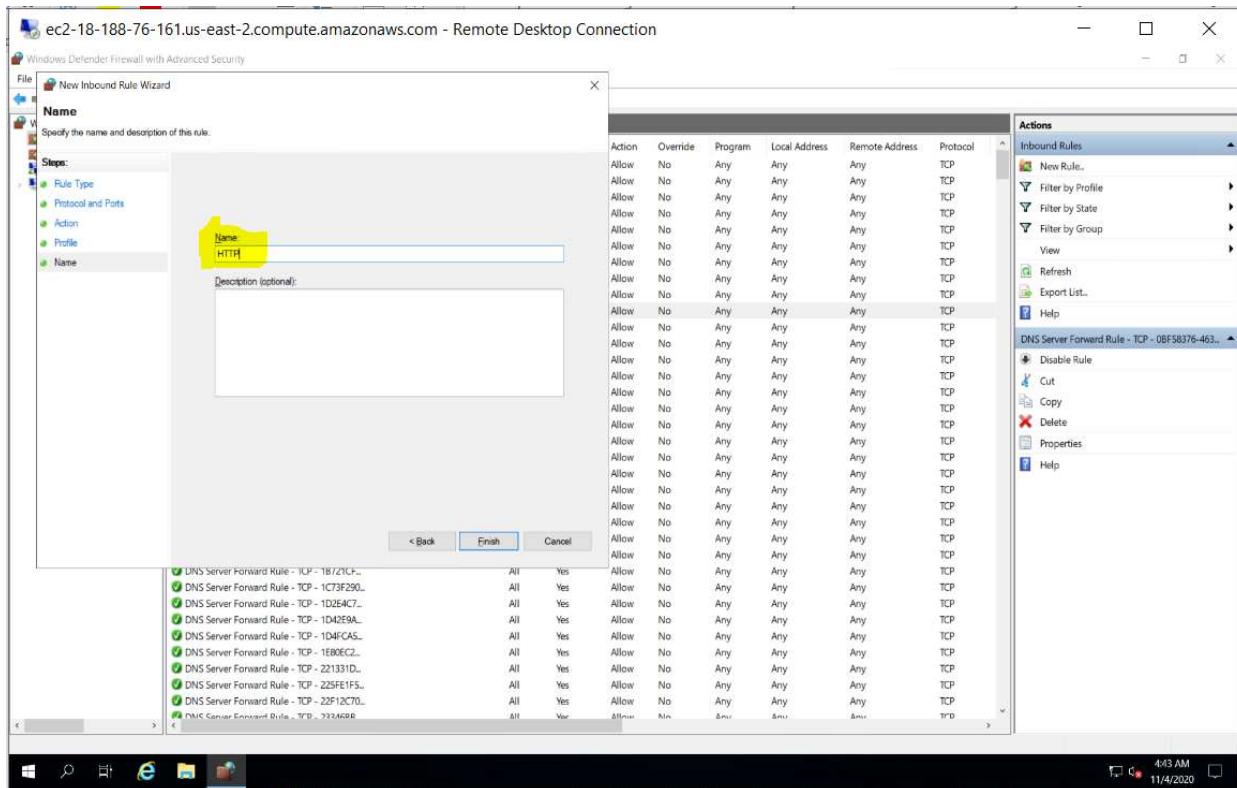
25. Click on Next



26. Click on Next



27. Enter Rules name HTTP , click on Finish.



28. Port 80 successfully configured on Windows Firewall configuration.

29. Next Step to configure port 80 on AWS console, select instance and select Security tab.

The screenshot shows the AWS EC2 Management Console. On the left, there's a navigation sidebar with various services like EC2 Dashboard, Events, Tags, Instances, Images, Elastic Block Store, Network & Security, and more. The main area displays a table of instances. One instance, with the ID i-077cd14fc46ca3e0e, is selected. The 'Security' tab is currently selected under the instance details. It shows the IAM Role (empty), Owner ID (765525232641), and Launch time (Wed Nov 04 2020 09:35:36 GMT+0530). Under 'Security groups', it lists sg-0fc1f01576f4f542a (launch-wizard-10). The 'Inbound rules' section shows a single rule: Port range 3389, Protocol TCP, Source 0.0.0.0/0, and Security groups launch-wizard-10.

30. Select associate security group

This screenshot is identical to the one above, showing the AWS EC2 Management Console. The instance i-077cd14fc46ca3e0e is selected. The 'Security' tab is active. The security group sg-0fc1f01576f4f542a (highlighted in yellow) is selected under the 'Security groups' section. The rest of the interface, including the inbound rules table, remains the same.

31. Click on Edit inbound rules.

Instances | EC2 Management Con x speed test google - Google Search x +

us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#SecurityGroups:securityGroupId=sg-Ofc1f01576f4f542a

EC2 Services ▾

New EC2 Experience Tell us what you think

EC2 Dashboard New

Events New

Tags

Limits

Instances Instances New

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts New

Capacity Reservations

Images AMIs

Elastic Block Store Volumes Snapshots Lifecycle Manager

Network & Security Security Groups New

Elastic IPs New

Placement Groups New

Feedback English (US) ▾

© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

32. Click on Add rule

Instances | EC2 Management Con x speed test google - Google Search x +

us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#ModifyInboundSecurityGroupRules:securityGroupId=sg-Ofc1f01576f4f542a

EC2 Services ▾

Security Groups sg-Ofc1f01576f4f542a - launch-wizard-10 Edit inbound rules

Inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

Type	Protocol	Port range	Source	Description - optional
RDP	TCP	3389	0.0.0.0/0	-

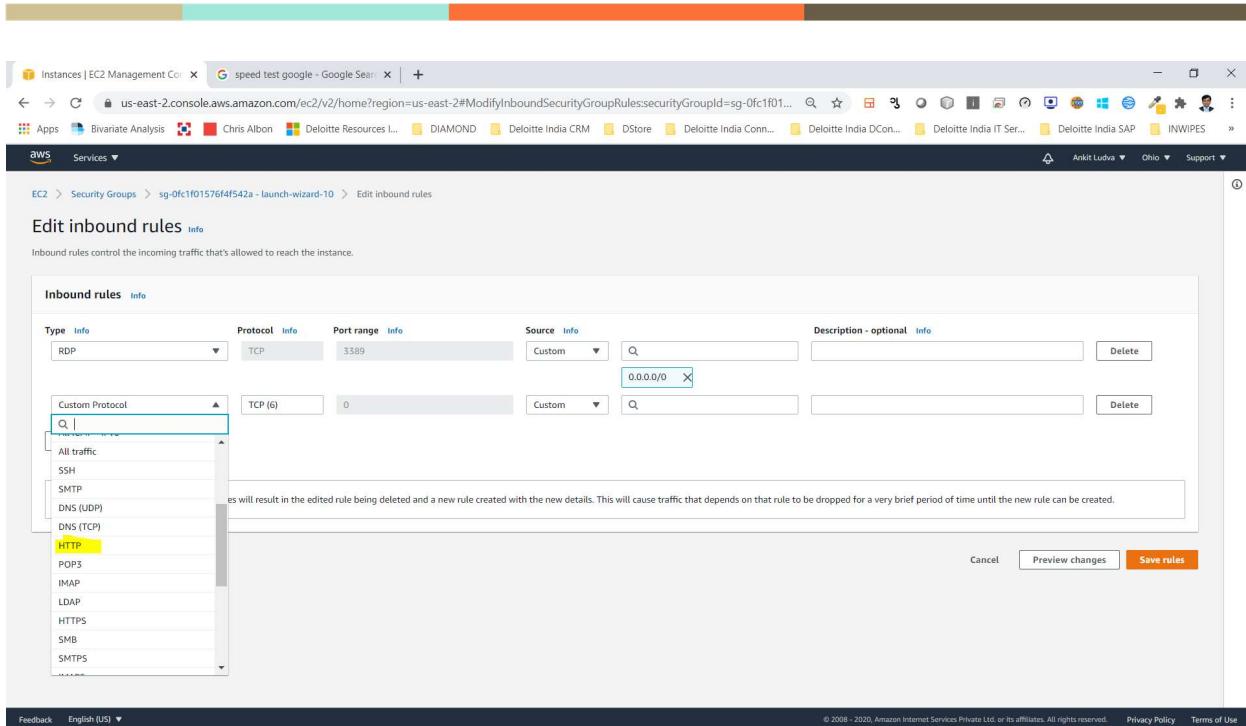
Add rule

⚠ NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

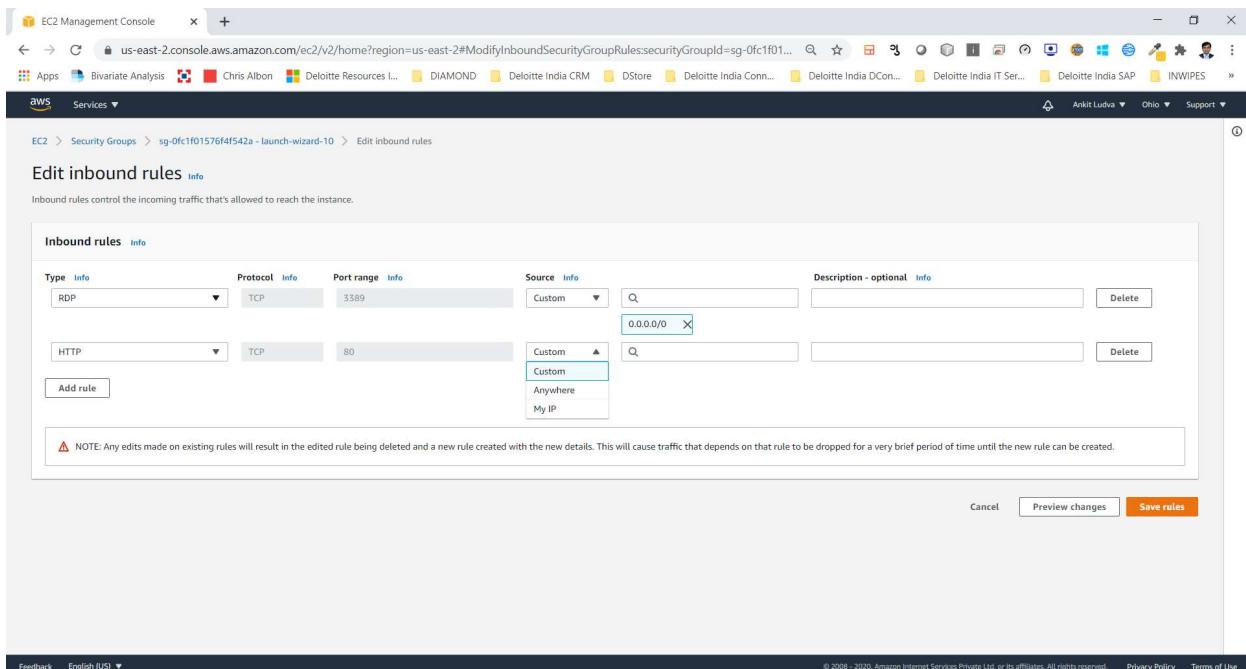
Cancel Preview changes Save rules

Feedback English (US) ▾

33. Select HTTP from the drop down



34. Select Anywhere from Source drop down



Inbound rules [Info](#)

Type Info	Protocol Info	Port range Info	Source Info	Description - optional Info
HTTP	TCP	80	Custom	0.0.0.0/0 X
HTTP	TCP	80	Custom	::/0 X
RDP	TCP	3389	Custom	0.0.0.0/0 X

[Add rule](#)

⚠ NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

[Cancel](#) [Preview changes](#) [Save rules](#)

35. Click on save rules.

Congratulation!!!! HTTP port configured successfully from AWS console and Windows firewall configuration.

Setup MySQL on Amazon RDS.

What is Amazon RDS?

Amazon Relational Database Service (Amazon RDS) is a web service that makes it easier to set up, operate, and scale a relational database in the cloud. It provides cost-efficient, resizable capacity for an industry-standard relational database and manages common database administration tasks.

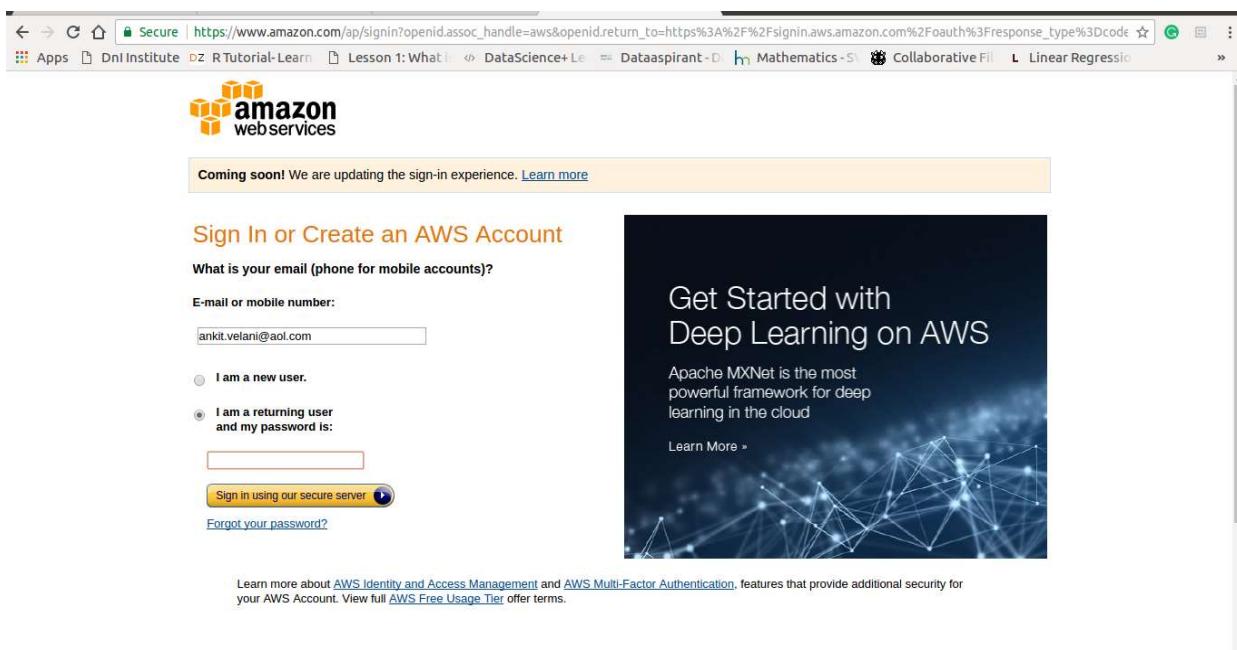
Why would you want a managed relational database service? Because Amazon RDS takes over many of the difficult or tedious management tasks of a relational database.

- When you buy a server, you get CPU, memory, storage, and IOPS, all bundled together. With Amazon RDS, these are split apart so that you can scale them independently. So, for example, if you need more CPU, less IOPS, or more storage, you can easily allocate them.
- Amazon RDS manages backups, software patching, automatic failure detection, and recovery.
- In order to deliver a managed service experience, Amazon RDS does not provide shell access to DB instances, and it restricts access to certain system procedures and tables that require advanced privileges.
- You can have automated backups performed when you need them, or create your own backup snapshot. These backups can be used to restore a database, and the Amazon RDS restore process works reliably and efficiently.
- You can get high availability with a primary instance and a synchronous secondary instance that you can failover to when problems occur. You can also use MySQL, MariaDB, or PostgreSQL Read Replicas to increase read scaling.
- You can use the database products you are already familiar with: MySQL, MariaDB, PostgreSQL, Oracle, Microsoft SQL Server, and the new, MySQL-compatible Amazon Aurora DB engine (for information, see Aurora on Amazon RDS).

- In addition to the security in your database package, you can help control who can access your RDS databases by using AWS IAM to define users and permissions. You can also help protect your databases by putting them in a virtual private cloud.

Setup MySQL on Amazon RDS.

1. Login to Amazon cloud portal. <https://console.aws.amazon.com>
2. Enter registered email and choose option 'I am returning user and my password is: _____'
3. Click on Sign in using our secure server.



4. After successfully login, Console landing page will appear.

The screenshot shows the AWS Home page. In the top left, there's a search bar with placeholder text "Find a service by name or feature (for example, EC2, S3 or VM, storage)." Below it, under "Recently visited services," are links for "Launch a virtual machine," "Build a web app," "Host a static website," "Connect an IoT device," "Start a development project," and "Register a domain." Further down, under "Learn to build," is a link to "See all." On the right side, there's a "Helpful tips" section with links to "Manage your costs" and "Create an organization," and an "Explore AWS" section with links to "New Product Announcements" and "Migrate from Oracle to Amazon Aurora."

5. Click on All Services ,

This screenshot is identical to the one above, but the "All services" link under "Recently visited services" has been clicked, causing the entire list of services to expand. Now, each service category like Compute, Storage, Database, etc., is shown with its sub-services listed below it.

6. Choose RDS, from category “ Database”

The screenshot shows the AWS Management Console navigation menu. The left sidebar lists several service categories:

- Storage**: S3, EFS, Glacier, Storage Gateway
- Database**: RDS, DynamoDB, ElastiCache, Redshift
- Networking & Content Delivery**: VPC, CloudFront, Direct Connect, Route 53
- Migration**: Application Discovery Service, DMS, Server Migration, Snowball
- Management Tools**: CloudWatch, CloudFormation, CloudTrail, Config, OpsWorks, Service Catalog, Trusted Advisor, Managed Services
- Game Development**: Amazon GameLift
- Mobile Services**: Mobile Hub, Cognito, Device Farm, Mobile Analytics, Pinpoint
- Security, Identity & Compliance**: IAM, Inspector, Certificate Manager, Directory Service, WAF & Shield, Artifact
- Analytics**: Athena, EMR, CloudSearch, Elasticsearch Service, Kinesis, Data Pipeline, QuickSight
- Application Services**: Step Functions, SWF, API Gateway, Elastic Transcoder
- Messaging**: Simple Queue Service, Simple Notification Service, SES
- Business Productivity**: WorkDocs, WorkMail, Amazon Chime

The right panel contains promotional text and links:

- Get started with the most scalable framework for deep learning in the cloud. [Learn more.](#)
- Build Applications with AWS Lambda. Run and scale code for Python, Node.js, Java, or C# without provisioning or managing servers. [Learn more.](#)
- Amazon DynamoDB. Fast and flexible NoSQL database service for any scale. [Learn more.](#)
- AWS Marketplace. Discover, procure, and deploy popular software products that run on AWS. [Learn more.](#)
- Have feedback? [Submit feedback](#) to tell us about your experience with the AWS Management Console.

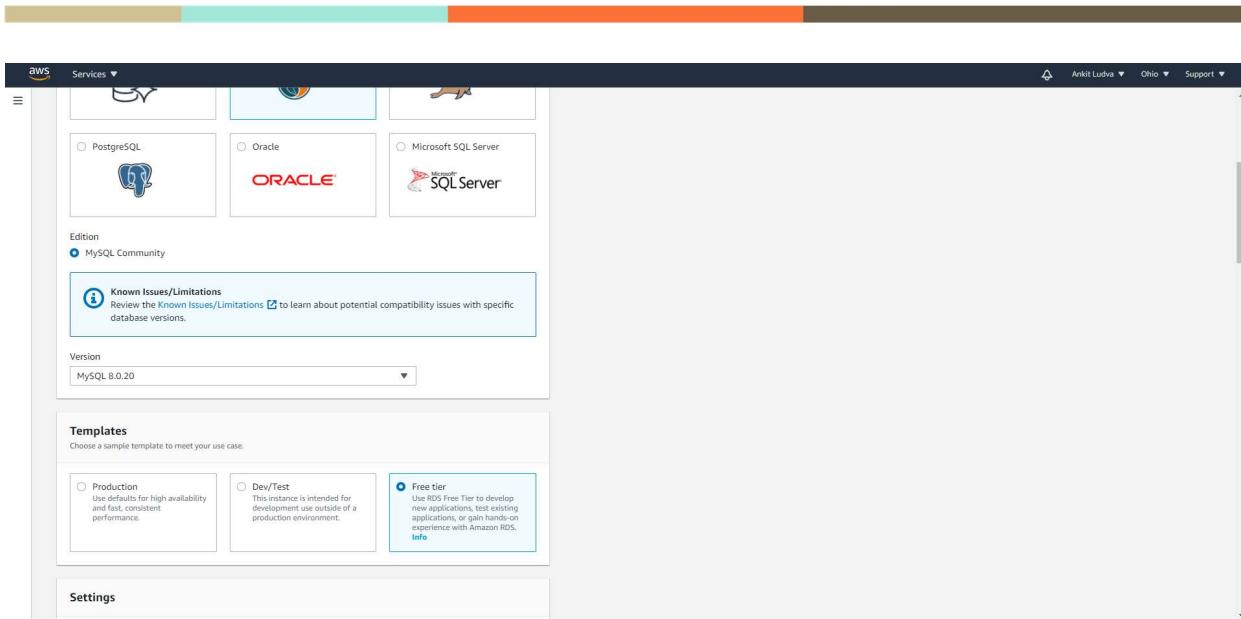
7. Click on **RDS**.

The screenshot shows the AWS RDS console. On the left, there's a sidebar with options like Dashboard, Databases, Query Editor, Performance Insights, etc. The main area is titled "Amazon Aurora" and contains information about it being a MySQL- and PostgreSQL-compatible database. It features a "Create database" button and a note about restoring from S3. Below this is a "Resources" section showing usage statistics for DB Instances, Parameter groups, Option groups, Snapshots, and Event subscriptions. To the right, there's a "Feature Spotlight" section with links to "RDS Storage Auto Scaling", "Aurora Serverless for PostgreSQL", and "Aurora Parallel Query".

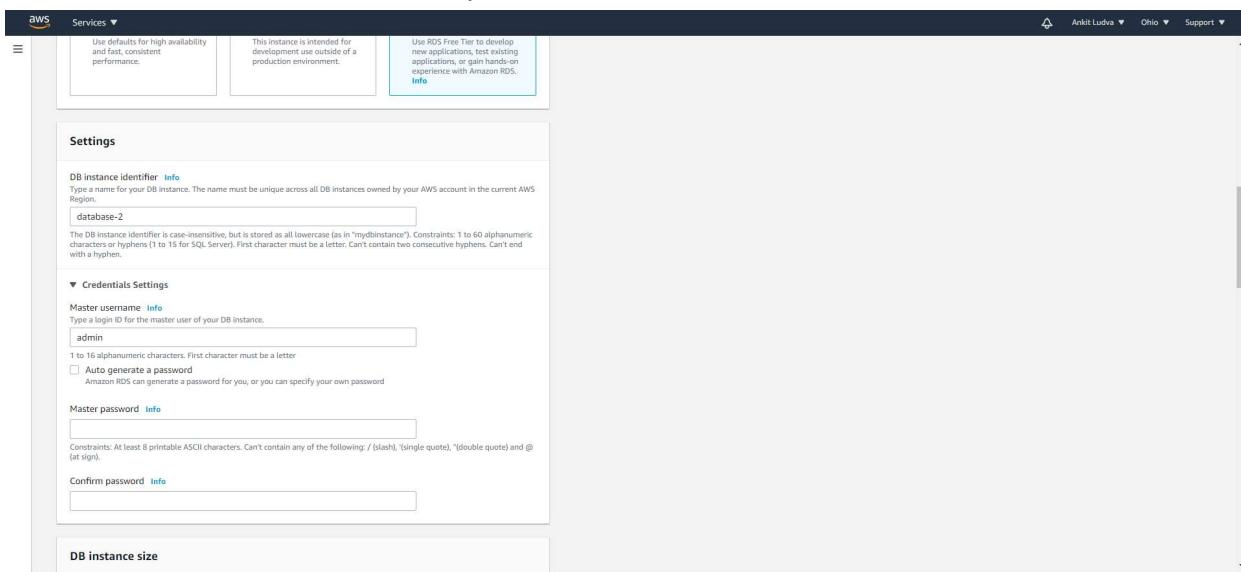
8. Click on **Create database**

The screenshot shows the "Create database" wizard. It starts with a step to "Choose a database creation method" where "Standard Create" is selected. Below this, the "Engine options" section allows choosing between various engines: Amazon Aurora, MySQL (selected), MariaDB, PostgreSQL, Oracle, and Microsoft SQL Server. Under "Edition", "MySQL Community" is chosen. At the bottom, there's a link to "Known Issues/Limitations".

9. Click on **Free tier eligible** only , Click on **MySQL**



10. Provide DB username and password.



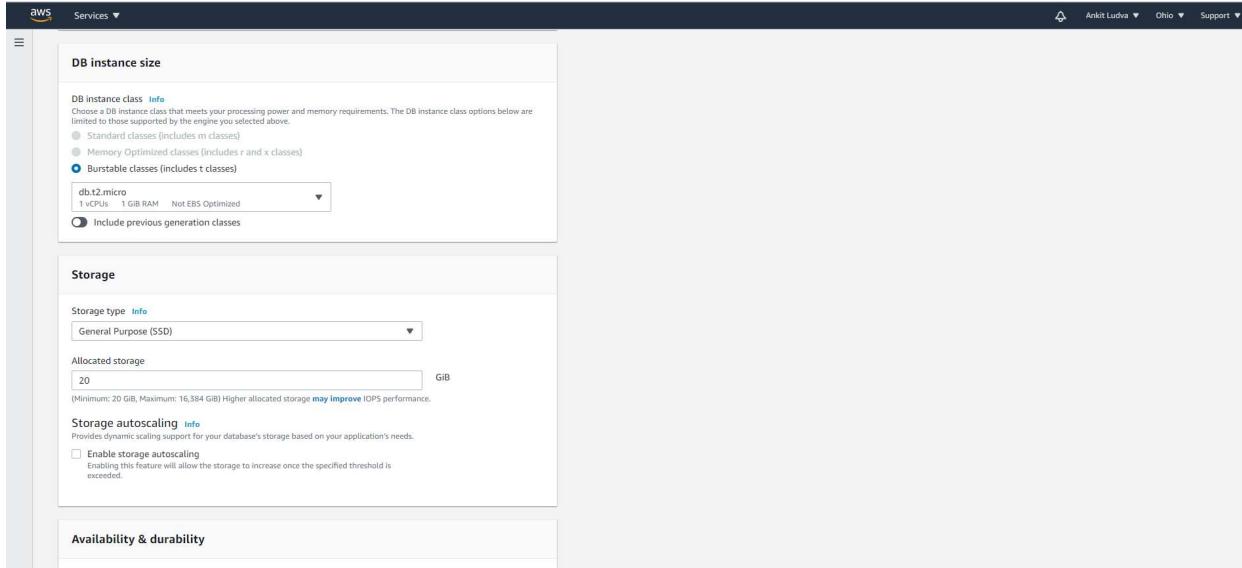
DBInstance Identifier :=====> Database Instance name: **dbserver**

Master Username :=====> root

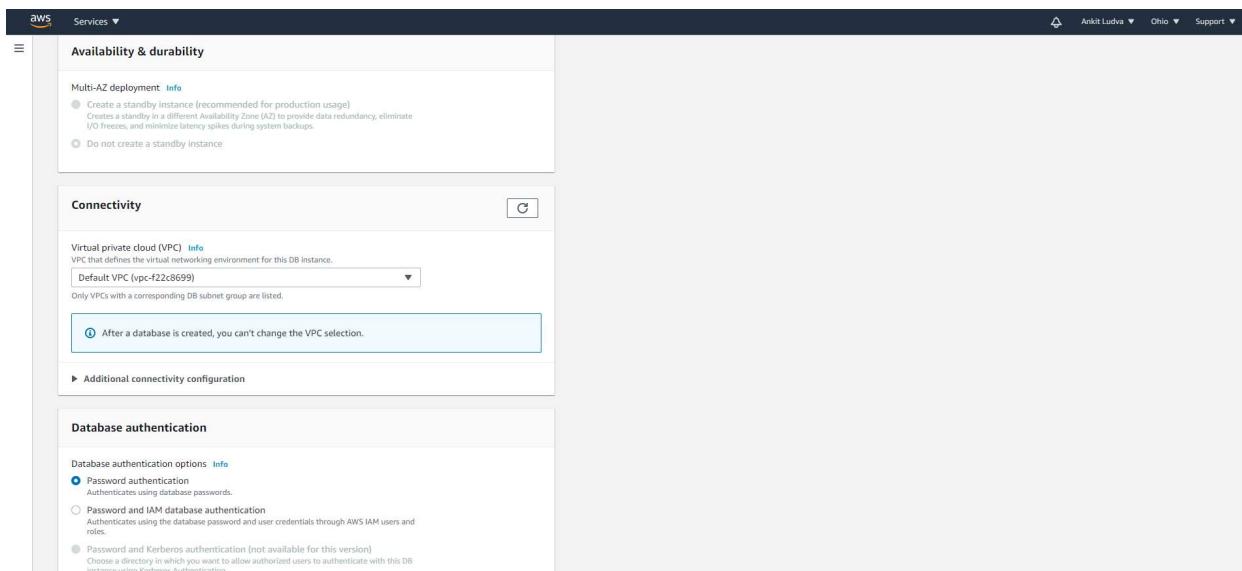
Master Password :=====> mypassword

Confirm Password :=====> mypassword

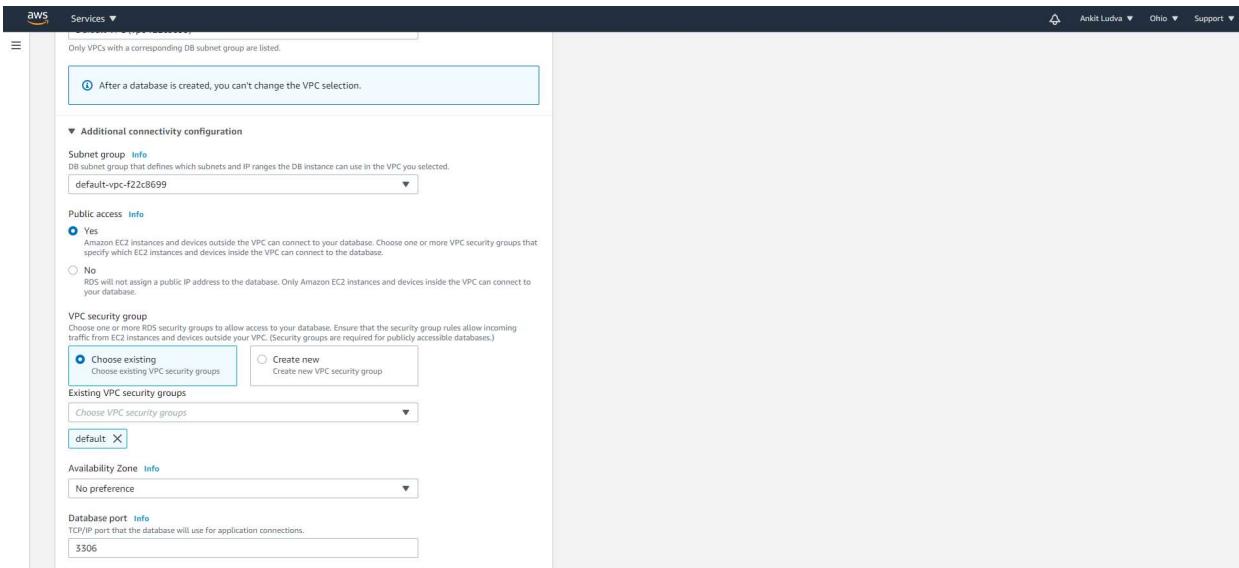
11. Free tier Instance configuration



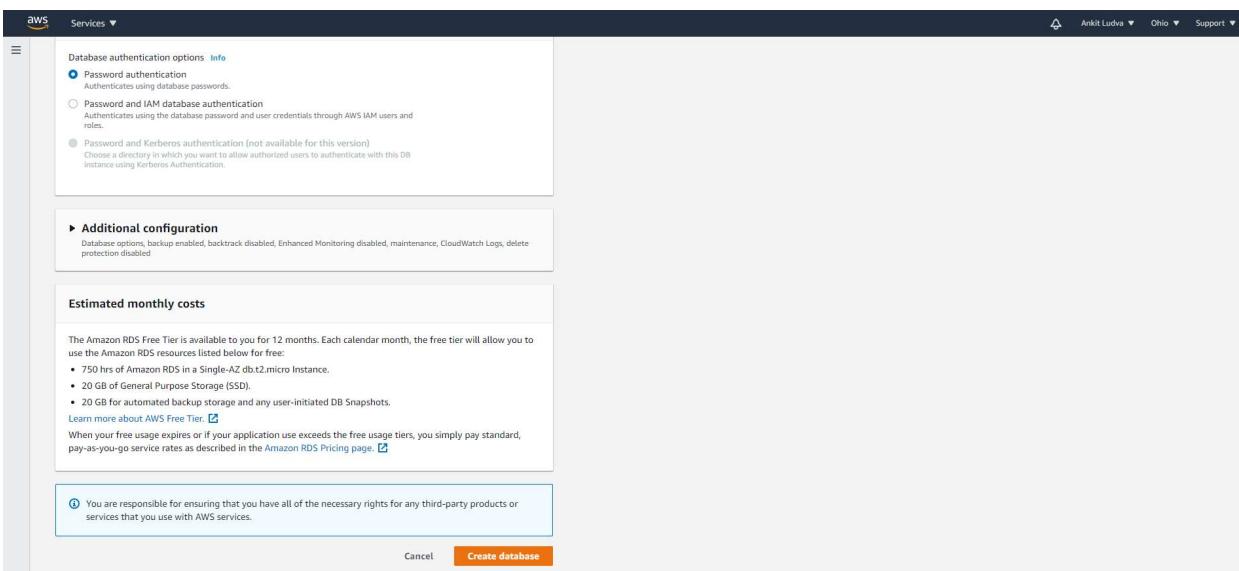
12. View to connectivity and allow as public access for the Database instance.



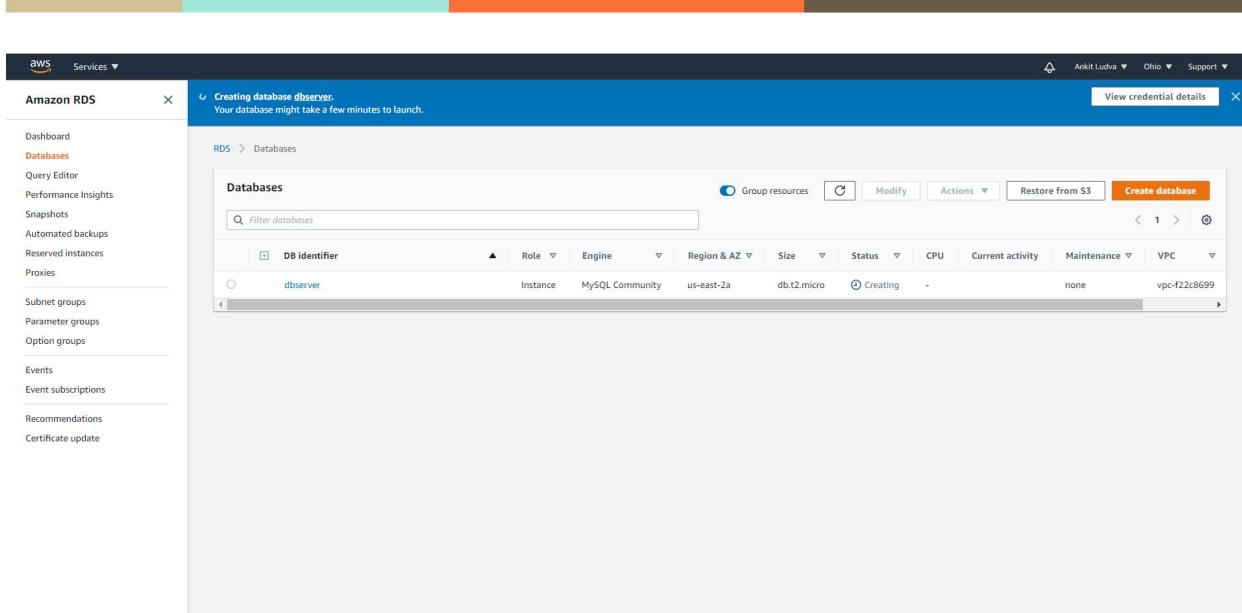
13. Click on Additional connectivity configuration, and select public access YES.



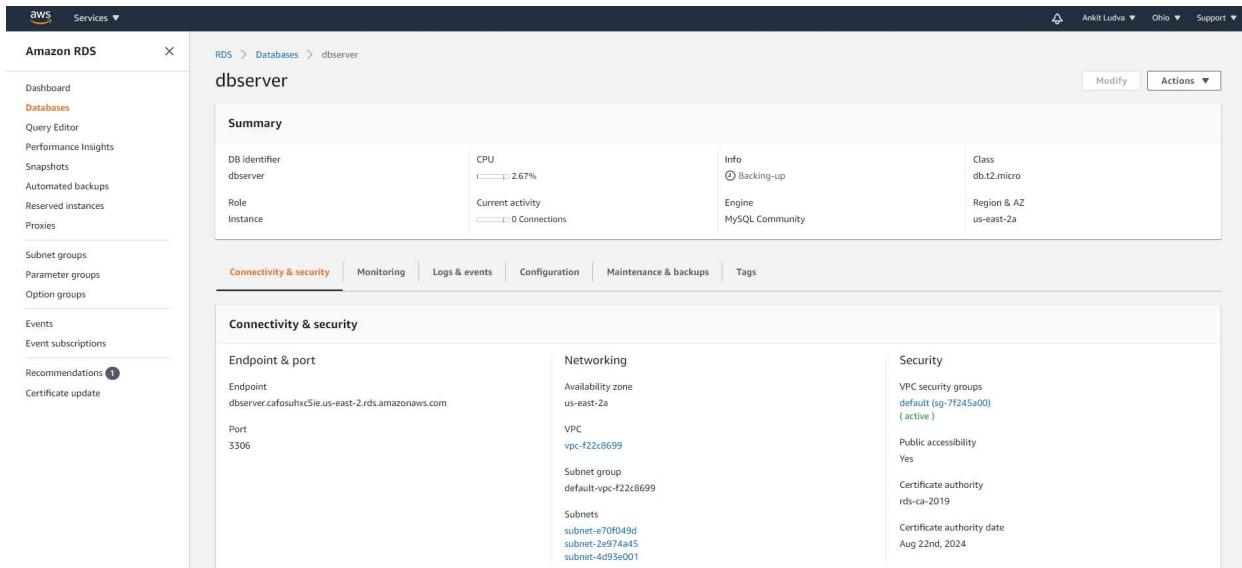
14. Scroll down and Click on **Create database**



15. After few minutes MySQL RDS instance will be available for use.



16. Once it available, select DB instance to view the endpoints and other configurations.



Congratulation :) Successfully created MySQL Database on Amazon RDS.

Working with RDS MySQL

1. Jump to RDS dashboard.

The screenshot shows the AWS RDS dashboard. On the left, there's a sidebar with links like Dashboard, Instances, Clusters, Performance Insights, Snapshots, Reserved instances, Subnet groups, Parameter groups, Option groups, Events, Event subscriptions, and Recommendations. The main area has a section for "Amazon Aurora" which says "Amazon Aurora is a MySQL- and PostgreSQL-compatible enterprise-class database, starting at <\$1/day. Aurora supports up to 64TB of auto-scaling storage capacity, 6-way replication across three availability zones, and 15 low-latency read replicas. Learn more." It includes a "Create database" button and a link to "Or, Restore Aurora DB cluster from S3". Below this is a "Resources" section showing usage statistics: DB Instances (1/40), Allocated storage (20.00 GB/100.00 TB), Reserved instances (0/40), Snapshots (33), and Recent events (0). To the right is an "Additional information" panel with links to Getting started with RDS, Overview and features, Documentation, Articles and tutorials, Data import guide for MySQL, Data import guide for Oracle, Data import guide for SQL Server, New RDS feature announcements, Pricing, and Forums. At the bottom is a "Create database" section with "Restore from S3" and "Create database" buttons.

2. Click on DB Instances.

The screenshot shows the AWS RDS Instances page. The sidebar is identical to the previous dashboard. The main area shows a table titled "Instances (1)". The table has columns for DB instance, Engine, Status, CPU, Current activity, Maintenance, Class, and VPC. There is one row for "db-server" which is MySQL, available, using 1.19% CPU, with 0 connections, no maintenance, db.t2.micro class, and vpc-f9e9da9 VPC. There are "Instance actions" and "Create database" buttons above the table.

The screenshot shows the AWS RDS console for a MySQL database instance named dbserver. The left sidebar includes links for Dashboard, Databases (selected), Query Editor, Performance Insights, Snapshots, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, Option groups, Events, Event subscriptions, Recommendations (1), and Certificate update.

Summary Tab:

DB identifier	CPU	Info	Class
dbserver	2.67%	Backing-up	db.t2.micro
Role	Current activity	Engine	Region & AZ
Instance	0 Connections	MySQL Community	us-east-2a

Connectivity & security Tab:

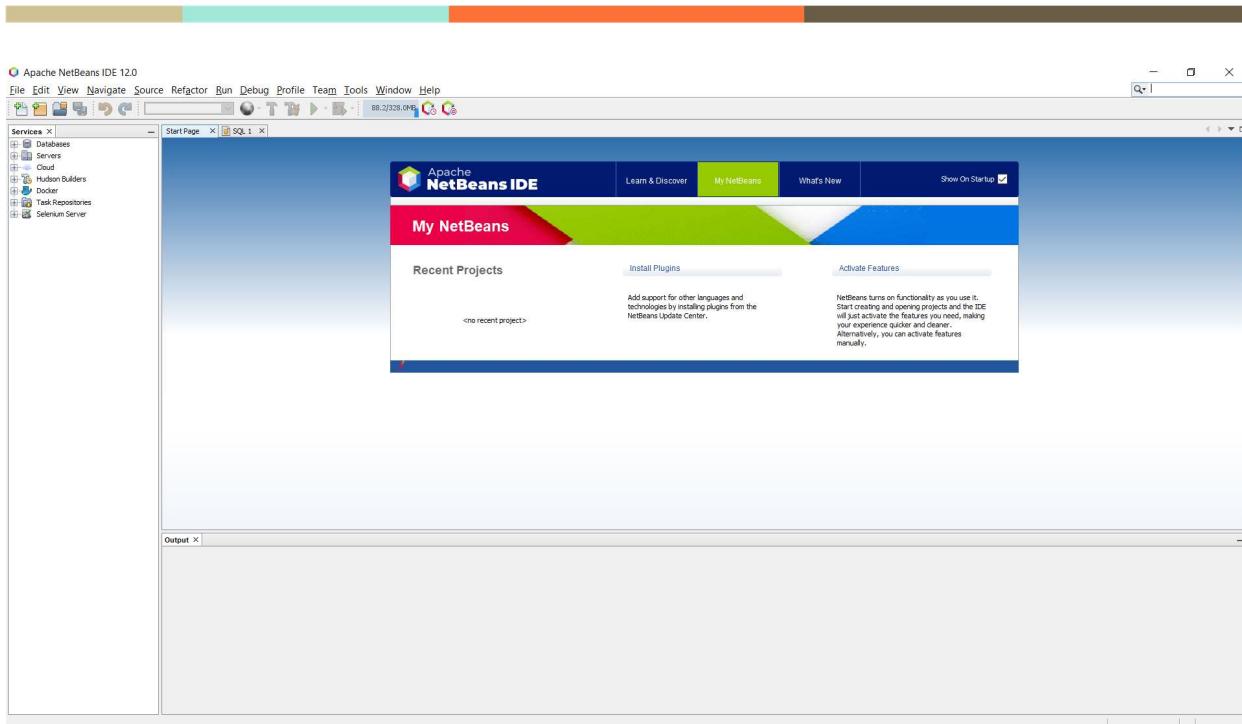
Endpoint & port	Networking	Security
Endpoint dbserver.cafosuhxc5ie.us-east-2.rds.amazonaws.com	Availability zone us-east-2a	VPC security groups default (sg-7f245a00) (active)
Port 3306	VPC vpc-f22c8699	Public accessibility Yes
	Subnet group default-vpc-f22c8699	Certificate authority rds-ca-2019
	Subnets subnet-e70f049d subnet-2e974a45 subnet-4d93e001	Certificate authority date Aug 22nd, 2024

3. Endpoint of the DB Instances help us to access MySQL.

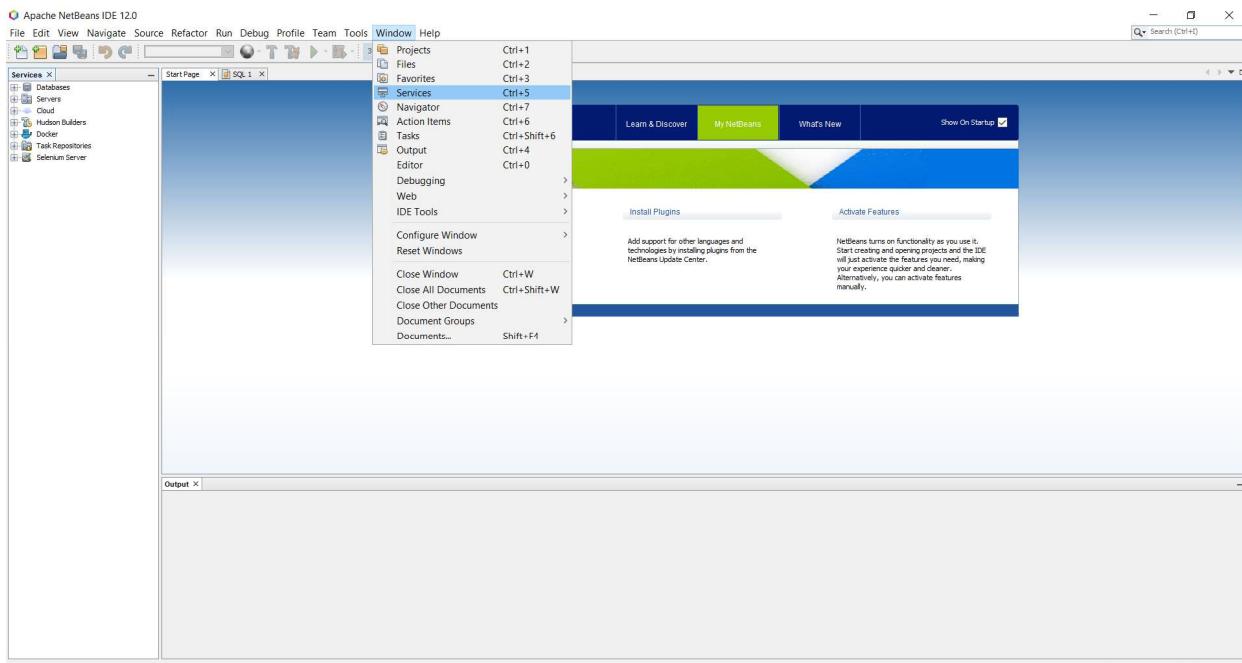
This screenshot shows the Connectivity & security tab for the dbserver instance. It displays the same detailed networking and security information as the previous screenshot, including VPC, subnet group, and certificate authority details.

4. We can use this Endpoint to connect with MySQL Database in any programming language or tools like MySQL Workbench, SqlWorkbenchJ, in NetBeans or Eclipse..Etc.

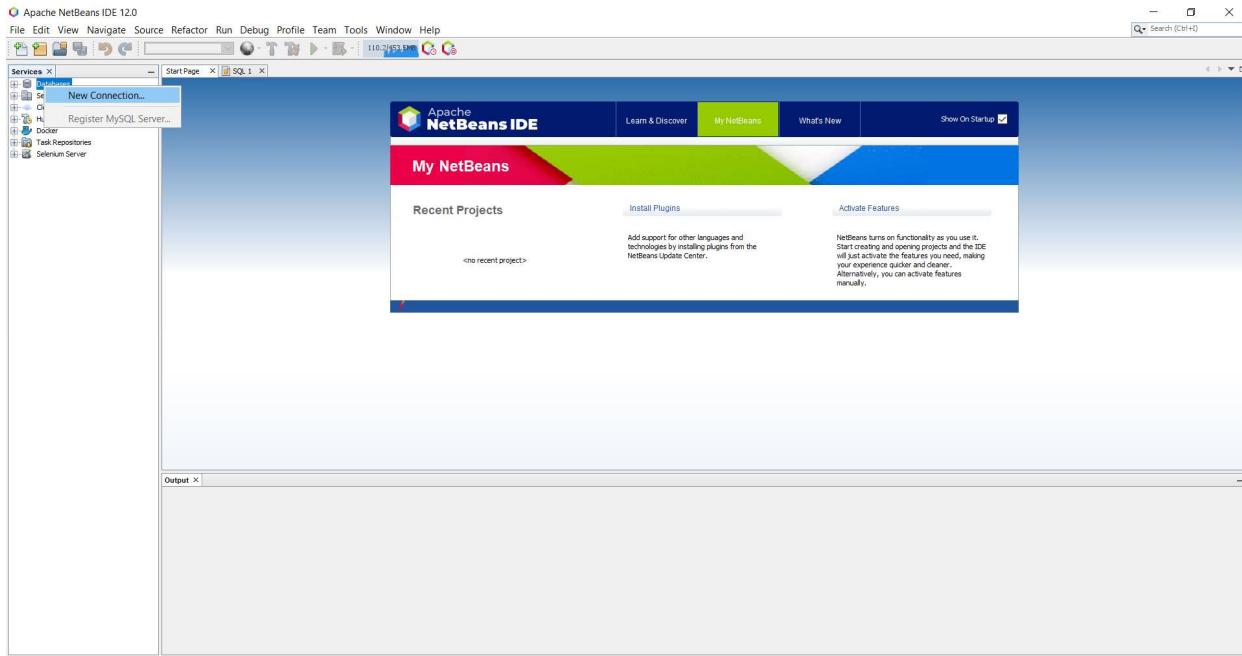
5. We will try to connect with NetBeans. Open **NetBeans**.



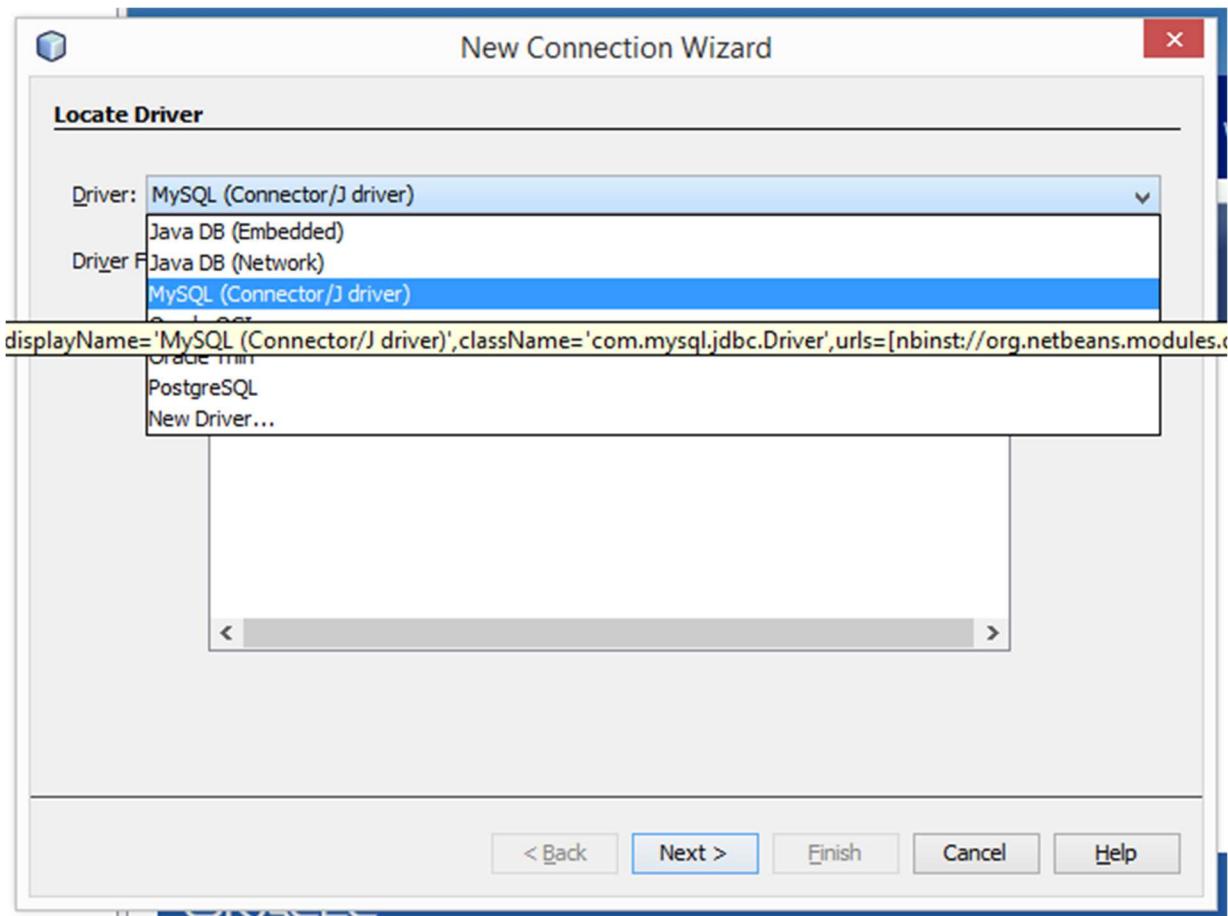
6. Go to Windows > Services



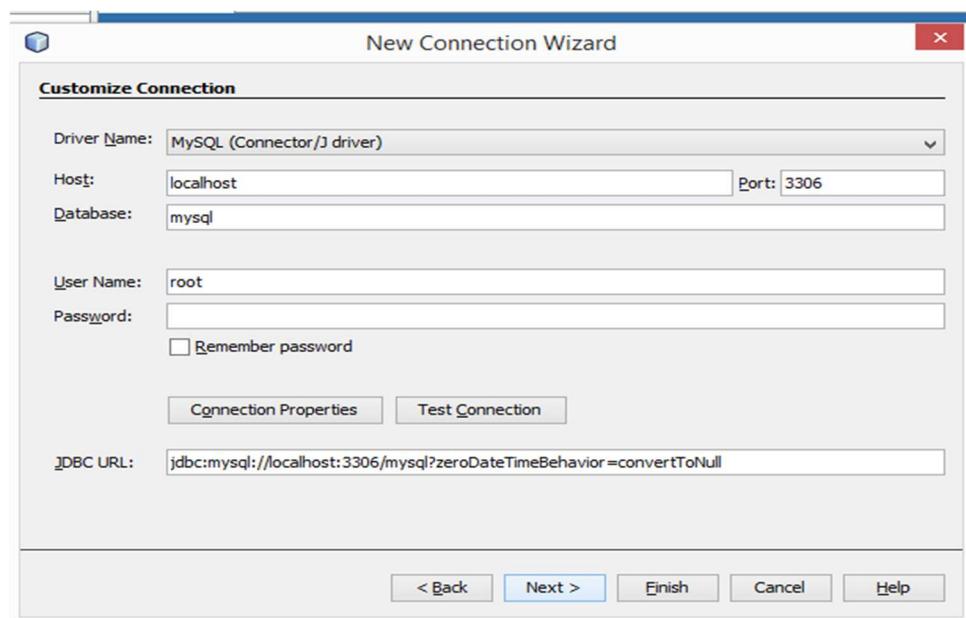
7. Right click on Databases > New Connection.



8. Select MySQL Driver from dropdown.



9. Click on **Next >**.



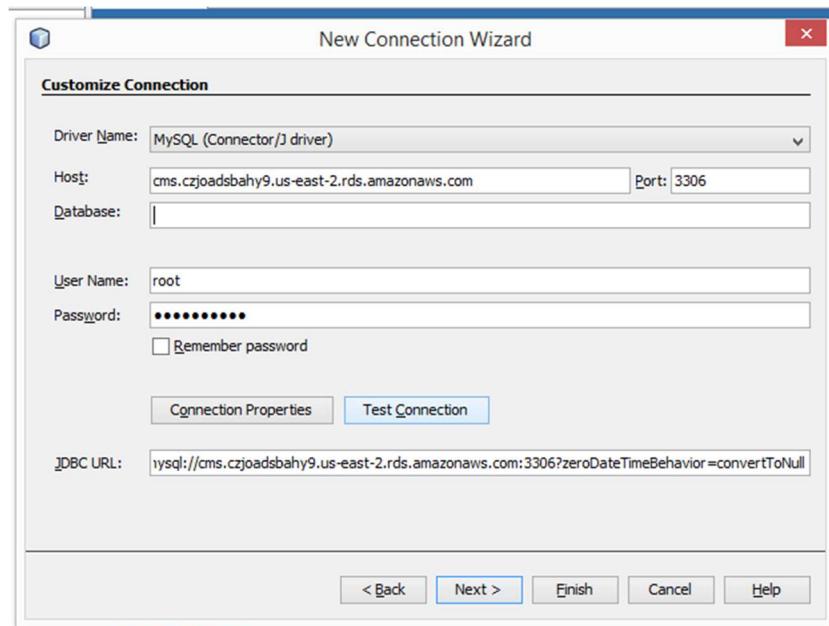
10. Provide connection details.

Host : Paste **Endpoint** of MySQL RDS.

Database: provide **database name** if you have already created otherwise **blank**.

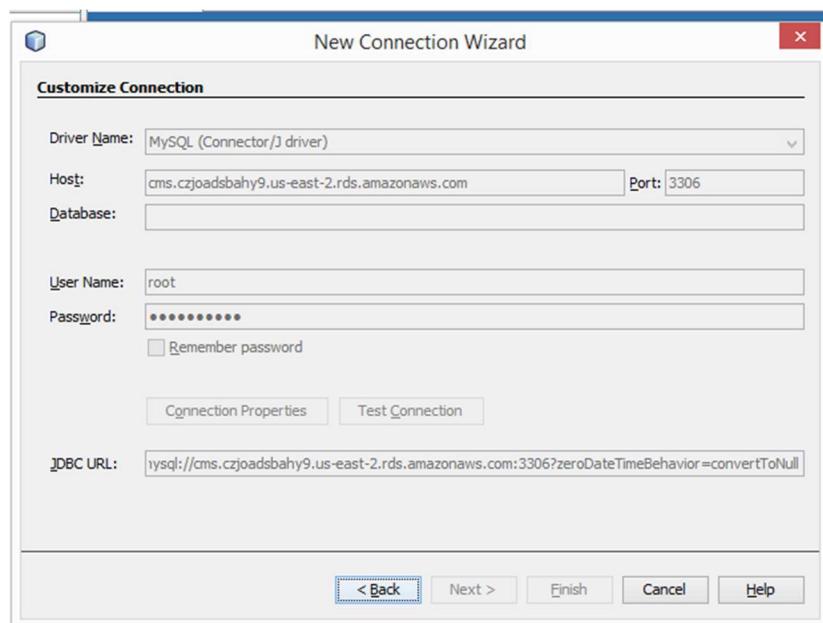
Username: Database **Username** (**root**)

Password: Database **Password** (**mypassword**)

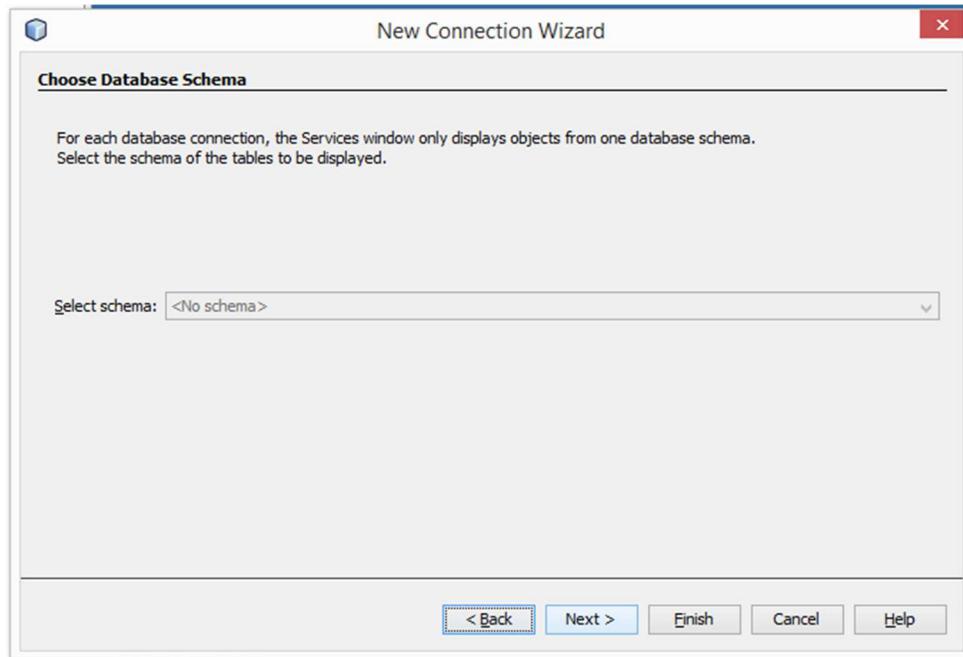


11. Click on Test Connection.

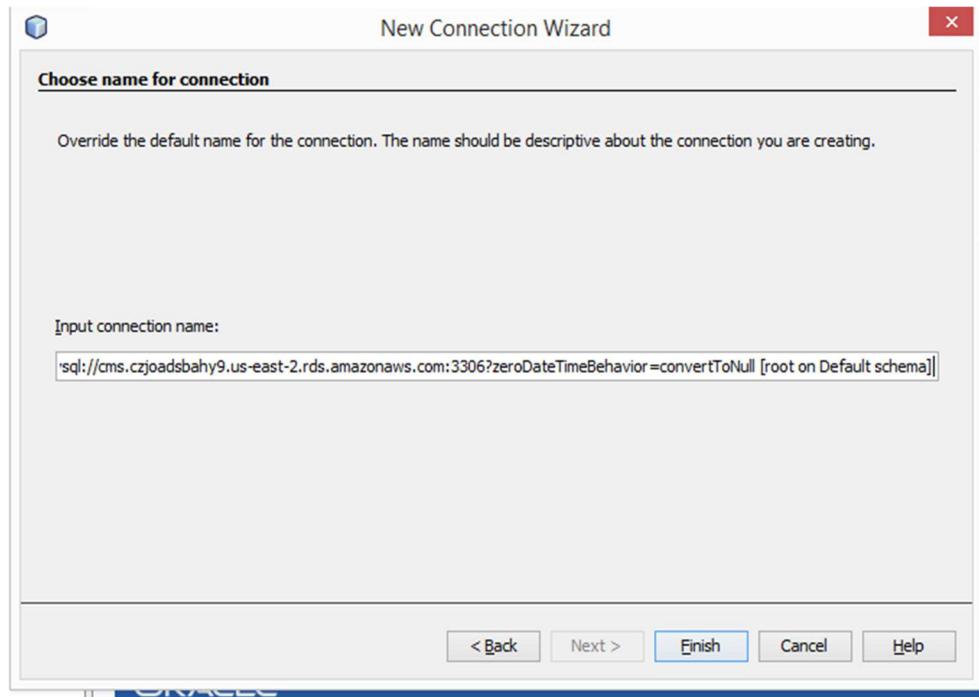
12. If Connection is successfully then click on Next.



13. Select Schema, We do not have any schema yet.

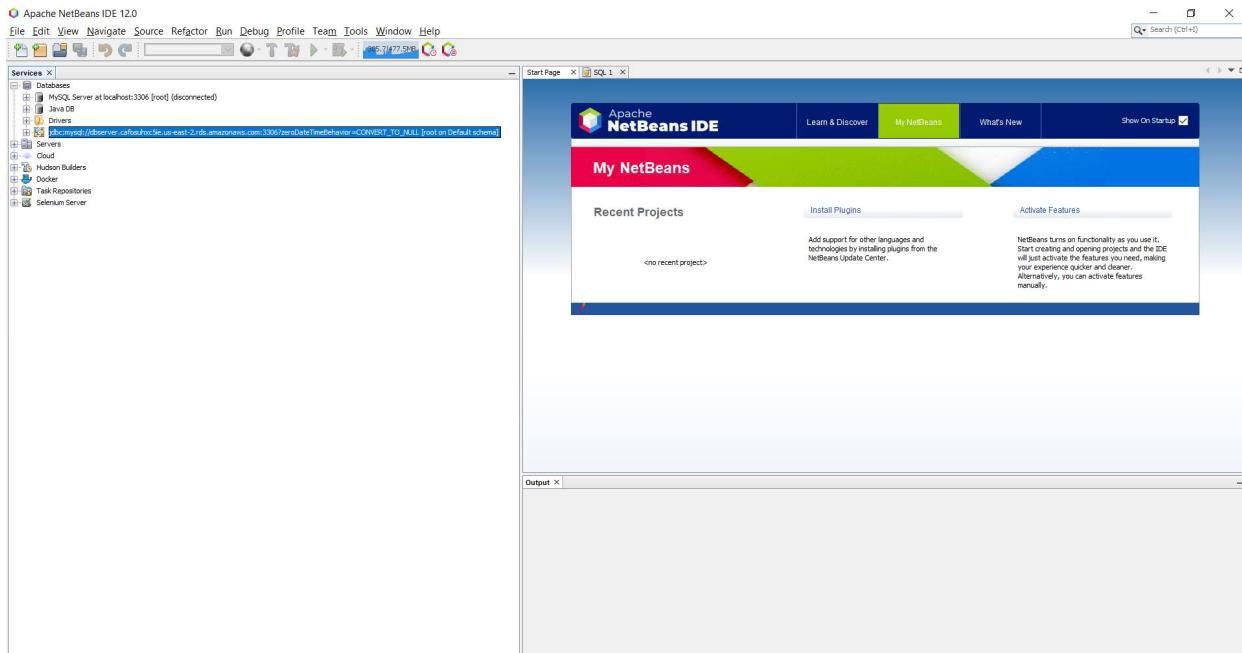


14. Click on Next.

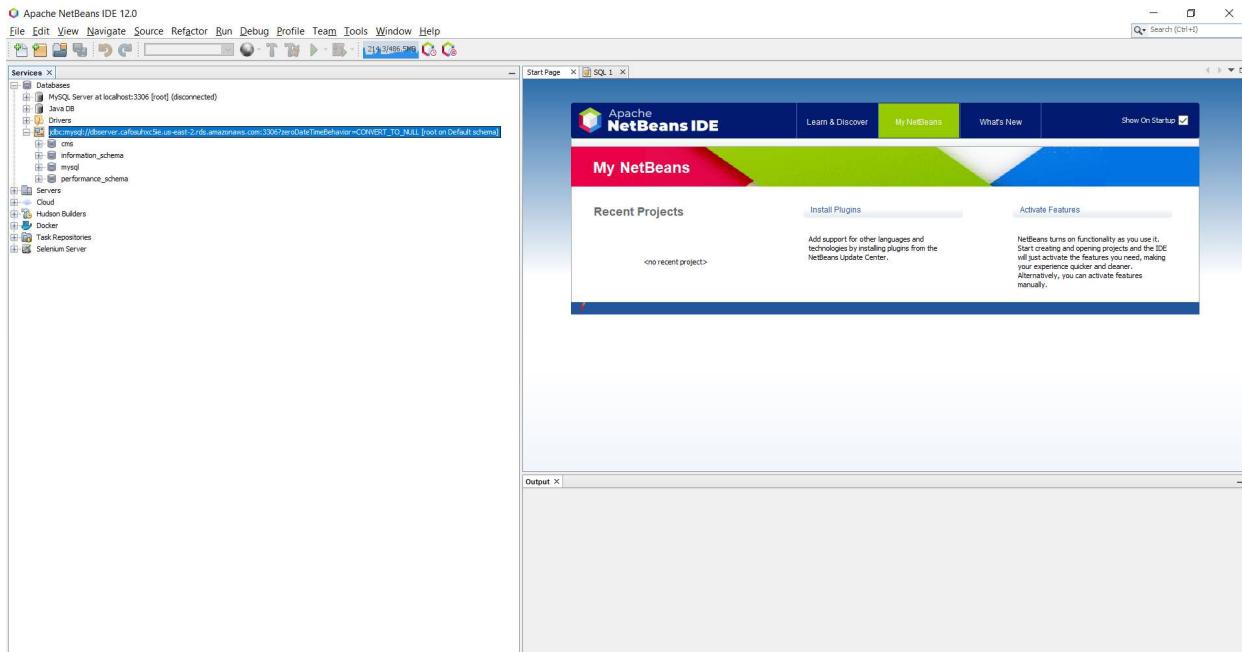


15. Click on Finish.

16. Now, Expand the Databases in NetBeans.



17. Expand MySQL RDS Connection String.



18. Database name will appear in the list. Now it's easy for us to create database, tables and also query on tables of MySQL RDS.

Congratulation :) Successfully Connected NetBeans with Amazon RDS.

