

# AWS: Amazon Web Services Lab Practice Guide

Document has been prepared for lab practice only not for production deployments

**Prepared for:**  
Public

**Prepared by:**  
Ankam Ravi Kumar

Follow Me on Social Networking Sites

[Facebook](#) | [Google Plus](#) | [Twitter](#) | [Reddit](#) | [LinkedIn](#) | [Website](#) | [Blog](#)

Reach me over Email: [aravikumar48@gmail.com](mailto:aravikumar48@gmail.com) or [aravi@server-computer.com](mailto:aravi@server-computer.com)

If you think this document helped a lot [Donate](#) a dollar as complementary

## Table of Contents

<b>1. About Author</b>	4
<b>2. Services we provide to our customers</b>	5
<b>3. Cloud Computing Models</b>	6
3.1. Infrastructure as a Service (IaaS):	6
3.2. Platform as a Service (PaaS):	6
3.3. Software as a Service (SaaS):	6
<b>4. Amazon Free Tier Account Creation</b>	7
<b>5. Enabling Multi-Factor Authentication to Secure Your Access</b>	11
<b>6. Creating First Linux Instance</b>	15
<b>7. Create your First EC2 windows instance</b>	20
<b>8. Assigning Elastic IP Addresses to Instance (Static IP Address)</b>	24
<b>9. Launching RDS Instance</b>	25
<b>10. Accessing MySQL Instance Using Workbench</b>	33
<b>11. AWS S3 Bucket</b>	38
11.1. AWS S3 Lifecycle Management	40
11.2. S3 Bucket Replication to Cross-Region	43
11.3. S3 Bucket Policies to control Access	45
<b>12. VPC – Virtual Private Cloud (isolated Network)</b>	46
12.1. Create subnets	48
12.2. Create Internet gateway and attach to VPC	49
12.3. Create Virtual Private Gateway and Attach to VPC	50
12.4. Create route tables and attach to subnets	51
<b>13. AWS Elastic Load Balancer (ELB)</b>	53



### 1. About Author

Ankam Ravi Kumar has more than 10+ years of experience in Information Technology Operations and production support streams. He served more than 5 companies in his career and still continuing.

We provide server and data center related services from purchasing of underlying hardware to provisioning the applications.

Solid industry experience in Infrastructure Management/Customer Support/Operations and Training Domains. I love to help people by sharing my knowledge and skills. I always believe “Power is gained by Sharing Knowledge not hoarding it”.

- Operating System Management Such has Linux Different Flavors, Red hat, Fedora, Ubuntu, AIX, Solaris and Windows
- Enterprise Server Management
- Installing and configuring Blade Servers
- Core Storage Management Dell-EMC, IBM and NetApp
- Database Management MSSQL, POSTGRESQL, MariaDB and MySQL
- Process Management ITIL
- Virtualization management RHEV, vSphere, VMware, KVM, Hyper-V and XEN
- Backup and Recovery Management NetVault, Commvault and Symantec Backup Exec
- Application Server Management and Storage Cluster Management
- Data Center Management and Hosting Solutions
- Programming Languages such as PHP and HTML
- Scripting Languages Shell, Perl and Python

Specialized in managing and building the Teams for IT services delivery and Service Support, Training and Operations in both smaller and larger companies. Rich experience and strong exposure in IT Infrastructure & Data Center Management.

Implementation of monitoring solutions for Enterprise, Using Tools Nagios, NagiosXI, Cacti, Solarwinds and LogicMonitor.

### 2. Services we provide to our customers



#### Data Storage

Any type of storage categories like DAS, NAS, SAN and Unified. Like Netapp, Dell-EMC, IBM, HP, Hitachi, Pure storage and Synology.



#### Backup and Recovery

We provide solutions for Online and Offline data backup. RPO and RTO less than ~5Minutes for any disaster recovery.



#### Networking

Switching and routing. Specialized in Paloalto firewall configurations and VPN. Spam filtering and proxy configurations.



#### Servers

Starting from server hardware configuration, requirement gathering to installing and configuring. Racking, Operating system and application to production. All brands.



#### Tape Libraries

We do provide tape library with backup software's. starting from LTO3, LTO4, LTO5, LTO6 and LTO7. Qualstar, Dell, Quantum, HP and IBM.



#### Telecommunication

Like PRI Lines, SIP, VoIP Services. Software and Hardware solutions for Inband and outband.



#### Virtualization

Virtualization environment implementation, configurations and migrations. Vmware, Hyper-V and RHEV.



#### Web Applications

Web application development. web designing and web development.



#### Application Migrations

We handle a large number of application migrations, data migrations from on-frame to cloud and cloud to on-frame. Any kind of old systems data CIFS shares, User data migrations we will handle with care.

### 3. Cloud Computing Models

There are three main models for cloud computing. Each model represents a different part of the cloud computing stack.

#### 3.1. Infrastructure as a Service (IaaS):

Infrastructure as a Service, sometimes abbreviated as IaaS, contains the basic building blocks for cloud IT and typically provide access to networking features, computers (virtual or on dedicated hardware), and data storage space. Infrastructure as a Service provides you with the highest level of flexibility and management control over your IT resources and is most similar to existing IT resources that many IT departments and developers are familiar with today.

#### 3.2. Platform as a Service (PaaS):

Platforms as a service remove the need for organizations to manage the underlying infrastructure (usually hardware and operating systems) and allow you to focus on the deployment and management of your applications. This helps you be more efficient as you don't need to worry about resource procurement, capacity planning, software maintenance, patching, or any of the other undifferentiated heavy lifting involved in running your application.

#### 3.3. Software as a Service (SaaS):

Software as a Service provides you with a completed product that is run and managed by the service provider. In most cases, people referring to Software as a Service are referring to end-user applications. With a SaaS offering you do not have to think about how the service is maintained or how the underlying infrastructure is managed; you only need to think about how you will use that particular piece software. A common example of a SaaS application is web-based email where you can send and receive email without having to manage feature additions to the email product or maintaining the servers and operating systems that the email program is running on.

## 4. Amazon Free Tier Account Creation

Read this conditions before creating a free tier account.

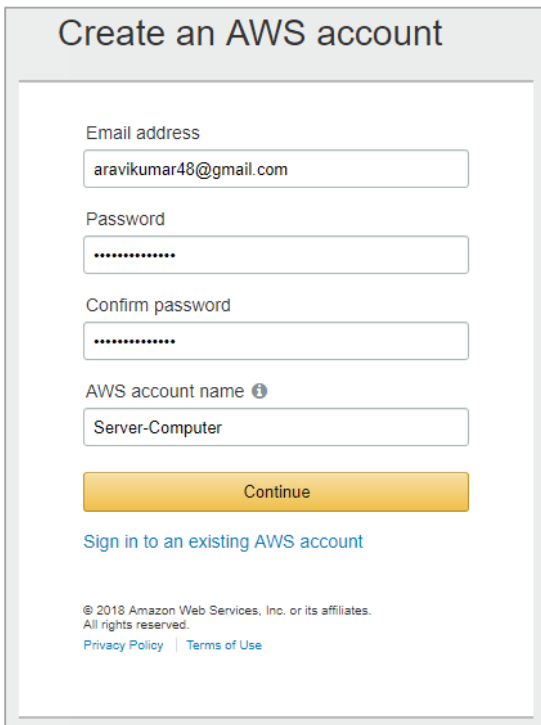
<https://aws.amazon.com/free/>

Prerequisites:

- Credit card with minimum 1\$ available balance
- Reachable mobile number for verification

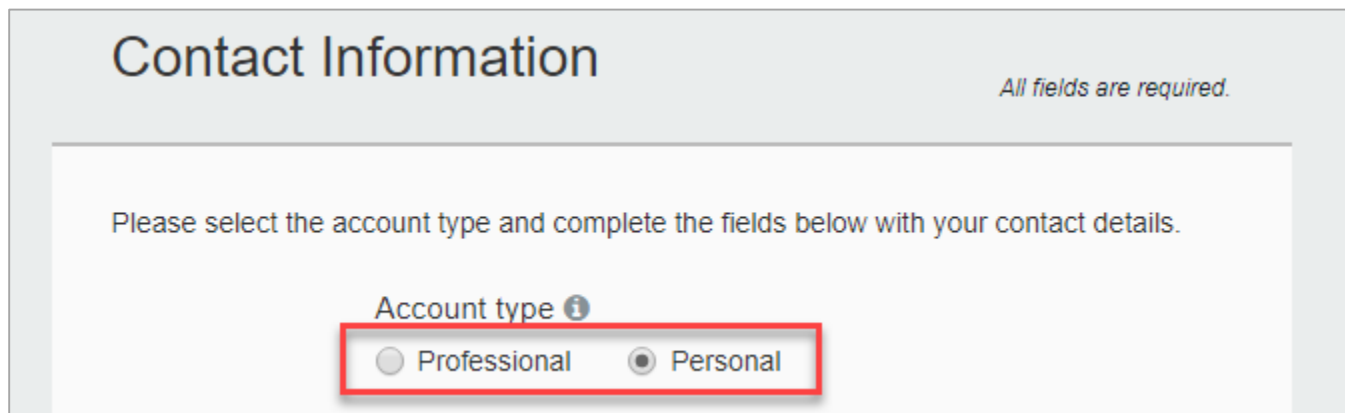
<https://aws.amazon.com/console/>

Click on 



The screenshot shows the 'Create an AWS account' form. It includes fields for 'Email address' (aravikumar48@gmail.com), 'Password' (masked with dots), 'Confirm password' (masked with dots), and 'AWS account name' (Server-Computer). A 'Continue' button is at the bottom. Below the button is a link 'Sign in to an existing AWS account'. At the very bottom, there is a copyright notice: '© 2018 Amazon Web Services, Inc. or its affiliates. All rights reserved.' and links for 'Privacy Policy' and 'Terms of Use'.

Fill the details example is shown above and **click continue**



**Contact Information** *All fields are required.*

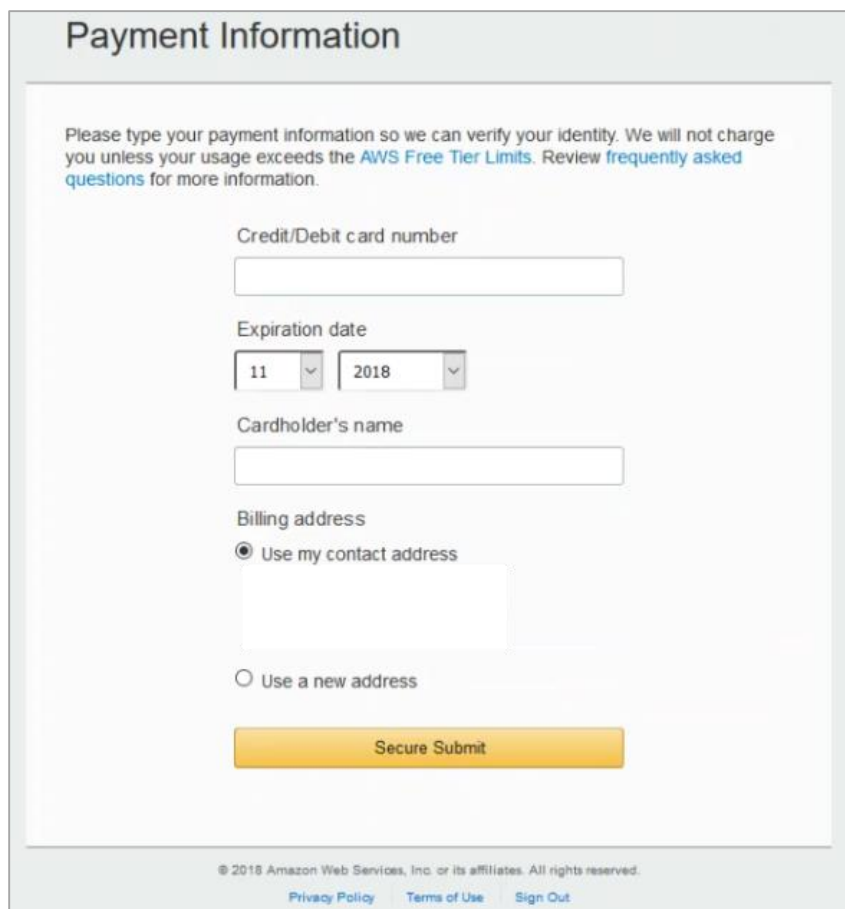
Please select the account type and complete the fields below with your contact details.

Account type ⓘ

☐ Professional ☒ Personal

Click on radio button

- Professional is for company
- Personal is for single person



**Payment Information**

Please type your payment information so we can verify your identity. We will not charge you unless your usage exceeds the [AWS Free Tier Limits](#). Review [frequently asked questions](#) for more information.

Credit/Debit card number

Expiration date

11 2018

Cardholder's name

Billing address

☒ Use my contact address

☐ Use a new address

Secure Submit

© 2018 Amazon Web Services, Inc. or its affiliates. All rights reserved.

[Privacy Policy](#) [Terms of Use](#) [Sign Out](#)

Provide your credit card details correctly, Card Number, Expiry Date and Card Holder Name

Click on **Secure Submit**



### Phone Verification

AWS will call you immediately using an automated system. When prompted, enter the 4-digit number from the AWS website on your phone keypad.

**Provide a telephone number**




Please enter your information below and click the "Call Me Now" button.

Country/Region code

India (+91)

Phone number  Ext

Security Check

Please type the characters as shown above

© 2018 Amazon Web Services, Inc. or its affiliates. All rights reserved.

[Privacy Policy](#) [Terms of Use](#) [Sign Out](#)

It will ask you to enter phone number, Security check then click on **Call Me Now**

Call in progress...

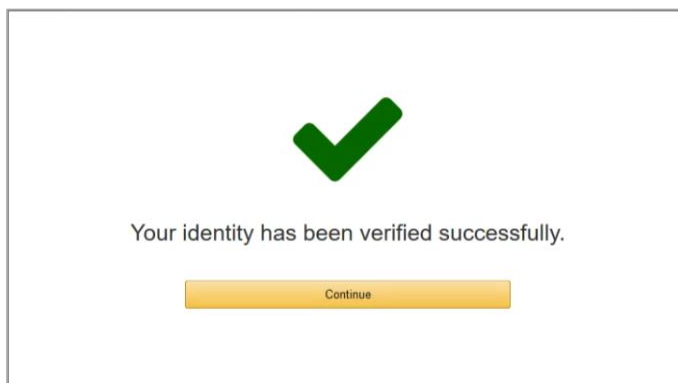
Please answer the call from AWS and, when prompted, enter the 4-digit number on your phone keypad.

2 9 0 2

You will receive a call from AWS tele communication and ask you to enter the code displayed on screen.

**Note:** Listen All the Details carefully and proceed by entering code displayed on screen.

After successful verification



## Continue

Select a Support Plan

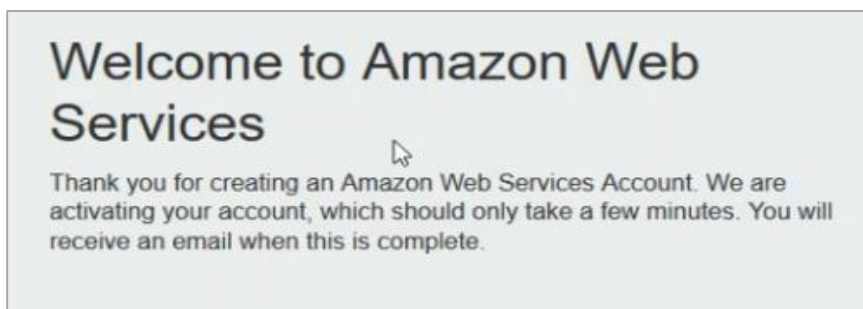
AWS offers a selection of support plans to meet your needs. Choose the support plan that best aligns with your AWS usage. [Learn more](#)

Basic Plan	Developer Plan	Business Plan
<b>Free</b>	From \$29/month	From \$100/month
<ul style="list-style-type: none"><li>• Included with all accounts</li><li>• 24/7 self-service access to forums and resources</li><li>• Best practice checks to help improve security and performance</li><li>• Access to health status and notifications</li></ul>	<ul style="list-style-type: none"><li>• For early adoption, testing and development</li><li>• Email access to AWS Support during business hours</li><li>• 1 primary contact can open an unlimited number of support cases</li><li>• 12-hour response time for nonproduction systems</li></ul>	<ul style="list-style-type: none"><li>• For production workloads &amp; business-critical dependencies</li><li>• 24/7 chat, phone, and email access to AWS Support</li><li>• Unlimited contacts can open an unlimited number of support cases</li><li>• 1-hour response time for production systems</li></ul>

**Need Enterprise level support?**  
Contact your account manager for additional information on running business and mission critical-workloads on AWS (starting at \$15,000/month). [Learn more](#)

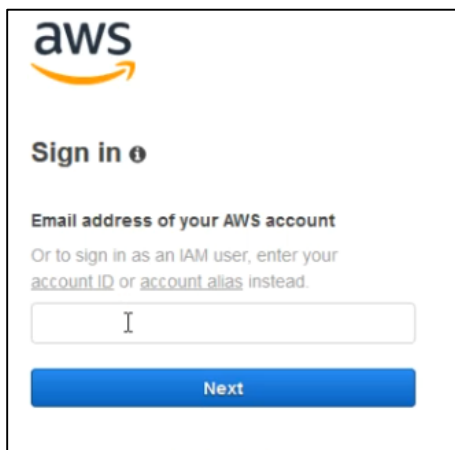
© 2018 Amazon Web Services, Inc. or its affiliates. All rights reserved.  
[Privacy Policy](#) [Terms of Use](#) [Sign Out](#)

Select Support plan in this case select **Free**



You successfully completed Free Tier Account Creation. Login and Enjoy AWS Free Tier.

### AWS Console



The image shows the AWS Sign in page. At the top is the AWS logo. Below it is the text "Sign in" with a help icon. Underneath is the prompt "Email address of your AWS account" followed by a smaller line of text: "Or to sign in as an IAM user, enter your [account ID](#) or [account alias](#) instead." There is a text input field with a cursor. Below the input field is a blue button labeled "Next".

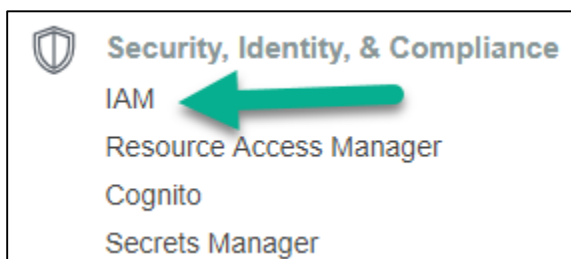


The image shows the AWS Root user sign in page. At the top is the AWS logo. Below it is the text "Root user sign in" with a help icon. Underneath is the "Email:" label followed by a text input field. Below that is the "Password" label followed by a text input field and a link "Forgot password?". At the bottom is a blue button labeled "Sign in". Below the button are two links: "Sign in to a different account" and "Create a new AWS account".

Provide your email address and password to **Sign In**

## 5. Enabling Multi-Factor Authentication to Secure Your Access

Go To IAM under Services → Security, Identify & Compliance → IAM



Click on Users → Add User

**Add user** 1 2 3 4 5

**Set user details**

You can add multiple users at once with the same access type and permissions. [Learn more](#)

User name\* administrator

+ Add another user

**Select AWS access type**

Select how these users will access AWS. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

**Access type\***

- ☐ **Programmatic access**  
Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.
- ☒ **AWS Management Console access**  
Enables a **password** that allows users to sign-in to the AWS Management Console.

**Console password\***

- ☐ Autogenerated password
- ☒ Custom password

\*\*\*\*\*

☐ Show password

**Require password reset** ☐ User must create a new password at next sign-in  
Users automatically get the [IAMUserChangePassword](#) policy to allow them to change their own password.

\* Required

Cancel **Next: Permissions**

Provide user name, select access type

- Programmatic Access – Required for automation, run any operation using programs
- AWS Management Console Access – User will have web console access

Click **Next Permissions**

**Add user** 1 2 3 4 5

▼ Set permissions

+ Add user to group

+ Copy permissions from existing user

+ Attach existing policies directly

Create policy

Filter policies Search Showing 375 results

	Policy name	Type	Used as	Description
<input checked="" type="checkbox"/>	AdministratorAccess	Job function	Permissions policy (1)	Provides full access to AWS services and...
<input type="checkbox"/>	AlexaForBusinessD...	AWS managed	None	Provide device setup access to AlexaFor...
<input type="checkbox"/>	AlexaForBusinessF...	AWS managed	None	Grants full access to AlexaForBusiness r...
<input type="checkbox"/>	AlexaForBusinessG...	AWS managed	None	Provide gateway execution access to Ale...
<input type="checkbox"/>	AlexaForBusinessR...	AWS managed	None	Provide read only access to AlexaForBus...
<input type="checkbox"/>	AmazonAPIGatewa...	AWS managed	None	Provides full access to create/edit/delete ...
<input type="checkbox"/>	AmazonAPIGatewa...	AWS managed	None	Provides full access to invoke APIs in Am...
<input type="checkbox"/>	AmazonAPIGatewa...	AWS managed	None	Allows API Gateway to push logs to user'...

► Set permissions boundary

Cancel Previous **Next: Tags**

Click **Next: Tags**

Add tags whatever required to identify user

## Add user

12**3**45

[www.server-computer.com](#)

### Add tags (optional)

IAM tags are key-value pairs you can add to your user. Tags can include user information, such as an email address, or can be descriptive, such as a job title. You can use the tags to organize, track, or control access for this user. [Learn more](#)

Key	Value (optional)	Remove
Created Date:	25th Oct 2018	×
Description	Administrator for My ABC Client	×
Add new key		

You can add 48 more tags.

[Cancel](#)[Previous](#)[Next: Review](#)

Click **Next: Review**

## Add user

123**4**5

[www.server-computer.com](#)

### Review

Review your choices. After you create the user, you can view and download the autogenerated password and access key.

#### User details

User name	administrator
AWS access type	AWS Management Console access - with a password
Console password type	Custom
Require password reset	No
Permissions boundary	Permissions boundary is not set

#### Permissions summary

The following policies will be attached to the user shown above.

Type	Name
Managed policy	<a href="#">AdministratorAccess</a>

#### Tags

The new user will receive the following tags

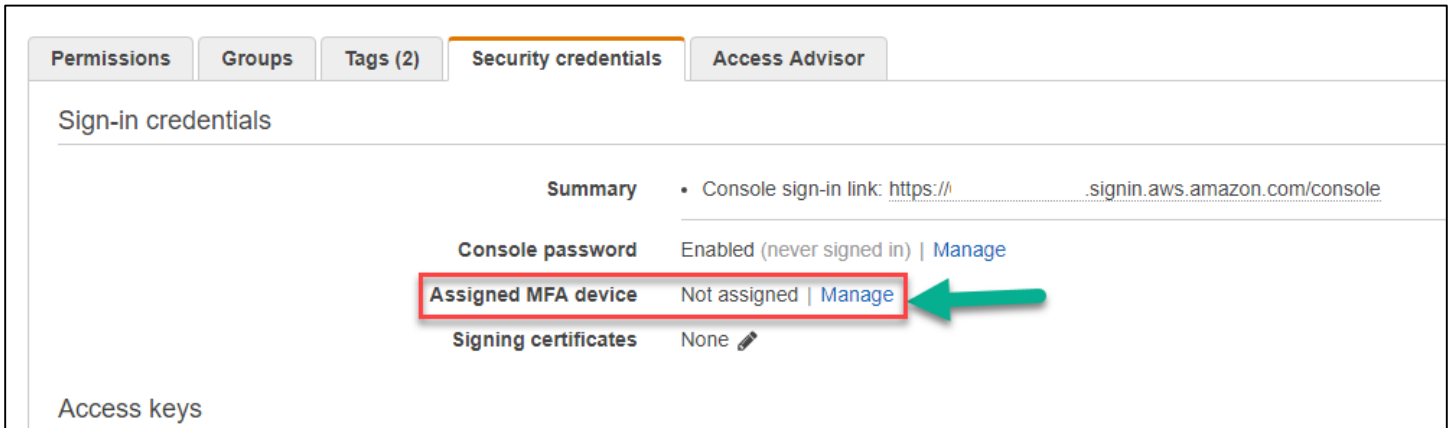
Key	Value
Created Date:	25th Oct 2018
Description	Administrator for My ABC Client

[Cancel](#)[Previous](#)[Create user](#)

Click **Create User**

User creation has been completed successfully now you will get on access URL with your account number. Note the URL.

Now Click on User name → Security credentials

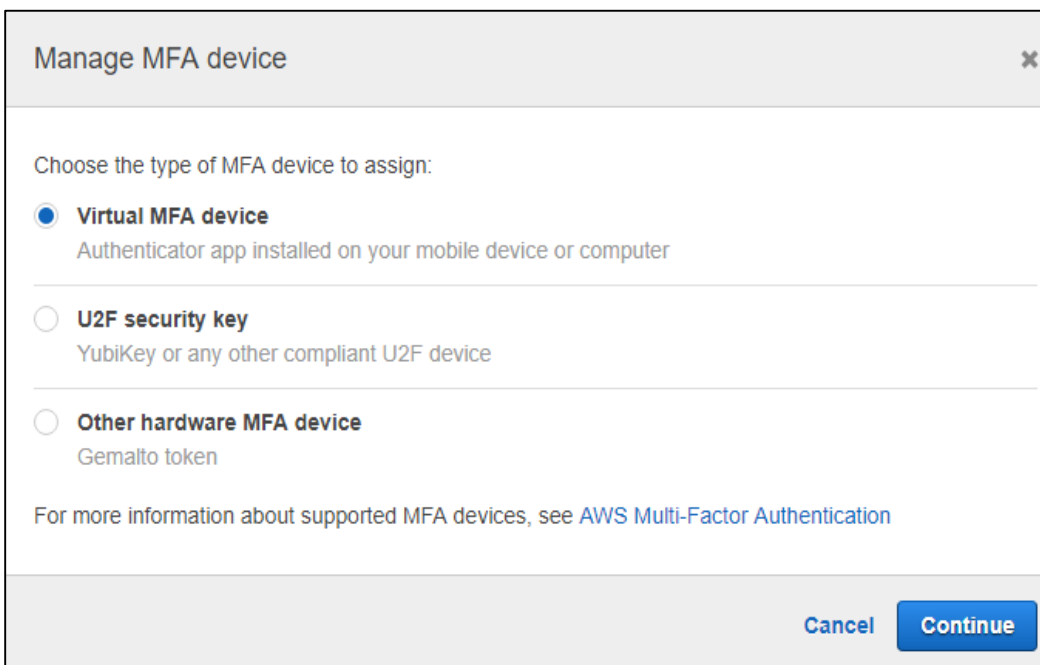


The screenshot shows the 'Security credentials' tab in the AWS IAM console. It displays the 'Sign-in credentials' section with a summary of the user's security settings. The 'Assigned MFA device' row is highlighted with a red box, and a green arrow points to the 'Manage' link next to it.

Sign-in credentials	
Summary	• Console sign-in link: <a href="https://...signin.aws.amazon.com/console">https://...signin.aws.amazon.com/console</a>
Console password	Enabled (never signed in)   <a href="#">Manage</a>
Assigned MFA device	Not assigned   <a href="#">Manage</a>
Signing certificates	None

Access keys

Click on Assigned MFA Device – Manage



The screenshot shows the 'Manage MFA device' dialog box. It prompts the user to choose the type of MFA device to assign. The 'Virtual MFA device' option is selected, and the 'Continue' button is highlighted.

Manage MFA device

Choose the type of MFA device to assign:

- ☒ **Virtual MFA device**  
Authenticator app installed on your mobile device or computer
- ☐ **U2F security key**  
YubiKey or any other compliant U2F device
- ☐ **Other hardware MFA device**  
Gemalto token

For more information about supported MFA devices, see [AWS Multi-Factor Authentication](#)

[Cancel](#) [Continue](#)

Use any method based on your requirement. Here I am showing Virtual MFA Device method

Install Google Authenticator in your smart phone and ready to pair

Click **Continue**

Set up virtual MFA device

1. Install a compatible app on your mobile device or computer  
See a list of compatible applications

2. Use your virtual MFA app and your device's camera to scan the QR code

Show QR code

[www.server-computer.com](#)

Alternatively, you can type the secret key. [Show secret key](#)

3. Type two consecutive MFA codes below

MFA code 1

MFA code 2

Cancel

Previous

Assign MFA

Click in Show QR Code and scan the same code from your Google authenticator App. It will generate 6 digit codes enter one code in first MFA code1 and second one in MFA Code 2 Click on **Assign MFA**

Set up virtual MFA device

✓

You have successfully assigned virtual MFA  
This virtual MFA will be required during sign-in.

[www.server-computer.com](#)

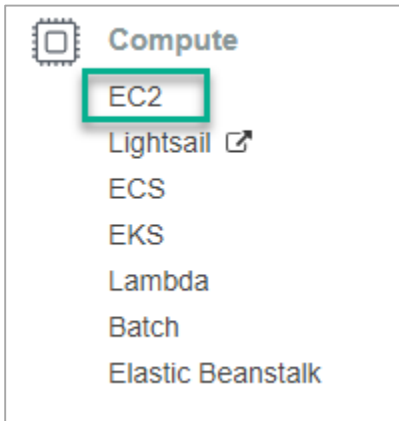
Close

That's it, now you successfully enabled MFA (Multi-Factor Authentication).

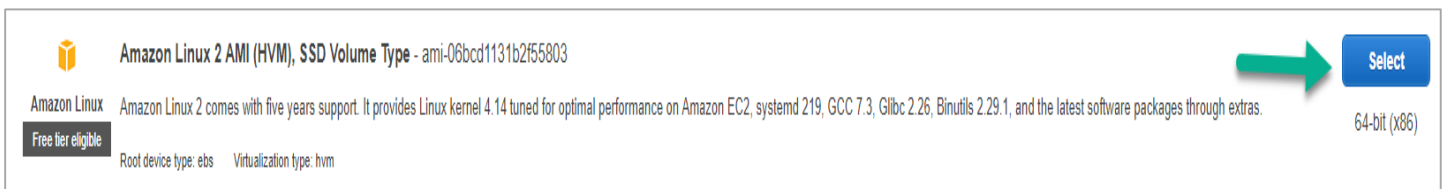
Here after if you want to login you have to enter credentials and MFA code to Login.

## 6. Creating First Linux Instance

Login to AWS console, services drop down click on EC2



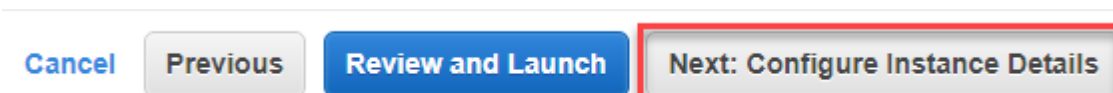
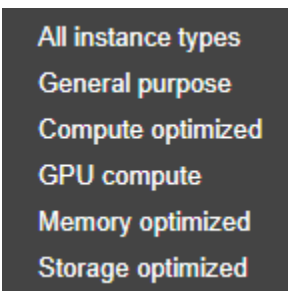
Click on **Launch instance**



I am selecting Free Tier instance Amazon Linux

	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 <small>Free tier eligible</small>	1	1	EBS only	-	Low to Moderate	Yes

We have below types of instances





## Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances	1	Launch into Auto Scaling Group
Purchasing option	<input type="checkbox"/> Request Spot instances	
Network	vpc-cbd4f2a3 (default)	Create new VPC
Subnet	No preference (default subnet in any Availability Zone)	Create new subnet
Auto-assign Public IP	Use subnet setting (Enable)	
Placement group	<input type="checkbox"/> Add instance to placement group.	
Capacity Reservation	Open	Create new Capacity Reservation
IAM role	None	Create new IAM role
Shutdown behavior	Stop	
Enable termination protection	<input type="checkbox"/> Protect against accidental termination	
Monitoring	<input type="checkbox"/> Enable CloudWatch detailed monitoring Additional charges apply.	
Tenancy	Shared - Run a shared hardware instance Additional charges will apply for dedicated tenancy.	
T2 Unlimited	<input type="checkbox"/> Enable Additional charges may apply	

[Cancel](#)[Previous](#)[Review and Launch](#)[Next: Add Storage](#)

## Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/xvda	snap-00f00b3a3718745e9	8	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted
Add New Volume								

Add storage – EBS Elastic Block Storage volume will attached to your instance

[Cancel](#)[Previous](#)[Review and Launch](#)[Next: Add Tags](#)

Tags to identify the details about instance (Production/Test/Dev/Client Name)

[Cancel](#)[Previous](#)[Review and Launch](#)[Next: Configure Security Group](#)

## Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: ☒ Create a new security group

☐ Select an existing security group

Security group name:

Description:

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop

Using security group we can allow/deny any ports

Verify the details and click on Launch

## 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.

Create a new key pair

Key pair name



You have to download the **private key file** (\*.pem file) before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

For the first time you create a new key pair and Download Key Pair

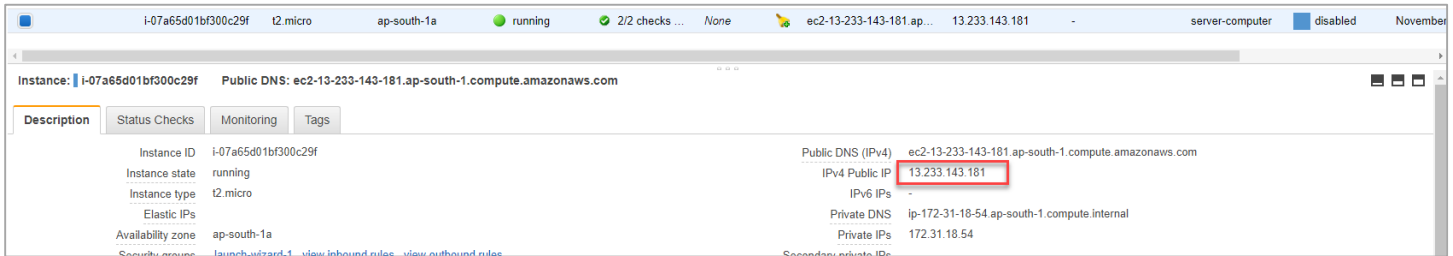
Server-computer.pem file will downloaded, **keep it safe**

## Launch Instances

Go to EC2 → See the instances

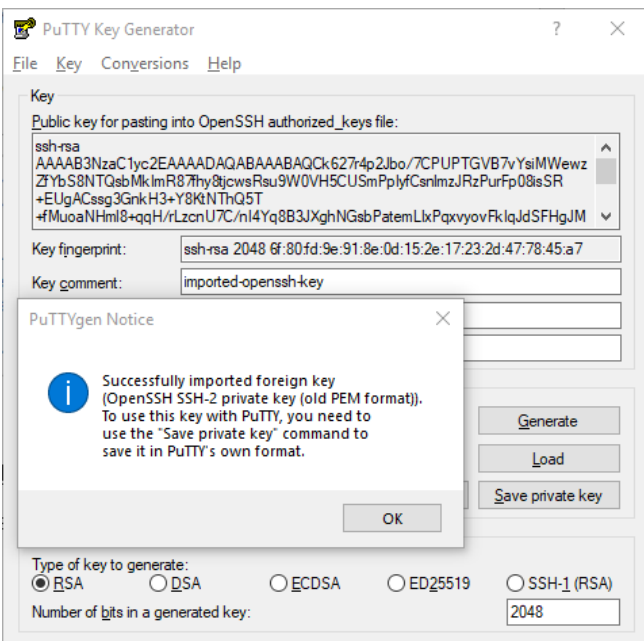
<input type="checkbox"/>	i-07a65d01bf300c29f	t2.micro	ap-south-1a	<span style="color: green;">●</span> running	Initializing	None	ec2-13-233-143-181.ap...	13.233.143.181	-	server-computer	disabled	November
--------------------------	---------------------	----------	-------------	--	--------------	------	--------------------------	----------------	---	-----------------	----------	----------

Click on instance and copy the Public IP Address



Install putty msi installer you will get PuttyGen and Putty for accessing Linux machine

Open puttyGen and load server-computer.pem file

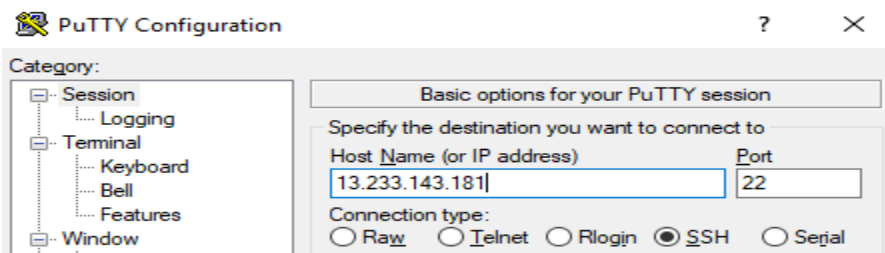


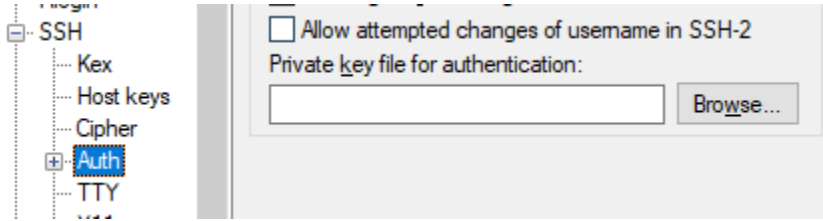
Click Ok.

## Save Private Key

In this case, I have used server-computer1.ppk

Open putty application and type IP address as shown below





Expand SSH → Click on Auth → Browse and attach .ppk file

Click on **Open**



You successfully logged into your Amazon Linux instance

As example, we are going to install web server in Linux server and access using web browser

```
sudo yum update
```

```
sudo yum install httpd
```

```
sudo service httpd start
```

```
sudo service httpd status
```

```
sudo chkconfig httpd on
```

Now go back to your EC2 → Security Groups and Add 80 port



Open browser and type your instance public IP address you can access web-server test page.

### 7. Create your First EC2 windows instance

Expand services EC2 → Launch Instance

 **Microsoft Windows Server 2016 Base** - ami-0711d827876cdd81a  
**Windows**  
**Free tier eligible**  
Microsoft Windows 2016 Datacenter edition. [English]  
Root device type: ebs    Virtualization type: hvm

**Select**  
64-bit (x86)

Select Windows Image

Choose an Instance Type → General Purpose (t2.micro) → Click **Next: Configure Instance Details** →

### Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access ma

**Number of instances** ⓘ

1

Launch into Auto Scaling Group ⓘ

**Purchasing option** ⓘ

☐ Request Spot instances

**Network** ⓘ

vpc-2c747344 (default) ▼

Create new VPC

**Subnet** ⓘ

subnet-750b241d | Default in us-east-2a ▼

Create new subnet

**Auto-assign Public IP** ⓘ

Enable ▼

**Placement group** ⓘ

☐ Add instance to placement group.

**Capacity Reservation** ⓘ

Open ▼

Create new Capacity Reservation

**Domain join directory** ⓘ

No directory

Create new directory

**IAM role** ⓘ

None ▼

Create new IAM role

Select VPC, subnet and enable Public IP address.

Click **Next: Add Storage**

Click **Next: Add Tags**

Add Tags to identify instance details Like Name, Purpose, Account and so and so

Click **Next: Configure Security Group**

### Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

**Assign a security group:** ☒ Create a new security group  
☐ Select an existing security group

**Security group name:**

WindowsSecurityGroup

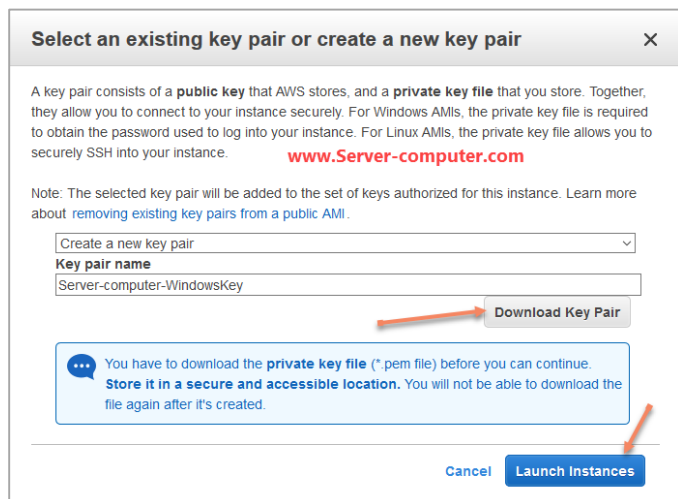
**Description:**

launch-wizard-1 created 2018-12-05T11:39:15.459+05:30

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ
RDP ▼	TCP	3389	Anywhere ▼ 0.0.0.0/0, ::/0

Add Rule

Click **Review and Launch**

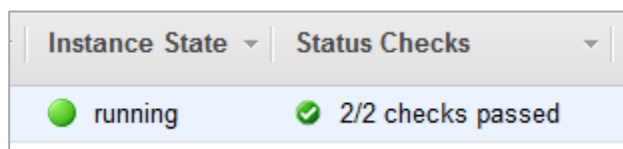


### Download Key Pair and Launch Instance

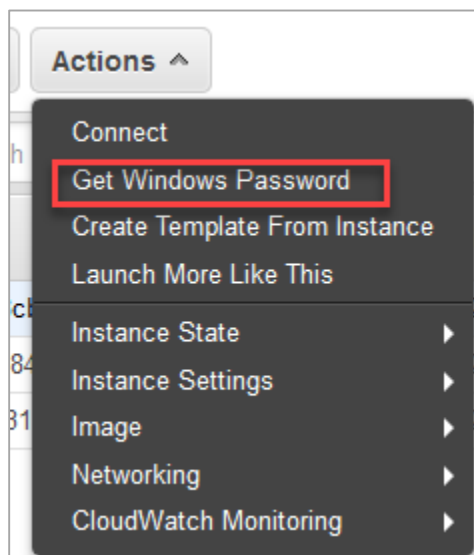
**Note:** Wait 4 Minutes instance to launch

It should display the following:

- Instance State: running
- Status Checks: 2/2 checks passed



Select instance you have launched → Actions



**Retrieve Default Windows Administrator Password** X

To access this instance remotely (e.g. Remote Desktop Connection), you will need your Windows Administrator password. A default password was created when the instance was launched and is available encrypted in the system log.

To decrypt your password, you will need your key pair for this instance. Browse to your key pair, or copy and paste the contents of your private key file into the text area below, then click Decrypt Password.

The following Key Pair was associated with this instance when it was created.

**Key Name** Server-computer-WindowsKey

In order to retrieve your password you will need to specify the path of this Key Pair on your local machine:

**Key Pair Path**  Server-computer-WindowsKey.pem

Or you can copy and paste the contents of the Key Pair below:

```
-----BEGIN RSA PRIVATE KEY-----
MIEowIBAAKCAQEAjurQSFEoBRrSHhUvr+F0f7fVfALgE8mtzVuq6y0XX0WHgROHwMhz34mRNAdH
m4C+pxbyttgKkIipq6ygh/WUKraipkE+XuZ9Ts34b8MxZNYeH/EHjwIDAQABAoIBAGJwLht+haZ
-----
```

Browse server-computer-WindowsKey.pem file to decrypt and get password

**Retrieve Default Windows Administrator Password** X

☒ **Password Decryption Successful**  
The password for instance i-0b43007700d8cbfe4 was successfully decrypted.

☐ **Password change recommended**  
We recommend that you change your default password. Note: If a default password is changed, it cannot be retrieved through this tool. It's important that you change your password to one that you will remember.

You can connect remotely using this information:

**Public DNS** ec2-3-16-76-250.us-east-2.compute.amazonaws.com

**User name** Administrator

**Password** [REDACTED]

Now you got password successfully. Click **Close**.

Go to your windows machine Start → Run → mstsc → Ok

Remote Desktop Connection

**Remote Desktop Connection**

Computer: ec2-3-16-76-250.us-east-2.compute.amazonaws.com

User name: None specified

You will be asked for credentials when you connect.

Click **connect** and type user name and password you are connected to your EC2 windows instance.

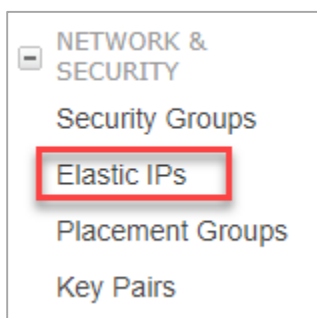
### 8. Assigning Elastic IP Addresses to Instance (Static IP Address)

Click on instance name and see instance details like Internal and external IP Address, Host name

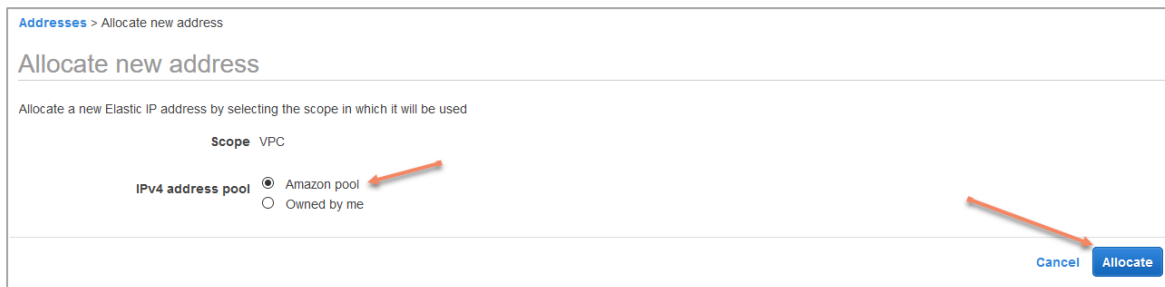
Public DNS (IPv4)	ec2-13-127-65-71.ap-south-1.compute.amazonaws.com
IPv4 Public IP	13.127.65.71
IPv6 IPs	-
Private DNS	ip-172-31-25-150.ap-south-1.compute.internal
Private IPs	172.31.25.150

However, after stop and start of instance assigned public IP address will release to the amazon free pool

If would like to assign an static public address then navigate to Elastic IP's

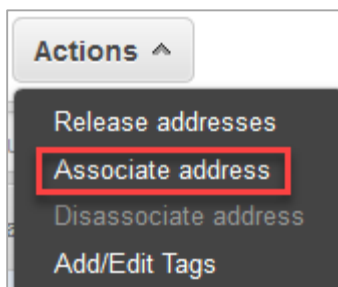


EC2 console right side bar go down → Elastic IPs → Allocate New Address

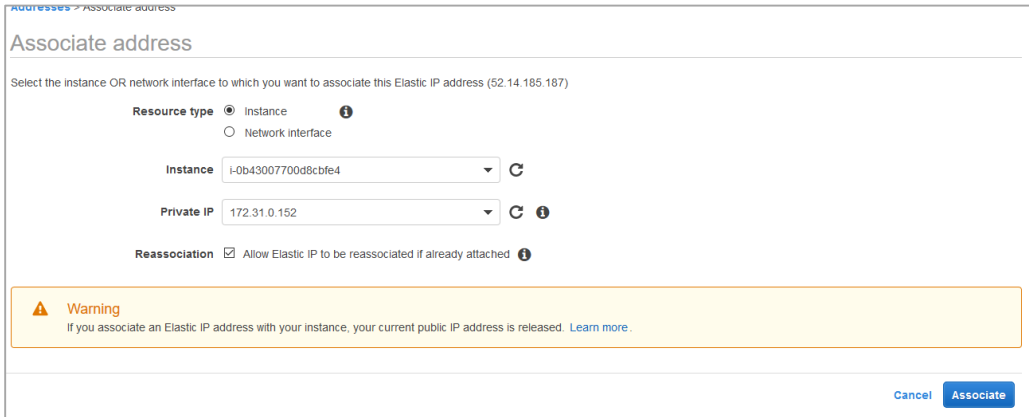


Click **Allocate**. Amazon allocate you static IP address

Select the IP from Elastic IPs console → Actions → Associate Address







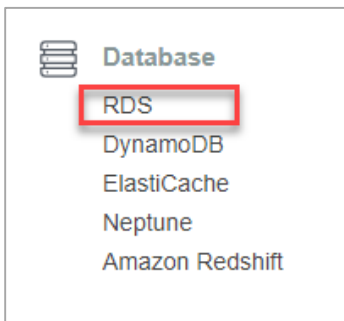
Select Instance ID check Instance ID before allocating. Click **Associate**

**Note:** If you have, multiple interfaces to the instance click on Radio button **Network Interface** and select correct NIC card name and Local IP Address.

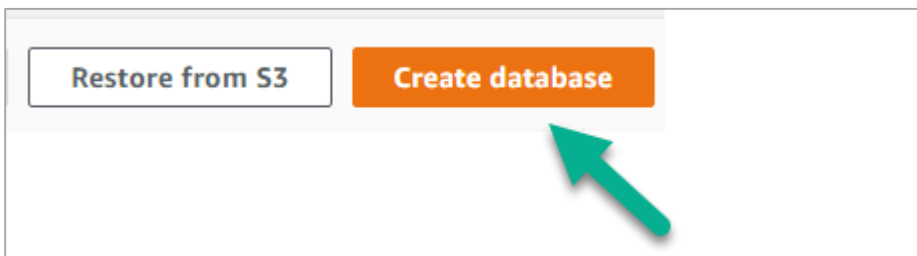
Now your existing instance has static Public IP address, if you restart your instance also you will get same IP address until you detach from instance.

## 9. Launching RDS Instance

Login to AWS Console and Click on services to list all services. Navigate to **Database → RDS**





Now we are going to create a new Database instance with empty database





Amazon will support below 5 types of Relational database engines as managed services


**Engine options**


☐ Amazon Aurora  


☒ MySQL  


☐ MariaDB  


☐ PostgreSQL  


☐ Oracle  


☐ Microsoft SQL Server  


Select any one of the database engine, which you want to launch and Click **Next**

**Note:** Careful if you are using free tier account. MSSQL and Oracle are charged.

**Choose use case**

**Use case**  
Do you plan to use this database for production purposes?  
[www.server-computer.com](https://www.server-computer.com)

Use case

☐ **Production - Amazon Aurora** Recommended  
MySQL-compatible, enterprise-class database at 1/10th the cost of commercial databases.

☐ **Production - MySQL**  
Use [Multi-AZ Deployment](#) and [Provisioned IOPS Storage](#) as defaults for high availability and fast, consistent performance.

☒ **Dev/Test - MySQL**  
This instance is intended for use outside of production or under the [RDS Free Usage Tier](#).

Billing is based on [RDS pricing](#).

Cancel

Previous

Next

Choose appropriate usage of your instance. In this scenario, I am using Dev/Test instance Click **Next**

## Specify DB details

[www.server-computer.com](https://www.server-computer.com)

### Instance specifications

Estimate your monthly costs for the DB Instance using the [AWS Simple Monthly Calculator](#)

DB engine

MySQL Community Edition

License model [Info](#)

general-public-license

Select Version

DB engine version [Info](#)

MySQL 5.6.40

In drop down, select appropriate and required MySQL Version.

**Note:** If you select Free Tier. Selected version and options will overwritten free options.

DB instance class [Info](#)

db.r4.xlarge — 4 vCPU, 30.5 GiB RAM

Multi-AZ deployment [Info](#)

☐ Create replica in different zone

Creates a replica in a different Availability Zone (AZ) to provide data redundancy, eliminate I/O freezes, and minimize latency spikes during system backups.

☒ No

Storage type [Info](#)

General Purpose (SSD)

Allocated storage

20

GiB

(Minimum: 20 GiB, Maximum: 32768 GiB) Higher allocated storage may improve IOPS performance.

**i** Provisioning less than 100 GiB of General Purpose (SSD) storage for high throughput workloads could result in higher latencies upon exhaustion of the initial General Purpose (SSD) IO credit balance. [Click here](#) for more details.

1. Select DB Instance class like required CPU Cores and RAM.
2. Create Replica in Different Zone. (Which means database will be replicated to another available zone for redundant(data protection))
3. General purpose (SSD) or provisioned IOPS (SSD)
  - a. General purpose is for low through put applications

- b. Provisioned IOPS is for most read/write operations
- 4. Size of the storage

### Settings

**DB instance identifier** [Info](#)  
Specify a name that is unique for all DB instances owned by your AWS account in the current region.

DB instance identifier is case insensitive, but stored as all lower-case, as in "mydbinstance". Must contain from 1 to 63 alphanumeric characters or hyphens (1 to 15 for SQL Server). First character must be a letter. Cannot end with a hyphen or contain two consecutive hyphens.

**Master username** [Info](#)  
Specify an alphanumeric string that defines the login ID for the master user.

Master Username must start with a letter. Must contain 1 to 16 alphanumeric characters.

**Master password** [Info](#) **Confirm password** [Info](#)

Master Password must be at least eight characters long, as in "mypassword". Can be any printable ASCII character except "/", "", or "@".

[Cancel](#) [Previous](#) [Next](#)

Provide

- Instance name should be unique
- Master username anything you can give without special characters
- Provide master password and remember

**Free tier**

The Amazon RDS Free Tier provides a single db.t2.micro instance as well as up to 20 GiB of storage, allowing new AWS customers to gain hands-on experience with Amazon RDS. Learn more about the RDS Free Tier and the instance restrictions [here](#).

☒ Only enable options eligible for RDS Free Usage Tier [Info](#)

**DO NOT FORGOT TO SELECT IF YOU'RE USING FREE TIER OTHERWISE YOU WILL BE CHARGED**

**Network & Security**

**Virtual Private Cloud (VPC)** [Info](#)  
VPC defines the virtual networking environment for this DB instance.

Default VPC (vpc-cbd4f2a3)

Only VPCs with a corresponding DB subnet group are listed.

**Subnet group** [Info](#)  
DB subnet group that defines which subnets and IP ranges the DB instance can use in the VPC you selected.

default

**Public accessibility** [Info](#)

☒ **Yes**  
EC2 instances and devices outside of the VPC hosting the DB instance will connect to the DB instances. You must also select one or more VPC security groups that specify which EC2 instances and devices can connect to the DB instance.

☐ **No**  
DB instance will not have a public IP address assigned. No EC2 instance or devices outside of the VPC will be able to connect.

**Availability zone** [Info](#)

ap-south-1a

**VPC security groups**  
Security groups have rules authorizing connections from all the EC2 instances and devices that need to access the DB instance.

☒ **Create new VPC security group**

☐ **Choose existing VPC security groups**

Select appropriate VPC and Subnet group (If any)

If you want access database from remote machine put “Public Accessibility” **Yes**

Choose existing VPC security groups if you have already or it will create new security group for this instance access.

### Database options

Database name [Info](#)

Note: if no database name is specified then no initial MySQL database will be created on the DB Instance.

Port [Info](#)

TCP/IP port the DB instance will use for application connections.

DB parameter group [Info](#)

Option group [Info](#)

IAM DB authentication [Info](#)

☐ Enable IAM DB authentication  
Manage your database user credentials through AWS IAM users and roles.

☒ Disable

### Encryption

Encryption

☒ Enable encryption [Learn more](#) [↗](#)  
Select to encrypt the given instance. Master key ids and aliases appear in the list after they have been created using the Key Management Service(KMS) console.

☐ Disable encryption


**i** The selected engine or DB instance class does not support storage encryption.

Provide database name, default port number is 3306 you can even customize the port number if you want.

Enabling IAM DB Authentication. IAM Users also can access your instance based on IAM policies.

For free tier encryption option is disabled

### Backup

 Please note that automated backups are currently supported for InnoDB storage engine only. If you are using MyISAM, refer to detail [here](#). [↗](#)

Backup retention period [Info](#)  
Select the number of days that Amazon RDS should retain automatic backups of this DB instance.

7 days ▼

Backup window [Info](#)

☐ Select window

☒ No preference

☒ Copy tags to snapshots

If you want database backups select, the retention max is **35 Days**

If you have particular backup window for database select it otherwise leave it default.

### Monitoring

Enhanced monitoring

☐ Enable enhanced monitoring  
Enhanced monitoring metrics are useful when you want to see how different processes or threads use the CPU.

☒ Disable enhanced monitoring

Enhanced monitoring will charged

## Log exports



Select the log types to publish to Amazon CloudWatch Logs

- ☐ Audit log
- ☐ Error log
- ☐ General log
- ☐ Slow query log

### IAM role

The following service-linked role is used for publishing logs to CloudWatch Logs.

RDS Service Linked Role

 Ensure that General, Slow Query, and Audit Logs are turned on. Error logs are enabled by default.  
[Learn more](#) 

## Maintenance

Auto minor version upgrade [Info](#)

- ☒ **Enable auto minor version upgrade**  
Enables automatic upgrades to new minor versions as they are released. The automatic upgrades occur during the maintenance window for the DB instance.
- ☐ Disable auto minor version upgrade

Maintenance window [Info](#)

Select the period in which you want pending modifications or patches applied to the DB instance by Amazon RDS.

- ☐ Select window
- ☒ No preference

Select the options you required



### Deletion protection

☐ **Enable deletion protection**  
Protects the database from being deleted accidentally. While this option is enabled, you can't delete the database.

Cancel

Previous

Create database

Enabling database protection, you cannot delete database

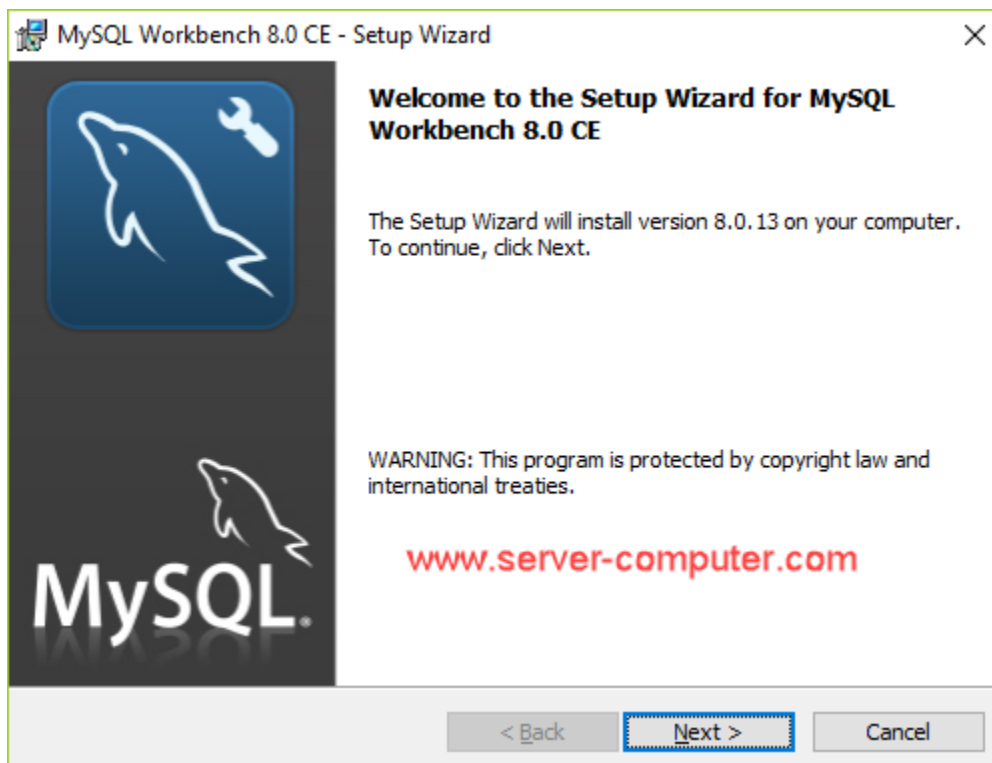
Click **Create Database**

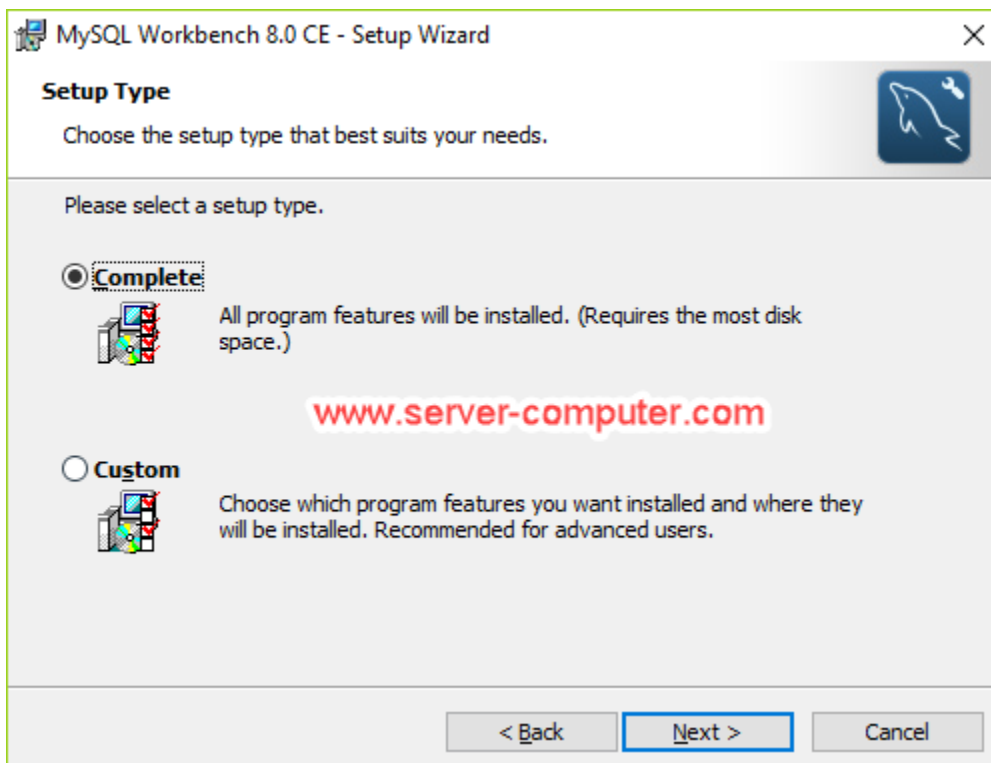
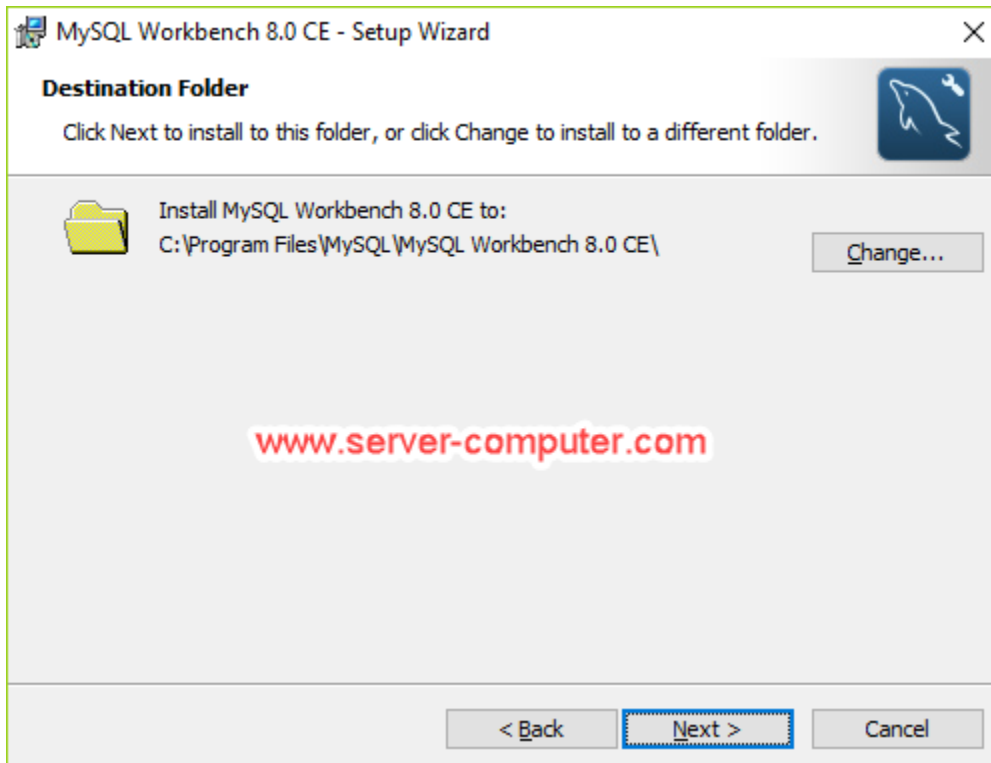
**Note:** Database instance creation will take at least 10minutes.

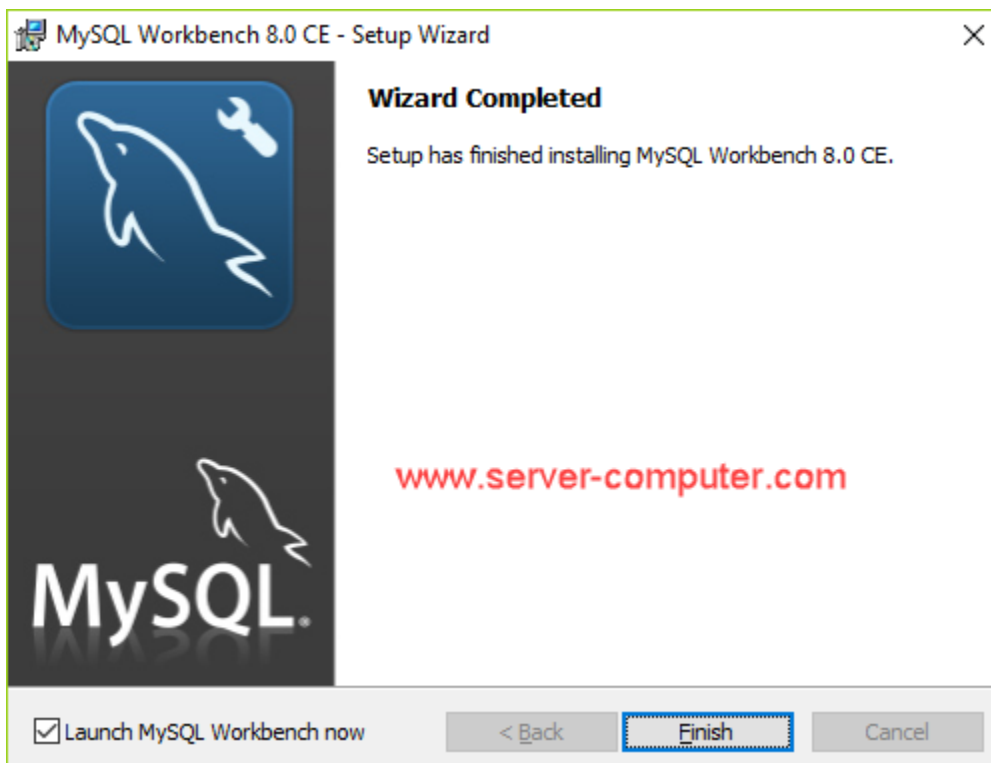
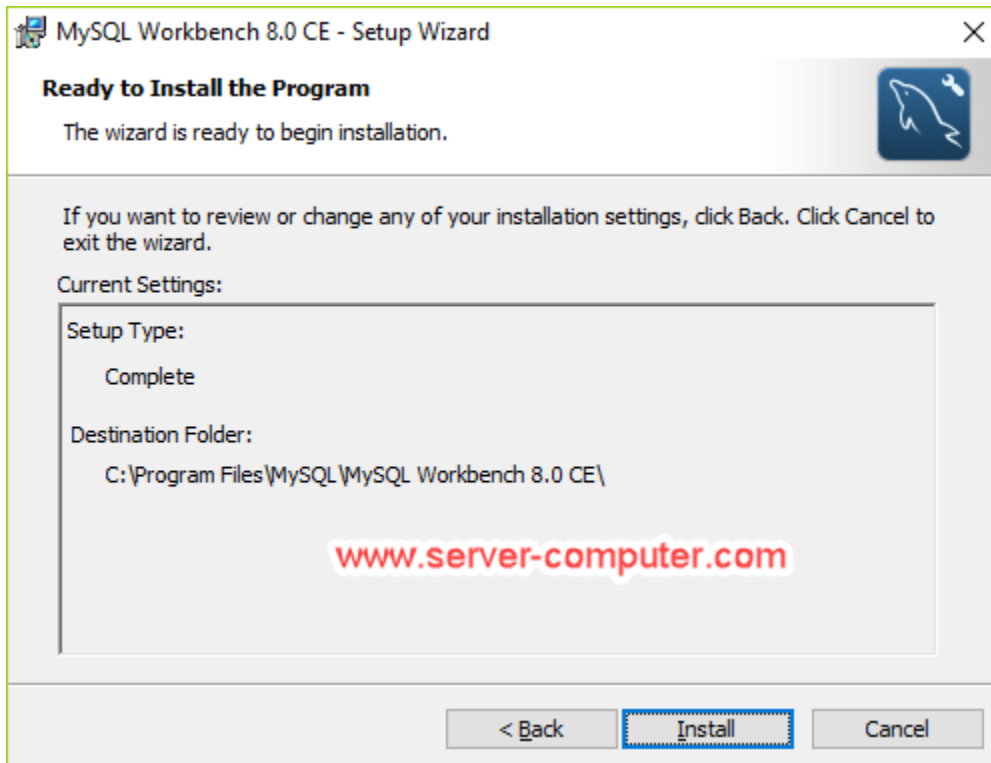
## 10.Accessing MySQL Instance Using Workbench

Download MySQL Workbench to access MySQL instance remotely

<https://dev.mysql.com/downloads/workbench/>







After successful creation you see like below

DB instance	Engine	Status	CPU	Current activity	Maintenance	Class	VPC	Multi-AZ	Replicat
techarkitdatabase	MySQL	available	1.00%	0 Connections	none	db.t2.micro	vpc-cbd4f2a3	No	

Click on Database name and come down copy the **Endpoint URL**

Open your MySQL workbench and create connection



Click on Plus (+) sign to create a New MySQL Connection

Connection Name:  Type a name for the connection

Connection Method:  Method to use to connect to the RDBMS

Parameters SSL Advanced

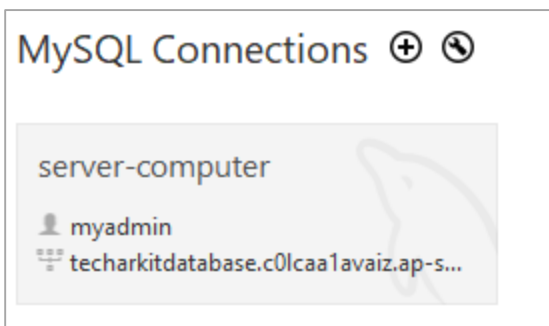
Hostname:  Port:  Name or IP address of the server host - and TCP/IP port.

Username:  Name of the user to connect with.

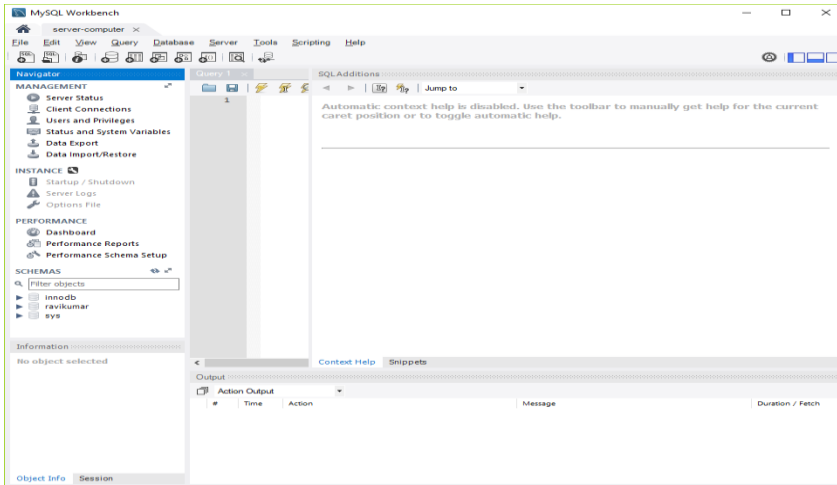
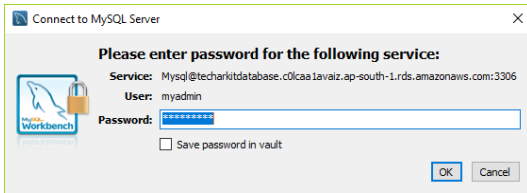
Password:   The user's password. Will be requested later if it's not set.

Default Schema:  The schema to use as default schema. Leave blank to select it later.

Click **OK**



After successful creation, Click on Connection it will ask you for the password



Successfully launched MySQL RDS Instance and accessed via MySQL Work bench.

Run below queries to create database and some tables on it.

```
create database 'DBNAME';  
use DBNAME;
```

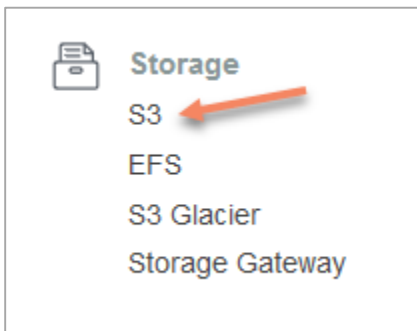
### **Create Table using below query**

```
create table students(  
    student_id INT NOT NULL AUTO_INCREMENT,  
    student_title VARCHAR(100) NOT NULL,  
    student_author VARCHAR(40) NOT NULL,  
    submission_date DATE,  
    PRIMARY KEY ( student_id )  
);  
show databases;  
use DBNAME;  
show tables;
```

If you know much more database queries like select, insert and delete statement try doing more. Good Luck.

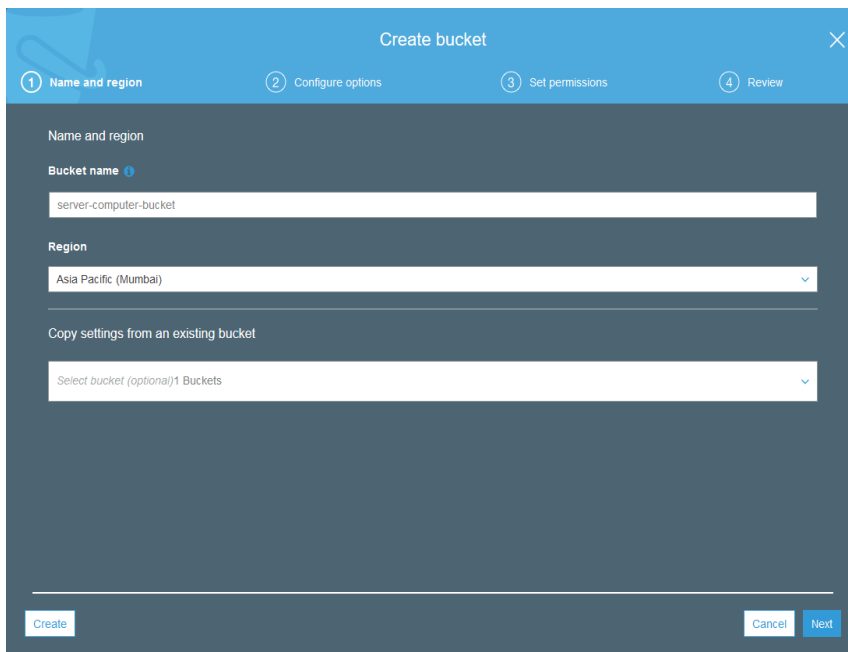
### 11. AWS S3 Bucket

Login to AWS Console and navigate to storage → S3



**+ Create bucket**

Click on

A screenshot of the AWS "Create bucket" wizard. The wizard has a blue header with the title "Create bucket" and a close button. Below the header is a progress bar with four steps: 1. Name and region, 2. Configure options, 3. Set permissions, and 4. Review. The first step, "Name and region", is active. It contains three sections: "Name and region" with a "Bucket name" field containing "server-computer-bucket", a "Region" dropdown menu set to "Asia Pacific (Mumbai)", and a "Copy settings from an existing bucket" section with a dropdown menu set to "Select bucket (optional) 1 Buckets". At the bottom of the form are three buttons: "Create", "Cancel", and "Next".

Provide bucket name, it should be a unique name. To Access your S3 bucket over internet it will create DNS entry.

Click **Next**

The screenshot shows the 'Create bucket' wizard in the AWS Management Console, specifically the 'Configure options' step. The wizard has four steps: 1. Name and region, 2. Configure options, 3. Set permissions, and 4. Review. Under 'Properties', there are several sections: 'Versioning' with a checkbox 'Keep all versions of an object in the same bucket'; 'Server access logging' with a checkbox 'Log requests for access to your bucket'; 'Tags' with a section for adding tags; 'Object-level logging' with a checkbox 'Record object-level API activity using AWS CloudTrail'; 'Default encryption' with a checkbox 'Automatically encrypt objects when they are stored in S3'; and 'Advanced settings' which includes 'Object lock' with a checkbox 'Permanently allow objects in this bucket to be locked'. At the bottom right, there are 'Previous' and 'Next' buttons.

- ✚ **Keep All Version of object** means it will not delete any files if you upload same file multiple times. It will keep all the files as multiple versions
- ✚ **Log Requests for access to your bucket** option will log all the actions users did on this particular S3 bucket
- ✚ **Object-level Logging** used to monitor all the object level modifications. Additional cost.
- ✚ **Encryption** You can encrypt S3 bucket data or Encrypt and upload the data either way your data is encrypted.
- ✚ **Object Lock**
- ✚ **Cloudwatch request metrics** for monitoring purpose

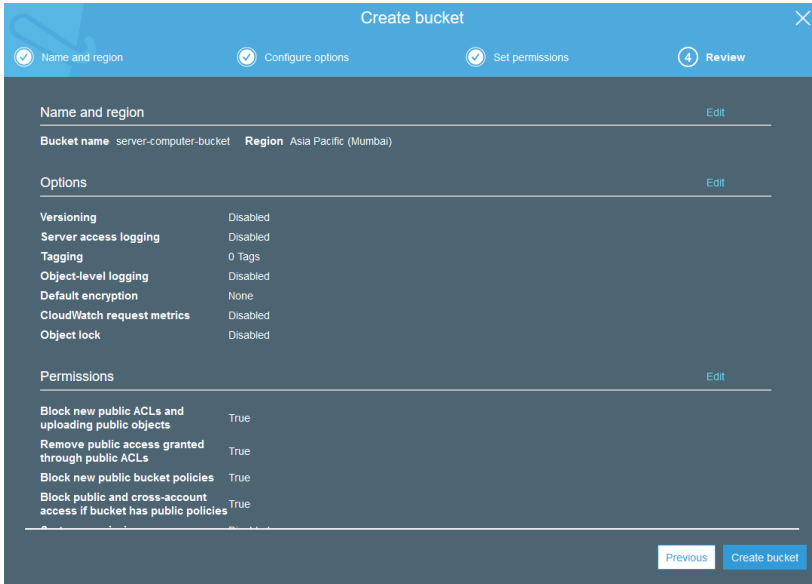
Click **Next**

The screenshot shows the 'Set permissions' step of the 'Create bucket' wizard. It includes a note about granting access to specific users. The 'Public access settings for this bucket' section explains the Amazon S3 block public access settings. There are two main sections for managing public access: 'Manage public access control lists (ACLs) for this bucket' and 'Manage public bucket policies for this bucket'. Both sections have checkboxes for 'Block new public ACLs and uploading public objects' and 'Remove public access granted through public ACLs'. The 'Manage system permissions' section has a dropdown menu for 'Do not grant Amazon S3 Log Delivery group write access to this bucket'. At the bottom right, there are 'Previous' and 'Next' buttons.

AWS recent update is to block public access by default, if you want to enable public access to your S3 bucket un-check all above tick marks.

Still you can provide access to other users on bucket level and object level.

Click **Next**



Final Step is to review selected options and Click **Create bucket**

Your S3 bucket created successfully. Click bucket name you will see all the options

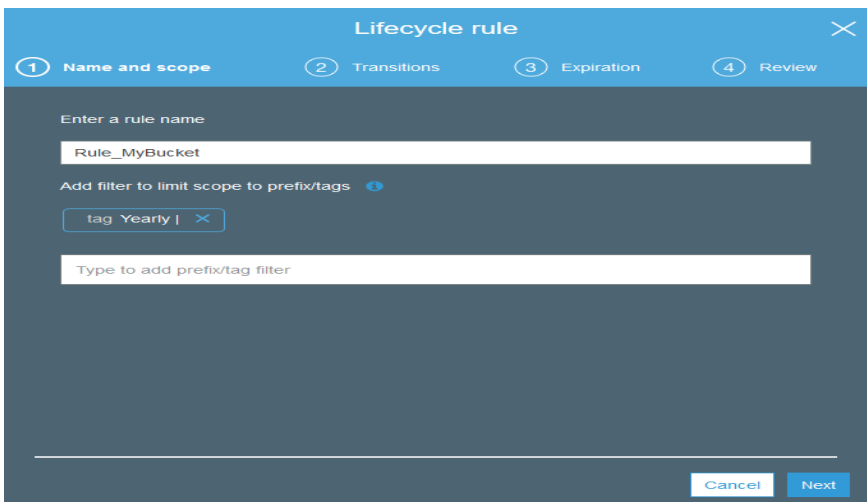
<https://s3.ap-south-1.amazonaws.com/server-computer-bucket>

Above is the example URL to access your S3 bucket over internet

### 11.1. AWS S3 Lifecycle Management

Click on **S3 Bucket → Management → Lifecycle**

You can manage an objects lifecycle using this feature/rule, which defines

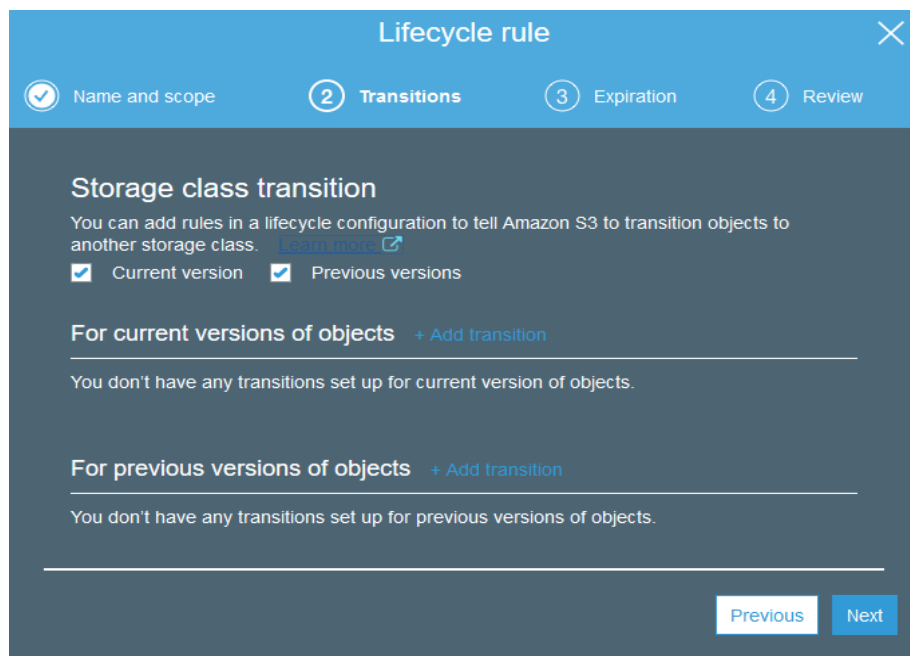


Enter Rule Name



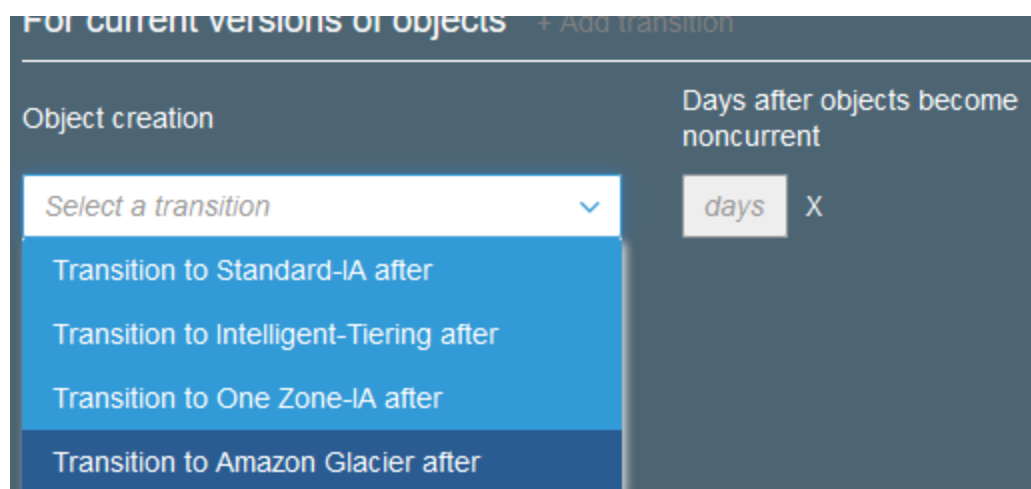
Tag Name if you do not want leave it blank

Click **Next**



- ☒ Current Versions
- ☒ Previous Versions

Based on selected versions action will be performed example if you want to keep current versions in A1 or maybe previous versions on Glacier as per your requirement



Click **Next**

The screenshot shows the 'Lifecycle rule' configuration window with a blue header and a dark grey body. The header contains four steps: 'Name and scope' (checked), 'Transitions' (checked), 'Expiration' (selected with a circled 3), and 'Review' (4). The main content area is titled 'Configure expiration' and shows the bucket name 'server-computer.com'. Under 'Configure expiration', there are two sections. The first section has a checkbox for 'Current version' (unchecked) and 'Previous versions' (checked). Below this, there is a checkbox for 'Permanently delete previous versions' (checked) with an information icon. A text field shows 'After 365 days from becoming a previous version'. The second section is titled 'Clean up expired object delete markers and incomplete multipart uploads'. It has two checkboxes: 'Clean up expired object delete markers' (checked) and 'Clean up incomplete multipart uploads' (checked), both with information icons. Below the second checkbox, a text field shows 'After 2 days from start of upload'. At the bottom right, there are 'Previous' and 'Next' buttons.

Explanation: Previous versions of files after 365 days means one year permanently delete from S3 bucket.

Clean up expired and incomplete uploads after 2 days.

Click **Next**

The screenshot shows the 'Lifecycle rule' configuration window in the AWS console, specifically the 'Review' step (Step 4 of 4). The previous steps are 'Name and scope', 'Transitions', and 'Expiration', all marked with checkmarks. The 'Name and scope' section shows the rule name as 'Rule\_MyBucket' and the scope as 'Whole bucket'. The 'Transitions' and 'Expiration' sections are currently empty. The 'Expiration' section shows 'Permanently delete after 365 days', 'Clean up expired object delete markers', and 'Clean up incomplete multipart uploads after 2 days'. There are 'Edit' links for each section. The window has a blue header and a close button in the top right corner.

Click **Save**.

## 11.2. S3 Bucket Replication to Cross-Region

S3 bucket **Name** → **Management** → **Replication**

**Note:** In order to enable Replication for S3 bucket **Versioning** should be enabled.

The screenshot shows the 'Replication rule' configuration window in the AWS console, specifically the 'Set source' step (Step 1 of 4). The subsequent steps are 'Set destination', 'Configure options', and 'Review'. Under 'Set source', the 'Entire bucket' option is selected, and the bucket name 'arkit-test123' is shown. The 'Prefix or tags' option is also available. Under 'Replication criteria', the 'Replicate objects encrypted with AWS KMS' checkbox is checked. A blue information box at the bottom states: 'Your CRR rule will be created using the new schema. Cross-region replication (CRR) now has a new schema that supports replication based on prefixes, one or more object tags or a combination of the two. As part of the new schema, you can set overlapping rules with priorities. The new schema does not support delete marker replication, which would prevent any delete actions from replicating. Learn more about cross-region replication.' At the bottom right, there are 'Cancel' and 'Next' buttons.

Click **Next**

The screenshot shows the 'Replication rule' configuration window in the AWS IAM console. The progress bar at the top indicates four steps: 1. Set source (completed), 2. Set destination (current step), 3. Configure options, and 4. Review. Under the 'Destination bucket' section, a dropdown menu is set to 'destinationserver1'. The 'Options' section has a checked checkbox for 'Change the storage class for the replicated object(s)'. Below this, the 'Storage class' dropdown is set to 'Glacier'. There is a link to 'Learn more' and a link to 'Amazon S3 pricing'. At the bottom of the options section, there is an unchecked checkbox for 'Change object ownership to destination bucket owner'. At the bottom right of the window are 'Previous' and 'Next' buttons.

Select Destination bucket within same account or another account

Options to Change Storage class and permissions in destination

Click **Next**

The screenshot shows the 'Replication rule' configuration window in the AWS IAM console, now at step 3: Configure options. The progress bar shows steps 1 and 2 as completed. The 'IAM role' section has a dropdown menu set to 'Create new role'. The 'Rule name' section has a text input field containing 'Test'. The 'Status' section has two radio buttons: 'Enabled' (selected) and 'Disabled'. At the bottom right of the window are 'Previous' and 'Next' buttons.

Select existing IAM Role or Create new for replication. In this case, I am creating new role for replication called Test

Click **Next**

Review final and Click **Save**

### 11.3. S3 Bucket Policies to control Access

Click on bucket Name → Permissions → bucket policy

<https://awspolicygen.s3.amazonaws.com/policygen.html>

Go to this above URL and generate policy if you do not know how to write a S3 bucket policy

**Step 1: Select Policy Type**  
A Policy is a container for permissions. The different types of policies you can create are an IAM Policy, an S3 Bucket Policy, an SNS Topic Policy, a VPC Endpoint Policy, and an SQS Queue Policy.

Select Type of Policy S3 Bucket Policy

**Step 2: Add Statement(s)**  
A statement is the formal description of a single permission. See a [description of elements](#) that you can use in statements.

Effect ☒ Allow ☐ Deny

Principal Test  
Use a comma to separate multiple values.

AWS Service Amazon S3 ☐ All Services ("\*")  
Use multiple statements to add permissions for more than one service.

Actions 1 Action(s) Selected ☐ All Actions ("\*")

Amazon Resource Name (ARN) arn:aws:s3:::arkit  
ARN should follow the following format: arn:aws:s3:::<bucket\_name>/<key\_name>.  
Use a comma to separate multiple values.

[Add Conditions \(Optional\)](#)

Add Statement

**Add Statement** and click on **Generate Policy**

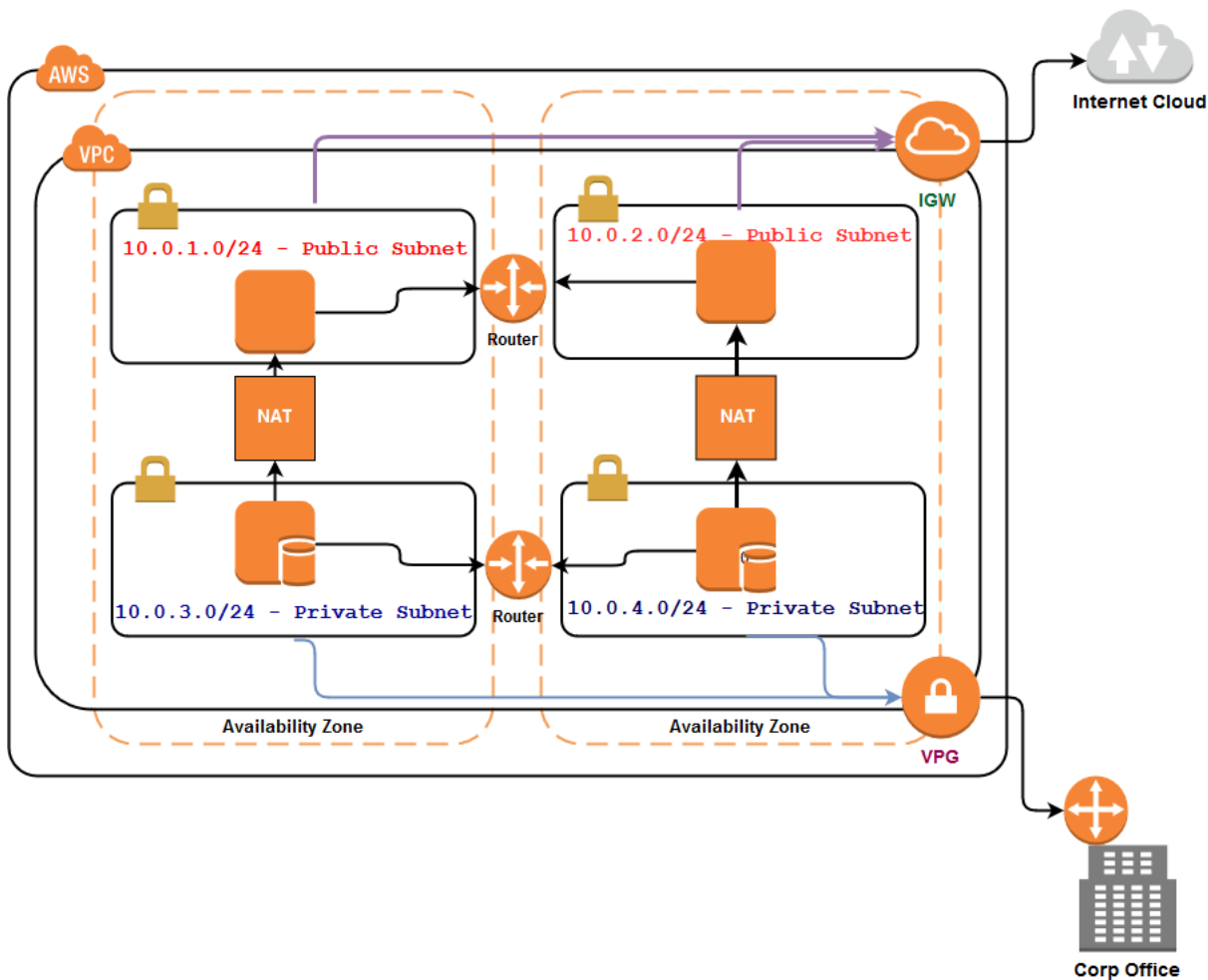
```
{
  "Id": "Policy1543401188367",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1543401184049",
      "Action": [
        "s3:ListBucket",
        "s3:ListBucketByTags",
        "s3:ListBucketVersions"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::arkit-prog",
      "Principal": {
        "AWS": [
          "test"
        ]
      }
    }
  ]
}
```

```
}  
}  
]  
}
```



Same policy copy and paste it in policy editor and **save**






## 12. VPC – Virtual Private Cloud (isolated Network)

A **virtual private cloud** (VPC) is a virtual network dedicated to your AWS account. It is logically isolated from other virtual networks in the AWS Cloud. You can launch your AWS resources, such as Amazon EC2 instances, into your VPC.



Picture: 1.1 Typical VPC Example

-  EC2 Instance
-  Virtual Private Gateway

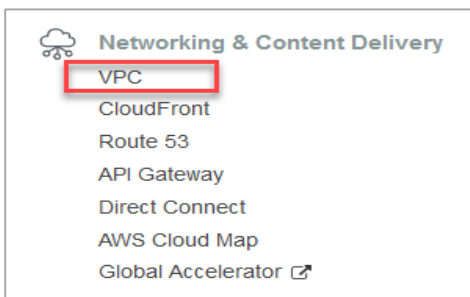
-  Router
-  Customer Gateway
-  Internet Gateway
-  Availability Zone
-  VPC subnet

### Architecture Explanation:

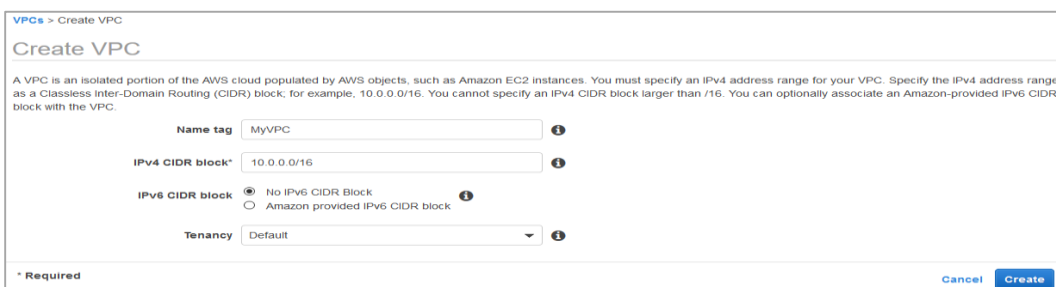
- AWS in single region
- Two Availability zones
- One Virtual Private Cloud
- Four Subnets Two Are Public and Two Are Private subnets
- Four instances Two App Servers, Two Database Servers
- One Internet Gateway to access internet
- One Virtual Private Gateway to Connect Corporate Office
- Two routers one is connected to private subnets, another is connected to public subnets

We would like to host web application with two web app servers and two Database servers. Two Tier architecture. Web app servers will serve to public, from public facing subnets. Database servers are in private network and only have access to app servers and corporate network (VPG).

When Database servers want to download any kind of files/patches from internet it routes through NAT Gateway and get the internet data from web app servers.



AWS Console → Services → Networking & Content Delivery → VPC → Your VPCs

A screenshot of the 'Create VPC' page in the AWS console. The page title is 'Create VPC'. Below the title, there is a descriptive paragraph about VPCs. The form contains the following fields: 'Name tag' with the value 'MyVPC', 'IPv4 CIDR block' with the value '10.0.0.0/16', 'IPv6 CIDR block' with the radio button selected for 'No IPv6 CIDR Block', and 'Tenancy' with the value 'Default'. At the bottom right, there are 'Cancel' and 'Create' buttons. A small asterisk and the word 'Required' are at the bottom left.

- VPC Name: MyVPC

- **IPv4 CIDR Block:** 10.0.0.0/16 ( Use this [CIDR Calculator](#) )

Click **Create**

Result	
CIDR Range	10.0.0.0/16
Netmask	255.255.0.0
Wildcard Bits	0.0.255.255
First IP	10.0.0.0
Last IP	10.0.255.255
Total Host	65536
CIDR	
10.0.0.0/16	

Create VPC

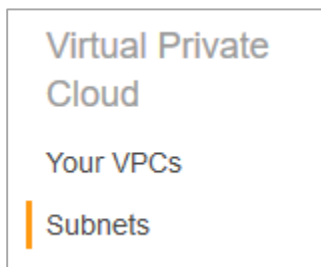
 The following VPC was created:  
VPC ID `vpc-02c316e5f1be2208a`

Close

Your VPC created successfully.

## 12.1. Create subnets

Inside VPC to divide smaller blocks and separation



Subnets > Create subnet

Create subnet

Specify your subnet's IP address block in CIDR format, for example, 10.0.0.0/24. IPv4 block sizes must be between a /16 netmask and /28 netmask, and can be the same size as your VPC. An IPv6 CIDR block must be a /64 CIDR block.

Name tag

VPC\*

VPC CIDRs	Status	Status Reason
CIDR 10.0.0.0/16	associated	

Availability Zone

IPv4 CIDR block\*

\* Required

Cancel Create



Create subnet

The following Subnet was created:

Subnet ID    subnet-01b0a1e5be742dde0

Close

In Similar way, create all four subnets

Subnet Name	Availability Zone	CIDR Block	Private/Public
S1-Private	Us-east-2a	10.0.1.0/24	Private
S2-Private	Us-east-2b	10.0.2.0/24	Private
S3-Public	Us-east-2a	10.0.3.0/24	Public
S4-Public	Us-east-2b	10.0.4.0/24	Public

<input type="checkbox"/>	Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR	Availability Zone	Availability Zone ID
<input type="checkbox"/>	S1-Private	subnet-01b0a1e5be742dde0	available	vpc-02c316e5f1be2208a   MyVPC	10.0.1.0/24	251	-	us-east-2a	use2-az1
<input type="checkbox"/>	S2-Private	subnet-0415e767640ae4ef9	available	vpc-02c316e5f1be2208a   MyVPC	10.0.2.0/24	251	-	us-east-2b	use2-az2
<input type="checkbox"/>	S3-Public	subnet-01f8724bb68578a99	available	vpc-02c316e5f1be2208a   MyVPC	10.0.3.0/24	251	-	us-east-2a	use2-az1
<input type="checkbox"/>	S4-Public	subnet-09d6c82e020a61325	available	vpc-02c316e5f1be2208a   MyVPC	10.0.4.0/24	251	-	us-east-2b	use2-az2

## 12.2. Create Internet gateway and attach to VPC

Internet Gateways. An internet gateway is a horizontally scaled, redundant, and highly available VPC component that allows communication between instances in your VPC and the internet. It therefore imposes no availability risks or bandwidth constraints on your network traffic.

Attach to S3 and S4, after attach S3 and S4 become public subnets.

Internet gateways > Create internet gateway

www.server-computer.com

Create internet gateway

An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.

Name tag    My-IGW

\* Required

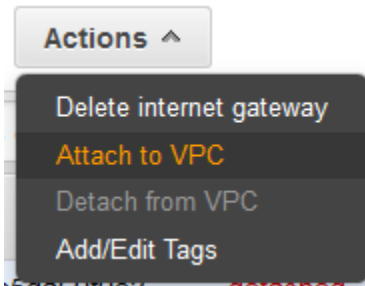
Cancel    Create

### Create internet gateway

The following internet gateway was created:

Internet gateway ID    igw-0b5da69f9e34ec455

Close



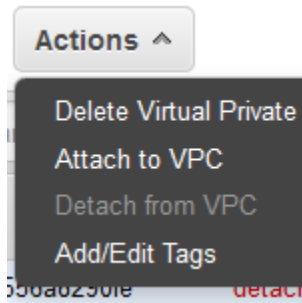
Now attach Internet Gateway to VPC

Select MyVPC in drop down menu Click **Attach**

### 12.3. Create Virtual Private Gateway and Attach to VPC

It can be a physical or software appliance. The anchor on the AWS side of the VPN connection is called a virtual private gateway. The following diagram shows your network, the customer gateway, the VPN connection that goes to the virtual private gateway, and the VPC.

#### Create Virtual Private Gateway



## Attach VGW to MyVPC

### Attach to VPC

Select the VPC to attach to the virtual private gateway.

Virtual Private Gateway Id vgw-0649463556a8290fe

VPC\*

[Cancel](#) [Yes, Attach](#)

## 12.4. Create route tables and attach to subnets

Route Tables. A route table contains a set of rules, called routes that are used to determine where network traffic is directed. Each subnet in your VPC must be associated with a route table; the table controls the routing for the subnet.

One route for Internet gateway, another for Virtual private gateway (R1-IGW and R2-VGW)

- Route - 0.0.0.0/0 to IGW
- Route - 192.168.0.0/16 to VGW

### Create route table

### Create route table

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

Name tag

VPC\*

\* Required

[Cancel](#) [Create](#)

### Create route table

✓ The following Route Table was created:

Route Table ID [rtb-08aa6cb351595eac2](#)

[Close](#)

Name tag

VPC\*

Now edit R1-IGW and add routing rule as mentioned below

Route Tables > Edit routes

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	active	No
<input type="text" value="0.0.0.0/0"/>	<input type="text" value="igw-0b5da69f9e34ec455"/>		No

Add route

\* Required

Cancel Save routes

Route Tables > Edit routes

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	active	No
<input type="text" value="192.168.0.0/16"/>	<input type="text" value="vgw-0649463556a8290fe"/>		No

Add route

\* Required

Cancel Save routes

Attach routing tables to subnets. R1-IGW to S3-Public and S4-Public, public network required to have internet access. Attach R2-VGW to S1-Private and S2-Private (No internet become a private subnets)

☐ Name Subnet ID State VPC

<input checked="" type="checkbox"/>	S1-Private	subnet-01b0a1e5be742dde0	available	vpc-02c316e5f1be2208a ...
<input type="checkbox"/>	S3-Public	subnet-01f8724bb68578a99	available	vpc-02c316e5f1be2208a ...

Subnet: subnet-01b0a1e5be742dde0

Description Flow Logs Route Table Network ACL Tags

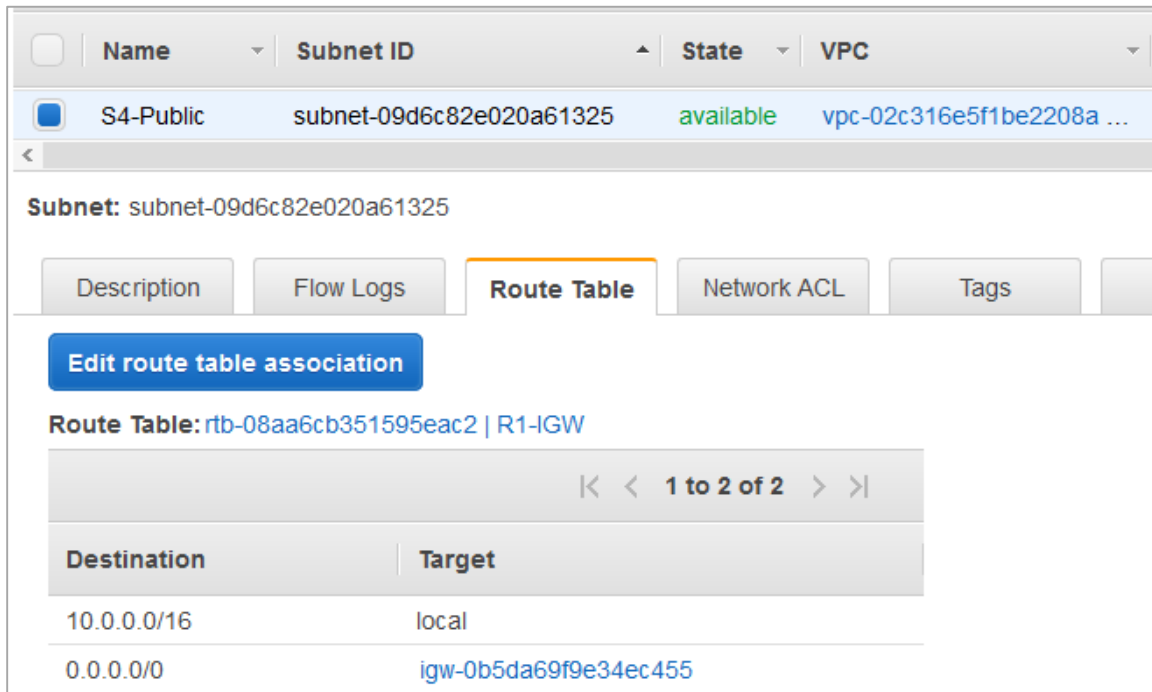
Edit route table association

www.server-computer.com

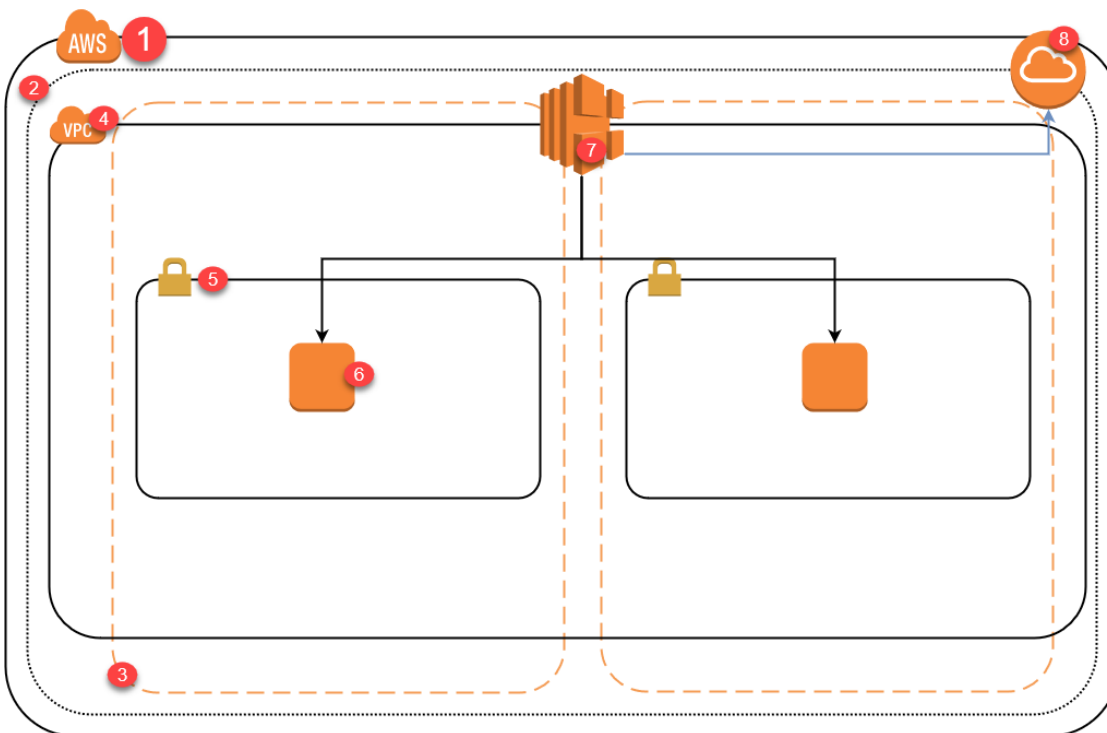
Route Table: rtb-0bd197f39222e69ea | R2-VGW

1 to 2 of 2

Destination	Target
192.168.0.0/16	vgw-0649463556a8290fe
10.0.0.0/16	local



### 13. AWS Elastic Load Balancer (ELB)



#### 2.1 Elastic Load Balancer Typical Architecture

1. AWS Cloud
2. Region
3. Availability Zone
4. VPC – Virtual Private Cloud
5. VPC Subnet
6. EC2 Instance Running Webserver
7. Elastic Load Balancer
8. Internet Gateway

**Elastic Load Balancing (ELB)** is a load-balancing service for Amazon Web Services (AWS) deployments. ELB automatically distributes incoming application traffic and scales resources to meet traffic demands.

A Managed Load Balancing service

- Distributes load incoming application traffic across multiple targets, such as amazon EC2 instances, containers, and IP Addresses
- Recognizes and responds to unhealthy instances
- Can be public or internal-facing
- Uses HTTP, HTTPS, TCP, and SSL Protocols
- Each Load Balancer is given a public DNS name
  - Internet-facing load balancers have DNS names which publicly resolve to the public IP Addresses of the load balancer of the load balancers nodes
  - Internal load balancers have DNS names, which publicly resolve to the private IP Addresses of the load balancers nodes.

### Types of ELB

1. Application Load Balancer
2. Network Load Balancer
3. Classic Load Balancer

### ELB Practical

- Launch two EC2 instances in different AZs
- Enable Web services
- Launch Load Balancer
- Add both instances under load balancer now check traffic

Follow **EC2 Linux instance launch steps** however in step two (configure Instance) go to down to the bottom in advanced section add below script will create auto webserver

```
#!/bin/bash
sudo yum update -y
sudo yum install httpd* -y
sudo service httpd start
sudo chkconfig httpd on
echo '<html><h1>Hello, Welcome to Server1</h1></html>' > /var/www/html/index.html
sudo service httpd restart
```

▼ Advanced Details

User data ⓘ

☒ As text ☐ As file ☐ Input is already base64 encoded

```
#!/bin/bash
sudo yum update -y
sudo yum install httpd* -y
sudo service httpd start
sudo chkconfig httpd on
echo '<html><h1>Hello, Welcome to Server1</h1></html>' > /var/www/html/index.html
sudo service httpd restart
```

**Note:** while launching second instance change echo statement to server2

```
echo '<html><h1>Hello, Welcome to Server2</h1></html>' > /var/www/html/index.html
```

## Creating Classic Elastic Load Balancer

Classic Load Balancer

PREVIOUS GENERATION  
for HTTP, HTTPS, and TCP

Create

Choose a Classic Load Balancer when you have an existing application running in the EC2-Classic network.

Learn more >

Step 1: Define Load Balancer

Basic Configuration

This wizard will walk you through setting up a new load balancer. Begin by giving your new load balancer a unique name so that you can identify it from other load balancers you might create. You will also need to configure ports and protocols for your load balancer. Traffic from your clients can be routed from any load balancer port to any port on your EC2 instances. By default, we've configured your load balancer with a standard web server on port 80.

Load Balancer name: server-computer

Create LB inside: vpc-02c316e5f1be2208a (10.0.0.0/16) | MyVPC

Create an internal load balancer: ☐ (what's this?)

Enable advanced VPC configuration: ☒

Listener Configuration:

Load Balancer Protocol	Load Balancer Port	Instance Protocol	Instance Port
HTTP	80	HTTP	80

Add

Select Subnets

You will need to select a Subnet for each Availability Zone where you wish traffic to be routed by your load balancer. If you have instances in only one Availability Zone, please select at least two Subnets in different Availability Zones to provide higher availability for your load balancer.

VPC vpc-02c316e5f1be2208a (10.0.0.0/16) | MyVPC

Available subnets

Actions	Availability Zone	Subnet ID	Subnet CIDR	Name
+	us-east-2a	subnet-01b0a1e5be742dde0	10.0.1.0/24	S1-Private
+	us-east-2b	subnet-0415e767640ae4ef9	10.0.2.0/24	S2-Private

Selected subnets

Actions	Availability Zone	Subnet ID	Subnet CIDR	Name
-	us-east-2a	subnet-01b0a1e5be742dde0	10.0.1.0/24	S1-Private
-	us-east-2b	subnet-0415e767640ae4ef9	10.0.2.0/24	S2-Private

Cancel

Next: Assign Security Groups

Click **Next: Assign Security Groups**

**Assign a security group:** ☒ Create a **new** security group  
☐ Select an **existing** security group

**Security group name:**

**Description:**

Type ⓘ	Protocol ⓘ	Port Range ⓘ
Custom TCP F ▾	TCP	80

**Add Rule**

Click **Next: Security Settings**

Click **Next: Configure Health Checks**

**Step 4: Configure Health Check**  
Your load balancer will automatically perform health checks on your

**Ping Protocol**

**Ping Port**

**Ping Path**

**Advanced Details**

<b>Response Timeout</b> ⓘ	<input type="text" value="5"/> seconds
<b>Interval</b> ⓘ	<input type="text" value="30"/> seconds
<b>Unhealthy threshold</b> ⓘ	<input type="text" value="2"/>
<b>Healthy threshold</b> ⓘ	<input type="text" value="10"/>

Specify your default web file in this example I am using /index.html

Click **Next: Add EC2 Instances**

**Step 5: Add EC2 Instances**  
The table below lists all your running EC2 Instances. Check the boxes in the Select column to add those instances to this load balancer.

VPC vpc-02c316e5f1be2208a (10.0.0.0/16) | MyVPC

Instance ▾	Name ▾	State ▾	Security groups
<input checked="" type="checkbox"/>	i-0e831d986cac3f5f6	● running	WebServer-Loadbalancer
<input checked="" type="checkbox"/>	i-0e02d814b0ce068bd	● running	WebServer-Loadbalancer

[www.server-computer.com](https://www.server-computer.com)

**Availability Zone Distribution**  
2 instances in us-east-2a

☒ Enable Cross-Zone Load Balancing ⓘ

☒ Enable Connection Draining ⓘ  seconds

Click **Next: Add Tags**

Click **Review and Create**

Click **Create**




Name	DNS name	State	VPC ID	Availability Zones	Type	Created At
server-computer	server-computer-921437411....		vpc-02c316e5f1be2208a	us-east-2a, us-east-2b	classic	December 5, 2018 at 6:01:1...

Load balancer: **server-computer**

Description Instances Health check Listeners Monitoring Tags Migration

Connection Draining: Enabled, 300 seconds ([Edit](#))

[Edit Instances](#)

Instance ID	Name	Availability Zone	Status	Actions
i-0e831d986cac3f5f6		us-east-2a	InService 	<a href="#">Remove from Load Balancer</a>

Check instances status should be InService

Load balancer: **server-computer**

Description Instances Health check Listeners Monitoring Tags Migration

**Basic Configuration**

Name	server-computer		
* DNS name	server-computer-	.us-east-2.elb.amazonaws.com (A Record)	

Load Balancer DNS Name copy it and paste in web browser now fresh twice you will see response is coming from Server1 and Server2



Which concludes load balancer is working fine.