

AWS Training Hands-on

Day 5

6th July 2023

Task 1 - Create VPC peering connection and connect an EC2 instance with a RDS instance.

Step 1 - Login to the AWS management console and browse to the VPC console.

Create a VPC for the application environment with the following configurations -

- **Name:** Test-VPC-App
- **IPv4 CIDR block:** Select “IPv4 CIDR manual input”
- **IPv4 CIDR:** 192.168.54.0/27
- **Tenancy:** Default

Click on “Click VPC”

Create VPC [Info](#)

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.

VPC settings

Resources to create [Info](#)
Create only the VPC resource or the VPC and other networking resources.

☒ VPC only ☐ VPC and more

Name tag - optional
Creates a tag with a key of 'Name' and a value that you specify.

Test-VPC-App

IPv4 CIDR block [Info](#)

☒ IPv4 CIDR manual input ☐ IPAM-allocated IPv4 CIDR block

IPv4 CIDR

192.168.54.0/27

IPv6 CIDR block [Info](#)

☒ No IPv6 CIDR block ☐ IPAM-allocated IPv6 CIDR block ☐ Amazon-provided IPv6 CIDR block ☐ IPv6 CIDR owned by me

Tenancy [Info](#)

Default ▼

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key

Value - optional

Q Name X

Q Test-VPC-App X

Remove tag

Add tag

You can add 49 more tags

Cancel **Create VPC**

Step 2 - Create a public subnet inside the above VPC.

Create subnet [Info](#)

VPC

VPC ID
Create subnets in this VPC.

vpc-0cd09d0d6b48c61d8 (Test-VPC-App) ▼

Associated VPC CIDRs

IPv4 CIDRs

192.168.54.0/27

Subnet settings

Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.

Test-VPC-App-Public

The name can be up to 256 characters long.

Availability Zone [Info](#)
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

US East (Ohio) / us-east-2a ▼

IPv4 CIDR block [Info](#)

Q 192.168.54.0/27 X

▼ **Tags - optional**

Key	Value - optional	
Q Name X	Q Test-VPC-App-Public X	Remove
<button>Add new tag</button>		
You can add 49 more tags.		
<button>Remove</button>		

Step 3 - Create an Internet gateway and attach it to the VPC.

Create internet gateway [Info](#)

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.

Internet gateway settings

Name tag
Creates a tag with a key of 'Name' and a value that you specify.

Test-VPC-App-IGW|

Step 4 - Create a route table for the public subnet and edit its subnet association and routes to route the internet facing traffic to the internet gateway.

Create route table

[Info](#)

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

Route table settings

Name - optional

Create a tag with a key of 'Name' and a value that you specify.

Test-VPC-App-Public-RT

VPC

The VPC to use for this route table.

vpc-0cd09d0d6b48c61d8 (Test-VPC-App)

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key

Value - optional

Q Name

X

Q Test-VPC-App-Public-RT

X

Remove

Add new tag

You can add 49 more tags.

Cancel

Create route table

Edit routes

Destination	Target	Status	Propagated
192.168.54.0/27	<div>Q local</div> <div>X</div>	Active	No
<div>Q 0.0.0.0/0</div> <div>X</div>	<div>Q igw-0fcdb15663a435465</div> <div>X</div>	-	No <div>Remove</div>

Add route

Cancel

Preview

Save changes

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (1/1)

Q Filter subnet associations

< 1 > ⌕

<input checked="" type="checkbox"/>	Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
<input checked="" type="checkbox"/>	Test-VPC-App-Public	subnet-0152e9b4ad4cc003b	192.168.54.0/27	-	rtb-043208b31decf8b21 / Test-VPC-A...

Selected subnets

subnet-0152e9b4ad4cc003b / Test-VPC-App-Public

X

Step 5 - Launch an EC2 instance in the public subnet.

Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags [Info](#)

Name

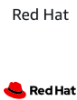
Test-VPC-App-PubInstance-01

[Add additional tags](#)

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Quick Start



[Browse more AMIs](#)

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 AMI

ami-03f38e546e3dc59e1 (64-bit (x86), uefi-preferred) / ami-009fb1b6af2b866d6 (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

Description

Amazon Linux 2023 AMI 2023.1.20230629.0 x86_64 HVM kernel-6.1

Architecture

64-bit (x86)

Boot mode

uefi-preferred

AMI ID

ami-03f38e546e3dc59e1

Verified provider

▼ Instance type [Info](#)

Instance type

t2.micro

Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Linux pricing: 0.0116 USD per Hour
On-Demand SUSE pricing: 0.0116 USD per Hour
On-Demand Windows pricing: 0.0162 USD per Hour
On-Demand RHEL pricing: 0.0716 USD per Hour

Free tier eligible

☒ All generations

[Compare instance types](#)

▼ Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

PubInstancePem

 [Create new key pair](#)

▼ Network settings [Info](#)

VPC - *required* [Info](#)

vpc-0cd09d0d6b48c61d8 (Test-VPC-App)
192.168.54.0/27



Subnet [Info](#)

subnet-0152e9b4ad4cc003b Test-VPC-App-Public
VPC: vpc-0cd09d0d6b48c61d8 Owner: 237042273450
Availability Zone: us-east-2a IP addresses available: 27 CIDR: 192.168.54.0/27

 [Create new subnet](#) 

Auto-assign public IP [Info](#)

Enable

Firewall (security groups) [Info](#)

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

☒ Create security group

☐ Select existing security group

Security group name - *required*

Public-SG

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and _-:/()#,@[]+=&;{}!\$*

Description - *required* [Info](#)

Allows traffic from internet

Inbound Security Group Rules

▼ Security group rule 1 (TCP, 22, 0.0.0.0/0)

[Remove](#)

Type [Info](#)

ssh

Protocol [Info](#)

TCP

Port range [Info](#)

22

Source type [Info](#)

Anywhere

Source [Info](#)

 Add CIDR, prefix list or security

0.0.0.0/0 

:::0 

Description - *optional* [Info](#)

e.g. SSH for admin desktop

Step 6 - Create a Second VPC for a private database.

- **Name:** Test-VPC-DB
- **IPv4 CIDR block:** Select “IPv4 CIDR manual input”
- **IPv4 CIDR:** 192.168.12.0/27
- **Tenancy:** Default

Create VPC [Info](#)

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.

VPC settings

Resources to create [Info](#)
Create only the VPC resource or the VPC and other networking resources.

☒ VPC only ☐ VPC and more

Name tag - optional
Creates a tag with a key of 'Name' and a value that you specify.

Test-VPC-DB

IPv4 CIDR block [Info](#)

☒ IPv4 CIDR manual input ☐ IPAM-allocated IPv4 CIDR block

IPv4 CIDR

192.168.12.0/27

IPv6 CIDR block [Info](#)

☒ No IPv6 CIDR block ☐ IPAM-allocated IPv6 CIDR block ☐ Amazon-provided IPv6 CIDR block ☐ IPv6 CIDR owned by me

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key

Value - optional

Q Name X

Q Test-VPC-DB X

Remove tag

Add tag

You can add 49 more tags

Cancel

Create VPC

Step 7 - Create two private subnets in the above VPC.

Create subnet [Info](#)

VPC

VPC ID
Create subnets in this VPC.

vpc-08f133c3e21dc7a72 (Test-VPC-DB) ▼

Associated VPC CIDRs

IPv4 CIDRs

192.168.12.0/27

Subnet 1 of 2

Subnet name

Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

Availability Zone [Info](#)

Choose the zone in which your subnet will reside, or let Amazon choose one for you.

IPv4 CIDR block [Info](#)

▼ Tags - optional

Key

Value - optional

Subnet 2 of 2

Subnet name

Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

Availability Zone [Info](#)

Choose the zone in which your subnet will reside, or let Amazon choose one for you.

IPv4 CIDR block [Info](#)

▼ Tags - optional

Key

Value - optional

Step 8 - Create a route table for the above subnet.

Create route table [Info](#)

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

Route table settings

Name - optional

Create a tag with a key of 'Name' and a value that you specify.

VPC

The VPC to use for this route table.

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key

Value - optional

You can add 49 more tags.

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (2/2)

Filter subnet associations

<input checked="" type="checkbox"/>	Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
<input checked="" type="checkbox"/>	Test-VPC-DB-2a	subnet-0ec75cc2d04df97bf	192.168.12.0/28	-	Main (rtb-0fccfbee1e867778c)
<input checked="" type="checkbox"/>	Test-VPC-DB-2c	subnet-0613a9941558c519a	192.168.12.16/28	-	Main (rtb-0fccfbee1e867778c)

Selected subnets

subnet-0ec75cc2d04df97bf / Test-VPC-DB-2a X subnet-0613a9941558c519a / Test-VPC-DB-2c X

Cancel Save associations

Step 9 - Create a RDS instance in the above VPC.

The RDS instance stores the backup data and logs for redundancy in other availability zones. And the data of the RDS instance is also stored in the other availability zone. Therefore the specified VPC requires at least two subnets each in different AZs.

Create database

Choose a database creation method [Info](#)

☒ Standard create

You set all of the configuration options, including ones for availability, security, backups, and maintenance.

☐ Easy create

Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

Engine options

Engine type [Info](#)

☐ Aurora (MySQL Compatible)



☐ Aurora (PostgreSQL Compatible)



☒ MySQL



☐ MariaDB



☐ PostgreSQL



☐ Oracle

ORACLE®

☐ Microsoft SQL Server



Edition

☒ MySQL Community

Templates

Choose a sample template to meet your use case.

☐ **Production**

Use defaults for high availability and fast, consistent performance.

☐ **Dev/Test**

This instance is intended for development use outside of a production environment.

☒ **Free tier**

Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS. [Info](#)

Settings

DB instance identifier [Info](#)

Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.

Test-VPC-UserDB

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

▼ Credentials Settings

Master username [Info](#)


Type a login ID for the master user of your DB instance.


admin

1 to 16 alphanumeric characters. First character must be a letter.

☐ **Manage master credentials in AWS Secrets Manager**

Manage master user credentials in Secrets Manager. RDS can generate a password for you and manage it throughout its lifecycle.

 If you manage the master user credentials in Secrets Manager, some RDS features aren't supported.

[Learn more](#) 

☐ **Auto generate a password**

Amazon RDS can generate a password for you, or you can specify your own password.

Master password [Info](#)

Constraints: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), '(single quote), "(double quote) and @ (at sign).

Confirm master password [Info](#)

Connectivity [Info](#)



Compute resource

Choose whether to set up a connection to a compute resource for this database. Setting up a connection will automatically change connectivity settings so that the compute resource can connect to this database.

☒ **Don't connect to an EC2 compute resource**

Don't set up a connection to a compute resource for this database. You can manually set up a connection to a compute resource later.

☐ **Connect to an EC2 compute resource**

Set up a connection to an EC2 compute resource for this database.

Network type [Info](#)

To use dual-stack mode, make sure that you associate an IPv6 CIDR block with a subnet in the VPC you specify.

☒ **IPv4**

Your resources can communicate only over the IPv4 addressing protocol.

☐ **Dual-stack mode**

Your resources can communicate over IPv4, IPv6, or both.

Virtual private cloud (VPC) [Info](#)

Choose the VPC. The VPC defines the virtual networking environment for this DB instance.

Test-VPC-DB (vpc-08f133c3e21dc7a72)

2 Subnets, 2 Availability Zones



Only VPCs with a corresponding DB subnet group are listed.

DB subnet group [Info](#)

Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB instance can use in the VPC that you selected.

default-vpc-08f133c3e21dc7a72

2 Subnets, 2 Availability Zones



Public access [Info](#)

☐ **Yes**

RDS assigns a public IP address to the database. Amazon EC2 instances and other resources outside of the VPC can connect to your database. Resources inside the VPC can also connect to the database. Choose one or more VPC security groups that specify which resources can connect to the database.

☒ **No**

RDS doesn't assign a public IP address to the database. Only Amazon EC2 instances and other resources inside the VPC can connect to your database. Choose one or more VPC security groups that specify which resources can connect to the database.

VPC security group (firewall) [Info](#)

Choose one or more VPC security groups to allow access to your database. Make sure that the security group rules allow the appropriate incoming traffic.

☐ **Choose existing**

Choose existing VPC security groups

☒ **Create new**

Create new VPC security group

New VPC security group name

Test-VPC-DB-SG

Availability Zone [Info](#)

us-east-2a



Manage your database user credentials through your DB engine's native password authentication features.

Database authentication

Database authentication options [Info](#)

☒ Password authentication
Authenticates using database passwords.

☐ Password and IAM database authentication
Authenticates using the database password and user credentials through AWS IAM users and roles.

☐ Password and Kerberos authentication
Choose a directory in which you want to allow authorized users to authenticate with this DB instance using Kerberos Authentication.

Estimated monthly costs

The Amazon RDS Free Tier is available to you for 12 months. Each calendar month, the free tier will allow you to use the Amazon RDS resources listed below for free:

- 750 hrs of Amazon RDS in a Single-AZ db.t2.micro, db.t3.micro or db.t4g.micro Instance.
- 20 GB of General Purpose Storage (SSD).
- 20 GB for automated backup storage and any user-initiated DB Snapshots.

[Learn more about AWS Free Tier.](#)

When your free usage expires or if your application use exceeds the free usage tiers, you simply pay standard, pay-as-you-go service rates as described in the [Amazon RDS Pricing page.](#)

The RDS instance is created successfully.

☐

test-vpc-userdb

☒ Available

Instance

MySQL Community

us-east-2a

db.t3.micro

Step 10 - Connect the RDS instance to the EC2 instance created in another VPC by making a Peering connection between the two VPCs.

A VPC peering connection is made by assigning a requester VPC and an acceptor VPC. The requester VPC requests the other VPC to establish a peering connection by accepting the request for the same.

Here the requester VPC is - “Test-VPC-App”, and acceptor VPC is - “Test-VPC-DB”

Create peering connection

A VPC peering connection is a networking connection between two VPCs that enables you to route traffic between them privately. [Info](#)

Peering connection settings

Name - optional

Create a tag with a key of 'Name' and a value that you specify.

Test-VPC-App-Db-Peering

Select a local VPC to peer with

VPC ID (Requester)

vpc-0cd09d0d6b48c61d8 (Test-VPC-App)

VPC CIDRs for vpc-0cd09d0d6b48c61d8 (Test-VPC-App)

CIDR	Status	Status reason
192.168.54.0/27	✔ Associated	-

Select another VPC to peer with

Account

☒ My account

☐ Another account

Region

☒ This Region (us-east-2)

☐ Another Region

VPC ID (Acceptor)

vpc-08f133c3e21dc7a72 (Test-VPC-DB)

VPC CIDRs for vpc-08f133c3e21dc7a72 (Test-VPC-DB)

CIDR	Status	Status reason
192.168.12.0/27	✔ Associated	-

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key

Q Name

Value - optional

Q Test-VPC-App-Db-Peering

Remove

Add new tag

You can add 49 more tags.

Cancel

Create peering connection

Step 11 - Accept the peering connection request.

As both the VPCs are in the same account therefore we can see the pending request dialog.

pcx-0de384fa646e614e2 / Test-VPC-App-Db-Peering



Pending acceptance

You can accept or reject this peering connection request using the 'Actions' menu. You have until Thursday, July 13, 2023 at 17:34:19 GMT- the request, otherwise it expires.

Details [Info](#)

Actions ▲

Accept request

Reject request

Edit DNS settings

Manage tags

Delete peering connection

Step 12 - Modify the route tables of both the VPCs.

✔ Your VPC peering connection (pcx-0de384fa646e614e2 | Test-VPC-App-Db-Peering) has been established.
To send and receive traffic across this VPC peering connection, you must add a route to the peered VPC in one or more of your VPC route tables. [Info](#)

Modify my route tables now

VPC > Peering connections > pcx-0de384fa646e614e2

pcx-0de384fa646e614e2 / Test-VPC-App-Db-Peering

Actions

1. Route table of the **“Test-VPC-App”** VPC should contain a route to forward the traffic for the **“Test-VPC-DB”** VPC i.e., for the CIDR block of **“192.168.12.0/27”** to the peering connection as a target.

Edit routes

Destination	Target	Status	Propagated
192.168.54.0/27	local	Active	No
192.168.12.0/27	pcx-0de384fa646e614e2	Active	No
0.0.0.0/0	igw-0fcd15663a435465	Active	No

Add route

Cancel Preview Save changes

2. Route table of the **“Test-VPC-DB”** VPC should contain a route to forward the traffic for the **“Test-VPC-App”** VPC i.e., for the CIDR block of **“192.168.54.0/27”** to the peering connection as a target.

Edit routes

Destination	Target	Status	Propagated
192.168.12.0/27	local	Active	No
192.168.54.0/27	pcx-0de384fa646e614e2	-	No

Add route

Cancel Preview Save changes

Step 13 - Checking if the RDS instance is accessible from the EC2 instance in the “Test-VPC-App” VPC.

Using **“nc”** command to listen to the port of the RDS instance and verify that the EC2 instance can connect to the RDS instance.

```
[ec2-user@ip-192-168-54-8 ~]$ nc -zv test-vpc-userdb.cso7be609p5s.us-east-2.rds.amazonaws.com 3306
Ncat: Version 7.93 ( https://nmap.org/ncat )
Ncat: Connected to 192.168.12.9:3306.
Ncat: 0 bytes sent, 0 bytes received in 0.04 seconds.
```

Step 14 - Connect to the RDS instance.

Use the “**mysql**” command to connect to the instance, enter the endpoint, port and username of the database.

Enter the password for the user.

```
[ec2-user@ip-192-168-54-8 ~]$ mysql --version
mysql Ver 15.1 Distrib 10.5.18-MariaDB, for Linux (x86_64) using EditLine wrapper
[ec2-user@ip-192-168-54-8 ~]$ mysql -h test-vpc-userdb.cso7be6o9p5s.us-east-2.rds.amazonaws.com -P 3306 -u admin -p
Enter password:
```

Step 15 - Display databases, create a new database.

```
MySQL [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
4 rows in set (0.001 sec)

MySQL [(none)]> create database users;
Query OK, 1 row affected (0.006 sec)

MySQL [(none)]> use users;
Database changed
MySQL [users]> show tables;
Empty set (0.002 sec)
```

Step 16 - Create table “userDetails” with fields - “UserId, userName, and UserAddress”

Use the “**CREATE TABLE**” command to create a new table.

```
MySQL [users]> CREATE TABLE userDetails (UserId int, UserName varchar(255), UserAddress varchar(255));
Query OK, 0 rows affected (0.023 sec)

MySQL [users]> show tables;
+-----+
| Tables_in_users |
+-----+
| userDetails |
+-----+
1 row in set (0.001 sec)
```

Step 17 - Insert data into the table.

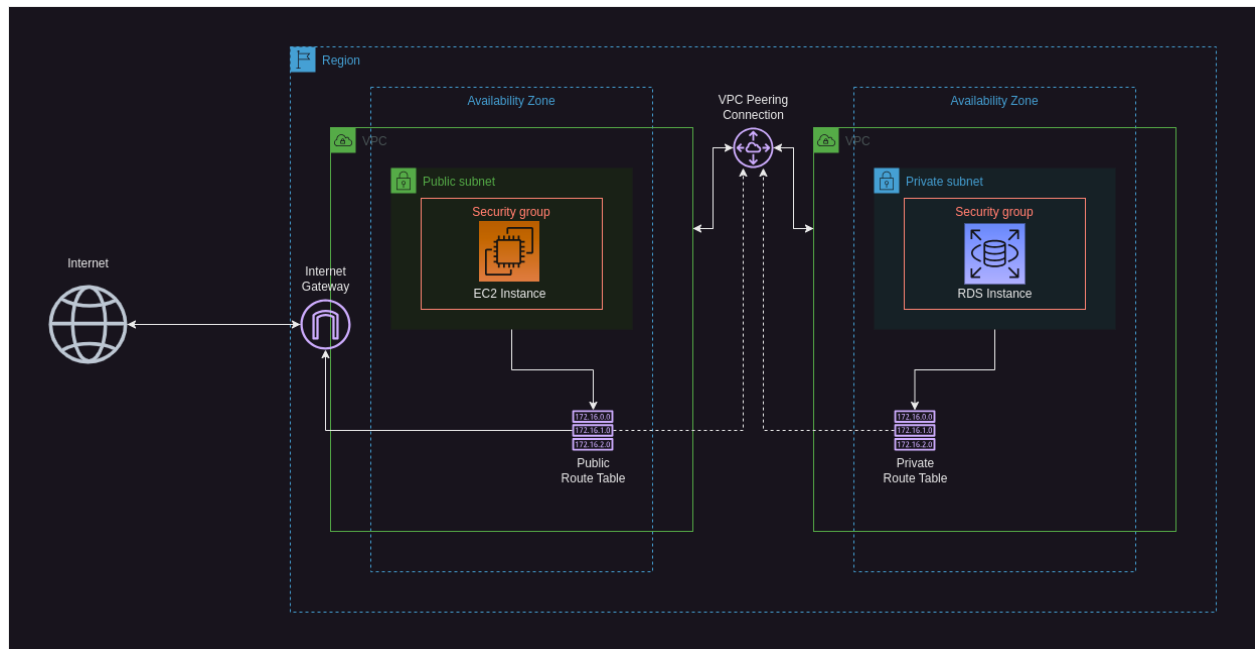
Use the “**INSERT INTO <tablename> VALUES**” command.

Use the “**SELECT**” command to display the content of the table.

```
MySQL [users]> INSERT INTO userDetails (UserId, UserName, UserAddress) VALUES (01, "User1SK", "Indore");
Query OK, 1 row affected (0.005 sec)

MySQL [users]> SELECT * FROM userDetails;
+-----+-----+-----+
| UserId | UserName | UserAddress |
+-----+-----+-----+
| 1 | User1SK | Indore |
+-----+-----+-----+
1 row in set (0.001 sec)
```

Architecture Diagram for the above solution



Task 2 - Create a gateway endpoint to connect EC2 inside a VPC to a S3 bucket.

Step 1 - Create a VPC with a CIDR block - “172.16.0.0/16”

Create VPC [Info](#)

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.

VPC settings

Resources to create [Info](#)
Create only the VPC resource or the VPC and other networking resources.

☒ VPC only ☐ VPC and more

Name tag - optional
Creates a tag with a key of 'Name' and a value that you specify.

Test-VPC-Gateway

IPv4 CIDR block [Info](#)
☒ IPv4 CIDR manual input
☐ IPAM-allocated IPv4 CIDR block

IPv4 CIDR
172.16.0.0/16

IPv6 CIDR block [Info](#)
☒ No IPv6 CIDR block
☐ IPAM-allocated IPv6 CIDR block
☐ Amazon-provided IPv6 CIDR block
☐ IPv6 CIDR owned by me

Tenancy [Info](#)
Default

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional	
<input type="text" value="Name"/>	<input type="text" value="Test-VPC-Gateway"/>	<input type="button" value="Remove tag"/>

You can add 49 more tags

Step 2 - Create a public subnet.

VPC

VPC ID
Create subnets in this VPC.

vpc-0fcc6b1d50be5c145 (Test-VPC-Gateway)

Associated VPC CIDRs

IPv4 CIDRs
172.16.0.0/16

Subnet settings

Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name

Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

Availability Zone [Info](#)

Choose the zone in which your subnet will reside, or let Amazon choose one for you.

IPv4 CIDR block [Info](#)

Step 2 - Create a private subnet in the VPC.

Subnet 2 of 2

Subnet name

Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

Availability Zone [Info](#)

Choose the zone in which your subnet will reside, or let Amazon choose one for you.

IPv4 CIDR block [Info](#)

▼ Tags - optional

Key

Value - optional

You can add 49 more tags.

Step 3 - Create an internet gateway and attach it to the VPC.

Create internet gateway [Info](#)

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.

Internet gateway settings

Name tag
Creates a tag with a key of 'Name' and a value that you specify.

Tags - optional

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional	
<input type="text" value="Name"/>	<input type="text" value="Test-VPC-Gateway-IGW"/>	<input type="button" value="Remove"/>

You can add 49 more tags.

Step 4 - Create route table and edit routes and subnet association.

Create route table [Info](#)

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

Route table settings

Name - optional
Create a tag with a key of 'Name' and a value that you specify.

VPC
The VPC to use for this route table.

vpc-0fcc6b1d50be5c145 (Test-VPC-Gateway) ▼

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional	
<input type="text" value="Name"/>	<input type="text" value="Test-VPC-Gateway-publicRT"/>	<input type="button" value="Remove"/>

You can add 49 more tags.

Step 5 - Create a private route table.

Create route table [Info](#)

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

Route table settings

Name - *optional*
Create a tag with a key of 'Name' and a value that you specify.

VPC
The VPC to use for this route table.

vpc-0fcc6b1d50be5c145 (Test-VPC-Gateway) ▼

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key

Value - *optional*

Remove

Add new tag

You can add 49 more tags.

Cancel **Create route table**

Step 6 - Create a S3 bucket.

Create bucket [Info](#)

Buckets are containers for data stored in S3. [Learn more](#)

General configuration

Bucket name

Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)

AWS Region

US East (Ohio) us-east-2 ▼

Copy settings from existing bucket - *optional*
Only the bucket settings in the following configuration are copied.

Choose bucket

Object Ownership [Info](#)

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

☒ ACLs disabled (recommended)

All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

☐ ACLs enabled

Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Object Ownership

Bucket owner enforced

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

☒ Block *all* public access

Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

Block public access to bucket objects

Block public access to bucket

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Block public access to bucket and objects

Step 6 - Launch EC2 instance in the public subnet.

It will be used to SSH into the private instance.

Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags [Info](#)

Name

Test-VPC-Gateway-publicInstance1

[Add additional tags](#)

▼ Instance type [Info](#)

Instance type

t2.micro Free tier eligible

Family: t2 1 vCPU 1 GiB Memory Current generation: true

On-Demand Linux pricing: 0.0116 USD per Hour

On-Demand SUSE pricing: 0.0116 USD per Hour

On-Demand Windows pricing: 0.0162 USD per Hour

On-Demand RHEL pricing: 0.0716 USD per Hour

☒ All generations

[Compare instance types](#)

▼ Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

PubInstancePem

[Create new key pair](#)

▼ Network settings [Info](#)

VPC - *required* [Info](#)

vpc-0fcc6b1d50be5c145 (Test-VPC-Gateway)

172.16.0.0/16

[Create new VPC](#)

Subnet [Info](#)

subnet-07fa403bbaca163df Test-VPC-Gateway-public-2a

VPC: vpc-0fcc6b1d50be5c145 Owner: 237042273450 Availability Zone: us-east-2a

IP addresses available: 251 CIDR: 172.16.1.0/24

[Create new subnet](#)

Auto-assign public IP [Info](#)

Enable

Firewall (security groups) [Info](#)

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

☒ Create security group

☐ Select existing security group

Security group name - *required*

public-SG

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and .-:/()#,@[]+=&:{}!\$*

Description - *required* [Info](#)

launch-wizard-1 created 2023-07-06T16:49:50.519Z

Inbound Security Group Rules

▼ Security group rule 1 (TCP, 22, 0.0.0.0/0)

[Remove](#)

Type [Info](#)

ssh

Protocol [Info](#)

TCP

Port range [Info](#)

22

Source type [Info](#)

Anywhere

Source [Info](#)

[Add CIDR, prefix list or security](#)

0.0.0.0/0

::/0

Description - *optional* [Info](#)

e.g. SSH for admin desktop

Step 7 - Launch EC2 instance in the private subnet.

Name and tags [Info](#)

Name

Test-VPC-Gateway-privateInstance

Add additional tags

▼ Instance type [Info](#)

Instance type

t2.micro

Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Linux pricing: 0.0116 USD per Hour
On-Demand SUSE pricing: 0.0116 USD per Hour
On-Demand Windows pricing: 0.0162 USD per Hour
On-Demand RHEL pricing: 0.0716 USD per Hour

Free tier eligible

☐ All generations

[Compare instance types](#)

▼ Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

PubInstancePem

[Create new key pair](#)

▼ Network settings [Info](#)

VPC - *required* [Info](#)

vpc-0fcc6b1d50be5c145 (Test-VPC-Gateway)
172.16.0.0/16

Subnet [Info](#)

subnet-0fa5110ce2fb0e92c Test-VPC-Gateway-private-2a
VPC: vpc-0fcc6b1d50be5c145 Owner: 237042273450 Availability Zone: us-east-2a
IP addresses available: 251 CIDR: 172.16.2.0/24

[Create new subnet](#)

Auto-assign public IP [Info](#)

Disable

Firewall (security groups) [Info](#)

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

☒ Create security group

☐ Select existing security group

Security group name - *required*

private-SG

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255.

Description - *required* [Info](#)

launch-wizard-1 created 2023-07-06T16:54:39.182Z

Inbound Security Group Rules

▼ Security group rule 1 (TCP, 22, 0.0.0.0/0)

Remove

Type [Info](#)

ssh

Protocol [Info](#)

TCP

Port range [Info](#)

22

Source type [Info](#)

Anywhere

Source [Info](#)

Q Add CIDR, prefix list or security

0.0.0.0/0 X ::/0 X

Description - *optional* [Info](#)

e.g. SSH for admin desktop

Step 8 - Create an endpoint for the VPC.

Create endpoint [Info](#)

There are three types of VPC endpoints – Interface endpoints, Gateway Load Balancer endpoints, and Gateway endpoints. Interface endpoints and Gateway Load Balancer endpoints are powered by AWS PrivateLink, and use an Elastic Network Interface (ENI) as an entry point for traffic destined to the service. Interface endpoints are typically accessed using the public or private DNS name associated with the service, while Gateway endpoints and Gateway Load Balancer endpoints serve as a target for a route in your route table for traffic destined for the service.

Endpoint settings

Name tag - *optional*

Creates a tag with a key of 'Name' and a value that you specify.

my-endpoint-01

Service category

Select the service category

☒ AWS services

Services provided by Amazon

☐ PrivateLink Ready partner services

Services with an AWS Service Ready designation

☐ AWS Marketplace services

Services that you've purchased through AWS Marketplace

☐ EC2 Instance Connect Endpoint

An elastic network interface that allow you to connect to resources in a private subnet

☐ Other endpoint services

Find services shared with you by service name

Services (1/2)

Q Find resources by attribute or tag

Service Name = com.amazonaws.us-east-2.s3 X

Clear filters

< 1 > ⚙

	Service Name	Owner	Type
<input checked="" type="radio"/>	com.amazonaws.us-east-2.s3	amazon	Gateway
<input type="radio"/>	com.amazonaws.us-east-2.s3	amazon	Interface

Step 10 - Modify the route table for the VPC endpoint. Attach the private route table to the VPC endpoint.

Manage route tables [Info](#)

Subnets associated with selected route tables will be able to access this endpoint.

Route tables (1/3)

Find resources by attribute or tag

<input type="checkbox"/>	Name	Route Table ID	Main	Associated Id
<input type="checkbox"/>	--	rtb-003f97b9caf605c5b	Yes	subnet-09107900
<input type="checkbox"/>	Test-VPC-Gateway-publicRT	rtb-097e14d0db87c7d94 (Test-VPC-Ga...	No	subnet-07fa403b
<input checked="" type="checkbox"/>	Test-VPC-Gateway-privateRT	rtb-0fa23574a0e0b0ade (Test-VPC-Gat...	No	subnet-0fa5110c

When you use an endpoint, the source IP addresses from your instances in your affected subnets for accessing the AWS service in the same region will be private IP addresses, not public IP addresses. Existing connections from your affected subnets to the AWS service that use public IP addresses may be dropped. Ensure that you don't have critical tasks running when you create or modify an endpoint.

Cancel **Modify route tables**

Step 11 - Creating a sample file and uploading it to the bucket.

```
[ec2-user@ip-172-16-2-70 ~]$ aws s3 ls
2023-07-06 15:42:28 test-bucket-gateway-sk
[ec2-user@ip-172-16-2-70 ~]$ vi sampleFile
[ec2-user@ip-172-16-2-70 ~]$ cat sampleFile
This is a sample file to be uploaded in S3 bucket
[ec2-user@ip-172-16-2-70 ~]$ aws s3 cp sampleFile s3://test-bucket-gateway-sk/sampleFile
upload: ./sampleFile to s3://test-bucket-gateway-sk/sampleFile
[ec2-user@ip-172-16-2-70 ~]$ aws s3 ls s3://test-bucket-gateway-sk/
2023-07-06 17:22:51      50 sampleFile
```

Step 12 - Download the uploaded file into the instance and display its content.

```
[ec2-user@ip-172-16-2-70 ~]$ aws s3 cp s3://test-bucket-gateway-sk/sampleFile downloadedSampleFile
download: s3://test-bucket-gateway-sk/sampleFile to ./downloadedSampleFile
[ec2-user@ip-172-16-2-70 ~]$ cat downloadedSampleFile
This is a sample file to be uploaded in S3 bucket
[ec2-user@ip-172-16-2-70 ~]$
```

Architecture diagram for the above solution.

