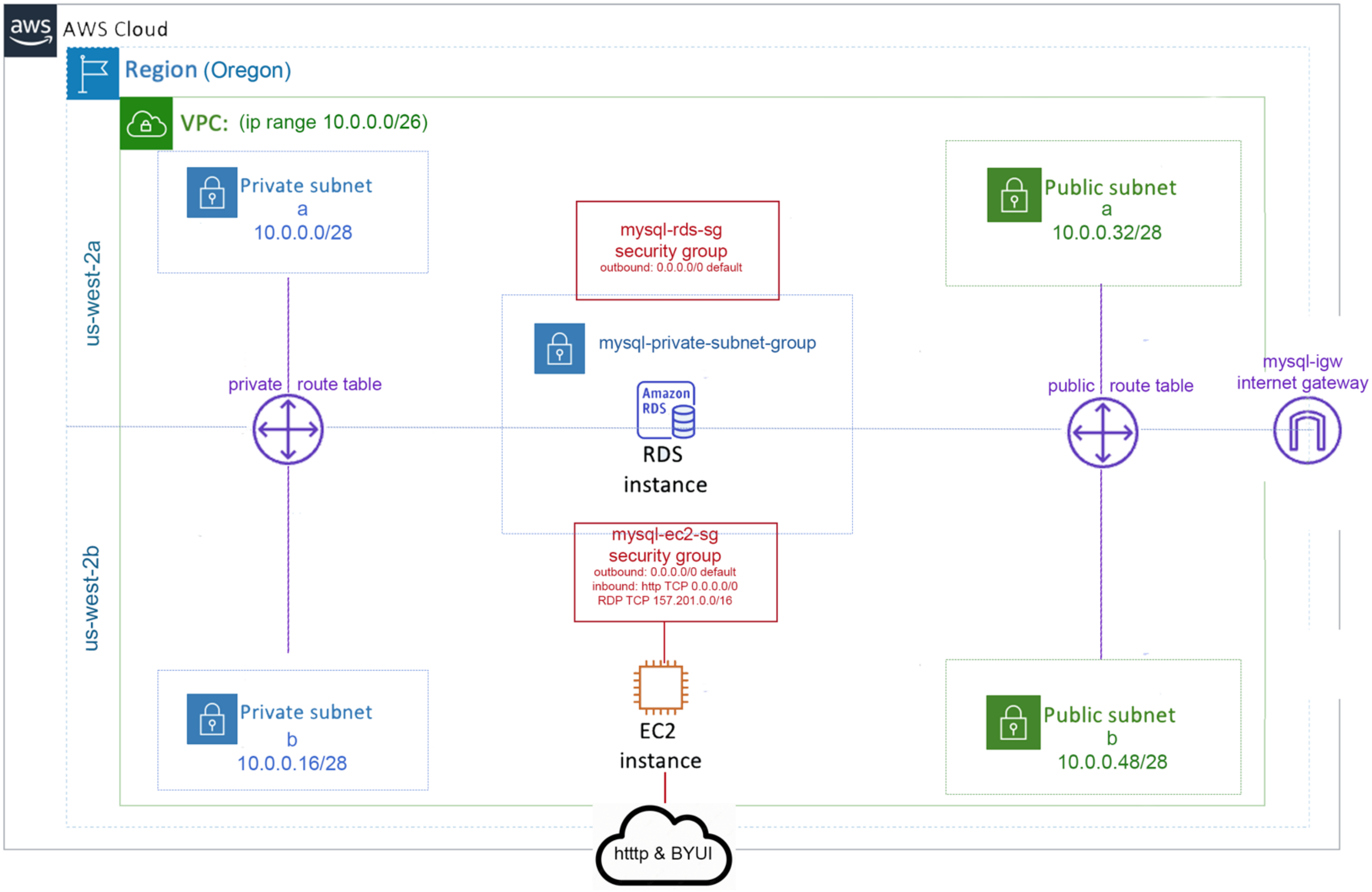
Lab C

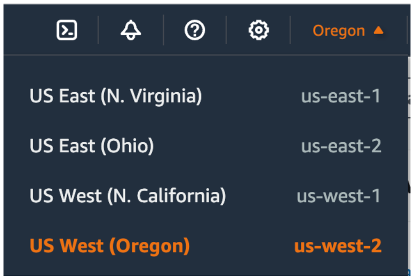
Creating a Virtual Network in the Cloud (VPC) with subnets (IP address ranges) and an EC2 with a MySQL RDS database within the VPC


AWS Cloud
Region (Oregon)
VPC: (ip range 10.0.0.0/26)
us-west-2a
us-west-2b
Private subnet a 10.0.0.0/28
Private subnet b 10.0.0.16/28
Public subnet a 10.0.0.32/28
Public subnet b 10.0.0.48/28
mysql-rds-sg security group
outbound: 0.0.0.0/0 default
mysql-private-subnet-group
Amazon RDS
RDS instance
mysql-ec2-sg security group
outbound: 0.0.0.0/0 default
inbound: http TCP 0.0.0.0/0
RDP TCP 157.201.0.0/16
EC2 instance
private route table
public route table
mysql-igw internet gateway
htttp & BYUI

1. Open the Learner Lab.

From our AWS canvas dashboard choose the Learner Lab and open the Modules and then the ‘Launch AWS Academy Learner Lab’ link. Make sure to click the Start lab link and wait for the AWS link to have the green circle next to it and open the lab.

Change the region to US West (Oregon) for this lab.


Oregon

US East (N. Virginia) us-east-1

US East (Ohio) us-east-2

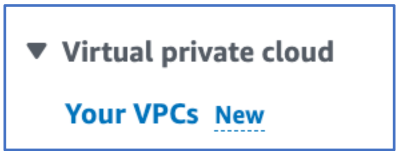
US West (N. California) us-west-1

US West (Oregon) us-west-2

2. Create a Virtual private cloud network

We will set up a virtual network that we have complete control over.

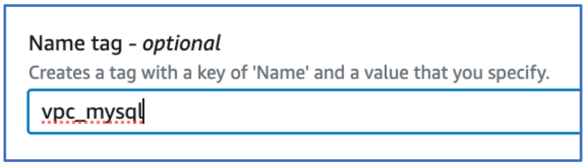
Open the VPC service and select 'Your VPCs' in the left side panel.



You are always provided default VPC(s). We will be creating a new one. Click 'Create VPC'.

Choose 'VPC only',

Give the VPC a Name tag.

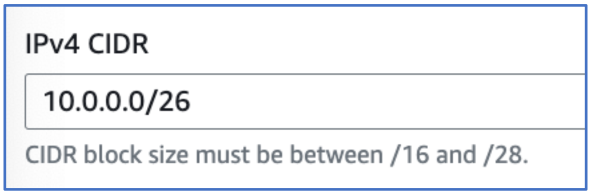

Name tag - *optional*

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

vpc_mysql

Leave the IPv4 CIDER (Classless Inter-domain routing) manual input.

We will be creating a range of 64 IP addresses. So, type in 10.0.0.0/26



IPv4 CIDR

10.0.0.0/26

CIDR block size must be between /16 and /28.

Leave the rest of the defaults and click ‘Create VPC’

Now when you click on ‘Your VPCs’ you will see your new VPC and see that it’s available.

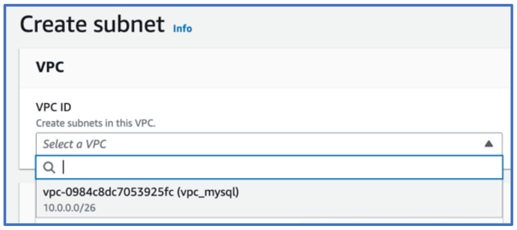
3. Create four subnets (ranges of IP addresses) inside your VPC.

Now click on ‘Subnets’ in the left side panel. You need subnets in your VPC that specify IP address ranges inside AZs to allocate virtual machines and other services.

Notice there are subnets already set up for our default VPC. We need to create subnets in our new VPC. Click ‘Create subnet’

We will set up 4 subnets, 2 public (like applications on the internet) and 2 private (like application servers). Public subnets have access to the internet whereas private subnets have no access to the internet. Each subnet will get 16 of the 64 IP addresses each. Multiple AZs insures high availability.

Select the VPC that you will be creating this subnet for. Choose the VPC you just created.



Create subnet Info

VPC

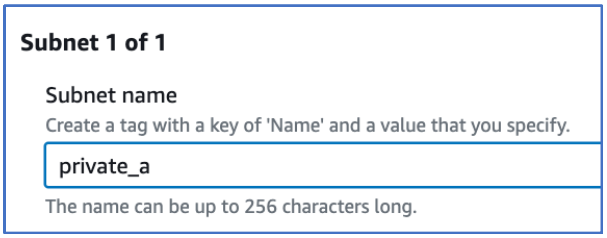
VPC ID

Create subnets in this VPC.

Select a VPC

vpc-0984c8dc7053925fc (vpc_mysql)
10.0.0.0/26

Name this first subnet private_a.



Subnet 1 of 1

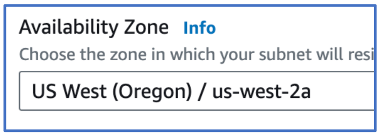
Subnet name

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

private_a

The name can be up to 256 characters long.

Choose an availability zone us-west-2a.

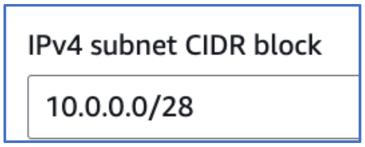


Availability Zone Info

Choose the zone in which your subnet will resi

US West (Oregon) / us-west-2a

Change the ‘IPv4 CIDR block’ to 10.0.0.0/28 for the first 16 IP addresses.

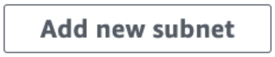


IPv4 subnet CIDR block

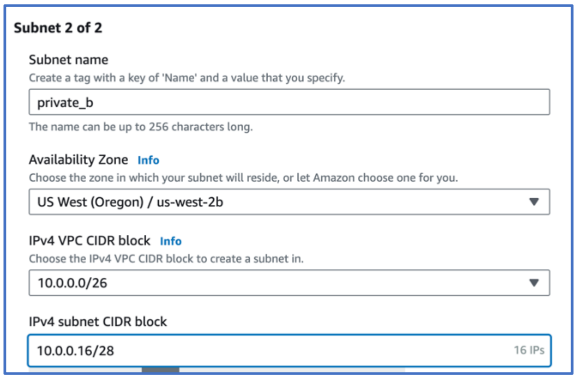
10.0.0.0/28

Use the rest of the defaults for the first subnet.

Click the ‘Add new subnet’ button.



For subnet 2 of 2, give it the name of private_b. Choose a different AZ us-west-2b. Change the ‘IPv4 CIDR block’ to 10.0.0.16/28 for the next 16 IP addresses.



Subnet 2 of 2

Subnet name

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

private_b

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 West (Oregon) / us-west-2b ▼

IPv4 VPC CIDR block Info

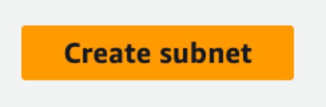
Choose the IPv4 VPC CIDR block to create a subnet in.

10.0.0.0/26 ▼

IPv4 subnet CIDR block

10.0.0.16/28 16 IPs

Leave the rest defaults and click ‘Create subnet’.



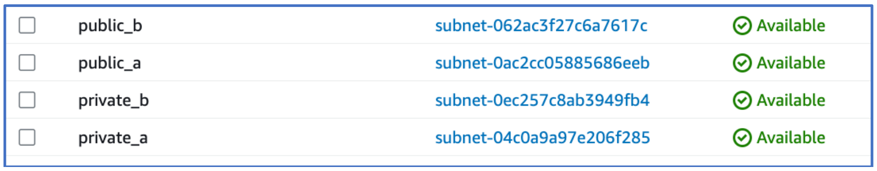
Now we have 2 private subnets in two AZs.

Repeat the process above to create two more subnets. The only difference will be that these 2 subnets will be public.

public_a in us-west-2a with the IP range for the first will be 10.0.0.32/28

public_b in us-west-2b with the IP range for the second will be 10.0.0.48/28

When you click ‘Subnets’ you should now see, in addition to the default subnets, your 4 new subnets (you might need to clear filters to see them)



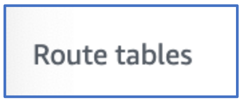
□	public_b	subnet-062ac3f27c6a7617c	✓ Available
□	public_a	subnet-0ac2cc05885686eeb	✓ Available
□	private_b	subnet-0ec257c8ab3949fb4	✓ Available
□	private_a	subnet-04c0a9a97e206f285	✓ Available

Notice under the available IPv4 addresses column it shows 11 in each subnet. That’s 11 for each not 16. The last 5 are reserved by AWS for IP networking purposes.

4. Create routing tables

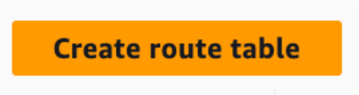
Routing tables determine where network traffic from your subnet or gateway is directed.

Click ‘Route tables’

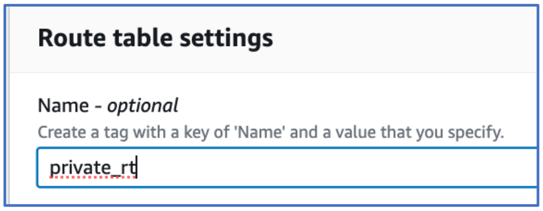


Again, you will see default route tables used with the default VPC. We will create 2 route tables for our VPC, one private and one public. We will associate our subnets into the route tables.

Click ‘Create route table’.



Name the first private route table.



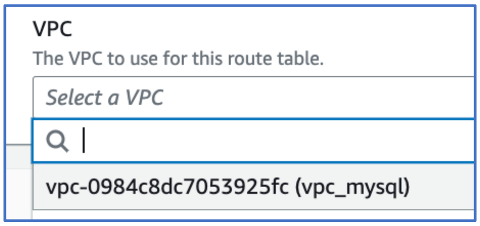
Route table settings

Name - *optional*

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

private_rt

Choose our VPC.



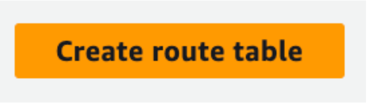
VPC

The VPC to use for this route table.

Select a VPC

vpc-0984c8dc7053925fc (vpc_mysql)

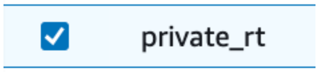
Click ‘Create route table’.



Create another public route table the same way, ‘public_rt’ using the same VPC.

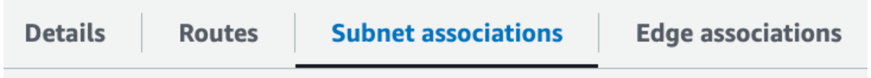
5. Associate your routing tables with the subnets.

Now when you click ‘Route tables’ you will see them listed. Check the private route table.

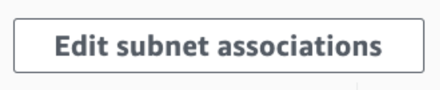


☑ private_rt

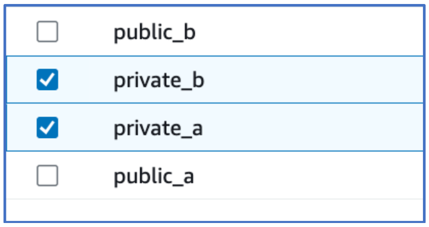
Click the ‘Subnet associations’ tab below.



Click the ‘Edit subnet associations’.

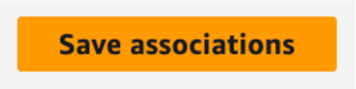


Check the 2 private subnets to associate with the private routing table.



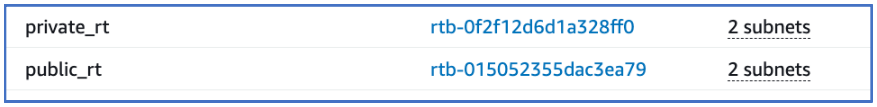
□ public_b
☑ private_b
☑ private_a
□ public_a

Click ‘Save associations’



Now check public route table and associate it with the 2 public subnets the same way we did the private ones.

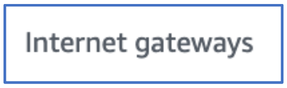
You should now see 2 subnets associated with each route table.



private_rt	rtb-0f2f12d6d1a328ff0	2 subnets
public_rt	rtb-015052355dac3ea79	2 subnets

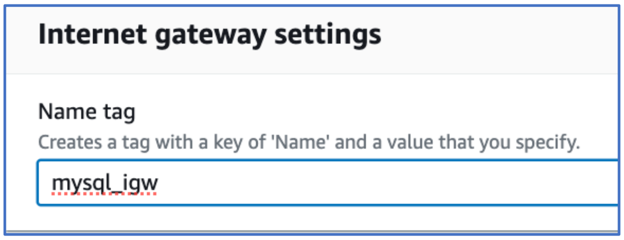
6. Create internet traffic to our public subnet with an Internet Gateway

Click ‘Internet gateways’ on the side panel.



There are already default gateway(s). We will create our own. Click ‘Create internet gateway’

Name your gateway.



Internet gateway settings

Name tag

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

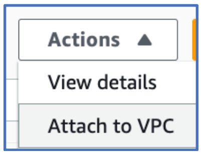
mysql_igw

Leave other defaults. Click ‘Create internet gateway’.

7. Attach the internet gateway to our VPC.

When you click ‘Internet gateways’ again, you will see the new gateway, but it will have the state of ‘Detached’. We need to attach it to our VPC.

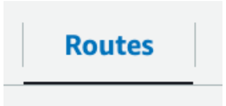
Check the box next to the new internet gateway and under ‘Actions’ click ‘Attach to VPC’.



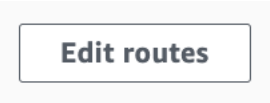
Select our VPC and then click ‘Attach internet gateway’.

8. Add a route to the public routing table to route internet traffic to the internet gateway.

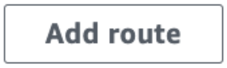
Click the ‘Route tables’ again and check the box next to the public route table. Then choose the ‘Routes’ tab.



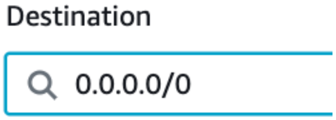
Click ‘Edit routes’



There is a default local destination. We want to add another. Click ‘Add route’.



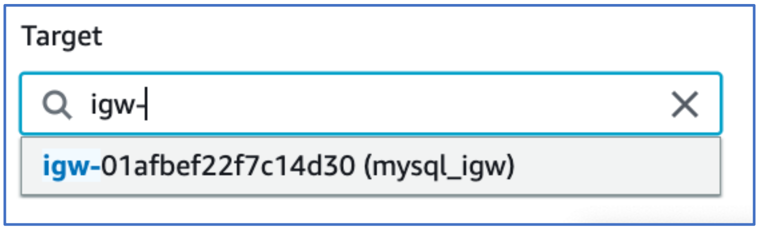
Add destination of 0.0.0.0/0 to allow all traffic to the Internet Gateway



Destination

0.0.0.0/0

And Target ‘Internet Gateway’. Then choose the internet gateway you created earlier.

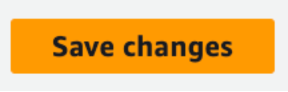


Target

igw-

igw-01afbef22f7c14d30 (mysql_igw)

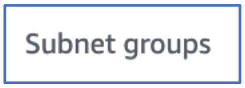
Click ‘Save changes’



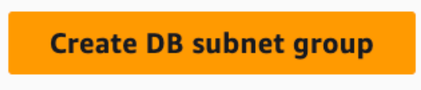
9. Create a Subnet for the Database

Inside our network we will be running a database. That's why we've been naming everything MySQL. We will use the RDS service.

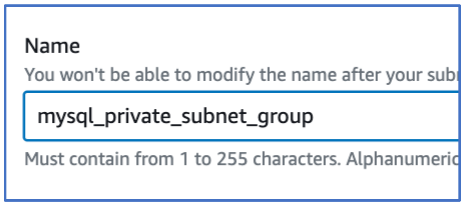
Before we create a database. We need to create a subnet group within RDS. Click 'Subnet groups' on the left.



Click 'Create DB subnet group'



Give it a name.

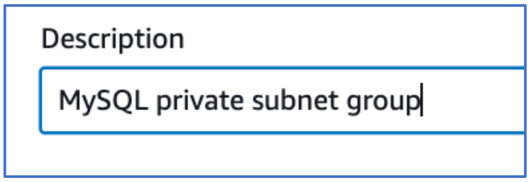

Name

You won't be able to modify the name after your sub

mysql_private_subnet_group

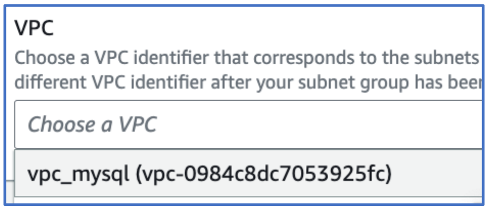
Must contain from 1 to 255 characters. Alphanumeric

Add a description


Description

MySQL private subnet group

Choose our VPC

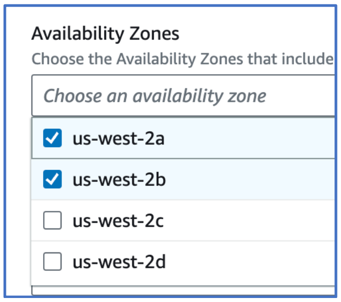

VPC

Choose a VPC identifier that corresponds to the subnets different VPC identifier after your subnet group has bee

Choose a VPC

vpc_mysql (vpc-0984c8dc7053925fc)

Choose the2 AZs we have used.



Availability Zones

Choose the Availability Zones that include

Choose an availability zone

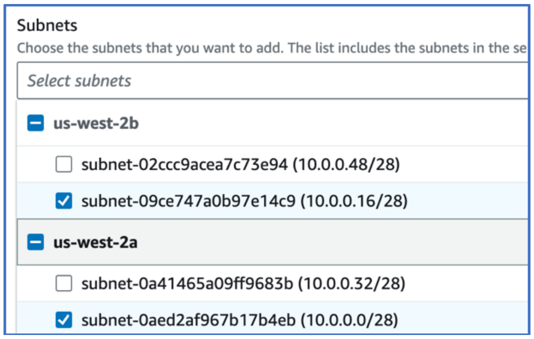
☑ us-west-2a

☑ us-west-2b

☐ us-west-2c

☐ us-west-2d

Select the private subnet from each AZ.
10.0.0.16/28 and 10.0.0.0/28



Subnets

Choose the subnets that you want to add. The list includes the subnets in the se

Select subnets

⊟ **us-west-2b**

☐ subnet-02ccc9acea7c73e94 (10.0.0.48/28)

☑ subnet-09ce747a0b97e14c9 (10.0.0.16/28)

⊟ **us-west-2a**

☐ subnet-0a41465a09ff9683b (10.0.0.32/28)

☑ subnet-0aed2af967b17b4eb (10.0.0.0/28)

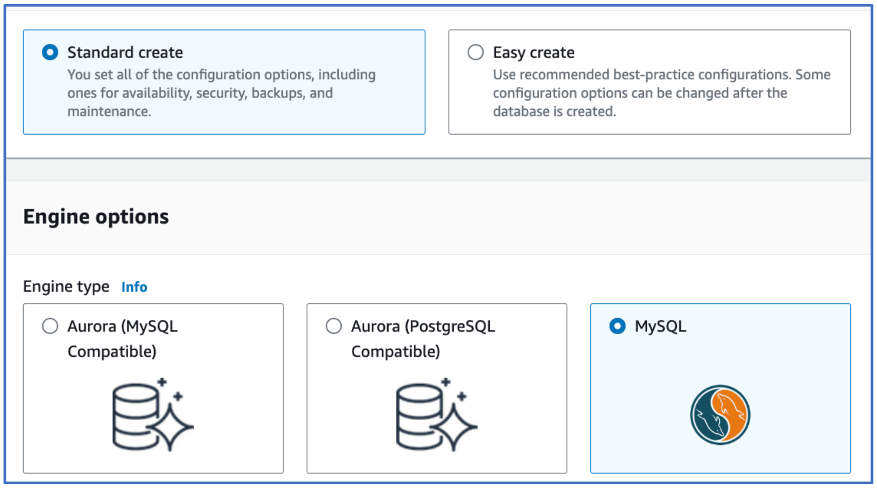
Then click ‘Create’

10. Create a MySQL Database Instance

Now we will create the database. Go back to RDS.

Click ‘Databases’ on the left panel and ‘Create database’

We will choose the ‘Standard create’ and the engine we will use is MySQL.



● 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

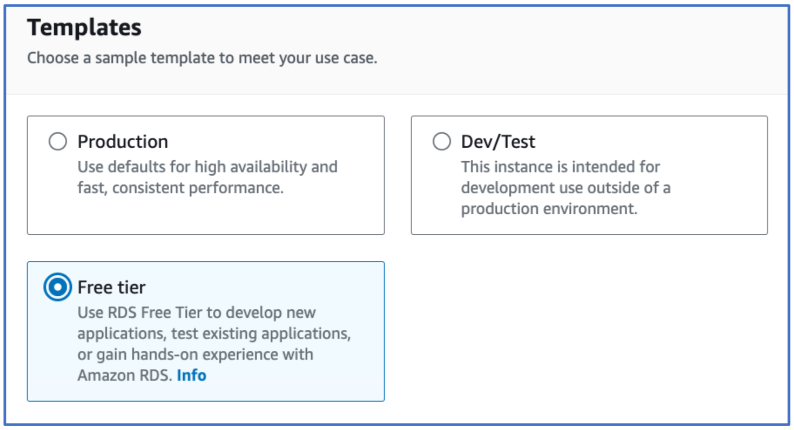
Engine type Info

○ Aurora (MySQL Compatible)

○ Aurora (PostgreSQL Compatible)

● MySQL

Choose ‘Free tier’ under Templates.



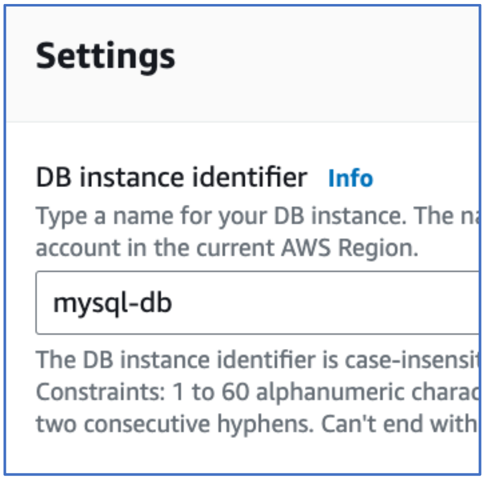
Templates
Choose a sample template to meet your use case.

○ 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

In the ‘Settings’ section. Give it a name of mysql-db and ***use a dash not an underscore here.***



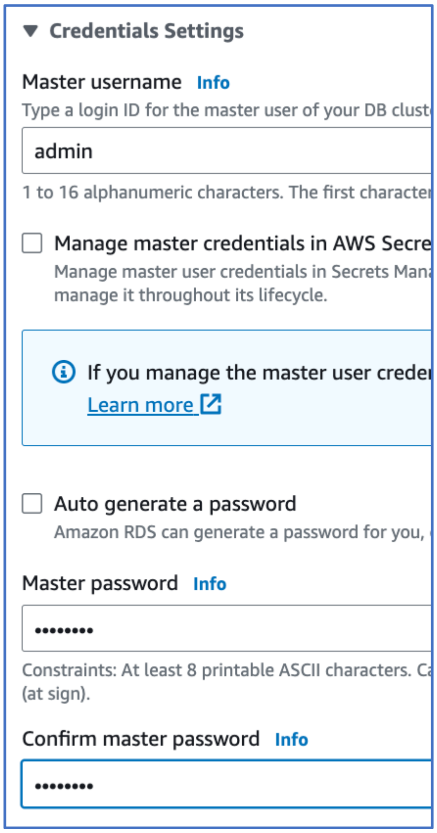
Settings

DB instance identifier Info
Type a name for your DB instance. The na
account in the current AWS Region.

mysql-db

The DB instance identifier is case-insensit
Constraints: 1 to 60 alphanumeric charac
two consecutive hyphens. Can't end with

Add login information. I used 'password' as the password for this lab and left the admin username.



▼ Credentials Settings

Master username Info

Type a login ID for the master user of your DB clust

admin

1 to 16 alphanumeric characters. The first characte

□ Manage master credentials in AWS Secre

Manage master user credentials in Secrets Man
manage it throughout its lifecycle.

ⓘ If you manage the master user crede
Learn more

□ Auto generate a password

Amazon RDS can generate a password for you,

Master password Info

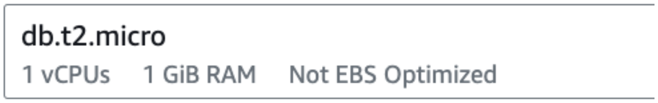
••••••••

Constraints: At least 8 printable ASCII characters. C
(at sign).

Confirm master password Info

••••••••

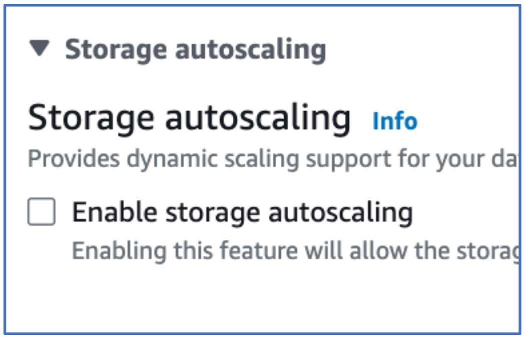
Choose the db.t2.micro.



db.t2.micro

1 vCPUs 1 GiB RAM Not EBS Optimized

Leave other defaults until 'Storage autoscaling' open that section up and uncheck 'Enable storage autoscaling'



▼ Storage autoscaling

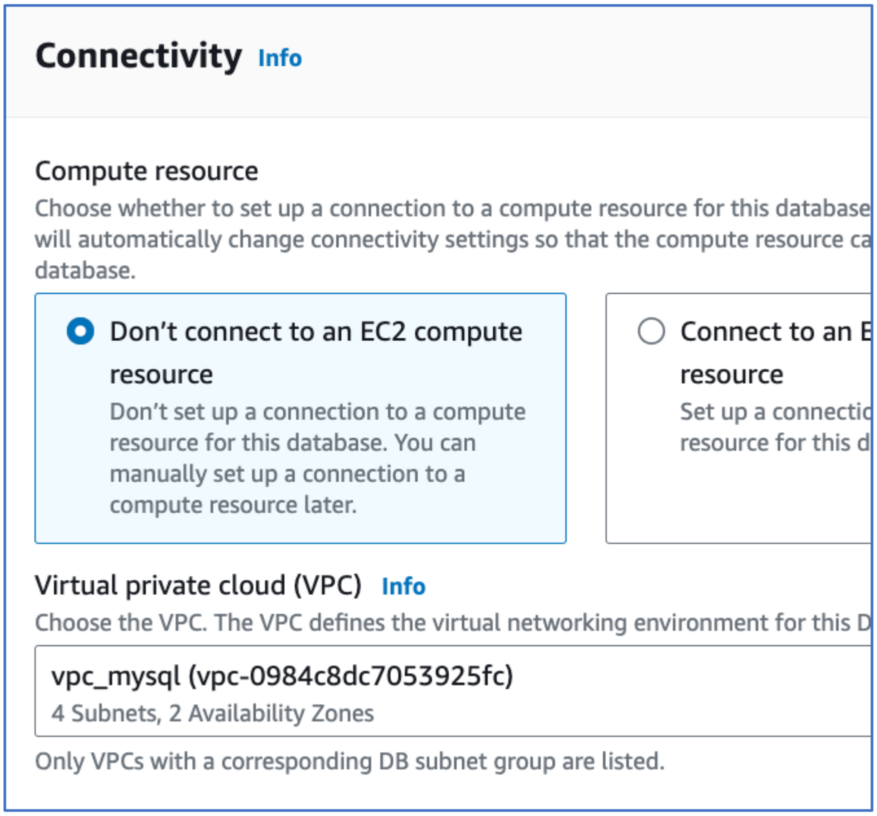
Storage autoscaling Info

Provides dynamic scaling support for your da

□ Enable storage autoscaling

Enabling this feature will allow the storag

In the ‘Connectivity’ section choose our VPC.



Connectivity Info

Compute resource

Choose whether to set up a connection to a compute resource for this database will automatically change connectivity settings so that the compute resource ca 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 E resource
Set up a connectic resource for this d

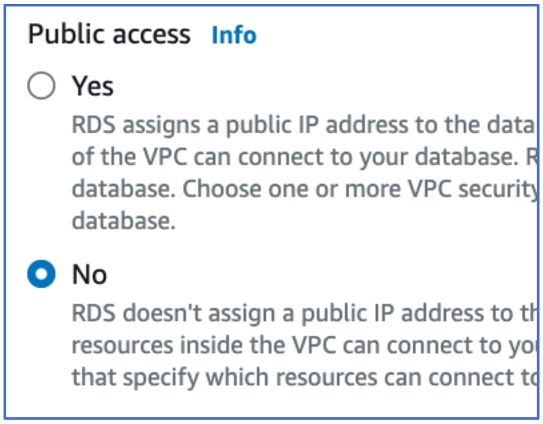
Virtual private cloud (VPC) Info

Choose the VPC. The VPC defines the virtual networking environment for this D

vpc_mysql (vpc-0984c8dc7053925fc)
4 Subnets, 2 Availability Zones

Only VPCs with a corresponding DB subnet group are listed.

We don’t want Public access, so leave it No.

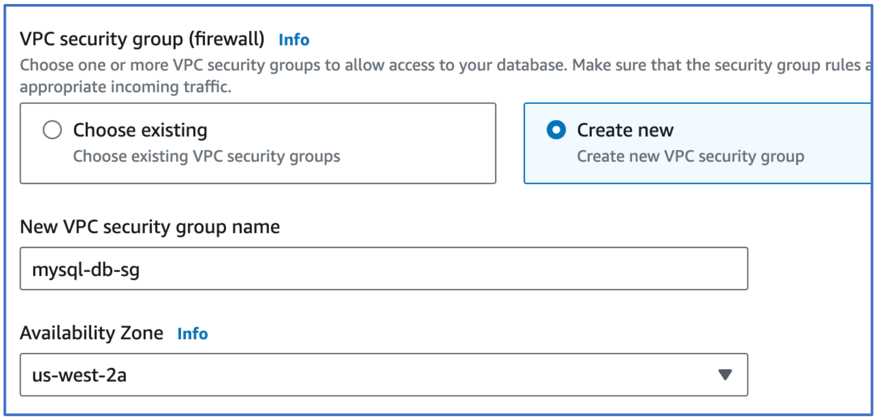


Public access Info

○ Yes
RDS assigns a public IP address to the data of the VPC can connect to your database. R database. Choose one or more VPC security database.

● No
RDS doesn't assign a public IP address to th resources inside the VPC can connect to yo that specify which resources can connect to

Create a new security group and name it. This is a firewall for our database. You use security groups to control traffic at the instance level and NACL (Network access control lists) to control access at the subnet level. Choose the AZ.


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 a appropriate incoming traffic.

○ Choose existing
Choose existing VPC security groups

● Create new
Create new VPC security group

New VPC security group name

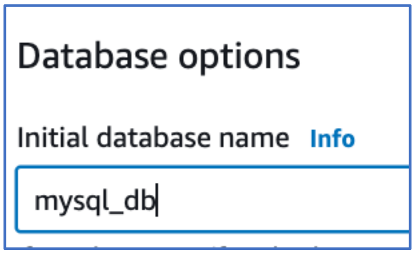
mysql-db-sg

Availability Zone Info

us-west-2a

Leave the password authentication.

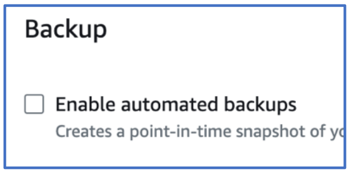
Under ‘Additional configuration’ give the database a name.


Database options

Initial database name Info

mysql_db

Uncheck enable automated backups.


Backup

□ Enable automated backups
Creates a point-in-time snapshot of yo

Leave the rest on defaults and click ‘Create database’.

If it suggests add-ons, just close that window.

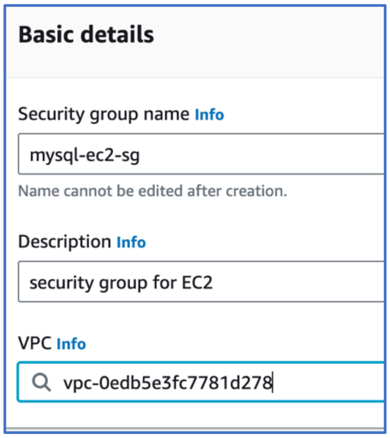
Wait for the database to set up. Do step 11 while you wait.

11. Set up a security group for our EC2.

Go to the EC2 service and click ‘Security groups’ on left side (scroll down a bit)

Click ‘Create security group’

Give it a name, description, and choose our VPC. You might have to delete the default VPC to see our VPC option. Choose our VPC.



Basic details

Security group name Info

mysql-ec2-sg

Name cannot be edited after creation.

Description Info

security group for EC2

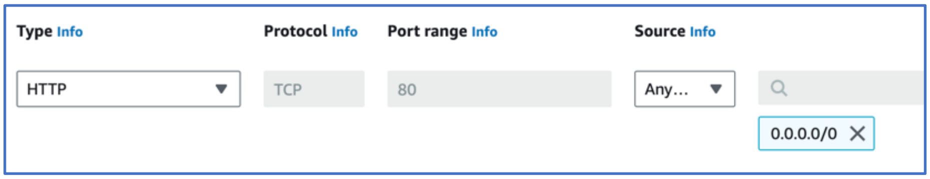
VPC Info

vpc-0edb5e3fc7781d278

Click ‘Add rule’ under Inbound rules.

Add two rules as follows:

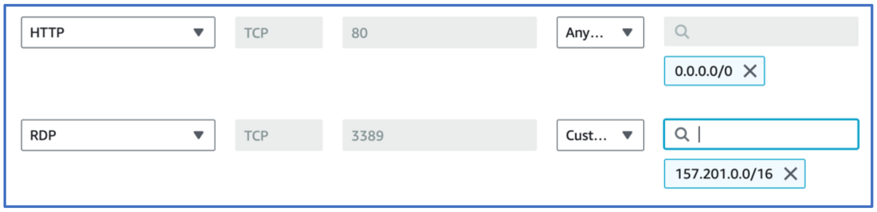
Should be:
Type info – http
Protocol info - leave TCP
Port range - leave 80
Source info – 0.0.0.0/0



Type Info	Protocol Info	Port range Info	Source Info	
HTTP	TCP	80	Any...	0.0.0.0/0

Add another inbound rule with the ‘Add rule’ button.

Type info – RDP
Protocol info - leave TCP
Port – leave default 3389
Source info - custom BYUI 157.201.0.0/16

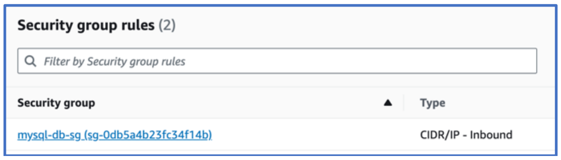


HTTP	TCP	80	Any...	0.0.0.0/0
RDP	TCP	3389	Cust...	157.201.0.0/16

Click ‘Create security group’

12. Edit the inbound rules for the database

Go back to RDS and click ‘Databases’ and check that your database finished.
Click on the database name.
Scroll down and click on inbound security groups



Security group rules (2)

Filter by Security group rules

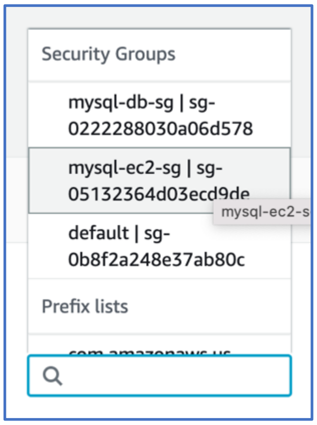
Security group	Type
mysql-db-sg (sg-0db5a4b23fc34f14b)	CIDR/IP - Inbound

Click on the inbound rule. Click on the inbound tab. Click ‘edit inbound rules’ button.

Add an inbound rule

Type info – MYSQL/Aurora
Protocol – leave TCP
Port range – leave 3306
Source info – custom

Choose the security group you had for your ec2 security group (You might have to close the default that is there and scroll down a bit in your choices to see it.)



Security Groups

mysql-db-sg | sg-0222288030a06d578

mysql-ec2-sg | sg-05132364d03ecd9de

default | sg-0b8f2a248e37ab80c

Prefix lists

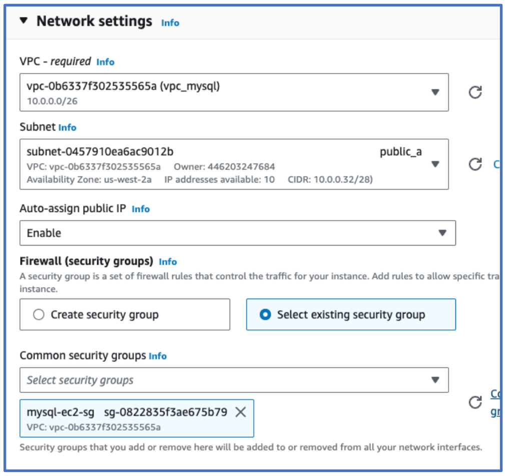
Click 'Save rules'

13. Create a Windows server

Create a windows server again like Lab B but
Use t3a.large (step 5 in lab b)

And Edit 'Network Settings':

Choose our VPC
Make sure the subnet is public not private
Enable Auto-assign public IP
Select existing security group and find our EC2 security group 'mysql-ec2-sg'



▼ **Network settings** Info

VPC - *required* Info
vpc-0b6337f302535565a (vpc_mysql)
10.0.0.0/26

Subnet Info
subnet-0457910ea6ac9012b public_a
VPC: vpc-0b6337f302535565a Owner: 446203247684
Availability Zone: us-west-2a IP addresses available: 10 CIDR: 10.0.0.32/28)

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 tra instance.

○ Create security group ● Select existing security group

Common security groups Info
Select security groups

mysql-ec2-sg sg-0822835f3ae675b79
VPC: vpc-0b6337f302535565a

Security groups that you add or remove here will be added to or removed from all your network interfaces.

Launch the instance. Your instance might take a few minutes to run and have 2/2 checks before you can connect with the next window. Once it’s done, check the box next to the instance and connect. Use the RDP tab and download remote desktop file. Double click the .rdp file that downloaded.

Use the ‘Get password’ to get the remote desktop password from the .pem file.

You could now download Workbench and connect to the database by using the EC2 end point as the host.