Project: Migrating data from a database MySQL on Amazon EC2 to a database MySQL amazon RDS

Scenario

An e-commerce company currently uses a relational database
MySQL hosted on an unmanaged EC2 instance to manage its transactions and store the customer data. This solution worked well when the company was launched, but
with the rapid growth in the number of users and the increase in transaction volume,
The technical team faces several challenges, including:

- Difficulty managing regular backups and restores .
- Lack of high availability and disaster recovery.
- Unplanned downtime and complex maintenance.
- Performance issues related to scalability.

The company decides to call on you as an AWS data engineer to migrate this database MySQL data to a more robust, fully managed solution.

Objective

Migrate MySQL database from EC2 instance to Amazon RDS MySQL while
minimizing downtime and ensuring business continuity.

Costs:

Export date: **28/08/2024 00:34:52** Language: **English**

Estimate URL: **https://calculator.aws/#/estimate?id=6dd80ea09d071be69be26447c1b2a673e3441999**

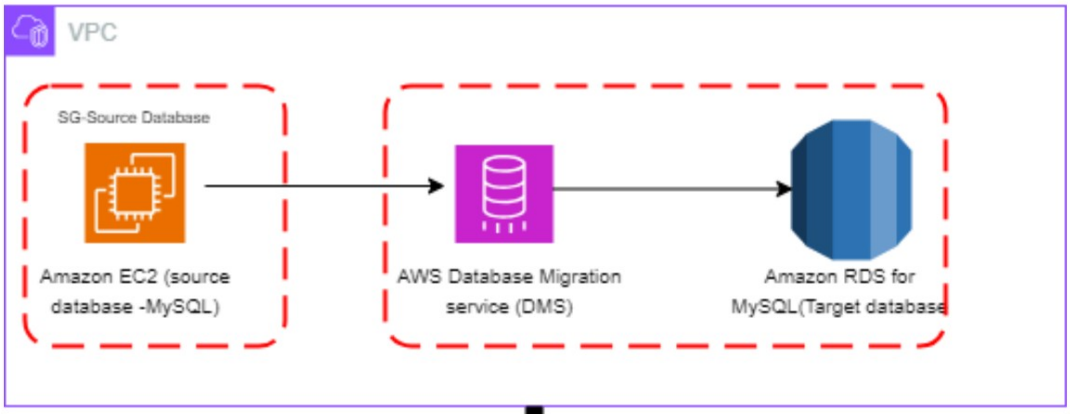
Estimate summary

Upfront cost	Monthly cost	Total 12 months cost
0.00 USD	82.38 USD	988.56 USD Includes upfront cost

Estimated time of completion: 2h

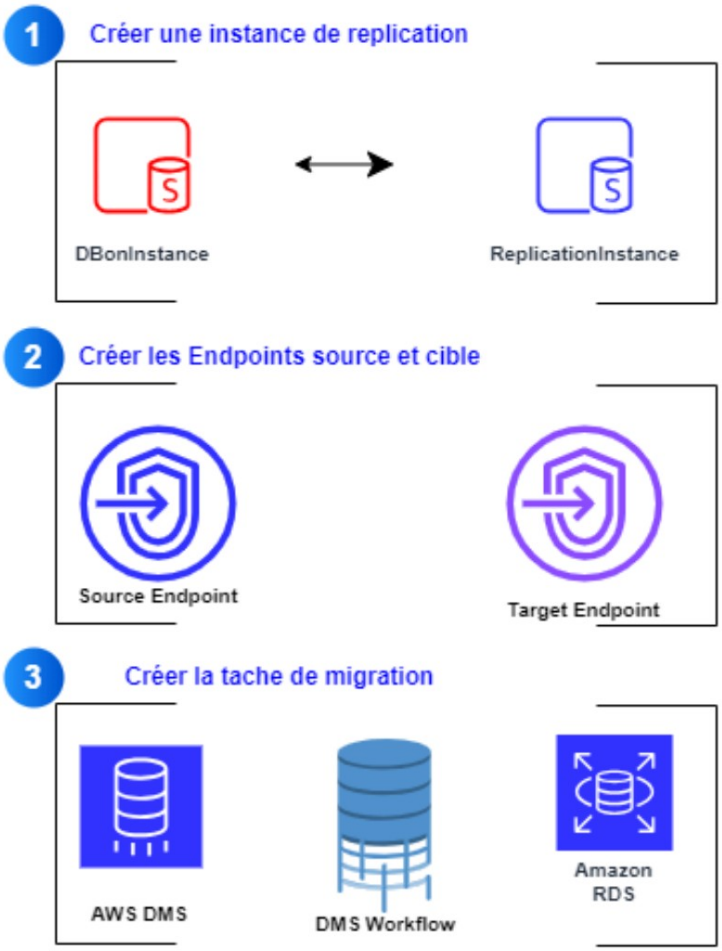
Solution architecture

Architecture de solution Database Migration



Workflow

AWS Database Migration Service



Implementing the solution

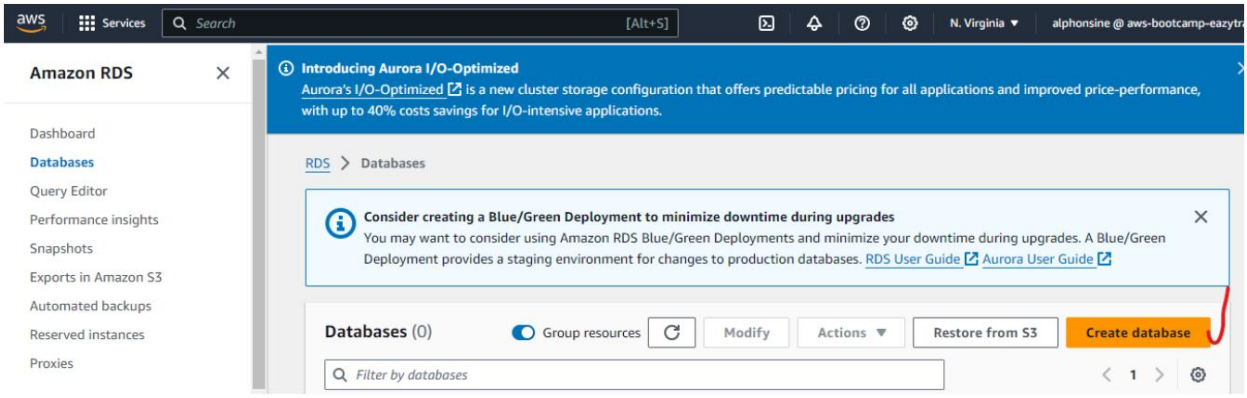
Step 1: Create an unmanaged EC2 database server and install mysql.

And which will be our source database.

Step 2: Create the target database on the Amazon RDS console

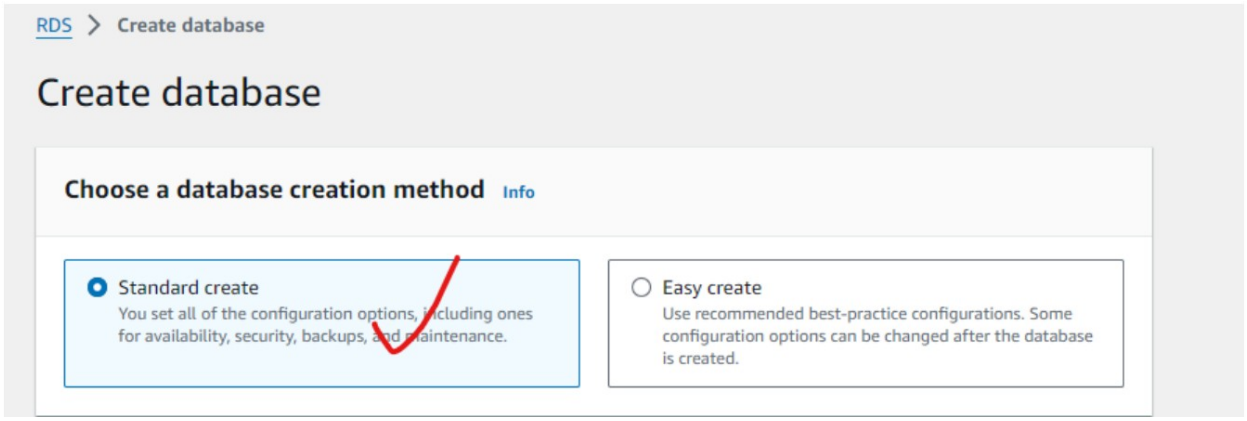
In this step, you create a MySQL database instance in Amazon RDS. This instance will be used as the primary database once you copy your existing data using AWS DMS.

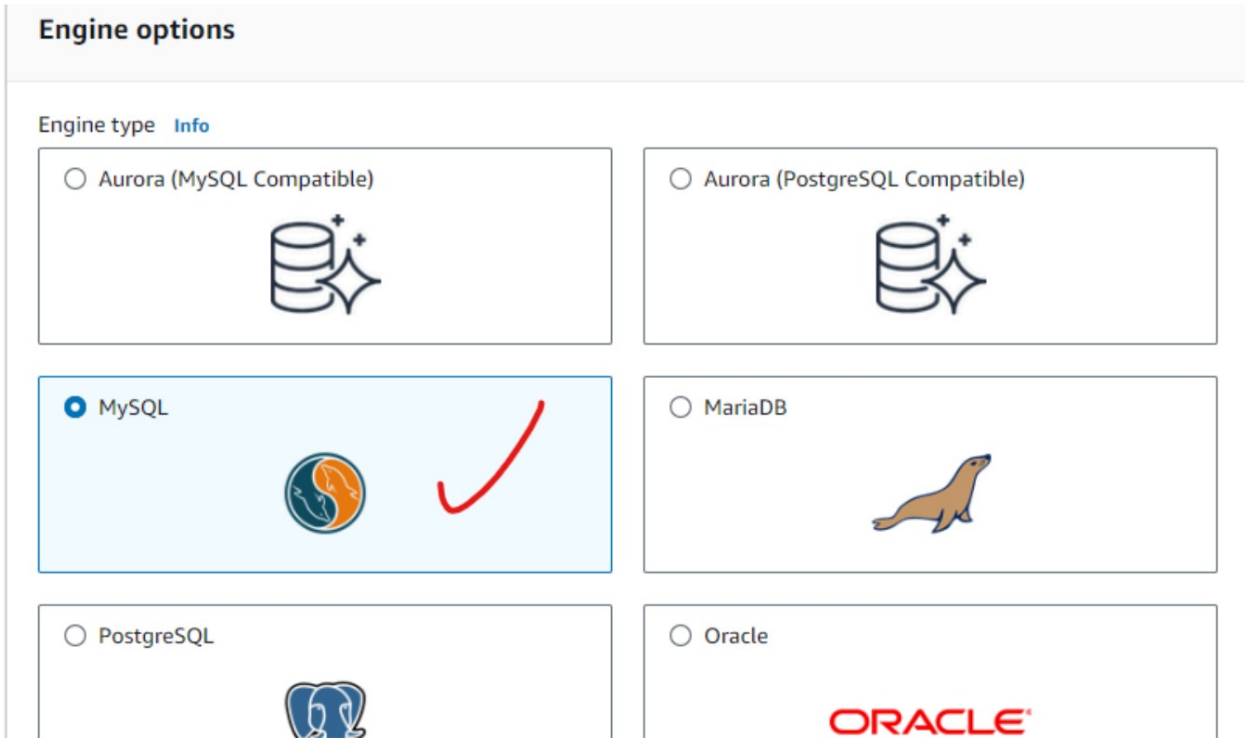
- To get started, go to the Amazon RDS console. On the main page, select Create Database to create a new database.

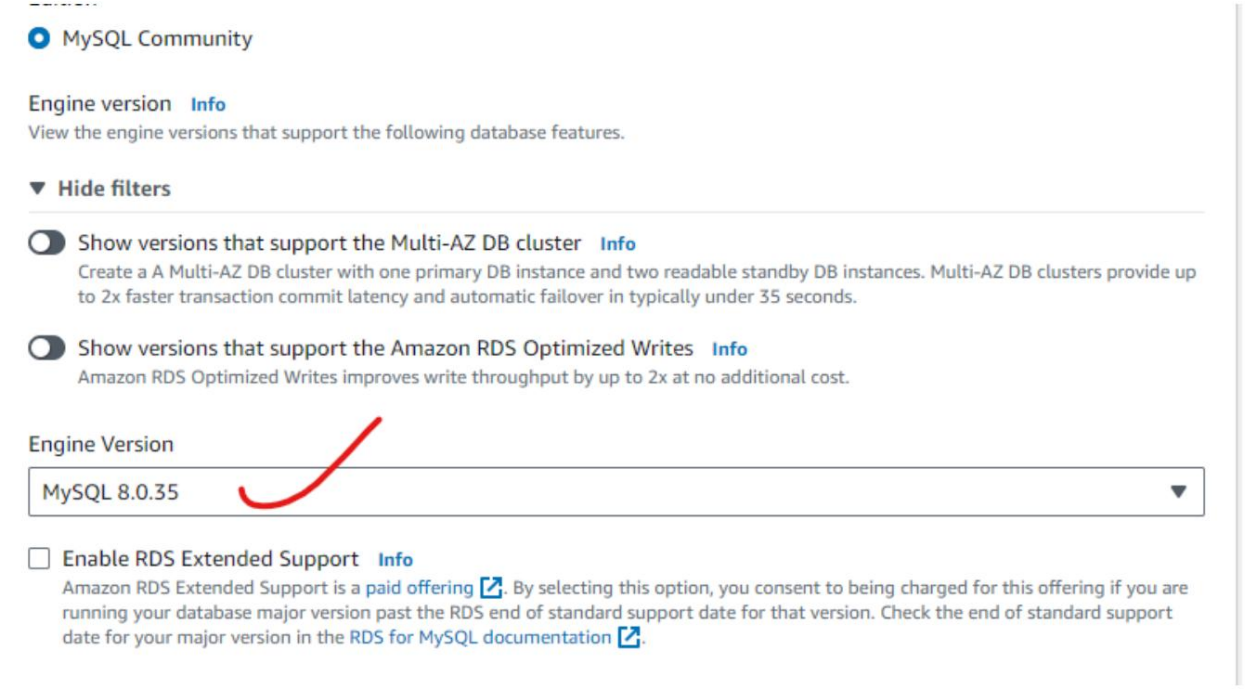


The database creation wizard is then launched. In the Engine Options section, Choose MySQL as the engine type. Then choose the version of MySQL that you want to wish to use.

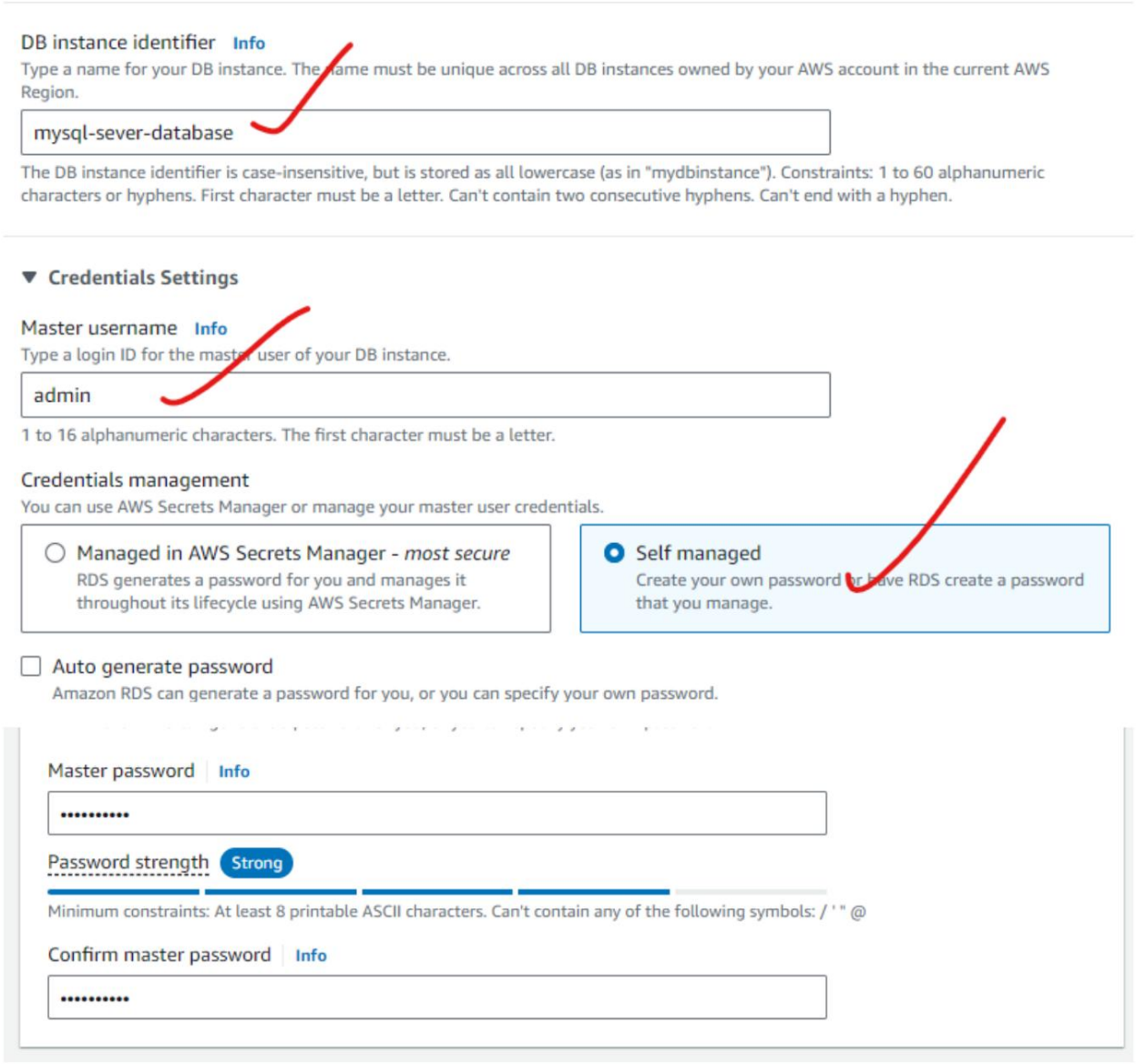
- The database creation wizard includes templates that make it easier to Configuring your Amazon RDS database settings. If you create this database to use it in production, you must choose the production model. For our case, we will choose free tier.



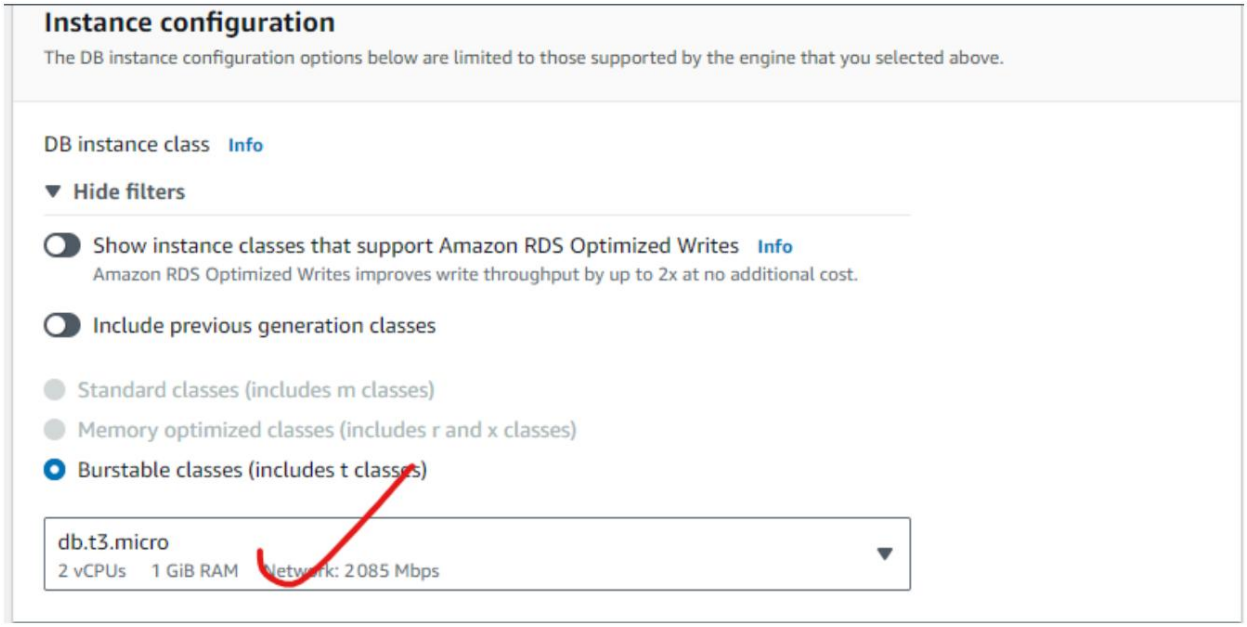




- In the Settings section, give your database a name and set the name username and master password. Do not generate your password automatically for this lab, and make sure to write down your password. You You will need this password to connect to your database and create additional users.

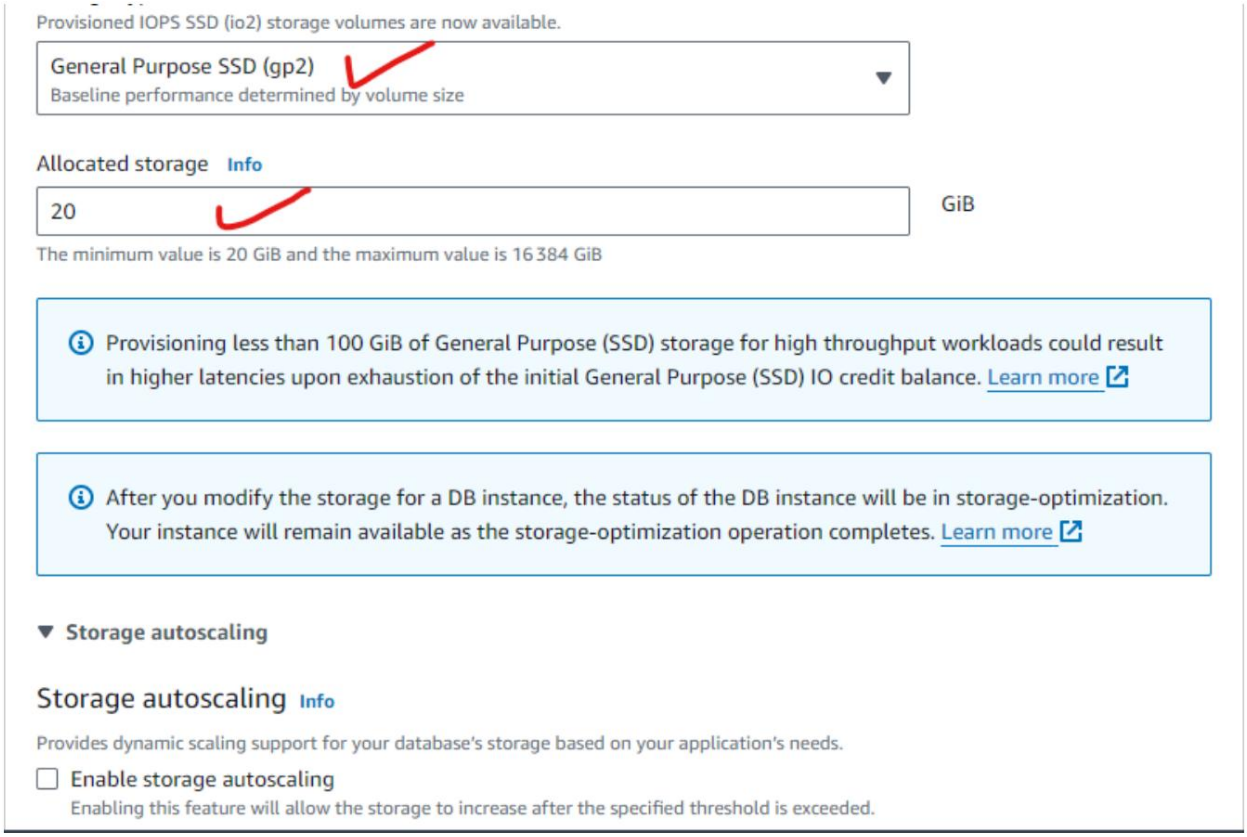


- Then choose the size of the database instance. You must make this choice depending on your estimated capacity. If you manage your own database on Amazon Elastic Compute Cloud (Amazon EC2), you can compare the size of your current Amazon EC2 instance to that of an Amazon RDS instance.

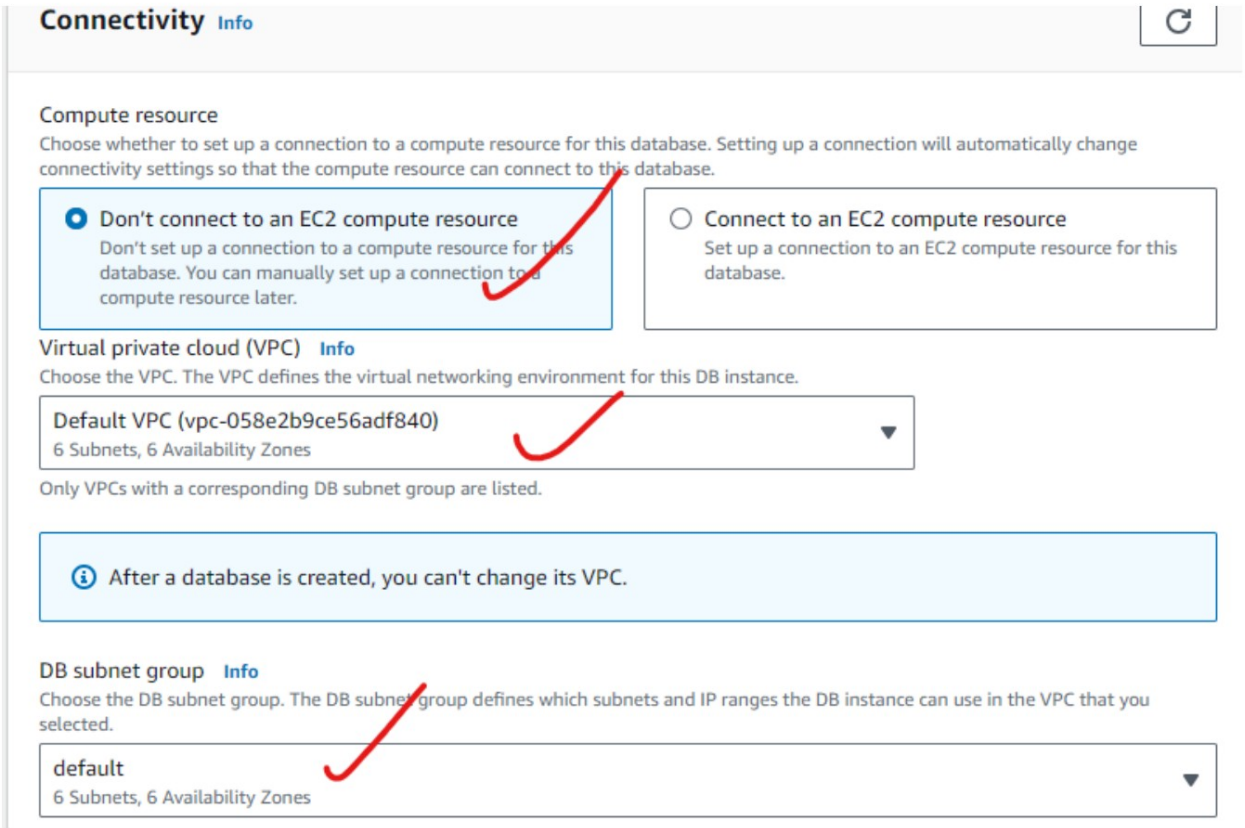


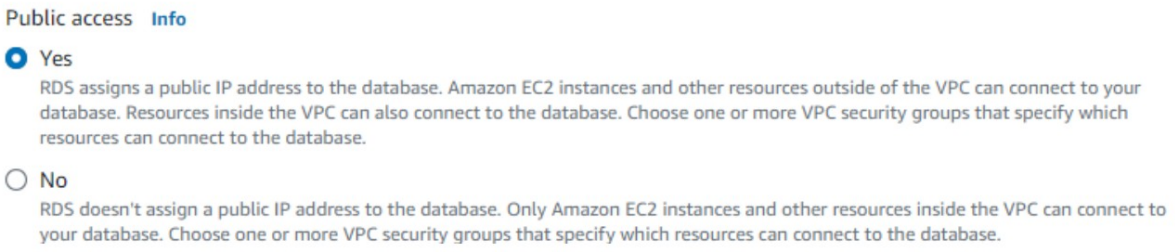
- Next, configure storage options for your Amazon RDS database. It

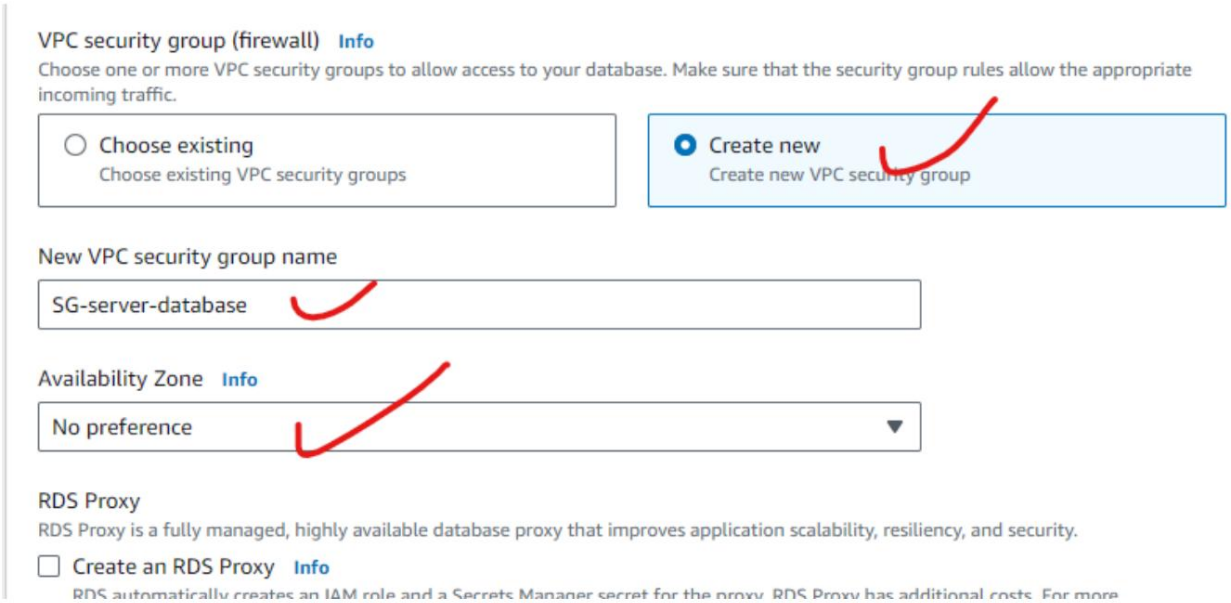
  There are two storage options in Amazon RDS: general purpose storage and Provisioned storage IOPS, or I/O operations per second. With provisioned storage Typically, you receive 3 IOPS per gigabyte of allocated storage. So, 100 gigabytes of storage corresponds to 300 IOPS. In addition, you benefit from burst capacity up to 3,000 IOPS.

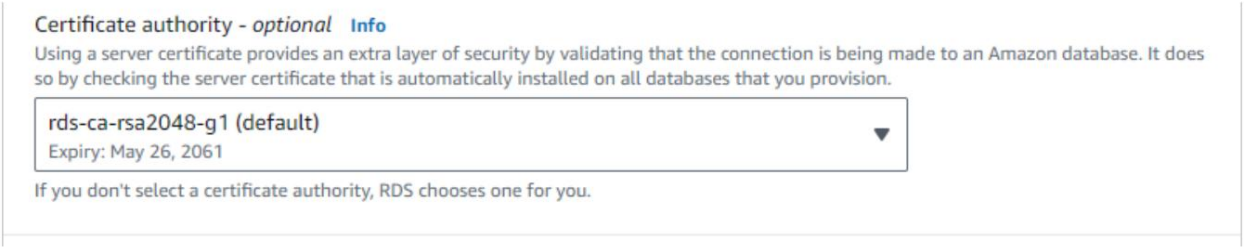


- The next section of the Amazon RDS Database Creation Wizard is about connectivity. You must specify the Amazon Virtual Private Cloud (Amazon VPC) in which your database resides, as well as the network subnet and groups of security for your database instance.
- If you are migrating from a self-managed database instance on Amazon EC2, you can use the same **Amazon VPC and security groups** as your home of existing data.
- If you are migrating from a database that is not hosted on AWS but your application is hosted on AWS, choose the same **Amazon VPC as the one used for your application.** Then create a new security group for your database instance.









- You can also configure additional options, including settings for backups, monitoring, maintenance and upgrades automated. The default settings are suitable for most situations, but you should check them to make sure they meet your needs.

KMS key ID

e9938378-fdc3-4399-b868-f9e224ff0b13

⚠ You can't change the KMS key after enabling Performance Insights.

▶ **Additional configuration**
Enhanced Monitoring

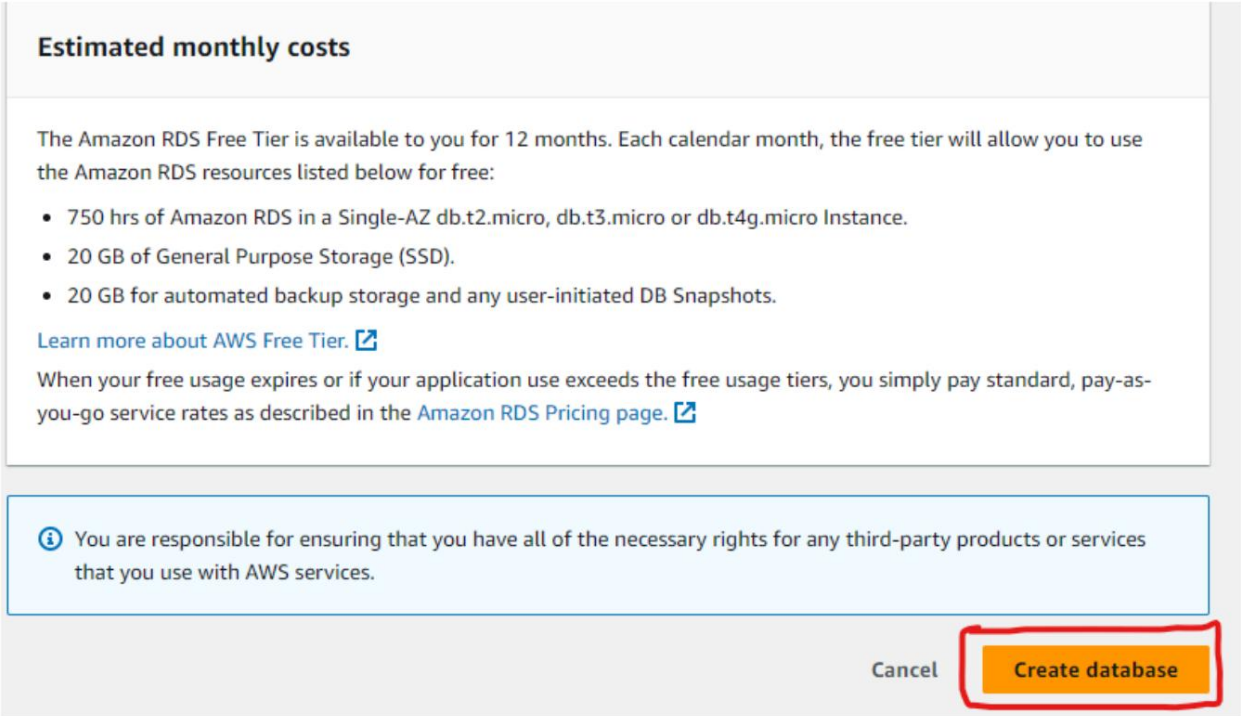
▶ **Additional configuration**
Database options, encryption turned on, backup turned on, backtrack turned off, maintenance, CloudWatch Logs, delete protection turned on.

Estimated Monthly costs

DB instance	12.41 USD
Storage	2.30 USD
Total	**14.71 USD**

This billing estimate is based on on-demand usage as described in Amazon RDS Pricing. Estimate does not include costs for backup storage, IOs (if applicable), or data transfer.

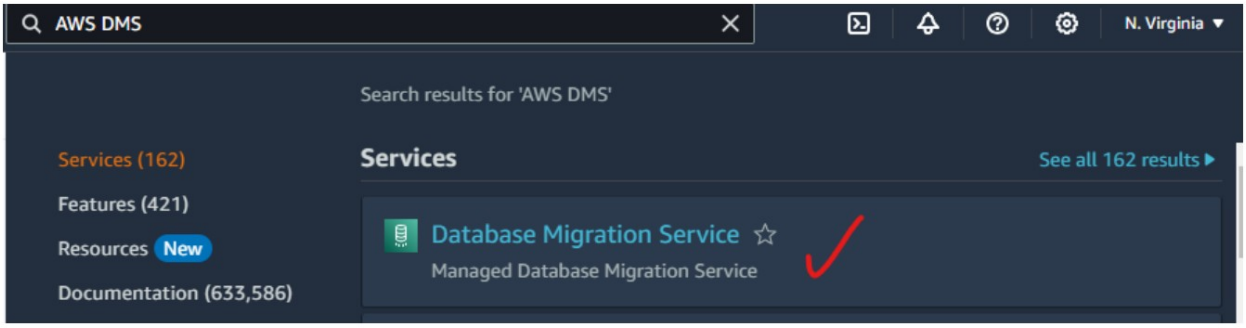
Estimate your monthly costs for the DB Instance using the AWS Simple Monthly Calculator.

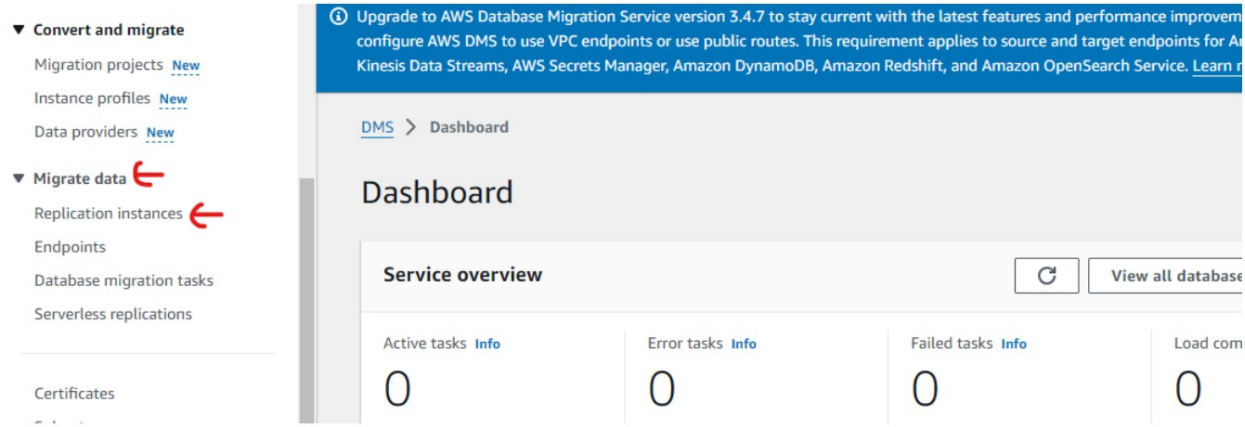


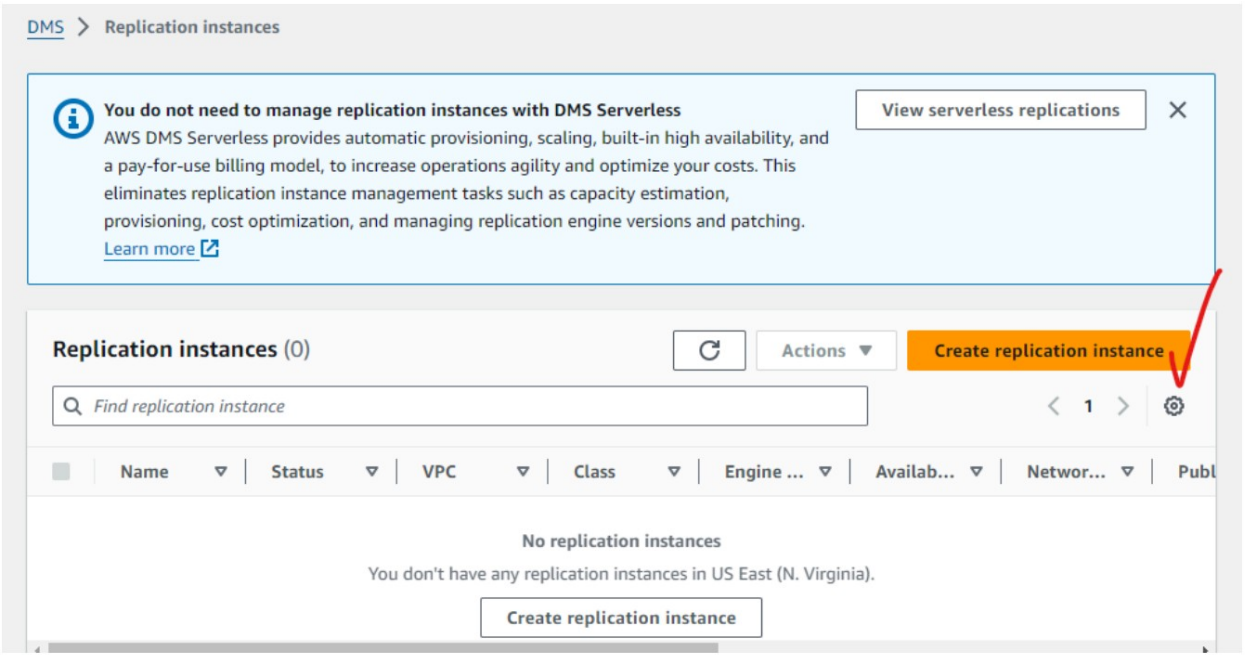
Step 3: Create a replication instance in AWS Database Migration Service (AWS DMS)

In this step, you create a replication instance in AWS DMS.

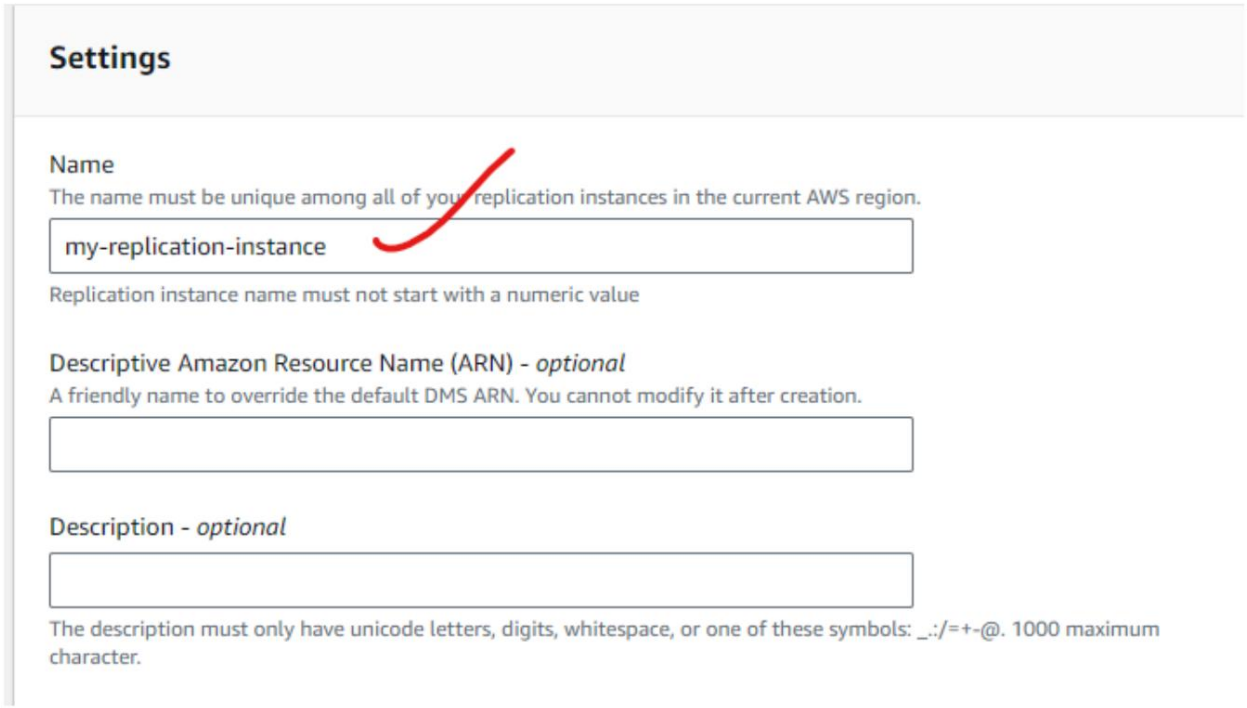
- **AWS DMS** is a service that copies data from an existing database to a fully managed database on AWS. **A replication instance** is a Amazon EC2 instance that can host replication tasks in AWS DMS. In The next step you will configure a replication task.
- To create a replication instance, go to the Replication Instances section of the AWS DMS console. Choose Create Replication Instance to launch the wizard of creating a replication instance.

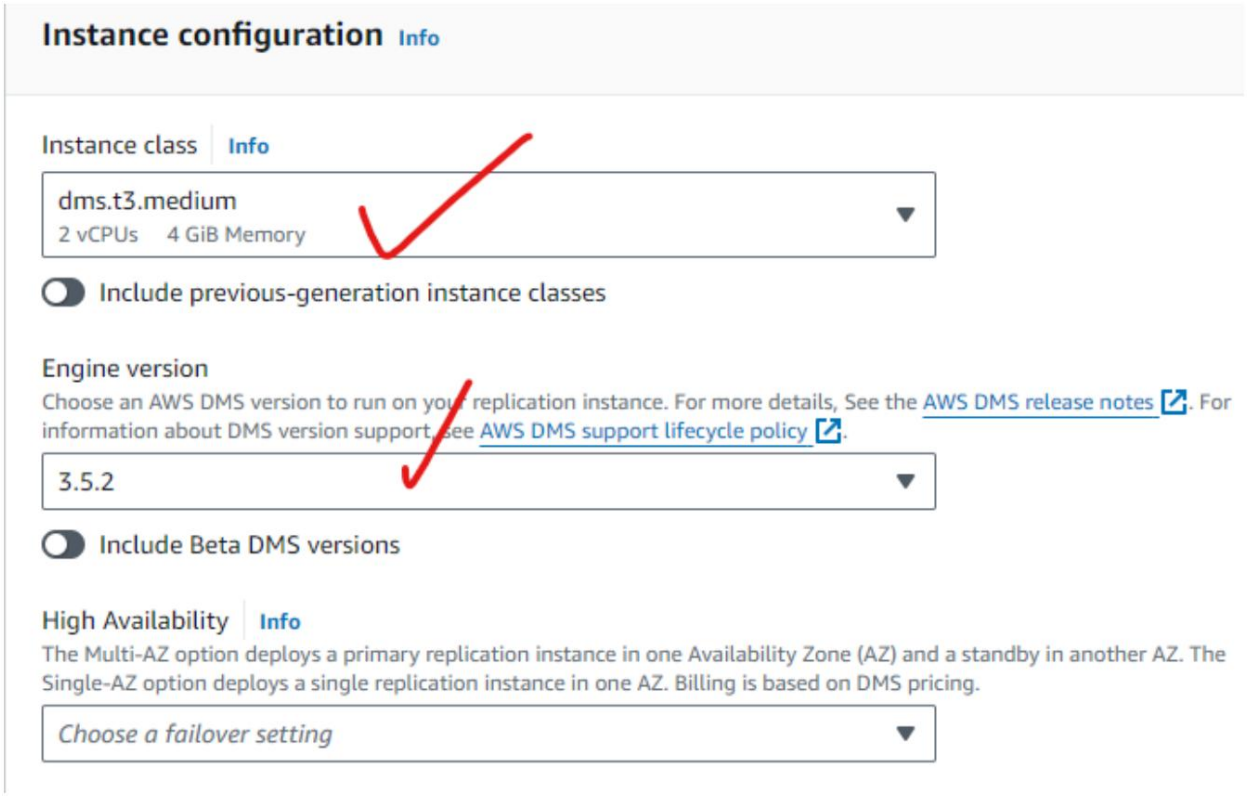




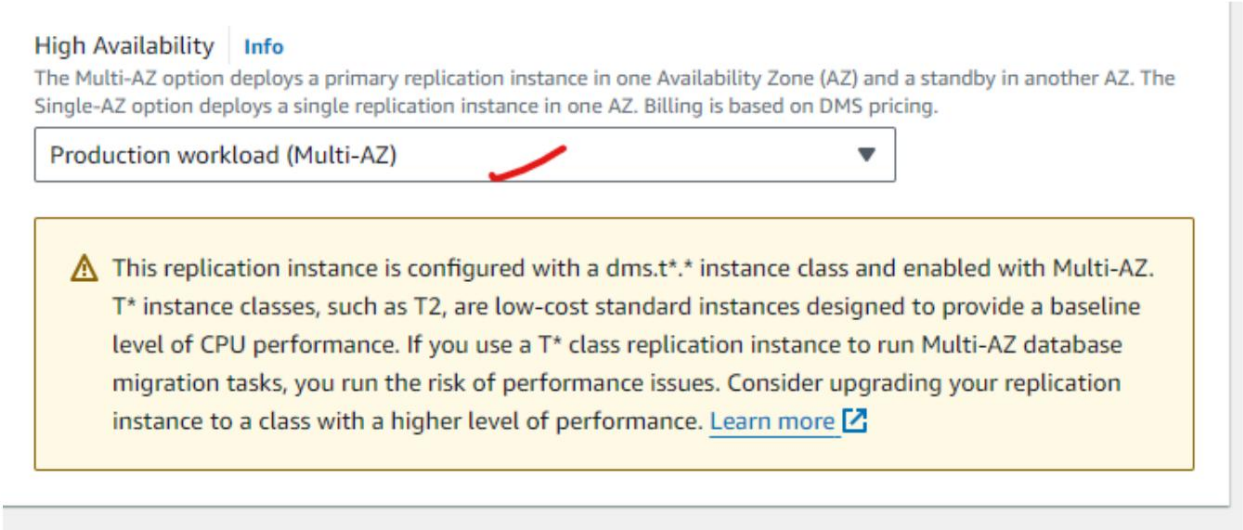


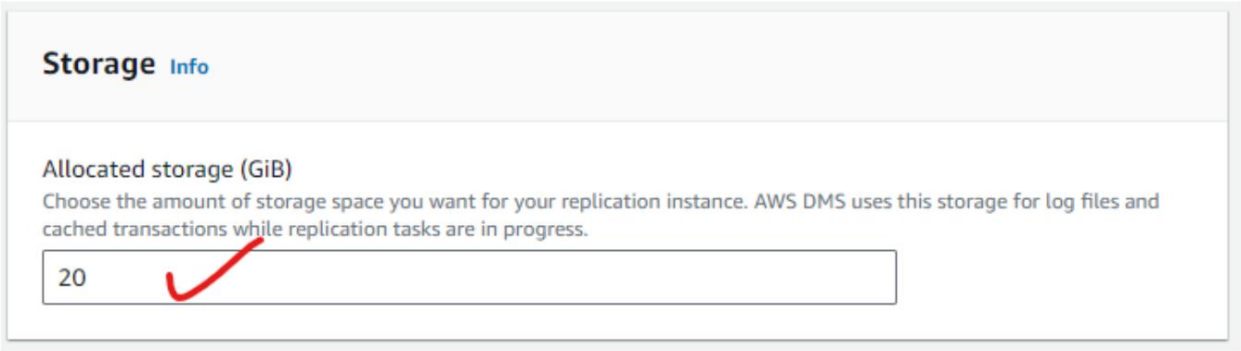
• In the Replication Instance Configuration section, provide a name and a description to your replication instance. Then choose your instance class. The The instance class you use depends on the size of your existing database and the amount of data circulating there.



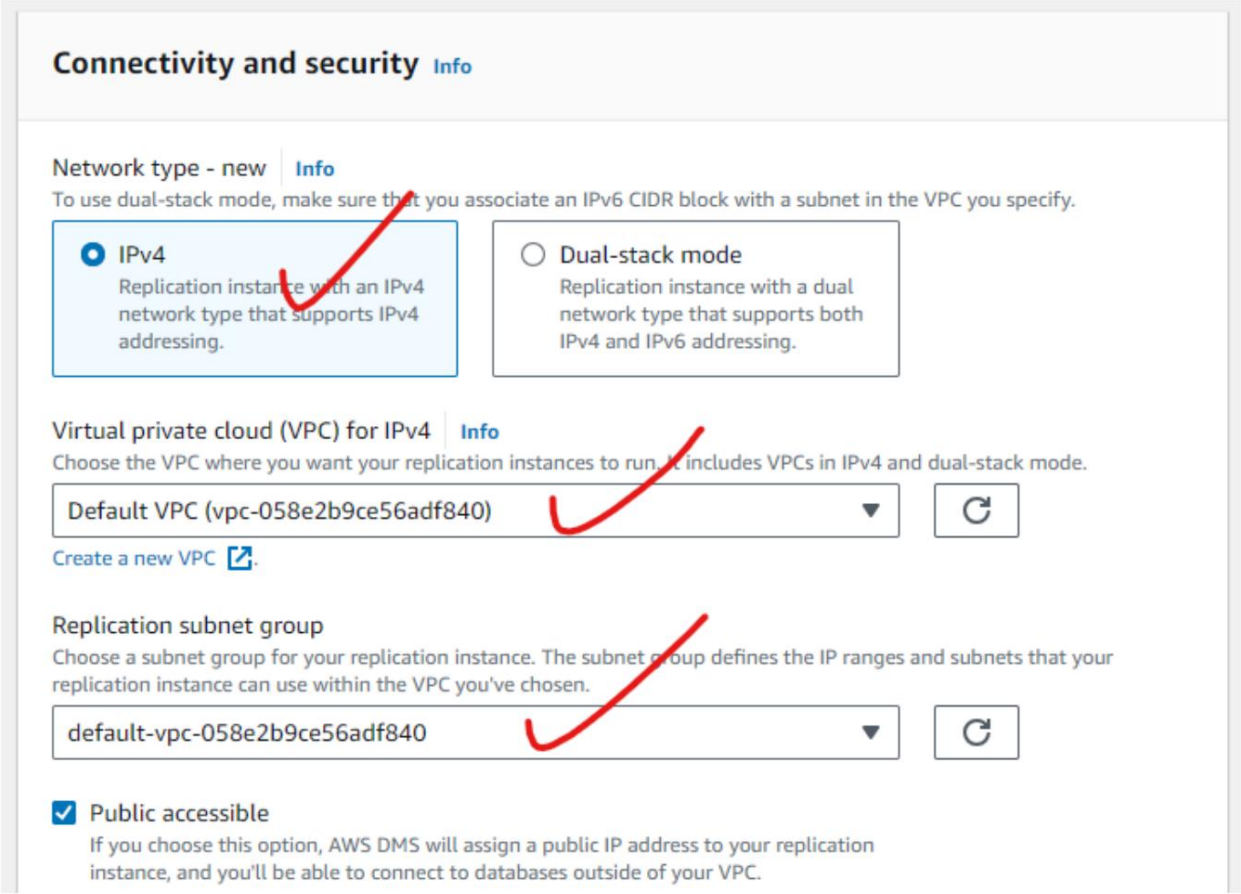


- In the Replication Instance Configuration section, choose a Amazon VPC for your replication instance. Choose the same Amazon VPC in which you have provisioned your Amazon RDS database to facilitate access to the network for the replication instance.
- You can choose a Multi-AZ configuration for your replication instance at redundancy purposes. If you are using AWS DMS to maintain synchronization of two databases over a long period of time, you can opt for a configuration Multi-AZ. If you are performing a one-time migration of your data from a database existing database to a fully managed database in Amazon RDS, you don't have to probably don't need a Multi-AZ setup.
- Finally, choose whether your replication instance should be publicly accessible. If your Existing database is in the same Amazon VPC as your new database data and your replication instance, it is not necessary for your replication instance to replication is publicly accessible. Otherwise, you must make your publicly accessible replication instance.



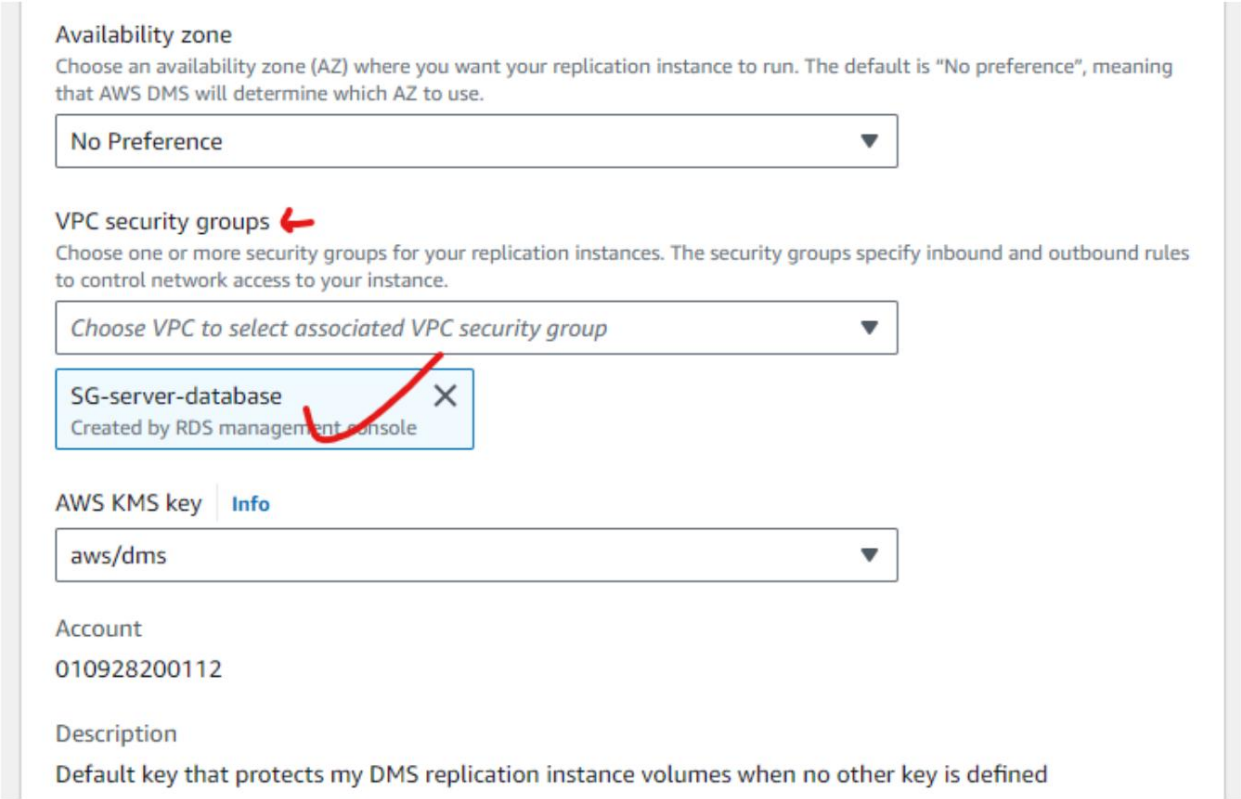


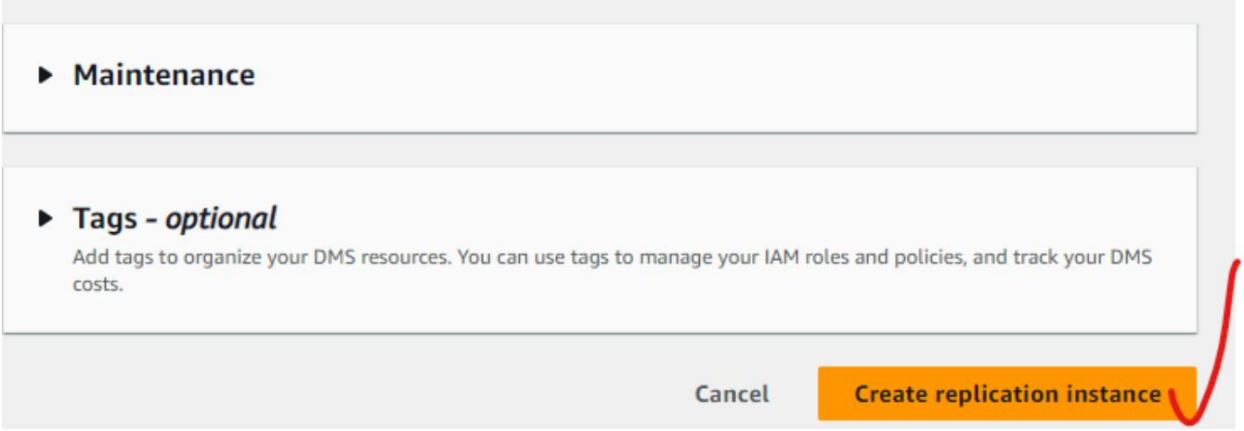
- Next, open the Advanced Security and Network Configuration section. For the VPC security groups configuration, choose the same security group as the one you attached to your Amazon RDS database. This allows your replication instance to access your Amazon RDS database.



You can also change maintenance and tag settings.

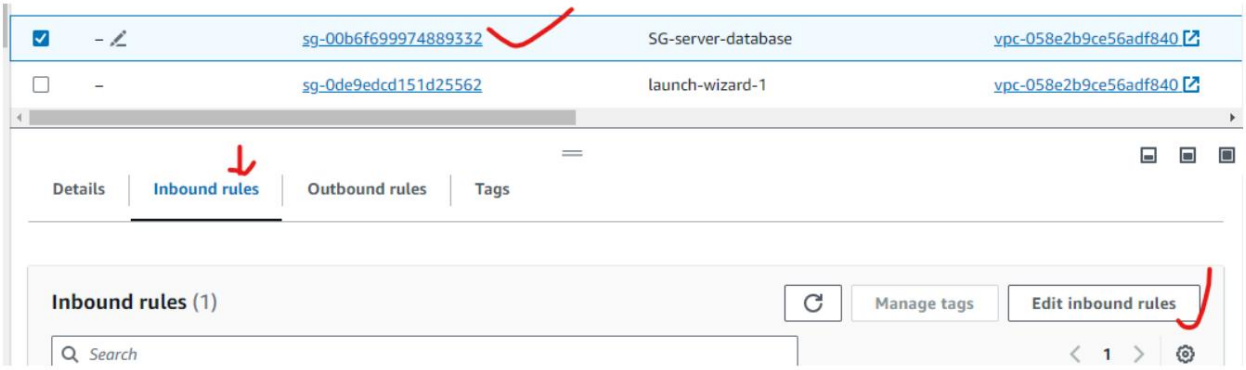
- When you are ready, click Create to create your replication instance in AWS DMS.



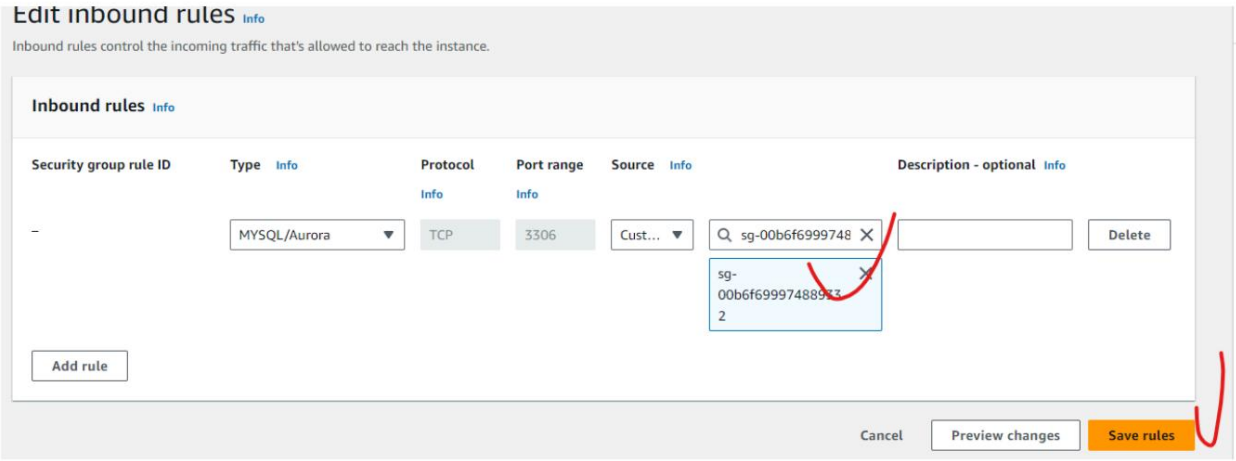


- While you wait for your replication instance to become available, go to the **Security Groups** section in the Amazon EC2 console. You must add a rule to your security group to allow your replication instance to access your database.

- In the **Security Groups section,** find the security group you have attached to your MySQL database instance and your replication instance, and select it.
- Select Edit receiving rules for your security group.



- Your security group has an existing rule that allows access to your instance MySQL from the IP address you used to create the database.
  Delete the existing IP address and enter the name of the security group used for your Amazon RDS database instance and your replication instance.
- Select Save Rules to save the updated rules for your security group.



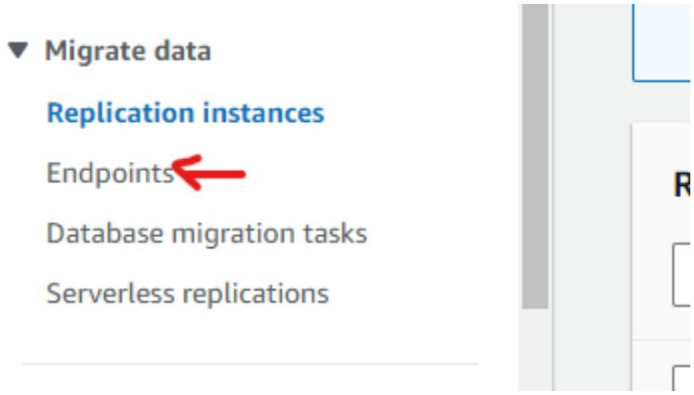
- When your replication instance is available and you have updated the rules of your security group, you can proceed to the next step.

Step 4: Create Source and Target Endpoints for your Database

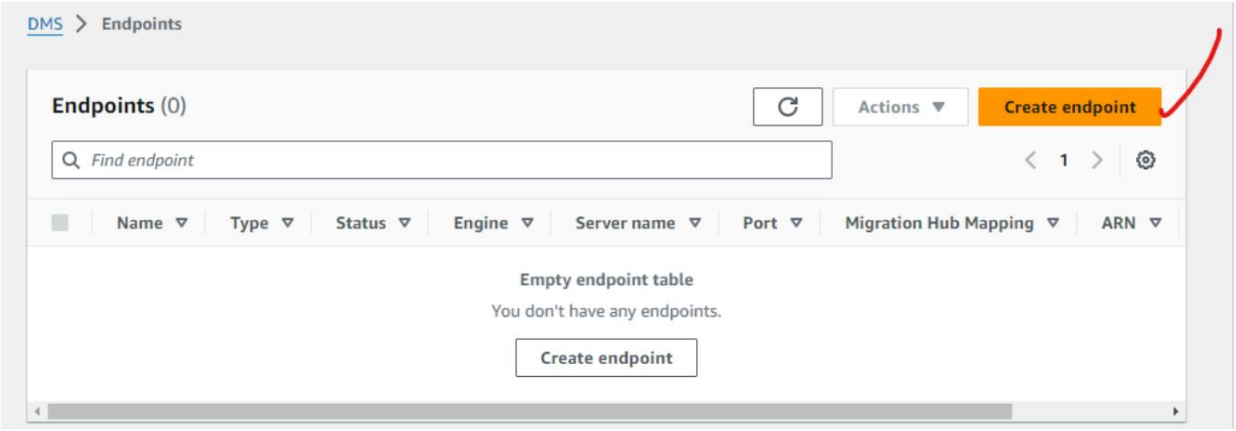
In this you create source and target endpoints for a replication task in AWS DMS.

A replication task involves migrating data from one database to another at Using AWS DMS. Before you can create a replication task, you must save replication points. endpoint for your source and target databases. An endpoint describes the address login details, credentials and other information needed to connect to a database.

- First, we will create the endpoint for your target database. This is the database you created in Amazon RDS.



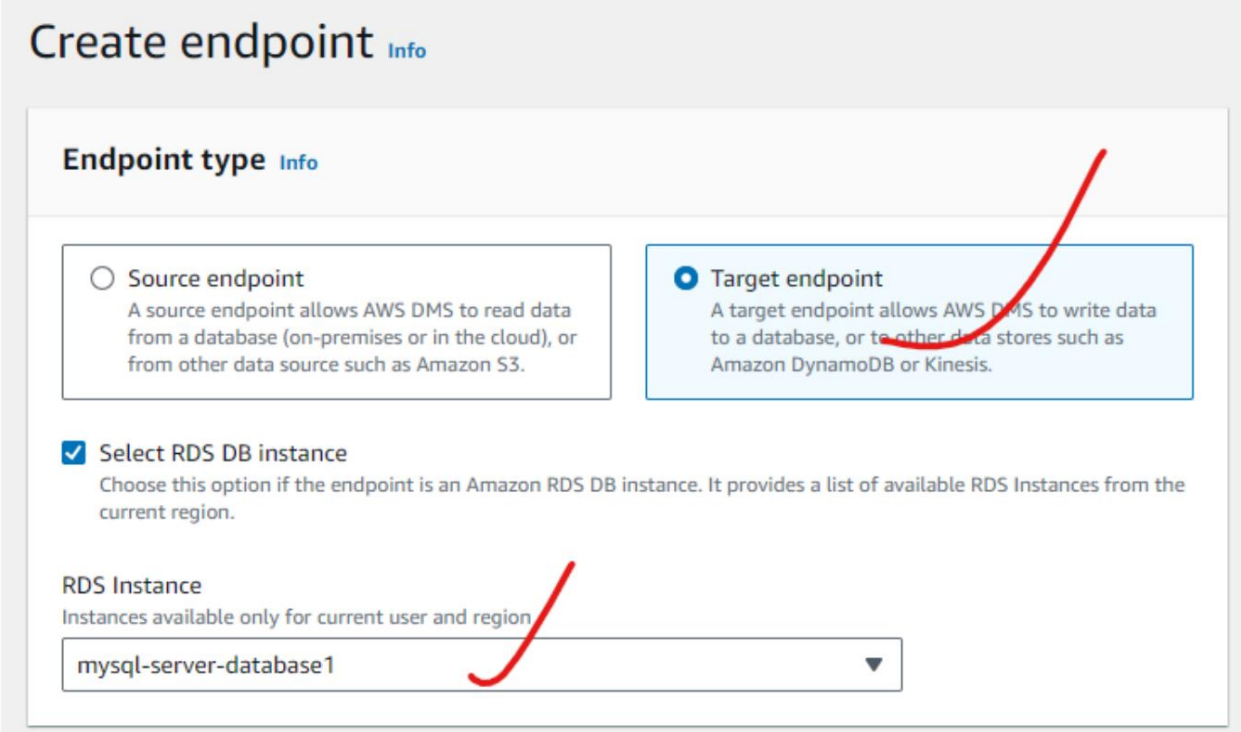
- Navigate to the Endpoints section of the AWS DMS console. Choose Create Endpoint termination to create a new endpoint



• In the Create Endpoint Wizard, choose to create an endpoint

target. Check the Select RDS DB instance box, then choose your database

Newly created Amazon RDS from the drop-down menu.



• Most of the Endpoint configuration details are now completed. You should

Enter your password and a database name at the bottom of the section.

Endpoint configuration

Endpoint identifier Info

A label for the endpoint to help you identify it.

mysql-server-database1

Descriptive Amazon Resource Name (ARN) - *optional*

A friendly name to override the default DMS ARN. You cannot modify it after creation.

Friendly-ARN-name

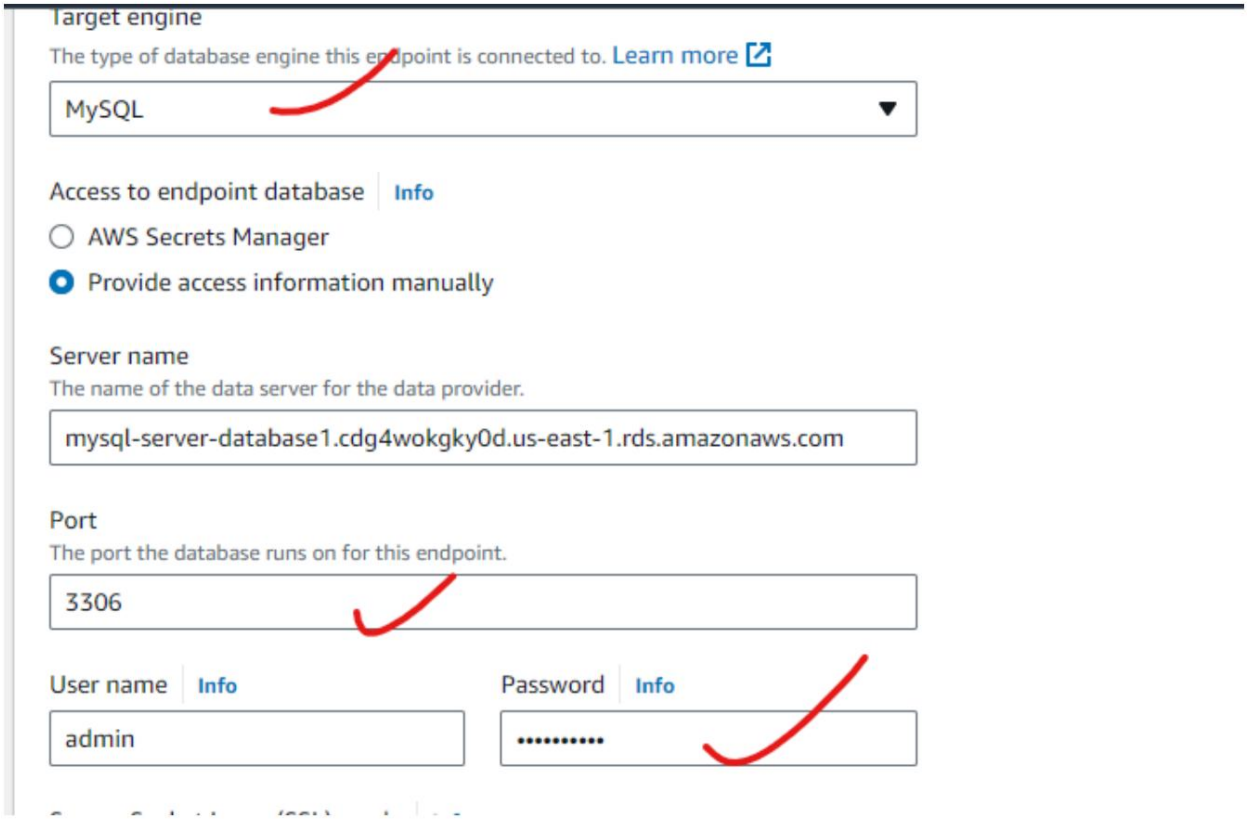
Target engine

The type of database engine this endpoint is connected to. Learn more

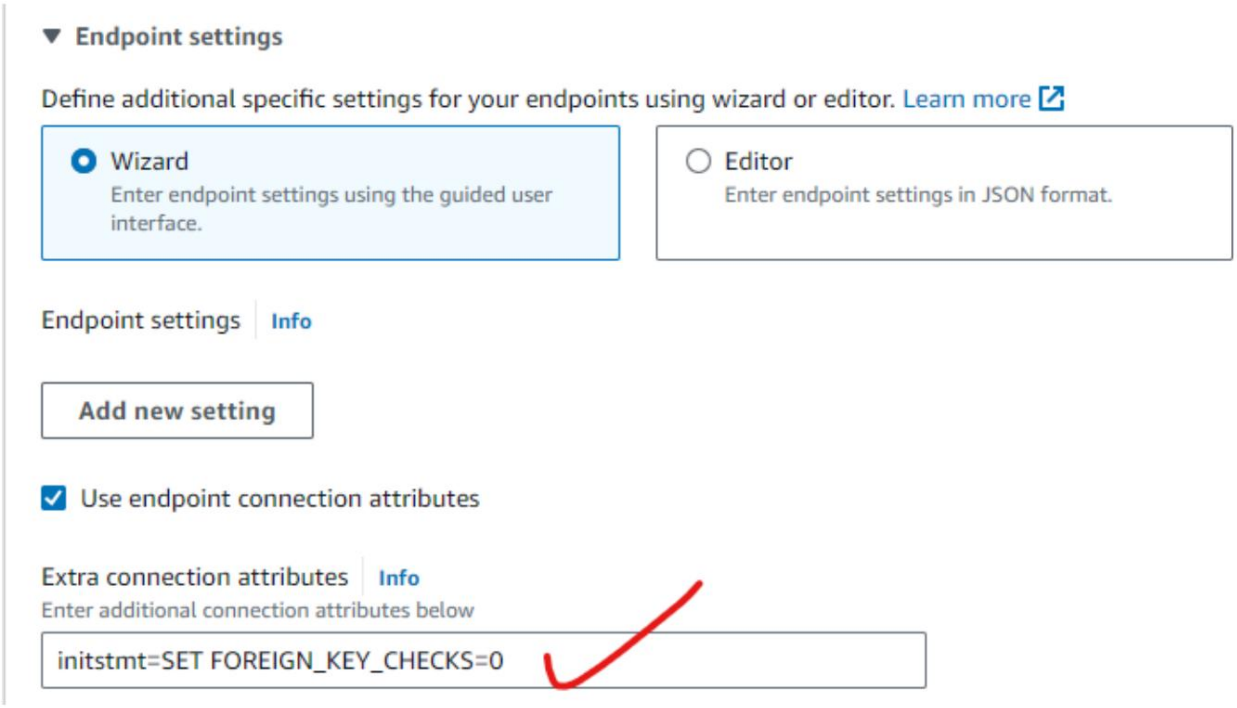
MySQL

Access to endpoint database Info

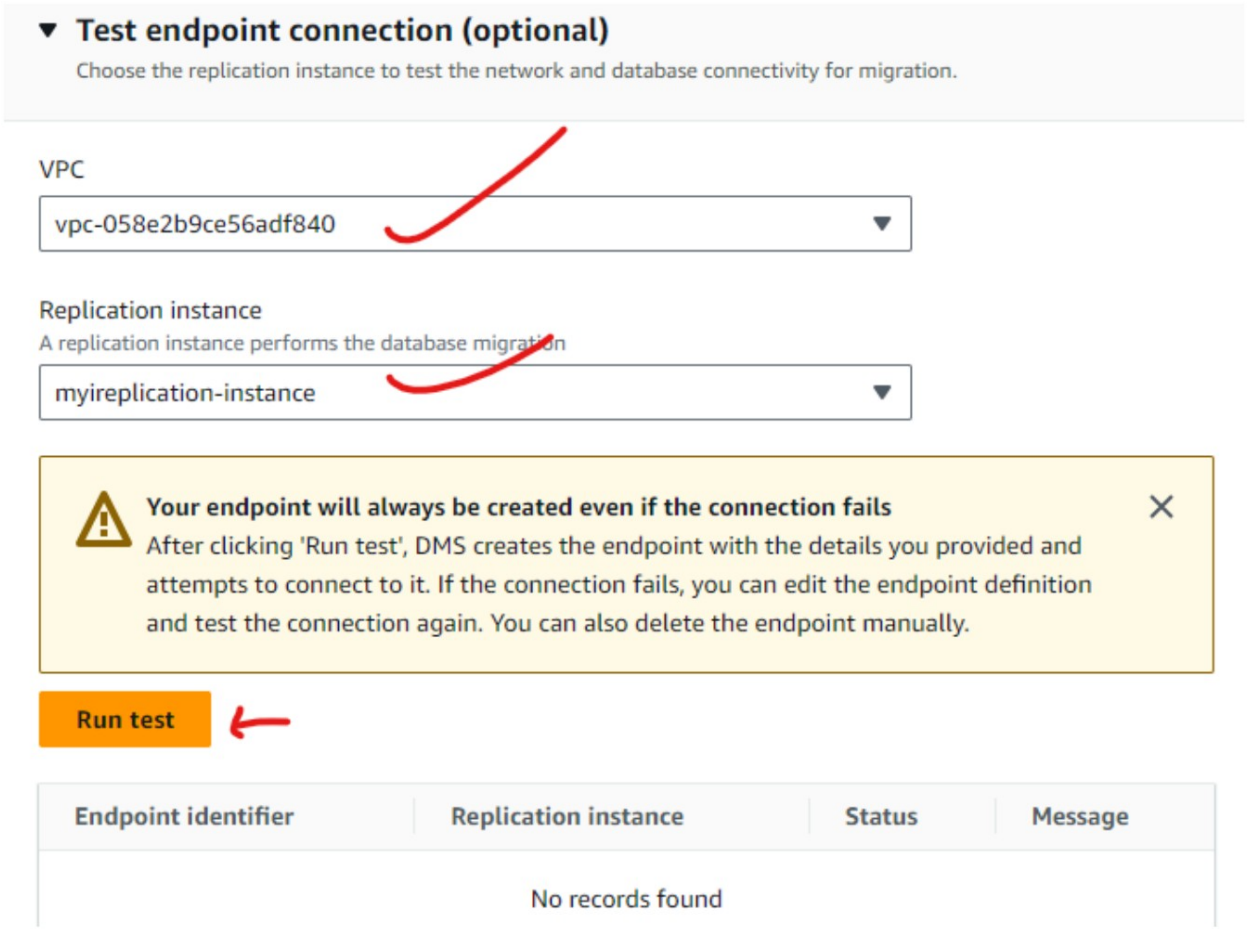
- (●) AWS Secrets Manager
- ( ) Provide access information manually

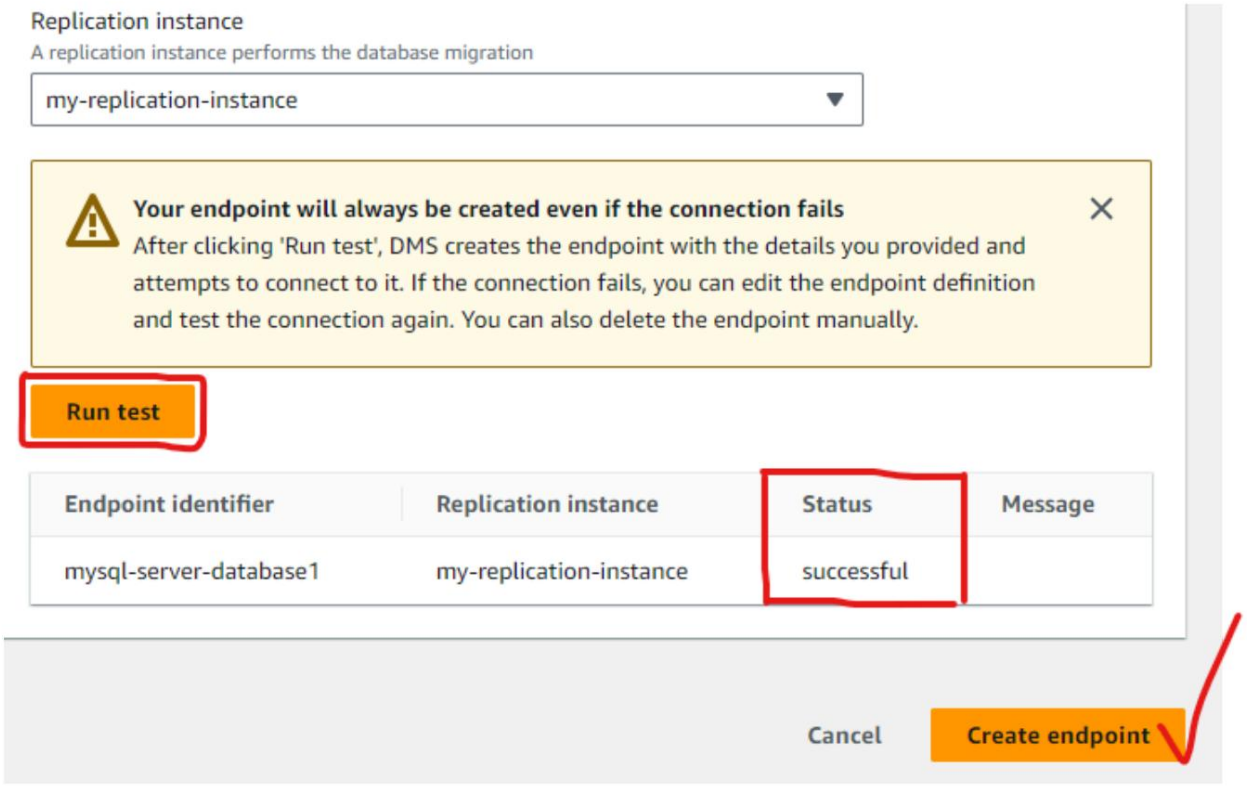


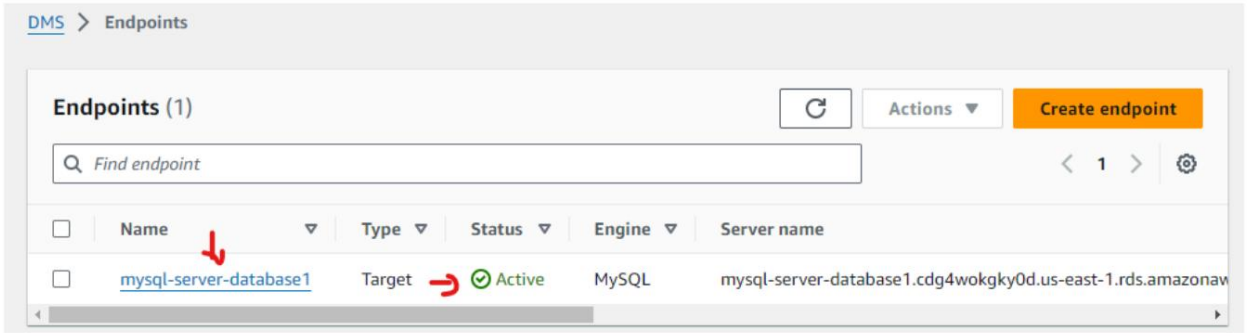
- When you load data into a MySQL database using AWS DMS, you need to disable foreign key checks. To do this, type **initstmt=SET FOREIGN_KEY_CHECKS=0** in the **Extra connection attributes.** box



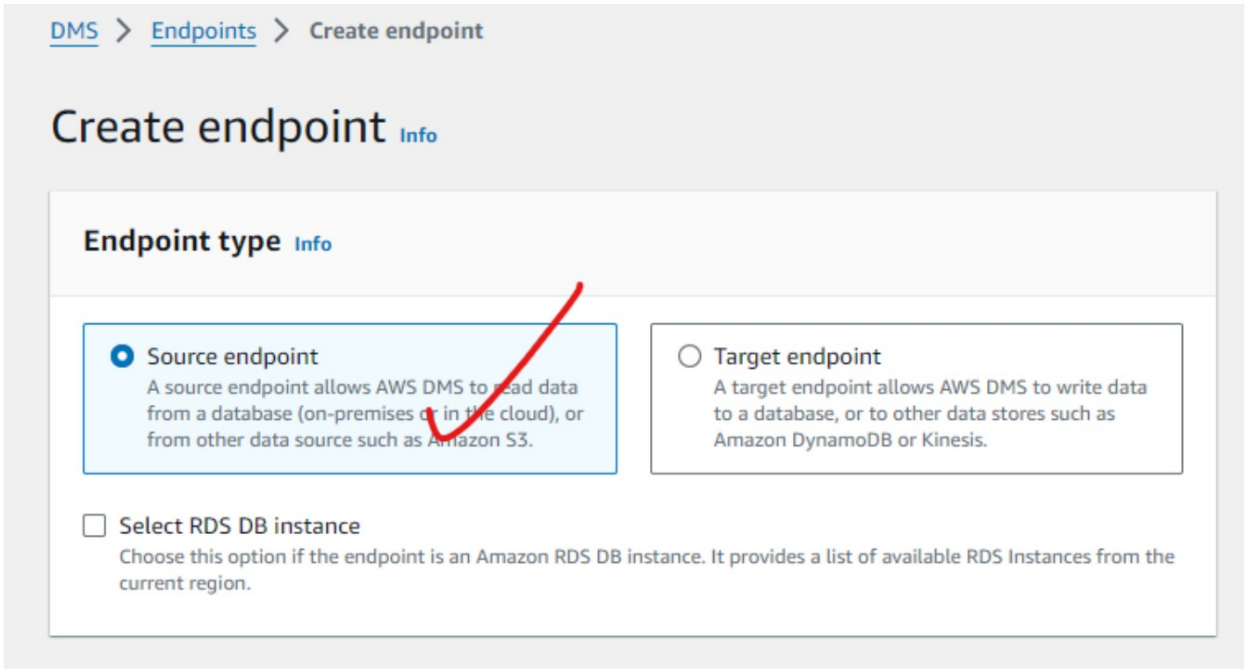
- Before saving your endpoint, test the connection to ensure that it has been configured correctly. To do this, open the Test Point Connection section end.
- Choose the replication instance you want to use, then select Run the test. After a few seconds, you should see a success status. This indicates that you have correctly configured your security group and your endpoint. To register your endpoint, select Create an endpoint endpoint.

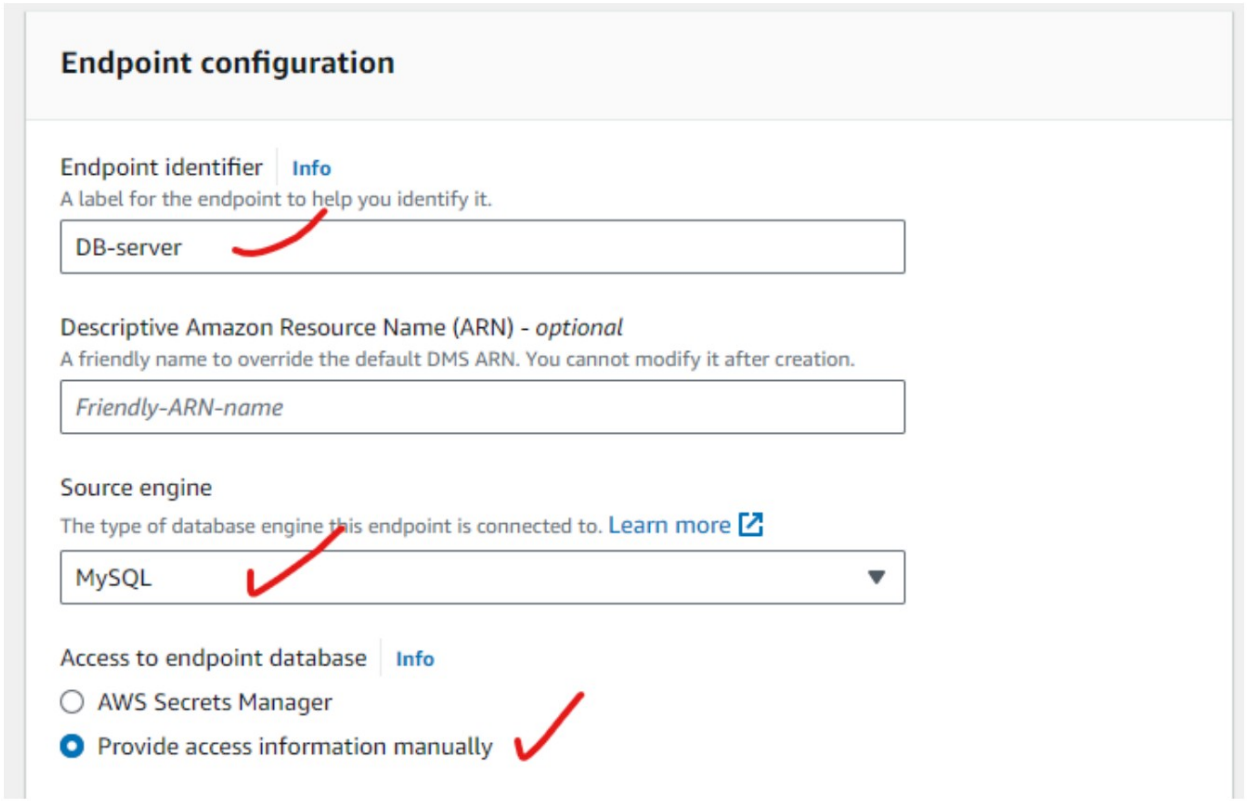


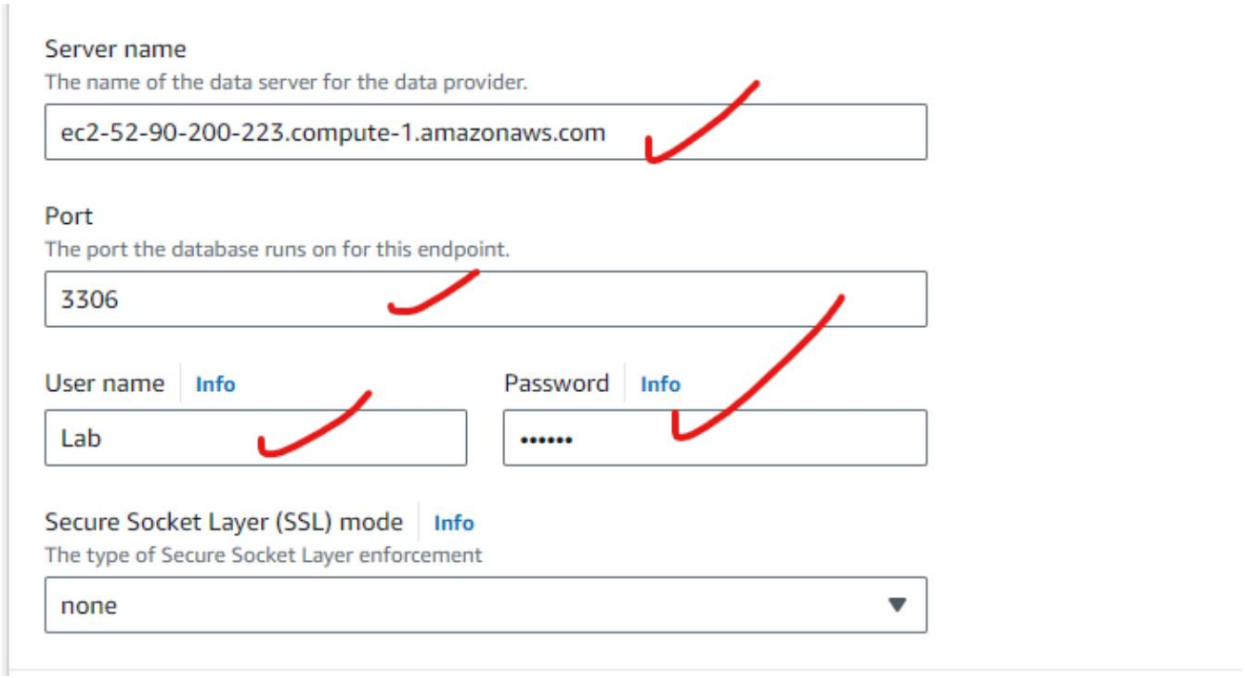




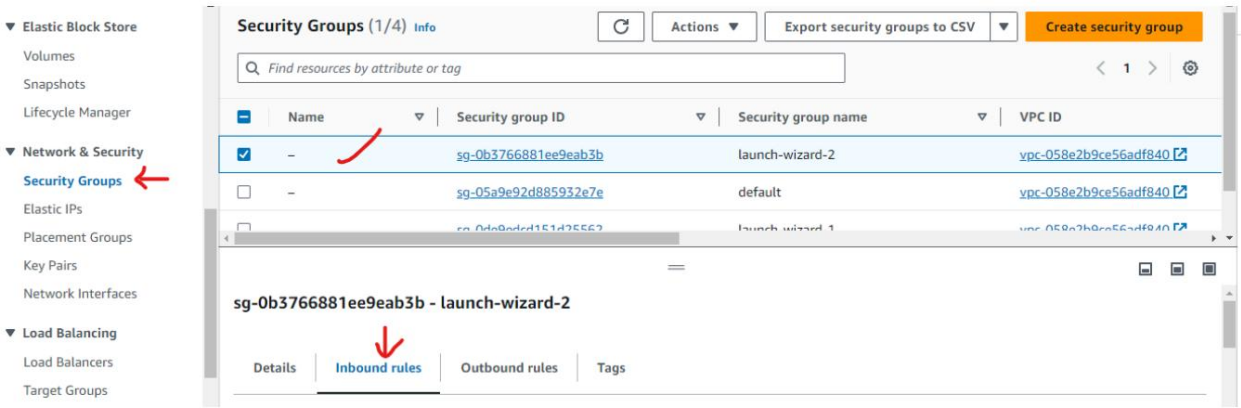
Step 5: Create the source endpoint



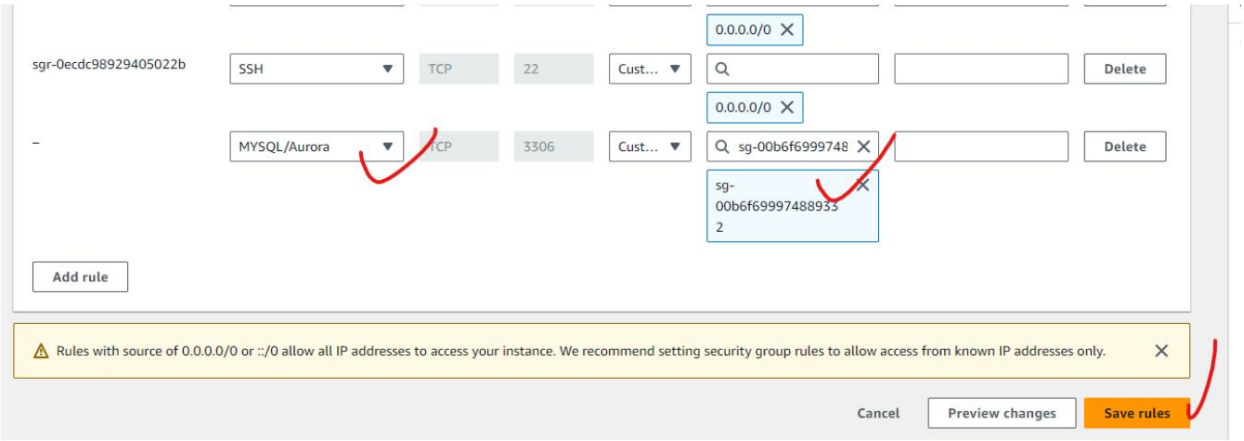




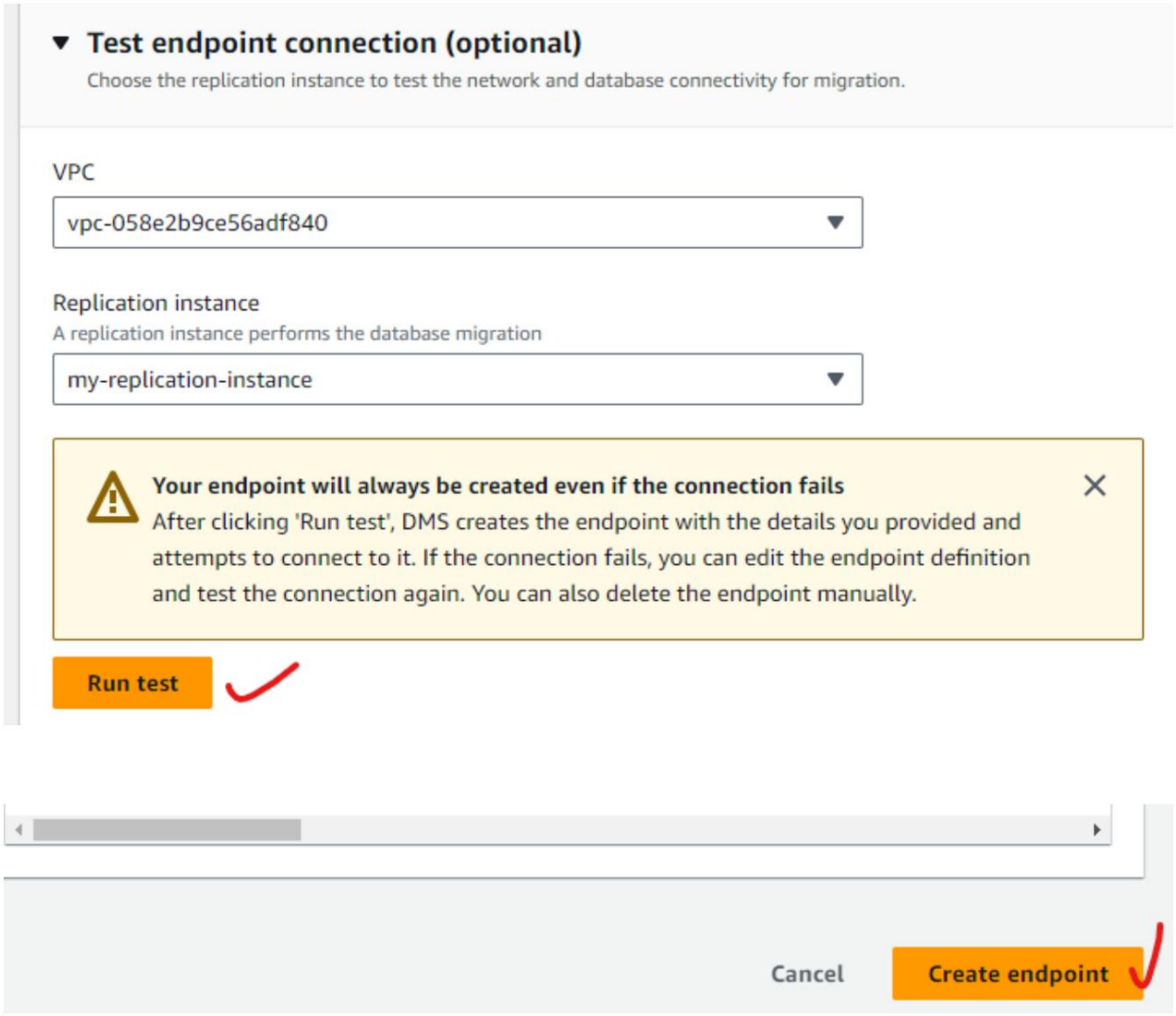
- You also need to ensure that your replication instance has access network to your source database. If your source database is hosted on Amazon EC2, allow traffic from your replication instance's security group to the security group of the source database. If your source database is not hosted on Amazon EC2, you must manage network settings accordingly from the location of your source database.



- Add the DMS replication instance security group to allow it to connect to your MySQL database on EC2 instance via port 3306.



- Then run the connection test on your endpoint

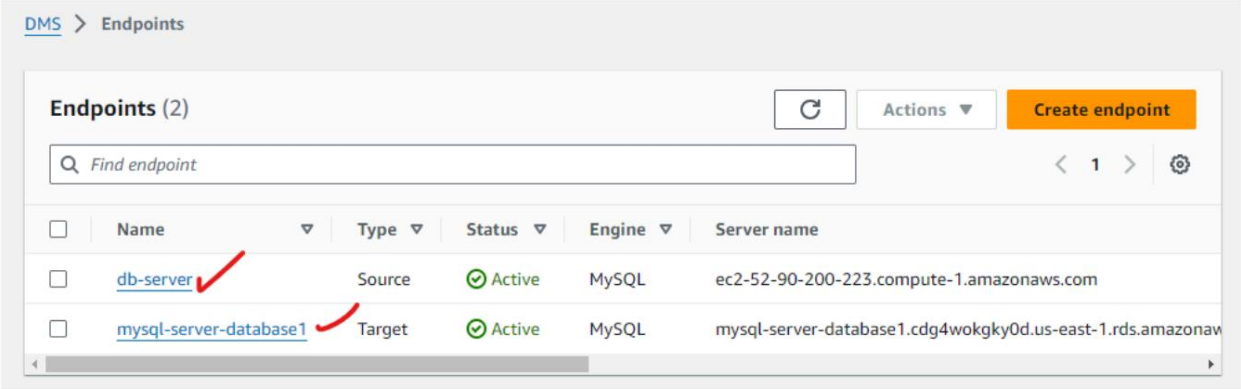


- Before proceeding to the next step, you must have configured two endpoints:

  one for your source database and one for your target database.

  Make sure you have tested both endpoints and can connect with them

  success to both databases.



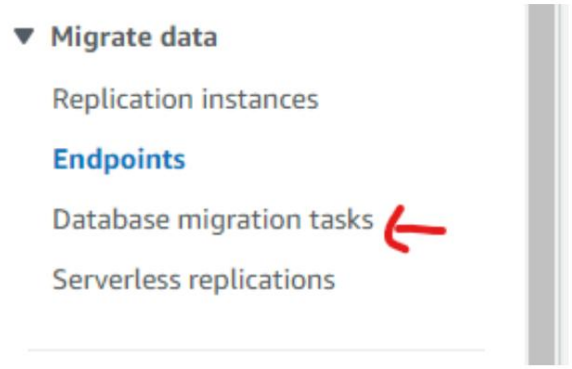
In this step, you created your endpoints to connect to your databases.
data. In the next step, you will use these endpoints to create a task
replication that copies data from your source database to your database
target.

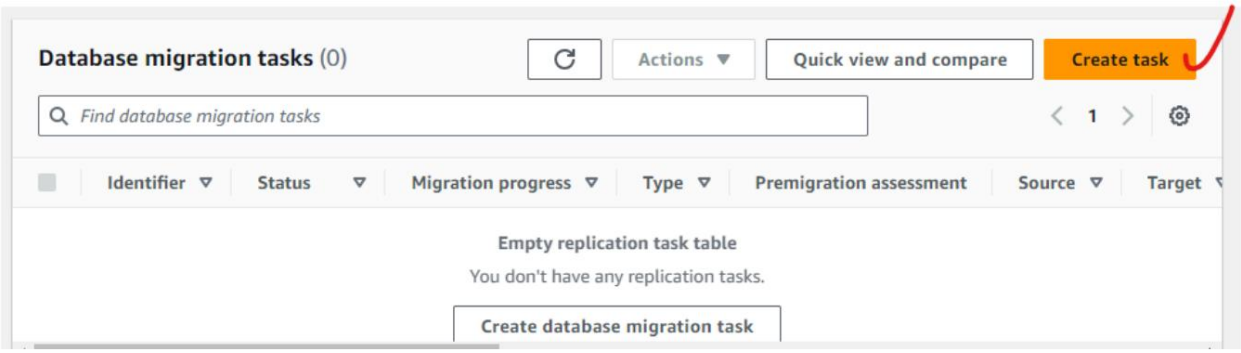
Step 6: Creating Replication Tasks

In this part, you create **a replication task in AWS DMS.**

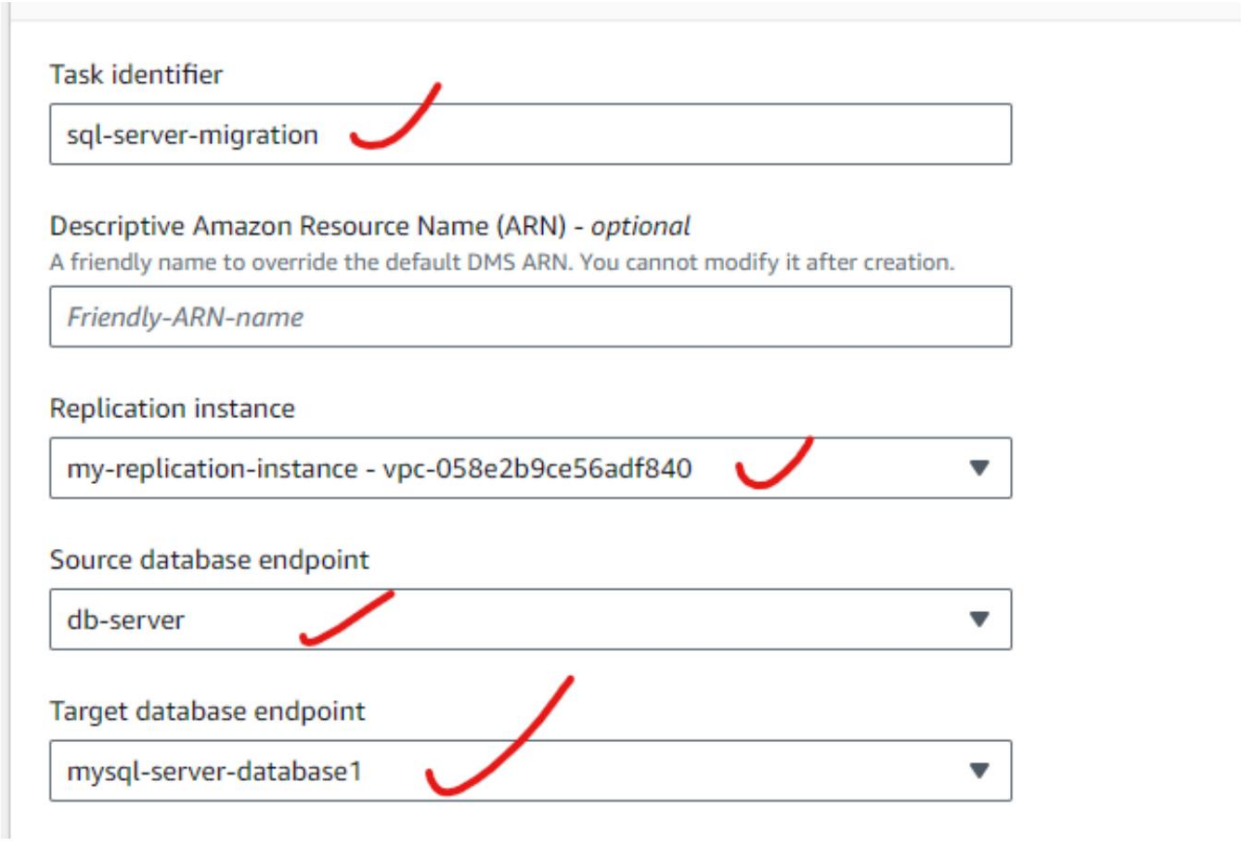
A replication task is responsible for migrating data from a source database to
a target database. In your case, you are moving data from a database
existing database to a newly created database on Amazon RDS.

- To get started, navigate to the Replication Tasks section of the AWS DMS console.
  Select Create Task to create a new replication task.





In the Task Configuration section, define the settings for your replication task. Give your task a name and choose the replication instance you created in a previous module. Then choose the source endpoint for your database existing and target endpoint for your fully managed database in Amazon RDS.



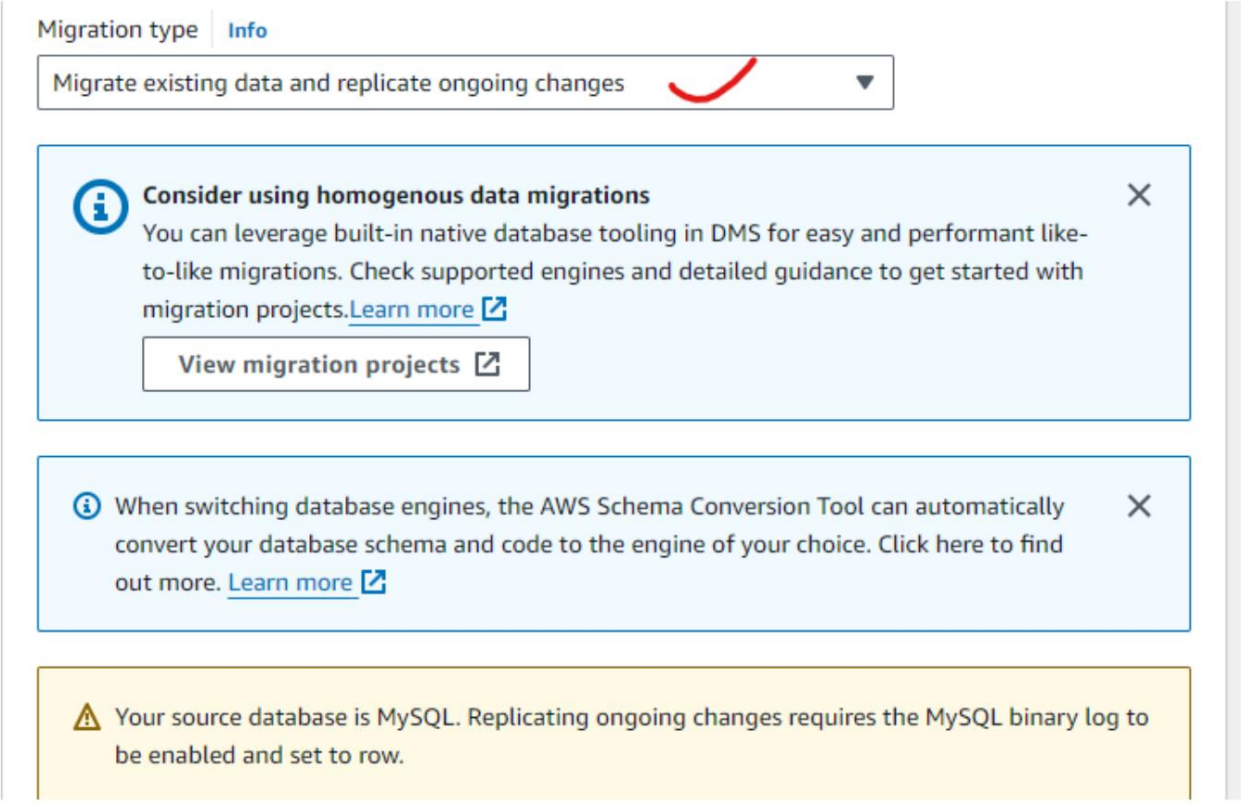
You must choose a migration type. There are 3 types of migration:

1. Migrate existing data, which performs a one-time process of copying data from the source database to target database.

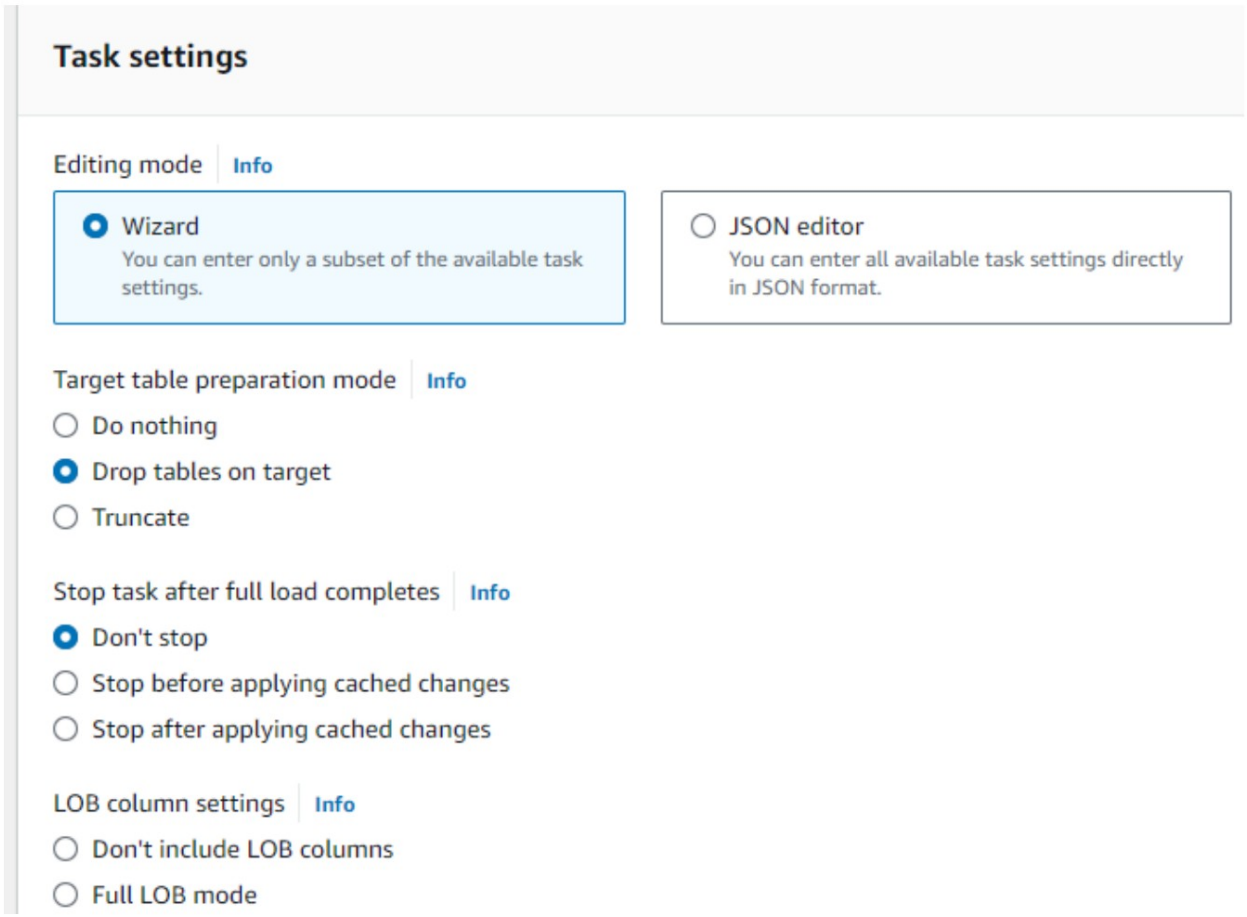
2. Reproduce current changes, which copies all current operations from your source database to your target database.

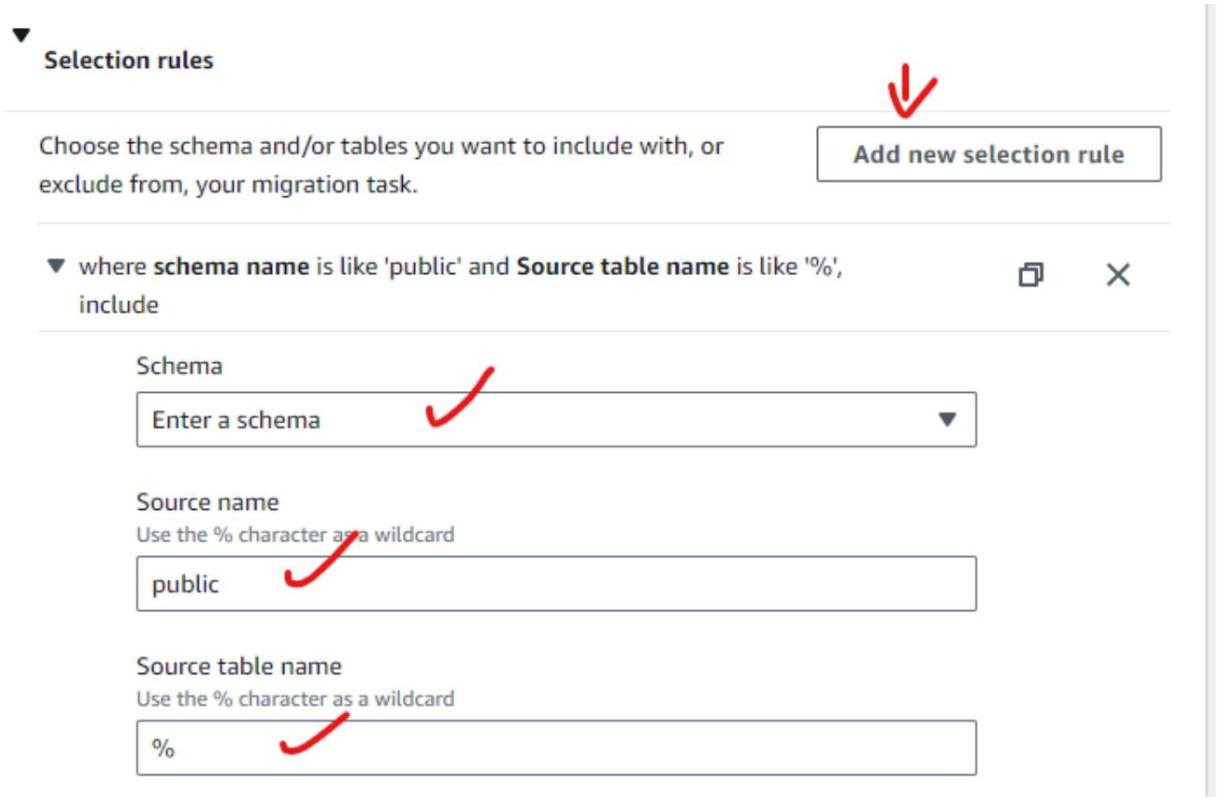
3. Replicate data changes only: This option allows you to capture and replicate only the changes to data in the source database after an initial migration or on a database already in use. This includes operations such as insertions, updates and deletions.

- If you are moving your application from a self-managed database to a managed database, fully managed data, you must use both types of replication. The first type copies all the data from your database and the second type ensures that All additional updates are replicated to your new database. data until you switch your application to use the new database. data.
- For Migration type, select Migrate existing data and replicate existing data. changes in progress. Note that logical replication must be enabled on your database of source data.



- In the Table settings section, choose the tables you want to copy. Enter the name of the schemas and tables you want to copy. You can use % as wildcard to copy multiple tables or schemas.





- When you are ready, select Create Task to start your task.

  replication.