

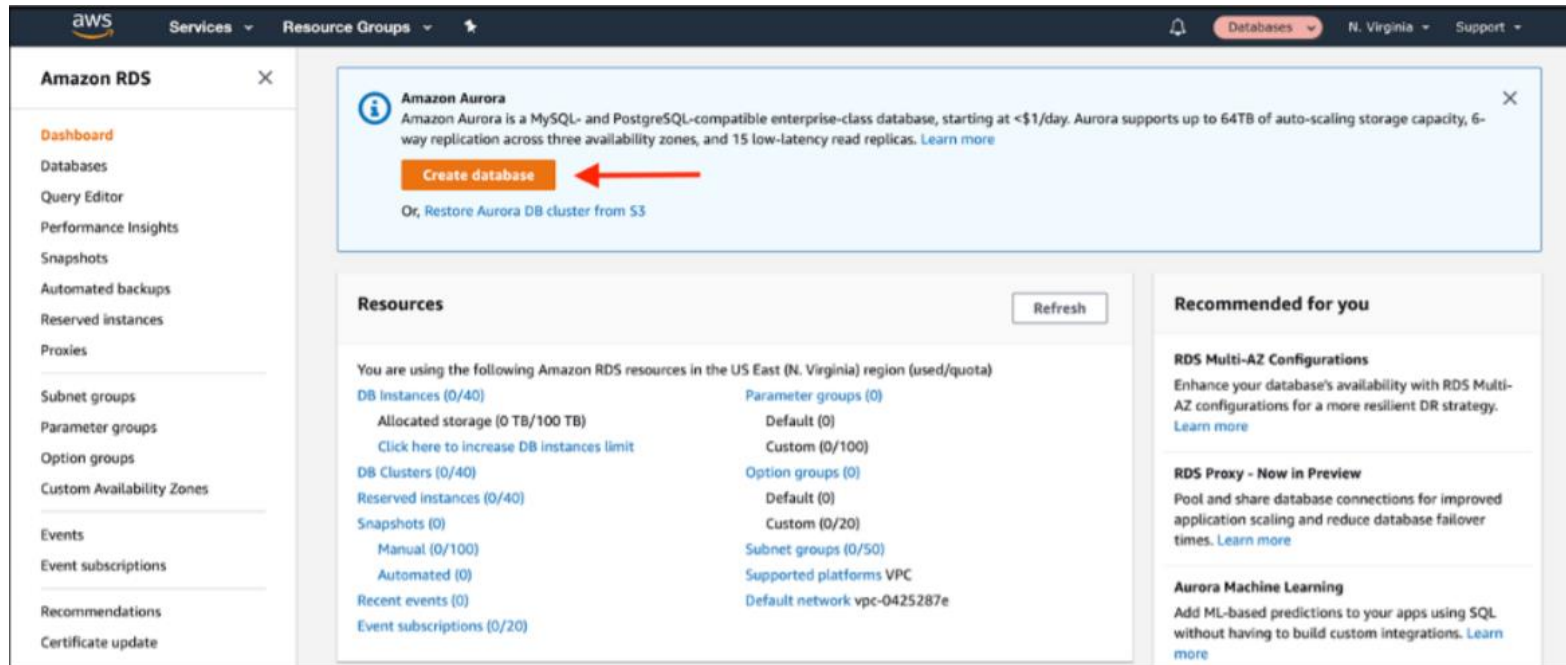
## Exercise: Migrating from one SQL Server to another in Amazon RDS

In this exercise, you will migrate from one SQL Server database to another in Amazon Relational Database Service (Amazon RDS).

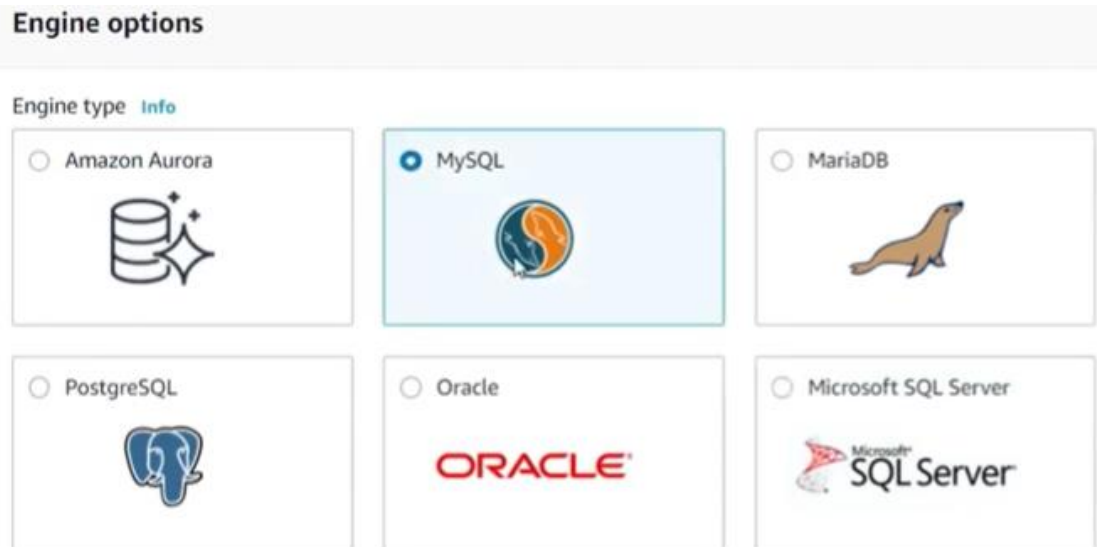
Migrating process:

1. Create **target** SQL Server **database instance** in Amazon RDS.
2. Create a **replication instance** in AWS Database Migration Service (AWS DMS)
3. Create **source** SQL Server **database instance** in Amazon RDS.
4. Create **source and target endpoints** for your database migration.
5. Create **replication task** in AWS DMS.
6. Complete the migration and clean up resources.

1. Navigate to the Amazon RDS console. On the main page, choose **Create database** to create a new database.



2. In the **Engine options** section, choose MySQL as the Engine type.



3. Choose the **Production** template. In the **Settings** section, give your database a name and set the master username and password

## Settings

**DB instance identifier** [Info](#)

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

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens (1 to 15 for SQL Server). 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.

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

☐ **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), "(double quote) and @ (at sign).

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

## 4. Choose the database instance size.

### DB instance class [Info](#)

Choose a DB instance class that meets your processing power and memory requirements. The DB instance class options below are limited to those supported by the engine you selected above.

- ☐ Standard classes (includes m classes)
- ☐ Memory Optimized classes (includes r and x classes)
- ☒ Burstable classes (includes t classes)

db.t2.micro

1 vCPUs

1 GiB RAM

Not EBS Optimized

☐ Include previous generation classes

## 5. Choose Storage. Enable **Storage autoscaling**, if you want.

### Storage

#### Storage type [Info](#)

General Purpose (SSD)

#### Allocated storage

20

GiB

(Minimum: 20 GiB, Maximum: 16 384 GiB) Higher allocated storage **may improve** IOPS performance.

## 6. Estimated monthly costs is just for info. You have to fit in “free tier” requirements. So, push **Create database**.

### DATABASE

Free Tier

12 MONTHS FREE

Amazon RDS

# 750 Hours

per month of db.t2.micro database usage  
(applicable DB engines)

Managed Relational Database Service  
for MySQL, PostgreSQL, MariaDB,  
Oracle BYOL, or SQL Server.

750 Hours per month of db.t2.micro database  
usage (applicable DB engines)

20 GB of General Purpose (SSD) database  
storage

20 GB of storage for database backups and  
DB Snapshots

### Estimated monthly costs

DB instance	12.41 USD
Storage	4.60 USD
Multi-AZ standby instance	12.41 USD
<b>Total</b>	<b>29.42 USD</b>

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

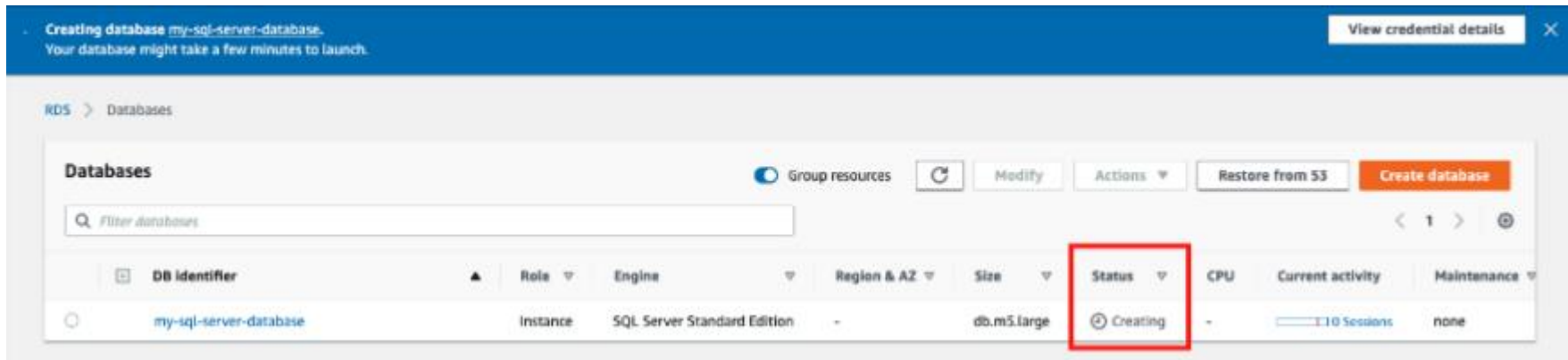
Estimate your monthly costs for the DB Instance using the [AWS Simple Monthly Calculator](#).

 You are responsible for ensuring that you have all of the necessary rights for any third-party products or services that you use with AWS services.

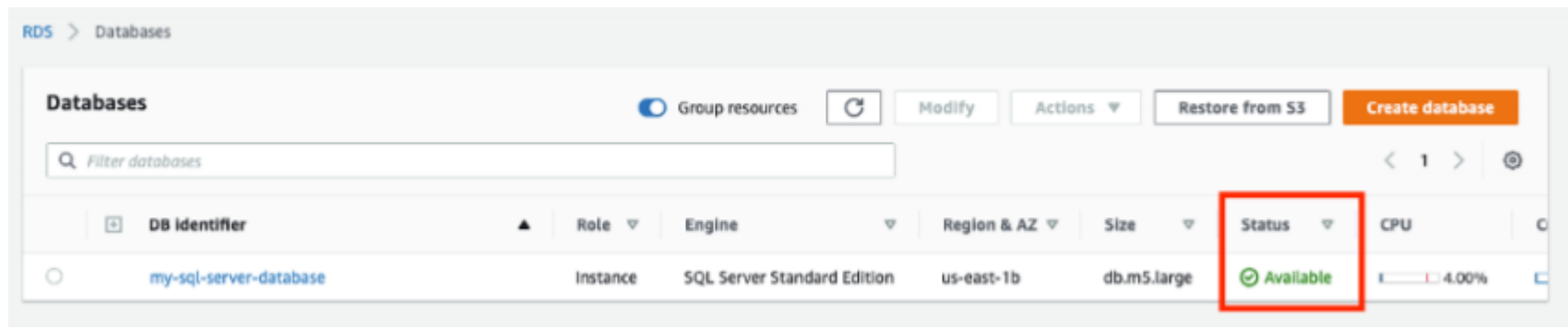
Cancel

Create database

7. As Amazon RDS is provisioning your infrastructure and initializing your database, the status of your database is **Creating**.



When your database is ready to use, its status is **Available**.



8. In AWS DMS console go to Replication instances and Create replication instance



9. Then choose an engine version for AWS DMS. Finally, choose the amount of allocated storage for your replication instance.

**Name**  
The name must be unique among all of your replication instances in the current AWS region.

Replication instance name must not start with a numeric value

**Descriptive Amazon Resource Name (ARN) - optional**  
A friendly name to override the default DMS ARN. You cannot modify it after creation.

**Description**

The description must only have unicode letters, digits, whitespace, or one of these symbols: \_-./=+ maximum character.

**Instance class** [Info](#)  
Choose an appropriate instance class for your replication needs. Each instance class provides different compute, network and memory capacity. [DMS pricing](#)

▼  
1 vCPUs 1 GiB Memory

☒ Include previous-generation instance classes

**Engine version**  
Choose an AWS DMS version to run on your replication instance. [DMS versions](#)

▼

☐ Include Beta DMS versions

**Allocated storage (GiB)**  
Choose the amount of storage space you want for your replication instance. AWS DMS uses this storage for log files and cached transactions while replication tasks are in progress.

10. You may choose to have a Multi-AZ setup for your replication instance for redundancy. Finally, choose whether your replication instance should be publicly accessible.


#### VPC

Choose an Amazon Virtual Private Cloud (VPC) where your replication instance should run.

vpc-0425287e ▼

☐ **Multi AZ**

If you choose this option, AWS DMS will perform a multi-AZ deployment, with a primary instance in one availability zone (AZ) and a standby instance in another AZ. This configuration provides a highly available, fault-tolerant replication environment.

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

☐ **Publicly accessible**

If you choose this option, AWS DMS will assign a public IP address to your replication instance, and you'll be able to connect to databases outside of your Amazon VPC.



11. Open the **Advanced security and network configuration** section. For the **VPC security group(s)** configuration, choose the same security group that you attached to your Amazon RDS database. This allows your replication instance to access your Amazon RDS database.

**▼ Advanced security and network configuration**

**Replication subnet group**  
Choose a subnet group for your replication instance. The subnet group defines the IP ranges and subnets that your replication instance can use within the Amazon VPC you've chosen.  


default-vpc-0425287e ▼

**Availability zone**  
Choose an availability zone (AZ) where you want your replication instance to run. The default is "No preference", meaning that AWS DMS will determine which AZ to use.  

No Preference ▼

**VPC security group(s)**  
Choose one or more security groups for your replication instances. The security group(s) specify inbound and outbound rules to control network access to your replication instance.  

Use default ▼

sql-server-database X 

**KMS master key** [Info](#)  

(Default) aws/dms ▼

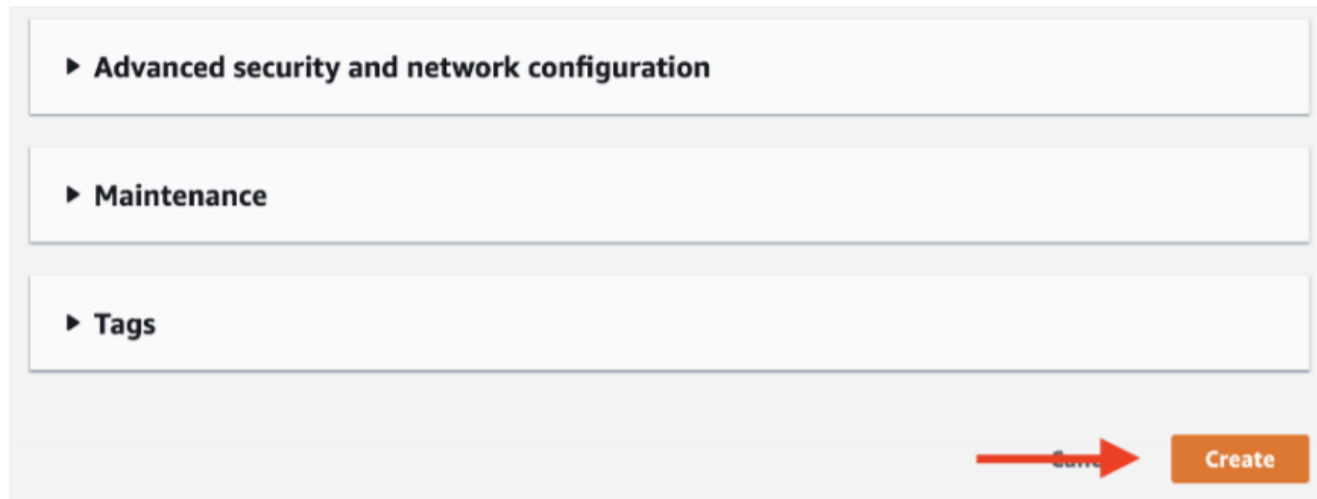
**Account**

**Description**

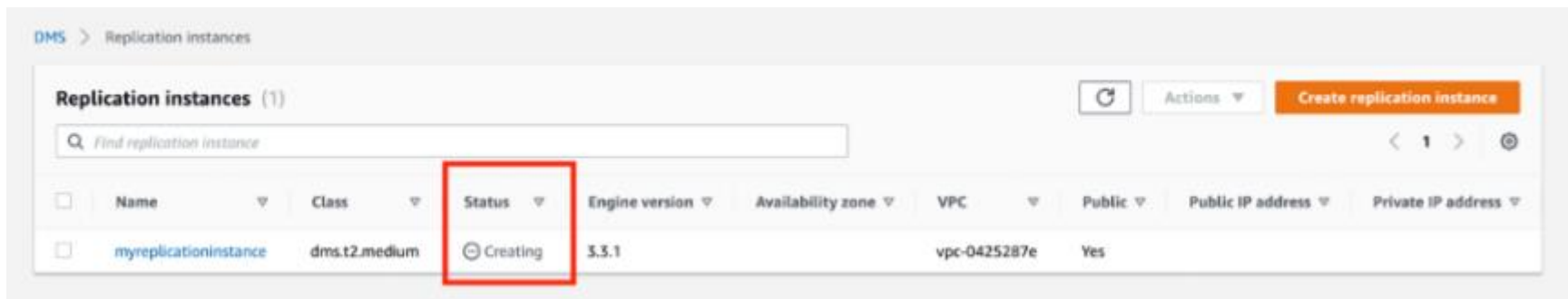
**Key ARN**



12. When you're ready, choose **Create** to create your replication instance in AWS DMS.



13. After you choose **Create**, AWS provisions your replication instance. It shows a status of **Creating** while AWS provisions and initializes your instance.

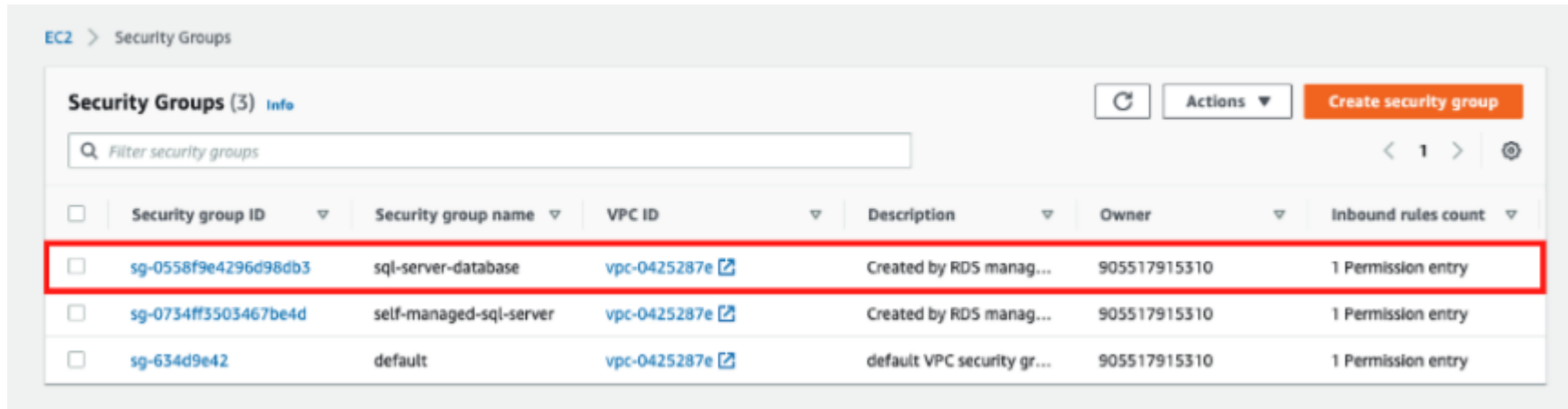


When your replication instance is ready to go, its status is **Available**.

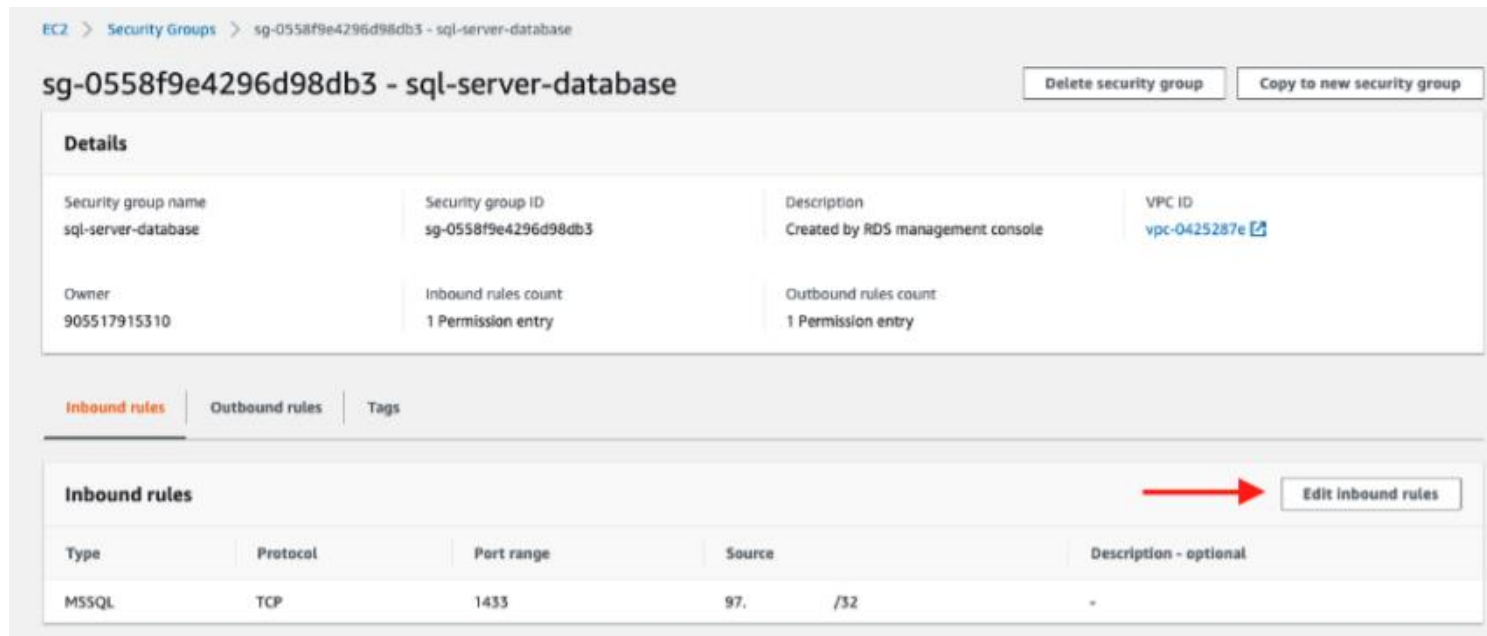


14. While you are waiting for your replication instance to be available, go to the Amazon EC2 console : **Security Groups** section.

In the **Security Groups** section, find the security group you attached to your SQL Server database instance and your replication instance, and choose it.



15. Choose **Edit inbound rules** for your security group.



16. Your security group has an existing rule that allows for access to your SQL Server instance from the IP address you used to create the database. Remove the existing IP address and enter the name of the security group used for your Amazon RDS database instance and replication instance. Your screen should look as follows.

**Edit inbound rules** [Info](#)

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

Type <a href="#">Info</a>	Protocol <a href="#">Info</a>	Port range <a href="#">Info</a>	Source <a href="#">Info</a>	Description - optional <a href="#">Info</a>
MSSQL	TCP	1433	Custom <input type="text" value="sg-0558f9e4296d98db3"/>	<input type="text"/>

[Add rule](#) [Delete](#)

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

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

17. Choose **Save rules** to save the updated rules for your security group.

## 18. Create **source** SQL Server **database instance** in Amazon RDS (see steps 1-7).

### 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.

source-my-sql-database-1

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens (1 to 15 for SQL Server). 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.

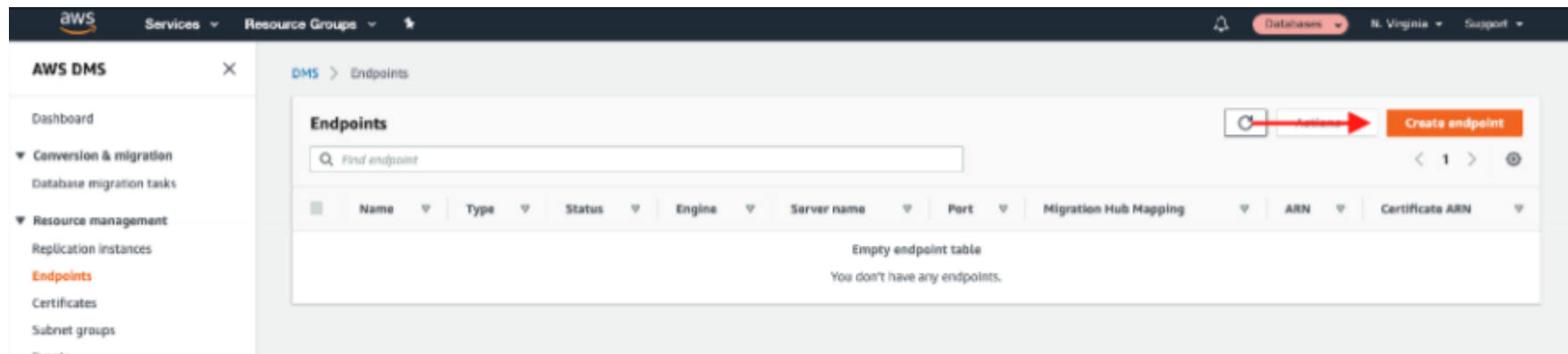
admin

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

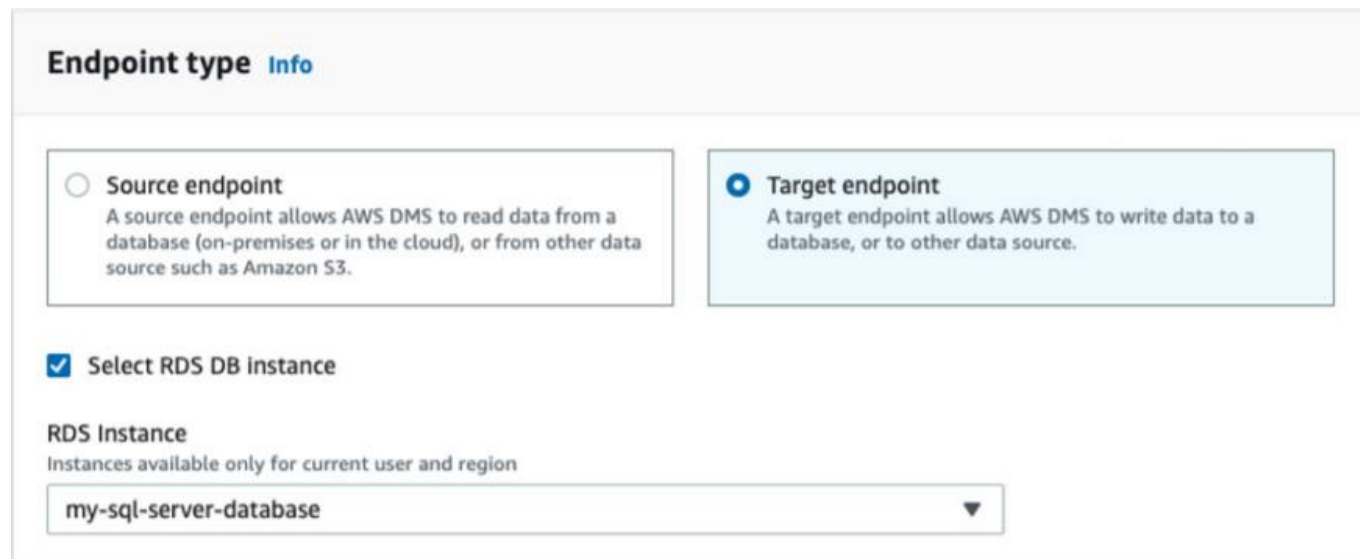
☐ **Auto generate a password**

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

19. In this step, you create source and target endpoints for a replication task in AWS DMS. An endpoint describes the connection address, credentials, and other information required to connect to a database. In the next step, you use these endpoints to create a replication task that copies data from your source database to your target database. First, let's create the endpoint for your target database. Navigate to the Endpoints section of the **AWS DMS** console. Choose **Create endpoint** to create a new endpoint.



20. In the endpoint creation wizard, choose to create a **Target endpoint**. Choose the check box to Select RDS DB Instance, and choose your **target** Amazon RDS database in the dropdown.



21. You need to enter your password and a database name near the bottom of the section.

## Endpoint configuration

### Endpoint identifier [Info](#)

A label for the endpoint to help you identify it.

my-sql-server-database

### Target engine

The type of database engine this endpoint is connected to.

sqlserver

### Server name

my-sql-server-database.cv6ykrzkgphs.us-east-1.rds.amazonaws.com

### Port

The port the database runs on for this endpoint.

1433

### Secure Socket Layer (SSL) mode

The type of Secure Socket Layer enforcement

none

### User name [Info](#)

admin

### Password [Info](#)

### Database name

dbo

22. Before you save your endpoint, you should test the connection to ensure it was configured correctly. Open the **Test endpoint connection** section to test your connection.

Choose the replication instance you want to use, and then choose **Run test**. After a few seconds, you should see a status of successful. This indicates you configured your security group and endpoint correctly. Choose **Create endpoint** to save your endpoint.

▼ **Test endpoint connection (optional)**

Test your endpoint connection by selecting a replication instance within your desired VPC. After clicking "Run test", an endpoint will be created with the details provided and attempt to connect to the instance. If the connection fails, you can edit and test it again. Endpoints that aren't saved will be deleted.

VPC

vpc-0425287e ▼

Replication instance

A replication instance performs the database migration

my-replication-instance ▼

Run test

←

After clicking "Run test", an endpoint will be created with the details provided and attempt to connect to the instance. If the connection fails, you can edit and test it again. Endpoints that aren't saved will be deleted.

Endpoint Identifier	Replication Instance	Status	Message
my-sql-server-database	my-replication-instance	successful	

→ Cancel

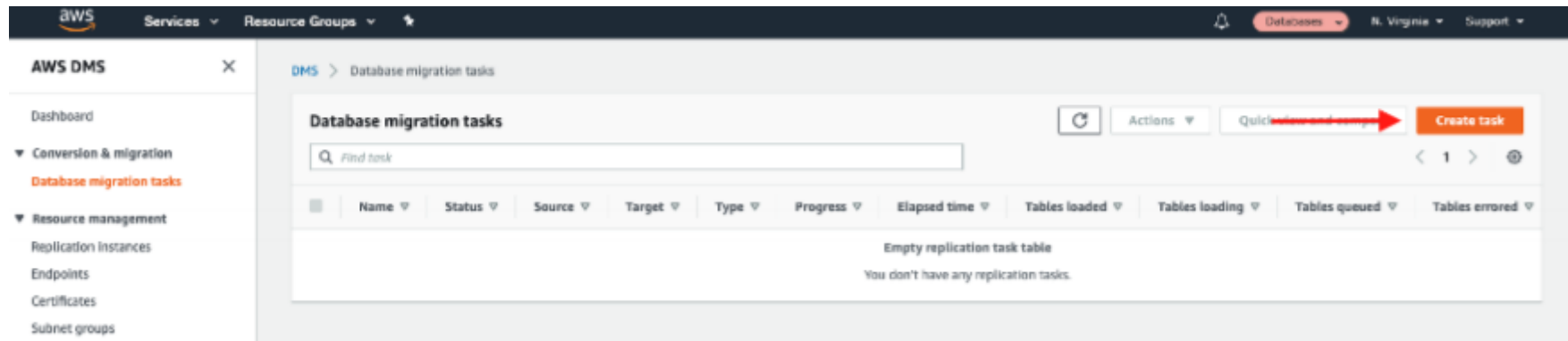
Create endpoint



23. Follow steps 19-22 to create an endpoint for your **source** database. Unlike the target database, you need to complete the connection endpoint, port, and credentials yourself.

Before moving on to the next step, you should have two endpoints configured: one for your source database and one for your target database. Make sure that you have tested both endpoints and can successfully connect to both databases. Then move on to the next step.

24. In this step, you create a replication task in **AWS DMS**. A replication task copies the data from your source database to your target database. To get started, navigate to the Replication tasks section of the **AWS DMS** console. Choose **Create task** to create a new replication task.



25. In the Task configuration section, set up the parameters of your replication task. Give your task a name and choose the replication instance you created in an earlier step. Then choose the source endpoint for your existing database and your target endpoint for your fully managed database in Amazon RDS.

You need to choose a migration type. There are two migration types:

1. Migrate existing data, which performs a one-time process to copy data from your source database to your target database.
2. Replicate data changes, which copies all ongoing operations from your source database to your target database.

If you are migrating your application from using a self-managed database to using a fully managed database, you want to use both types. The first type copies all data in your database, and the second type ensures that all additional updates are replicated to your new database until you switch your application to use the new database. For the migration type, choose Migrate existing data and replicate ongoing changes. Note that this requires you to have logical replication enabled on your source database.

## Task configuration

Task identifier

sql-server-migration

Replication instance

my-replication-instance - vpc-0425287e

Source database endpoint

self-managed-sql-server

Target database endpoint

my-sql-server-database

Migration type [Info](#)

Migrate existing data and replicate ongoing changes



When switching database engines, the AWS Schema Conversion Tool can automatically convert your database schema and code to the engine of your choice. Click here to find out more. [Learn more](#)



☒ Start task on create

26. In the **Table mappings** section, tell AWS DMS which tables you want to copy. Enter the name of the schemas and tables you want to copy. You can use % as a wildcard character to copy multiple tables or schemas.

The screenshot shows the 'Table mappings' configuration window in AWS DMS. At the top, there is a section titled 'Selection rules' with a dropdown arrow. Below this, a text prompt says 'Choose the schema and/or tables you want to include with, or exclude from, your migration task.' followed by an 'Info' link. To the right of this text is a button labeled 'Add new selection rule'. Below the text, a rule is listed: 'where schema name is like '%' and table name is like '%', exclude'. To the right of this rule are icons for a copy and a close (X) button. The rule details are shown in a form with three main sections: 'Schema' with a dropdown menu currently showing 'Enter a schema'; 'Schema name' with a text input field containing '%'; and 'Table name' with a text input field containing '%'. Below these is an 'Action' section with a text prompt 'Choose "Include" to migrate your selected objects, or "Exclude" to ignore them during the migration.' and a dropdown menu currently set to 'Exclude'. At the bottom of the window, there is a section for 'Transformation rules' with a right-pointing arrow. The bottom right corner of the window has 'Cancel' and 'Save' buttons.

▼ Selection rules

Choose the schema and/or tables you want to include with, or exclude from, your migration task. [Info](#) Add new selection rule

▼ where **schema name** is like '%' and **table name** is like '%', exclude 📄 ✕

Schema  
Enter a schema ▼

Schema name  
Use the % character as a wildcard  
%

Table name  
Use the % character as a wildcard  
%

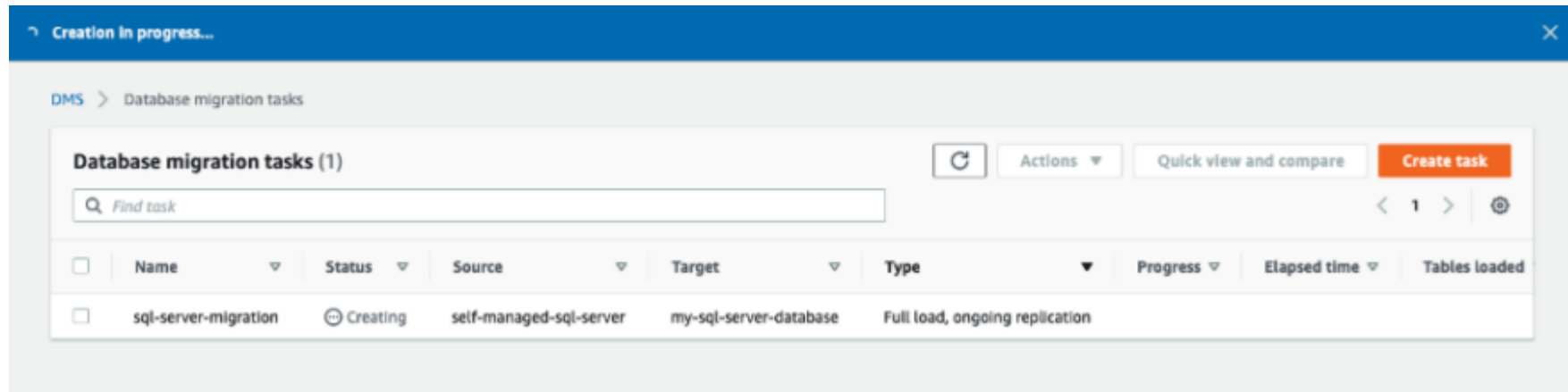
Action  
Choose "Include" to migrate your selected objects, or "Exclude" to ignore them during the migration.  
Exclude ▼

► Transformation rules

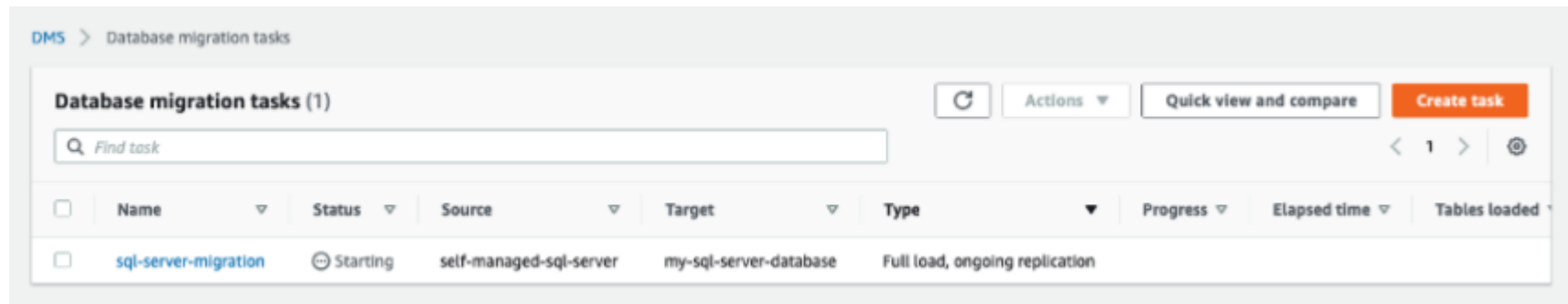
Cancel Save

**Action:** you can leave Exclude, because it is not important in these Exercise

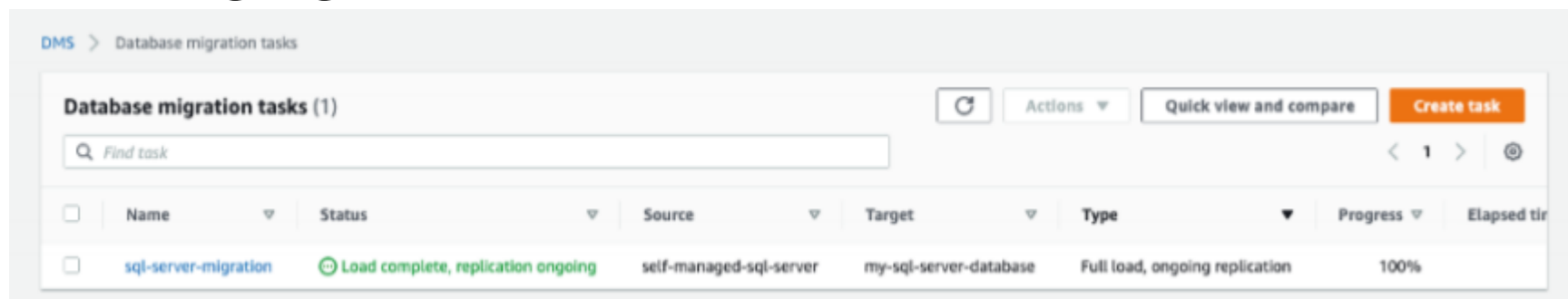
27. When you are ready, choose **Create task** to start your replication task. After you create your task, your task is shown in the Database migration tasks section with a status of **Creating**.



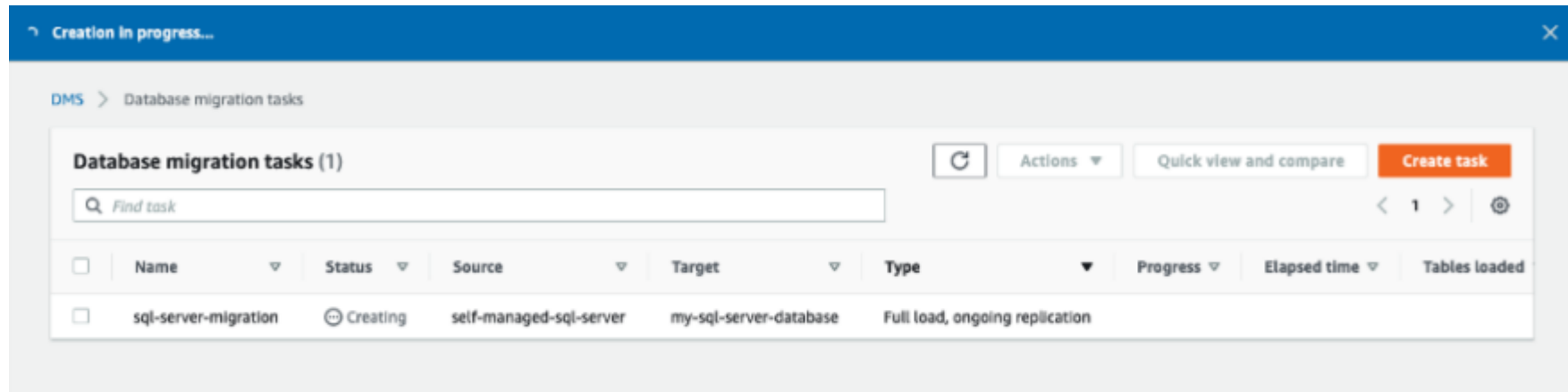
28. After the task is initialized, its status is **Starting**



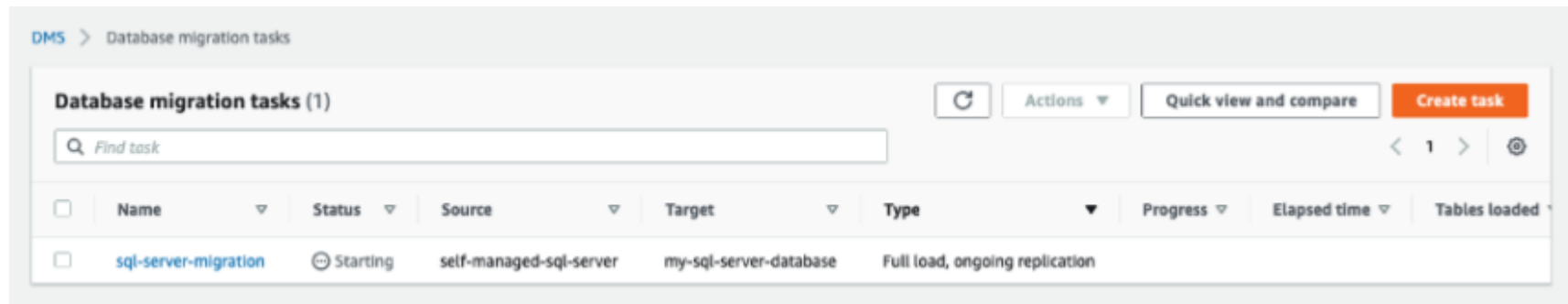
29. After the migration of existing data is complete, it shows a status of **Load complete, replication ongoing**.



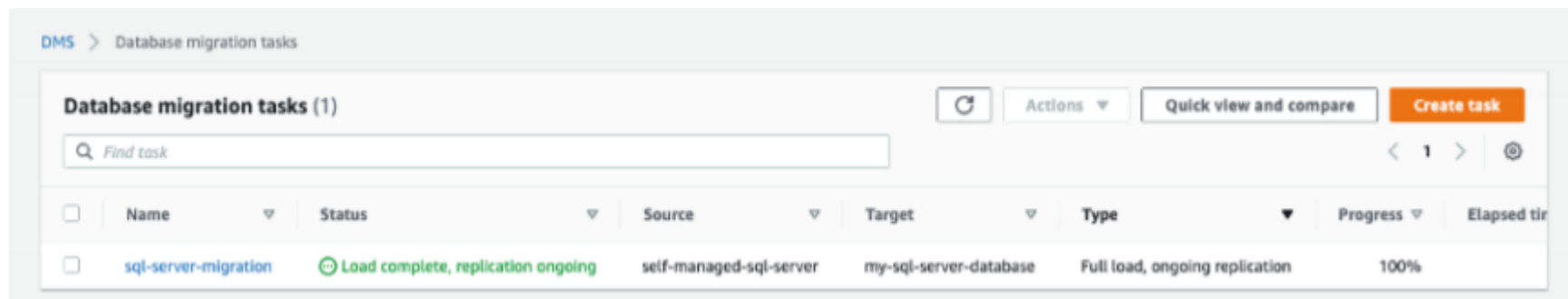
27. When you are ready, choose **Create task** to start your replication task. After you create your task, your task is shown in the Database migration tasks section with a status of **Creating**.



28. After the task is initialized, its status is **Starting**



29. After the migration of existing data is complete, it shows a status of **Load complete, replication ongoing**.



27. Then you will get Failed status, because instance has not enough memory.

DMS > Database migration tasks

### Database migration tasks (1)

Find database migration tasks

<input type="checkbox"/>	Identifier	Status	Progress	Type	Source
<input type="checkbox"/>	sql-server-migration	<span>Failed</span>	<div><div></div>100%</div>	Full load, ongoing replication	source

### Summary

Status	Type	Source	Target
<span>Failed</span>	Full load, ongoing replication	source-my-sql-database-1	my-sql-server-database-1

[Overview details](#) | [Table statistics](#) | [CloudWatch metrics](#) | [Mapping rules](#) | [Premigration assessments](#) New | [Assessment results](#) | [Tags](#)

### Overview details

#### Basic configuration

Task ARN

arn:aws:dms:us-east-2:832025209844:task:GNR2HXGHGUF6UM5O6FLYH4GQXBIPBIVCJCBUSJI

Progress

0%

Created

Replication instance

my-replication-instance

Last failure message

Last Error Not enough memory to allocate. Stop Reason FATAL\_ERROR Error Level FATAL


Started



## 27. If you get problems with database connections


### Overview details

#### Basic configuration

Task ARN  
arn:aws:dms:us-east-2:832025209844:task:SSQ7QUZYAPY5F2ZFQ6VDYDSOGI3T57NIZXPFMZV 

Progress  
 100%  
Failed

Created  
17.10.2020, 13:10:10 GMT+0400

Stopped  
17.10.2020, 13:36:55 GMT+0400 

#### Replication instance

-

#### Last failure message

Last Error Failed to connect to database. Task error notification received from subtask 0, thread 0 [reptask/replicationtask.c:2801] [1020414] Error 2019 (Can't initialize character set unknown (path: compiled\_in)) connecting to MySQL server 'source-my-sql-database-1.czbmyderbsia.us-east-2.rds.amazonaws.com'; Errors in MySQL server binary logging configuration. Follow all prerequisites for 'MySQL as a source in DMS' from [https://docs.aws.amazon.com/dms/latest/userguide/CHAP\\_Source.MySQL.html](https://docs.aws.amazon.com/dms/latest/userguide/CHAP_Source.MySQL.html) or 'MySQL as a target in DMS' from [https://docs.aws.amazon.com/dms/latest/userguide/CHAP\\_Target.MySQL.html](https://docs.aws.amazon.com/dms/latest/userguide/CHAP_Target.MySQL.html) ; Failed while preparing stream component 'st\_0\_TF7G75M3TBWX2YWTI6OONVPKIGJ4NYBW77QI46Y'; Cannot initialize subtask; Stream component 'st\_0\_TF7G75M3TBWX2YWTI6OONVPKIGJ4NYBW77QI46Y' terminated [reptask/replicationtask.c:2808] [1020414] Stop Reason RECOVERABLE\_ERROR Error Level RECOVERABLE

... you can add security rule which allow access from every address. Of course it is less secure.

### sg-051b32b6ac2e6adf6 - sql-server-database

#### Details

Security group name sql-server-database	Security group ID sg-051b32b6ac2e6adf6	Description Created by RDS management console
Owner 832025209844	Inbound rules count 4 Permission entries	Outbound rules count 1 Permission entry

Inbound rules

Outbound rules

Tags

#### Inbound rules

Type	Protocol	Port range	Source
All traffic	All	All	0.0.0.0/0
MYSQL/Aurora	TCP	3306	sg-051b32b6ac2e6adf6 (sql-server-database)
MYSQL/Aurora	TCP	3306	sg-0c5f1a36391004761 (default)
Custom TCP	TCP	0	sg-02d30c3b837da0cdd (launch-wizard-1)