

AWS DMS Migration Project Documentation

Migrating MySQL RDS to Aurora MySQL using AWS DMS

1. Project Overview

This project demonstrates the migration of a **MySQL RDS database** to an **Aurora MySQL RDS database** using **AWS Database Migration Service (DMS)**.

2. Architecture Setup

- **Public EC2 Instance (Bastion Host)**
 - Acts as a jump server for SSH access to the private EC2 instance.
- **Private EC2 Instance**
 - Used to connect to the RDS databases for verification.
- **Source Database: MySQL RDS**
- **Target Database: Aurora MySQL RDS**
- **Migration Tool: AWS DMS**

3. Security Group Configuration

- **Bastion Host SG (`bastion-sg`)**
 - Allows **SSH (Port 22)** from **My IP**.
- **Private EC2 SG (`private-ec2-sg`)**
 - Allows **SSH (Port 22)** from **Bastion Host SG (`bastion-sg`)**.
- **RDS SG (`rds-sg`)**
 - Allows **MySQL (Port 3306)** from **Private EC2 SG (`private-ec2-sg`)**.

And this the two ec2 instance created with security group configuration mentioned

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with 'EC2' selected under 'Instances'. The main area displays a table of instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status
private-ec2	i-03b3f4729cc5790d5	Running	t2.micro	2/2 checks passed	View alarms
public-ec2	i-054e112d73ba8454a	Running	t2.micro	2/2 checks passed	View alarms

Below the table, a message says 'Select an instance'. At the bottom of the page, there's a standard Windows taskbar with icons for File Explorer, Edge, and other applications.

And also created

- **Source Database: MySQL RDS**
- **Target Database: Aurora MySQL RDS**

The screenshot shows the AWS RDS Databases page. On the left, there's a sidebar with 'Amazon RDS' selected under 'Databases'. The main area displays a table of databases:

DB identifier	Status	Role	Engine	Region ...	Size
auroramysql-db-for-dms-project-target-	Available	Regional c...	Aurora My...	us-east-1	2 instances
auroramysql-db-for-dms-project-targe	Available	Writer ins...	Aurora My...	us-east-1b	db.t3.medium
auroramysql-db-for-dms-project-targe	Available	Reader ins...	Aurora My...	us-east-1a	db.t3.medium
mysql-db-for-dms-project-sourcedb	Available	Instance	MySQL Co...	us-east-1b	db.t3.micro

Below the table, a message says 'Subnet groups', 'Parameter groups', and 'Option groups'. At the bottom of the page, there's a standard Windows taskbar with icons for File Explorer, Edge, and other applications.

4 .MySQL Database Setup (Source RDS)

- **Creating the Database**

- CREATE DATABASE company_db;
- USE company_db;

- **Creating Tables**

- CREATE TABLE employees (
• emp_id INT AUTO_INCREMENT PRIMARY KEY,
• first_name VARCHAR(50) NOT NULL,
• last_name VARCHAR(50) NOT NULL,
• department VARCHAR(50),
• salary DECIMAL(10,2),
• hire_date DATE
•);

- CREATE TABLE departments (
• dept_id INT AUTO_INCREMENT PRIMARY KEY,
• dept_name VARCHAR(50) NOT NULL,
• location VARCHAR(50)
•);

- **Inserting Sample Data**

- INSERT INTO employees (first_name, last_name, department, salary, hire_date) VALUES
• ('John', 'Doe', 'IT', 60000.00, '2022-01-15'),
• ('Jane', 'Smith', 'HR', 55000.00, '2021-07-23'),
• ('Robert', 'Brown', 'Finance', 75000.00, '2020-03-10'),
• ('Alice', 'Johnson', 'Marketing', 58000.00, '2023-06-01');

- INSERT INTO departments (dept_name, location) VALUES
• ('IT', 'New York'),
• ('HR', 'Los Angeles'),
• ('Finance', 'Chicago'),
• ('Marketing', 'San Francisco');

- **Verifying Data**

- SELECT * FROM employees;
- SELECT * FROM departments;

```
[ec2-user@ip-10-0-138-79 ~]$ mysql -h mysql-db-for-dms-project-sourcedb.ca708qu08j9s.us-east-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 59
Server version: 8.0.40 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> CREATE DATABASE company_db;
Query OK, 1 row affected (0.032 sec)

MySQL [(none)]> USE company_db;
Database changed
MySQL [company_db]> CREATE TABLE employees (
    ->     emp_id INT AUTO_INCREMENT PRIMARY KEY,
    ->     first_name VARCHAR(50) NOT NULL,
    ->     last_name VARCHAR(50) NOT NULL,
    ->     department VARCHAR(50),
    ->     salary DECIMAL(10,2),
    ->     hire_date DATE
    -> );
Query OK, 0 rows affected (0.148 sec)

MySQL [company_db]> CREATE TABLE departments (
    ->     dept_id INT AUTO_INCREMENT PRIMARY KEY,
    ->     dept_name VARCHAR(50) NOT NULL,
    ->     location VARCHAR(50)
    -> );
Query OK, 0 rows affected (0.061 sec)

MySQL [company_db]> INSERT INTO employees (first_name, last_name, department, salary, hire_date) VALUES
    ->     ('John', 'Doe', 'IT', 60000.00, '2022-01-15'),
    ->     ('Jane', 'Smith', 'HR', 55000.00, '2021-07-23'),
    ->     ('Robert', 'Brown', 'Finance', 75000.00, '2020-03-10'),
    ->     ('Alice', 'Johnson', 'Marketing', 58000.00, '2023-06-01');
Query OK, 4 rows affected (0.007 sec)
Records: 4  Duplicates: 0  Warnings: 0

MySQL [company_db]> INSERT INTO departments (dept_name, location)
    ->     ('IT', 'New York'),
    ->     ('HR', 'Los Angeles'),
    ->     ('Finance', 'Chicago'),
```

```
ec2-user@ip-10-0-138-79: ~
->     dept_name VARCHAR(50) NOT NULL,
->     location VARCHAR(50)
-> );
Query OK, 0 rows affected (0.061 sec)

MySQL [company_db]> INSERT INTO employees (first_name, last_name, department, salary, hire_date) VALUES
-> ('John', 'Doe', 'IT', 60000.00, '2022-01-15'),
-> ('Jane', 'Smith', 'HR', 55000.00, '2021-07-23'),
-> ('Robert', 'Brown', 'Finance', 75000.00, '2020-03-10'),
-> ('Alice', 'Johnson', 'Marketing', 58000.00, '2023-06-01');
Query OK, 4 rows affected (0.007 sec)
Records: 4 Duplicates: 0 Warnings: 0

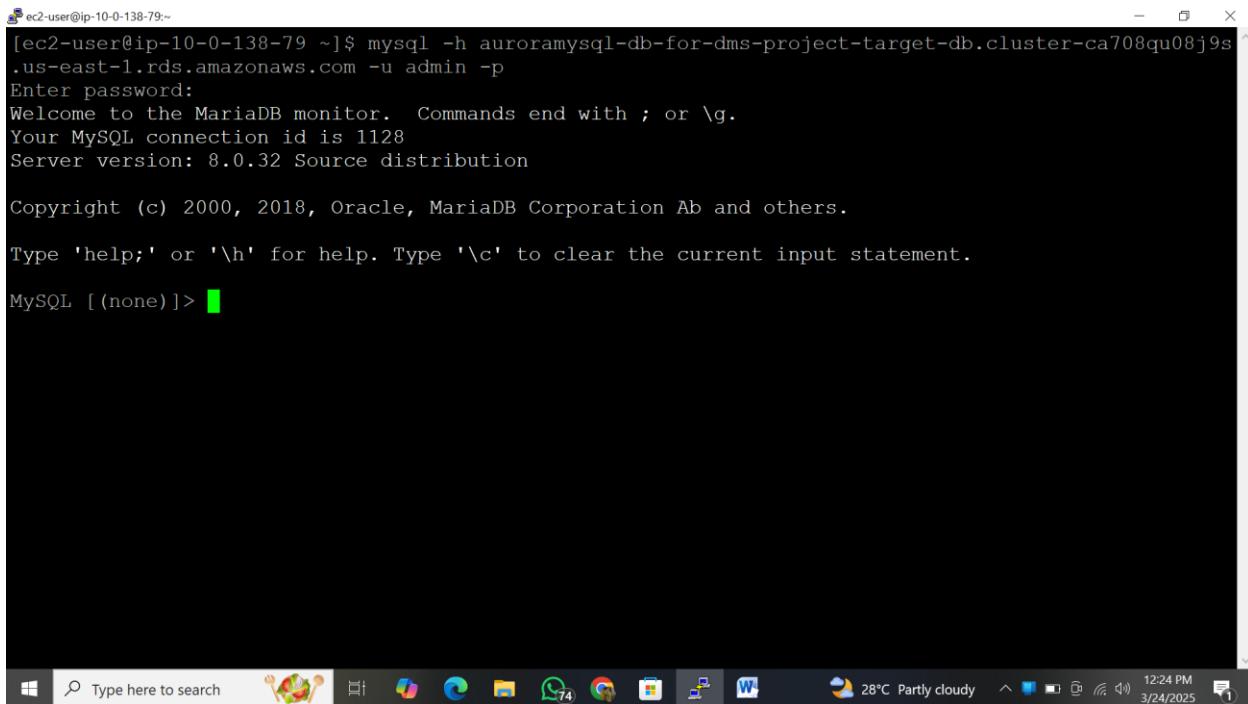
MySQL [company_db]> INSERT INTO departments (dept_name, location) VALUES
-> ('IT', 'New York'),
-> ('HR', 'Los Angeles'),
-> ('Finance', 'Chicago'),
-> ('Marketing', 'San Francisco');
Query OK, 4 rows affected (0.005 sec)
Records: 4 Duplicates: 0 Warnings: 0

MySQL [company_db]> SELECT * FROM employees;
+-----+-----+-----+-----+-----+
| emp_id | first_name | last_name | department | salary | hire_date |
+-----+-----+-----+-----+-----+
| 1 | John | Doe | IT | 60000.00 | 2022-01-15 |
| 2 | Jane | Smith | HR | 55000.00 | 2021-07-23 |
| 3 | Robert | Brown | Finance | 75000.00 | 2020-03-10 |
| 4 | Alice | Johnson | Marketing | 58000.00 | 2023-06-01 |
+-----+-----+-----+-----+-----+
4 rows in set (0.001 sec)

MySQL [company_db]> SELECT * FROM departments;
+-----+-----+-----+
| dept_id | dept_name | location |
+-----+-----+-----+
| 1 | IT | New York |
| 2 | HR | Los Angeles |
| 3 | Finance | Chicago |
| 4 | Marketing | San Francisco |
+-----+-----+-----+
4 rows in set (0.001 sec)

MySQL [company_db]>
```

And This is the aurora mysql db

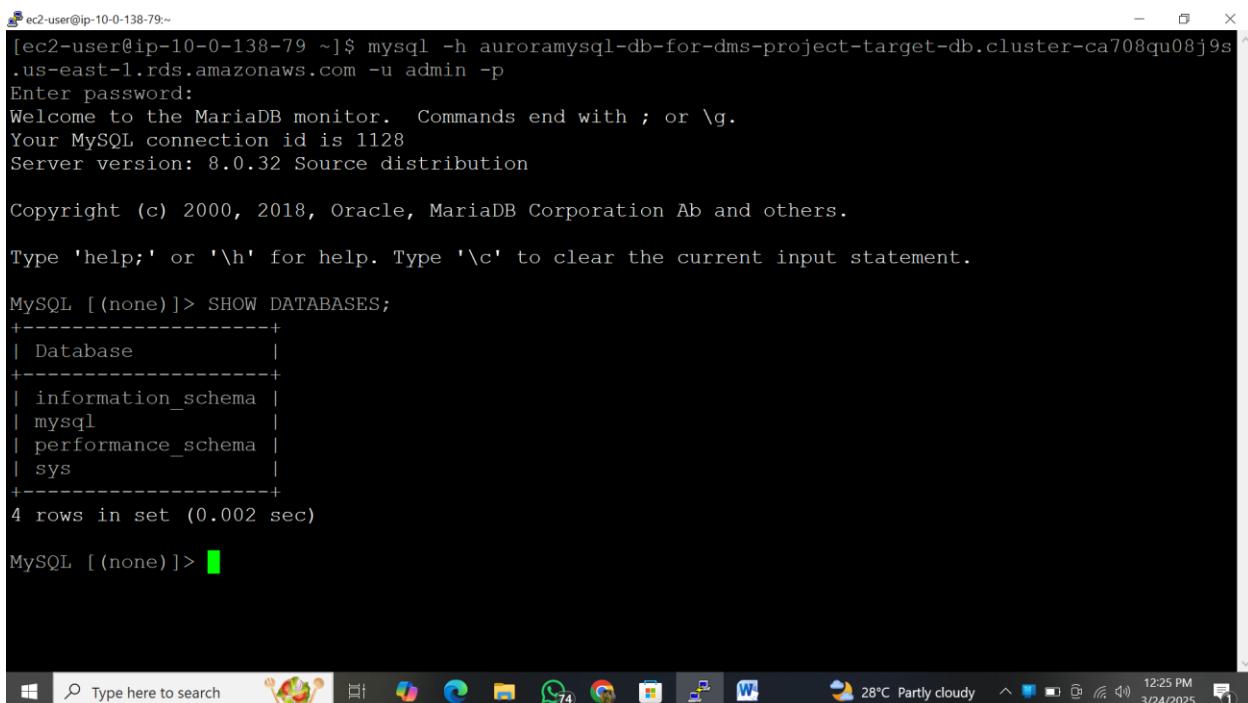


```
[ec2-user@ip-10-0-138-79 ~]$ mysql -h auroramysql-db-for-dms-project-target-db.cluster-ca708qu08j9s.us-east-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 1128
Server version: 8.0.32 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> 
```



```
[ec2-user@ip-10-0-138-79 ~]$ mysql -h auroramysql-db-for-dms-project-target-db.cluster-ca708qu08j9s.us-east-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 1128
Server version: 8.0.32 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> SHOW DATABASES;
+-----+
| Database      |
+-----+
| information_schema |
| mysql          |
| performance_schema |
| sys            |
+-----+
4 rows in set (0.002 sec)

MySQL [(none)]> 
```

Has you see there is no database and has well as no table because this the target db we have migrate the rds mysql db data to this db let begin creating AWS DMS

5. AWS DMS Migration Steps

1. Create a DMS Replication Instance

- Choose an instance size based on workload.

2. Create Source Endpoint (MySQL RDS)

- Use RDS endpoint, username, and password.

3. Create Target Endpoint (Aurora MySQL RDS)

- Use Aurora endpoint, username, and password.

4. Create a DMS Migration Task

- Select **Full Load + Ongoing Replication**.

5. Run & Monitor the Migration Task

- Check for errors and validate data in the target database.

The screenshot shows the AWS DMS console interface. The top navigation bar includes tabs for VPC, Databases, Database Migration, Instances, RDS SG Config, and other video-related items. A blue banner at the top asks for user experience feedback with five stars. The main content area is titled "Subnet groups (0) Info". It features a search bar labeled "Find subnet group" and a table with columns for "Name" and "Status". A message states "Empty subnet group table" and "You don't have any subnet groups." A prominent orange button labeled "Create subnet group" is located at the bottom of this section. On the left sidebar, under the "DMS" section, "Subnet groups" is selected. Other options like "Monitor", "Certificates", and "New feature announcements" are also listed. The bottom of the screen shows the Windows taskbar with various pinned icons and system status information.

IN DMS console first create a subnet group for the replication instance in my case I deployed in a subnet where my db is

The screenshot shows the AWS DMS Subnet groups configuration page. On the left, there's a sidebar with options like Monitor, Subnet groups (which is selected), and VPC. The main area is titled "Subnet group configuration" and contains fields for "Name" (set to "replication-subnet-group"), "Description" (set to "replication-subnet-group"), and "VPC" (set to "vpc-04b5df1b2f8e057fa - project-dms-vpc"). Below these, there's a section titled "Add subnets" where two subnets are listed: "subnet-Off540d7fc2122dc - project-dms-subnet-private4-us-east-1b" and "subnet-0eedf7ba3d8e279ab - project-dms-subnet-private3-us-east-1a". At the bottom right are "Cancel" and "Create subnet group" buttons.

1.

This screenshot is identical to the one above, showing the "Subnet group configuration" page. The "Add subnets" section now shows the two subnets listed with small "X" icons next to them, indicating they have been added to the group. The "Create subnet group" button is visible at the bottom right.

Click create

NOTE : before creating subnet group you need to create IAM role called dms-vpc-role and we should add this policy

- AmazonDMSVPCManagementRole

Your IAM user must be able to pass this role to AWS DMS.

1. Go to AWS IAM Console → [IAM Users](#)
2. Click your **IAM user**
3. Go to **Permissions** → Click **Add Permissions**
4. Click **Create Inline Policy**
5. Select **JSON tab** and paste this:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": "iam:PassRole",  
      "Resource": "  
arn:aws:iam::688567302802:role/dms-vpc-role"  
    }  
  ]  
}
```

Then only we able to create the subnet group itself

The screenshot shows the AWS DMS console with the URL https://us-east-1.console.aws.amazon.com/dms/v2/home?region=us-east-1#subnetGroup. The left sidebar has sections for Database migration tasks, Serverless replications, Monitor (Enhanced monitoring, Events, Event subscriptions), Certificates, and Subnet groups (which is selected). The main content area displays 'Subnet groups (1) Info' with a table:

Name	Status	VPC ID
replication-subnet-group	Complete	vpc-04b5df1b2f8e057fa

At the bottom, there's a Windows taskbar with icons for CloudShell, Feedback, Start, File Explorer, Edge, Task View, File, Word, and Power BI.

Know we want to replication instance

NOTE:

before creating EC2 Instance you need to create IAM role called dms-cloudwatch-logs-role and Search for and Attach the following policy:

* **AmazonDMSCloudWatchLogsRole**

Then only you able create replication instance.

The screenshot shows the 'Create replication instance' page in the AWS DMS console. A yellow warning box at the top left states: '⚠ The required IAM role dms-cloudwatch-logs-role doesn't exist. This role allows DMS to publish logs to Amazon CloudWatch. Go to IAM console to create this role with required permissions.' Below this, the 'Create replication instance' form is shown with the following fields:

- Name:** mysql-aurora-replication-instance
- Descriptive Amazon Resource Name (ARN) - optional:** (empty)
- Description - optional:** (empty)

On the right side, there's a 'Configuration best practices' section with a link to 'How was your experience using configuration best practices?' and a 'Provide feedback' button. Below it is a 'Topics (7)' section with a link to 'Instance configuration - Instance sizing'.

The screenshot shows the 'instance configuration' page for creating a replication instance. It includes the following sections:

- Instance class:** dms.t3.small (2 vCPUs, 2 GiB Memory)
- Engine version:** 3.5.3
- High Availability:** Dev or test workload (Single-AZ)
- Storage:** (Info)

On the right side, there are two expanded sections with links:

- VPC as your replication instance. This AMI includes tools that can help you test connectivity and troubleshoot any network-related issues.
- Connectivity and security - AWS Secrets Manager. To use AWS Secrets Manager with a replication instance in a private subnet, you need to configure additional settings, such as creating a Secrets Manager VPC endpoint, changing the replication instance security group egress rules, and adding endpoint setting secretsManagerEndpointOverride. For more information on using Secrets Manager with a replication instance in a private subnet, see the guide.

The screenshot shows the 'Create replication instance' step in the AWS DMS wizard. It's a multi-step process with the current step being 'Choose VPC security group'. The 'Maintenance' section is collapsed, and the 'Kerberos authentication - optional' section is also collapsed. A note on the right side provides information about using AWS Secrets Manager with a replication instance in a private subnet.

VPC as your replication instance. This AMI includes tools that can help you test connectivity and troubleshoot any network-related issues.

Connectivity and security - AWS Secrets Manager

To use AWS Secrets Manager with a replication instance in a private subnet, you need to configure additional settings, such as creating a Secrets Manager VPC endpoint, changing the replication instance security group egress rules, and adding endpoint setting secretsManagerEndpointOverride. For more information on using Secrets Manager with a replication instance in a private subnet, see [guide](#).

Then click create

The screenshot shows the 'Replication instances' page after a replication instance has been created. A green success message at the top states: 'The replication instance mysql-aurora-replication-instance was created successfully.' Below this, the 'Replication instances (1)' table is displayed, showing one row for 'mysql-aur...'. The table includes columns for Name, Status, VPC, Class, Engine, Availability, Network, and Public. The status is 'Available' and the engine is '3.5.3'.

Name	Status	VPC	Class	Engine	Availability	Network	Public
mysql-aur...	Available	vpc-04b5...	dms.t3.sm...	3.5.3	us-east-1a	IPv4	No

CREATING THE ENDPOINTS

(Source and target endpoints)

First source endpoint

The screenshot shows the 'Create endpoint' page in the AWS DMS console. In the 'Endpoint type' section, the 'Source endpoint' radio button is selected. Below it, there is a note: 'A source endpoint allows AWS DMS to read data from a database (on-premises or in the cloud), or from other data source such as Amazon S3.' To the right, the 'Target endpoint' radio button is shown with its note: 'A target endpoint allows AWS DMS to write data to a database, or to other data stores such as Amazon DynamoDB or Kinesis.' Below these options is a checkbox for 'Select RDS DB instance', which is unchecked. A note next to it says: 'Choose this option if the endpoint is an Amazon RDS DB instance. It provides a list of available RDS Instances from the current region.' On the right side of the page, there is a 'Configuration best practices' sidebar with a heading 'Topics (3)' and three items: 'How was your experience using configuration best practices?', 'Endpoint configuration - Supported versions', and 'Endpoint configuration - AWS Secrets Manager'. The main content area also includes sections for 'Endpoint configuration' and 'Endpoint identifier'.

This screenshot shows the same 'Create endpoint' page as the previous one, but with a key difference: the 'Select RDS DB instance' checkbox is now checked. A note below it says: 'Choose this option if the endpoint is an Amazon RDS DB instance. It provides a list of available RDS Instances from the current region.' Below this, a dropdown menu titled 'RDS Instance' shows the selected value 'mysql-db-for-dms-project-sourcedb'. The rest of the interface remains the same, including the 'Configuration best practices' sidebar and the 'Endpoint configuration' and 'Endpoint identifier' sections.

The screenshot shows the 'Create endpoint' step in the AWS DMS console. The 'Server name' is set to 'mysql-db-for-dms-project-sourcedb.ca708qu08j9s.us-east-1.rds.amazonaws'. The 'Port' is '3306'. The 'User name' is 'admin' and the 'Password' is masked. Under 'Secure Socket Layer (SSL) mode', 'none' is selected. The 'Endpoint settings' section is collapsed.

Server name
The name of the data server for the data provider.
mysql-db-for-dms-project-sourcedb.ca708qu08j9s.us-east-1.rds.amazonaws

Port
The port the database runs on for this endpoint.
3306

User name | [Info](#) **Password** | [Info](#)
admin

Secure Socket Layer (SSL) mode | [Info](#)
The type of Secure Socket Layer enforcement
none

Endpoint settings

The screenshot shows the 'Test endpoint connection - optional' step. A warning message box states: 'Your endpoint will always be created even if the connection fails. After clicking 'Run test', DMS creates the endpoint with the details you provided and attempts to connect to it. If the connection fails, you can edit the endpoint definition and test the connection again. You can also delete the endpoint manually.' A 'Run test' button is visible below the message.

Test endpoint connection - optional
Choose the replication instance to test the network and database connectivity for migration.

Replication instance
A replication instance performs the database migration.
mysql-aurora-replication-instance
Version: 3.5.3 VPC: vpc-04b5df1b2f8e057fa Public accessible: No

Warning: Your endpoint will always be created even if the connection fails.
After clicking 'Run test', DMS creates the endpoint with the details you provided and attempts to connect to it. If the connection fails, you can edit the endpoint definition and test the connection again. You can also delete the endpoint manually.

Run test

Endpoint identifier	Replication instance	Status	Message
No records found			

The screenshot shows the final status of the endpoint creation process. The message indicates that the endpoint was successfully created and tested.

CloudShell Feedback
Type here to search 26°C Mostly clear 5:17 PM 3/24/2025
Privacy Terms Cookie preferences

https://us-east-1.console.aws.amazon.com/dms/v2/home?region=us-east-1#endpoint-created

aws Search [Alt+S] United States (N. Virginia) Thaufig @ thaufig123-aws

DMS > Endpoints > Create endpoint

Endpoint created and tested successfully.

The screenshot shows the AWS DMS (Database Migration Service) console. The URL in the browser is <https://us-east-1.console.aws.amazon.com/dms/v2/home?region=us-east-1#testEndpointConnection/arn:aws:dms:us-east-1:688567302802...>. The page title is "Test endpoint connection".

The left sidebar navigation includes:

- Recommendations [New](#)
- Convert and migrate
 - Migration projects [New](#)
 - Instance profiles [New](#)
 - Data providers [New](#)
- Migrate data
 - Replication instances
 - Endpoints**
 - Database migration tasks
 - Serverless replications
- Monitor
 - Enhanced monitoring [New](#)
 - Events
 - Event subscriptions

The main content area displays the "Test endpoint connection" section. It shows a "Replication instance" dropdown set to "mysql-aurora-replication-instance" (Version: 3.5.3, VPC: vpc-04b5df1b2f8e057fa, Public accessible: No). A "Run test" button is present. Below it is a table:

Endpoint identifier	Replication instance	Status	Message
mysql-db-for-dms-project-sourcedb	mysql-aurora-replication-instance	successful	

The bottom of the screen shows a Windows taskbar with various icons and a system tray indicating 26°C, Mostly clear, 5:27 PM, and 3/24/2025.

Note : before run the test make sure the security group is configured correctly the security we created for db(mysql-db-sg) only allows mysql/aurora from privateec2-sg know we have add another rule for replication instance in (my mysql-db-sg) mysql/aurora from mysql-db-sg has we specify the same sg while creating replication instance Other wise the connectivity test files.

Creating target endpoint (same as the source endpoint)

The screenshot shows a Microsoft Edge browser window with multiple tabs open at the top. The active tab is 'https://us-east-1.console.aws.amazon.com/dms/v2/home?region=us-east-1#createNewEndpoint'. The page title is 'Create endpoint' under the 'Endpoints' section of the AWS DMS service.

Create endpoint Info

Endpoint type Info

Source endpoint
A source endpoint allows AWS DMS to read data from a database (on-premises or in the cloud), or from other data source such as Amazon S3.

Target endpoint
A target endpoint allows AWS DMS to write data to a database, or to other data stores such as Amazon DynamoDB or Kinesis.

Select RDS DB instance
Choose this option if the endpoint is an Amazon RDS DB instance. It provides a list of available RDS Instances from the current region.

Endpoint configuration

Endpoint identifier Info
A label for the endpoint to help you identify it.

Topics (3)

- [Endpoint configuration - Supported versions](#)
Review the supported database versions for your source [source](#) and [target](#) endpoints.
- [Endpoint configuration - AWS Secrets Manager](#)
AWS DMS supports authenticating with databases using AWS Secrets Manager. If your replication instance is in a private subnet make sure to add the additional configuration `secretsManagerEndpointOverride` on your endpoint setting. For more information about AWS Secret Manager, see [guide](#).
- [KMS Key](#)
To keep your endpoint data secure during migration, AWS DMS encrypts passwords and other important information using AWS Key Management Service (AWS KMS). Encryption requires an AWS KMS key. AWS DMS creates a default key for you if you don't already have one. For more information, see [guide](#).

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences Type here to search 542 PM Liam Lawson's Seat 1... 3/24/2025

The screenshot shows the 'Create endpoint' step in the AWS DMS console. The left panel displays the 'RDS instance' section, where a dropdown menu is open, showing the selected option: 'auroramysql-db-for-dms-project-target-db-instance-1'. The right panel contains detailed information about endpoint configuration, including sections for 'Endpoint configuration - AWS Secrets Manager', 'Endpoint settings - Additional configuration', and 'KMS Key'. The status bar at the bottom indicates the user is in the United States (N. Virginia) region.

The screenshot shows the 'Create endpoint' step in the AWS DMS console. The left panel displays the 'Provide access information manually' section, where the 'Server name' is set to 'auroramysql-db-for-dms-project-target-db-instance-1.ca708qu08js', the 'Port' is set to '3306', and the 'User name' and 'Password' fields are filled with 'admin' and '.....' respectively. The right panel contains detailed information about endpoint configuration, including sections for 'Endpoint configuration - AWS Secrets Manager', 'Endpoint settings - Additional configuration', and 'KMS Key'. The status bar at the bottom indicates the user is in the United States (N. Virginia) region.

Screenshot of the AWS DMS 'Create endpoint' wizard.

Endpoint identifier: amysql-db-for-dms-project-target-db-instance-1

Replication instance: mysql-aurora-replication-instance

Status: successful

Source and target endpoints:

- source endpoint: mysql-aurora-replication-instance
- target endpoint: amysql-db-for-dms-project-target-db-instance-1

Endpoint configuration - AWS Secrets Manager: AWS DMS supports authenticating with databases using AWS Secrets Manager. If your replication instance is in a private subnet make sure to add the additional configuration secretsManagerEndpointOverride on your endpoint setting. For more information about AWS Secret Manager, see [guide](#).

Endpoint settings - Additional configuration: The functionality of DMS endpoints can be customized for individual use cases and requirements using DMS extra connection settings. For more information, see [guide](#).

KMS Key: To keep your endpoint data secure during migration, AWS DMS encrypts passwords and other important information using AWS Key Management Service (AWS KMS). Encryption requires an AWS KMS key. AWS DMS creates a default key for you if you don't already have one. For more information, see [guide](#).

Create endpoint

Screenshot of the AWS DMS 'Endpoints' page.

Endpoints (2):

Name	Type	Status	Engine	Server name
auroramysql-db-for-dms-project-target-db-instance-1	Target	Active	Amazon Aurora MySQL	auroramysql-db-for-dms-project-target-db-instance-1
mysql-db-for-dms-project-sourcedb	Source	Active	MySQL	mysql-db-for-dms-project-sourcedb

Actions: Create endpoint, Actions dropdown, Delete, Edit, Copy, Share.

Filter: Find endpoint.

Navigation: 1 of 1 pages.

Left sidebar:

- Recommendations
- Convert and migrate
 - Migration projects
 - Instance profiles
 - Data providers
- Migrate data
 - Replication instances
 - Endpoints
 - Database migration tasks
 - Serverless replications
- Monitor
 - Enhanced monitoring
 - Events
 - Event subscriptions

Create a DMS Migration Task

(migrate the data from source to destination)

The screenshot shows the AWS Lambda console with the URL <https://us-east-1.console.aws.amazon.com/dms/v2/home?region=us-east-1#CreateNewTask>. The page is titled "Create database migration task". On the left, there's a "Task configuration" section with fields for "Task identifier" (set to "mysql-aurora-migration-task"), "Descriptive Amazon Resource Name (ARN)" (optional), "Replication instance", "Source database endpoint", and "Target database endpoint". On the right, there's a "Configuration best practices" sidebar with a "Topics (5)" section and a "Provide feedback" button.

This screenshot shows the same AWS Lambda console page as the first one, but with different configuration values. The "Task identifier" is now "mysql-aurora-migration-task". The "Replication instance" dropdown is set to "mysql-aurora-replication-instance - vpc-04b5df1b2f8e057fa". The "Source database endpoint" dropdown is set to "mysql-db-for-dms-project-sourcedb". The "Target database endpoint" dropdown is set to "auroramysql-db-for-dms-project-target-db-instance-1". The "Migration type" dropdown is currently set to "Info". The rest of the interface is identical to the first screenshot.

The screenshot shows the 'Create database migration task' page in the AWS DMS console. The left sidebar lists 'Migration type' (selected 'Migrate'), 'Task settings' (selected 'Wizard'), and 'Target table preparation mode' (selected 'Drop tables on target'). The right sidebar features a 'Configuration best practices' section with a question about experience and a 'Provide feedback' button, followed by a 'Topics (5)' section and a link to 'Task settings - Optimizing full load for a table'.

Migration type

- Migrate
- Migrate and replicate
- Replicate

Task settings

Editing mode

- Wizard
- JSON editor

Target table preparation mode

- Do nothing
- Drop tables on target
- Truncate

Configuration best practices

Review topics below which describe most commonly experienced configuration issues. Note that, changing source engine, target engine or migration type will update the topics below.

How was your experience using configuration best practices?

Topics (5)

[Task settings - Optimizing full load for a table](#)

To improve the performance of loading large tables, consider using the parallel load or the filter option with a separate task. [Learn more](#)

This screenshot shows the same configuration page but with different selected options. Under 'Migration type', 'Replicate' is selected. Under 'Task settings', 'JSON editor' is selected. Under 'Target table preparation mode', 'Truncate' is selected. The rest of the interface is identical to the first screenshot, including the 'Configuration best practices' sidebar and the 'Topics (5)' section.

Migration type

- Migrate
- Migrate and replicate
- Replicate

Task settings

Editing mode

- Wizard
- JSON editor

Target table preparation mode

- Do nothing
- Drop tables on target
- Truncate

Configuration best practices

Review topics below which describe most commonly experienced configuration issues. Note that, changing source engine, target engine or migration type will update the topics below.

How was your experience using configuration best practices?

Topics (5)

[Task settings - Optimizing full load for a table](#)

To improve the performance of loading large tables, consider using the parallel load or the filter option with a separate task. [Learn more](#)

Screenshot of the AWS DMS console showing the 'Create database migration task' wizard. The left pane shows schema and table details: Schema 'EnterSchema' (selected), Source name 'company_db', and Source table name 'employees'. Action is set to 'Include'. The right pane displays 'Configuration best practices' with a feedback section and a 'Topics (5)' sidebar.

Screenshot of the AWS DMS console showing the 'Create database migration task' wizard. The left pane shows 'Advanced settings' and 'Migration task startup configuration' (Automatically on create selected). The right pane displays 'Configuration best practices' with a feedback section and a 'Topics (5)' sidebar.

Click the create database migration task

The screenshot shows the AWS DMS (Database Migration Service) console. The left sidebar has sections for 'Convert and migrate' (Migration projects, Instance profiles, Data providers), 'Migrate data' (Replication instances, Endpoints, Database migration tasks, Serverless replications), 'Monitor' (Enhanced monitoring, Events, Event subscriptions), and Certificates. The main area displays 'Database migration tasks (1)' with a table showing one task named 'mysql-aurora-migration-task' with status 'Load complete' at 100% progress, labeled as 'Full load' and 'Not assessed'. The bottom navigation bar includes CloudShell, Feedback, and system status.

Lets we connect to aurora db and check
Copy the auroramysql endpoint

The screenshot shows the AWS RDS (Relational Database Service) console. The left sidebar has sections for Dashboard, Databases (selected), Query Editor, Performance insights, Snapshots, Exports in Amazon S3, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, and Option groups. The main area shows the 'Connectivity & security' tab for a database named 'auroramysql-db-for-dms-project-target-db'. It displays two endpoints: 'auroramysql-db-for-dms-project-target-db.cluster-ca708qu08j9s.us-east-1.rds.amazonaws.com' (Writer, Available) and 'auroramysql-db-for-dms-project-target-db.cluster-ro-ca708qu08j9s.us-east-1.rds.amazonaws.com' (Reader, Available). Below this, there's a 'Manage IAM roles' section with a note to 'Select IAM roles to add to this cluster'. The bottom navigation bar includes CloudShell, Feedback, and system status.

```
ec2-user@ip-10-0-138-79:~ [ec2-user@ip-10-0-138-79 ~]$ mysql -h auroramysql-db-for-dms-project-target-db.cluster-ca708qu08j9s.us-east-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 1763
Server version: 8.0.32 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| awsdsms_control |
| company_db |
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
6 rows in set (0.003 sec)

MySQL [(none)]> USE company_db;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MySQL [company_db]> SELECT * FROM employees;
+----+----+----+----+----+
| emp_id | first_name | last_name | department | salary | hire_date |
+----+----+----+----+----+
| 1 | John | Doe | IT | 60000.00 | 2022-01-15 |
| 2 | Jane | Smith | HR | 55000.00 | 2021-07-23 |
| 3 | Robert | Brown | Finance | 75000.00 | 2020-03-10 |
| 4 | Alice | Johnson | Marketing | 58000.00 | 2023-06-01 |
+----+----+----+----+----+
4 rows in set (0.002 sec)

MySQL [company_db]>
```

We