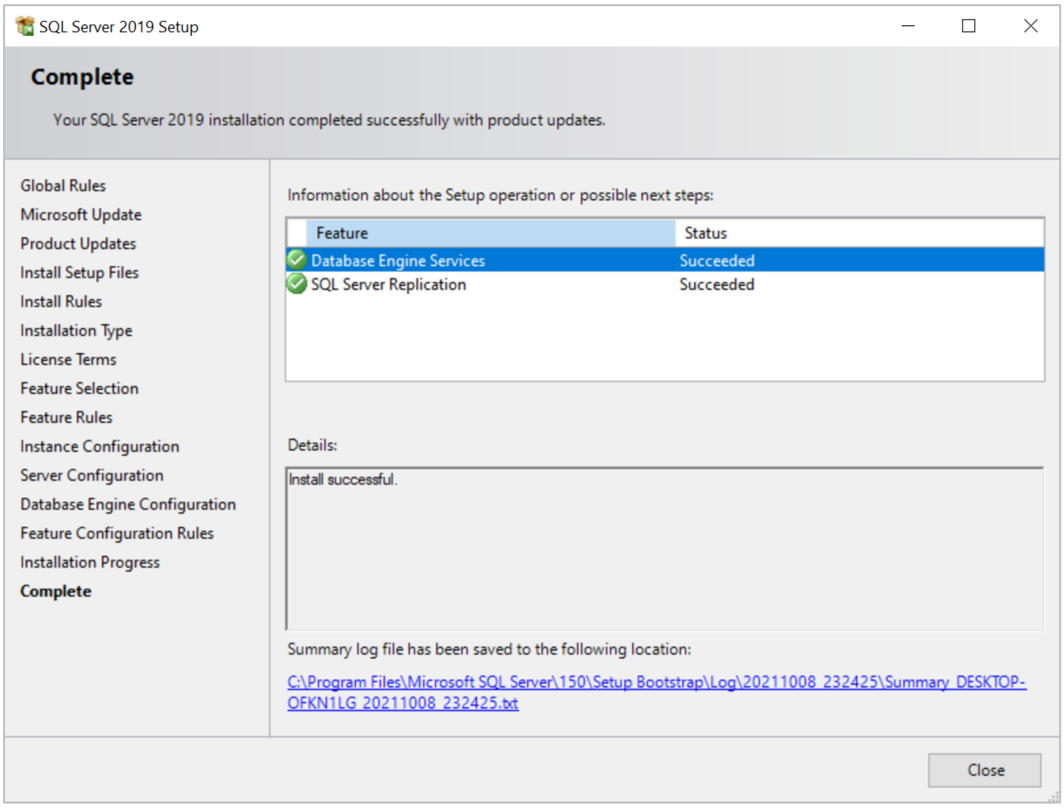
Migrating data between on-premises and cloud-based databases

1. Creating a sample database

1.1 Install a relational database on my machine

- I downloaded **SQL Server 2019 Express** and installed it on my computer.
- Open **SQL Server Installation Center**, click **Installation** on the left, click **New SQL Server stand-alone installation**, select **C:\SQL2019\Express_ENU**.
- In **SQL Server 2019 Setup**:
    - Microsoft Update: Use Microsoft Update to check for updates
    - Installation Type: Perform a new installation of SQL Server 2019
    - LIcense Terms: I accept the license terms and Privacy Statement
    - Feature Selection: Database Engine Services – SQL Server Replication
    - Instance Configuration:
        - Named instance: SQLExpress2
        - Instance ID: SQLEXPRESS2
    - Server Configuration: Keep default
    - Database Engine Configuration:
        - Server Configuration: Windows authentication mode
        - Data Directories: Keep default
        - TempDB: Keep default



Complete

Your SQL Server 2019 installation completed successfully with product updates.

Information about the Setup operation or possible next steps:

Feature	Status
✓ Database Engine Services	Succeeded
✓ SQL Server Replication	Succeeded

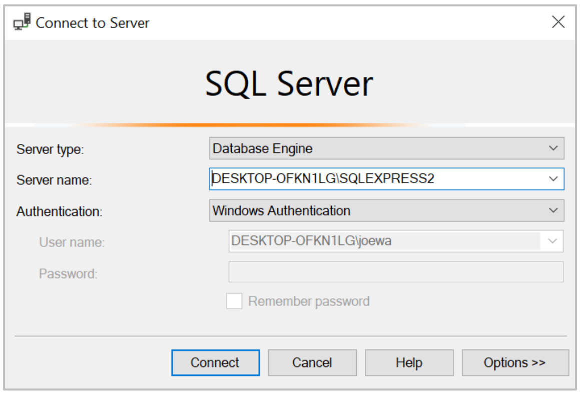
Details:

Install successful.

Summary log file has been saved to the following location:

C:\Program Files\Microsoft SQL Server\150\Setup Bootstrap\Log\20211008_232425\Summary_DESKTOP-OFKN1LG_20211008_232425.txt

- Open **SQL Server Management Studio**


Connect to Server

SQL Server

Server type: Database Engine

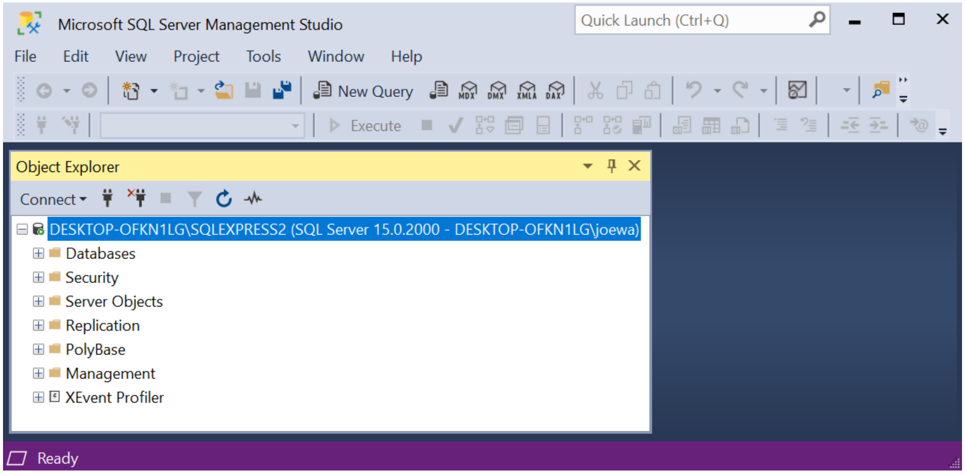
Server name: DESKTOP-OFKN1LG\SQLEXPRESS2

Authentication: Windows Authentication

User name: DESKTOP-OFKN1LG\joewa

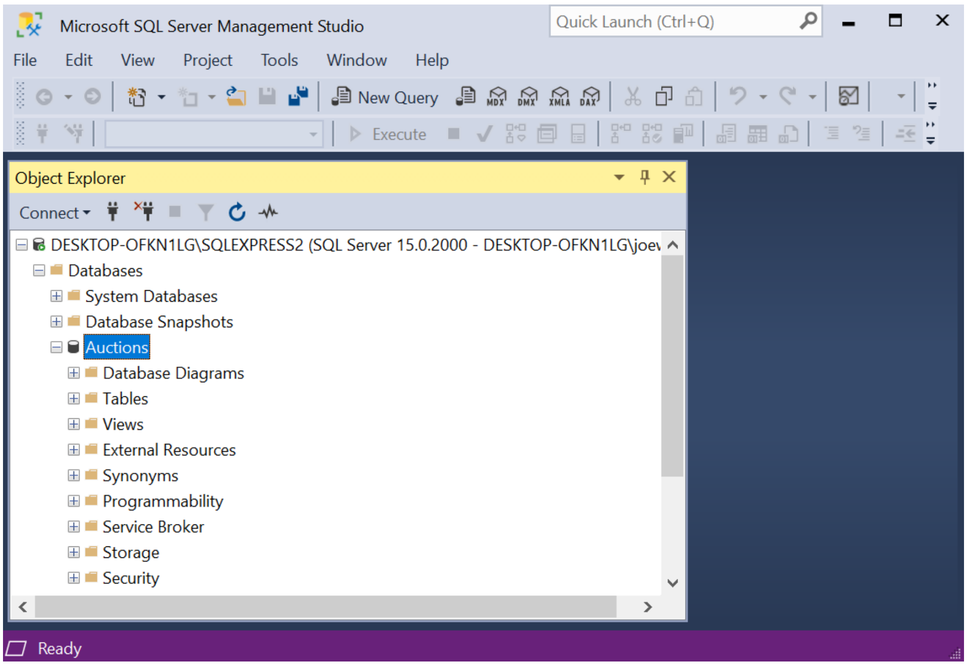
Password:

□ Remember password



1.2 Create a database titled ‘Auctions’

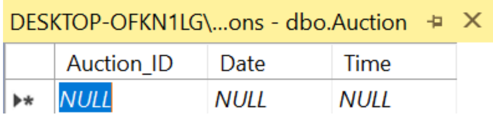
- In **Object Explorer**, right click **Databases** and click **New Database**.
  - Database name: Auctions



1.3 Create tables

- For each new table, click **New Query**.

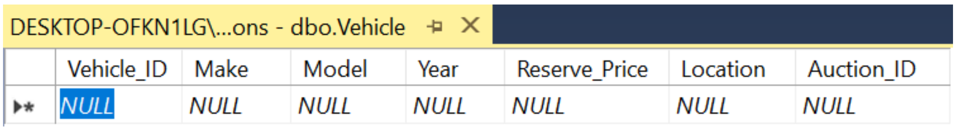
```
CREATE TABLE Auction (
        Auction_ID int IDENTITY(1001,1) PRIMARY KEY,
        Date date NOT NULL,
        Time time NOT NULL);
```



DESKTOP-OFKN1LG\...ons - dbo.Auction

	Auction_ID	Date	Time
▸*	NULL	NULL	NULL

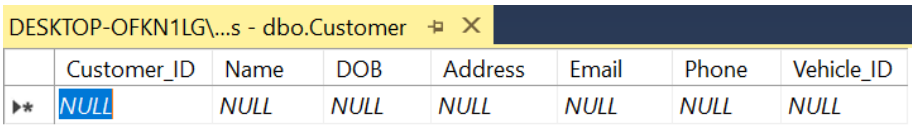
```
CREATE TABLE Vehicle (
        Vehicle_ID int IDENTITY(1001,1) PRIMARY KEY,
        Make varchar(20) NOT NULL,
        Model varchar(20) NOT NULL,
        Year int NOT NULL,
        Reserve_Price money NOT NULL,
        Location varchar(30) NOT NULL,
        Auction_ID int REFERENCES Auction(Auction_ID) );
```



DESKTOP-OFKN1LG\...ons - dbo.Vehicle

	Vehicle_ID	Make	Model	Year	Reserve_Price	Location	Auction_ID
▸*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

```
CREATE TABLE Customer (
        Customer_ID int IDENTITY(1001,1) PRIMARY KEY,
        Name varchar(50) NOT NULL,
        DOB date NOT NULL,
        Address varchar(100) NOT NULL,
        Email varchar(30) NOT NULL,
        Phone int NOT NULL,
        Vehicle_ID int REFERENCES Vehicle(Vehicle_ID) );
```



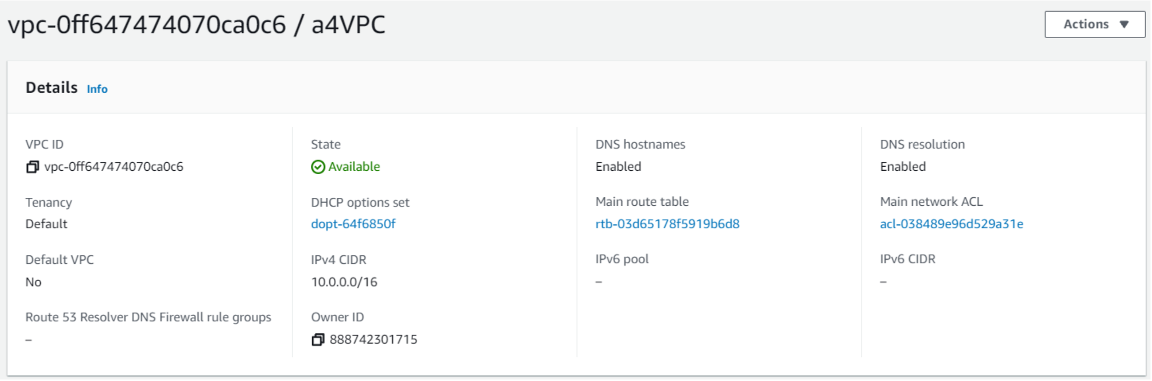
DESKTOP-OFKN1LG\...s - dbo.Customer

	Customer_ID	Name	DOB	Address	Email	Phone	Vehicle_ID
▸*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

2. Migrating sample database to the cloud

2.1 Create an RDS instance (Microsoft SQL Server) on AWS

- Log in to AWS Management Console
- Go to VPC Dashboard
- Click **Create VPC**
    - Name tag: ‘a4VPC’
    - IPv4 CIDR block: 10.0.0.0/16
    - No IPv6 CIDR block
    - Tenancy: Default
    - DNS hostnames: Enabled



vpc-0ff647474070ca0c6 / a4VPC

Details Info

VPC ID
vpc-0ff647474070ca0c6

State
Available

DNS hostnames
Enabled

DNS resolution
Enabled

Tenancy
Default

DHCP options set
dopt-64f6850f

Main route table
rtb-03d65178f5919b6d8

Main network ACL
acl-038489e96d529a31e

Default VPC
No

IPv4 CIDR
10.0.0.0/16

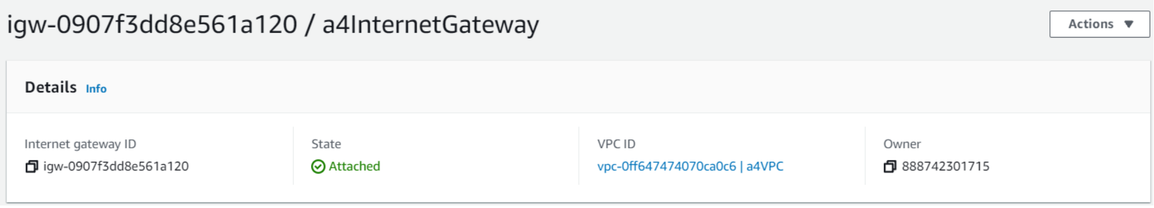
IPv6 pool
–

IPv6 CIDR
–

Route 53 Resolver DNS Firewall rule groups
–

Owner ID
888742301715

- A VPC requires an internet gateway
- In the **VPC Dashboard**, click **Internet Gateways**, **Create internet gateway**
  - Name tag: a4InternetGateway
- Click **Action**, **Attach to VPC**
  - Select: a4VPC



igw-0907f3dd8e561a120 / a4InternetGateway

Details Info

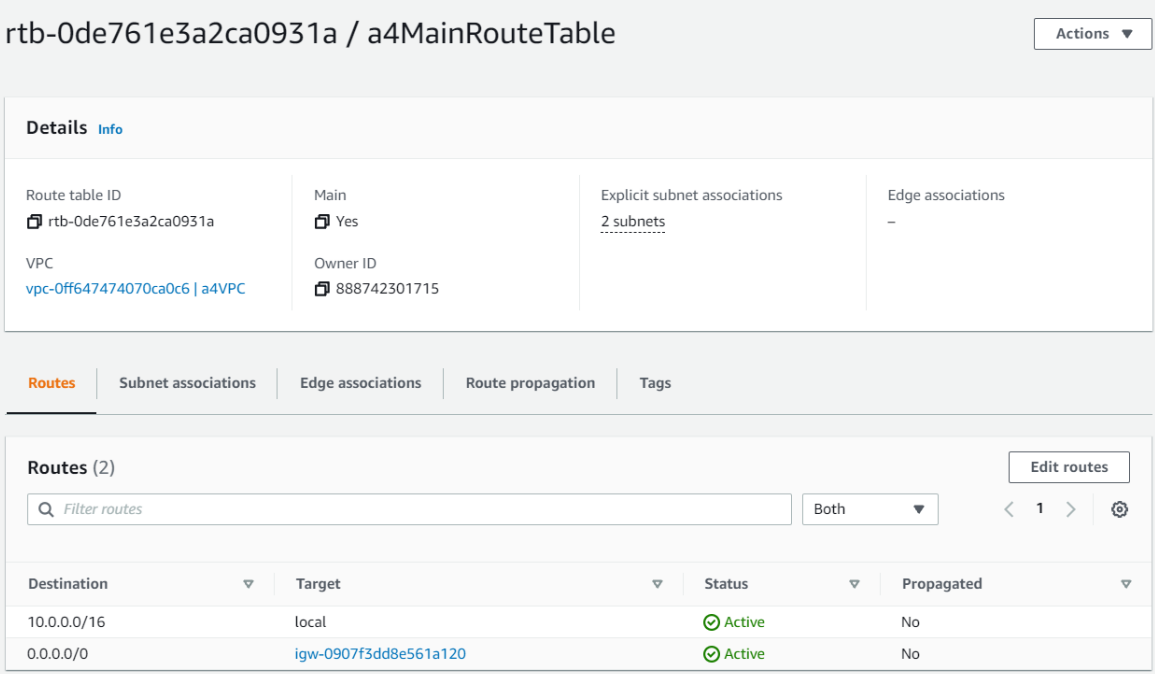
Internet gateway ID
igw-0907f3dd8e561a120

State
Attached

VPC ID
vpc-0ff647474070ca0c6 | a4VPC

Owner
888742301715

- Create a route table for a4VPC
  - In the **VPC Dashboard**, click **Route Tables**, **Create route table**
  - Name: a4MainRouteTable
  - VPC: a4VPC
  - Routes: Destination 0.0.0.0/0, Target a4InternetGateway
  - I will associate a4MainRouteTable with the 2 subnets later.



rtb-0de761e3a2ca0931a / a4MainRouteTable

Details Info

Route table ID
rtb-0de761e3a2ca0931a

Main
Yes

Explicit subnet associations
2 subnets

Edge associations
–

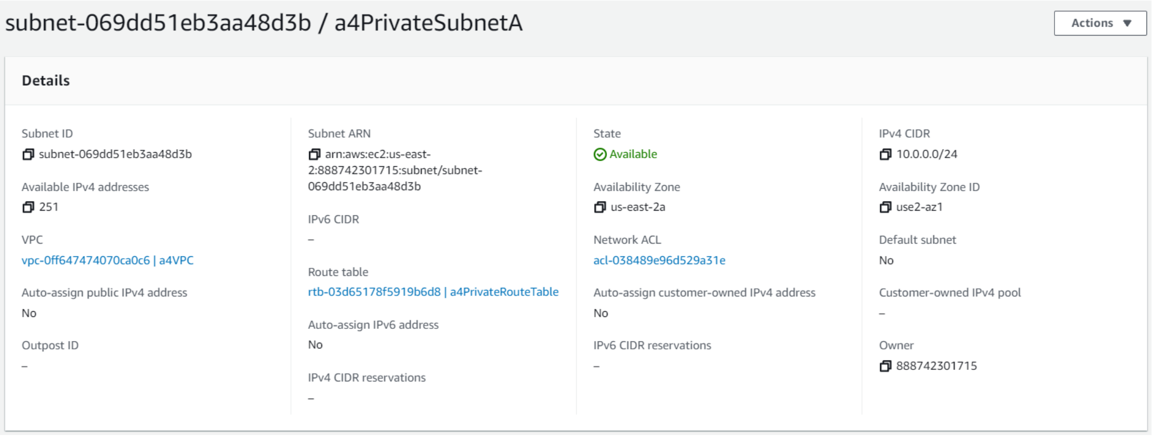
VPC
vpc-0ff647474070ca0c6 | a4VPC

Owner ID
888742301715

Routes (2)

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	igw-0907f3dd8e561a120	Active	No

- An RDS instance requires a Subnet Group with subnets in at least two availability zones.
- I will create 2 subnets here:
    - Create a4PrivateSubnetA
        - VPC: a4VPC
        - Availability Zone: US East (Ohio) / us-east-2a
        - IPv4 CIDR block: 10.0.0.0/24
        - Name: a4PrivateSubnetA
    - Create a4PrivateSubnetB
        - VPC: a4VPC
        - Availability Zone: US East (Ohio) / us-east-2b
        - IPv4 CIDR block: 10.0.1.0/24
        - Name: a4PrivateSubnetB



subnet-069dd51eb3aa48d3b / a4PrivateSubnetA

Details

Subnet ID
subnet-069dd51eb3aa48d3b

Available IPv4 addresses
251

VPC
vpc-0ff647474070ca0c6 | a4VPC

Auto-assign public IPv4 address
No

Outpost ID
–

Subnet ARN
arn:aws:ec2:us-east-2:888742301715:subnet/subnet-069dd51eb3aa48d3b

IPv6 CIDR
–

Route table
rtb-03d65178f5919b6d8 | a4PrivateRouteTable

Auto-assign IPv6 address
No

IPv4 CIDR reservations
–

State
✓ Available

Availability Zone
us-east-2a

Network ACL
acl-038489e96d529a31e

Auto-assign customer-owned IPv4 address
No

IPv6 CIDR reservations
–

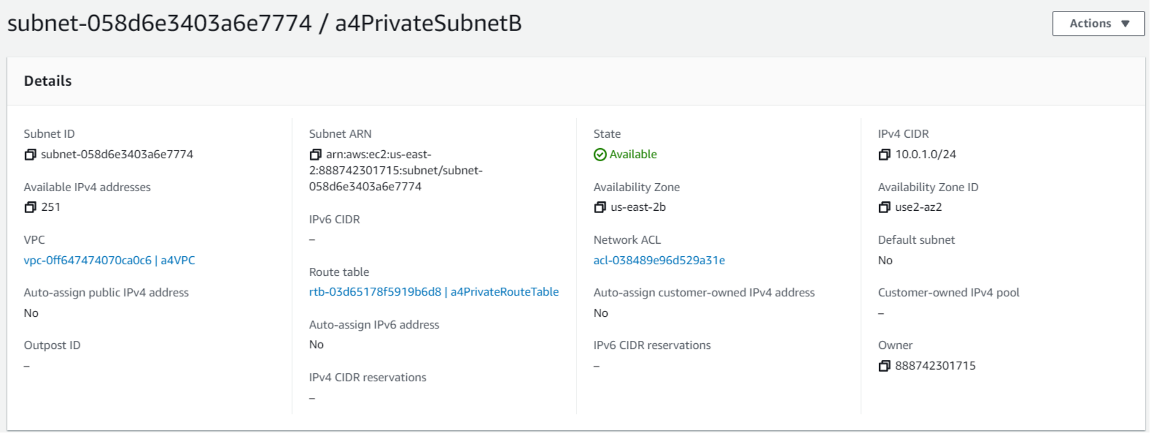
IPv4 CIDR
10.0.0.0/24

Availability Zone ID
use2-az1

Default subnet
No

Customer-owned IPv4 pool
–

Owner
888742301715



subnet-058d6e3403a6e7774 / a4PrivateSubnetB

Details

Subnet ID
subnet-058d6e3403a6e7774

Available IPv4 addresses
251

VPC
vpc-0ff647474070ca0c6 | a4VPC

Auto-assign public IPv4 address
No

Outpost ID
–

Subnet ARN
arn:aws:ec2:us-east-2:888742301715:subnet/subnet-058d6e3403a6e7774

IPv6 CIDR
–

Route table
rtb-03d65178f5919b6d8 | a4PrivateRouteTable

Auto-assign IPv6 address
No

IPv4 CIDR reservations
–

State
✓ Available

Availability Zone
us-east-2b

Network ACL
acl-038489e96d529a31e

Auto-assign customer-owned IPv4 address
No

IPv6 CIDR reservations
–

IPv4 CIDR
10.0.1.0/24

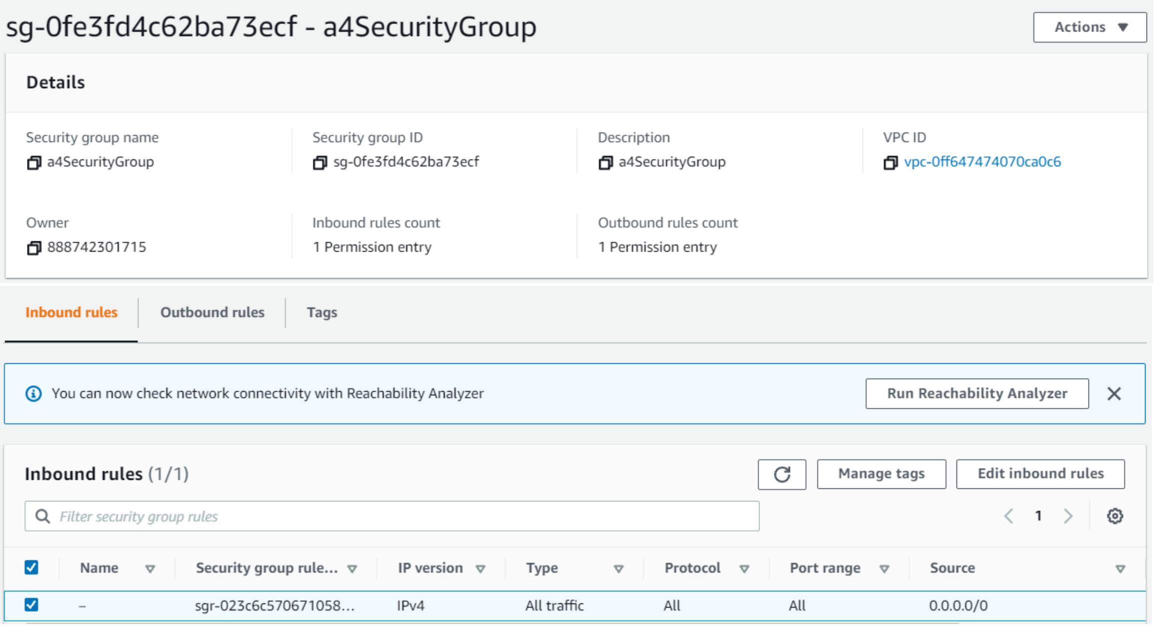
Availability Zone ID
use2-az2

Default subnet
No

Customer-owned IPv4 pool
–

Owner
888742301715

- Create a security group
    - Navigate to **VPC Dashboard**, click **Security Groups**, **Create security group**
    - Security group name: a4SecurityGroup
    - VPC: a4VPC
    - Inbound rules: All traffic Protocol All Port All Source 0.0.0.0/0
    - Outbound rules: All traffic Protocol All Port All Source 0.0.0.0/0



sg-0fe3fd4c62ba73ecf - a4SecurityGroup

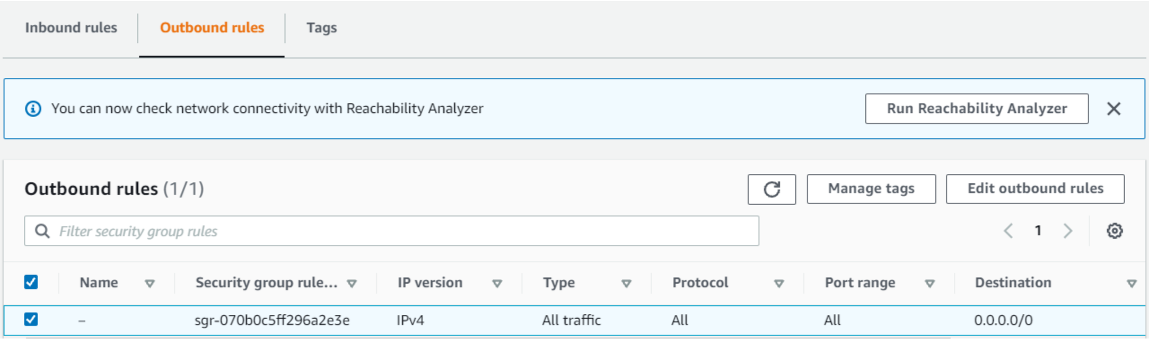
Details

Security group name	Security group ID	Description	VPC ID
a4SecurityGroup	sg-0fe3fd4c62ba73ecf	a4SecurityGroup	vpc-0ff647474070ca0c6
Owner	**Inbound rules count**	**Outbound rules count**	
888742301715	1 Permission entry	1 Permission entry	

You can now check network connectivity with Reachability Analyzer

Inbound rules (1/1)

☑	Name	Security group rule...	IP version	Type	Protocol	Port range	Source
☑	–	sgr-023c6c570671058...	IPv4	All traffic	All	All	0.0.0.0/0

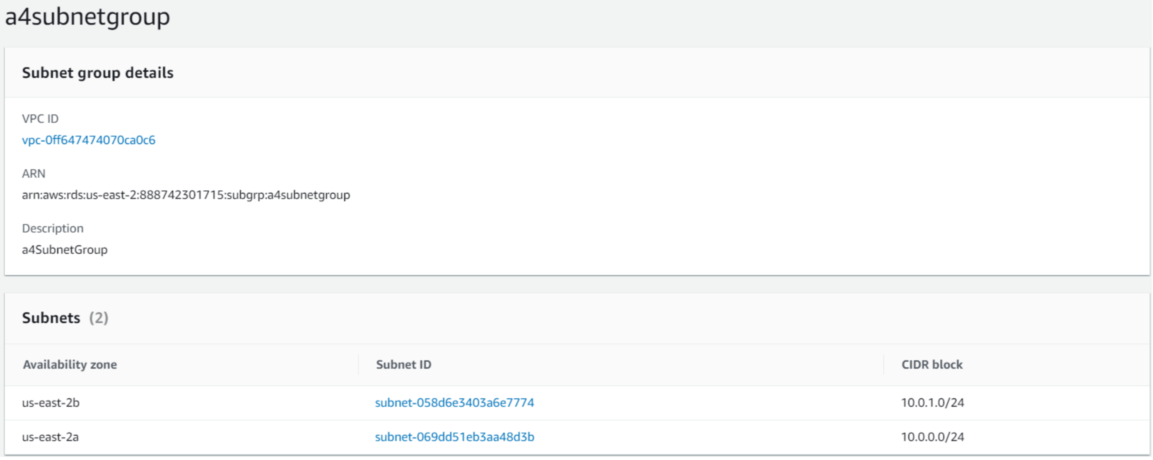


You can now check network connectivity with Reachability Analyzer

Outbound rules (1/1)

☑	Name	Security group rule...	IP version	Type	Protocol	Port range	Destination
☑	–	sgr-070b0c5ff296a2e3e	IPv4	All traffic	All	All	0.0.0.0/0

- Navigate to RDS Dashboard, click **Subnet Groups**, **Create DB Subnet Group**
  - Name: a4SubnetGroup
  - VPC: a4VPC
  - Availability Zones: us-east-2a & us-east-2b
  - Subnets: a4PrivateSubnetA & a4PrivateSubnetB



a4subnetgroup

Subnet group details

VPC ID
vpc-0ff647474070ca0c6

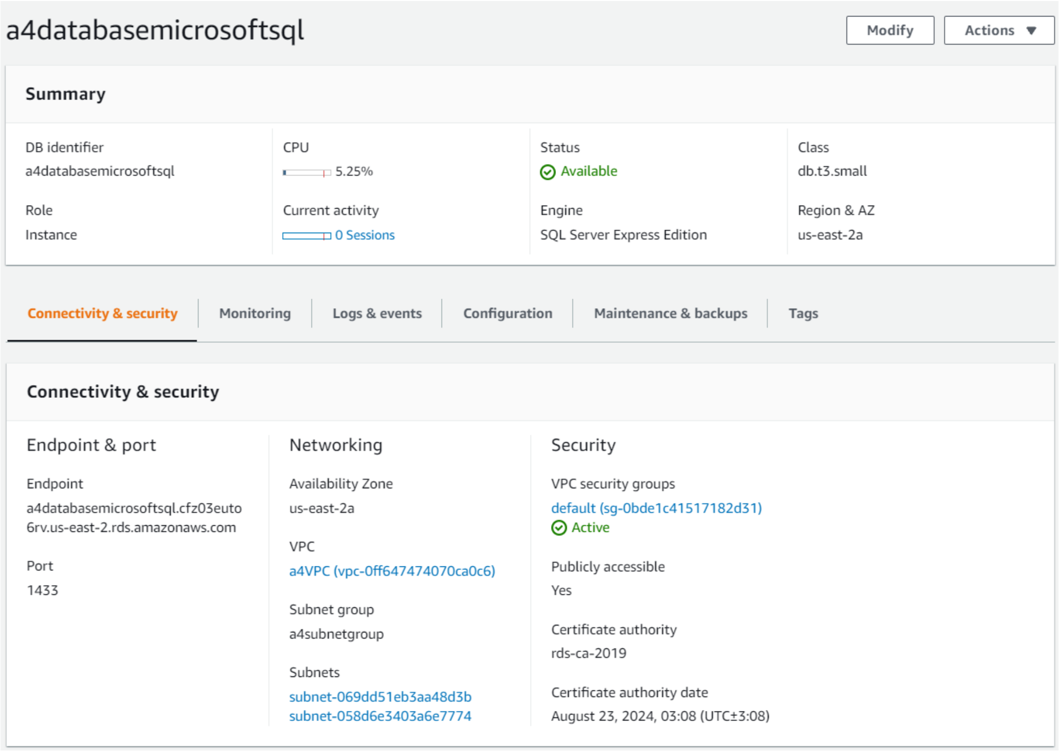
ARN
arn:aws:rds:us-east-2:888742301715:subgrp:a4subnetgroup

Description
a4SubnetGroup

Subnets (2)

Availability zone	Subnet ID	CIDR block
us-east-2b	subnet-058d6e3403a6e7774	10.0.1.0/24
us-east-2a	subnet-069dd51eb3aa48d3b	10.0.0.0/24

- Navigate to RDS Dashboard, click **Databases**, **Create database**
  - Database creation method: Standard create
  - Engine options: Microsoft SQL Server Express Edition (Version 2019 15.00.4073.23.v1)
  - Settings:
    - DB instance identifier: a4DatabaseMicrosoftSQL
    - Master username: admin
    - Master password: ********
    - DB instance class: db.t3.small
    - Storage: General Purpose SSD (gp2) 20 GiB
    - Storage autoscaling: Enabled
    - VPC: a4VPC
    - Subnet group: a4subnetgroup
    - Public access: Yes
    - VPC security group: Choose existing: default
    - Availability Zone: us-east-2a
    - Database port: 1433



a4databasemicrosoftsql

Summary

DB identifier
a4databasemicrosoftsql

Role
Instance

CPU
5.25%

Current activity
0 Sessions

Status
✓ Available

Engine
SQL Server Express Edition

Class
db.t3.small

Region & AZ
us-east-2a

Connectivity & security

Endpoint & port

Endpoint
a4databasemicrosoftsql.cfz03euto6rv.us-east-2.rds.amazonaws.com

Port
1433

Networking

Availability Zone
us-east-2a

VPC
a4VPC (vpc-0ff647474070ca0c6)

Subnet group
a4subnetgroup

Subnets
subnet-069dd51eb3aa48d3b
subnet-058d6e3403a6e7774

Security

VPC security groups
default (sg-0bde1c41517182d31)
✓ Active

Publicly accessible
Yes

Certificate authority
rds-ca-2019

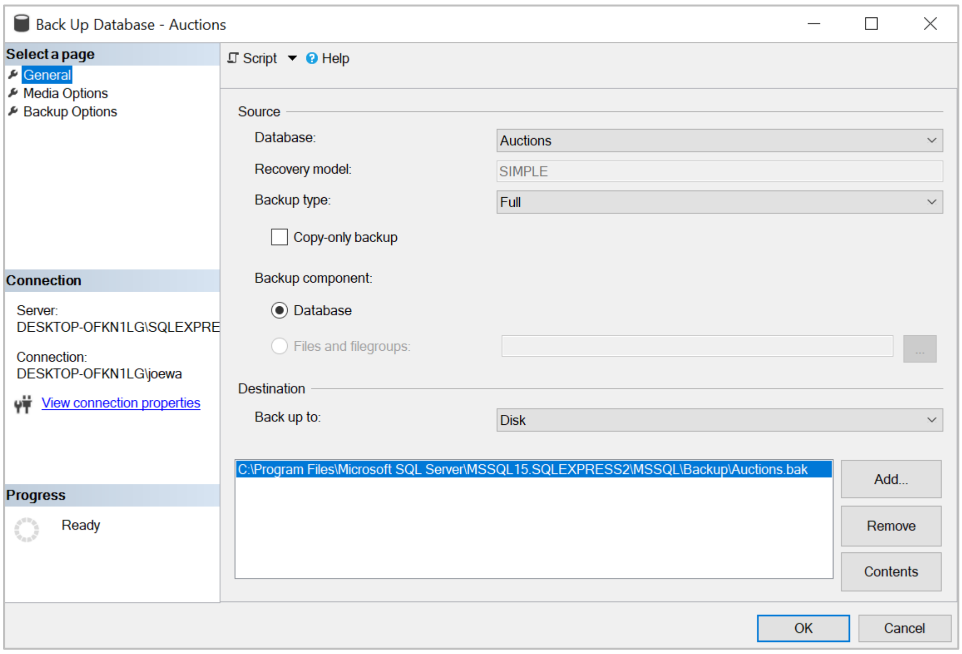
Certificate authority date
August 23, 2024, 03:08 (UTC±3:08)

2.2 Transfer the sample database from my machine to the cloud

I am using a native backup and restore method which involves the following steps:

- Making a backup of the Microsoft SQL Server database via the Microsoft SQL Server Management Studio.
- Transferring the backup file to Amazon S3.
- Restoring the backup file in Amazon RDS for SQL Server.

- Create a backup for the Auctions database
  - Open **SQL Server Management Studio**
  - Under **Databases**, right click **Auctions**, click **Tasks** and **Backup**



Source

Database: Auctions

Recovery model: SIMPLE

Backup type: Full

☐ Copy-only backup

Backup component:

● Database

○ Files and filegroups:

Destination

Back up to: Disk

C:\Program Files\Microsoft SQL Server\MSSQL15.SQLEXPRESS2\MSSQL\Backup\Auctions.bak

Connection

Server:
DESKTOP-OFKN1LG\SQLEXPRE

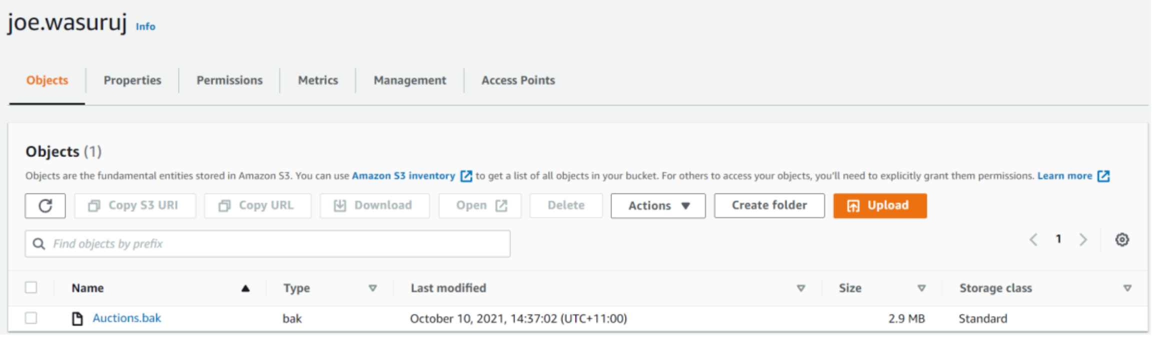
Connection:
DESKTOP-OFKN1LG\joewa

View connection properties

Progress

Ready

- Create an S3 bucket
  - Navigate to **Amazon S3**, click **Buckets**, **Create bucket**
  - Bucket name: joe.wasuruj
  - AWS Region: US East (Ohio) us-east-2
  - Block all public access: Off
  - Bucket versioning: Disabled
  - Default encryption: Disabled
- Upload the Auctions database backup file to the joe.wasuruj bucket

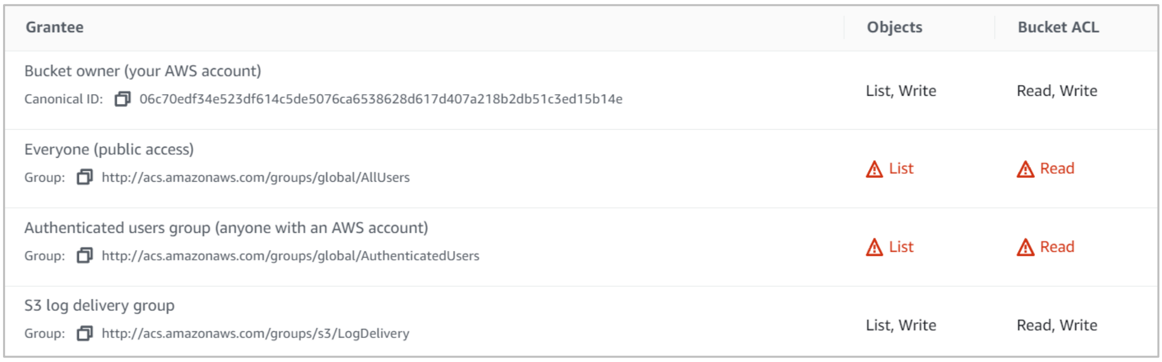


Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

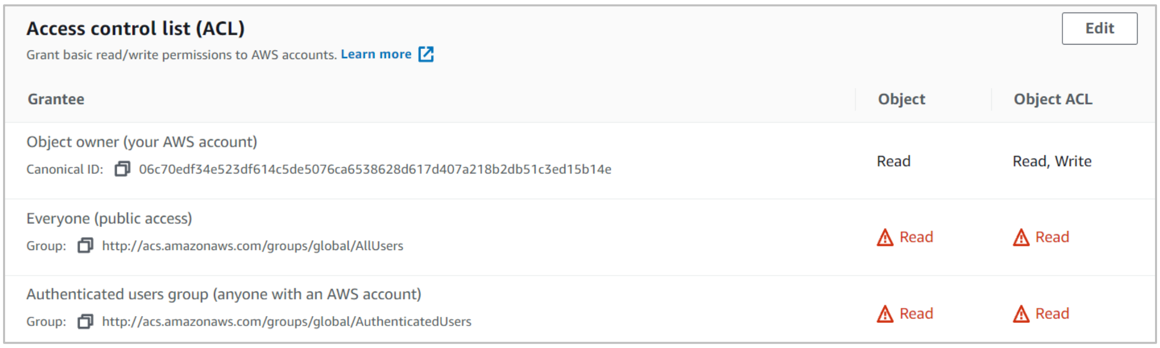
Name	Type	Last modified	Size	Storage class
Auctions.bak	bak	October 10, 2021, 14:37:02 (UTC+11:00)	2.9 MB	Standard

- Update the joe.wasuruj bucket ACL as follows:



Grantee	Objects	Bucket ACL
Bucket owner (your AWS account) Canonical ID: 06c70edf34e523df614c5de5076ca6538628d617d407a218b2db51c3ed15b14e	List, Write	Read, Write
Everyone (public access) Group: http://acs.amazonaws.com/groups/global/AllUsers	⚠ List	⚠ Read
Authenticated users group (anyone with an AWS account) Group: http://acs.amazonaws.com/groups/global/AuthenticatedUsers	⚠ List	⚠ Read
S3 log delivery group Group: http://acs.amazonaws.com/groups/s3/LogDelivery	List, Write	Read, Write

- Edit access control list for the Auctions.bak file as follows:

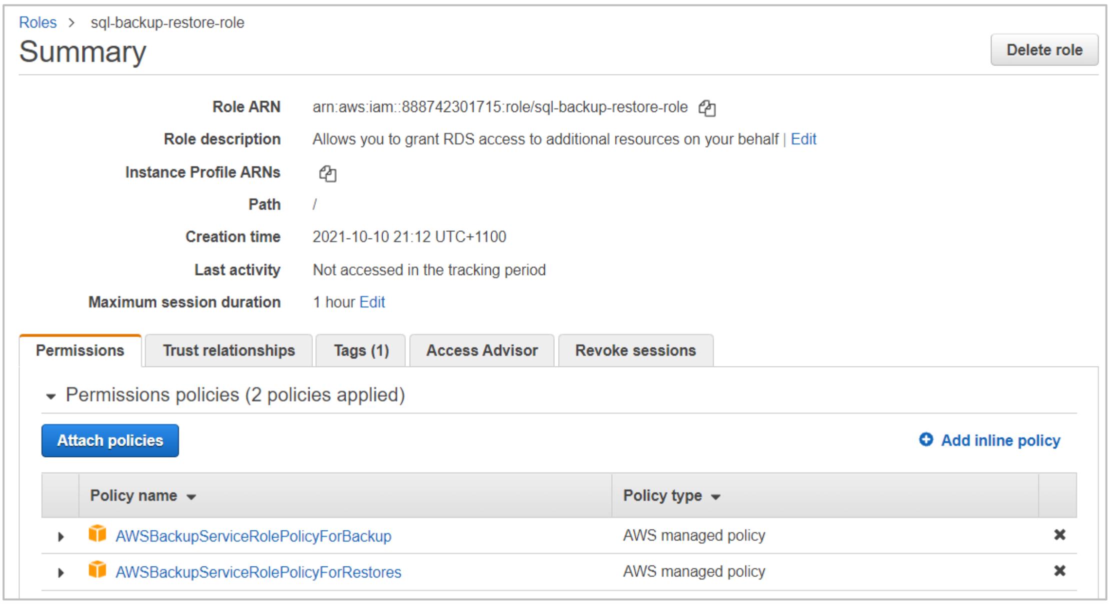


Access control list (ACL) Edit

Grant basic read/write permissions to AWS accounts. Learn more

Grantee	Object	Object ACL
Object owner (your AWS account) Canonical ID: 06c70edf34e523df614c5de5076ca6538628d617d407a218b2db51c3ed15b14e	Read	Read, Write
Everyone (public access) Group: http://acs.amazonaws.com/groups/global/AllUsers	⚠ Read	⚠ Read
Authenticated users group (anyone with an AWS account) Group: http://acs.amazonaws.com/groups/global/AuthenticatedUsers	⚠ Read	⚠ Read

- Create a new IAM role for backup and restore
  - Navigate to **IAM Dashboard**, click **Roles**, **Create role**
  - Choose a use case: AWS Backup
  - Attach permissions policies:
    - AWSBackUpServiceRolePolicyForBackup
    - AWSBackUpServiceRolePolicyForRestores
  - Tags: Name: sql-backup-restore-role
  - Role description: Allows you to grant RDS access to additional resources on your behalf



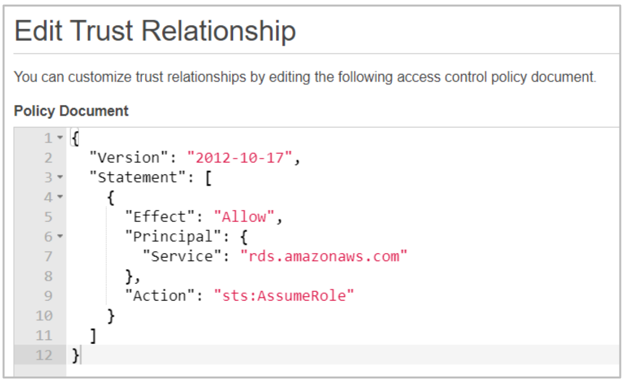
Summary

Role ARN	arn:aws:iam::888742301715:role/sql-backup-restore-role
Role description	Allows you to grant RDS access to additional resources on your behalf | Edit
Instance Profile ARNs	
Path	/
Creation time	2021-10-10 21:12 UTC+1100
Last activity	Not accessed in the tracking period
Maximum session duration	1 hour Edit

Permissions policies (2 policies applied)

Policy name	Policy type
AWSBackupServiceRolePolicyForBackup	AWS managed policy
AWSBackupServiceRolePolicyForRestores	AWS managed policy

- In the **Trust relationships** tab, edit **Service** to **rds.amazonaws.com**



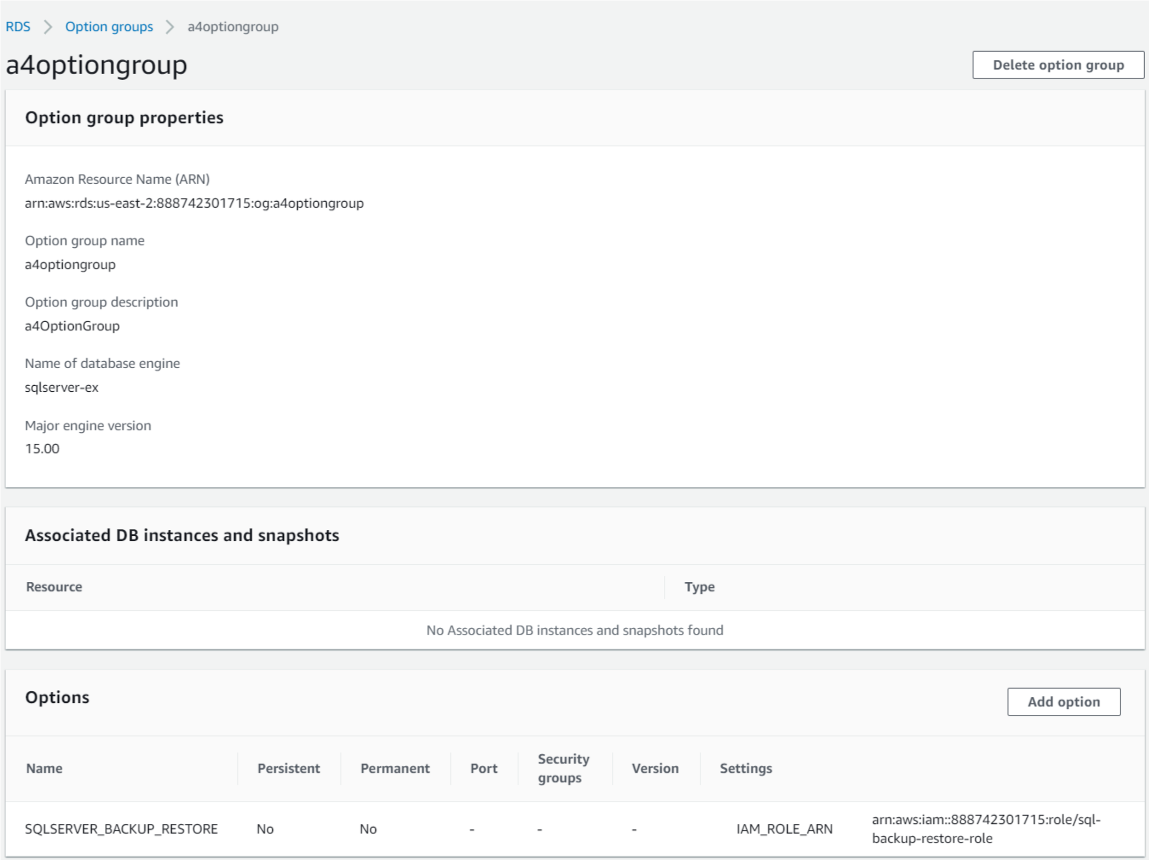
Edit Trust Relationship

You can customize trust relationships by editing the following access control policy document.

Policy Document

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "rds.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

- Restore the Auctions database in RDS
  - Navigate to **RDS Dashboard**, click **Option groups**, **Create group**
  - Option group details:
    - Name: a4OptionGroup
    - Engine: sqlserver-ex
    - Major Engine Version: 15.00
  - In the option group a4OptionGroup:
    - Add option: SQLSERVER_BACKUP_RESTORE
    - IAM role: sql-backup-restore-role
    - Scheduling for adding option: Immediately



RDS > Option groups > a4optiongroup

a4optiongroup

Option group properties

Amazon Resource Name (ARN)
arn:aws:rds:us-east-2:888742301715:og:a4optiongroup

Option group name
a4optiongroup

Option group description
a4OptionGroup

Name of database engine
sqlserver-ex

Major engine version
15.00

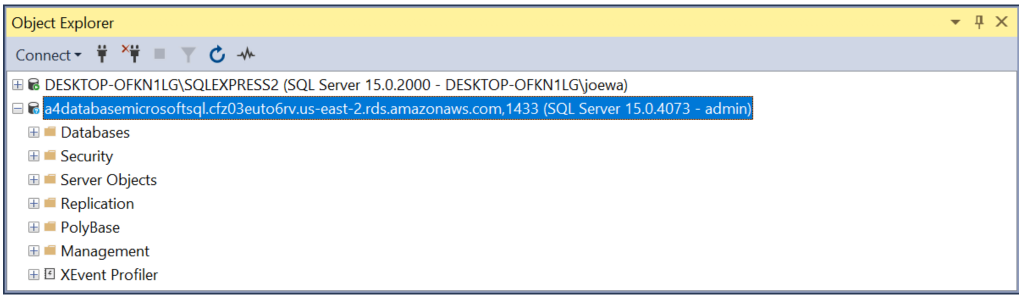
Associated DB instances and snapshots

Resource	Type
No Associated DB instances and snapshots found	

Options

Name	Persistent	Permanent	Port	Security groups	Version	Settings	
SQLSERVER_BACKUP_RESTORE	No	No	-	-	-	IAM_ROLE_ARN	arn:aws:iam::888742301715:role/sql-backup-restore-role

- Connect to the a4DatabaseMicrosoftSQL via Microsoft SQL Server Management Studio
  - Server type: Database Engine
  - Server name: a4databasemicrosoftsql.cfz03euto6rv.us-east-2.rds.amazonaws.com,1433
  - Authentication: SQL Server Authentication
  - Login: admin
  - Password: ********

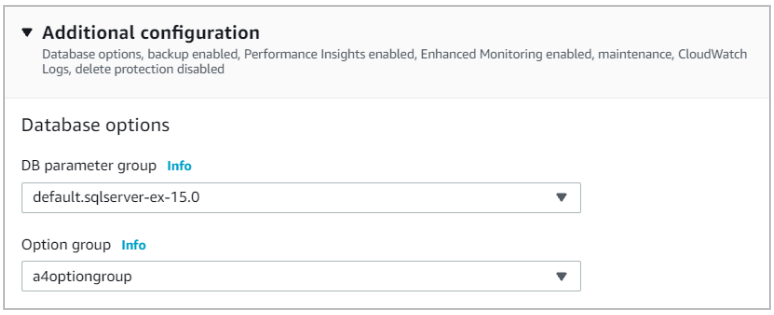


Object Explorer

DESKTOP-OFKN1LG\SQLEXPRESS2 (SQL Server 15.0.2000 - DESKTOP-OFKN1LG\joewa)
a4databasemicrosoftsql.cfz03euto6rv.us-east-2.rds.amazonaws.com,1433 (SQL Server 15.0.4073 - admin)
Databases
Security
Server Objects
Replication
PolyBase
Management
XEvent Profiler

https://github.com/joe-wasuruj/aws-step-by-step-guides

- Update **a4databasemicrosoftsql** to have **Option groups** = **a4optiongroup**



Additional configuration

Database options, backup enabled, Performance Insights enabled, Enhanced Monitoring enabled, maintenance, CloudWatch Logs, delete protection disabled

Database options

DB parameter group Info

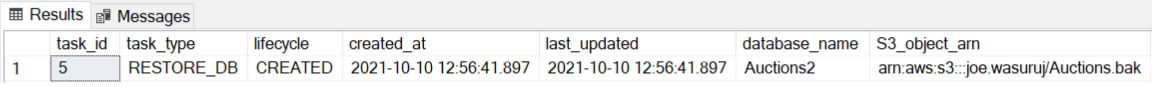
default.sqlserver-ex-15.0

Option group Info

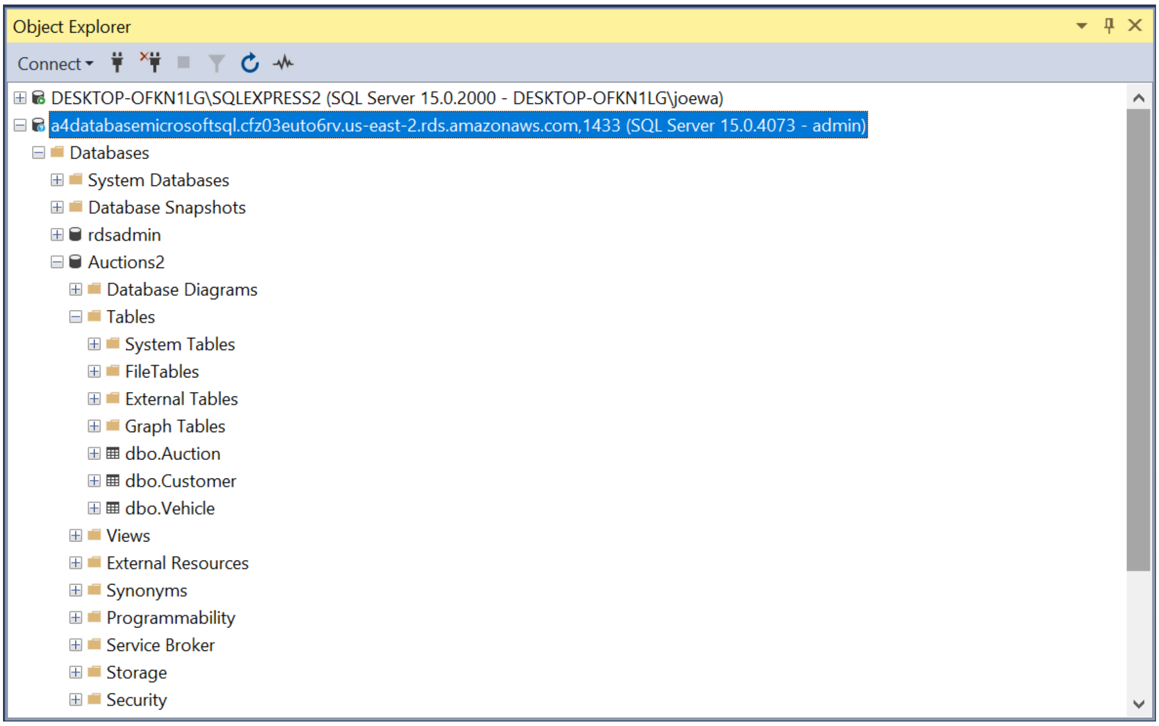
a4optiongroup

- Right click the **a4databasemicrosoftsql** database, click **New query** and execute the code below:

```
exec msdb.dbo.rds_restore_database
       @restore_db_name='Auctions2',
       @s3_arn_to_restore_from='arn:aws:s3:::joe.wasuruj/Auctions.bak',
       @with_norecovery=0;
```



	task_id	task_type	lifecycle	created_at	last_updated	database_name	S3_object_arn
1	5	RESTORE_DB	CREATED	2021-10-10 12:56:41.897	2021-10-10 12:56:41.897	Auctions2	arn:aws:s3:::joe.wasuruj/Auctions.bak

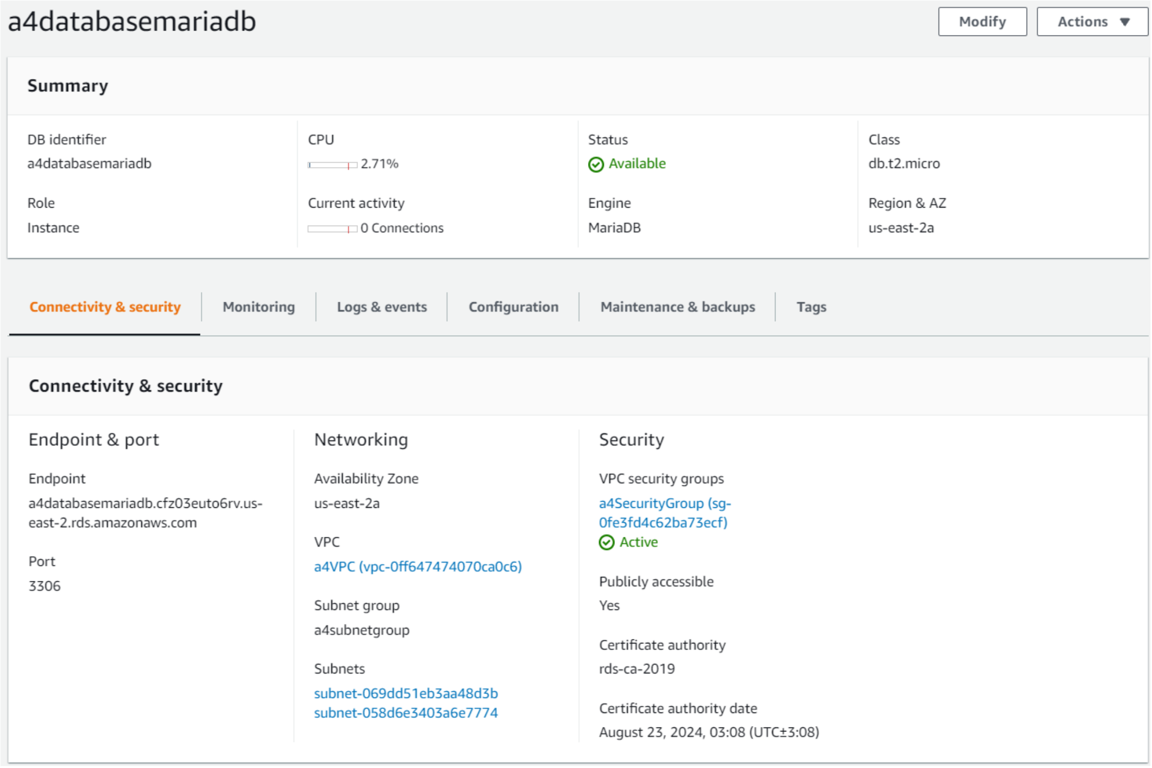


- DESKTOP-OFKN1LG\SQLEXPRESS2 (SQL Server 15.0.2000 - DESKTOP-OFKN1LG\joewa)
- a4databasemicrosoftsql.cfz03euto6rv.us-east-2.rds.amazonaws.com,1433 (SQL Server 15.0.4073 - admin)
  - Databases
    - System Databases
    - Database Snapshots
    - rdsadmin
    - Auctions2
      - Database Diagrams
      - Tables
        - System Tables
        - FileTables
        - External Tables
        - Graph Tables
        - dbo.Auction
        - dbo.Customer
        - dbo.Vehicle
      - Views
      - External Resources
      - Synonyms
      - Programmability
      - Service Broker
      - Storage
      - Security

3. Inter instance migration

3.1 Create an RDS instance (MariaDB) on AWS

- Navigate to RDS Dashboard, click **Databases**, **Create database**
  - Database creation method: Standard create
  - Engine options: MariaDB (Version MariaDB 10.5.12)
  - Templates: Free tier
  - Settings:
    - DB instance identifier: a4DatabaseMariaDB
    - Master username: admin
    - Master password: ********
    - DB instance class: db.t2.micro
    - Storage: General Purpose SSD (gp2) 20 GiB
    - Storage autoscaling: Enabled
    - VPC: a4VPC
    - Subnet group: a4subnetgroup
    - Public access: Yes
    - VPC security group: Choose existing: default
    - Availability Zone: us-east-2a
    - Database port: 3306



a4databasemariadb

Summary

DB identifier
a4databasemariadb

Role
Instance

CPU
2.71%

Current activity
0 Connections

Status
✓ Available

Engine
MariaDB

Class
db.t2.micro

Region & AZ
us-east-2a

Connectivity & security

Endpoint & port

Endpoint
a4databasemariadb.cfz03euto6rv.us-east-2.rds.amazonaws.com

Port
3306

Networking

Availability Zone
us-east-2a

VPC
a4VPC (vpc-0ff647474070ca0c6)

Subnet group
a4subnetgroup

Subnets
subnet-069dd51eb3aa48d3b
subnet-058d6e3403a6e7774

Security

VPC security groups
a4SecurityGroup (sg-0fe3fd4c62ba73ecf)
✓ Active

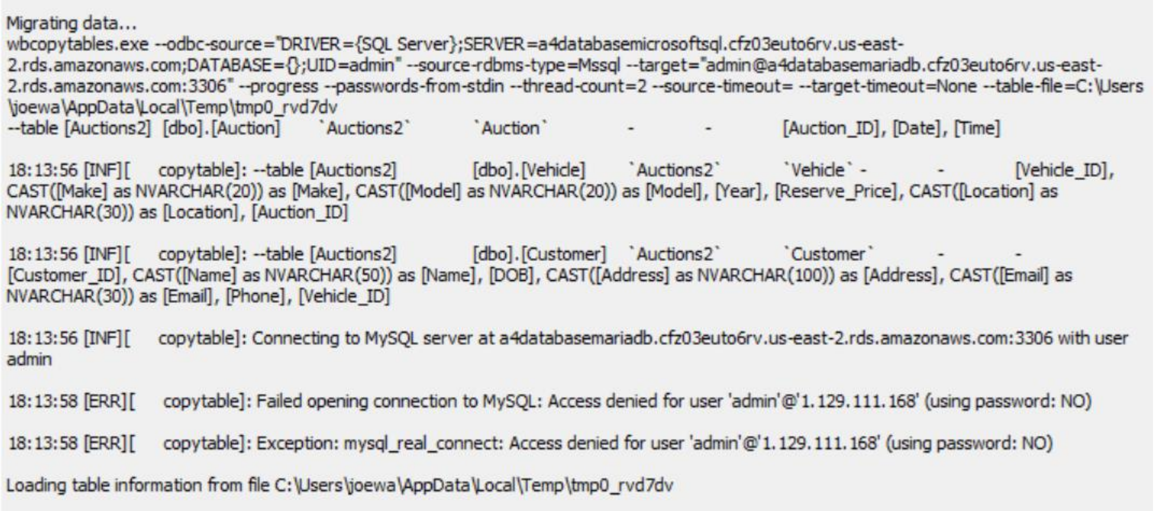
Publicly accessible
Yes

Certificate authority
rds-ca-2019

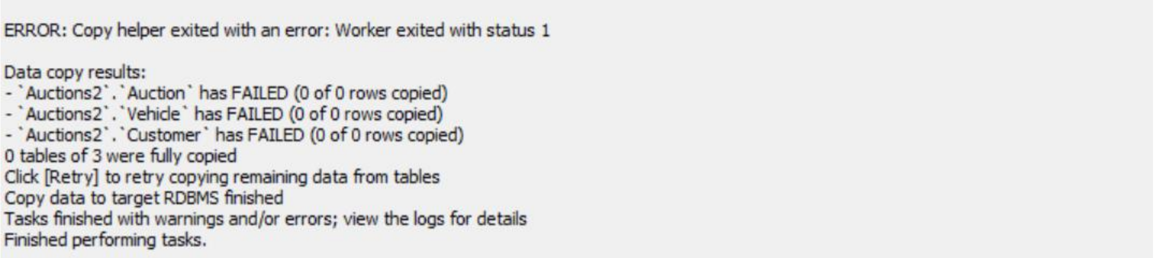
Certificate authority date
August 23, 2024, 03:08 (UTC±3:08)

3.2 Transfer the data from the Microsoft SQL Server instance to the MariaDB instance

- Install and launch **MySQL Workbench 8.0 for Windows**
- From the toolbar, click **Database**, **Migration Wizard**, **Start Migration**
- Source selection:
  - Database system: Microsoft SQL Server
  - Connection method: ODBC (native)
  - Parameters:
    - Driver: SQL Server
    - Server: a4databasemicrosoftsql.cfz03euto6rv.us-east-2.rds.amazonaws.com
    - Username: admin
- Target selection:
  - Connection method: Standard (TCP/IP)
  - Parameters:
    - Hostname: a4databasemariadb.cfz03euto6rv.us-east-2.rds.amazonaws.com
    - Port: 3306
    - Username: admin
- Schemas selection: Catalog/Schema: Auctions2 – dbo
- Source objects: Auctions2.Auction, Auctions2.Customer, Auctions2.Vehicle
- Target creation options: Create schema in target RDBMS
- Data transfer setup: Online copy of table data to target RDBMS
- The transfer results show as FAILED because there were no rows in the tables, but **the tables were transferred successfully**.



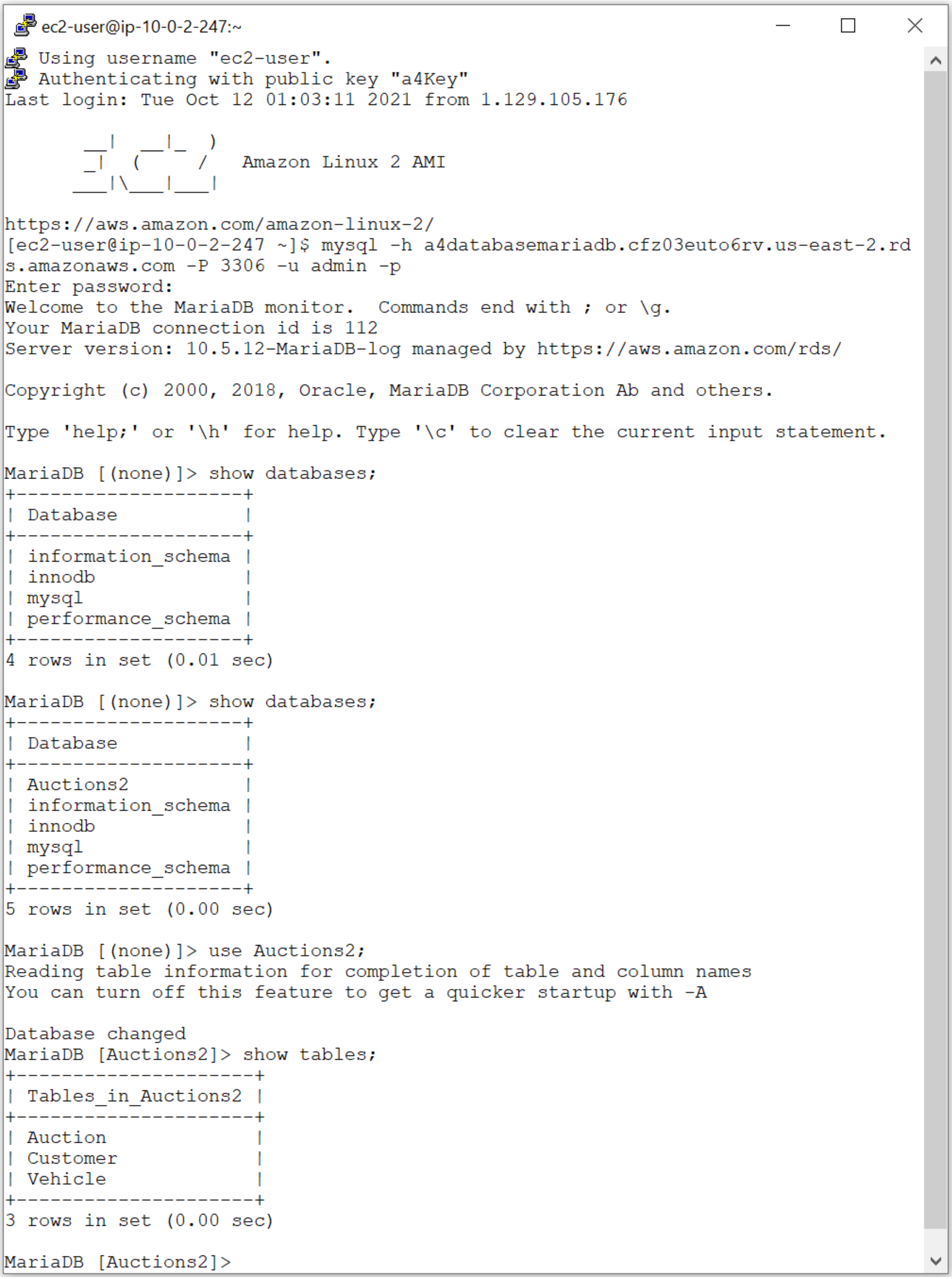
```
Migrating data...
wbcopytables.exe --odbc-source="DRIVER={SQL Server};SERVER=a4databasemicrosoftsql.cfz03euto6rv.us-east-
2.rds.amazonaws.com;DATABASE={};UID=admin" --source-rdbms-type=Mssql --target="admin@a4databasemariadb.cfz03euto6rv.us-east-
2.rds.amazonaws.com:3306" --progress --passwords-from-stdin --thread-count=2 --source-timeout= --target-timeout=None --table-file=C:\Users
\joewa\AppData\Local\Temp\tmp0_rvd7dv
--table [Auctions2]  [dbo].[Auction]     `Auctions2`       `Auction`        -        -        [Auction_ID], [Date], [Time]

18:13:56 [INF][    copytable]: --table [Auctions2]          [dbo].[Vehicle]     `Auctions2`       `Vehicle` -        -        [Vehicle_ID],
CAST([Make] as NVARCHAR(20)) as [Make], CAST([Model] as NVARCHAR(20)) as [Model], [Year], [Reserve_Price], CAST([Location] as
NVARCHAR(30)) as [Location], [Auction_ID]

18:13:56 [INF][    copytable]: --table [Auctions2]          [dbo].[Customer]    `Auctions2`       `Customer`        -        -
[Customer_ID], CAST([Name] as NVARCHAR(50)) as [Name], [DOB], CAST([Address] as NVARCHAR(100)) as [Address], CAST([Email] as
NVARCHAR(30)) as [Email], [Phone], [Vehicle_ID]

18:13:56 [INF][    copytable]: Connecting to MySQL server at a4databasemariadb.cfz03euto6rv.us-east-2.rds.amazonaws.com:3306 with user
admin

18:13:58 [ERR][    copytable]: Failed opening connection to MySQL: Access denied for user 'admin'@'1.129.111.168' (using password: NO)

18:13:58 [ERR][    copytable]: Exception: mysql_real_connect: Access denied for user 'admin'@'1.129.111.168' (using password: NO)

Loading table information from file C:\Users\joewa\AppData\Local\Temp\tmp0_rvd7dv
```



```
ERROR: Copy helper exited with an error: Worker exited with status 1

Data copy results:
- `Auctions2`.`Auction` has FAILED (0 of 0 rows copied)
- `Auctions2`.`Vehicle` has FAILED (0 of 0 rows copied)
- `Auctions2`.`Customer` has FAILED (0 of 0 rows copied)
0 tables of 3 were fully copied
Click [Retry] to retry copying remaining data from tables
Copy data to target RDBMS finished
Tasks finished with warnings and/or errors; view the logs for details
Finished performing tasks.
```

- I created an EC2 instance to access the MariaDB instance to check whether the Auctions2 database has been migrated.

```
mysql -h a4databasemariadb.cfz03euto6rv.us-east-2.rds.amazonaws.com -P 3306 -u admin -p
```



```
Using username "ec2-user".
Authenticating with public key "a4Key"
Last login: Tue Oct 12 01:03:11 2021 from 1.129.105.176

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-10-0-2-247 ~]$ mysql -h a4databasemariadb.cfz03euto6rv.us-east-2.rd
s.amazonaws.com -P 3306 -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 112
Server version: 10.5.12-MariaDB-log managed by https://aws.amazon.com/rds/

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

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

MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| innodb             |
| mysql              |
| performance_schema |
+--------------------+
4 rows in set (0.01 sec)

MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| Auctions2          |
| information_schema |
| innodb             |
| mysql              |
| performance_schema |
+--------------------+
5 rows in set (0.00 sec)

MariaDB [(none)]> use Auctions2;
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
MariaDB [Auctions2]> show tables;
+---------------------+
| Tables_in_Auctions2 |
+---------------------+
| Auction             |
| Customer            |
| Vehicle             |
+---------------------+
3 rows in set (0.00 sec)

MariaDB [Auctions2]>
```