# RDS

*Kiran*

## Table of Contents



## 1. Relational Database Service (RDS)

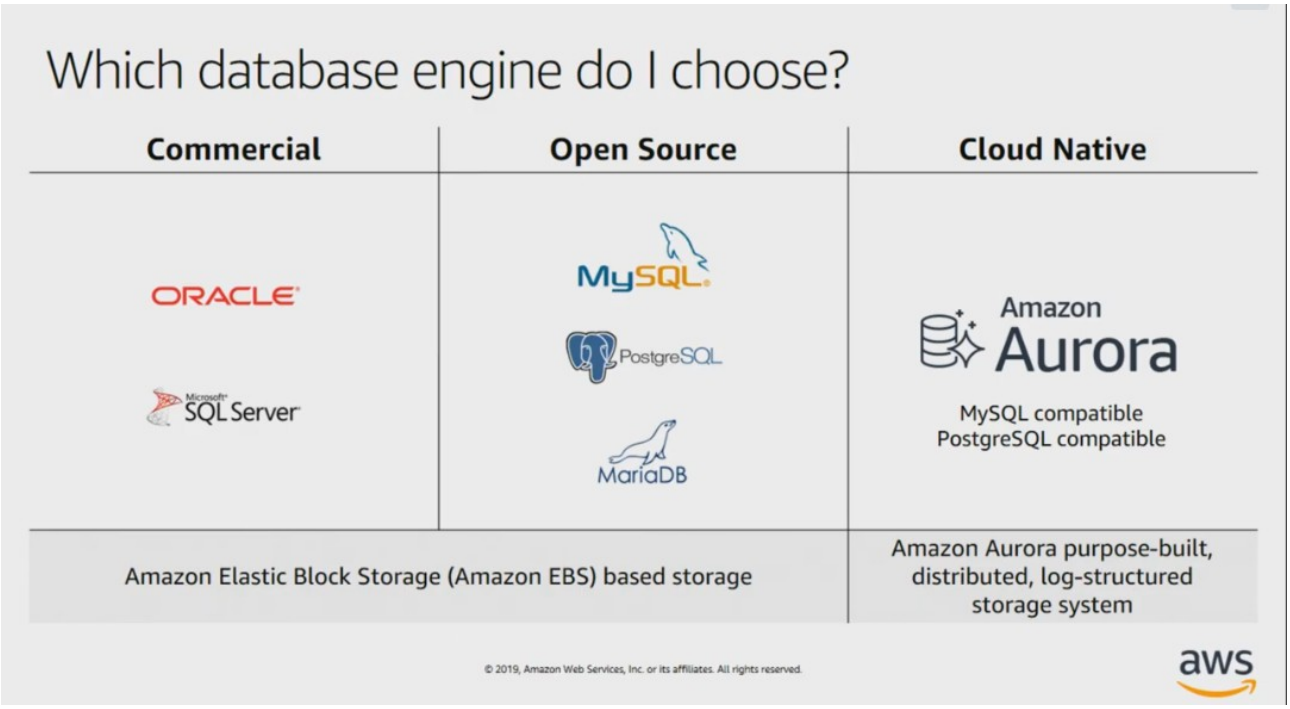
Amazon RDS is managed database sevice offered by AWS - where users can create a Database Instance and start using it without having to set up a cluster of servers and install the required SQL Database software.

RDS makes it easy to set up, operate, and scale a relational database in the cloud. It provides cost-efficient and resizable capacity while automating time-consuming administration tasks such as hardware provisioning, database setup, patching and backups. It frees you to focus on your applications so you can give them the fast performance, high availability, security and compatibility they need.

Amazon RDS is available on several database instance types - optimized for memory, performance or I/O.

AWS provides you with six familiar database engines:

- Amazon Aurora.
- PostgreSQL.
- MySQL
- MariaDB
- Oracle Database
- MS SQL Server.



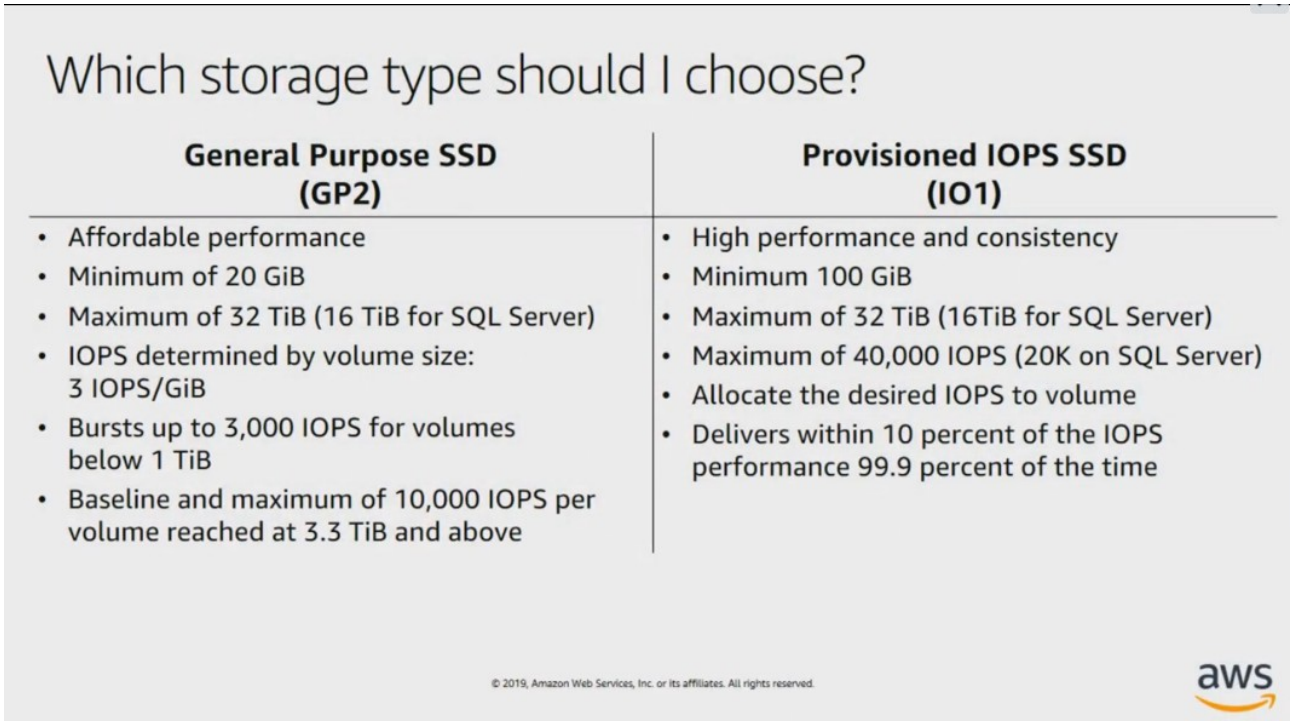

### 1.1.1. Amazon Aurora

Amazon Aurora is a MySQL and PostgreSQL-compatible relational database built for the cloud, that combines the performance and availability of traditional enterprise databases with the simplicity and cost-effectiveness of open source databases.

Amazon Aurora is up to five times faster than standard MySQL databases and three times faster than standard PostgreSQL databases. It provides the security, availability, and reliability of commercial databases at 1/10th the cost. Amazon Aurora is fully managed by Amazon Relational Database Service (RDS), which automates time-consuming administration tasks like hardware provisioning, database setup, patching, and backups.

You can use the AWS Database Migration Service to easily migrate or replicate your existing databases to Amazon RDS.

## *1.2. Instance Types*



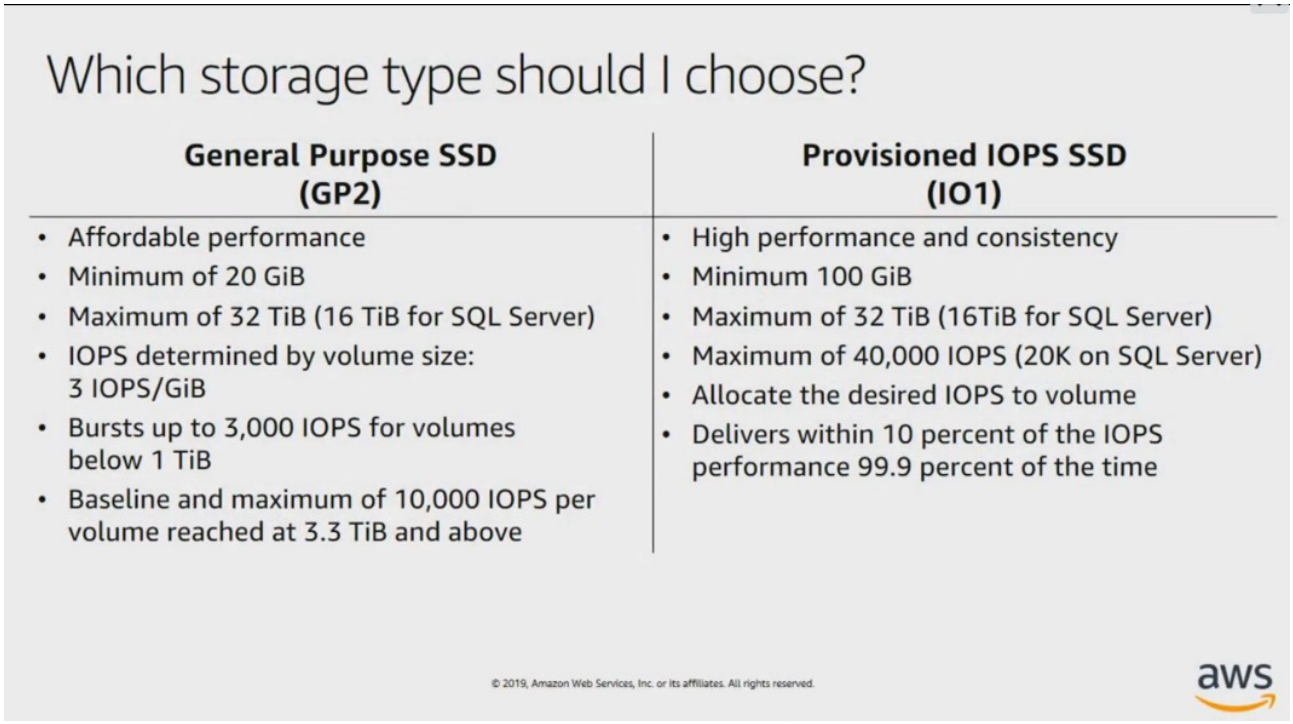

## *1.3. Storage types*

### 1.3.1. General Purpose SSD

- Affordable Performance
- Minimum of 20GB and Maximum of 32 TiB (16 TiB for SQL Server)
- IOPS determined by volume size - 3 IOPS/GiB
- Bursts up to 3,000 IOPS coumes belwow 1 TiB
- Baseline and maximum of 10,000 IOPS per volume reached at 3.3 TiB and over

### 1.3.2. Provisioned IOPS

- High Performance and Consistency.
- Minimum of 100 GiB and Maximum of 32 TiB (16 TiB for SQL Server).
- Maximum of 40,000 IOPS (20K on SQL Server).
- Allocate the desired IOPS to volumes.




## *1.4. High Availability*

### 1.4.1. Multi-AZ Configuration

There are 2 Database instances. Primary and secondary/Standby in 2 different AZs.

Each DB instance manages a set of Amazon EBS volumes with a full copy of the data. Data is synchronously (real-time) replicated from primary to standby instance.

In case of failure of the primary DB, RDS automatically initiates the failover to the Standby instance.

The standby instance now becomes the primary instance. A new DB instance is created in the other AZ and data starts replicating to that instance.

This set up only detects DB infrastructure issues but not any DB engine problems.

## *1.5. Read Scalability*

RDS provides read replication to decrease load on the source Database with additional read capacity.

The replication is asynchronous.

You can create up to 5 replicas per source database.

## *1.6. Backups*

### 1.6.1. Storage Level snapshots

Manual snapshots taken at EBS levels.

### 1.6.2. Automated Backups

Provides a point-in-time restore options. Transaction logs are stored every five minutes. Can restore up to 35 days agouu – kkkk

NN00tt@@55ttrruunngg00nnee

ccdd DDeess

ll

vviimm aazznnootteess..mmdd

https://www.youtube.com/watch?v=TdCc4reQYbU .

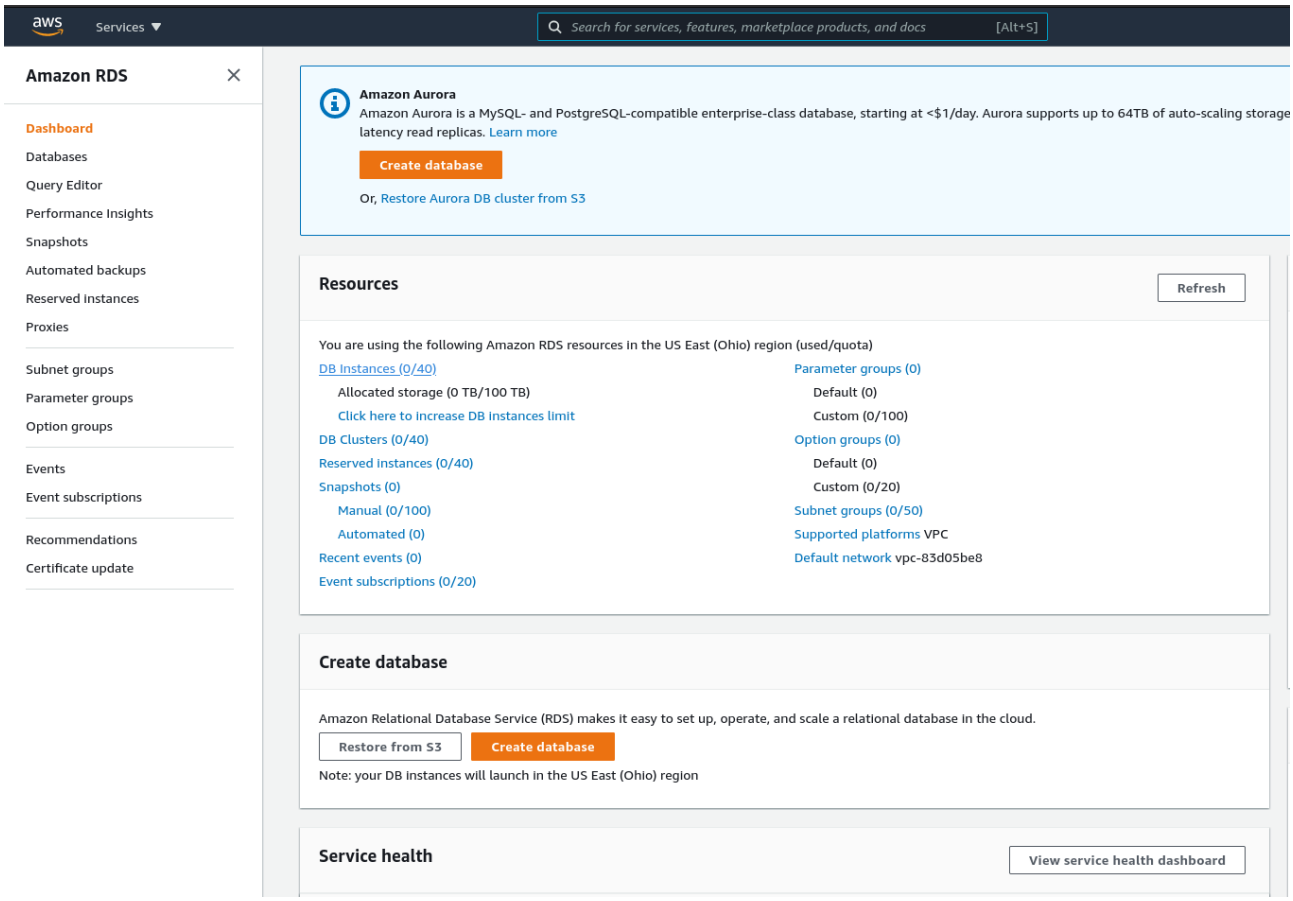
## *1.7. Disaster Recovery*

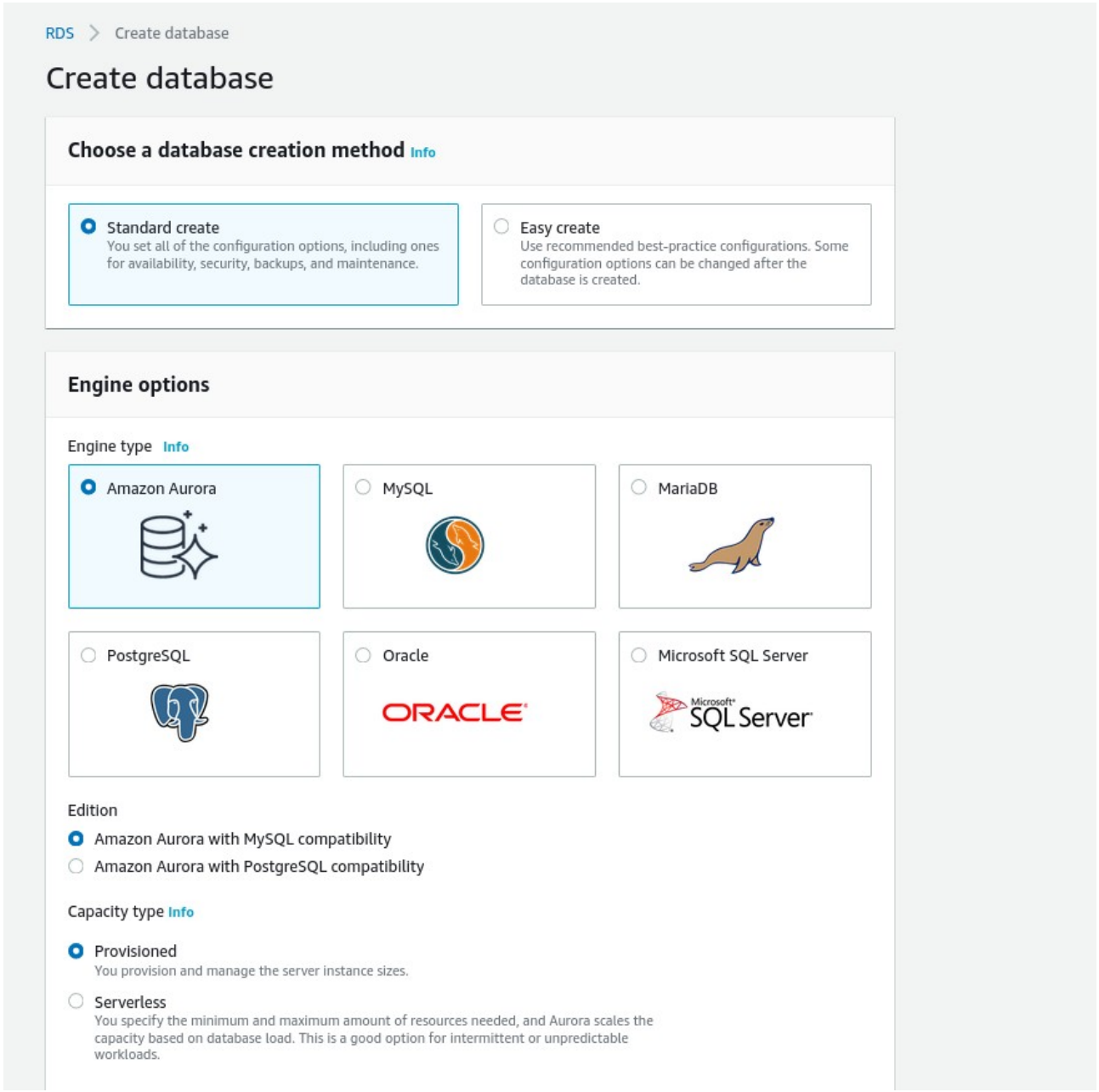
Use Automated Backups and Manual snapshots

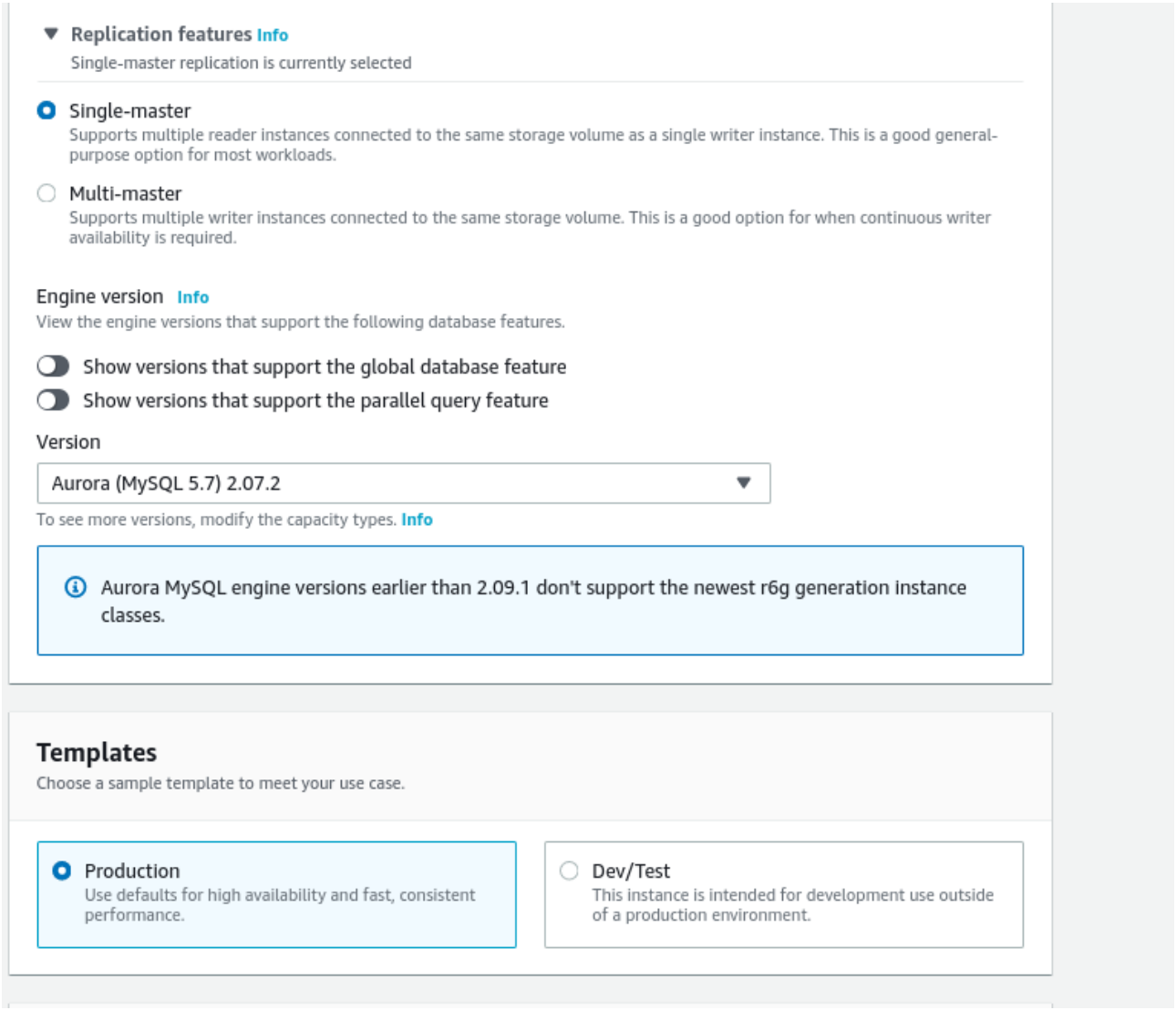
Use cross region read replica as a standby database for recovery in case of disaster in the region where your source db is located.

Read replicas can be configured for Multi-AZ to reduce recovery time.

## *1.8. Create an RDS Instance*




RDS > Create database
Create database
Choose a database creation method Info
Standard create
You set all of the configuration options, including ones for availability, security, backups, and maintenance.
Easy create
Use recommended best-practice configurations. Some configuration options can be changed after the database is created.
Engine options
Engine type Info
Amazon Aurora
MySQL
MariaDB
PostgreSQL
Oracle
ORACLE
Microsoft SQL Server
Microsoft SQL Server
Edition
Amazon Aurora with MySQL compatibility
Amazon Aurora with PostgreSQL compatibility
Capacity type Info
Provisioned
You provision and manage the server instance sizes.
Serverless
You specify the minimum and maximum amount of resources needed, and Aurora scales the capacity based on database load. This is a good option for intermittent or unpredictable workloads.


Replication features Info
Single-master replication is currently selected
Single-master
Supports multiple reader instances connected to the same storage volume as a single writer instance. This is a good general-purpose option for most workloads.
Multi-master
Supports multiple writer instances connected to the same storage volume. This is a good option for when continuous writer availability is required.
Engine version Info
View the engine versions that support the following database features.
Show versions that support the global database feature
Show versions that support the parallel query feature
Version
Aurora (MySQL 5.7) 2.07.2
To see more versions, modify the capacity types. Info
Aurora MySQL engine versions earlier than 2.09.1 don't support the newest r6g generation instance classes.
Templates
Choose a sample template to meet your use case.
Production
Use defaults for high availability and fast, consistent performance.
Dev/Test
This instance is intended for development use outside of a production environment.

## Settings

DB cluster identifier Info

Type a name for your DB cluster. The name must be unique across all DB clusters owned by your AWS account in the current AWS Region.

database-1

The DB cluster identifier is case-insensitive, but is stored as all lowercase (as in "mydbcluster"). Constraints: 1 to 60 alphanumeric characters or hyphens. 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

Master password Info

Constraints: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), '(single quote), "(double quote) and @ (at sign).

Confirm password Info

## DB instance class

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.

◉ Memory optimized classes (includes r classes)

○ Burstable classes (includes t classes)

db.r5.large
2 vCPUs 16 GiB RAM Network: 4,750 Mbps

ⓘ New instance classes are available for specific engine versions. Info

Include previous generation classes

## Availability & durability

Multi-AZ deployment Info

- ○ Don't create an Aurora Replica
- ◉ Create an Aurora Replica or Reader node in a different AZ (recommended for scaled availability)
  Creates an Aurora Replica for fast failover and high availability.

## Connectivity

Virtual private cloud (VPC) Info
VPC that defines the virtual networking environment for this DB cluster.

Default VPC (vpc-83d05be8)

Only VPCs with a corresponding DB subnet group are listed.

ⓘ After a database is created, you can't change the VPC selection.

Subnet group Info
DB subnet group that defines which subnets and IP ranges the DB instance can use in the VPC you selected.

default

Public access Info

- ○ Yes
  Amazon EC2 instances and devices outside the VPC can connect to your database. Choose one or more VPC security groups that specify which EC2 instances and devices inside the VPC can connect to the database.
- ◉ No
  RDS will not assign a public IP address to the database. Only Amazon EC2 instances and devices inside the VPC can connect to your database.

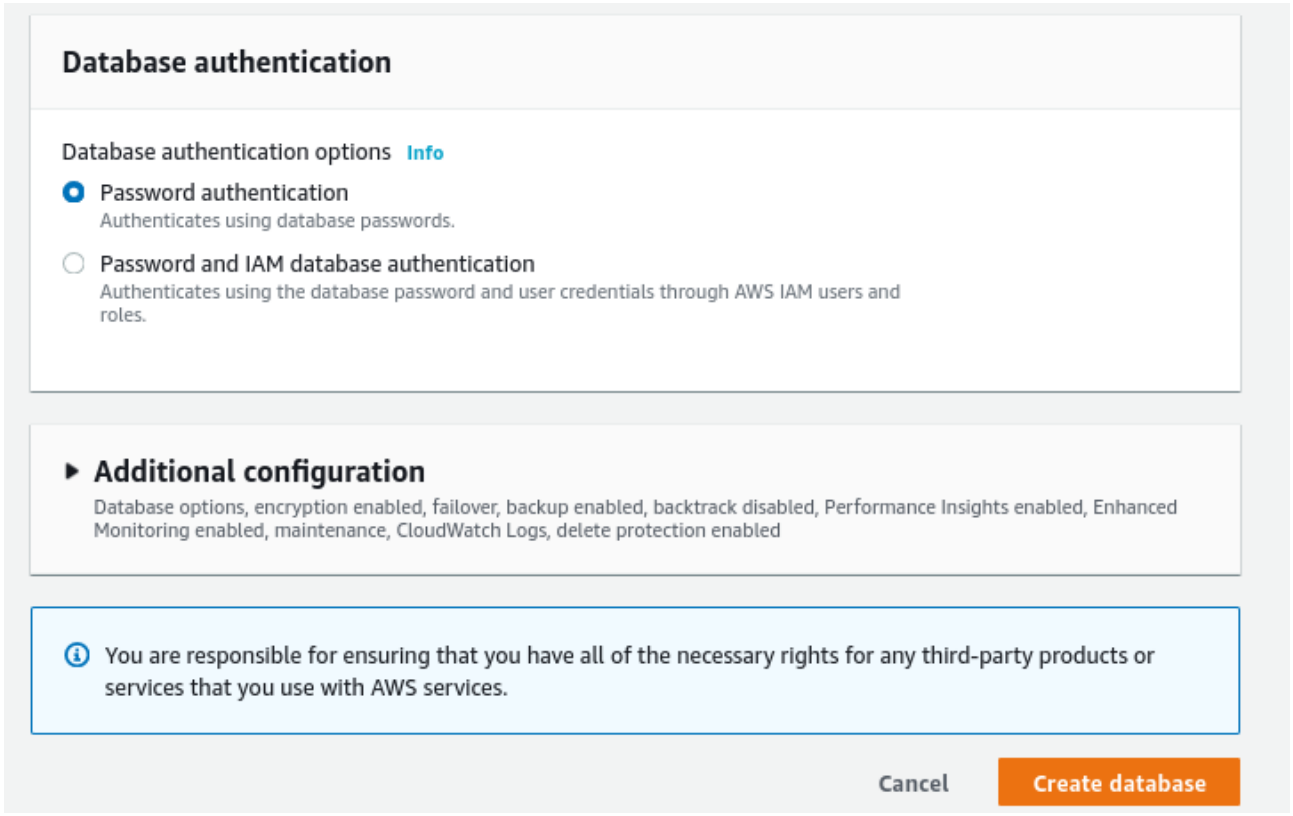
VPC security group
Choose a VPC security group to allow access to your database. Ensure that the security group rules allow the appropriate incoming traffic.

- ◉ Choose existing
  Choose existing VPC security groups
- ○ Create new
  Create new VPC security group

Existing VPC security groups

*Choose VPC security groups*

default ×


Database authentication
Database authentication options Info
Password authentication
Authenticates using database passwords.
Password and IAM database authentication
Authenticates using the database password and user credentials through AWS IAM users and roles.
Additional configuration
Database options, encryption enabled, failover, backup enabled, backtrack disabled, Performance Insights enabled, Enhanced Monitoring enabled, maintenance, CloudWatch Logs, delete protection enabled
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