**RDS -- Relational Database service**

**Database**:System that store and mange data

It will help us to setup and maintain databases in aws cloud

divided into two types:

**Relational(SQL):** structured Query language(SQL)

it will store the data in tables.these tables will have shares information between them

Example:MYSQL,oracle,MariaDB,Postgres amd ms-SQL

**non-Relational(NOSQL):** not only SQL(NOSQL)

it will not use tables,fields, and colums structured data concept from relational data base.

Example:MongoDB,Bigtable,redis,RavenDb,Cassandra and CouchDB

**Databases on EC2 instance :**

webserver

|

|

Application

|

|

Database

ec2

or

webserver

|

|

application

Ec2

database

EC2

This architecture is belongs to 3-tier architecture

use two availability zones because one is down another will be work

**Why should we run DBs on EC2** :

* Access to the DB instance Os because in the RDS you don’t get to see what is the OS they are using or you don’t get the access to the OS level in that situation we will use EC2 to run the DB os level.
* Advanced DB option tuning (DBROOT)
* vendor demands.(ask them why we need to run database on EC2 until proper requirement)
* DB or DB version that AWS doesn’t provide
* you might need a specific version of an OS and DB that AWS doesn’t provide.(updating themself but in case you database not there then you hoste your DB or run the Os)

**Why should’t we run DBs on EC2** :

* Admin overhead.(you have to to manage your patching,you have manage your OS upgrade all those things you will have to manage and there is a lot for the admin team so it will be an admin overhead)
* Bcckup and DB
* EC2 is running in a single AZ.(ec2 is running in single availability zone if that means availability zone goes down the entire application web server and your datatabase everything will be gone so all the data that you store everything will be gone.)
* will miss out on feature from aws DB products.(if you don’t use it in a right way)
* skills and setup time to monitor.(24/7)
* Performance will be slower than AWS options.

**To installing Mariadb:**

sudo su -

yum -y install mariadb-server wget

systemctl enable mariadb

systemctl start mariadb

yum -y update

**Set Environmental Variables**

DBName=ec2db

DBPassword=admin123456

DBRootPassword=admin123456

DBUser=ec2dbuser

**Database Setup on EC2 Instance:**

echo "CREATE DATABASE ${DBName};" >> /tmp/db.setup

echo "CREATE USER '${DBUser}' IDENTIFIED BY '${DBPassword}';" >> /tmp/db.setup

echo "GRANT ALL PRIVILEGES ON \*.\* TO '${DBUser}'@'%';" >> /tmp/db.setup

echo "FLUSH PRIVILEGES;" >> /tmp/db.setup

mysqladmin -u root password "${DBRootPassword}"

mysql -u root --password="${DBRootPassword}" < /tmp/db.setup

rm /tmp/db.setup

**Relational Database Service (RDS)**:

* Database-as-a-service(DBaas)-not realy true.it is more of database server-as-a-service
* Managed Database instance for one or more database(aws is providing you database as one of the service to you so it is completely managed you don’t need to take care of anything everything comes manged for you.so you just have to create database on the RDS instances and start integrating with your application or your website) you can host one or more database on the RDS instance
* RDS suports: MySQL,MariaDB,PostgreSQL,Oracle,MicrosoftSQl.
* Amazon Aurora-this is so different from normal RDS,it is a separate product.

**RDS Database instance**:

* Database connects with a CNAME.The RDS will uses standard database engines.
* .the data base can be optimized for : db.m5 general,db.r5 memory,db.t3 burst.
* when you provision an instances,you provision storage that is dedicated to that instance.this is EBS storage located in the same availability zone.
* RDS is vulnerable to failure in that AZ.
* The storage can be allocated with SSD or magnetic
* billing will be hourly basis .your billing based on the storage.

**Migrating DB from EC2 to RDS:**

step 1: get the dump of your existing DB on EC2

step 2: connect to you RDS DB instance.

step3:Migrate the DB Dump that you have taken in step 1 to RDS.

step 4: verify if the data is available.

**RDS multi AZ (High-Availability):**

* RDS access only via database CNAME.The CNAME will point at the primary instance.You cannot access the standby replica for any reason via RDS.
* The standby replica cannot be use for extra capacity.
* synchronous Replication.(changes of right operation that will occurring in the primary database at the same time that right operation will happening in your standby replication also ) read and write operations are occuring in same time in both primary and secondary replica.
* stand by replica commits all your rights.

**Points to Remember:**

1. Multi AZ feature is not free and we need to pay twice the price.
2. Standy replcia cannot be accessed directly unless a failure

occurs.

1. Failover is highly available, not fault tolerant.
2. Backups taken from standby (Removes performance impacts)
3. Multi AZ will be created in same region.
4. AZ outage,primary failure,manual failover,instance type change, and software patching.

**RDS Backup and Restores:**

**RPO:**Recovery point Objective

-time between the last backup and when the failure occurred.

-Amount of maximum data loss.

-influences technical solution and cost.

-Business usually provides an RPO value.

**RTO**:Recovery Time Objective.

-time between the Disaster event and full recovery.

-influenced by process,staff,tech and documentation.

**RDS Backups:**

1. First snap is full size of consumed data.

After first snapshot it will only capture the latest data written to db.

1. Manual snapshots will remain in your aws account even after the life of snapshot,these needs to be deleted manually.
2. **Automatic Snapshots.**

* Evry 5 minutes translation logs are saved to s3.a database can be restore to a 5 min snapshots in time.
* Automatic cleanups can be anywhere from 0 to 35 days.
* When you delete the db,they can be retained but they will

expire based on thier retention period.

**Points to Remember:**

* When performing a restore, RDS creates a new RDS with a new endpoint address.
* When restoring a manual snapshot, you are setting it to a single point in time. This influences the RPO value.
* Automated backups are different, any 5-minute point in time.
* Backups are restored and transaction logs are replayed to bring DB to desired point in time.
* Restores aren't fast, think about RTO.

**RDS Read-Replicas:**

* Kept in sync using asynchronous replication
* It is written fully to the primary instance. Once its stored-on disk, it is then pushed to the replica. This means there could be a small lag.These can be created in the same region or a different region. This is a cross region replication.

**Performanse Read Replica:**

* 5 direct read-replicas per DB instance
* Each of these provides an additional instance of read performance
* This allows you to scale out read operations for an instance
* Read-replicas can chain, but lag will become a problem
* Can provide global performance improvements