# AWS – Certified Security Architect

## Databases

|  |  |
| --- | --- |
| Relational | Non Relational |
| Database | Database |
| Tables | Collections |
| Row | Document |
| Field | Key Value Pairs |

RDS - Relational Databases (OLTP)

* MSSQL
* Oracle
* MySQL
* PostgreSQL
* Amazon Aurora
* MariaDB

Non-Relational – No SQL

* DynamoDB

OLAP

* RedShift

Data Warehousing

This is used for Business Intelligence. Used to pull in large and complex volumes of data and execute queries on them.

OLTP vs OLAP: Online Transaction Processing vs Online Analytics processing

OLTP: Interact with one record at a time

OLAP: Pulls in a large number of records to query

OLAP systems are architected differently

## Elasticache

Allows you to deploy, operate and scale an in memory cache in the cloud. Allows for fast retrieval of frequently accessed information and reduces load on database servers

AWS Hosts:

* Memcached
* Redis

### Database Migration Service

Migrate production database to AWS. AWS manages all complexities of migration process. Source database remains fully operational. Both homogenous (Oracle to Oracle) as well as heterogeneous migrations are supported (Oracle to Aurora or Microsoft SQL). Can also be used for continuous data replication with high availability

**AWS Schema migration tool makes heterogeneous database** - migrations - easy by automatically converting the source database schema and a majority of the custom code, including views, stored procedures, and functions, to a format compatible with the target database. Any code that cannot be automatically converted is clearly marked so that it can be manually converted.

## LAB 1: Set up a RDS Database

Console 🡪 Databases 🡪 Relational Database Service 🡪 Instances 🡪 MySQL

Create a EC2 Instance with the following in the advanced section to bootstrap

#!/bin/bash

yum install httpd php php-mysql -y

yum update -y

chkconfig httpd on

service httpd start

echo "<?php phpinfo();?>" > /var/www/html/index.php

cd /var/www/html

wget <https://s3.eu-west-2.amazonaws.com/acloudguru-example/connect.php>

VPC 🡪 Security Groups 🡪 MYSQL Security group

Add 3306(mysql) from security group of the EC2 instance

connect.php:

<?php

$username = "acloudguru";

$password = "acloudguru";

$hostname = "yourhostnameaddress";

$dbname = "acloudguru";

//connection to the database

$dbhandle = mysql\_connect($hostname, $username, $password) or die("Unable to connect to MySQL");

echo "Connected to MySQL using username - $username, password - $password, host - $hostname<br>";

$selected = mysql\_select\_db("$dbname",$dbhandle) or die("Unable to connect to MySQL DB - check the database name and try again.");

?>

Get the **Endpoint** from the DB instance and insert into $hostname

http to ec2instanceIP/connect.php

BACKUPS

* Automated Backups:
  + Allow you to restore to any point in the “**Retention Period**”.
  + The retention period is from 1-35 days
  + Daily Backups + Transaction Logs
  + Enabled automatically
  + Stored in S3
  + Free storage equivalent to size of database
  + Fixed Backup time
  + Storage IO may be suspended during backup causing latency
* Database snapshots
  + Fixed Point in time image
  + Done manually
  + Persist after database deletion
* Restore
  + Restored instance becomes the new instance with DNS changed from original to restored. You have a new endpoint
* Encryption
  + Available for RDS
  + Done using KMS
  + Backups, snapshots and read replicas are encrypted as well
  + Cannot encrypt an existing Database
  + Snapshot, Copy, encrypt the copy
* Snapshots
  + Done manually. They are stored even after you delete the instance.
  + You can copy snapshots across regions.
  + You can publish the snapshot to make it publicly available.
  + Restoring Backups/ Snapshots – The restored version will be a new RDS instance with new end point.
  + You can check the instance size to restore.
  + You cannot restore to existing instance
* Multi-AZ
  + Synchronously replicated to another AZ
  + For DR only – if AZ1 dies or planned maintenance or DB instance failure, the RDS instance will automatically failover to the other AZ
  + DNS address will point to the failover AZ
  + Not used for performance scaling
* Read Replica
  + All transactions are pushed to replicas of the main DB
  + Can have up to 5 Read Replicas
  + Read operations are distributed
  + Can have a Read Replica of a read replica (may have some latency)
  + Read replicas could be in same or different AZ or Region (MySQL and PostgreSQL)
  + Asynchronous replication
  + Not available for Oracle or SQL Server
  + Used for scaling out
  + Must have automatic backups turned on
  + Each read replica has its own DNS endpoint
  + Read Replicas can be multi AZ
  + Can create read replicas of multi AZ DBs
  + Can be promoted to become their own DBs (This breaks the replication)
  + Not used for DR only for scaling

## DynamoDB

* Fast and flexible NoSQL database
* Consistent, single digit millisecond latency
* Consistency
* Stored on SSD
* Fully managed DB – supports both document based & Key-value data models.
* Great fit for mobile, IoT, web, gaming etc. applications.
* Spread across 3 geographically distinct data centers
* Eventual Consistency Reads (Default)
  + Reached within 1 sec (best read performance)
* Strongly Consistent Reads
  + Reflects all writes that received a successful response prior to read
* Pricing
  + Provisioned Throughput capacity
    - Read + write
  + Storage costs
* Expensive for writes

### RDS v/s DynamoDB

* Use DynamoDB for Push button scaling. With RDS – to scale horizontally a new instance has to be created.
* DynamoDB is cheap for writes and expensive for reads.
* Observe workload characteristics and decide
* Use RDS if data requires joins and relationships.
* In RDBMS database structure cannot be dynamically altered. With DynamoDB you can.

## Redshift (OLAP)

Fast, managed petabyte scale data warehousing service

10X faster than OLTP

Configuration

* Single node (160GB)
* Multi-Node
  + Leader node (manages connections and receives queries)
  + Compute Node (store data and perform queries) Up to 12

**Columnar Data Storage**

* Instead of storing data as a series of rows, data is organized by columns.
* Since only those needed for storage and stored sequentially, fewer IOs are used.
* Better compression
* Does not need indexes, so less storage requirement
* Automatically analyses and compresses optimally

**Massive Parallel Processing (MPP)**

Automatically distributes data and query load across all nodes

Pricing

* Compute node hours
* No Leader node charges
* Backup
* Data Transfer (within VPC)

Security

* Encrypted using SSL
* AES 256
* Default Redshift manages keys automatically
* Can use HSP or KMS
* Available on 1 AZ only (performance)
* Can restore snapshots in other AZs

## Elasticache

* ***TIP: Use for Read heavy and relatively static data***
* Easy to deploy, operate and scale an in-memory cache in the cloud.
* Improve performance by avoiding repeated calls to DB.
* Improve latency and throughput for read-heavy applications.
* Can be used for compute intensive data
* Types
  + Memcached
    - Protocol compatible so existing tooling can port over
  + Redis
    - Key value store supporting data structures like sorted sets and lists
    - Supports Master/slave replication
    - Multi AZ

## Aurora

* Bespoke AWS only Database engine
* MySQL compatible
* 5X performance
* .1X cost
* No push button scaling without downtime (unlike DynamoDB)
* **Scaling**
  + Starts with 10Gb, scales in 10Gb incrments to 64Tb
  + Compute scale to 32vCPUs and 244 GB memory
* **Fault tolerance**
  + 2 copies of data in each of at least 3 AZs (6 copies of data)
  + Transparently handle loss of up to 2 copies of data without affecting Writes and 3 copies without affecting Reads
  + Self-healing
  + Failover is in order of tier0, tier1, tier2…
* **Replicas**
  + Aurora Replicas
    - Up to 15
    - Automatically failover
    - Cluster Endpoint
  + MySQL Read Replicas
    - Up to 5
    - Does not automatically failover to a read replica
    - Individual Endpoint

## Exam Tips

* Why you can’t connect to DB Server from DMZ. Check the security group – if it is removed or added
* Have separate groups for EC2 Instance and RDS Instance.
* Multi-AZ for Disaster Recovery only. Not for performance improvement. For performance improvement use, multiple read-replicas
* Dynamo DB v/s RDS

If you want push button scaling, without any downtown, you will always want to use DynamoDB.

With RDS scaling is not so easy, you have to use a bigger instance or add read replicas (manual process).

* If you are using Amazon RDS Provisioned IOPS storage with a MySQL or Oracle database engine, what is the maximum size RDS volume you can have by default? – **6TB**
* What data transfer charge is incurred when replicating data from your primary RDS instance to your secondary RDS instance? - **There is no charge associated with this action**.
* When you have deployed an RDS database into multiple availability zones, can you use the secondary database as an independent read node? – **No**
* RDS automatically creates RDS Security Group w/ TCP port # 3306 enabled.
* In VPC Security Group, the answer would be YES because you will have manually specify access to port & protocol.