

Capstone Project - 2

1. Created a MySQL RDS

The screenshot shows the AWS RDS 'Create database' wizard and the resulting RDS console.

Create database (Left Panel):

- Choose a database creation method:**
 - 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:**
 - Aurora (MySQL Compatible)
 - Aurora (PostgreSQL Compatible)
 - MySQL
 - MariaDB

RDS Console (Right Panel):

- DB instance identifier:** db1
- Credentials Settings:**
 - Master username:** admin
 - Credentials management:**
 - Managed in AWS Secrets Manager - most secure: RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.
 - Self managed: Create your own password or have RDS create a password that you manage.
 - Auto generate password:** Amazon RDS can generate a password for you, or you can specify your own password.
- Master password:** (Redacted)
- Confirm master password:** (Redacted)

Amazon RDS Dashboard (Bottom Left):

- Databases:** Shows a single database named **db1** in the process of being created.
- Actions:** Group resources, Modify, Actions, Restore from S3, Create database.

2. Created an environment of PHP with sample code and managed the same security group as instances “Launch wizard 5”

The screenshot shows the AWS Elastic Beanstalk Launch Wizard and the resulting environment overview.

Elastic Beanstalk Launch Wizard (Left Panel):

- Step 1: Configure environment**
 - Environment information:** Environment tier: Web server environment, Application name: php-app, Application code: Sample application.
 - Environment tier:** PHP application running on Amazon Linux 2023/4.0.9.
- Step 2: Configure service access**
 - Service access:** Service role: arn:aws:iam::339712851316:role/service-role/elasticbeanstalkfull, EC2 key pair: kk, EC2 instance profile: elasticbeanstalkfull.
- Step 3 - optional: Set up networking, database, and tags**
- Step 4 - optional: Configure instance traffic and scaling**
- Step 5 - optional: Configure updates, monitoring, and logging**
- Step 6: Review**

Elastic Beanstalk Environment Overview (Right Panel):

- Environment successfully launched.**
- Environment overview:**
 - Health:** Green
 - Environment ID:** e-ummmppbw
 - Domain:** Ph-app-envv.eba-m67m352.us-west-2.elasticbeanstalk.com
 - Application:** php-app
 - Platform:** Python 3.11 running on 64bit Amazon Linux 2023/4.0.9
 - Running version:** -
 - Platform state:** Supported
- Events (12):**
 - March 25, 2024 08:33:43 (UTC+5:30) INFO Successfully launched environment: Ph-app-envv
 - March 25, 2024 08:33:42 (UTC+5:30) INFO Application available at Ph-app-envv.eba-m67m352.us-west-2.elasticbeanstalk.com

The screenshot shows the 'Instance traffic and scaling' configuration page for an environment named 'php-app-envvv'. The left sidebar lists navigation options like Applications, Environments, and Configuration. Under Configuration, 'Events', 'Health', 'Logs', 'Monitoring', 'Alarms', 'Managed updates', and 'Tags' are listed. The main content area displays various settings:

Instances		
IMDSv1	EC2 Security Groups	sg-0a61a8c770dd9e93f, sg-0253145153fc5214e
Deactivated		
Capacity		
Environment type	Fleet composition	On-demand base
Single instance	On-Demand instance	0
On-demand above base	Capacity rebalancing	Scaling cooldown
0	Deactivated	360
Processor type	Instance types	AMI ID
x86_64	t3.micro, t3.small	ami-0f2bea1a4f2acf364
Availability Zones		
Any		
Load balancer		
Load balancer visibility	Load balancer subnets	
Any	Subnet 1, Subnet 2	

At the bottom, there are links for CloudShell, Feedback, © 2024, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.

3. Attached the RDS with the environment by mentioning the hostname and the port

The screenshot shows the 'Configuration' page for the same environment 'php-app-envvv'. A green success message box at the top says 'Environment successfully launched.' The configuration details include:

Setting	Value	Description	Status
Port	600	Port number	Ok
Ignore health check	false	Ignore health check	Instance replacement
Platform software			
Lifecycle	false	Log streaming	NumProcesses
NumThreads	15	Deactivated	1
Logs retention	7	WSGIPath	Proxy server
X-Ray enabled	Deactivated	application	nginx
Environment properties			
Key	Value		
db1.cnso80ui8fi7.us-west-2.rds.amazonaws.com	3306		
PYTHONPATH	/var/app/venv/staging-LQM1test/bin		

At the bottom, there are links for CloudShell, Feedback, © 2024, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.