Deploying a web application in split-configuration in a two availability zones with secure, scalable and highly available configuration

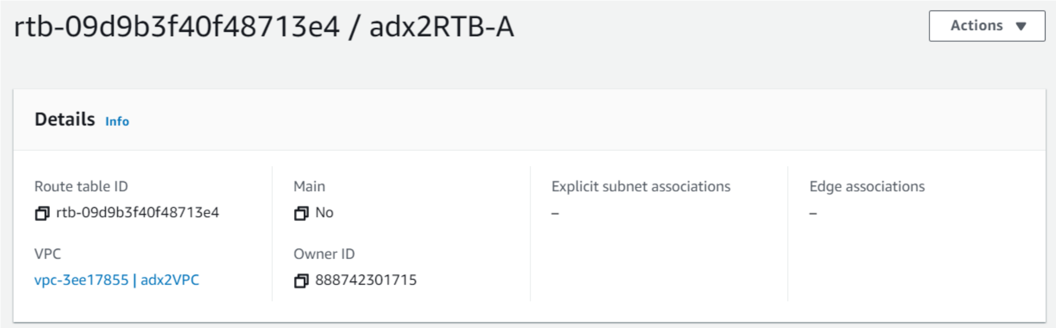
1. Provision of AWS Stack and Security

- Use the default VPC adx2VPC and default Internet Gateway adx2IGW from Task 3.

Use us-east-2c as Available Zone A for:	Use us-east-2b as Available Zone B for:
• adx2RTB-A • adx2PublicSubnetA (default subnet-3280a77e) • adx2PrivateSubnetA ○ adx2WebserverA1 ○ adx2WebServerA2 ○ Amazon RDS	• adx2RTB-B • adx2PublicSubnetB (default subnet-123dc26f) • adx2PrivateSubnetB ○ adx2WebserverB1 ○ adx2WebServerB2

Availability Zone A

- Create Route Table adx2RTB-A
    - Navigate to VPC Dashboard, click “Route Tables”, “Create route table”
        - Name: adx2RTB-A
        - VPC: adx2VPC
        - Edit routes: Add route: Destination 0.0.0.0/0, Target: adx2IGW

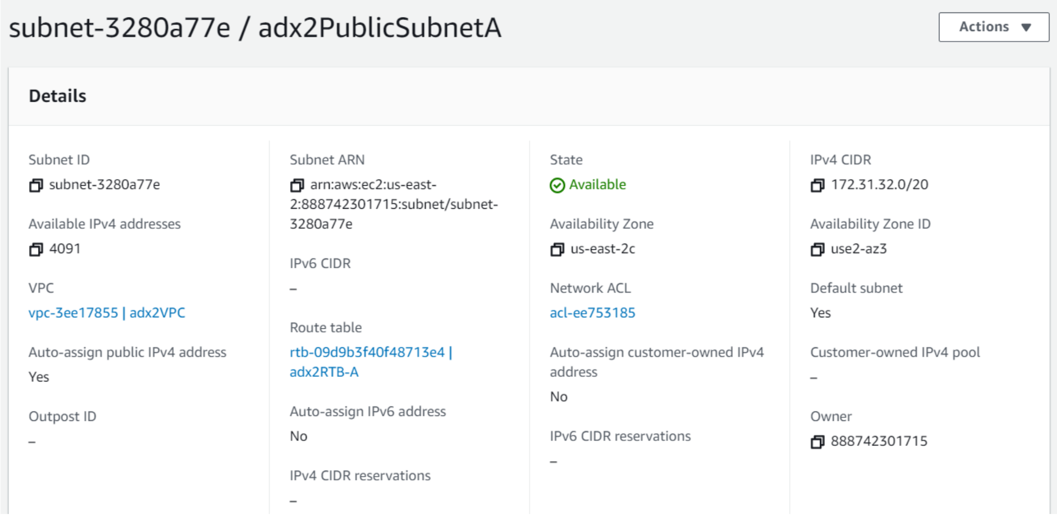


rtb-09d9b3f40f48713e4 / adx2RTB-A

Details Info

Route table ID	Main	Explicit subnet associations	Edge associations
rtb-09d9b3f40f48713e4	No	–	–
VPC	Owner ID		
vpc-3ee17855 | adx2VPC	888742301715		

- Configure adx2PubicSubnetA
    - Add tags: Key: Name, Value: adx2PublicSubnetA
    - Edit route table associations: adx2RTB-A



subnet-3280a77e / adx2PublicSubnetA

Details

Subnet ID: subnet-3280a77e

Available IPv4 addresses: 4091

VPC: vpc-3ee17855 | adx2VPC

Auto-assign public IPv4 address: Yes

Outpost ID: –

Subnet ARN: arn:aws:ec2:us-east-2:888742301715:subnet/subnet-3280a77e

IPv6 CIDR: –

Route table: rtb-09d9b3f40f48713e4 | adx2RTB-A

Auto-assign IPv6 address: No

IPv4 CIDR reservations: –

State: ✓ Available

Availability Zone: us-east-2c

Network ACL: acl-ee753185

Auto-assign customer-owned IPv4 address: No

IPv6 CIDR reservations: –

IPv4 CIDR: 172.31.32.0/20

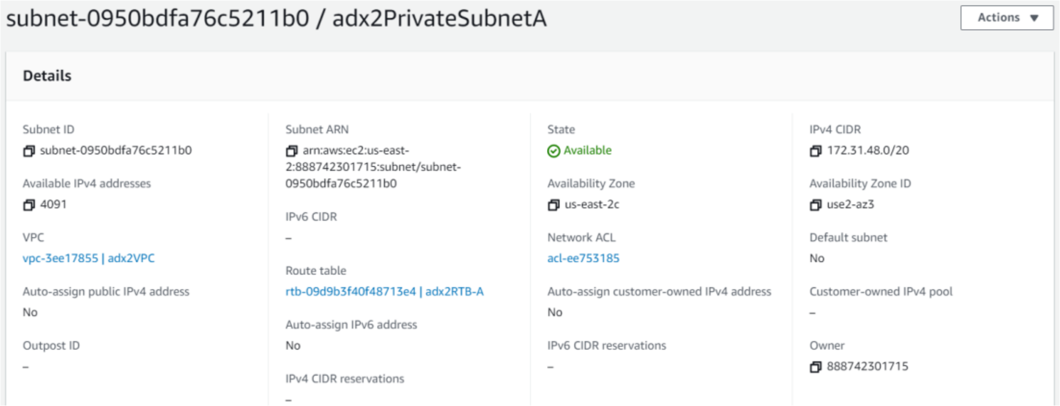
Availability Zone ID: use2-az3

Default subnet: Yes

Customer-owned IPv4 pool: –

Owner: 888742301715

- Create adx2PrivateSubnetA
    - VPC ID: adx2VPC
    - Subnet name: adx2PrivateSubnetA
    - Availability Zone: us-east-2c (Availity Zone A)
    - IPv4 CIDR block: 172.31.48.0/20
    - Edit route table associations: adx2RTB-A
    - Auto-assign public IPv4 address: No



subnet-0950bdfa76c5211b0 / adx2PrivateSubnetA

Details

Subnet ID
subnet-0950bdfa76c5211b0

Available IPv4 addresses
4091

VPC
vpc-3ee17855 | adx2VPC

Auto-assign public IPv4 address
No

Outpost ID
–

Subnet ARN
arn:aws:ec2:us-east-2:888742301715:subnet/subnet-0950bdfa76c5211b0

IPv6 CIDR
–

Route table
rtb-09d9b3f40f48713e4 | adx2RTB-A

Auto-assign IPv6 address
No

IPv4 CIDR reservations
–

State
Available

Availability Zone
us-east-2c

Network ACL
acl-ee753185

Auto-assign customer-owned IPv4 address
No

IPv6 CIDR reservations
–

IPv4 CIDR
172.31.48.0/20

Availability Zone ID
use2-az3

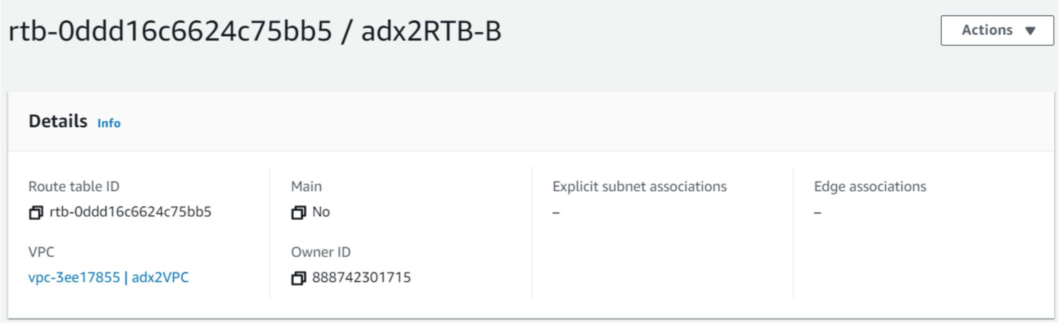
Default subnet
No

Customer-owned IPv4 pool
–

Owner
888742301715

Availability Zone B

- Create Route Table adx2RTB-B
    - Navigate to VPC Dashboard, click “Route Tables”, “Create route table”
        - Name: adx2RTB-B
        - VPC: adx2VPC
        - Edit routes: Add route: Destination 0.0.0.0/0, Target: adx2IGW



rtb-0ddd16c6624c75bb5 / adx2RTB-B

Details Info

Route table ID
rtb-0ddd16c6624c75bb5

VPC
vpc-3ee17855 | adx2VPC

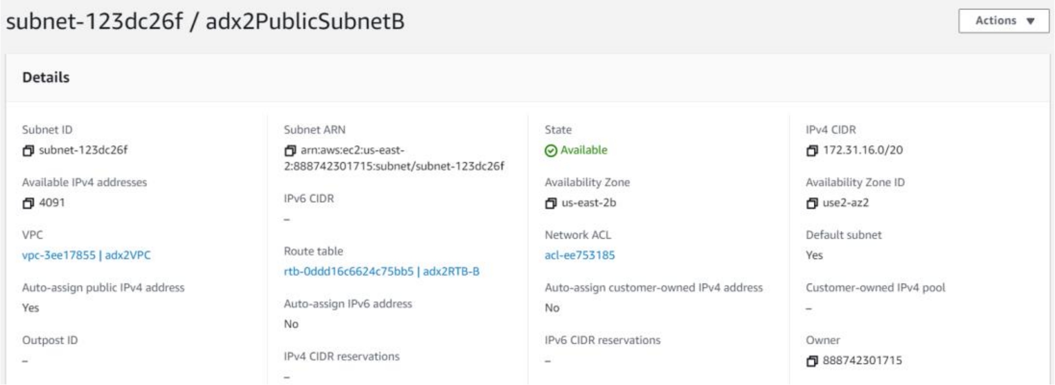
Main
No

Owner ID
888742301715

Explicit subnet associations
–

Edge associations
–

- Configure adx2PubicSubnetB
    - Add tags: Key: Name, Value: adx2PublicSubnetB
    - Edit route table associations: adx2RTB-B



subnet-123dc26f / adx2PublicSubnetB

Details

Subnet ID
subnet-123dc26f

Available IPv4 addresses
4091

VPC
vpc-3ee17855 | adx2VPC

Auto-assign public IPv4 address
Yes

Outpost ID
–

Subnet ARN
arn:aws:ec2:us-east-2:888742301715:subnet/subnet-123dc26f

IPv6 CIDR
–

Route table
rtb-0ddd16c6624c75bb5 | adx2RTB-B

Auto-assign IPv6 address
No

IPv4 CIDR reservations
–

State
Available

Availability Zone
us-east-2b

Network ACL
acl-ee753185

Auto-assign customer-owned IPv4 address
No

IPv6 CIDR reservations
–

IPv4 CIDR
172.31.16.0/20

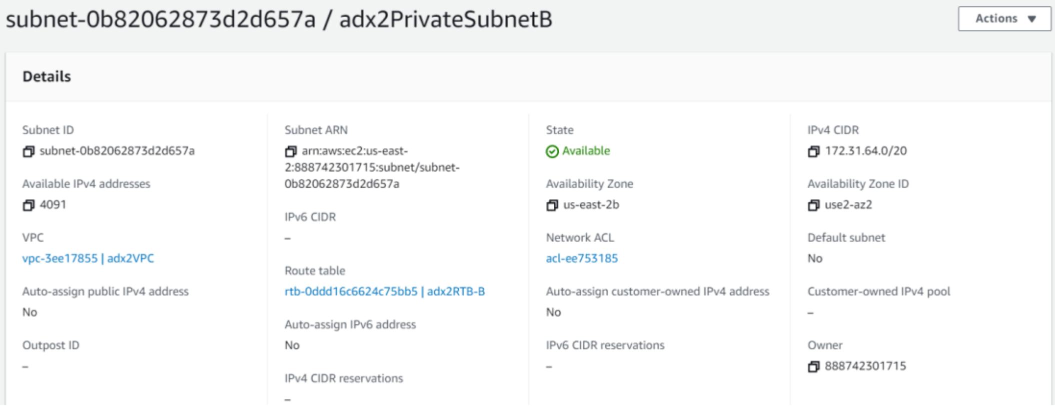
Availability Zone ID
use2-az2

Default subnet
Yes

Customer-owned IPv4 pool
–

Owner
888742301715

- Create adx2PrivateSubnetB
  - VPC ID: adx2VPC
  - Subnet name: adx2PrivateSubnetB
  - Availability Zone: us-east-2b (Availity Zone B)
  - IPv4 CIDR block: 172.31.64.0/20
  - Edit route table associations: adx2RTB-B
  - Auto-assign public IPv4 address: No



subnet-0b82062873d2d657a / adx2PrivateSubnetB

Details

Subnet ID
subnet-0b82062873d2d657a

Available IPv4 addresses
4091

VPC
vpc-3ee17855 | adx2VPC

Auto-assign public IPv4 address
No

Outpost ID
–

Subnet ARN
arn:aws:ec2:us-east-2:888742301715:subnet/subnet-0b82062873d2d657a

IPv6 CIDR
–

Route table
rtb-0ddd16c6624c75bb5 | adx2RTB-B

Auto-assign IPv6 address
No

IPv4 CIDR reservations
–

State
Available

Availability Zone
us-east-2b

Network ACL
acl-ee753185

Auto-assign customer-owned IPv4 address
No

IPv6 CIDR reservations
–

IPv4 CIDR
172.31.64.0/20

Availability Zone ID
use2-az2

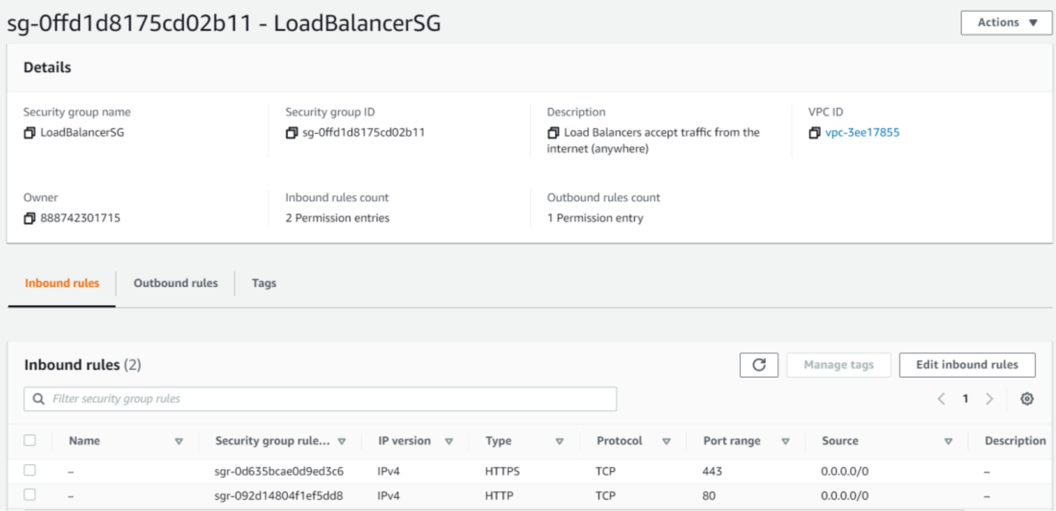
Default subnet
No

Customer-owned IPv4 pool
–

Owner
888742301715

Security Groups

- Create LoadBalancerSG
  - Security group name: LoadBalancerSG
  - Inbound rules:
    - HTTP Protocol TCP Port 80 Source 0.0.0.0/0
    - HTTPS Protocol TCP Port 443 Source 0.0.0.0/0
  - Outbound rules:
    - All traffic Protocol All Port all Destination 0.0.0.0/0



sg-0ffd1d8175cd02b11 - LoadBalancerSG

Details

Security group name
LoadBalancerSG

Security group ID
sg-0ffd1d8175cd02b11

Description
Load Balancers accept traffic from the internet (anywhere)

VPC ID
vpc-3ee17855

Owner
888742301715

Inbound rules count
2 Permission entries

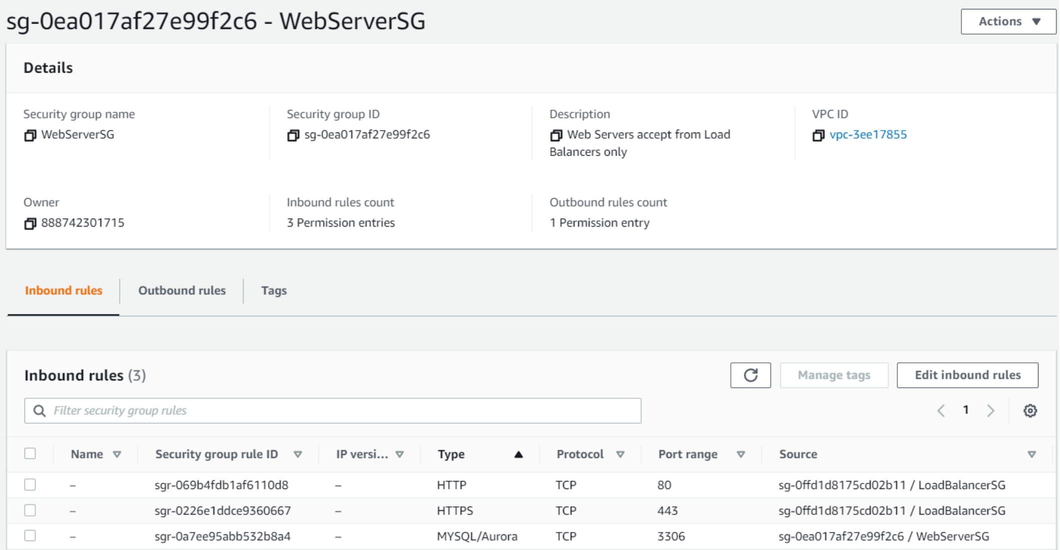
Outbound rules count
1 Permission entry

Inbound rules (2)

□	Name	Security group rule...	IP version	Type	Protocol	Port range	Source	Description
□	–	sgr-0d635bcae0d9ed3c6	IPv4	HTTPS	TCP	443	0.0.0.0/0	–
□	–	sgr-092d14804f1ef5dd8	IPv4	HTTP	TCP	80	0.0.0.0/0	–

- Create WebServerSG
  - Security group name: WebServerSG
  - Inbound rules:
    - HTTP Protocol TCP Port 80 Source LoadBalanceSG sg-0ffd1d8175cd02b11
    - HTTPS Protocol TCP Port 443 Source LoadBalanceSG sg-0ffd1d8175cd02b11
    - MySQL/Aurora TCP Port 3306 Source WebServerSG sg-0ea017af27e99f2c6 (to enable communication between Web Servers and RDS database)

- Outbound rules:
  - All traffic Protocol All Port all Destination 0.0.0.0/0



sg-0ea017af27e99f2c6 - WebServerSG

Details

Security group name: WebServerSG

Security group ID: sg-0ea017af27e99f2c6

Description: Web Servers accept from Load Balancers only

VPC ID: vpc-3ee17855

Owner: 888742301715

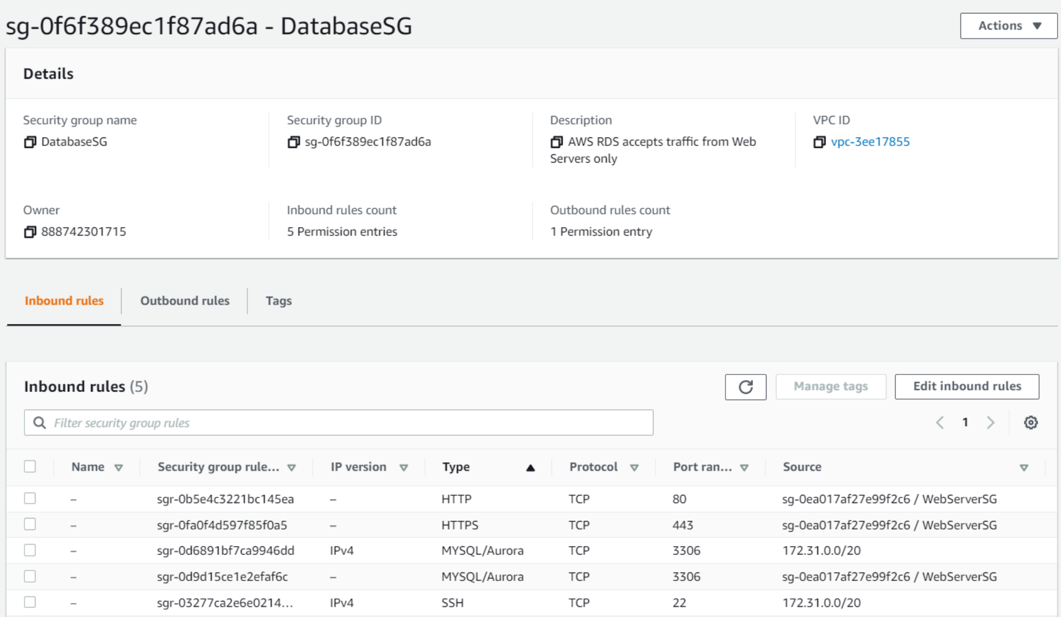
Inbound rules count: 3 Permission entries

Outbound rules count: 1 Permission entry

Inbound rules (3)

□	Name	Security group rule ID	IP versi...	Type	Protocol	Port range	Source
□	–	sgr-069b4fdb1af6110d8	–	HTTP	TCP	80	sg-0ffd1d8175cd02b11 / LoadBalancerSG
□	–	sgr-0226e1ddce9360667	–	HTTPS	TCP	443	sg-0ffd1d8175cd02b11 / LoadBalancerSG
□	–	sgr-0a7ee95abb532b8a4	–	MYSQL/Aurora	TCP	3306	sg-0ea017af27e99f2c6 / WebServerSG

- Create DatabaseSG
  - Security group name: DatabaseSG
  - Inbound rules:
    - HTTP Protocol TCP Port 80 Source WebServerSG sg-0ea017af27e99f2c6
    - HTTPS Protocol TCP Port 443 Source WebServerSG sg-0ea017af27e99f2c6
    - MySQL/Aurora TCP Port 3306 Source WebServerSG sg-0ea017af27e99f2c6
    - MySQL/Aurora TCP Port 3306 Source 172.31.0.0/20
      (This is the subnet mask of adx2EC2 I will use for SSH access)
    - SSH Protocol TCP Port 22 Source 172.31.0.0/20
      (To enable SSH access for Task 4.2)
  - Outbound rules:
    - All traffic Protocol All Port all Destination 0.0.0.0/0



sg-0f6f389ec1f87ad6a - DatabaseSG

Details

Security group name: DatabaseSG

Security group ID: sg-0f6f389ec1f87ad6a

Description: AWS RDS accepts traffic from Web Servers only

VPC ID: vpc-3ee17855

Owner: 888742301715

Inbound rules count: 5 Permission entries

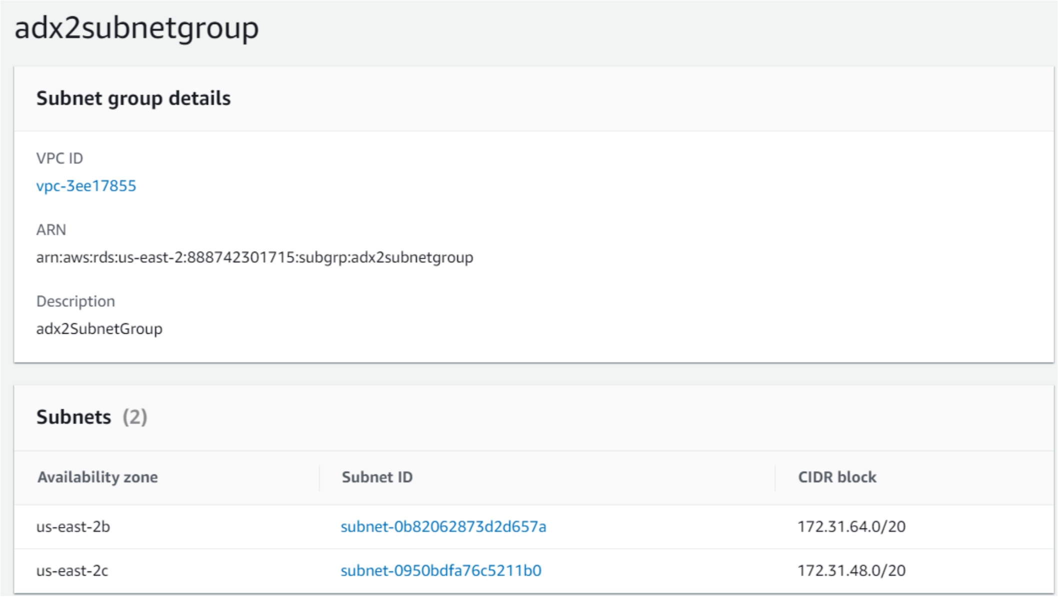
Outbound rules count: 1 Permission entry

Inbound rules (5)

□	Name	Security group rule...	IP version	Type	Protocol	Port ran...	Source
□	–	sgr-0b5e4c3221bc145ea	–	HTTP	TCP	80	sg-0ea017af27e99f2c6 / WebServerSG
□	–	sgr-0fa0f4d597f85f0a5	–	HTTPS	TCP	443	sg-0ea017af27e99f2c6 / WebServerSG
□	–	sgr-0d6891bf7ca9946dd	IPv4	MYSQL/Aurora	TCP	3306	172.31.0.0/20
□	–	sgr-0d9d15ce1e2efaf6c	–	MYSQL/Aurora	TCP	3306	sg-0ea017af27e99f2c6 / WebServerSG
□	–	sgr-03277ca2e6e0214...	IPv4	SSH	TCP	22	172.31.0.0/20

2. Create an RDS Instance

- Create a Subnet Group
    - Name: adx2SubnetGroup
    - VPC: adx2VPC
    - Availability Zones:
        - us-east-2c (Availability Zone A)
        - us-east-2b (Availability Zone B)
    - Subnets:
        - adx2PrivateSubnetA (subnet-0950bdfa76c5211b0)
        - adx2PrivateSubnetB (subnet-0b82062873d2d657a)



adx2subnetgroup

Subnet group details

VPC ID
vpc-3ee17855

ARN
arn:aws:rds:us-east-2:888742301715:subgrp:adx2subnetgroup

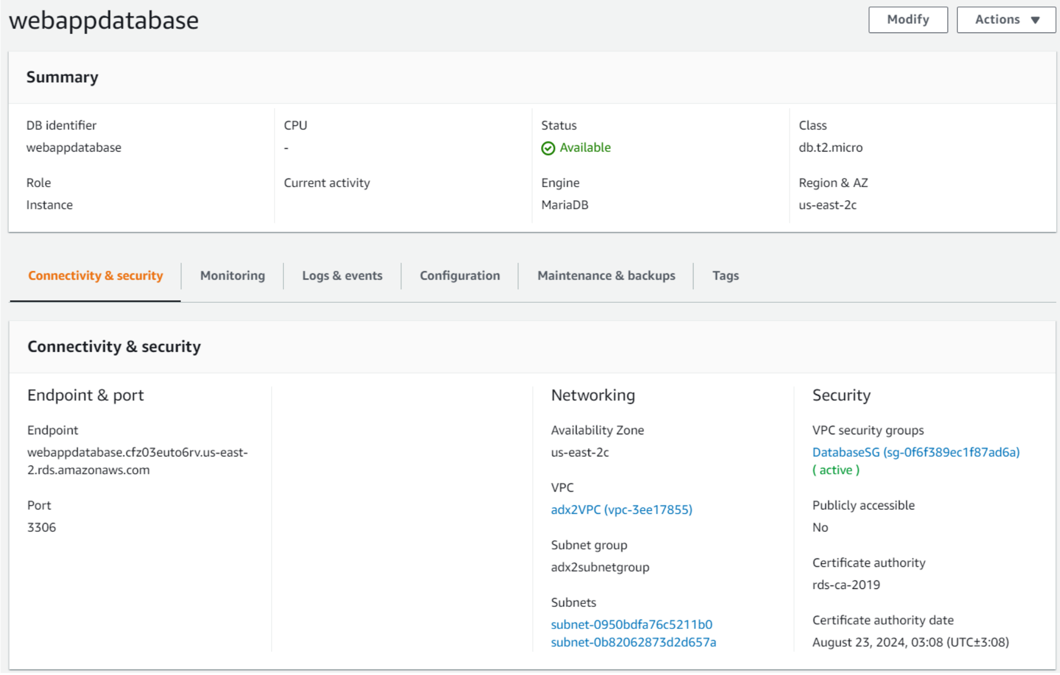
Description
adx2SubnetGroup

Subnets (2)

Availability zone	Subnet ID	CIDR block
us-east-2b	subnet-0b82062873d2d657a	172.31.64.0/20
us-east-2c	subnet-0950bdfa76c5211b0	172.31.48.0/20

- Create an RDS instance
    - Choose a database creation method: Standard create
    - Engine options
        - Engine type: MariaDB, Version: 10.4.13, Templates: Dev/Test
    - Settings
        - DB instance identifier: WebAppDatabase
        - Master username: root
        - Master password: root1234
    - DB instance class: Burstable classes db.t2.micro
    - Storage
        - Storage type: General purpose SSD (gp2)
        - Allocated storage: 20 GiB
    - Availability & durability: Do not create a standby instance
    - Connectivity
        - VPC: a2VPC
        - Subnet group: adx2SubnetGroup
        - Public access: No
        - VPC security group: Choose existing: DatabaseSG
        - Availability Zone: us-east-2c (Availability Zone A)
        - Database port: 3306

- Additional configuration
  - DB parameter group: default.mariadb10.4
  - Option group: default:mariadb-10-4
  - Backup: Enable automated backups
  - Maintenance: Enable auto minor version upgrade
  - Monitoring: Enable Enhanced monitoring: No
  - Delete protection: Enable deletion protection: No



webappdatabase

Summary

DB identifier: webappdatabase
CPU: -
Status: ✓ Available
Class: db.t2.micro
Role: Instance
Current activity
Engine: MariaDB
Region & AZ: us-east-2c

Connectivity & security

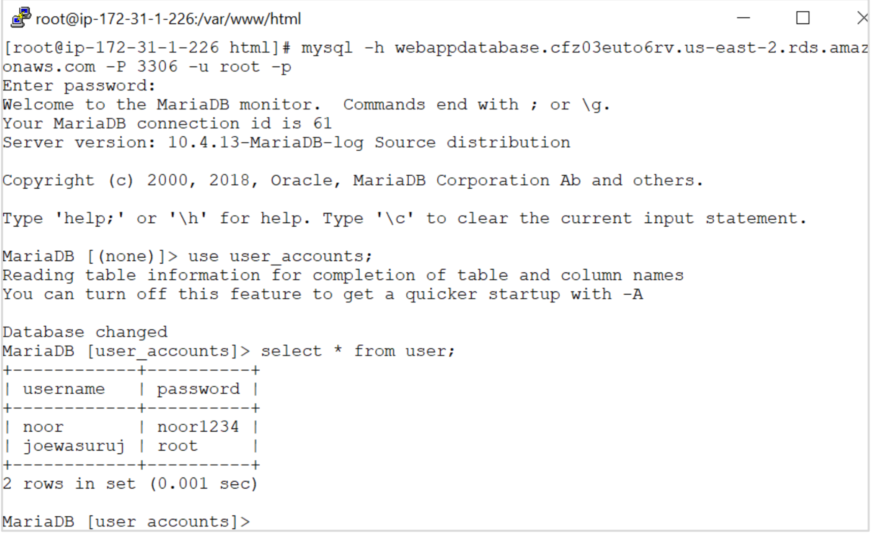
Endpoint & port
Endpoint: webappdatabase.cfz03euto6rv.us-east-2.rds.amazonaws.com
Port: 3306

Networking
Availability Zone: us-east-2c
VPC: adx2VPC (vpc-3ee17855)
Subnet group: adx2subnetgroup
Subnets: subnet-0950bdfa76c5211b0 subnet-0b82062873d2d657a

Security
VPC security groups: DatabaseSG (sg-0f6f389ec1f87ad6a) (active)
Publicly accessible: No
Certificate authority: rds-ca-2019
Certificate authority date: August 23, 2024, 03:08 (UTC±3:08)

- Accessing adx2EC2 from Task 3 via SSH PuTTY using adx2EC2 public IPv4: ec2-user@3.141.12.18
- To transfer data to the RDS database

```
mysql -h webappdatabase.cfz03euto6rv.us-east-2.rds.amazonaws.com -P 3306 -u root -p
// enter password ‘root1234’
create database user_accounts;
mysql -h webappdatabase.cfz03euto6rv.us-east-2.rds.amazonaws.com -P 3306 -u root -p
user_accounts < user_accounts.sql
```


root@ip-172-31-1-226:/var/www/html

```
[root@ip-172-31-1-226 html]# mysql -h webappdatabase.cfz03euto6rv.us-east-2.rds.amaz
onaws.com -P 3306 -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 61
Server version: 10.4.13-MariaDB-log Source distribution

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)]> use user_accounts;
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 [user_accounts]> select * from user;
+------------+----------+
| username   | password |
+------------+----------+
| noor       | noor1234 |
| joewasuruj | root     |
+------------+----------+
2 rows in set (0.001 sec)

MariaDB [user accounts]>
```

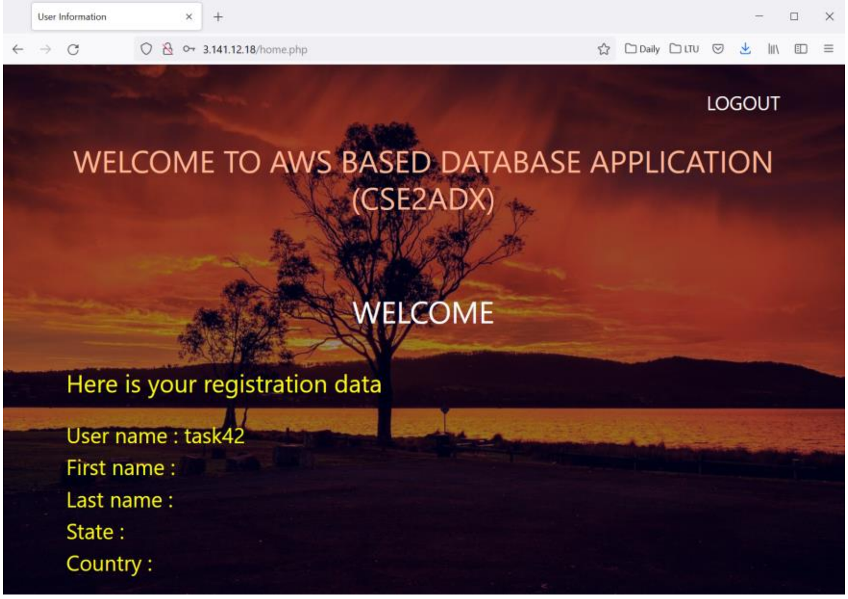
- Modify the validation.php file to point to the RDS database

```
$connection=mysqli_connect('webappdatabase.cfz03euto6rv.us-east-2.rds.amazonaws.com'
,'root', 'root1234', 'user_accounts') or die("COULD NOT CONNCT".mysqli_connect_error
());
```

- Modify the registration.php file to point to the RDS database

```
header('location:login.php');
$connection=mysqli_connect('webappdatabase.cfz03euto6rv.us-east-2.rds.amazonaws.com'
,'root', 'root1234');
mysqli_select_db($connection, 'user_accounts');
```

- Test that the redirection to the RDS database works by opening the adx2EC2 public IPv4 address 3.141.12.18 in a browser and register a new user 'task42'



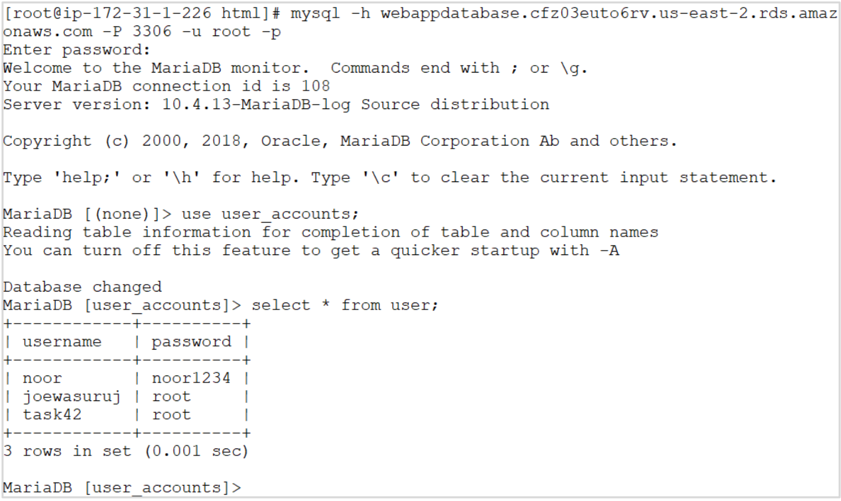
LOGOUT

WELCOME TO AWS BASED DATABASE APPLICATION (CSE2ADX)

WELCOME

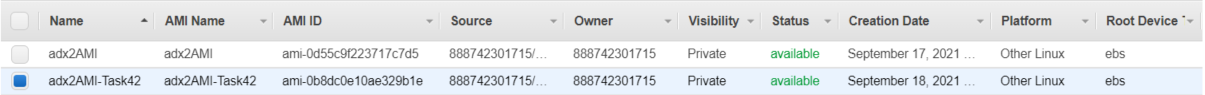
Here is your registration data

User name : task42
First name :
Last name :
State :
Country :



```
[root@ip-172-31-1-226 html]# mysql -h webappdatabase.cfz03euto6rv.us-east-2.rds.amaz
onaws.com -P 3306 -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 108
Server version: 10.4.13-MariaDB-log Source distribution

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)]> use user_accounts;
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 [user_accounts]> select * from user;
+------------+----------+
| username   | password |
+------------+----------+
| noor       | noor1234 |
| joewasuruj | root     |
| task42     | root     |
+------------+----------+
3 rows in set (0.001 sec)

MariaDB [user_accounts]>
```

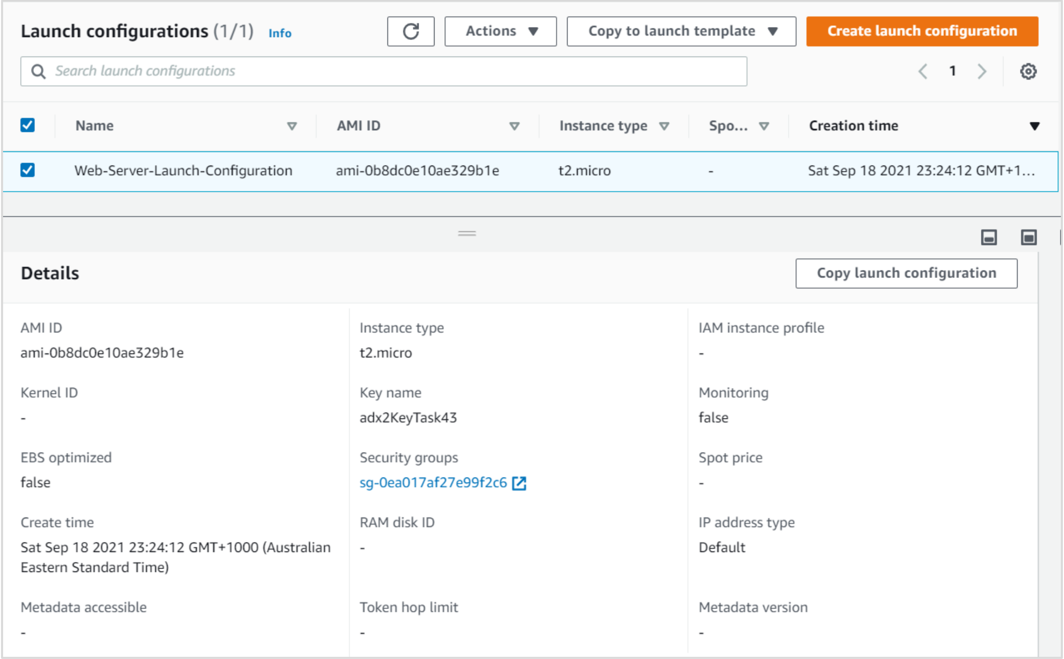
NOTE: The AMI from Task 3 has its web application pointing to the local database. If I use it for the launch configuration, I will have to modify the .php files for all new instances. For efficiency, I will create a new AMI called adx2AMI-Task42 and use it for the subsequent tasks.



□	Name	AMI Name	AMI ID	Source	Owner	Visibility	Status	Creation Date	Platform	Root Device
□	adx2AMI	adx2AMI	ami-0d55c9f223717c7d5	888742301715/...	888742301715	Private	available	September 17, 2021 ...	Other Linux	ebs
■	adx2AMI-Task42	adx2AMI-Task42	ami-0b8dc0e10ae329b1e	888742301715/...	888742301715	Private	available	September 18, 2021 ...	Other Linux	ebs

3. Create a Launch Template

- Create a launch configuration
    - Navigate to EC2 Dashboard, under Auto Scaling, click “Launch configurations”
    - Name: Web-Server-Launch-Configuration
    - AMI: adx2AMI-Task42
    - Instance type: t2.micro
    - Storage: default
    - Security Group: WebServerSG
    - Key pair (login): adx2KeyTask43


Launch configurations (1/1) Info
Actions
Copy to launch template
Create launch configuration
Search launch configurations
1
Copy launch configuration

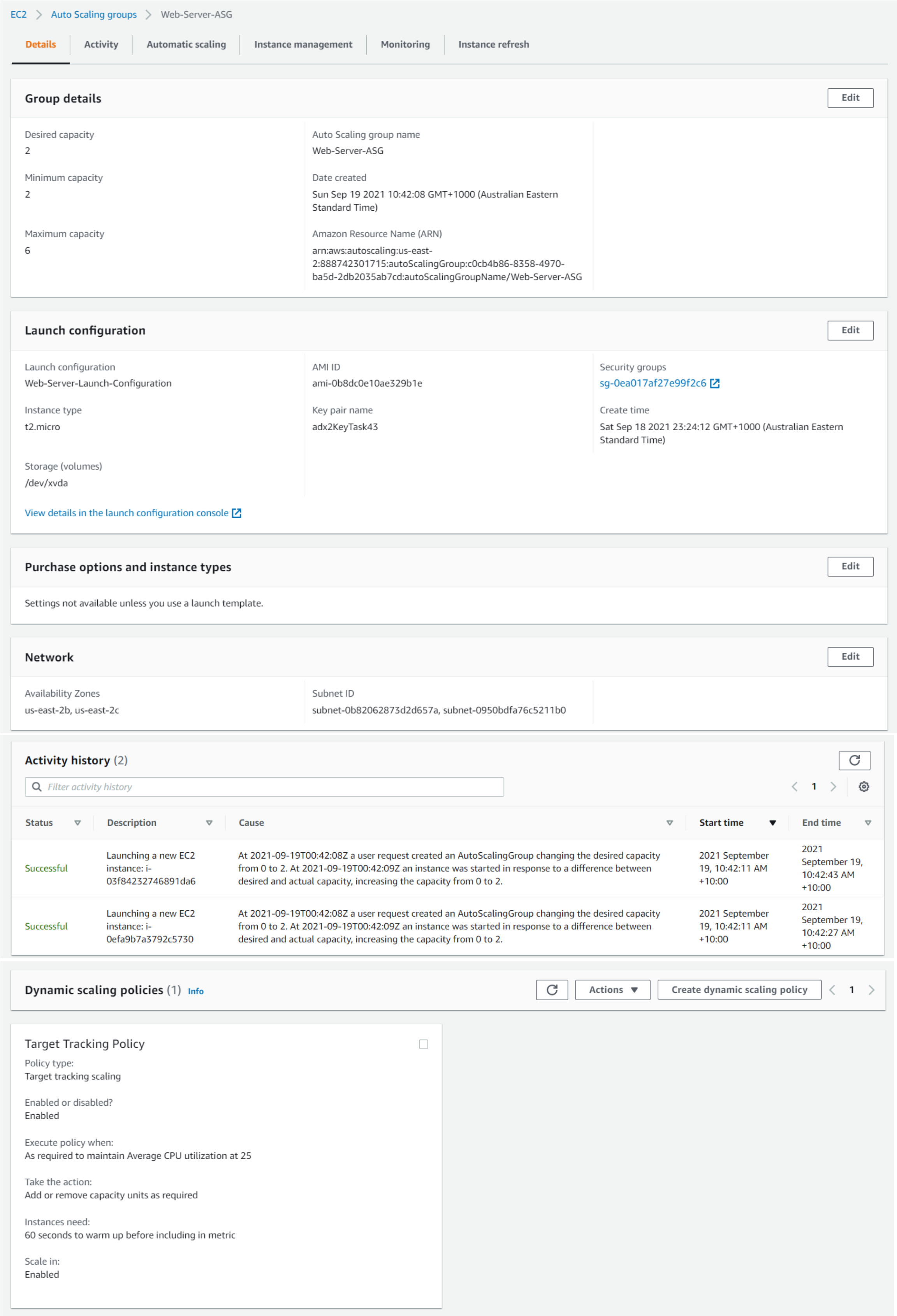
☑	Name	AMI ID	Instance type	Spo...	Creation time
☑	Web-Server-Launch-Configuration	ami-0b8dc0e10ae329b1e	t2.micro	-	Sat Sep 18 2021 23:24:12 GMT+1...

Details

AMI ID ami-0b8dc0e10ae329b1e	Instance type t2.micro	IAM instance profile -
Kernel ID -	Key name adx2KeyTask43	Monitoring false
EBS optimized false	Security groups sg-0ea017af27e99f2c6	Spot price -
Create time Sat Sep 18 2021 23:24:12 GMT+1000 (Australian Eastern Standard Time)	RAM disk ID -	IP address type Default
Metadata accessible -	Token hop limit -	Metadata version -

4. Create an Auto Scaling Group

- Create an Auto Scaling Group
    - Name: Web-Server-ASG
    - Launch configuration: Web-Server-Launch-Configuration
    - VPC: adx2VPC
    - Subnets: adx2PrivateSubnetA & adx2PrivateSubnetB
    - Load balancing: No
    - Desired capacity: 2
    - Minimum capacity: 2
    - Maximum capacity: 6
    - Target tracking scaling policy:
        - Metric type: Average CPU utilization
        - Target value: 25
        - Instances need: 60 seconds warm up before including in metric



Group details

Desired capacity
2

Minimum capacity
2

Maximum capacity
6

Auto Scaling group name
Web-Server-ASG

Date created
Sun Sep 19 2021 10:42:08 GMT+1000 (Australian Eastern Standard Time)

Amazon Resource Name (ARN)
arn:aws:autoscaling:us-east-2:888742301715:autoScalingGroup:c0cb4b86-8358-4970-ba5d-2db2035ab7cd:autoScalingGroupName/Web-Server-ASG

Launch configuration

Launch configuration
Web-Server-Launch-Configuration

Instance type
t2.micro

Storage (volumes)
/dev/xvda

AMI ID
ami-0b8dc0e10ae329b1e

Key pair name
adx2KeyTask43

Security groups
sg-0ea017af27e99f2c6

Create time
Sat Sep 18 2021 23:24:12 GMT+1000 (Australian Eastern Standard Time)

View details in the launch configuration console

Purchase options and instance types

Settings not available unless you use a launch template.

Network

Availability Zones
us-east-2b, us-east-2c

Subnet ID
subnet-0b82062873d2d657a, subnet-0950bdfa76c5211b0

Activity history (2)

Status	Description	Cause	Start time	End time
Successful	Launching a new EC2 instance: i-03f84232746891da6	At 2021-09-19T00:42:08Z a user request created an AutoScalingGroup changing the desired capacity from 0 to 2. At 2021-09-19T00:42:09Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 2.	2021 September 19, 10:42:11 AM +10:00	2021 September 19, 10:42:43 AM +10:00
Successful	Launching a new EC2 instance: i-0efa9b7a3792c5730	At 2021-09-19T00:42:08Z a user request created an AutoScalingGroup changing the desired capacity from 0 to 2. At 2021-09-19T00:42:09Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 2.	2021 September 19, 10:42:11 AM +10:00	2021 September 19, 10:42:27 AM +10:00

Dynamic scaling policies (1) Info

Target Tracking Policy ☐

Policy type:
Target tracking scaling

Enabled or disabled?
Enabled

Execute policy when:
As required to maintain Average CPU utilization at 25

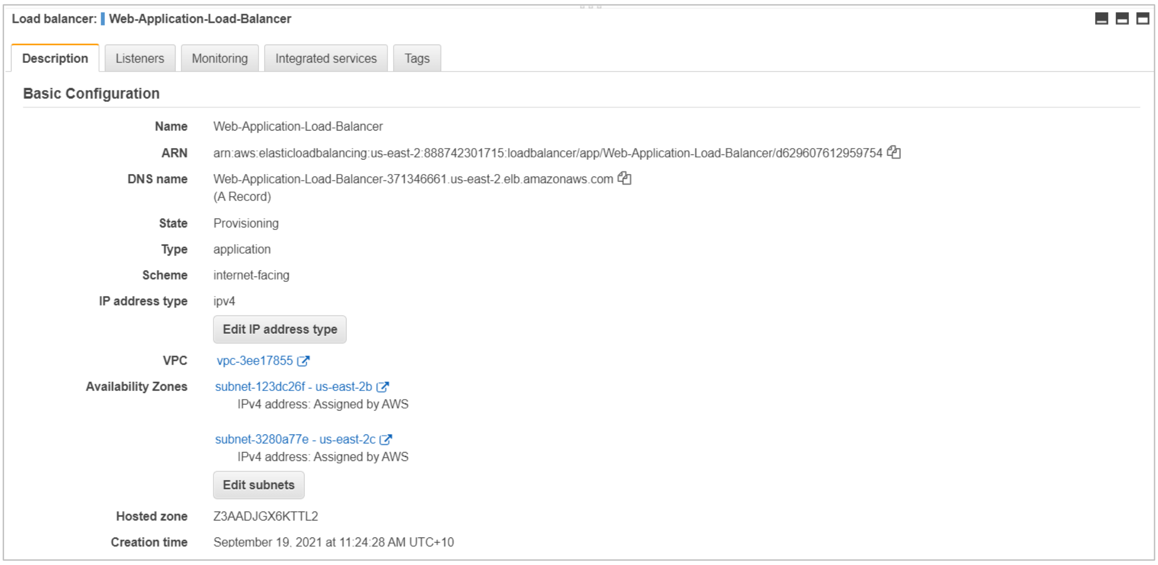
Take the action:
Add or remove capacity units as required

Instances need:
60 seconds to warm up before including in metric

Scale in:
Enabled

5. Create an HTTP Application Load Balancer

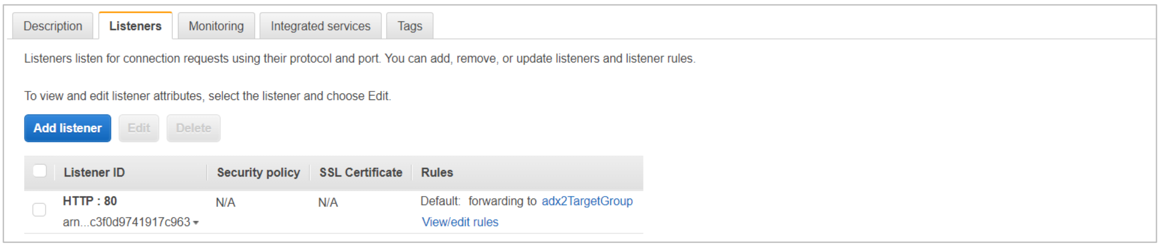
- Create a Target Group adx2TargetGroup for the web servers
    - Navigate to EC2 Dashboard, click “Target Groups”, “Create target group”
    - Target type: Instances
    - Target group name: adx2TargetGroup
    - Protocol: HTTP, Port: 80
    - VPC: adx2VPC
    - Protocol version: HTTP1
    - Health checks: HTTP
    - Register targets: Skips registering targets
- Create an Application Load Balancer
    - Navigate to EC2 Dashboard, click “Load Balancers”, “Create Load Balancer”, “Application Load Balancer”, “Create”
    - Basic configuration:
        - Load balancer name: Web-Application-Load-Balancer
        - Scheme: Internet-facing, IP address type: IPv4
    - Network mapping:
        - VPC: adx2VPC
        - Mappings:
            - us-east-2b & adx2PublicSubnetB
            - us-east-2c & adx2PublicSubnetA
    - Security groups: LoadBalancerSG
    - Listeners and routing:
        - Protocol: HTTP, Port: 80, Forward to: adx2TargetGroup



Load balancer: Web-Application-Load-Balancer

Basic Configuration

Name	Web-Application-Load-Balancer
ARN	arn:aws:elasticloadbalancing:us-east-2:888742301715:loadbalancer/app/Web-Application-Load-Balancer/d629607612959754
DNS name	Web-Application-Load-Balancer-371346661.us-east-2.elb.amazonaws.com (A Record)
State	Provisioning
Type	application
Scheme	internet-facing
IP address type	ipv4 Edit IP address type
VPC	vpc-3ee17855
Availability Zones	subnet-123dc26f - us-east-2b IPv4 address: Assigned by AWS subnet-3280a77e - us-east-2c IPv4 address: Assigned by AWS Edit subnets
Hosted zone	Z3AADJGX6KTTL2
Creation time	September 19, 2021 at 11:24:28 AM UTC+10

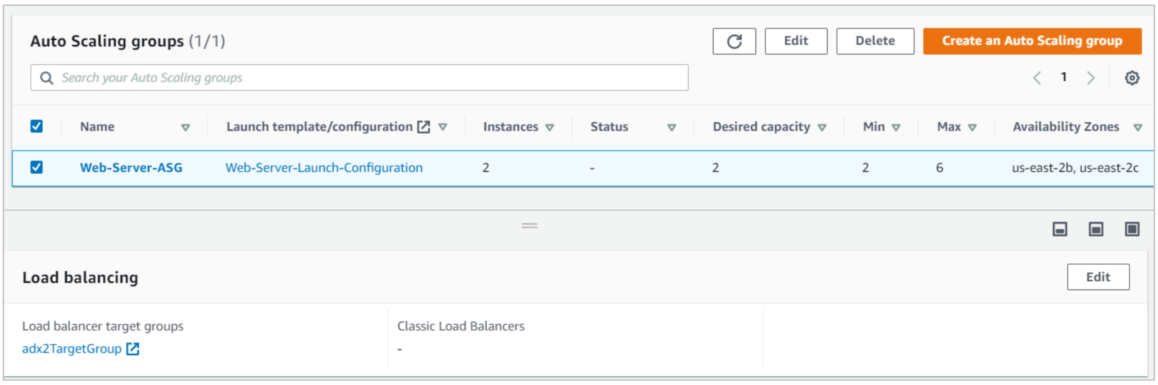


Listeners listen for connection requests using their protocol and port. You can add, remove, or update listeners and listener rules.

To view and edit listener attributes, select the listener and choose Edit.

Listener ID	Security policy	SSL Certificate	Rules
HTTP : 80 arn...c3f0d9741917c963	N/A	N/A	Default: forwarding to adx2TargetGroup View/edit rules

- Add the Load Balancer adx2TargetGroup to the Auto Scaling Group Web-Server-ASG
    - In Auto Scaling Groups, select ‘Web-Server-ASG’
    - Go to Load balancing, click “Edit”
    - Application Load Balancer target groups: adx2TargetGroup HTTP



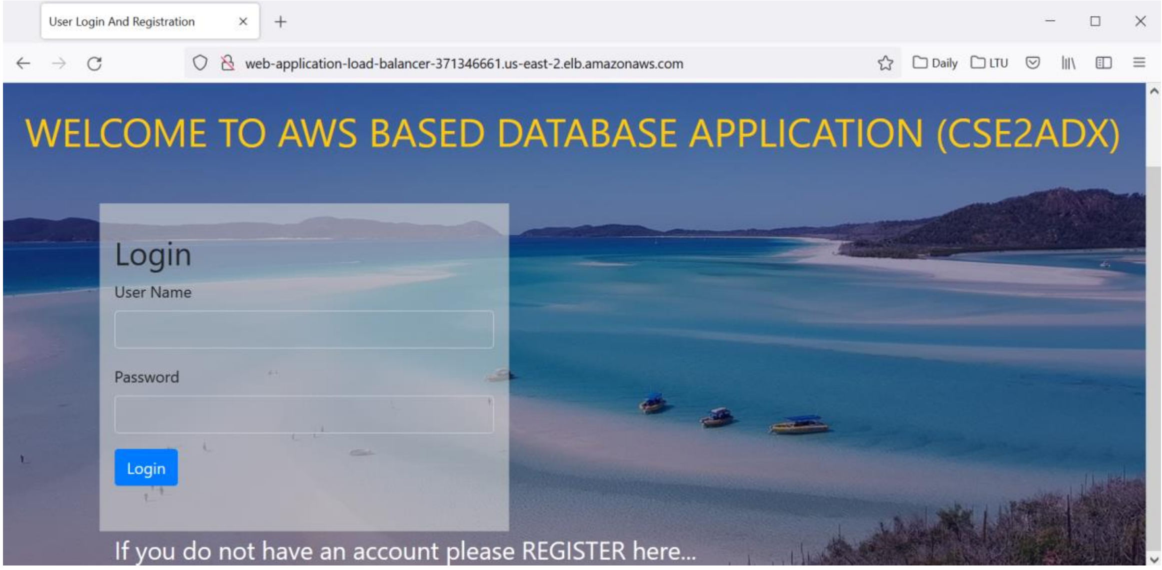
Auto Scaling groups (1/1)

☑	Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Availability Zones
☑	Web-Server-ASG	Web-Server-Launch-Configuration	2	-	2	2	6	us-east-2b, us-east-2c

Load balancing

Load balancer target groups	Classic Load Balancers
adx2TargetGroup	-

- LB DNS name: Web-Application-Load-Balancer-371346661.us-east-2.elb.amazonaws.com
- Test the database web application by opening the LB DNS name in a browser


User Login And Registration
web-application-load-balancer-371346661.us-east-2.elb.amazonaws.com
Daily
LTU

WELCOME TO AWS BASED DATABASE APPLICATION (CSE2ADX)

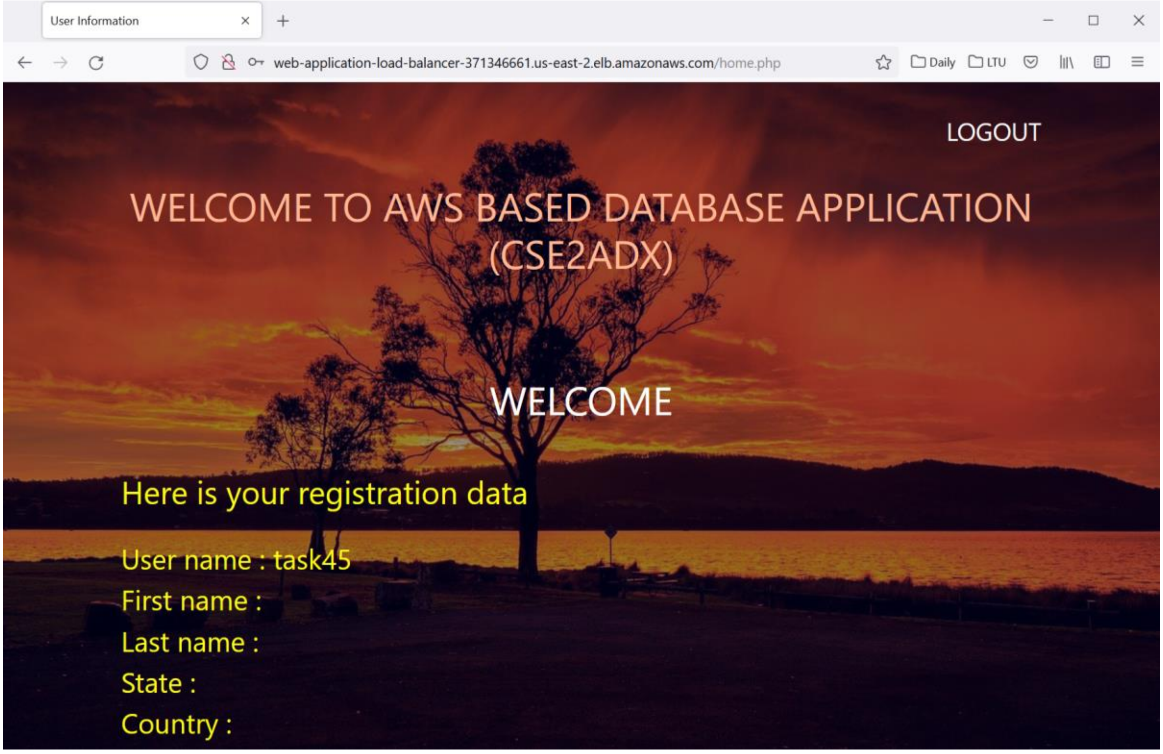
Login

User Name

Password

Login

If you do not have an account please REGISTER here...

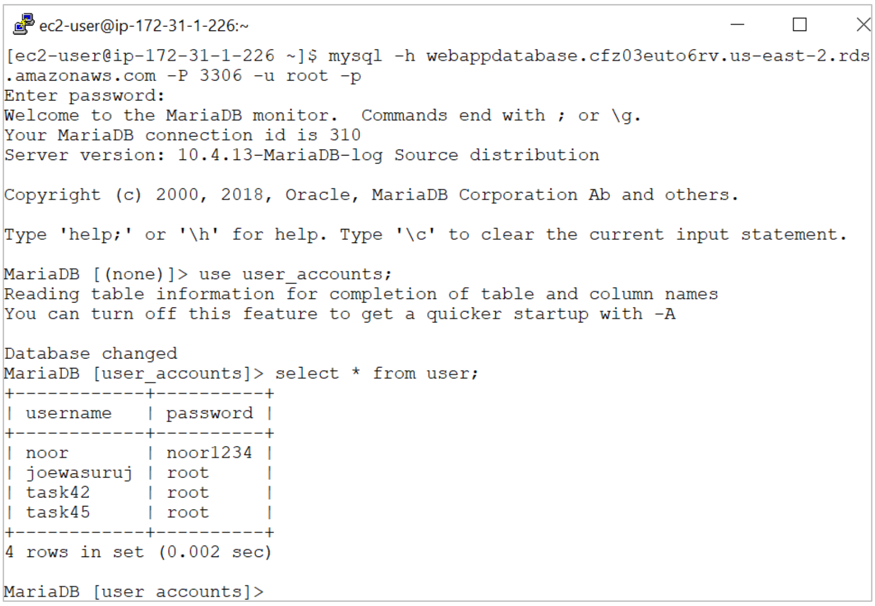

User Information
web-application-load-balancer-371346661.us-east-2.elb.amazonaws.com/home.php
Daily
LTU
LOGOUT

WELCOME TO AWS BASED DATABASE APPLICATION (CSE2ADX)

WELCOME

Here is your registration data

User name : task45
First name :
Last name :
State :
Country :



```
[ec2-user@ip-172-31-1-226 ~]$ mysql -h webappdatabase.cfz03euto6rv.us-east-2.rds
.amazonaws.com -P 3306 -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 310
Server version: 10.4.13-MariaDB-log Source distribution

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)]> use user_accounts;
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 [user_accounts]> select * from user;
+------------+----------+
| username   | password |
+------------+----------+
| noor       | noor1234 |
| joewasuruj | root     |
| task42     | root     |
| task45     | root     |
+------------+----------+
4 rows in set (0.002 sec)

MariaDB [user accounts]>
```

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