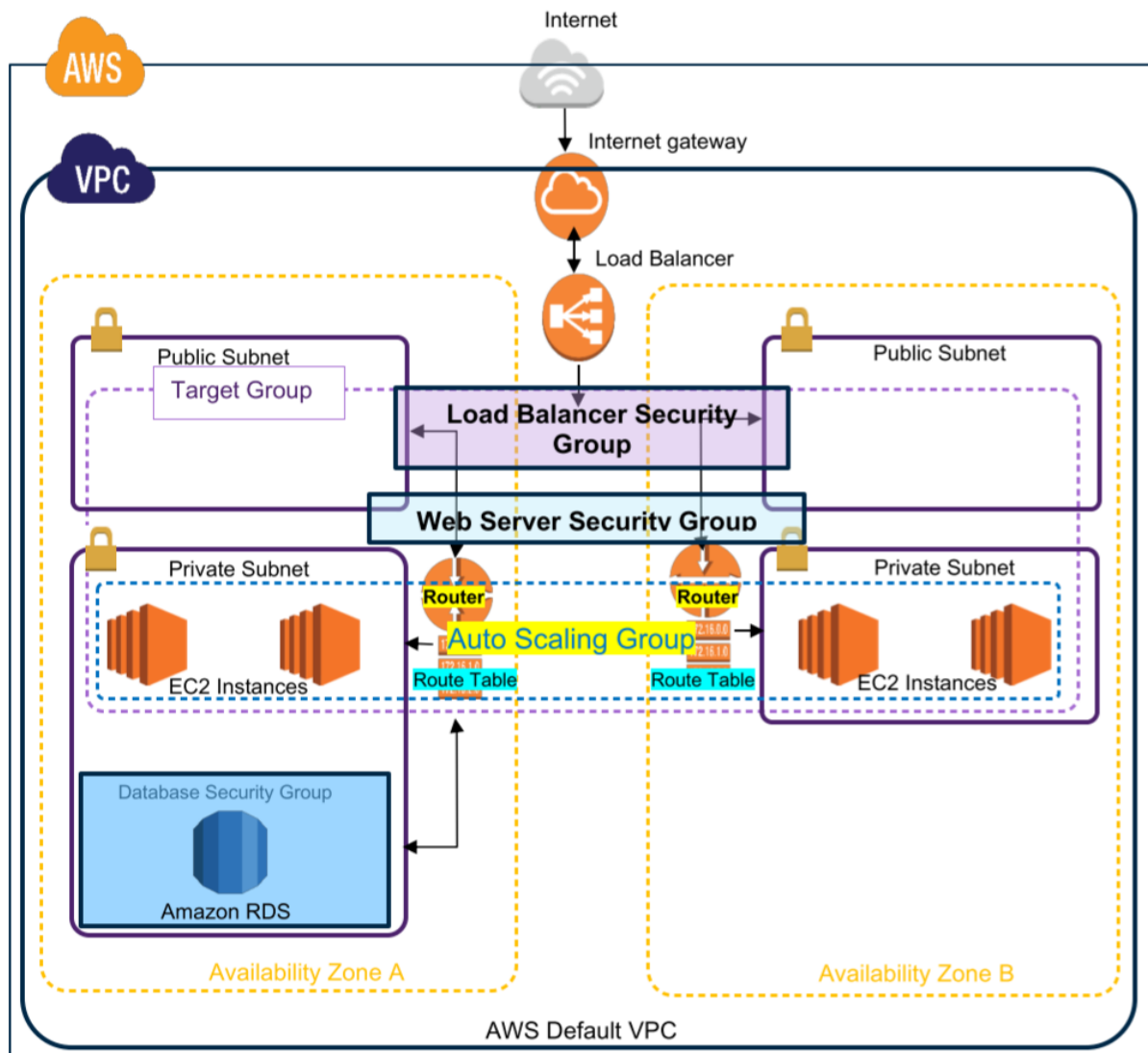


## Deploying a web application in split-configuration in a two availability zones with secure, scalable and highly available configuration



### 1. Provision of AWS Stack and Security

- We will use the following structure:

<p>Use us-east-2c as <b>Available Zone A</b> for:</p> <ul style="list-style-type: none"> <li>adx2RTB-A</li> <li>adx2PublicSubnetA (default subnet-3280a77e)</li> <li>adx2PrivateSubnetA <ul style="list-style-type: none"> <li>adx2WebserverA1</li> <li>adx2WebServerA2</li> <li>Amazon RDS</li> </ul> </li> </ul>	<p>Use us-east-2b as <b>Available Zone B</b> for:</p> <ul style="list-style-type: none"> <li>adx2RTB-B</li> <li>adx2PublicSubnetB (default subnet-123dc26f)</li> <li>adx2PrivateSubnetB <ul style="list-style-type: none"> <li>adx2WebserverB1</li> <li>adx2WebServerB2</li> </ul> </li> </ul>
--	--

#### 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 Actions ▼

Details [Info](#)

Route table ID rtb-09d9b3f40f48713e4	Main No	Explicit subnet associations –	Edge associations –
VPC <a href="#">vpc-3ee17855</a>   <a href="#">adx2VPC</a>	Owner ID 888742301715		

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

subnet-3280a77e / adx2PublicSubnetA Actions ▼

Details

Subnet ID subnet-3280a77e	Subnet ARN arn:aws:ec2:us-east-2:888742301715:subnet/subnet-3280a77e	State <span>Available</span>	IPv4 CIDR 172.31.32.0/20
Available IPv4 addresses 4091	IPv6 CIDR –	Availability Zone us-east-2c	Availability Zone ID use2-az3
VPC <a href="#">vpc-3ee17855</a>   <a href="#">adx2VPC</a>	Route table <a href="#">rtb-09d9b3f40f48713e4</a>   <a href="#">adx2RTB-A</a>	Network ACL <a href="#">acl-ee753185</a>	Default subnet Yes
Auto-assign public IPv4 address Yes	Auto-assign IPv6 address No	Auto-assign customer-owned IPv4 address No	Customer-owned IPv4 pool –
Outpost ID –	IPv4 CIDR reservations –	IPv6 CIDR reservations –	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 Actions ▼

Details

Subnet ID subnet-0950bdfa76c5211b0	Subnet ARN arn:aws:ec2:us-east-2:888742301715:subnet/subnet-0950bdfa76c5211b0	State <span>Available</span>	IPv4 CIDR 172.31.48.0/20
Available IPv4 addresses 4091	IPv6 CIDR –	Availability Zone us-east-2c	Availability Zone ID use2-az3
VPC <a href="#">vpc-3ee17855</a>   <a href="#">adx2VPC</a>	Route table <a href="#">rtb-09d9b3f40f48713e4</a>   <a href="#">adx2RTB-A</a>	Network ACL <a href="#">acl-ee753185</a>	Default subnet No
Auto-assign public IPv4 address No	Auto-assign IPv6 address No	Auto-assign customer-owned IPv4 address No	Customer-owned IPv4 pool –
Outpost ID –	IPv4 CIDR reservations –	IPv6 CIDR reservations –	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 Actions ▼

**Details** [Info](#)

Route table ID rtb-0ddd16c6624c75bb5	Main No	Explicit subnet associations –	Edge associations –
VPC vpc-3ee17855   adx2VPC	Owner ID 888742301715		

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

subnet-123dc26f / adx2PublicSubnetB Actions ▼

**Details**

Subnet ID subnet-123dc26f	Subnet ARN arn:aws:ec2:us-east-2:888742301715:subnet/subnet-123dc26f	State Available	IPv4 CIDR 172.31.16.0/20
Available IPv4 addresses 4091	IPv6 CIDR –	Availability Zone us-east-2b	Availability Zone ID use2-az2
VPC vpc-3ee17855   adx2VPC	Route table rtb-0ddd16c6624c75bb5   adx2RTB-B	Network ACL acl-ee753185	Default subnet Yes
Auto-assign public IPv4 address Yes	Auto-assign IPv6 address No	Auto-assign customer-owned IPv4 address No	Customer-owned IPv4 pool –
Outpost ID –	IPv4 CIDR reservations –	IPv6 CIDR reservations –	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 Actions ▼

**Details**

Subnet ID subnet-0b82062873d2d657a	Subnet ARN arn:aws:ec2:us-east-2:888742301715:subnet/subnet-0b82062873d2d657a	State Available	IPv4 CIDR 172.31.64.0/20
Available IPv4 addresses 4091	IPv6 CIDR –	Availability Zone us-east-2b	Availability Zone ID use2-az2
VPC vpc-3ee17855   adx2VPC	Route table rtb-0ddd16c6624c75bb5   adx2RTB-B	Network ACL acl-ee753185	Default subnet No
Auto-assign public IPv4 address No	Auto-assign IPv6 address No	Auto-assign customer-owned IPv4 address No	Customer-owned IPv4 pool –
Outpost ID –	IPv4 CIDR reservations –	IPv6 CIDR reservations –	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 Actions ▾

**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** | Outbound rules | Tags

**Inbound rules (2)** Refresh Manage tags Edit inbound rules

Filter security group rules

<input type="checkbox"/>	Name ▾	Security group rule... ▾	IP version ▾	Type ▾	Protocol ▾	Port range ▾	Source ▾	Description
<input type="checkbox"/>	-	sgr-0d635bcae0d9ed3c6	IPv4	HTTPS	TCP	443	0.0.0.0/0	-
<input type="checkbox"/>	-	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 LoadBalancerSG sg-0ffd1d8175cd02b11
    - HTTPS Protocol TCP Port 443 Source LoadBalancerSG 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 Actions ▾

**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** | Outbound rules | Tags

**Inbound rules (3)** Refresh Manage tags Edit inbound rules

Filter security group rules

<input type="checkbox"/>	Name ▾	Security group rule ID ▾	IP versi... ▾	Type ▲	Protocol ▾	Port range ▾	Source ▾
<input type="checkbox"/>	-	sgr-069b4fdb1af6110d8	-	HTTP	TCP	80	sg-0ffd1d8175cd02b11 / LoadBalancerSG
<input type="checkbox"/>	-	sgr-0226e1ddce9360667	-	HTTPS	TCP	443	sg-0ffd1d8175cd02b11 / LoadBalancerSG
<input type="checkbox"/>	-	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 Actions ▾

**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** | Outbound rules | Tags

**Inbound rules (5)** Manage tags Edit inbound rules

Filter security group rules

<input type="checkbox"/>	Name ▾	Security group rule... ▾	IP version ▾	Type ▲	Protocol ▾	Port ran... ▾	Source ▾
<input type="checkbox"/>	-	sgr-0b5e4c3221bc145ea	-	HTTP	TCP	80	sg-0ea017af27e99f2c6 / WebServerSG
<input type="checkbox"/>	-	sgr-0fa0f4d597f85f0a5	-	HTTPS	TCP	443	sg-0ea017af27e99f2c6 / WebServerSG
<input type="checkbox"/>	-	sgr-0d6891bf7ca9946dd	IPv4	MySQL/Aurora	TCP	3306	172.31.0.0/20
<input type="checkbox"/>	-	sgr-0d9d15ce1e2efaf6c	-	MySQL/Aurora	TCP	3306	sg-0ea017af27e99f2c6 / WebServerSG
<input type="checkbox"/>	-	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	<a href="#">subnet-0b82062873d2d657a</a>	172.31.64.0/20
us-east-2c	<a href="#">subnet-0950bdfa76c5211b0</a>	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 Modify Actions ▾

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

Connectivity & security | Monitoring | Logs & events | Configuration | Maintenance & backups | Tags

**Connectivity & security**

Endpoint & port	Networking	Security
Endpoint webappdatabase.cfz03euto6rv.us-east-2.rds.amazonaws.com	Availability Zone us-east-2c	VPC security groups <a href="#">DatabaseSG (sg-0f6f389ec1f87ad6a)</a> (active)
Port 3306	VPC <a href="#">adx2VPC (vpc-3ee17855)</a>	Publicly accessible No
	Subnet group adx2subnetgroup	Certificate authority rds-ca-2019
	Subnets <a href="#">subnet-0950bdfa76c5211b0</a> <a href="#">subnet-0b82062873d2d657a</a>	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.amazonaws.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)

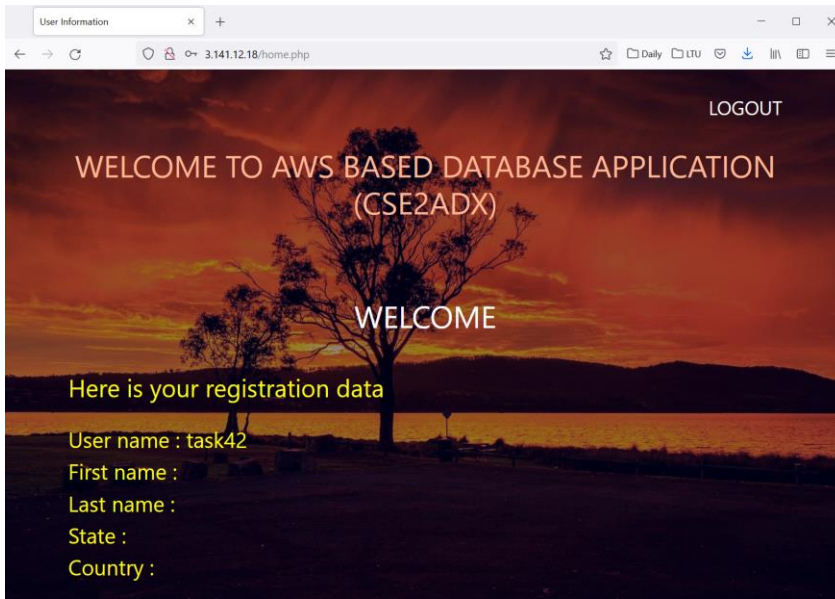
MariaDB [user accounts]>
```

- 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 CONNECT".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'



```
[root@ip-172-31-1-226 html]# 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 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]>
```

### 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

< 1 > ⚙️

<input checked="" type="checkbox"/>	Name ▾	AMI ID ▾	Instance type ▾	Spo... ▾	Creation time ▾
<input checked="" type="checkbox"/>	Web-Server-Launch-Configuration	ami-0b8dc0e10ae329b1e	t2.micro	-	Sat Sep 18 2021 23:24:12 GMT+1...

---

**Details** Copy launch configuration

AMI ID ami-0b8dc0e10ae329b1e	Instance type t2.micro	IAM instance profile -
Kernel ID -	Key name adx2KeyTask43	Monitoring false
EBS optimized false	Security groups <a href="#">sg-0ea017af27e99f2c6</a> <span>🔗</span>	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

EC2 > Auto Scaling groups > Web-Server-ASG

Details | Activity | Automatic scaling | Instance management | Monitoring | Instance refresh

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

Edit

Launch configuration

Launch configuration

Web-Server-Launch-Configuration

Instance type

t2.micro

Storage (volumes)

/dev/xvda

View details in the launch configuration console

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)

Edit

Purchase options and instance types

Settings not available unless you use a launch template.

Edit

Network

Availability Zones

us-east-2b, us-east-2c

Subnet ID

subnet-0b82062873d2d657a, subnet-0950bdfa76c5211b0

Edit

Activity history (2)

Q Filter activity history

< 1 > ⚙

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

⌂

Actions

Create dynamic scaling policy

< 1 >

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

Description Listeners Monitoring Integrated services Tags

### 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 <a href="#">Edit IP address type</a>
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  <a href="#">Edit subnets</a>
Hosted zone	Z3AADJGX6KTTL2
Creation time	September 19, 2021 at 11:24:28 AM UTC+10

---

Description **Listeners** Monitoring Integrated services Tags

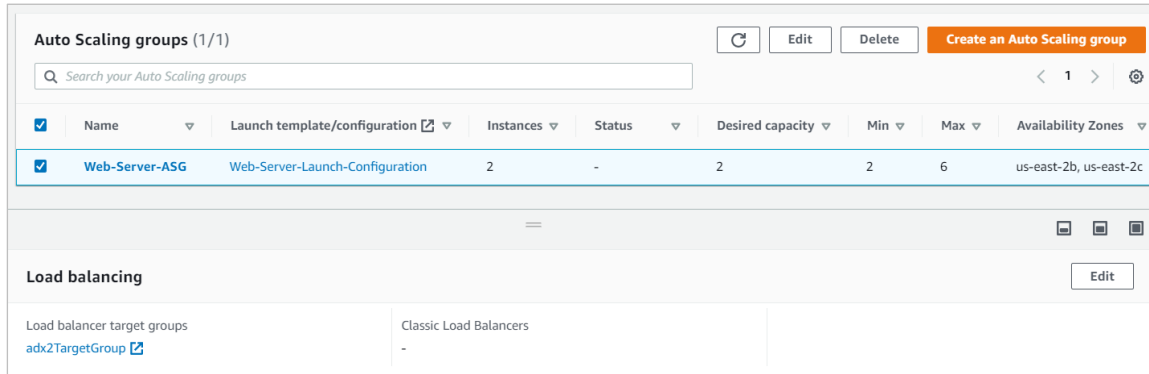
Listeners listen for connection requests using their protocol and port. You can add, remove, or update listeners and listener rules.

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

[Add listener](#) [Edit](#) [Delete](#)

<input type="checkbox"/>	Listener ID	Security policy	SSL Certificate	Rules
<input type="checkbox"/>	HTTP : 80 arn...c3f0d9741917c963-	N/A	N/A	Default forwarding to <a href="#">adx2TargetGroup</a> <a href="#">View/edit rules</a>

- 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



The screenshot shows the AWS Management Console interface for 'Auto Scaling groups (1/1)'. The 'Web-Server-ASG' group is selected, and the 'Load balancing' section is expanded, showing the 'adx2TargetGroup' target group.

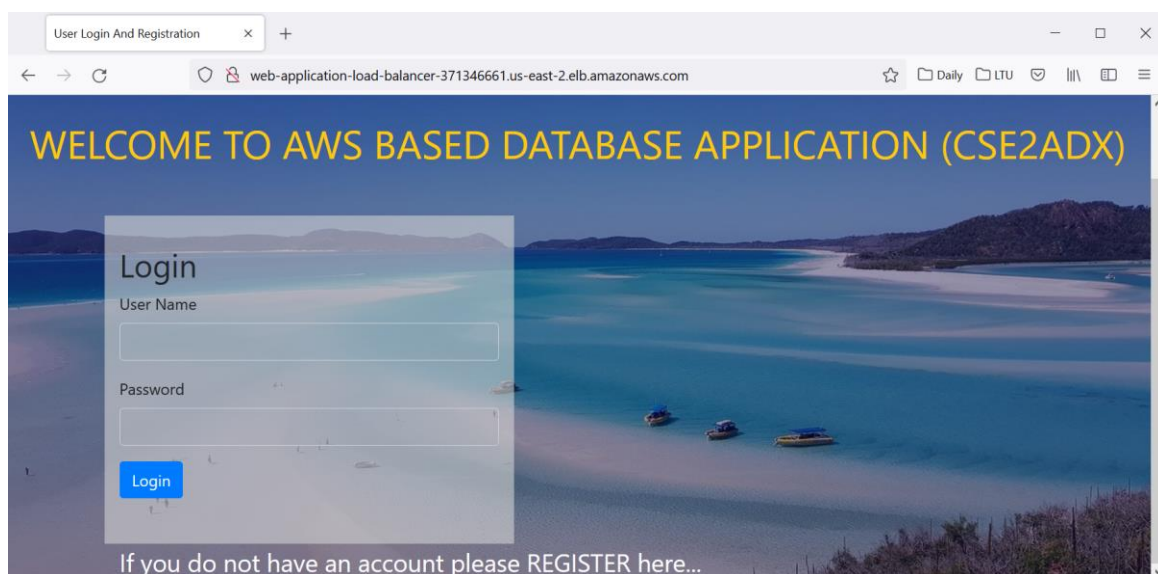
Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Availability Zones
Web-Server-ASG	Web-Server-Launch-Configuration	2	-	2	2	6	us-east-2b, us-east-2c

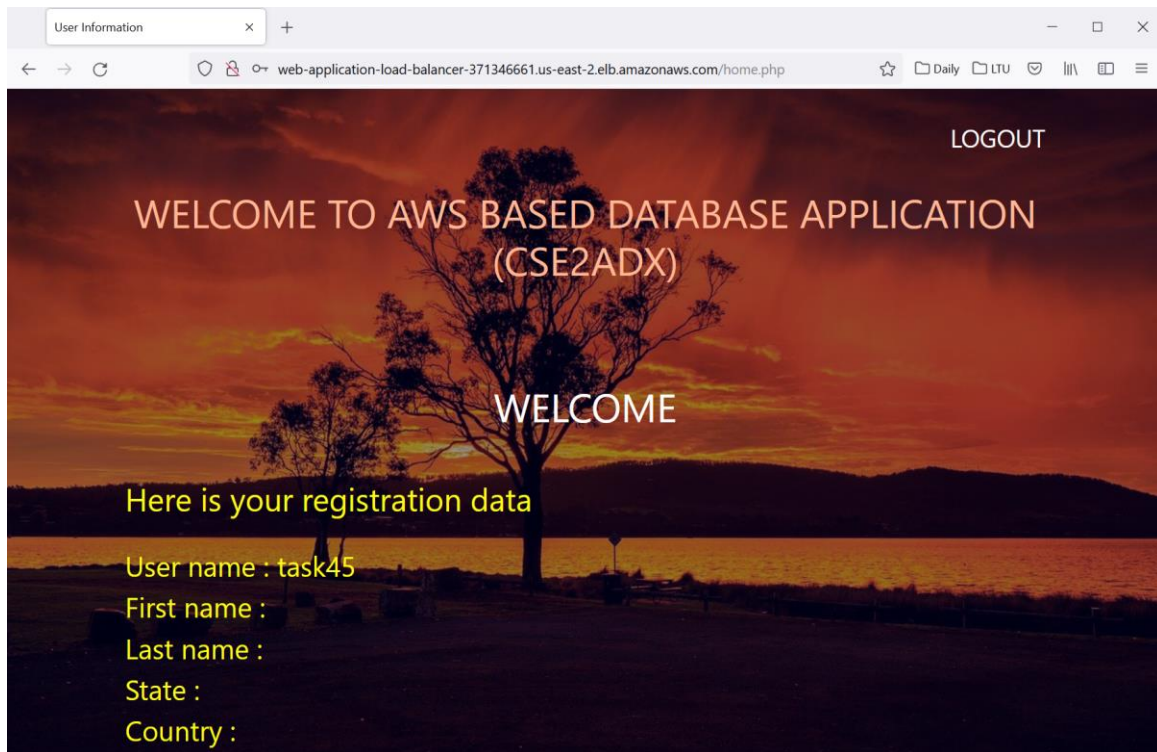
**Load balancing**

Load balancer target groups: adx2TargetGroup

Classic Load Balancers: -

- 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





```
ec2-user@ip-172-31-1-226:~  
[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]>
```