Project 7
Design and Implement Highly Available Web Application

https://catalog.us-east-1.prod.workshops.aws/workshops/3de93ad5-ebbe-4258-b977-b45cdfe661f1/en-US

INTRODUCTION

In this workshop, we will explore how to configure Amazon VPC, Amazon RDS, Amazon EC2 and Amazon EFS to build a highly available, auto-scaling multi-tier web application.

We chose eu-west-1 (Ireland) as our region.

1. Amazon VPC was created across multiple availability zones in your region of choice.
2. You will then deploy a highly available relational database across those availability zones using Amazon RDS.
3. Application tier was created across two availability zone.
4. Using Amazon EFS you will create shared file system spanning multiple availability zones.
5. Load-balanced group of web servers that will automatically scale in response to load to complete your application tier.
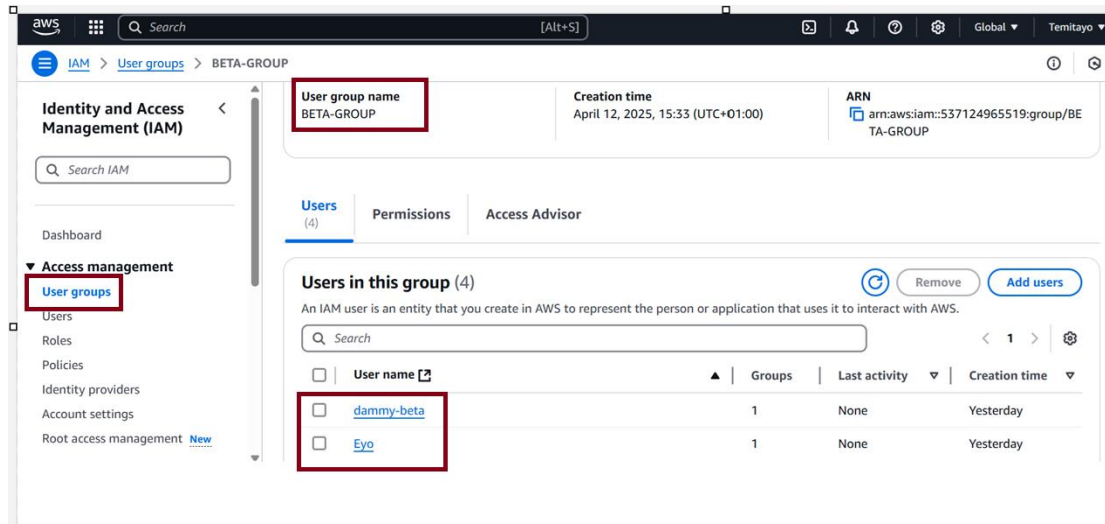

TOPOLOGY
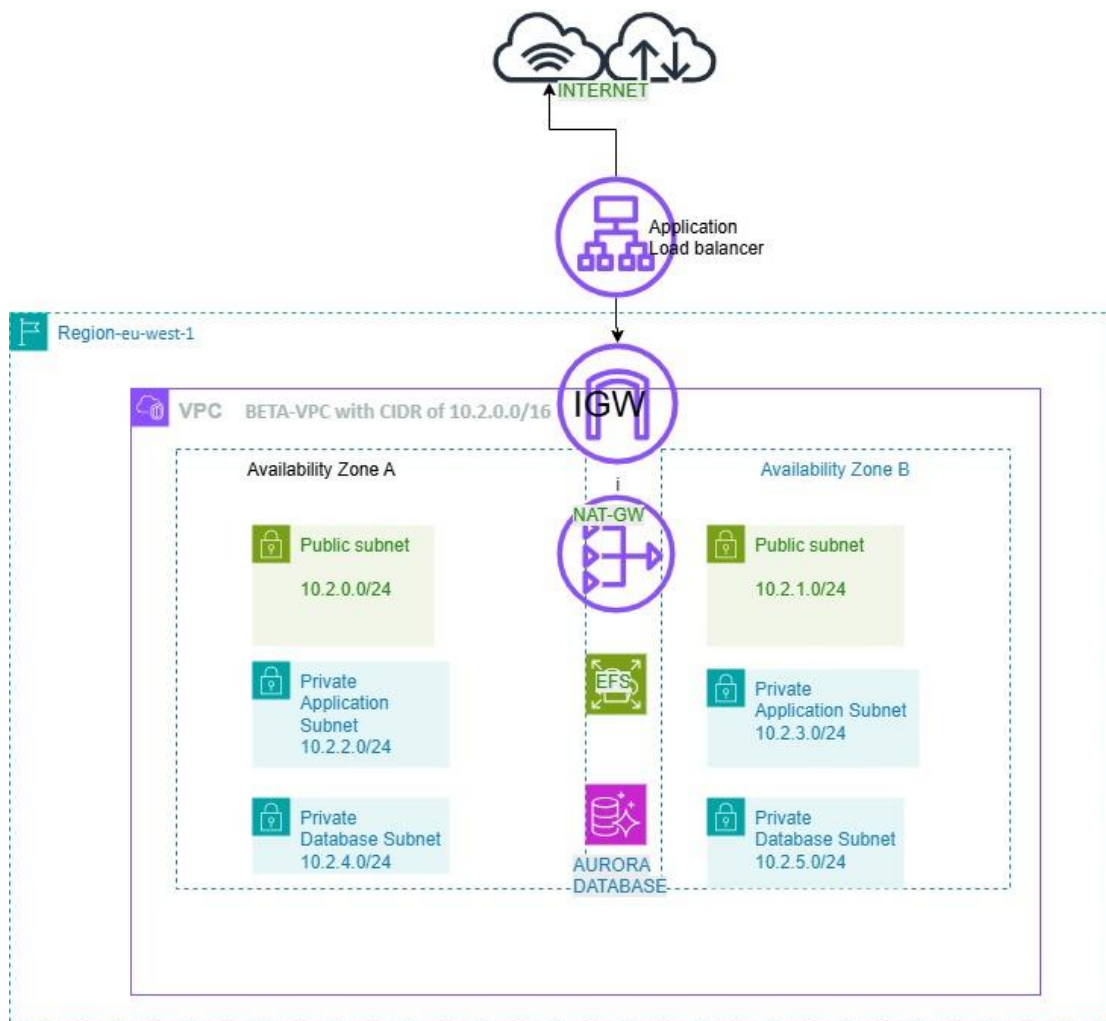
STEP 1

IAM user account was created attached to administrator group with privilege to access AWS services

BETA group was created which we attached all users account created to this group

LAB 1 (CONFIGUARTION OF NETWORK)
LAB 1.1

VIRTUAL PRIVATE CLOUD (VPC) was created with a name BETA-VPC with CIDR of 10.2.0.0/16

Click on **Actions** and then select **Edit VPC Settings**. Enable **DNS resolution** and **DNS hostnames** under **DNS Settings**

LAB 1.2

CREATE PUBLIC AND PRIVATE SUBNETS

Create public and private subnets. Six subnets will be created with three for each Avaibility Zone (AZ). Public subnet will have Internet Gateway, load balancers and NAT gateways. Application Subnet will contain application servers and your shared EFS filesystem. Database subnets has relational database. It will be accessible to other resources in the VPC but will have no access to the Internet and cannot be addressed by the Internet or the load balancers.

Name	IPv4 CIDR
Public Subnet A	10.2.0.0/24
Public Subnet B	10.2.1.0/24
Application Subnet A	10.2.2.0/24
Application Subnet B	10.2.3.0/24
Data Subnet A	10.2.4.0/24
Data Subnet B	10.2.5.0/24

LAB 1.3.

CREATE INTERNET GATEWAY

Create Internet gateway name BETA-IGW and attach it to BETA-VPC

LAB 1.4.
CREATE ROUTE TABLE FOR PUBLIC SUBNET.

The default route in public subnet is pointed to internet gateway while the two public subnets created are associated with the public route

LAB 1.5.

CREATE A NAT GATEWAY IN EACH PUBLIC SUBNET

LAB 1.6.

APPLICATION ROUTE

Create application route and use NAT gateway as default gateway.

rtb-0b886f74da48d058c / BETA-APP-NAT-ROUTE-A   Actions ▼

Details Info

Route table ID	Main	Explicit subnet associations	Edge associations
▢ rtb-0b886f74da48d058c	▢ No	subnet-0633d6706977d0c73 / Application Subnet A	–
VPC	**Owner ID**		
vpc-06ee73f02460db061 | BETA-VPC	▢ 537124965519		

Routes | Subnet associations | Edge associations | Route propagation | Tags

Routes (2)   Both ▼   Edit routes

Q Filter routes   ‹ 1 › ⚙

Destination ▽	Target ▽	Status ▽	Propagated ▽
0.0.0.0/0	nat-0f5e3a43f445e3cc4	⊘ Active	No
10.2.0.0/16	local	⊘ Active	No

rtb-0a8c6e87f598f5a3a / BETA-APP-NAT-ROUTE-B   Actions ▼

Details Info

Route table ID	Main	Explicit subnet associations	Edge associations
▢ rtb-0a8c6e87f598f5a3a	▢ No	subnet-01ef4265c5bee3361 / Application Subnet B	–
VPC	**Owner ID**		
vpc-06ee73f02460db061 | BETA-VPC	▢ 537124965519		

Routes | Subnet associations | Edge associations | Route propagation | Tags

Routes (2)   Both ▼   Edit routes

Q Filter routes   ‹ 1 › ⚙

Destination ▽	Target ▽	Status ▽	Propagated ▽
0.0.0.0/0	nat-0d06cf19c135418e1	⊘ Active	No
10.2.0.0/16	local	⊘ Active	No

LAB 1.7.

Route verification using Resource map

Resource map Info

VPC Show details	Subnets (6)	Route tables (4)	Network connecti
Your AWS virtual network	Subnets within this VPC	Route network traffic to resources	Connections to other netwo
BETA-VPC	**eu-west-1a**	BETA-APP-NAT-ROUTE-A	BETA-IGW
	Ⓐ Public Subnet A	rtb-020a709a40726906b	BETA-NATGW-A
	Ⓐ Data Subnet A	BETA-PUBLIC-ROUTE	BETA-NATGW-B
	Ⓐ Application Subnet A	BETA-APP-NAT-ROUTE-B	
	eu-west-1b		
	Ⓑ Public Subnet B		
	Ⓑ Data Subnet B		
	Ⓑ Application Subnet B		

LAB 2   SETTING UP RELATION DATABASE USING AURORA

LAB 2.1

Create two security groups for RDS. One for the clients named BETA DATABASE CLIENTS while the second one is called BETA DATABASE for the database itself.

SECURITY GROUP FOR RDS CLIENTS

sg-0cbb33f9282acc28c - BETA DATABASE CLIENTS   Actions ▼

Details

Security group name	Security group ID	Description	VPC ID
BETA DATABASE CLIENTS	sg-0cbb33f9282acc28c	BETA DATABASE CLIENTS	vpc-06ee73f02460db061 ↗

Owner	Inbound rules count	Outbound rules count	
537124965519	0 Permission entries	1 Permission entry	

Inbound rules | Outbound rules | Sharing - *new* | VPC associations - *new* | Tags

Inbound rules   ↻ Manage tags   Edit inbound rules

🔍 Search   ‹ 1 › ⚙

☐	Name ▽	Security group rule ID ▽	IP version ▽	Type ▽	Protocol ▽	Port range ▽
			No security group rules found			

SECURITY GROUP FOR RDS ALLOWING PORT 3306

sg-0c4178a9a98a53586 - BETA-DATABASE-1   Actions ▼

Details

Security group name	Security group ID	Description	VPC ID
BETA-DATABASE-1	sg-0c4178a9a98a53586	BETA-DATABASE-1	vpc-06ee73f02460db061 ↗

Owner	Inbound rules count	Outbound rules count	
537124965519	1 Permission entry	1 Permission entry	

Inbound rules | Outbound rules | Sharing - *new* | VPC associations - *new* | Tags

Inbound rules (1)   ↻ Manage tags   Edit inbound rules

🔍 Search   ‹ 1 › ⚙

Security group rule ID ▽	IP version ▽	Type ▽	Protocol ▽	Port ra... ▽	Source
sgr-0fbbe1ce6bc67dfe7	–	MYSQL/Aurora	TCP	3306	sg-0cbb33f9282acc28c / BETA DATABASE CLIENTS

LAB 2.2
CREATE RDS SUBNET GROUP
Click Amazon RDS console and create RDS subnet group

LAB 2.3
CREATE AURORA DATABASE INSTANCE

LAB 3   SETTING UP RELATION DATABASE USING AURORA

LAB 3.1   Create Elastic file system attached to BETA -VPC

Create file system

Create a file system with the recommended settings shown below by choosing Create file system. To view all settings or to customize your file system, choose Customize. **Learn more** [↗]

Name - *optional*
Name your file system.

> BETA-EFS

Name can include letters, numbers, and +-=._:/ symbols, up to 256 characters.

Virtual Private Cloud (VPC)
Choose the VPC where you want EC2 instances to connect to your file system.

> vpc-06ee73f02460db061
> BETA-VPC ▼

Recommended settings

Your file system is created with the following recommended settings unless you choose to customize the file system. You will be charged for storage and throughput. We recommend reviewing pricing for these features using the **AWS Pricing Calculator** [↗].

Setting	Value	Editable after creation
Throughput mode **Learn more** [↗]	Elastic	Yes
Transition into Infrequent Access (IA)	30 day(s) since last access	Yes

Cancel   Customize   **Create file system**

LAB 3.2   Click on customize

File system settings

General

Name - *optional*
Name your file system.

> BETA-EFS

File system type
Choose to either store data across multiple Availability Zones or within a single Availability Zone. **Learn more** [↗]

- ● **Regional**
  Offers the highest levels of availability and durability by storing file system data across multiple Availability Zones within an AWS Region.

- ○ **One Zone**
  Provides continuous availability to data within a single Availability Zone within an AWS Region.

Automatic backups
Automatically backup your file system data with AWS Backup using recommended settings. Additional pricing applies. **Learn more** [↗]

☐ Enable automatic backups

> ⚠ We recommend that you create a backup policy for your file system

Lifecycle management

Automatically save money as access patterns change by moving files into the Infrequent Access (IA) or Archive storage class. **Learn more** [↗]

Transition into Infrequent Access (IA)
Transition files to IA based on the time since they were last accessed in Standard storage.

| 30 day(s) since last access ▼ |

Transition into Archive
Transition files to Archive based on the time since they were last accessed in Standard storage.

| 90 day(s) since last access ▼ |

Transition into Standard
Transition files back to Standard storage based on when they are first accessed in IA or Archive storage.

| None ▼ |

Encryption

Choose to enable encryption of your file system's data at rest. Uses the AWS KMS service key (aws/elasticfilesystem) by default. **Learn more** [↗]

☑ Enable encryption of data at rest

▶ **Customize encryption settings**

Performance settings

Throughput mode
Choose a method for your file system's throughput limits. **Learn more** [↗]

◉ **Enhanced**
Provides more flexibility and higher throughput levels for workloads with a range of performance requirements.

○ **Bursting**
Provides throughput that scales with the amount of storage for workloads with basic performance requirements.

◉ **Elastic (Recommended)**
Use this mode for workloads with unpredictable I/O. With Elastic Throughput, performance automatically scales with your workload activity and you only pay for the throughput you use (data transferred for your file systems per month). **Learn more** [↗]

On the Network access page, under Mount targets, choose the two subnets created for the Data tier (Data subnet A and B). On the right side, under Security groups, associate the WP EFS security group created above to each mount target and remove the association with the Default security group.

Network access

Network

Virtual Private Cloud (VPC)   **Learn more** [↗]
Choose the VPC where you want EC2 instances to connect to your file system.

| vpc-06ee73f02460db061 | ▼ |
| BETA-VPC | |

Mount targets

A mount target provides an NFSv4 endpoint at which you can mount an Amazon EFS file system. We recommend creating one mount target per Availability Zone. **Learn more** [↗]

Availability zone	Subnet ID	IP address	Security groups	
eu-west-1a ▼	subnet-08f22a0509... ▼	Automatic	Choose security groups ▼	Remove
			sg-02b7e9151f3dd3d8e ✕ WP EFS	
eu-west-1b ▼	subnet-0c4853bd5af... ▼	Automatic	Choose security groups ▼	Remove
			sg-02b7e9151f3dd3d8e ✕ WP EFS	

Add mount target

Lab 4: Create Application load balancer

Create two security groups. One for the load balancer and the other for the webserver. Allow port 80 on the inbound rule of both security groups.

Lab 4.1 LOAD BALANCER SECUITY GROUP

EC2 > Security Groups > sg-08e563ce24985450b - WP LOAD BALANCER > Edit inbound rules

Edit inbound rules Info
Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info

Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info		Description - optional Info	
sgr-08aff799c1c72b864	HTTP ▼	TCP	80	My IP ▼	Q		Delete
					197.253.0.253/32 ✕		

Add rule

Cancel   Preview changes   Save rules

Lab 4.2 WP WEB SERVER SECUITY GROUP

EC2 > Security Groups > sg-099a2a95190e77371 - WP WEB SERVERS > Edit inbound rules

Edit inbound rules Info
Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info

Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info		Description - optional Info	
sgr-06e33a2700f340a1e	HTTP ▼	TCP	80	Custom ▼	Q		Delete
					sg-08e563ce24985450b ✕		

Add rule

Cancel   Preview changes   Save rules
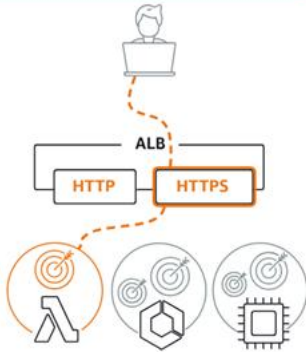
Lab 4.3 CREATE APPLICATION LOAD BALANCER

Create application load balancer named BETA-ALB
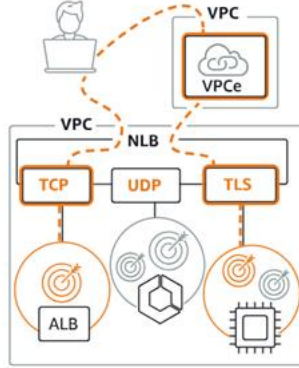
Compare and select load balancer type

A complete feature-by-feature comparison along with detailed highlights is also available. Learn more ⧉

Load balancer types

Application Load Balancer Info | **Network Load Balancer** Info | **Gateway Load Balancer** Info

Basic configuration

Load balancer name
Name must be unique within your AWS account and can't be changed after the load balancer is created.

> BETA-ALB

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Scheme | Info
Scheme can't be changed after the load balancer is created.

🔘 **Internet-facing**
- Serves internet-facing traffic.
- Has public IP addresses.
- DNS name is publicly resolvable.
- Requires a public subnet.

⚪ **Internal**
- Serves internal traffic.
- Has private IP addresses.
- DNS name is publicly resolvable.
- Compatible with the **IPv4** and **Dualstack** IP address types.

Load balancer IP address type | Info
Select the front-end IP address type to assign to the load balancer. The VPC and subnets mapped to this load balancer must include the selected IP address types. Public IPv4 addresses ha

🔘 **IPv4**
Includes only IPv4 addresses.

⚪ **Dualstack**
Includes IPv4 and IPv6 addresses.

⚪ **Dualstack without public IPv4**
Includes a public IPv6 address, and private IPv4 and IPv6 addresses. Compatible with **internet-facing** load balancers only.

Network mapping Info
The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

VPC | Info
The load balancer will exist and scale within the selected VPC. The selected VPC is also where the load balancer targets must be hosted unless routing to Lambda or on-premises targets, or if using VPC peering. To confirm the VPC for you
For a new VPC, create a VPC ⧉.

> BETA-VPC
> vpc-06ee73f02460db061
> IPv4 VPC CIDR: 10.2.0.0/16

IP pools - *new* | Info
You can optionally choose to configure an IPAM pool as the preferred source for your load balancers IP addresses. Create or view **Pools** in **Amazon VPC IP Address Manager console** ⧉.
☐ Use IPAM pool for public IPv4 addresses
   The IPAM pool you choose will be the preferred source of public IPv4 addresses. If the pool is depleted IPv4 addresses will be assigned by AWS.

Availability Zones and subnets | Info
Select at least two Availability Zones and a subnet for each zone. A load balancer node will be placed in each selected zone and will automatically scale in response to traffic. The load balancer routes traffic to targets in the selected Availa

☐ **eu-west-1a (euw1-az2)**

☐ **eu-west-1b (euw1-az3)**

Lab 5: Create a launch Template

Create a launch template named WP-Webservers-LT and attach security group for web server clients, database clients and EFS clients

Lab 5.2: Use Amazon Linux 2023

Lab 5.3 : Use Amazon Linux 2023

Attach clients SG for web,database and EFS .

Lab 5.4

Expand Advanced details and use the script below to populate the User Data field as text

Paste the script inside LAB5.3 or upload the script

```bash
#!/bin/bash

DB_NAME="database-1-beta"
DB_USERNAME="admin"
DB_PASSWORD="Beta12345"
DB_HOST="database-1-beta-instance-1.czs0uqeyypcc.eu-west-1.rds.amazonaws.com"
EFS_FS_ID="fs-067e64d854c3565b1"

dnf update -y

#install wget, apache server, php and efs utils
dnf install -y httpd wget php-fpm php-mysqli php-json php amazon-efs-utils

#create wp-content mountpoint
mkdir -p /var/www/html/wp-content
mount -t efs $EFS_FS_ID:/ /var/www/html/wp-content

#install wordpress
cd /var/www
wget https://wordpress.org/latest.tar.gz
tar -xzf latest.tar.gz
cp wordpress/wp-config-sample.php wordpress/wp-config.php
rm -f latest.tar.gz

#change wp-config with DB details
cp -rn wordpress/* /var/www/html/
sed -i "s/database_name_here/$DB_NAME/g" /var/www/html/wp-config.php
sed -i "s/username_here/$DB_USERNAME/g" /var/www/html/wp-config.php
sed -i "s/password_here/$DB_PASSWORD/g" /var/www/html/wp-config.php
sed -i "s/localhost/$DB_HOST/g" /var/www/html/wp-config.php

#change httpd.conf file to allowoverride
#  enable .htaccess files in Apache config using sed command
sed -i '/<Directory "\/var\/www\/html">/,/<\/Directory>/ s/AllowOverride None/AllowOverride All/'
/etc/httpd/conf/httpd.conf

# create phpinfo file
echo "<?php phpinfo(); ?>" > /var/www/html/phpinfo.php

# Recursively change OWNER of directory /var/www and all its contents
chown -R apache:apache /var/www

systemctl restart httpd
systemctl enable httpd
```

Lab 6: Create the app server

Create autoscalling group called   BETA-ASG

Choose launch template Info

Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group.

Name

Auto Scaling group name
Enter a name to identify the group.

```
BETA-ASG
```

Must be unique to this account in the current Region and no more than 255 characters.

Launch template Info

ⓘ For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.

Launch template
Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.

```
WP-Webservers-LT                                              ▼
```
Create a launch template ↗

Version
```
Default (2)                        ▼
```

```
Default (2)                        ▼
```
Create a launch template version ↗

Description	Launch template	Instance type
-	WP-Webservers-LT ↗ lt-0c82a8726fc0c8c10	t3.micro
AMI ID ami-01ff9fc7721895c6b	**Security groups** -	**Request Spot Instances** No
Key pair name -	**Security group IDs** sg-0cbb33f9282acc28c ↗ sg-099a2a95190e77371 ↗ sg-02d9aa36dc2019c85 ↗ sg-0c4178a9a98a53586 ↗	

Additional details

Storage (volumes)	Date created
-	Sat Apr 12 2025 14:26:05 GMT+0100 (West Africa Standard Time)

Cancel | Next

Lab 6.1 From **Network plane** ,attach application subnet A and subnet B

Network

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.

Availability Zones and subnets
Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.

```
Select Availability Zones and subnets                         ▼
```

eu-west-1a | subnet-0633d6706977d0c73 (Application Subnet A) ✕
10.2.2.0/24

eu-west-1b | subnet-01ef4265c5bee3361 (Application Subnet B) ✕
10.2.3.0/24

Create a subnet ↗

Availability Zone distribution - *new*
Auto Scaling automatically balances instances across Availability Zones. If launch failures occur in a zone, select a strategy.

● **Balanced best effort**
If launches fail in one Availability Zone, Auto Scaling will attempt to launch in another healthy Availability Zone.

○ **Balanced only**
If launches fail in one Availability Zone, Auto Scaling will continue to attempt to launch in the unhealthy Availability Zone to preserve balanced distribution.

Cancel | Update

Lab 6.2

For Grouping size and scaling Policy .
Desired capacity is 2
Minimum Capacity is 2

Maximum desired capacity 2

Group size

Specify the size of the Auto Scaling group by changing the desired capacity. You can also specify minimum and maximum scaling limits.

Desired capacity type

Choose the unit of measurement for the desired capacity value. vCPUs and Memory(GiB) are only supported for mixed instances groups configured with a set of instance attributes.

Units (number of instances) ▼

Desired capacity
Specify your group size.

2

Scaling limits
Set limits on how much your desired capacity can be increased or decreased.

Min desired capacity

2

Equal or less than desired capacity

Max desired capacity

4

Equal or greater than desired capacity

Cancel   **Update**

Lab 6.3
Registed targets are healthy in your target group

2 Total targets	⊘ 2 Healthy	⊗ 0 Unhealthy	⊖ 0 Unused	⊘ 0 Initial	⊖ 0 Draining
	0 Anomalous				

▶ **Distribution of targets by Availability Zone (AZ)**
  Select values in this table to see corresponding filters applied to the Registered targets table below.

Targets   Monitoring   Health checks   Attributes   Tags

Registered targets (2) Info   ⓘ Anomaly mitigation: **Not applicable** ↻ Deregister   Register targets

Target groups route requests to individual registered targets using the protocol and port number specified. Health checks are performed on all registered targets according to the target group's health check settings. Anomaly detection is automatically applied to HTTP/HTTPS target groups with at least 3 healthy targets.

Q Filter targets   < 1 > ⚙

☐	Instance ID ▽	Name ▽	Port ▽	Zone ▽	Health status ▽	Health status details	Admini... ▽	Overri... ▽
☐	i-0cfe43be259716a06	APP-SERVER-A	80	eu-west-1a (eu...	⊘ Healthy	-	⊖ No override.	No overri...
☐	i-04bbbdadb3476b22f	APP-SERVER-B	80	eu-west-1b (eu...	⊘ Healthy	-	⊖ No override.	No overri...

Lab 6.3
Open the DNS name (http://beta-alb-1247648413.eu-west-1.elb.amazonaws.com/) for your Application Load Balancer in your web browser to view your newly created Wordpress installation.

← → C ⚠ Not secure beta-alb-1247648413.eu-west-1.elb.amazonaws.com

⊞ | 🔴 Audio Messages – E... G Google ⅢⅢ Ubiquiti Airmax Po... ▶ (81) N380.flv - YouT... ⌁ Free Penetration Tes... ⬛ Overview in Network Autor

504 Gateway Time-out

```
C:\Users\Temitope>ping beta-alb-1247648413.eu-west-1.elb.amazonaws.com

Pinging beta-alb-1247648413.eu-west-1.elb.amazonaws.com [54.77.83.0] with 32 bytes of data:
Reply from 54.77.83.0: bytes=32 time=173ms TTL=245
Reply from 54.77.83.0: bytes=32 time=125ms TTL=245
Reply from 54.77.83.0: bytes=32 time=125ms TTL=245
Reply from 54.77.83.0: bytes=32 time=126ms TTL=245

Ping statistics for 54.77.83.0:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 125ms, Maximum = 173ms, Average = 137ms
```

CHALLENGES

The challenges we encountered include :

1. We mistakenly attached a wrong VPC to one of the security Groups created. Which was later corrected .

⊗ **Auto Scaling group could not be created**
One or more security groups in the launch template are not linked to the VPCs configured in the Auto Scaling group

🔍 Diagnose with Amazon Q

2. The second issue is inability to access the web application due to server error as shown below:

← → C ⚠ Not secure beta-alb-1247648413.eu-west-1.elb.amazonaws.com

⊞ | ☁ Audio Messages – E... G Google ɪᴇ Ubiquiti Airmax Po... ▶ (81) N380.flv - YouT... ℂ Free Penetration Tes... ⊞ Overview 🅸🅽 Network Auton

504 Gateway Time-out

3. We do not know how to explore AWS Database