

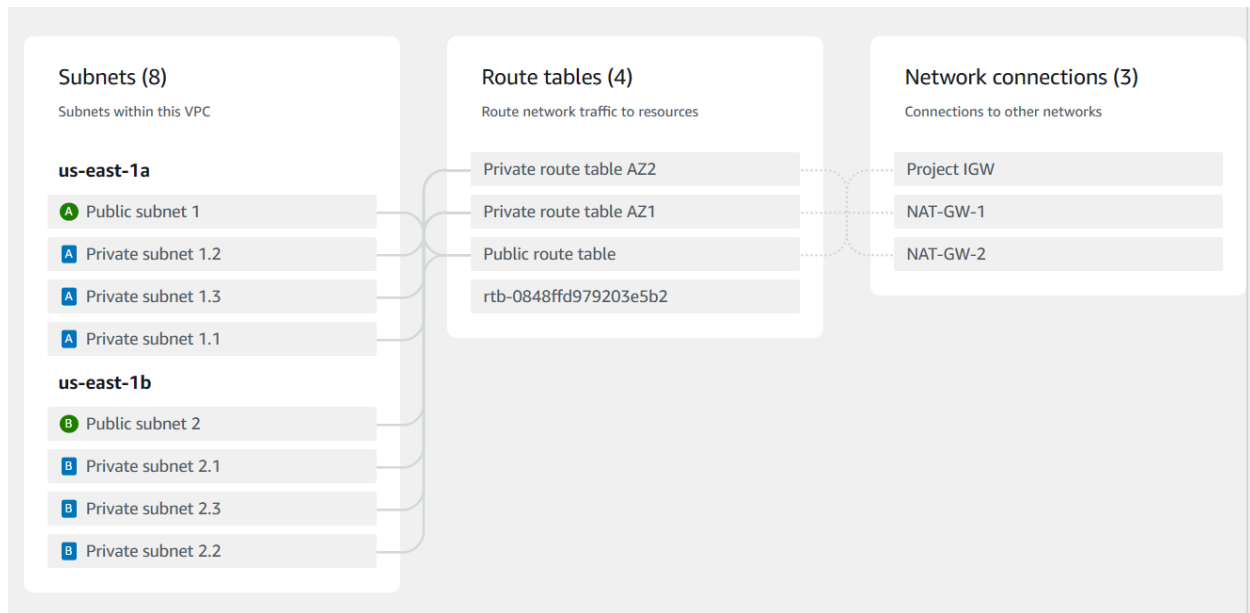
Cloud 3-tier architecture project

Adam Zouari

- 1.VPC
- 2.Security groups
- 3.RDS
- 4.App Server
- 5.Web Server

VPC

I started by creating a VPC named "project" in the us-east-1 region with two availability zones (AZ1 being us-east-1a and AZ2 being us-east-1b), each containing one public subnet and three private subnets.



First, I created an Internet Gateway and attached it to the VPC.

Internet gateways (1/2) Info					Actions	Create internet gateway
<input type="text" value="Search"/>					View details	
					Attach to VPC	
					Detach from VPC	
					Manage tags	
					Delete internet gateway	
Name	Internet gateway ID	State	VPC	Owner		
-	igw-0c0bba9ed7ed58340	Attached	vpc-	161967204		
Project IGW	igw-0c7d14a4a8751747a	Attached	vpc-	161967204		

Next, I created two NAT Gateways for each availability zone.

Name	NAT gateway ID	Connectivity...	State	State message	Primary public I...	Primar
NAT-GW-1	nat-06d3d2d4da28d73ab	Public	Available	-	34.202.2.70	10.0.0.
NAT-GW-2	nat-0cf19f076d822d775	Public	Available	-	54.80.217.228	10.0.0.

Then, I created the routing tables:

- A public routing table for the public subnets.
- A private routing table for AZ1 in us-east-1a and a private routing table for AZ2 in us-east-1b.

Route tables (5) Info				
<input type="text" value="Find resources by attribute or tag"/>				
<input type="checkbox"/>	Name ▾	Route table ID ▾	Explicit subnet associ... ▾	Edge a
<input type="checkbox"/>	Private route table AZ2	rtb-040bc930023b00458	3 subnets	–
<input type="checkbox"/>	–	rtb-01f8bd384fbc03401	–	–
<input type="checkbox"/>	Private route table AZ1	rtb-07b6614c512cbf57c	3 subnets	–
<input type="checkbox"/>	Public route table	rtb-0d2a8971dc43705ba	2 subnets	–
<input type="checkbox"/>	–	rtb-0848ffd979203e5b2	–	–

For Public route table :

Routes (2)

Filter routes

Both

Edit routes

<

1

>

Destination	Target	Status	Propagated
0.0.0.0/0	igw-0c7d14a4a8751747a	Active	No
10.0.0.0/24	local	Active	No

Explicit subnet associations (2)

Find subnet association

Edit subnet associations

<

1

>

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
Public subnet 2	subnet-0d214e5f682e11f5b	10.0.0.64/28	–
Public subnet 1	subnet-0739a64e869e9c0a4	10.0.0.0/28	–

For Private route table AZ1 :

Routes (2)

Filter routes

<1>

Destination

▼

Target

▼

Status

▼

Propagated

▼

0.0.0.0/0

[nat-06d3d2d4da28d73ab](#)

✓

Active

No

10.0.0.0/24

local

✓

Active

No

Explicit subnet associations (3)

Edit subnet associations

Find subnet association

<1>

Name

▼

Subnet ID

▼

IPv4 CIDR

▼

IPv6 CIDR

▼

Private subnet 1.2

[subnet-09605f89fc027a971](#)

10.0.0.32/28

–

Private subnet 1.3

[subnet-0ca7ee071d6ae281f](#)

10.0.0.48/28

–

Private subnet 1.1

[subnet-030876a311056ad87](#)

10.0.0.16/28

–

For the private routing table in AZ2:

It is configured similarly to the private routing table in AZ1, except it is attached to its own NAT Gateway and the private subnets created earlier.

After completing the network configuration, I proceeded to the next step.

Security groups

First, I created a security group for the database.

sg-0466e0ca84b2e988e - DBSG

Actions ▼

Details

Security group name
DBSG

Security group ID
sg-0466e0ca84b2e988e

Description
SG for DB

VPC ID
[vpc-06acb7200c15b487e](#)

Owner
161967204562

Inbound rules count
2 Permission entries

Outbound rules count
1 Permission entry

Inbound rules

Outbound rules

Tags

Inbound rules (2)



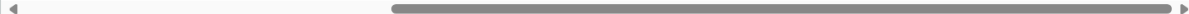
Manage tags

Edit inbound rules

Search

< 1 > ⚙

▼	Type	▼	Protocol	▼	Port range	▼	Source	▼	Description	▼
	MySQL/Aurora		TCP		3306		sg-0b944567fd6f4c48...		-	
	MySQL/Aurora		TCP		3306		sg-0fff5e03758ec758...		-	



sg-0466e0ca84b2e988e - DBSG

Actions ▼

Details

Security group name
DBSG

Security group ID
sg-0466e0ca84b2e988e

Description
SG for DB

VPC ID
[vpc-06acb7200c15b487e](#)

Owner
161967204562

Inbound rules count
2 Permission entries

Outbound rules count
1 Permission entry

Inbound rules

Outbound rules

Tags

Outbound rules (1)



Manage tags

Edit outbound rules

Search

< 1 > ⚙

▼	Type	▼	Protocol	▼	Port range	▼	Destination	▼	Description	▼
	All traffic		All		All		0.0.0.0/0		-	



Next, I created a security group for the bastion host.

sg-0b944567fd6f4c483 - Bastion SG

Actions

Details

Security group name

Bastion SG

Security group ID

sg-0b944567fd6f4c483

Description

Allows SSH access to developers.

VPC ID

[vpc-06acb7200c15b487e](#)

Owner

161967204562

Inbound rules count

1 Permission entry

Outbound rules count

1 Permission entry

Inbound rules

Outbound rules

Tags

Inbound rules (1)

Manage tags

Edit inbound rules

Search

< 1 >

Type	Protocol	Port range	Source	Description
SSH	TCP	22	0.0.0.0/0	-

sg-0b944567fd6f4c483 - Bastion SG

Actions

Details

Security group name

Bastion SG

Security group ID

sg-0b944567fd6f4c483

Description

Allows SSH access to developers.

VPC ID

[vpc-06acb7200c15b487e](#)

Owner

161967204562

Inbound rules count

1 Permission entry

Outbound rules count

1 Permission entry

Inbound rules

Outbound rules

Tags

Outbound rules (1)

Manage tags

Edit outbound rules

Search

< 1 >

Name	Security group rule...	IP version	Type	Protocol	Port range
-	sgr-005f06e4fc5851cc3	IPv4	All traffic	All	All

Next, I created a security group for the internet-facing load balancer.

sg-03405d0f3d35878e1 - Public LB SG

Actions

Details

Security group name

Public LB SG

Security group ID

sg-03405d0f3d35878e1

Description

Allow HTTP access for users

VPC ID

[vpc-06acb7200c15b487e](#)

Owner

161967204562

Inbound rules count

1 Permission entry

Outbound rules count

1 Permission entry

Inbound rules

Outbound rules

Tags

Outbound rules (1)

Manage tags

Edit outbound rules

Search

< 1 > ⚙

	Name	Security group rule...	IP version	Type	Protocol	Port range
<input type="checkbox"/>	-	sgr-05583bde554382...	-	HTTP	TCP	80

sg-03405d0f3d35878e1 - Public LB SG

Actions

Details

Security group name

Public LB SG

Security group ID

sg-03405d0f3d35878e1

Description

Allow HTTP access for users

VPC ID

[vpc-06acb7200c15b487e](#)

Owner

161967204562

Inbound rules count

1 Permission entry

Outbound rules count

1 Permission entry

Inbound rules

Outbound rules

Tags

Outbound rules (1)

Manage tags

Edit outbound rules

Search

< 1 > ⚙

	Type	Protocol	Port range	Destination	Description
<input type="checkbox"/>	HTTP	TCP	80	sg-02fe17ef32c150fe...	-

Next, I created a security group for the web servers.

sg-02fe17ef32c150fe5 - Web servers SG

Actions ▾

Details

Security group name

Web servers SG

Security group ID

sg-02fe17ef32c150fe5

Description

SG for web servers

VPC ID

[vpc-06acb7200c15b487e](#)

Owner

161967204562

Inbound rules count

2 Permission entries

Outbound rules count

1 Permission entry

Inbound rules

Outbound rules

Tags

Inbound rules (2)

↻

Manage tags

Edit inbound rules

Q Search

< 1 > ⚙

▼	Type	▼	Protocol	▼	Port range	▼	Source	▼	Description	▼
	SSH		TCP		22		sg-0b944567fd6f4c48...		–	
	HTTP		TCP		80		sg-03405d0f3d35878...		–	

sg-02fe17ef32c150fe5 - Web servers SG

Actions ▾

Details

Security group name

Web servers SG

Security group ID

sg-02fe17ef32c150fe5

Description

SG for web servers

VPC ID

[vpc-06acb7200c15b487e](#)

Owner

161967204562

Inbound rules count

2 Permission entries

Outbound rules count

1 Permission entry

Inbound rules

Outbound rules

Tags

Outbound rules (1)

↻

Manage tags

Edit outbound rules

Q Search

< 1 > ⚙

▼	Type	▼	Protocol	▼	Port range	▼	Destination	▼	Description	▼
	All traffic		All		All		0.0.0.0/0		–	

Next, I created a security group for the internal load balancer.

sg-003b234e3a18db32f - Private LB SG

Actions

Details

Security group name

Private LB SG

Security group ID

sg-003b234e3a18db32f

Description

SG for private LB

VPC ID

vpc-06acb7200c15b487e

Owner

161967204562

Inbound rules count

1 Permission entry

Outbound rules count

1 Permission entry

Inbound rules

Outbound rules

Tags

Inbound rules (1)

Manage tags

Edit inbound rules

Search

< 1 >

Type	Protocol	Port range	Source	Description
HTTP	TCP	80	sg-02fe17ef32c150fe...	-

sg-003b234e3a18db32f - Private LB SG

Actions

Details

Security group name

Private LB SG

Security group ID

sg-003b234e3a18db32f

Description

SG for private LB

VPC ID

vpc-06acb7200c15b487e

Owner

161967204562

Inbound rules count

1 Permission entry

Outbound rules count

1 Permission entry

Inbound rules

Outbound rules

Tags

Outbound rules (1)

Manage tags

Edit outbound rules

Search

< 1 >

Type	Protocol	Port range	Destination	Description
Custom TCP	TCP	3000	sg-0fff5e03758ec758...	-

Then, I created one for the application servers.

sg-0fff5e03758ec7589 - App servers SG

Actions

Details

Security group name

App servers SG

Security group ID

sg-0fff5e03758ec7589

Description

SG for App servers

VPC ID

[vpc-06acb7200c15b487e](#)

Owner

161967204562

Inbound rules count

2 Permission entries

Outbound rules count

1 Permission entry

Inbound rules

Outbound rules

Tags

Inbound rules (2)

Manage tags

Edit inbound rules

Search

< 1 >

Type	Protocol	Port range	Source	Description
Custom TCP	TCP	3000	sg-003b234e3a18db3...	-
SSH	TCP	22	sg-0b944567fd6f4c48...	-

sg-0fff5e03758ec7589 - App servers SG

Actions

Details

Security group name

App servers SG

Security group ID

sg-0fff5e03758ec7589

Description

SG for App servers

VPC ID

[vpc-06acb7200c15b487e](#)

Owner

161967204562

Inbound rules count

2 Permission entries

Outbound rules count

1 Permission entry

Inbound rules

Outbound rules

Tags

Outbound rules (1)

Manage tags

Edit outbound rules

Search

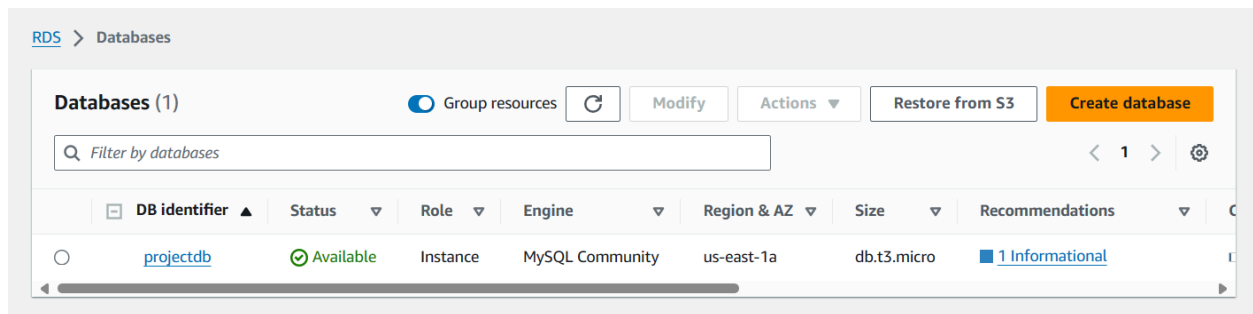
< 1 >

Type	Protocol	Port range	Destination	Description
All traffic	All	All	0.0.0.0/0	-

Then, I moved on to the next step.

RDS

I created a MySQL database instance `db.t3.micro` with 20 GB of storage named `projectdb` in AZ1, with a standby instance in AZ2.



Availability and durability

Deployment options [Info](#)

The deployment options below are limited to those supported by the engine you selected above.

- ☐ **Multi-AZ DB Cluster**
Creates a DB cluster with a primary DB instance and two readable standby DB instances, with each DB instance in a different Availability Zone (AZ). Provides high availability, data redundancy and increases capacity to serve read workloads.
- ☒ **Multi-AZ DB instance**
Creates a primary DB instance and a standby DB instance in a different AZ. Provides high availability and data redundancy, but the standby DB instance doesn't support connections for read workloads.
- ☐ **Single DB instance**
Creates a single DB instance with no standby DB instances.

DB instance class [Info](#)

▼ Hide filters

- ☒ Show instance classes that support Amazon RDS Optimized Writes [Info](#)
Amazon RDS Optimized Writes improves write throughput by up to 2x at no additional cost.
- ☐ Include previous generation classes
- ☐ Standard classes (includes m classes)
- ☐ Memory optimized classes (includes r and x classes)
- ☒ Burstable classes (includes t classes)

db.t3.micro

2 vCPUs 1 GiB RAM Network: 2,085 Mbps

Storage

Storage type [Info](#)

Provisioned IOPS SSD (io2) storage volumes are now available.

General Purpose SSD (gp3)

Performance scales independently from storage

Allocated storage [Info](#)

20

GiB

Minimum: 20 GiB. Maximum: 6,144 GiB

Amazon RDS

Dashboard

Databases

Query Editor

Performance insights

Snapshots

Exports in Amazon S3

Automated backups

Reserved instances

Proxies

Subnet groups

Parameter groups

Option groups

Custom engine versions

Zero-ETL integrations [New](#)

Events

Event subscriptions

Summary

DB identifier
projectdb

CPU

3.92%

Status

Available

Class
db.t3.micro

Role
Instance

Current activity

1 Connections

Engine
MySQL Community

Region & AZ
us-east-1a

Recommendations

1 Informational

Connectivity & security

Monitoring

Logs & events

Configuration

Zero-ETL integrations

Maintenance & backups

Tag

Connectivity & security

Endpoint & port

Endpoint

projectdb.cryby2rgpj6y.us-east-1.rds.amazonaws.com

Port
3306

Networking

Availability Zone
us-east-1a

VPC
[Project \(vpc-06acb7200c15b487e\)](#)

Subnet group
project db

Subnets
[subnet-0ca7ee071d6ae281f](#)
[subnet-002d5153d5c2630ba](#)

Security

VPC security groups
[DBSG \(sg-0466e0ca84b2e988e\)](#)

Active

Publicly accessible
No

Certificate authority [Info](#)
rds-ca-rsa2048-g1

Certificate authority date
May 26, 2061, 00:34 (UTC+01:00)

Then, I proceeded to the next step.

App Servers

I created three instances:

- An EC2 Bastion host.
- An EC2 web server.
- An EC2 application server.

I generated an SSH key pair during the creation of the EC2 Bastion host and used the same key pair (Host.pem) for creating the other instances.

Instance summary for i-055fb20d7dfd97942 (Bastion Host) [Info](#)

Connect

Instance state ▼

Actions ▼

Updated less than a minute ago

Instance ID i-055fb20d7dfd97942 (Bastion Host)	Public IPv4 address 34.226.77.122 (Bastion Host) open address	Private IPv4 addresses 10.0.0.14
IPv6 address -	Instance state Running	Public IPv4 DNS -
Hostname type IP name: ip-10-0-0-14.ec2.internal	Private IP DNS name (IPv4 only) ip-10-0-0-14.ec2.internal	
Answer private resource DNS name -	Instance type t2.micro	Elastic IP addresses 34.226.77.122 (Bastion Host) [Public IP]
Auto-assigned IP address -	VPC ID vpc-06acb7200c15b487e (Project) Link	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. Learn more
IAM Role -	Subnet ID subnet-0739a64e869e9c0a4 (Public subnet 1) Link	Auto Scaling Group name -
IMDSv2 Required		

Instance summary for i-0775249dc03980490 (App Server) [Info](#)

↻

Connect

Instance state ▼

Actions ▼

Updated less than a minute ago

Instance ID i-0775249dc03980490 (App Server)	Public IPv4 address -	Private IPv4 addresses 10.0.0.40
IPv6 address -	Instance state Running	Public IPv4 DNS -
Hostname type IP name: ip-10-0-0-40.ec2.internal	Private IP DNS name (IPv4 only) ip-10-0-0-40.ec2.internal	
Answer private resource DNS name -	Instance type t2.micro	Elastic IP addresses -
Auto-assigned IP address -	VPC ID vpc-06acb7200c15b487e (Project)	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. Learn more
IAM Role -	Subnet ID subnet-09605f89fc027a971 (Private subnet 1.2)	Auto Scaling Group name -
IMDSv2 Required		

Instance summary for i-0cfa519260757343b (Web Server) [Info](#)

↻

Connect

Instance state ▼

Actions ▼

Updated less than a minute ago

Instance ID i-0cfa519260757343b (Web Server)	Public IPv4 address -	Private IPv4 addresses 10.0.0.23
IPv6 address -	Instance state Running	Public IPv4 DNS -
Hostname type IP name: ip-10-0-0-23.ec2.internal	Private IP DNS name (IPv4 only) ip-10-0-0-23.ec2.internal	
Answer private resource DNS name -	Instance type t2.micro	Elastic IP addresses -
Auto-assigned IP address -	VPC ID vpc-06acb7200c15b487e (Project)	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. Learn more
IAM Role -	Subnet ID subnet-030876a311056ad87 (Private subnet 1.1)	Auto Scaling Group name -
IMDSv2 Required		

Elastic IP addresses (3)			
<input type="text" value="Find resources by attribute or tag"/>			
<input type="checkbox"/>	Name ▾	Allocated IPv4 addr... ▾	Type
<input type="checkbox"/>	NAT-GW-1	34.202.2.70	Public IP
<input type="checkbox"/>	Bastion Host	34.226.77.122	Public IP
<input type="checkbox"/>	NAT-GW-2	54.80.217.228	Public IP

```
aws | Services Search [Alt+S]
```

```
#_
~\_####_ Amazon Linux 2023
~~\_#####\
~~ \###|
~~ \#/_____ https://aws.amazon.com/linux/amazon-linux-2023
~~ v~' '->
~~~
~~~.-.-.-/-/
~/m/'
```

Last login: Fri Apr 12 16:47:56 2024 from 18.206.107.28
[ec2-user@ip-10-0-0-14 ~]\$ ssh -i Host.pem ec2-user@10.0.0.40

```
#_
~\_####_ Amazon Linux 2023
~~\_#####\
~~ \###|
~~ \#/_____ https://aws.amazon.com/linux/amazon-linux-2023
~~ v~' '->
~~~
~~~.-.-.-/-/
~/m/'
```

Last login: Fri Apr 12 17:15:21 2024 from 10.0.0.14
[ec2-user@ip-10-0-0-40 ~]\$

First, I downloaded MySQL and connected to the database using these commands:

```
-install mysql
```

```
-sudo yum install
```

```
https://dev.mysql.com/get/mysql80-community-release-el9-5.noarch.rpm
```

```
-sudo yum install mysql-community-server
```

```
-sudo systemctl enable --now mysqld
```

```
mysql -h projectdb.cryby2rgpj6y.us-east-1.rds.amazonaws.com
```

```
-P 3306 -u ProjectDB -p ProjectDB
```



```
[ec2-user@ip-10-0-0-40 ~]$ mysql -h projectdb.cryby2rgpj6y.us-east-1.rds.amazonaws.com -P 3306 -u ProjectDB -p ProjectDB
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2347
Server version: 8.0.35 Source distribution

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

```
mysql> use DrOrlowDB;
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
mysql> show tables;
+-----+
| Tables_in_DrOrlowDB |
+-----+
| users                |
+-----+
1 row in set (0.00 sec)

mysql> select * from users;
```

identifier	name	surname	gender	password	email	pemail	birthday
Adem	Adem	Zouari	Male	Adem	ademzouari55@gmail.com	ademzouari55@gmail.com	2003-01-05
fares	fares	DKHILI	Male	lafriis	dkhili.fares2002@gmail.com	dkhili.fares2002@gmail.com	2002-12-21
يحيى	rebhi	mohamed	Male	يحيى	adam15.mohamed17@gmail.com	adam15.mohamed17@gmail.com	2024-04-10

```
3 rows in set (0.00 sec)

mysql>
```

Then, I executed these commands to prepare the server.

```
curl -o-  
https://raw.githubusercontent.com/nvm-sh/nvm/v0.38.0/install.sh |  
bash  
  
source ~/.bashrc  
  
nvm install 16  
  
nvm use 16  
  
npm install -g pm2  
  
sudo yum install git  
  
git clone https://github.com/Adem-Zouari/DrOrlow.git app-tier  
  
cd app-tier  
  
rm -r public  
  
cd server  
  
npm install mysql express body-parser  
  
pm2 start app.js  
  
pm2 save
```

I created a public GitHub repository containing the frontend and backend files, then downloaded the files from this repository.

```
[ec2-user@ip-10-0-0-40 ~]$ pm2 list
```

id	name	namespace	version	mode	pid	uptime	U	status	cpu	mem	user	watching
0	app	default	N/A	fork	2116	2h	0	online	0%	52.8mb	ec2-user	disabled

```
[ec2-user@ip-10-0-0-40 ~]$ pm2 logs
```

```
[TAILING] Tailing last 15 lines for [all] processes (change the value with --lines option)
```

```
/home/ec2-user/.pm2/pm2.log last 15 lines:
```

```
PM2 | 2024-04-12T15:22:39: PM2 log: PM2 version : 5.3.1
PM2 | 2024-04-12T15:22:39: PM2 log: Node.js version : 16.20.2
PM2 | 2024-04-12T15:22:39: PM2 log: Current arch : x64
PM2 | 2024-04-12T15:22:39: PM2 log: PM2 home : /home/ec2-user/.pm2
PM2 | 2024-04-12T15:22:39: PM2 log: PM2 PID file : /home/ec2-user/.pm2/pm2.pid
PM2 | 2024-04-12T15:22:39: PM2 log: RPC socket file : /home/ec2-user/.pm2/rpc.sock
PM2 | 2024-04-12T15:22:39: PM2 log: BUS socket file : /home/ec2-user/.pm2/pub.sock
PM2 | 2024-04-12T15:22:39: PM2 log: Application log path : /home/ec2-user/.pm2/logs
PM2 | 2024-04-12T15:22:39: PM2 log: Worker Interval : 30000
PM2 | 2024-04-12T15:22:39: PM2 log: Process dump file : /home/ec2-user/.pm2/dump.pm2
PM2 | 2024-04-12T15:22:39: PM2 log: Concurrent actions : 2
PM2 | 2024-04-12T15:22:39: PM2 log: SIGTERM timeout : 1600
PM2 | 2024-04-12T15:22:39: PM2 log: =====
PM2 | 2024-04-12T15:22:39: PM2 log: App [app:0] starting in -fork mode-
PM2 | 2024-04-12T15:22:40: PM2 log: App [app:0] online
```

```
/home/ec2-user/.pm2/logs/app-error.log last 15 lines:
```

```
/home/ec2-user/.pm2/logs/app-out.log last 15 lines:
```

```
0|app | Server is running on port 3000
```

For the next step, I created an image of the application server.

Instances (1/7) Info

Find Instance by attribute or tag (case-sensitive)

All states

	Name	Instance ID	Instance state	Instance type	Status check
<input type="checkbox"/>	Bastion Host	i-055fb20d7dfd97942	Running	t2.micro	2/2 checks passed
<input type="checkbox"/>		i-0b74eef7561fb5d66	Running	t2.micro	2/2 checks passed
<input type="checkbox"/>	Web Server	i-0cfa519260757343b	Running	t2.micro	2/2 checks passed
<input checked="" type="checkbox"/>	App Server	i-0775249dc03980490	Running	t2.micro	2/2 checks passed
<input type="checkbox"/>		i-053d724de7bda7eaf	Running	t2.micro	2/2 checks passed
<input type="checkbox"/>		i-027a3785311545223	Running	t2.micro	2/2 checks passed

Connect

Instance state

Actions

Launch instances

Connect

View details

Manage instance state

Instance settings

Networking

Security

Image and templates

Monitor and troubleshoot

View alarms

us-east-1a

Create image

Create template from instance

Launch more like this

Image summary for ami-03412703dfd5ae656

EC2 Image Builder

Actions

Launch instance from AMI

AMI ID	Image type	Platform details	Root device type
ami-03412703dfd5ae656	machine	Linux/UNIX	EBS
AMI name	Owner account ID	Architecture	Usage operation
WebImage	161967204562	x86_64	RunInstances
Root device name	Status	Source	Virtualization type
/dev/xvda	Available	161967204562/WebImage	hvm
Boot mode	State reason	Creation date	Kernel ID
uefi-preferred	-	Fri Apr 12 2024 16:30:09 GMT+0100 (Central European Standard Time)	-
Description	Product codes	RAM disk ID	Deprecation time
Image for Web Servers	-	-	-
Last launched time	Block devices		
-	/dev/xvda=snap-0f38dc97212c909c7:8:true:gp3		

Then, the AppTargetGroup.

EC2 > Target groups > AppTargetGroup

AppTargetGroup

Actions

Details

arn:aws:elasticloadbalancing:us-east-1:161967204562:targetgroup/AppTargetGroup/ca77bdf08de7671

Target type	Protocol : Port	Protocol version	VPC
Instance	HTTP: 3000	HTTP1	vpc-06acb7200c15b487e
IP address type	Load balancer		
IPv4	App-LB		

2	2 Healthy	0 Unhealthy	0 Unused	0 Initial	0 Draining
Total targets	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.

Then, the internal load balancer.

Load balancer type	Status	VPC	IP address type
Application	Active	vpc-06acb7200c15b487e	IPv4
Scheme	Hosted zone	Availability Zones	Date created
Internal	Z35SXD0TRQ7X7K	subnet-09605f89fc027a971 us-east-1a (use1-az6) subnet-0809410460407a934 us-east-1b (use1-az1)	April 12, 2024, 16:22 (UTC+01:00)

Load balancer ARN	DNS name
arn:aws:elasticloadbalancing:us-east-1:161967204562:loadbalancer/app/App-LB/274abb0e40e64798	internal-App-LB-612751870.us-east-1.elb.amazonaws.com (A Record)

Listeners and rules

Network mapping

Resource map - new

Security

Monitoring

Integrations

Attributes

Tags

Listeners and rules (1) Info

Manage rules

Manage listener

Add listener

A listener checks for connection requests on its configured protocol and port. Traffic received by the listener is routed according to the default action and any additional rules.

Filter listeners

< 1 >

	Protocol:Port	Default action	Rules	ARN	Security policy	De
<input type="checkbox"/>	HTTP:80	<div>Forward to target group<ul style="list-style-type: none">AppTargetGroup: 1 (100%)Group-level stickiness: Off</div>	1 rule	<input type="checkbox"/> ARN	Not applicable	No

Then, I created AppTemplate using the image.

Details

Versions

Template tags

Launch template version details

Actions ▼Delete template version

Version

1 (Default) ▼

Description

Template for App Servers

Date created

2024-04-12T15:24:33.000Z

Created by

arn:aws:sts::161967204562:assumed-role/voclabs/user3116876=ademzouari55@gmail.com

Instance details

Storage

Resource tags

Network interfaces

Advanced details

AMI ID

ami-096f1b477b4accf8d

Instance type

t2.micro

Availability Zone

-

Key pair name

Host

Security groups

-

Security group IDs

sg-0fff5e03758ec7589

And finally: App Auto Scaling Group

[EC2](#) > [Auto Scaling groups](#) > App-ASG

App-ASG

[Details](#) | [Activity](#) | [Automatic scaling](#) | [Instance management](#) | [Monitoring](#) | [Instance refresh](#)

Group details

Auto Scaling group name

App-ASG

Date created

Fri Apr 12 2024 16:26:39 GMT+0100
(Central European Standard Time)

Desired capacity

2

Minimum capacity

2

Maximum capacity

4

Desired capacity type

Units (number of instances)

Status

-

Amazon Resource Name (ARN)

 arn:aws:autoscaling:us-east-1:161967204562:autoScalingGroup:e812734f-0f8c-4b7d-9f0e-31385c14c84e:autoScalingGroupName/App-ASG

Launch template

Launch template

 lt-0b06fbb65441bb46b
AppTemplate

AMI ID

 ami-096f1b477b4accf8d

Instance type

t2.micro

Owner

arn:aws:sts::161967204562:assumed-role/voclabs/user3116876=ademzouari55@gmail.com

Network

Availability Zones

us-east-1a, us-east-1b

Subnet ID

subnet-09605f89fc027a971, subnet-0809410460407a934

Instance type requirements

Your Auto Scaling group adheres to the launch template for purchase option and instance type.

Load balancing

Load balancer target groups

[AppTargetGroup](#)

Classic Load Balancers

-

Then, I proceeded to the next step.

Web Server

I connected to the web server using the Bastion host, then entered these commands:

```
-curl -o-  
https://raw.githubusercontent.com/nvm-sh/nvm/v0.38.0/install.sh |  
bash  
  
-source ~/.bashrc  
  
-nvm install 16  
  
-nvm use 16  
  
-sudo yum install git  
  
-git clone https://github.com/Adem-Zouari/DrOrlow.git web-tier  
  
-cd web-tier  
  
-rm -r server  
  
-sudo yum install nginx  
  
-sudo truncate -s 0 /etc/nginx/nginx.conf  
  
-chmod -R 755 /home/ec2-user  
  
-sudo chkconfig nginx on
```

Then, I used the command `sudo nano /etc/nginx/nginx.conf` and added a health check and a submit endpoint :


```
#health check
location /health {
    default_type text/html;
    return 200 "<!DOCTYPE html><p>Web Tier Health
Check</p>\n";
}

#front end files
location / {
    root    /home/ec2-user/web-tier/public;
    index   Home.html "About me.html";
    try_files $uri /Home.html;
}

# New location block to handle submit endpoint
location /submit {
    proxy_pass
http://internal-App-LB-612751870.us-east-1.elb.amazonaws.com:
80;

    proxy_set_header Host $host;
```

```

        proxy_set_header X-Real-IP $remote_addr;

        proxy_set_header X-Forwarded-For
$proxy_add_x_forwarded_for;

        proxy_set_header X-Forwarded-Proto $scheme;
    }

}

```

```

GNU nano 5.8 /etc/nginx/nginx.conf
user nginx;
worker_processes auto;
error_log /var/log/nginx/error.log;
pid /run/nginx.pid;

# Load dynamic modules. See /usr/share/doc/nginx/README.dynamic.
include /usr/share/nginx/modules/*.conf;

events {
    worker_connections 1024;
}

http {
    log_format main '$remote_addr - $remote_user [$time_local] "$request" '
        '$status $body_bytes_sent "$http_referer" '
        '"$http_user_agent" "$http_x_forwarded_for"';

    access_log /var/log/nginx/access.log main;

    sendfile        on;
    tcp_nopush      on;
    tcp_nodelay      on;
    keepalive_timeout 65;
}

```

And then

-sudo systemctl start nginx

-sudo systemctl status nginx

```

Last login: Fri Apr 12 17:15:26 2024 from 10.0.0.14
[ec2-user@ip-10-0-0-23 ~]$ sudo systemctl status nginx
● nginx.service - The nginx HTTP and reverse proxy server
   Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: disabled)
   Active: active (running) since Fri 2024-04-12 15:32:08 UTC; 2h 44min ago
     Process: 2132 ExecStartPre=/usr/bin/rm -f /run/nginx.pid (code=exited, status=0/SUCCESS)
     Process: 2141 ExecStartPre=/usr/sbin/nginx -t (code=exited, status=0/SUCCESS)
     Process: 2158 ExecStart=/usr/sbin/nginx (code=exited, status=0/SUCCESS)
    Main PID: 2162 (nginx)
       Tasks: 2 (limit: 1114)
      Memory: 5.1M
         CPU: 60ms
    CGroup: /system.slice/nginx.service
            └─2162 "nginx: master process /usr/sbin/nginx"
              └─2163 "nginx: worker process"

Apr 12 15:32:07 ip-10-0-0-23.ec2.internal systemd[1]: Starting nginx.service - The nginx HTTP and reverse proxy server...
Apr 12 15:32:08 ip-10-0-0-23.ec2.internal nginx[2141]: nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
Apr 12 15:32:08 ip-10-0-0-23.ec2.internal nginx[2141]: nginx: configuration file /etc/nginx/nginx.conf test is successful
Apr 12 15:32:08 ip-10-0-0-23.ec2.internal systemd[1]: Started nginx.service - The nginx HTTP and reverse proxy server.
[ec2-user@ip-10-0-0-23 ~]$

```

And finally, I set up the Web Auto Scaling Group and internet-facing load balancer using the image and WebTemplate for the web servers.

Instances (1/7) Info

Find Instance by attribute or tag (case-sensitive)

All states

	Name	Instance ID	Instance state	Instance type	Status check
<input type="checkbox"/>		i-0b74eef7561fb5d66	Running	t2.micro	2/2 checks passed
<input checked="" type="checkbox"/>	Web Server	i-0cfa519260757343b	Running	t2.micro	2/2 checks passed
<input type="checkbox"/>	App Server	i-0775249dc03980490	Running	t2.micro	2/2 checks passed
<input type="checkbox"/>	Bastion Host	i-055fb20d7dfd97942	Running	t2.micro	2/2 checks passed
<input type="checkbox"/>		i-053d724de7bda7eaf	Running	t2.micro	2/2 checks passed
<input type="checkbox"/>		i-027a3785311545223	Running	t2.micro	2/2 checks passed

Create image

Create template from instance

Launch more like this

Connect

View details

Manage instance state

Instance settings

Networking

Security

Image and templates

Monitor and troubleshoot

View alarms

Launch instances

us-east-1a

Instance: i-0cfa519260757343b (Web Server)

EC2 > AMIs > ami-03412703dfd5ae656

Image summary for ami-03412703dfd5ae656

EC2 Image Builder

Actions

Launch instance from AMI

AMI ID	Image type	Platform details	Root device type
ami-03412703dfd5ae656	machine	Linux/UNIX	EBS
AMI name	Owner account ID	Architecture	Usage operation
WebImage	161967204562	x86_64	RunInstances
Root device name	Status	Source	Virtualization type
/dev/xvda	Available	161967204562/WebImage	hvm
Boot mode	State reason	Creation date	Kernel ID
uefi-preferred	-	Fri Apr 12 2024 16:30:09 GMT+0100 (Central European Standard Time)	-
Description	Product codes	RAM disk ID	Deprecation time
Image for Web Servers	-	-	-
Last launched time	Block devices		
-	/dev/xvda=snap-0f38dc97212c909c7:8:true:gp3		

Permissions

Storage

Tags

Image share permission

WebTargetGroup

Actions ▼

Details

arn:aws:elasticloadbalancing:us-east-1:161967204562:targetgroup/WebTargetGroup/94c7b62fa94d60e9

Target type Instance	Protocol : Port HTTP: 80	Protocol version HTTP1	VPC vpc-06acb7200c15b487e
IP address type IPv4	Load balancer Web-LB		

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

Load balancer type Application	Status Active	VPC vpc-06acb7200c15b487e	IP address type IPv4
Scheme Internet-facing	Hosted zone Z35SXDOTRQ7X7K	Availability Zones subnet-0d214e5f682e11f5b us-east-1b (use1-az1) subnet-0739a64e869e9c0a4 us-east-1a (use1-az6)	Date created April 12, 2024, 16:40 (UTC+01:00)

Load balancer ARN
arn:aws:elasticloadbalancing:us-east-1:161967204562:loadbalancer/app/Web-LB/827b563d99addcb2

DNS name [Info](#)
Web-LB-17362500.us-east-1.elb.amazonaws.com (A Record)

Listeners and rules | Network mapping | Resource map - new | Security | Monitoring | Integrations | Attributes | Tags

Listeners and rules (1) [Info](#)

Refresh

Manage rules ▼

Manage listener ▼

Add listener

A listener checks for connection requests on its configured protocol and port. Traffic received by the listener is routed according to the default action and any additional rules.

Filter listeners

< 1 > ⚙

<input type="checkbox"/>	Protocol:Port ▼	Default action ▼	Rules ▼	ARN ▼	Security policy ▼	De
<input type="checkbox"/>	HTTP:80	<div>Forward to target group<ul style="list-style-type: none">WebTargetGroup: 1 (100%)Group-level stickiness: Off</div>	1 rule	<div><div>ARN</div></div>	Not applicable	No

Details

Versions

Template tags

Launch template version details

Actions ▼Delete template version

Version

1 (Default) ▼

Description

Template for Web Servers

Date created

2024-04-12T15:42:03.000Z

Created by

arn:aws:sts::161967204562:assumed-role/voclabs/user3116876=ademzouari55@gmail.com

Instance details

Storage

Resource tags

Network interfaces

Advanced details

AMI ID

ami-03412703dfd5ae656

Instance type

t2.micro

Availability Zone

-

Key pair name

Host

Security groups

-

Security group IDs


sg-02fe17ef32c150fe5

Web-ASG

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




Group details

Edit

Auto Scaling group name Web-ASG	Desired capacity 2	Desired capacity type Units (number of instances)	Amazon Resource Name (ARN)  arn:aws:autoscaling:us-east-1:161967204562:autoScalingGroup:d5e63304-5d46-4b31-958f-d97ff81d0562:autoScalingGroupName/Web-ASG
Date created Fri Apr 12 2024 16:43:23 GMT+0100 (Central European Standard Time)	Minimum capacity 2	Status -	
	Maximum capacity 4		

Launch template

Edit

Launch template  lt-066d64258ea147532 WebTemplate	AMI ID  ami-03412703dfd5ae656	Instance type t2.micro	Owner arn:aws:sts::161967204562:assumed-role/voclabs/user3116876=ademzouari55@gmail.com
Latest	-	 sg-02fe17ef32c150fe5	Fri Apr 12 2024 16:42:03 GMT+0100 (Central European Standard Time)
Description Template for Web Servers	Storage (volumes) -	Key pair name Host	Request Spot Instances No

View details in the launch template console

Network

Edit

Availability Zones us-east-1a, us-east-1b	Subnet ID subnet-0e355d87d7ee7e91c, subnet-030876a311056ad87	
--	---	--

Instance type requirements

Edit

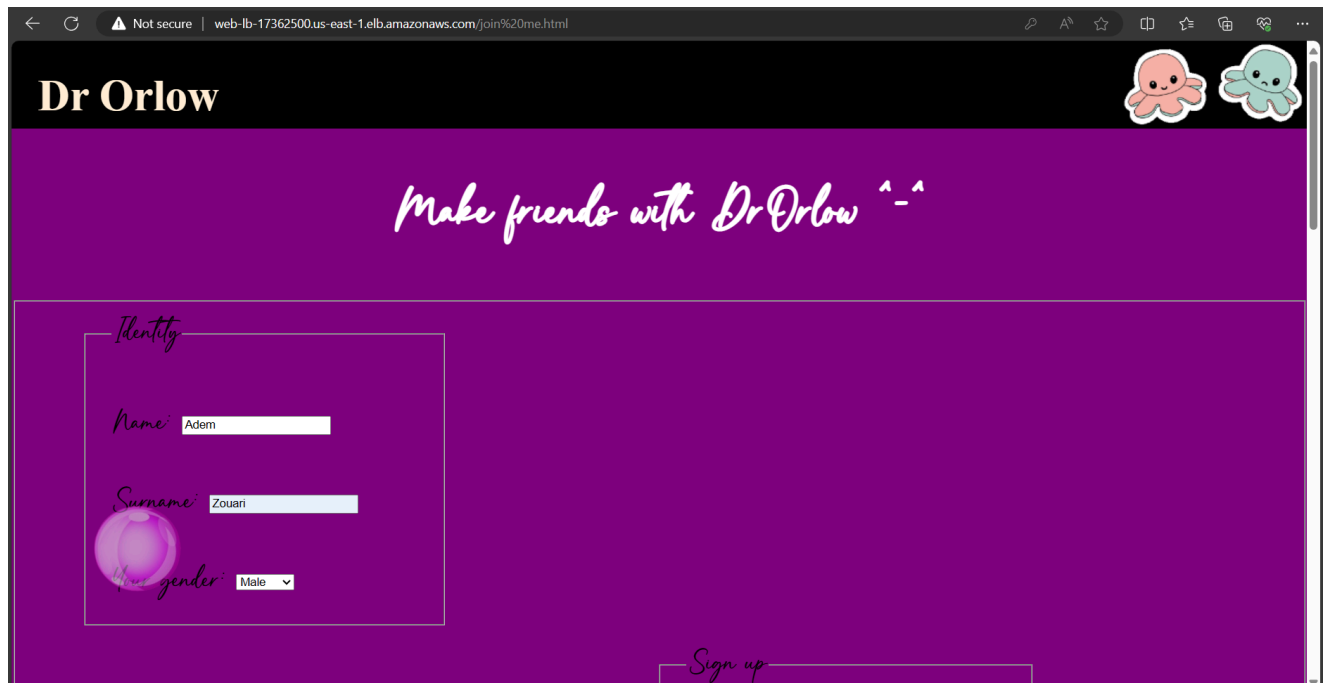
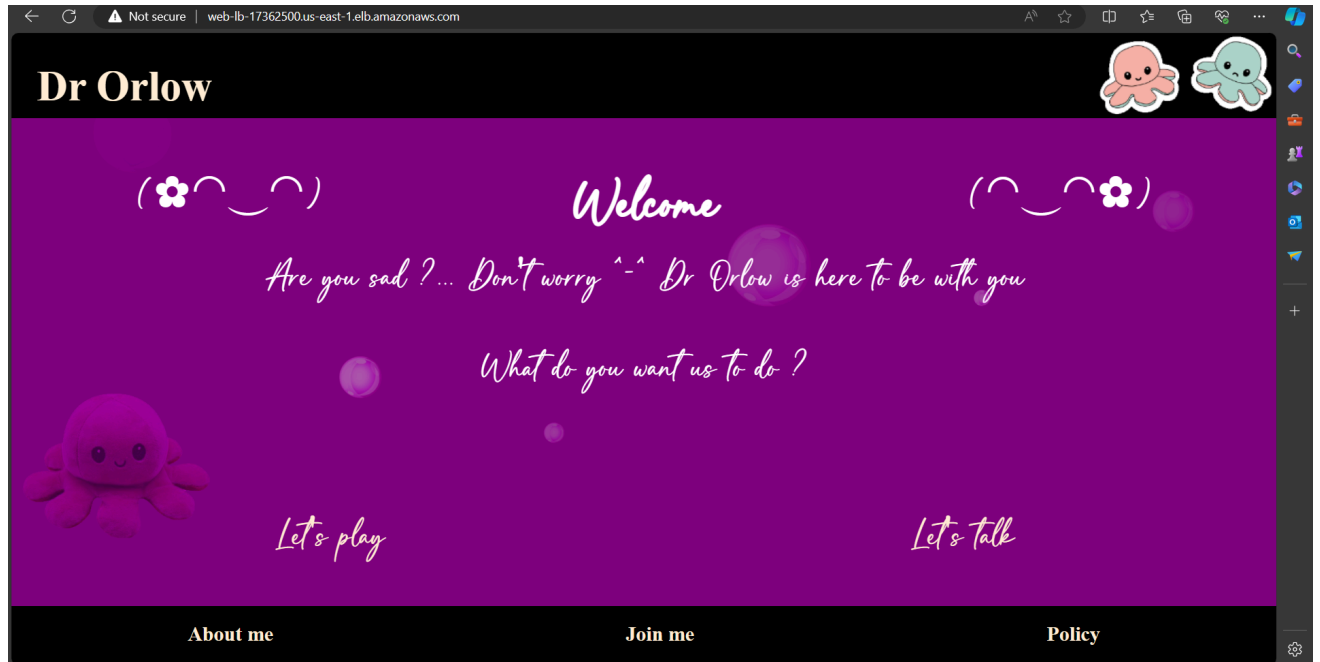
Your Auto Scaling group adheres to the launch template for purchase option and instance type.

Load balancing

Edit

Load balancer target groups WebTargetGroup	Classic Load Balancers -	
---	-----------------------------	--

Finally, here is the website.



←

↺

⚠ Not secure | web-lb-17362500.us-east-1.elb.amazonaws.com/join%20me.html

🔍

A

☆

📑

⌵

🔒

🌐

⋮

What do you think of Dr Orlov?

• He is cute

• He is so cute

☒ He is so so cute

Submit

Reset

Home

About me

Policy