Virtual Coffee Shop Launch

|		
☰ Author	Olatunji Olayinka Oluwaseun	
🔗 LinkedIn	https://www.linkedin.com/in/olatunji-olayinka-coder/	
☰ GitHub	https://github.com/olatunji-weber	
⚙ Status	Done	

Solution for a Virtual Coffee Shop Launch

On going working through the solution for a project that I was assigned, I decided to outline my journey through it here with the thought that it might help someone.

Problem:

You're the cloud engineer for a virtual coffee shop chain that wants to move their operations to the cloud. They need a scalable and cost-effective solution to ensure their website and backend are always up and running.

Solution:

In order to achieve the end result of this project, I created a VPC called "**coffee-shop-vpc**".
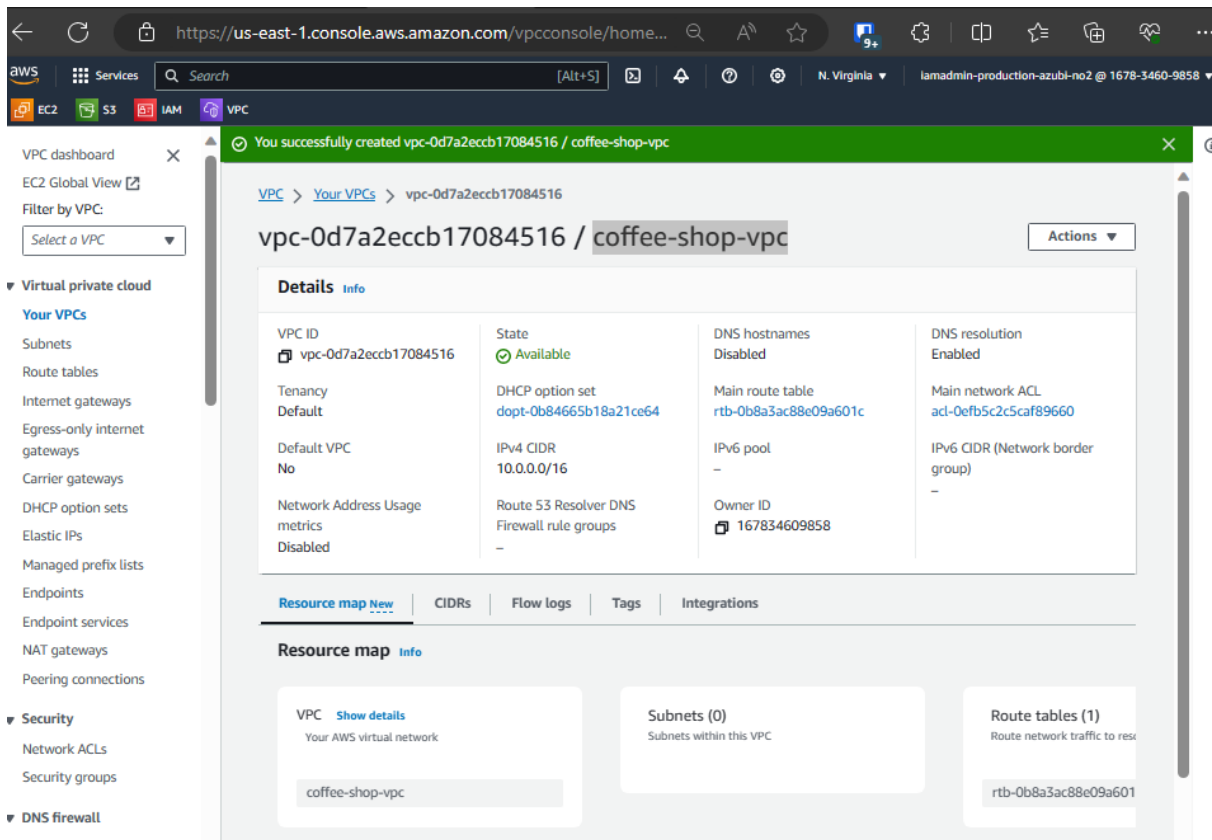
Then, created 2 subnets (one public and the other private) in 2 Availability zones, (named "coffee-shop-public" as the public subnet) and ("coffee-shop-private" as the private subnet), in "us-east-1a" and "us-east-1d" AZs respectively to ensure high availability always.
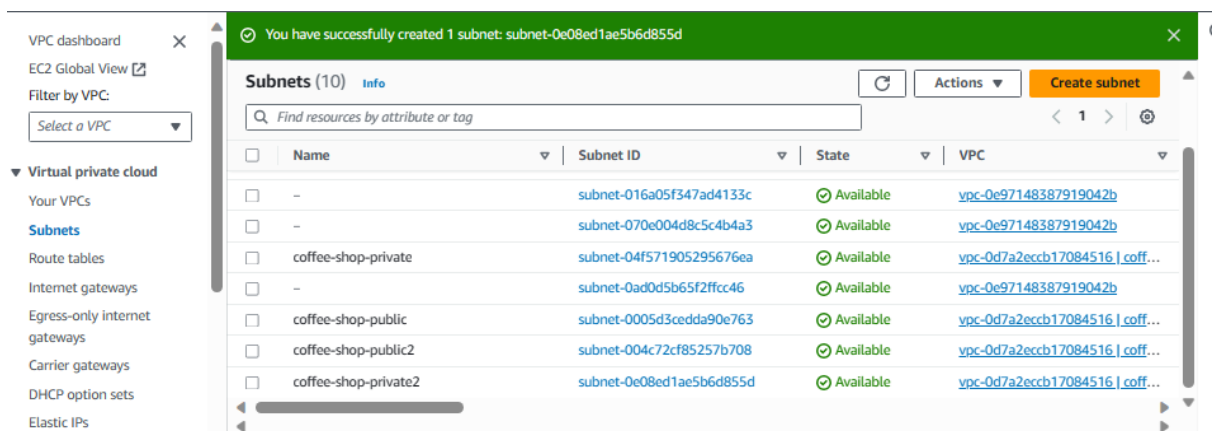
Launched EC2 instance named "coffee-shop-public" in the public subnet and EC2 instance named "coffee-shop-private" in the private subnet (Making 2 instances in the 2 different AZs).

Then, I setup the security groups for each instance, while creating them, to control inbound and outbound traffic.

Added an Internet Gateway for the "coffee-shop-vpc" for access to the public internet and a NAT Gateway for the private subnet for access from the private subnet to the outside but not the other way around. And added the necessary Public and Private Route Tables and associated them with the custom VPC for routing within and for exiting the VPC.

VPC > Route tables > rtb-09fb06db83871dd75

rtb-09fb06db83871dd75 / PublicRouteTable

Actions ▼

Details Info

Route table ID	Main	Explicit subnet associations	Edge associations
rtb-09fb06db83871dd75	No	2 subnets	–
VPC	Owner ID		
vpc-0d7a2eccb17084516 | coffee-shop-vpc	167834609858		

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	igw-0ba4ac8c610e6fb65	⊘ Active	No
10.0.0.0/16	local	⊘ Active	No


```
~~      \###|
~~       \#/ ___      https://aws.amazon.com/linux/amazon-linux-2023
 ~~      V~' '->
  ~~~         /
   ~~._.   _/
      _/ _/
    _/m/'
[ec2-user@ip-10-0-0-218 ~]$ whoami
ec2-user
[ec2-user@ip-10-0-0-218 ~]$ ls
[ec2-user@ip-10-0-0-218 ~]$ ls /
bin  boot  dev  etc  home  lib  lib64  local  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  va
r
[ec2-user@ip-10-0-0-218 ~]$ ls /home
ec2-user
[ec2-user@ip-10-0-0-218 ~]$ sudo yum update -y
Amazon Linux 2023 repository                                                    [===
                    Amazon Linux 2023 repository
  58% [=================================================Amazon Linux 2023 repository

         45 MB/s |  21 MB     00:00
Amazon Linux 2023 Kernel Livepatch repository                                  [===
                    Amazon Linux 2023 Kernel Livepatch repository
                                                           758 kB/s | 162 kB
   00:00
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-10-0-0-218 ~]$
```

i-085b6cef63122ae08 (coffee-shop-public)   ×

PublicIPs: 54.211.201.64   PrivateIPs: 10.0.0.218

Listing of all the other EC2 Instances:

```
PS C:\Users\Olatunji Olayinka\Documents\FormerHDD\Programming\Node-React-NPM&More\IntroToNode&Express\AzubiParole\Azubi2023Work-Cloud
Class\AzubiProjects\week13_projects\TODO\virtual-coffee-shop> Get-EC2Instance

GroupNames    : {}
Groups        : {}
Instances     : {coffee-shop-keypair}
OwnerId       : 167834609858
RequesterId   :
ReservationId : r-0c159bc0945fb7b00

GroupNames    : {}
Groups        : {}
Instances     : {coffee-shop-keypair}
OwnerId       : 167834609858
RequesterId   :
ReservationId : r-0d1c9e7389e55325f

GroupNames    : {}
Groups        : {}
Instances     : {coffee-shop-keypair}
OwnerId       : 167834609858
RequesterId   :
ReservationId : r-042d0b3a135332d04

GroupNames    : {}
Groups        : {}
Instances     : {coffee-shop-keypair}
OwnerId       : 167834609858
RequesterId   :
ReservationId : r-03586565271176772
```
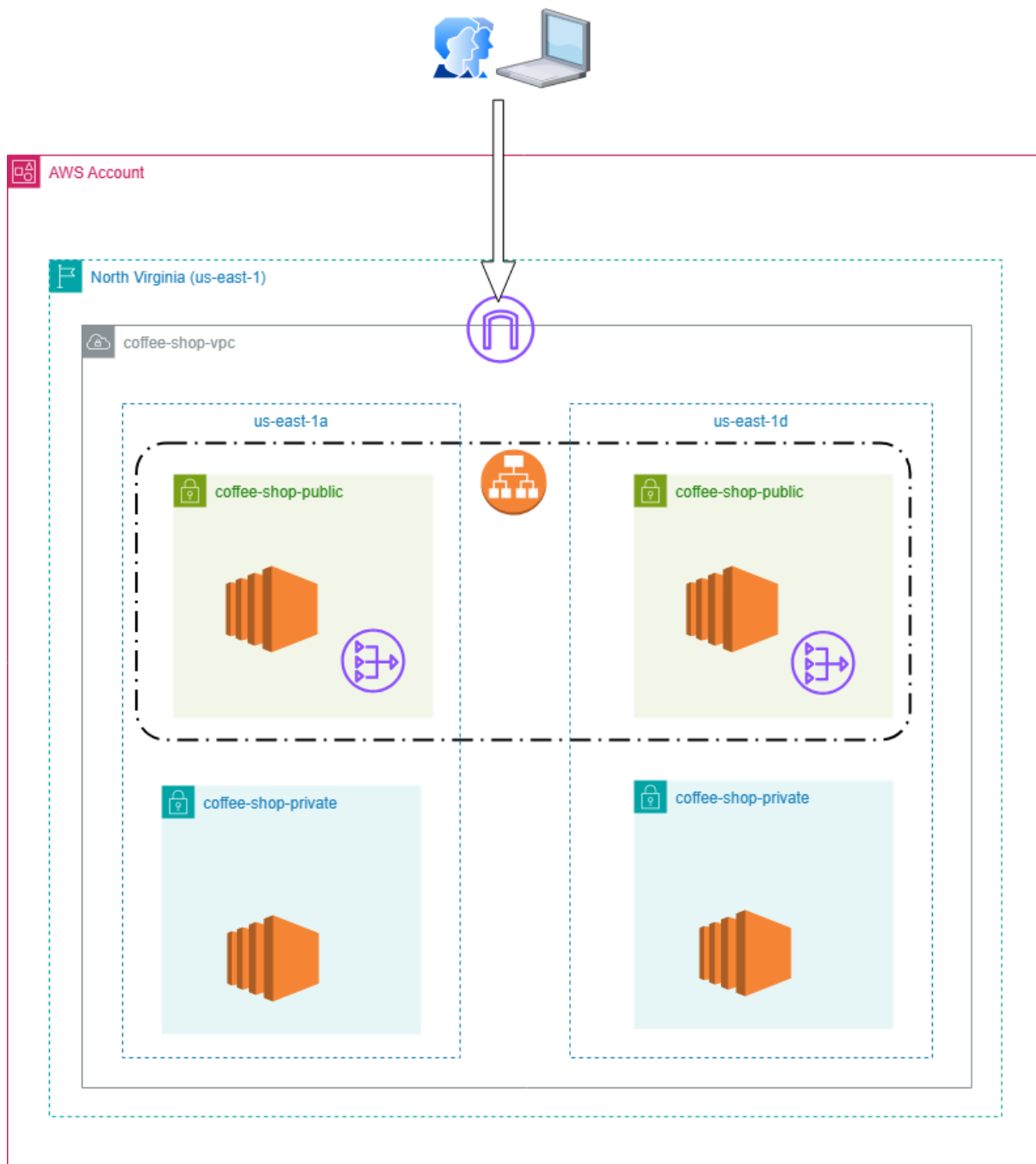
And setup Network ACLs to provide additional layer of security.

And finally, created and configured an Application Load Balancer in order to ensure smooth and balanced flow of traffic across multiple instances in a target group to provide maximum throughput and availability.

VIRTUAL COFFEE SHOP LAUNCH

You can view the associated GitHub here