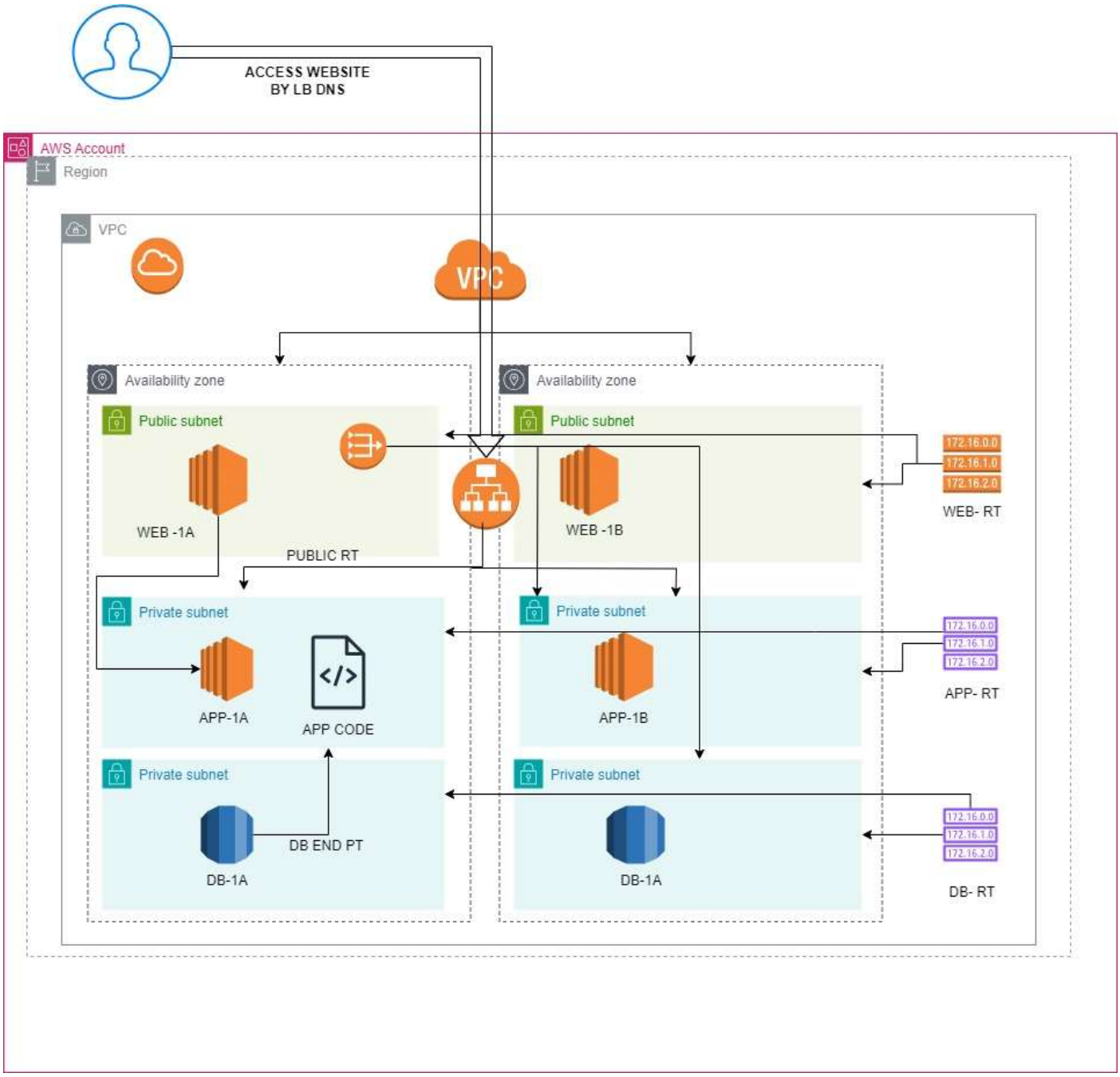
Create Three-Tier Application and Access it by using Load Balancer End point



Aim :- Create a three tier application i.e web tier, app tier, data base tier.

Resources Used :- Vpc, Application Load balancer, RDS, Php, apache, and MyAdmin Code(This is application code)

Procedure :-

1. Create VPC i.e **MyVPC** (name)
2. Create 6 subnets(**2 public-web-subnets, 2 is private-app-subnets, 2 private-DB-Subnets**) 3 in each availability Zone i.e us-east-1a,1b
3. Create **Internet gate way** in MyVPC and NAT gate way in public-web-1a.
4. Create **three route tables** they are **public-web-RT, private-App-RT, Private-DB-RT**
5. Attach Internet gate way to public-web-RT for accessing internet to vpc and attach Nat gate way to both app and database Route tables to access internet to private subnets
6. Create Security groups namely

   Web-server-SG ----- allow port number 20 (ssh) from anywhere i.e 0.0.0.0/0.
   App-Server-SG ----- allow ssh and give Web-server-SG in the Source
   DB-Server-SG ----- select Type MySQL port 3306 and give App-Server-SG in the source
   My-ALB_SG ----- allow Http from anywhere i.e 0.0.0.0/0 (give this SG while creating load balancer)
7. Now launch EC2 instace i.e public web server in public-web-1a of MyVPC with Web-server-SG and allow public Ip by click on enable and use amazon linux2 version because we have code which is supported by **2nd version only**.
8. Now launch two PHP-App_Server-01,02 in private-app-1a,1b subnets with respective security group and select Amazon linux2 AMI. (Note: we don't want public ip because we are lauching in private subnets)
9. Connect Public-Web-server through EC2 instance connect
10. Now we have to connect to private php-app server-01 from web-server, follow the below commands

    # sudo su –
    # vi keypair,pem
    add .pem data of private key which is downloaded in our local machine
    :wq
    # chmod 400 keypair.pem
    # ssh -i user@<private ip of app server 01>
11. Now instance is connected successfully to app server 01
12. Now we **need to install Lamp Server in php-app-1** for that click the below link and follow the steps

    https://docs.aws.amazon.com/linux/al2/ug/ec2-lamp-amazon-linux-2.html

13. After installation of lamp server now scroll down that page we can see php my admin application follow respective commands to install php my admin application in app server 01
14. After completion follow below commands to create the index file

    # ls

We can see PhpMyAdmin directory

 # cd /var/www/html

 # echo "PHP-server-01" > index.html

15. Follow the same procedure to connect to php server 02 and install lamp server and php my admin app
16. Now **Create Target group** i.e my-php-alb-tg and **register two php app servers** to this target group
17. Now **create application load balancer** in our vpc i.e MyVPC and **subnets public web 1a, 1b and add listner I,e target group** which we created earlier on Http:80 port
18. Now for verification **copy the end point and access it in browser** we will get data which is present in index.html file of php server 01, if we refresh the tab now we get index.html file of php server 02

    Give this in browser ----- **< ALB End Point >/index.html**

19. As of now we created two tiers now its time two create third tier i.e database tier
20. Go to aws and search RDS and select sunet group and **create subnet group with private database subnets 1a,1b.**
21. Now **create RDS** where **we select VPC Security groups as subnet group** which we created now
22. After creating of RDS **copy the DB End point** and add it in php config file to store the application data, for that follow below commands

    # cd phpMyAdmin

    # mv config.sample.inc.php config.inc.php ---- rename the file

    # vi config.inc.php

    Go to host section and in place of "Host Name" give RDS end point which we copied earlier

    :wq --- write and quit

23. Go to browser and and give **"<ALB End point>/PhpMyAdmin"** now it loads application
24. When you enter the data and saving it will show some error I.e load balacer store the value with itself because both instaces are healthy so we have to define that store the datsa in any one instance for that **enable stickiness** in attributes section of target groups **so that ALB will send the data to any one of the server**, now enter again the data and after submit check the RDS we have our data.