

AWS 3 TIER PROJECT (BLOOD BANK)

CREATE VPC: 10.0.0.0/22

PUB-SN-1 : 10.0.0.0/25

PUB-SN-2: 10.0.0.128/25

PRV-SN-1: 10.0.1.0/25

PRV-SN-2 : 10.0.1.128/25

PRV-SN-3 : 10.0.2.0/24

CREATE IGW AND ATTACH TO VPC

CREATE 2 RT (PUBLIC & PRIVATE)

PUBLIC - RT ----> ATTACH IGW AND SUBNET ASS

PRIVATE RT ----> NAT GW ATTACH & SUBNET ASS ONLY

CREATE SG FOR BANK-VPC WITH ALL TRAFFIC

CREATE DATABASE:

DB instance identifier: mysqlldb

username: admin

pass : admin123

VPC: BANK-VPC

SG: BANK-SG

endpoint: mysqlldb.c7wqmm44gl92.us-east-1.rds.amazonaws.com

CREATE EC2 INSTANCE ON PRIVATE SUB-1

IT WILL NOT CONNET SO ALLOW IGW FOR THATS PRIVATE RT FOR SOMETIME (UNTILL THE SCRIPT EXECUTES) LATER WE NEED TO REMOVE IT.

run script from github

clone the repo

copy files to httpd path

then we can access the app

CONFIGURE DATABASE:

update databse end point on these files

vi donate-blood.php

vi find-donor.php

vi config.php

vi search.php

vi signup.php

vi deletedata.php

connect to database : mysql -h endpoint -u user -p

enter password

CREATE DATABASES & TABLES

get the queries from github

THEN CHECK IT

select * from table_name;

lets create AMI, it will take some time

LAUNCH 3 INSTANCES ON 3 PRIVATE SUBNETS

REMOVE IGW FROM PRIVATE RT

CREATE TARGET GROUP BY USING 3 INSTANCES

CREATE LB ON 2 PUBLIC SN AND CHECK WITH DNS NAME

NOW TERMINATE ALL INSTANCES

CREATE LAUNCH TEMPLATE FROM AMI

CREAE ASG GROUP FROM LAUNCH TEMPLATE AND ATTACH LOAD BALANCER.

FOR MICROSERVICES:

CREATE VOLUME : docker volume create database

UPDATE HOST AS mysqldb ON ALL PHP FILES

AND USERNAME AS root

DOCKERFILE FOR APP:

FROM php:7.4-apache

Install mysqli extension

RUN docker-php-ext-install mysqli

Copy your application files

COPY . /var/www/html/

BUILD THE IMAGE : docker build -t image1 .

RUN IMAGE : docker run -itd --name myapp -p 8089:80 --link mysqldb:mysqlcon image1

DOCKERFILE FOR DB:

Dockerfile for custom MySQL

FROM mysql/mysql-server:5.7

Copy initialization SQL script to MySQL container

COPY init.sql /docker-entrypoint-initdb.d/

#DB_PASSWORD

ENV MYSQL_ROOT_PASSWORD=admin123

BUILD THE IMAGE : docker build -t database .

RUN IMAGE : docker run -d --name mysqldb -v database:/mydb -p 3309:3306 database

GRANT ALL PRIVILEGES ON *.* TO 'root'@'%' WITH GRANT OPTION;

FLUSH PRIVILEGES;