Seneca

Assignment 1 Report : Create Dockerfile, build docker image and deploy docker container on Amazon Linux EC2

Name : Arpit Gupta
Student Id : 169572237

Link for Github Repo : <u>Arpit-commits/Arpit-Assignment</u>

Error/Issues Faced While Working on the assignment Are -

1 – Security Group Issue
Problem: EC2 instance couldn't connect because security group rules were incorrect.
Solution: Updated Terraform to allow necessary inbound/outbound traffic for SSH, HTTP, and the application ports.

2 - Git Large File Issue:
Problem: Tried to push large Terraform files exceeding GitHub's 100MB limit.
Solution: Removed unnecessary .terraform directories and used .gitignore to avoid tracking them.

3 - Branch Checkout Issue:
Problem: Couldn't switch branches due to untracked files.
Solution: Moved or removed conflicting files before switching branches. As I was using 3 Branches in Total Development , Staging and Main

4 – Key Issue Authentication
Problem : First I created a private key to login to the newly created and then pushed the key with the code in the github then once again going through the documentation I realized that's not necessary as I can connect to the instance directly from the AWS connect
Solution : I removed the key when I created the staging branch where the code is mostly sorted and removed more un-necessary files

5 - Missing ECR Repository:
Problem: Docker push failed because mysql-repo and webapp-repo didn't exist.
Solution: Added Terraform code to create mysql-repo and webapp-repo in AWS ECR, This happened because when I ran the terraform code locally the previous repo were already there as they did not got destroyed , but when I reset and started fresh again I needed to create the repo in ECR so later added them in the terraform code

voclabs:~/environment/Arpit-Assignment (main) $ docker push 135893829551.dkr.ecr.us-east-1.amazonaws.com/mysql:latest The push refers to

repository [135893829551.dkr.ecr.us-east-1.amazonaws.com/mysql] 21e57c4a201c: Preparing a3d707e222b9: Preparing 9dad147013d2: Preparing f130fbfe77ff: Waiting b50bc0309705: Waiting 69b327c9e57e: Waiting 4683b2db23ba: Waiting 88ae57e8be69: Waiting ca760287abdb: Waiting edb9a1d4fec8: Waiting 477d917f8fdf: Waiting 7600fdef234b: Waiting name unknown: The repository with name 'mysql' does not exist in the registry with id '135893829551' voclabs:~/environment/Arpit-Assignment (main) $

6- Ping Command not found
Problem : Had issue when trying to ping 1 web app to the another one
Solution:as ping utility was missing installed it and then we were able to ping the other webApps

7 – Ubuntu Package Manager Issue:
Problem: Ubuntu doesn't support yum, needed for installing dependencies.
Solution: Used apt instead of yum to install required packages.

8 - Multiple Subnets
Problem: AWS environment, and Terraform cannot decide which one to use without more specific criteria
Solution: added aws_subnet block in the terraform code

9 – Duplicate Definitions
Problem : I Put output in both places , Output.tf and Main.tf in terraform code
Solution : Removed the duplicate output definitions and changed to only main.tf

10 – AWS Image
Problem – AWS image not found
RunInstances, https response error StatusCode: 400, RequestID: 903a45c5-0fef-4356-b64e-231a860c2922, api error InvalidAMIID.NotFound: The image id '[ami-0c55b159cbfafe1f0]' does not exist
Solution : Used Dynamic AMI id

11 - Argument is deprecated
Problem :
│ Warning: Argument is deprecated │ │ with aws_eip.nat_ip, │ on main.tf line 43, in resource "aws_eip" "nat_ip": │ 43: vpc = true │ │ use domain attribute instead │ │
Solution : solution is to use associate_with_private_ip , used eip block in the code resource "aws_eip" "nat_ip" { domain = "vpc" # Use "domain" instead of "vpc = true" }