***Title: using terraform (IAAC) for cloud state management and launchinf the vprofile application.***

Introduction : here in this project we are using terraform that is a iaac tool, we will basically write scripts or .tf files that will setup all the infrastructure on aws cloud and also provision it for our vprofile application. In this projectr we wil be using the terraform branch from vprofile project. And I have written my full code that will be in the repo of personal muhmmadayan github. Alsowe will be implementing good devops practices like version cotrol for iaac, use s3 to maintain the state, fmt the iaac to make it more readable , use vars file to define all the variables used which will help if we recreate the propject later. Like id pass for the servers, zones used, keys, username, ips, cidr info etc.

Now lets see the flow of execution.

1. terraform setup and s3 for backend : first download the terraform version on local mahine.since in an agile env a lot of people will be using terraform to deploy app, so we need to preserve the state of current infra, so we will be using the s3 bucket for it. Also don’t put public or priv key in iaac, firs configure awscli . then witre backen ftf file with mentioning bucket for backend.

2. write vars.tf and providers.tf file : providers.tf file is simple the aws provider mentioning. Amd in vars.f all the variables are gonna be in it. Firsly region, ami depending on the zone that we are currently using, priv key path, pub key path, username for instacnes, myip, username and pass for rabbitmq ,db, then vpc anme , zones for vpc, pubsubcidr 3 , privsubcidr3, .thatall in the vars.tf file.

3. key pair.tf file : we need keypair to login to the instances that we are gonna be using in this propject right ? so we will actually create it locally, then put pub and priv file in terraform folder. Then we have already mentioned their path in vars file. But first we need to upload our locally created .pub to aws accont for that we will create tf file that wll use resource aws\_key\_pair , and we will mention its path, it will upload its contents on aws accounts.

4. vpc creation : now we have to crate a vpc using iaac. We will use module vpc and provide name and cidr info, znes,priv,pubsub,natgateeway,igw,. And it will simple create full vpc with privsub,pubsub in 3 azs and also attach igw nat gateway to respective subnets.

5. secgrp.tf: now this file is important because it will define or gobvern all the inbond and outbound rules for instacnes we are gonna use like for backend , frontemd etc. we use resource aws\_security\_grou to create scgrps for elb ir loadbalancer, bastionhost, tomcat, backend.

6. backend-services.f: here in this tf file we will be usingit for launching all the backed servicdes like rds for databse, activemq, and elasticache. We mentioned rds info, active mq,elastic cache with proper ports and username,pass,names for our vprofile app. Now we will have these servives up in cloud, we will be needing their endpoints later to connect our front end to backend such that smooth communication happens between them. (check the code comments to better understand them.)

7. beanenv : now our main hero ir tomcat server, we will be using elastic beanstalk for deployment of iut vprofil application. We have bacend servies running perfecetly and we also hae their endpiints. Now we will use amazonlinux 2 v4.3.10 with coretto 11 and tomcat 8.5. we will launch it in priv subnet only as we wil be lauching all the bgackend servers and frontend servers in prv subnet sonly to better secure our vprofil application. Now we will launch all the settings like autoscaling,subnets,lauchconfig,zones,env,update polic,lb,stickeyness enabled,batchsize,securitygrp of it and elb.

\now at this time we have backend serves and tomcat running default application working perfectly, now we have to deploy our vprofile application on this infra.

8. bastionhost.tf: now we have to lauch and use bastion host to actually login to instacnes to our priv subnet ie tomcat and then upload artifact in it. We will now use batin host to first assign rds enpoint dbuser,dbaoss and then run a script ir the template script wihich will clone a git repo and then install mysql client and then upload the db.sql file to rds databse. It will put all the data for our vproapp in rds. Now we have rds also provisioned and first we edit the application. Prop file in localsytem and edit it with endpoints of db,activemqq,elastic cache and then build the artifact locally. We will then manually upload artifact to branstalk env and our application is deployed fully.

9. validate the application and everything should work fine.

***Summary: in this project we will use terraform(IAAC) to deploy our vprofile application using beantalk , rds,activemq,elasticache in 3 priv subnets in a vpc with 3 zones for high availability.***