AWS Practice

**Install Tomcat/Jenkins**

Login to RedHat Ec2

Sudo -i => to navigate to root user

Yum install nginx

Systemctl enable nginx

Systemctl start nginx

Systemctl status nginx

sudo yum -y install java-11-openjdk java-11-openjdk-devel

# yum install wget zip unzip (to get packages from outside to Ec2 machine, we have to use wget(webget))

Search wget download tomcat  
right click on Zip and copy link  
  
in terminal wget paste the link

# ls

“apache-tomcat-8.5.86.zip”

# unzip apache-tomcat-8.5.86.zip

# ls -l

# df -h

# ls

# mv apache-tomcat-8.5.86 /opt/tomcat

# ls

# cd /opt/tomcat

# ls -l

# cd

# ls

# cd /

# ls -ltra

# cd

# chmod -R 755 /opt/tomcat/ To assign privileges to tomcat package

# /opt/tomcat/bin/startup.sh

Search Jenkins war -> go to Jenkins.war

Right click on any war file and copy link

Wget paste link

Now copy war file to tomcat in webapps

Once check browser with public IP:8080

If you are not able to open -> go to EC2 page in aws console

Open instance -> security tab -> click on security groups -> edit inbound rules -> add rule -> in place of 0 replace it with 8080 -> anywhere IPV4 -> save rules -> NOW TRY IN BROWSER

cp jenkins.war /opt/tomcat/webapps

now go to browser

<http://44.213.110.38:8080/jenkins> => will redirect to Jenkins

to get administrative password, copy the red line on screen

cat paste the link

use password -> Install suggested plugins

Jenkins is nothing but the collection of plugins, plugins are used to coordinate all the tools Jenkins URL

JenkinsNew

Virat@2014

<http://44.213.110.38:8080/jenkins/>

**Building a small project in Jenkins**

+New item -> name – Project1 -> freestyle -> ok -> Description - This is a sample project -> Add build step – execute shell -> date(inside box) – apply and save -> Build Now

#ls -ltra

You can find .jenkins

#cd .jenkins

#cd workspace/

#ls

You can find project1

To know whether the tomcat is active or not

ps -ef |grep tomcat

opt/tomcat/bin/shutdown.sh

Ps -ef |grep tomcat => single line output

**IAM – Identity Access Management**

Console -> IAM -> Users(to add a new user) -> Add User -> Siva ->   
Provide user access to the AWS Management Console – *optional ->*

I want to create an IAM user

Autogenerated password

Next ->   
Add user to group -> Create group -> User group name – Dev

|  |  |  |
| --- | --- | --- |
|  | [AmazonS3FullAccess](https://us-east-1.console.aws.amazon.com/iam/home#/policies/arn%3Aaws%3Aiam%3A%3Aaws%3Apolicy%2FAmazonS3FullAccess) -> **Create user group** |  |

* Next -> <https://191208257919.signin.aws.amazon.com/console>

Siva

pP{X0U6!

Access the above link and try to login

New password: Dharma@2014

**MFA**

IAM > Security credentials > Assign MFA device – any name > next > showqr code > in mobile download Microsoft authenticater > scan the QR and enter 2 sequence codes here> Add MFA > signout from account > Signin > MFA code > sign in

IAM > Security credentials > Virtual -> remove

**S3**

**Amazon S3 > Buckets > Create Bucket > mys3bucket1(should be unique and lower case)**  


ACLs enabled

**Block all public access – uncheck**

I acknowledge

Bucket Versioning



Disable



Enable

**Go Inside the bucket and upload file**

**Try to access with** Object URL **In browser -> is it doesn’t work**

Go to console > go to document > object actions > Make public > try accessing now

To create replication set

Create one more bucket in another location  
go to source bucket > management > Create replication rule > name(Saranya-rc) >   
destination – browse s3 > choose new bucket and choose path > IAM role – create new role – save > Yes – submit > brows3 s3 – new bucket > save

Upload new file in source bucket and test

**Classic Load Balancer**

EC2 > Load Balancer > create load balancer > classic load balancer > create > name- saranya-clb > select VPC > always select public subnets for load balancer > assign security groups > create new security group > clb-sg > configure security settings > configure health check > ping protocol -TCP > interval -15 > healthy -5 > add EC2 instances > can add all the Ec2’s you have > add tags > name – clb > review > create > close

To check status of load balancer > go to load balancer > instances > status

**Application Load balancer**

EC2 > Load Balancer > create load balancer > application load balancer > create > Saranya-alb > select VPC > select public subnets > VPC-sg > right click on create target group and select open new tab > target group name – alb-tg > next > select vpc > create target group > go back – refresh the page and select target group > create load balancer > view

\*\*what is the difference b/w classic and application load balancer??

**RDS**

EC2 > Security groups > create security group > RDS-sg > description- RDS > select VPC > add rules > select MySQL instead Custom TCP > copy EC2 security group Id and paste under source info > create security group

RDS > create database > standard create > MySQL > Templates-free tire > Master password(Virat@2014) > instance configuration (t2 micro) > select VPC > select security group(RDS-sg) > select region 1a > create database >

Connect to instance

Browse > how to install mysql and redhat9 > run

dnf install mysql-server

systemctl start mysqld.service

systemctl enable mysqld.service

Go to database instance > copy endpoint

#mysql -h ‘paste endpoint’ -u admin -p

Enter DB password

mysql> show databases;

mysql > exit

Try to browse public IP

Select database > actions delete > untick the create final snapshot > untick retain automated backups > tick I acknowledge > delete me > delete

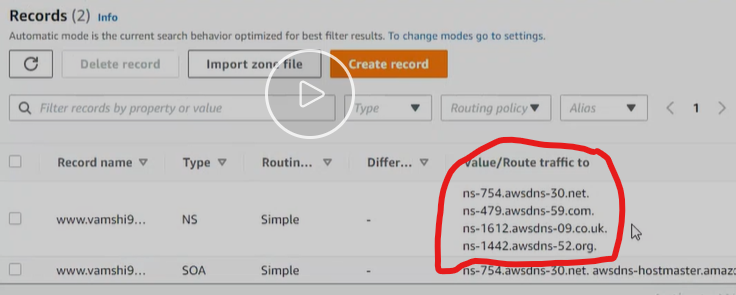
Route53

Go to freenom site > check availability for Saranya.tk or try saranta708.tk

After getting new domain

Go to Route53

Create hosted zone > give www.domain name(created) > create hosted zone > copy 4 name servers > goto freenom website > managed dns > name servers > use custom name servers > paste one by one name servers from aws to empty boxes > change name servers  
aws > create record > value (Ec2 public IP) so that newly created domain will be assigned to this particular Ec2 public IP > create records



Hosted zone > open domain > tick first domain – delete record >

Route 53 > hosted zone > select domain > delete

Terminate instance > > delete subnets > delete rout table > delete IGW > delete NAT > delete VPC

AutoScaling:

AWS Auto Scaling monitors your applications and automatically adjusts capacity to maintain steady, predictable performance at the lowest possible cost.

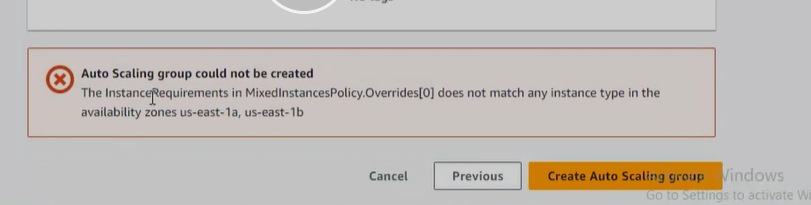
horizontal scaling is about adding more machines of similar capacity to the infrastructure.

vertical scaling is about changing the instance up and down

SNS : **SNS** stands for Simple Notification Service. · It is a web service which makes it easy to set up, operate, and send a notification from the cloud

Topic name – Saranya-sns > next > display name - sns1 > create topic > create subscription > protocol – email – [mesaranyagogineni@gmail.com](mailto:mesaranyagogineni@gmail.com) > create subscription > goto email and confirm >

Go to EC2 > Autoscaling > Autoscaling Group > Create Auto scaling Group > Saranya-ASG > right click on create launch templete and open new tab > name - saranya-LT > select currently in use > select an existing security group – select vpc > create launch template > view launch template > go to autoscaling page and select template > next > select VPC > select public availability zones > next > desired capacity -3 , min -2, max-3 > next > add notification select sns > next > next >



Cancel

Create autoscaling group > Saranya-asg > select template > next > select vpc > select regions > next > add notification > next > create autoscaling groups > still same issue   
 try deleting template and create new > .

name - saranya-LT1 > select currently in use > instance type - t2.micro > select an existing security group – select vpc > create launch template

Cancel

Create autoscaling group > Saranya-asg2 > select template > next > select vpc > select regions > next > add notification > next > create autoscaling groups

EC2 dashboard > check instances > new machines created > if we terminate once machine immediately new instance get created >

Delete autoscaling groups

GIT

Install Git  
create an account in git hub.com  
create a folder name gitpractice  
from that folder right click and open git bash here

Git init

Git status  
create new file with touch or cat >>

Git config --global user.email “git email”

Git config --global user.name “git username”

Git add `new file name`

Git commit -m “This is the first commit” `new file name`

Git status

Git log

Git remote -v

git remote add origin <https://github.com/azb26cloud/devops.git>

Git remote -v

Files got added

$ git config --global http.sslVerify false > to disable certificate verification

Git push -u origin master

Git remote remove origin > to remove existing origin and add new origin

git remote add origin <https://github.com/azb26cloud/aws.git>

git branch

to create new branch

git branch dev > check git branch

to update the branch from git bash

git checkout dev > it will switch to dev branch

git checkout -b qa > will create and switch to new branch

git branch -D qa > to delete the branch > it will only delete branch locally not in central repository for that

git push origin --delete remotebranchname

GIT PROMETHEUS

Copy clone code from it

Gitbash terminal > git clone <https://github.com/prometheus/prometheus.git>

Git log > to know what are all the files committed by us

Working area staging area

File created adding file

git reset --soft 893575df8b12378846194ab4a7a181759dd2ddad (last commit ID from `git log`)

git status > to check the reset status

soft will bring back that commit to index or staging area

hard will delete the file permanently

revert will undo the last commit changes, creates new commit ID

git reset --hard 893575df8b12378846194ab4a7a181759dd2ddad (last commit ID from `git log`)

$ls

git revert 893575df8b12378846194ab4a7a181759dd2ddad (last commit ID from `git log`)

To delete an individual file from file system and index

git rm file1.txt

git commit -m "deleted the file1.txt from the file system"

git push

$ git ls-tree -r master > to verify to list files in the current index

Branching Strategy