Aws Project Infrastructure Setup

Custom Vpc Setup:

- **Log into Aws management Console.**
- **In Search box search vpc.**
- **Create Vpc and Give you required CiDR Example (10.100.0.0/16)**
  This CIDR defending upon the Project Requirement.
  For Best practices defending upon Availability zone Create the Subnets For Mumbai 3 we need to create 6 3 is Public and 3 Private. CIDR FOR Subnets Example (10.100.0.0/24 10.100.1.0/24 like this)
  It is varying defending upon Availability Zones.
- **Create the Internet gateway and Nat Gateway and Internet gateway attach to Vpc and should remember the Nat gateway always placed the Public Subnet.**
- **Create the Public and Private Route Table and add the routes and subnets.**
- **For Public Route table Route 0.0.0.0/0 and Internet gateway and Private Route table Route table 0.0.0.0/0 Nat gateway.**

How to Create the SSM role and what is the Use:

- **Go to IAM section and select the IAM role and Give select the Ec2**
- **In permissions tab just search the SSM and give you ssm full permissions.**
- **As well for ECR in Search box just search the ECR and give the permissions.**
- **And give Role Name and Create the role.**

Use of SSM Role

- **To connect the our instance to secure way that's reason we using the IAM role.**
- **Actually Role is useful to to establish connection between to services in our Project To useful to ECR logins.**

Launch the OpenVPN for Private Instances:

- **Go to Ec2 dashboard and click on Launch Instance and in Ami selection Select the OpenVPN and Click On subscribe Button.**
- **In network Section select the our vpc and subnet is public subnet and should enable the enable public ip.**
- **Go to Advanced setting and select IAM profile and add the our SSM Role.**
- **Launch Instance.**
- **After the Launch the instance just select the instance and click on Connect Button by using Session Manager, we Connect the Server or Instance.**
- **Just type Sudo Su this command to useful to log our session.**
- **Next you able to see one OpenVPN page you just give yes and click Enter button and they asked the password you must the give your password and confirm password again and just enter.**
  Now you able see the admin and Client URLS just click Client URL Download the Profile.
  - **Please follow the below link you able to download the OpenVPN Client the Client helpful to our OpenVPN Connection.**
    OpenVPN Connect - Client Software For Windows | OpenVPN
  - **After Setup just OpenVPN and click + button and Upload the download Profile.**
  - **Now you able see the OpenVPN is connected.**

ECR Setup

- **Just open the ECR service click the create and select the Private and just give repository name and click the create.**

Creation Of Custom Security Group

- **Just open ec2 dashboard in Security section just create the security Group and add the required port numbers in our case we need to 8080 and 80 and 443 and 22 Anywhere,Give the Security Group Name.**
- **In Below you see to one option Vpc select the our Vpc.**

Jenkins Server Setup, Project clone and Image Push to ECR Repo.

- **Just Go to Ec2 dash board click the launch the instance and AMI section select the [amazon Linux 2023] instance type defending upon project requirement in our case we choose the t2.xlarge.**
- **In Network section select the our Vpc and select the Private Subnet.**
- **And select our Security Group. [previously created Security Group.]**
- **And Advanced just Give IAM role in IAM profile section.**
- **Launch Instance.**
- **After just select the instance and connect the instance and Sudo -i this command is useful to log in our server.**
- **Please follow below commands to helpful to install the Git and Java and Docker.**
  Java is should be installed for Jenkins.
  Git is should be installed for clone our project.
  Docker is should be installed for ECR image.

  #yum install git -y [install git]
  #yum install java -y [install java]
  #yum install docker [install docker]
  #systemctl start docker [start the docker service]
  #systemctl enable docker [enable the docker service]
  #systemctl status docker [to check the docker status]

Now Installation of Jenkins

Add the Jenkins repo using the following command:

```
1.sudo wget -O /etc/yum.repos.d/jenkins.repo \
  https://pkg.jenkins.io/redhat-stable/jenkins.repo
```

Import a key file from Jenkins-CI to enable installation from the package:

```
2.sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io-2023.key
```

Install Jenkins:

```
3.sudo yum install jenkins -y
```

Enable the Jenkins service to start at boot:

```
4.sudo systemctl enable Jenkins
```

Start Jenkins as a service:

sudo systemctl start jenkins

Below link is official document link to install Jenkins in amazon linux.

Jenkins on AWS [just right click and open the hyperlink option you able to see website.]

- **And next execute the below commands**

  #visudo

  Now you able to see the one sudor's file and go to below and choose Root user below root user add the below line.

  jenkins ALL=(ALL) NOPASSWD: ALL

  :wq! Save the file.

- **Next follow the below commands:**
- sudo usermod -aG docker jenkins
- sudo chmod 666 /var/run/docker.sock

- **Now just copy the Jenkins Private IP and browse the Private IP like bewlo:**

  IP:8080

  Now you able to see the Jenkins dashboard

  Copy the path [/var/lib/jenkins/secrets/initialAdminPassword]

  And in Jenkins server cat /var/lib/jenkins/secrets/initialAdminPassword now you able to see the password just copy the password and paste in to Jenkins Dashboard.

- **Next Select Installed Plugins and next setup the username and Password and Name and Email and setup.**
- **Next Go to Manage Jenkins and Credential section add credential and Password section paste the GitHub token.**

  For GitHub Token Log in to GitHub and go right corner click on symbol and click on Setting and Select the developer settings and tokens select the classic tokens and click on all options and create the token and copy the token and paste above as mentioned

- **Now Clone the Project**
  Below I provide the github repo just copy it.
  https://github.com/anildevops7702/vv.git just copy the url
  in our Jenkins server follow the below command.

  #git clone https://github.com/anildevops7702/vv.git
  Now you able to see vv folder
  Just go to that folder by using [cd vv] command cd vv is command.
- **Now Go to ECR dashboard and select repo and click on Push commands.**

Please all Push Commands now you able to see the Your image pushed into Jenkins server into ECR repo.

Route53, ACM, ECS And Load Balancer Setup

- **First you need to one domine for route53 because that go-to go daddy website and purchase the one domine.**
- **Next in go to our Aws Route53 service and create the hosted Zone ns now able to see the names servers.**
- **Go to Go Daddy website and select the your domine and in DNS option you able to see default nameserver and edit and add our own nameservers.**
- **Just Copy the one-by-one nameservers totally 4 in Route53service and paste Into Go daddy Name server s just wait for 24 hours for activating the name servers.**
- **Next Go to Certified Manager service create the certificate and below you able the see the one option create the records in route52 just click and add the records in route53.**
- **Go to ECS service and Create the Cluster and type is Ec2 instances and Select the on-Demand Auto scaling Group and in Root Volume size just select the 30.**
- **In Network Section just select the our Vpc and select the private subnets and select the our previously created Security Group.**
- **Next Create the Cluster.**

- **Next select Task Definitions and Give me name and select the EC2 launch type and in container details give ECR repo details like name and repo URI and port number give Container is 8080 Is our project Port Number it is varying from different Projects just remember.**
- **Next Resource allocation Memory hard limit is 2 and soft limit is 1 just remember don't forget five these details.**
- **Next Create the task definition.**

Next create the Load balancers:

- **Go to Load balancers tab and select the target group and select target type is the Ip address type.**
- **Next select the our Vpc Click the next and add our instances ip address by means ECS Creation the some instances copy the private ip address and paste and including as pending option and click on Register the targets.**
- **Next creation of load Balancers and select the application type and Give name and internet facing and Network section select the our vpc and select the public subnets.**
  Next Enable the WAF for security.
- **Create the load balancer and next go to listener add the listener and add Https and 443 and forward to Tg and select our Certification.**
- **Next select the 80 listener and edit and 80 forward to 443.**
- **Next Go to ECS service and go to Task definitions and select the task definitions Deploy option select the service.**
- **Select the cluster and select the launch type and select the EC2 and give the service name and Networking section select the our vpc and Private subnets and select the our security group.**
- **In load balancer section add our load balancers.**
- **Create the service.**
- **Next Go to Load balancers just copy the load balancers DNS and go to route53 service and create record Give name and paste the DNS in value place and select the C-Name.**
- **Copy your Record name and browse it know you bale to see your website in secure way.**