**VPC CLI**

**Step 1: Create VPC**

VPC\_ID=$(aws ec2 create-vpc \

--cidr-block 10.0.0.0/16 \

--tag-specifications 'ResourceType=vpc,Tags=[{Key=Name,Value=UST-VPC-B}]' \

--query 'Vpc.VpcId' --output text)

**Step 2: Create Subnets**

PUBLIC\_SUBNET\_ID=$(aws ec2 create-subnet \

--vpc-id $VPC\_ID \

--cidr-block 10.0.0.0/17 \

--availability-zone us-east-1a \

--tag-specifications 'ResourceType=subnet,Tags=[{Key=Name,Value=VPC-B-PubSub }]' \

--query 'Subnet.SubnetId' --output text)

PRIVATE\_SUBNET\_ID=$(aws ec2 create-subnet \

--vpc-id $VPC\_ID \

--cidr-block 10.0.128.0/17 \

--availability-zone us-east-1b \

--tag-specifications 'ResourceType=subnet,Tags=[{Key=Name,Value= VPC-B-PriSub }]' \

--query 'Subnet.SubnetId' --output text)

aws ec2 modify-subnet-attribute \

--subnet-id $PUBLIC\_SUBNET\_ID \

--map-public-ip-on-launch

**Step 3: Create and Attach Internet Gateway**

IGW\_ID=$(aws ec2 create-internet-gateway \

--tag-specifications 'ResourceType=internet-gateway,Tags=[{Key=Name,Value=MyIGW1}]' \

--query 'InternetGateway.InternetGatewayId' --output text)

aws ec2 attach-internet-gateway \

--internet-gateway-id $IGW\_ID \

--vpc-id $VPC\_ID

**Step 4: Route Tables**

PUBLIC\_RT\_ID=$(aws ec2 create-route-table \

--vpc-id $VPC\_ID \

--tag-specifications 'ResourceType=route-table,Tags=[{Key=Name,Value=B-PubSub-RT }]' \

--query 'RouteTable.RouteTableId' --output text)

aws ec2 create-route \

--route-table-id $PUBLIC\_RT\_ID \

--destination-cidr-block 0.0.0.0/0 \

--gateway-id $IGW\_ID

aws ec2 associate-route-table \

--subnet-id $PUBLIC\_SUBNET\_ID \

--route-table-id $PUBLIC\_RT\_ID

PRIVATE\_RT\_ID=$(aws ec2 create-route-table \

--vpc-id $VPC\_ID \

--tag-specifications 'ResourceType=route-table,Tags=[{Key=Name,Value= B-PriSub-RT }]' \

--query 'RouteTable.RouteTableId' --output text)

aws ec2 associate-route-table \

--subnet-id $PRIVATE\_SUBNET\_ID \

--route-table-id $PRIVATE\_RT\_ID

**Step 5: Create Security Group**

SG\_ID=$(aws ec2 create-security-group \

--group-name WebSG \

--description "Allow HTTP and SSH" \

--vpc-id $VPC\_ID \

--tag-specifications 'ResourceType=security-group,Tags=[{Key=Name,Value= WebSG }]' \

--query 'GroupId' --output text)

# Allow SSH & HTTP

aws ec2 authorize-security-group-ingress --group-id $SG\_ID --protocol tcp --port 22 --cidr 0.0.0.0/0

aws ec2 authorize-security-group-ingress --group-id $SG\_ID --protocol tcp --port 80 --cidr 0.0.0.0/0

**Step 6: Launch EC2 with Bootstrap Script**

read -r -d '' USERDATA <<EOF

#!/bin/bash

while ! ping -c 1 -W 5 8.8.8.8 &>/dev/null; do

echo "Waiting for internet connection..."

sleep 5

done

sudo yum update -y

sudo yum install -y httpd

sudo systemctl start httpd

sudo systemctl enable httpd

sudo chown -R apache:apache /var/www/html

sudo chmod -R 755 /var/www/html

echo "<h1> CLI Commands Done \$(hostname -f) </h1>" | sudo tee /var/www/html/index.html

EOF

INSTANCE\_ID=$(aws ec2 run-instances \

--image-id ami-06616d5385afedaa5 \

--instance-type t2.micro \

--key-name bhargav1 \

--security-group-ids $SG\_ID \

--subnet-id $PUBLIC\_SUBNET\_ID \

--associate-public-ip-address \

--user-data "$USERDATA" \

--tag-specifications 'ResourceType=instance,Tags=[{Key=Name,Value=WebServer1}]' \

--query 'Instances[0].InstanceId' --output text)





