**AWS CLOUDING**

|  |  |  |
| --- | --- | --- |
|  | **VPC** (Virtual private cloud) | virtual net |
|  | **SUBNET** | nets inside the VPC net with a range of IPS to be used. Allows organization and security inside the VPC. |
| **Icon  Description automatically generated** | **ROUTING TABLE** | contains a set of rules, called routes, that determine where network traffic from your subnet or gateway is directed |
| **Icon  Description automatically generated** | **INTERNET GATEWAY** | component that allows communication between your VPC and the internet |
|  | **SECURITY GROUPS** | controls the traffic that is allowed to reach and leave the resources that it is associated with. |
| **Icon  Description automatically generated with low confidence** | **EC2 INSTANCE** | Virtual server inside aws cloud also known as virtual machine (azure). Elastic Compute Cloud (scalable) |
| **Icon  Description automatically generated** | **ELASTIC NETWORK INTERFACE** | Logical network component to specify private ips, eip and security groups |
| **Shape, arrow  Description automatically generated** | **ELASTIC IP** | a static public IPv4 address associated with your AWS account in a specific Region. (scalable) |
| **Icon  Description automatically generated** | **PEERING** | networking connection between two VPCs that enables you to route traffic between them using private IPv4 addresses or IPv6 addresses. Instances in either VPC can communicate with each other as if they are within the same network |
|  | **CIDER BLOKS** | allocating IP addresses and for IP routing. (IP range) |
| **Icon  Description automatically generated** | **NAT** | Translation of DB private IP to enable access to DB subnets |
|  | **DB** | PRIVATE |
|  | **WEB** | PUBLIC |
|  | **IAM** | Identity Access Management: secure access |
|  | **GLOBAL ACELERATOR** | reate accelerators to improve the performance of your applications for local and global users |
|  | **S3** | Simple Storage Service: buckets (can use IAM policies) |

**Network Layout:**

**Timeline

Description automatically generated with medium confidence**

**PART 1:** VPC WEB

1. **Create the public VPC (VPC-web).**
2. **Create subnet web:** choose CIDR with part of the range of the VPC.
3. **Create RT:** associate with subnet-web.
4. **Create IGW:** connect to VPC-web and add new route IGW to rt-web.
5. **Create SG:** this case allows HTTP and SSH traffic from/to anywhere.
6. **Create an ENI:** behind SG created and allocate EIP (this step is only if you need to keep same IP address)
7. **Initiate EC2 instance:** connect with instance

**Text

Description automatically generated**

1. **Ping to verify connection:**

**Text

Description automatically generated**

**PART 2:** VPC SHARED DB

1. **Create private VPC (shared-vpc)**
2. **Create a subnet for DB:** CIDR with part of the range of VPC.
3. **Create RT shared-vpc:** associate with subnet-db.
4. **Create Ec2 instance for DB:** choose IP inside subnet range.
5. **Create peering between 2 VPCs.**

**PART 3:** NAT (secure egress to DB)

1. **Create IGW:** attach to VPC shared.
2. **Create NAT Subnet:** associate with subnet NAT **(**diff range of IP of subnet-db but also inside VPC range.)
3. **Create RT NAT Subnet:** new route for IGW shared VPC.
4. **Create ec2 NAT instance:** in networking turn of change source/ destination (new sg)
5. **Associate EIP to subnet NAT.**
6. **Create Routes for IGW and peering on rt-vpc-shared.**
7. **Test connection with ec2 NAT:**

**Text

Description automatically generated**