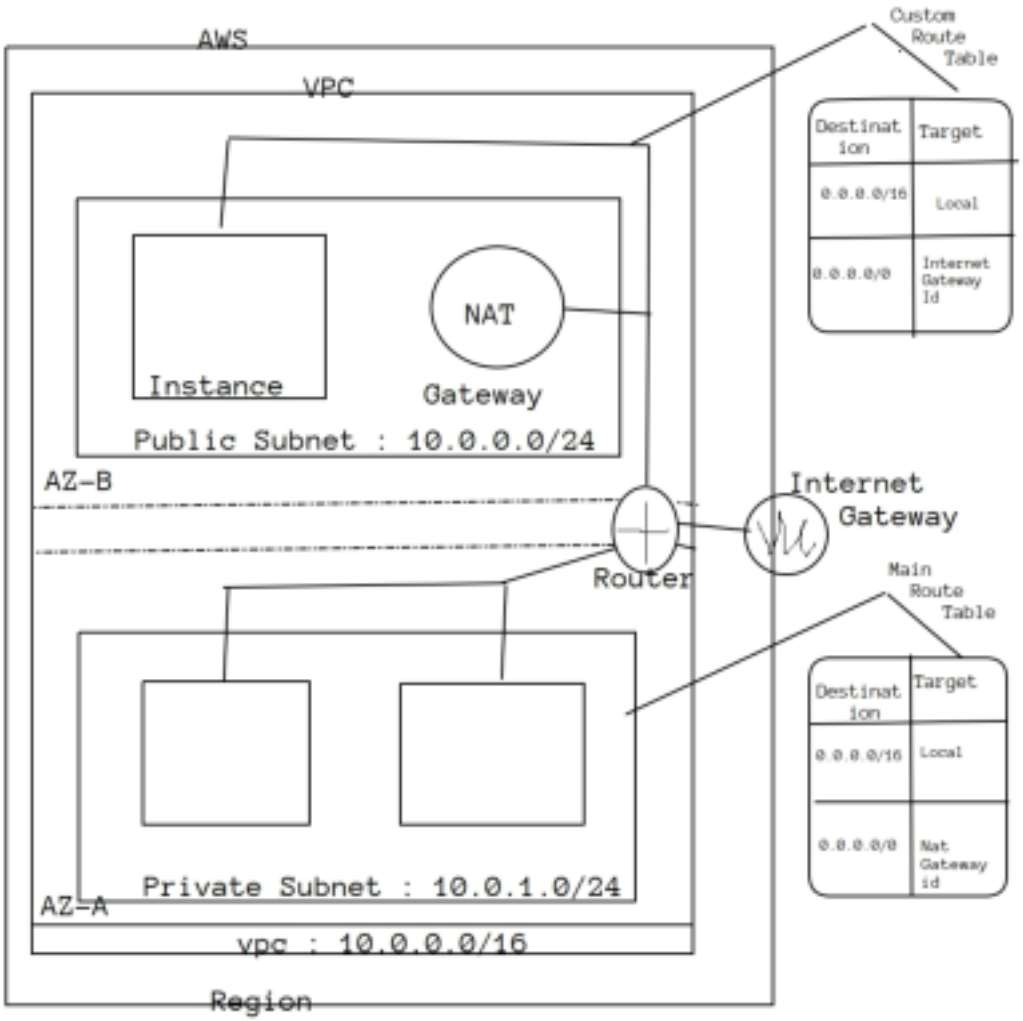
Session 1 : Intoduction to Virtual Private Cloud



Virtual Private Cloud :

• A vpc is a virtual network that closely resembles a traditional networking that we operate in our own datacentre,with the benefits of using the scalable infrastructure of AWS
• To Simply say vpc is a virtual network or datacenter inside aws for one client
• It is logically isolated from other virtual n/w in the aws cloud
• Max 5 vpc can be created inside one region and 200 subnets in 1 vpc
• We can allocate max 5 elastic ip's
• Once we created a VPC, dhcp,nacl and security group will be automatically created
• A vpc is confined to an aws region and does not extend between regions

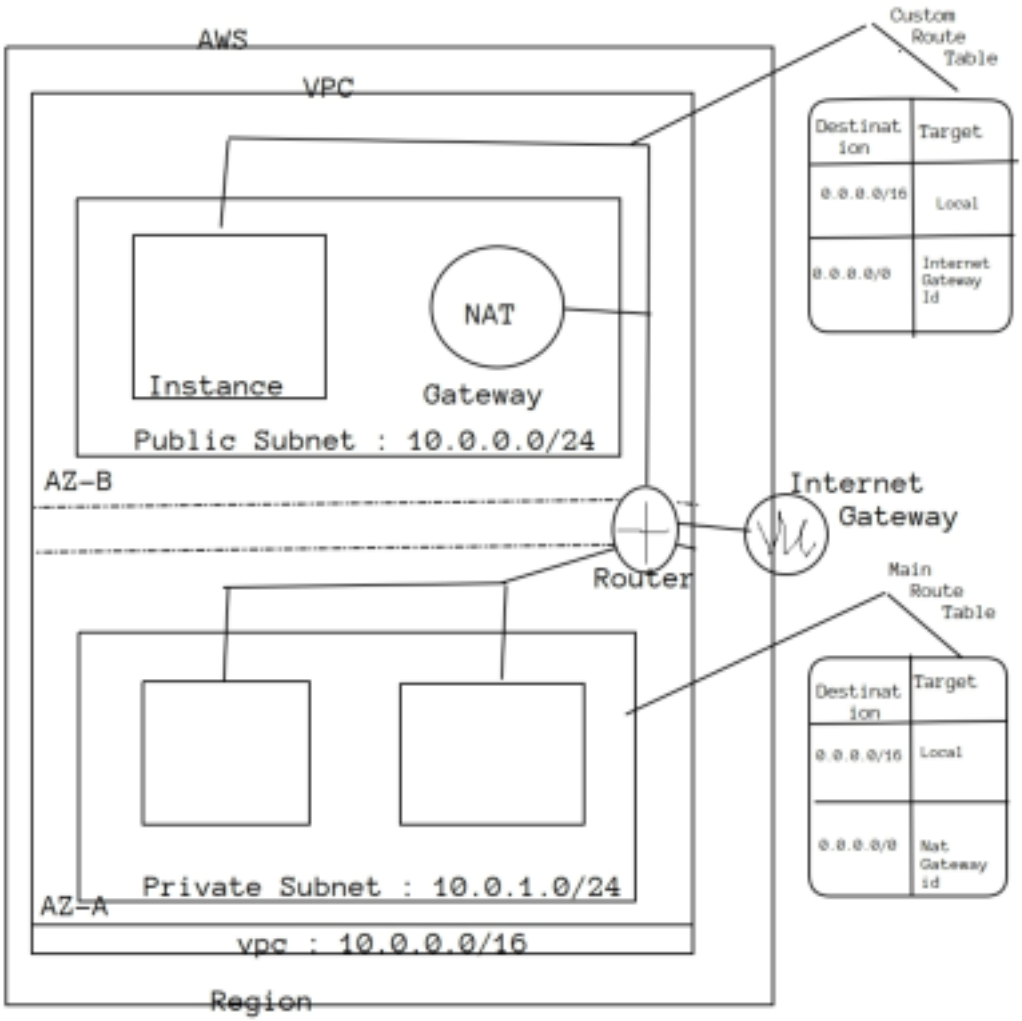
• Once the vpc is created,we cannot change its CIDR,block range
• If you need a diff cidr size,create a new vpc
• The diff subnets within a vpc cannot overlap
• We can however expand our vpc cidr by adding new /extra ip address ranges(except American gov cloud & AWS China)

Components of VPC :

CIDR & IP Address Subnets,
• Implied router & routing table
• Internet gateway
• Security groups
• Network ACL
• Virtual private gateway
• Peering connections

• Elastic ip

Session 2 : Types of VPC,Public & Private Subnets



VPC Types : Defaut VPC,Custom VPC

Default VPC :

• Created in each aws region when an aws account is created
• Has default cidr,security group,NACL and route table settings
• Has an internet gateway by default

Custom VPC :

• Is an aws account admin creates
• AWS user creating custom vpc can decide the CIDR,
• Has its own default security group,network acl, and route table
• Does not have an internet gateway by default,one needs to be created when needed

Steps to follow to create a VPC :
1.Create VPC
2.Subnet
3.Internet Gateway
4.Route Table

Public Subnet :

• If a subnets traffic is routed to an internet gateway,the subnet is known as public subnet

• If we want our instance in a public subnet to communicate with the internet over ipv4,it must have a public ipv4 addr or an elastic IP addr

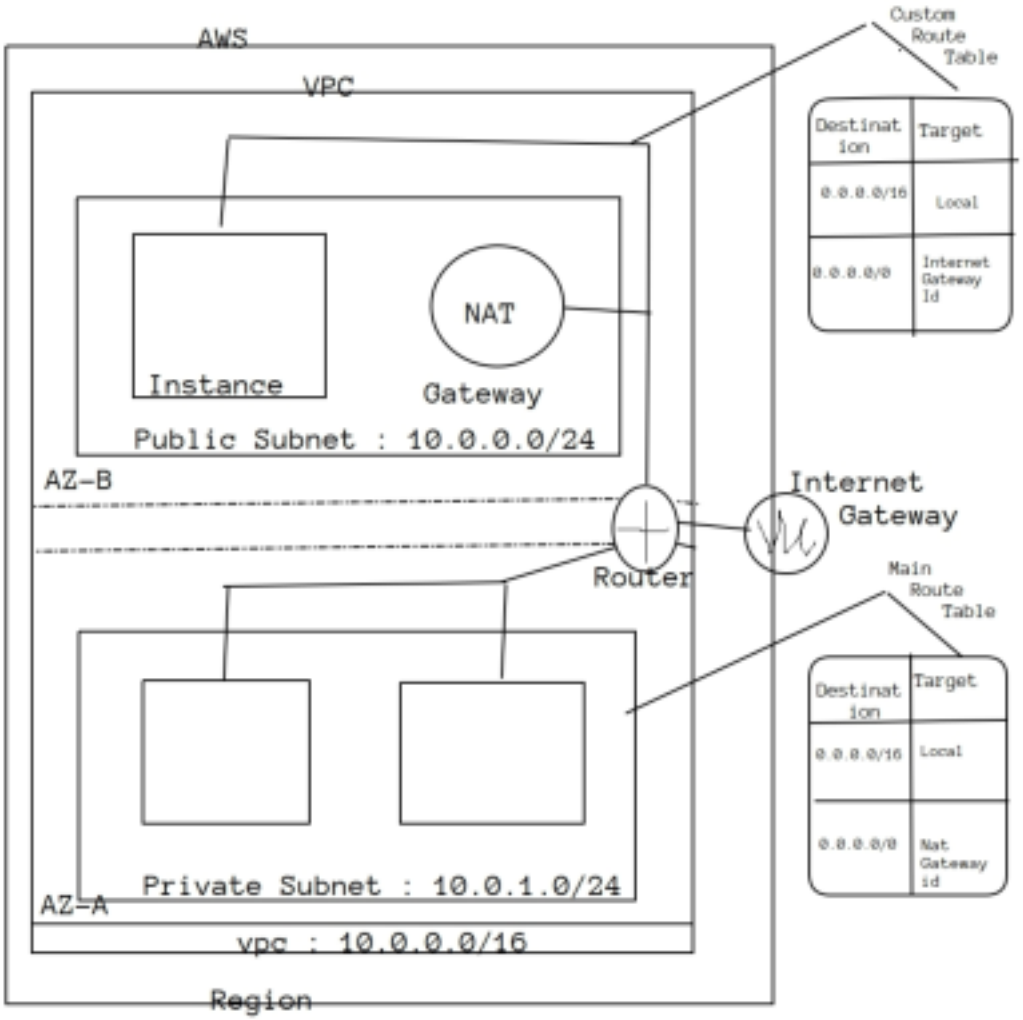
Private Subnet :

• If a subnet does not have a route to the internet gateway,the subnet is Known as a private subnet

Note : When we create a vpc,we must specify an ipv4 cidr block for the vpc.The allowed block size is between /16 to /28 and the first four & last ip addr of a subnet cannot be assigned

Eg : 10.0.0.0/24 addrs following are reserved as follows:
10.0.0.0 ------>Network Addr
10.0.0.1-------->reserved by aws for the vpc router
10.0.0.2--------->reserved by aws,The ip addr of dns server
10.0.0.3-------->reserved for future use
10.0.0.25------->broadcast addr
• Aws does not support broadcast in a vpc,but reserves the address

Session 3 : Implied Router,Route Table,Internet Gateway



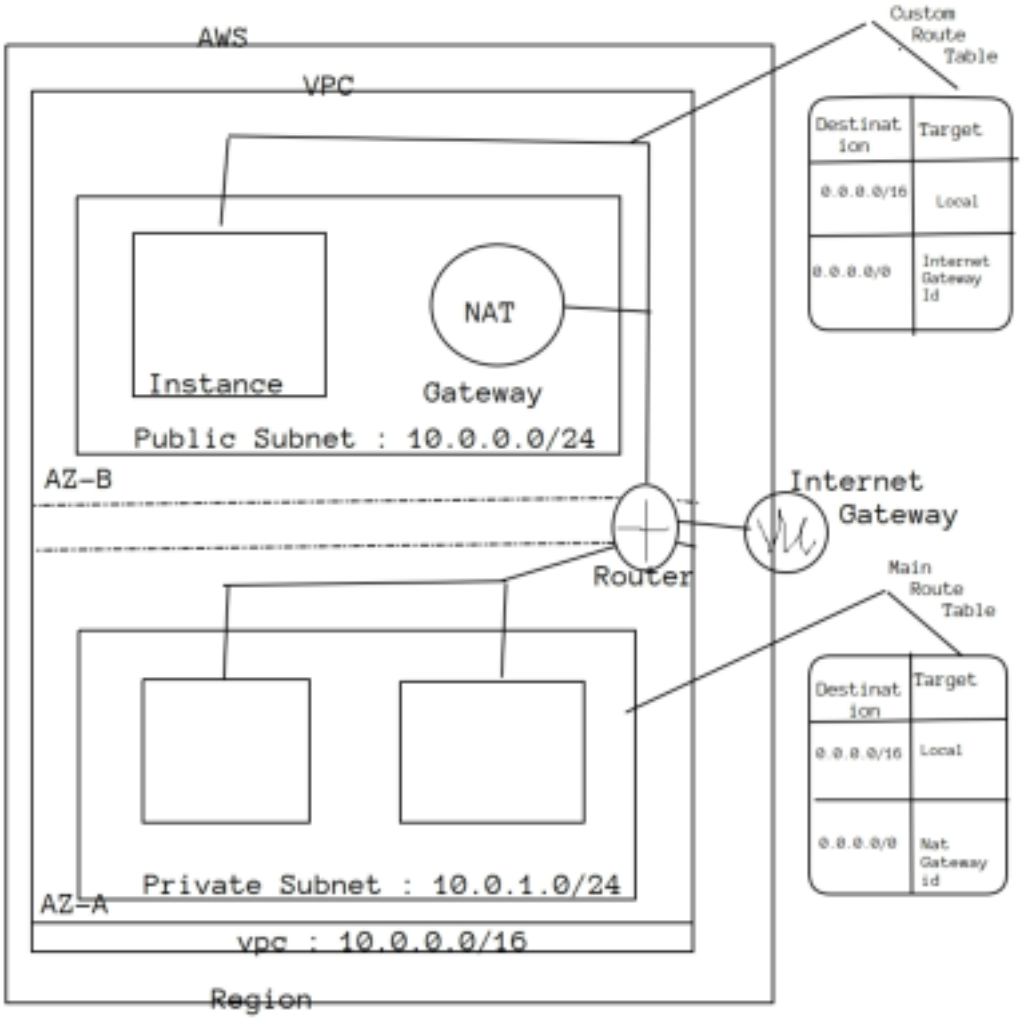
Implied Router & Route Table :

• It is the central routing function
• It connects the different AZ together and connects the vpc to the internet gateway
• We can have upto 200 route tables per vpc
• we can have upto 50 route entries per route table

• Each subnet must be associated with only one route table at any given time
• If we do not specify a subnet to route table association,the subnet will be associated with the default vpc route table
• We can also edit the main route table if we need,but we cannot delete main route table
• However we can make a custom route table manually, make it the main route table then we can delete the former main,as it is no longer a main route table
• We can associate multiple subnets with the same route table

InterNet Gateway : IGW
• An IGW is virtual router that connects a vpc to the internet
• Default vpc is already attached with an IGW
• If we create a new vpc then we must attach the igw in order to access the internet
• Ensure that our subnet's route table points to the internet gateway
• It performs nat between our private and public ipv4 addrs
• It supports both ipv4 and ipv6

Session 4 : NAT Gateway,Security Group,NACL,VPC Peering



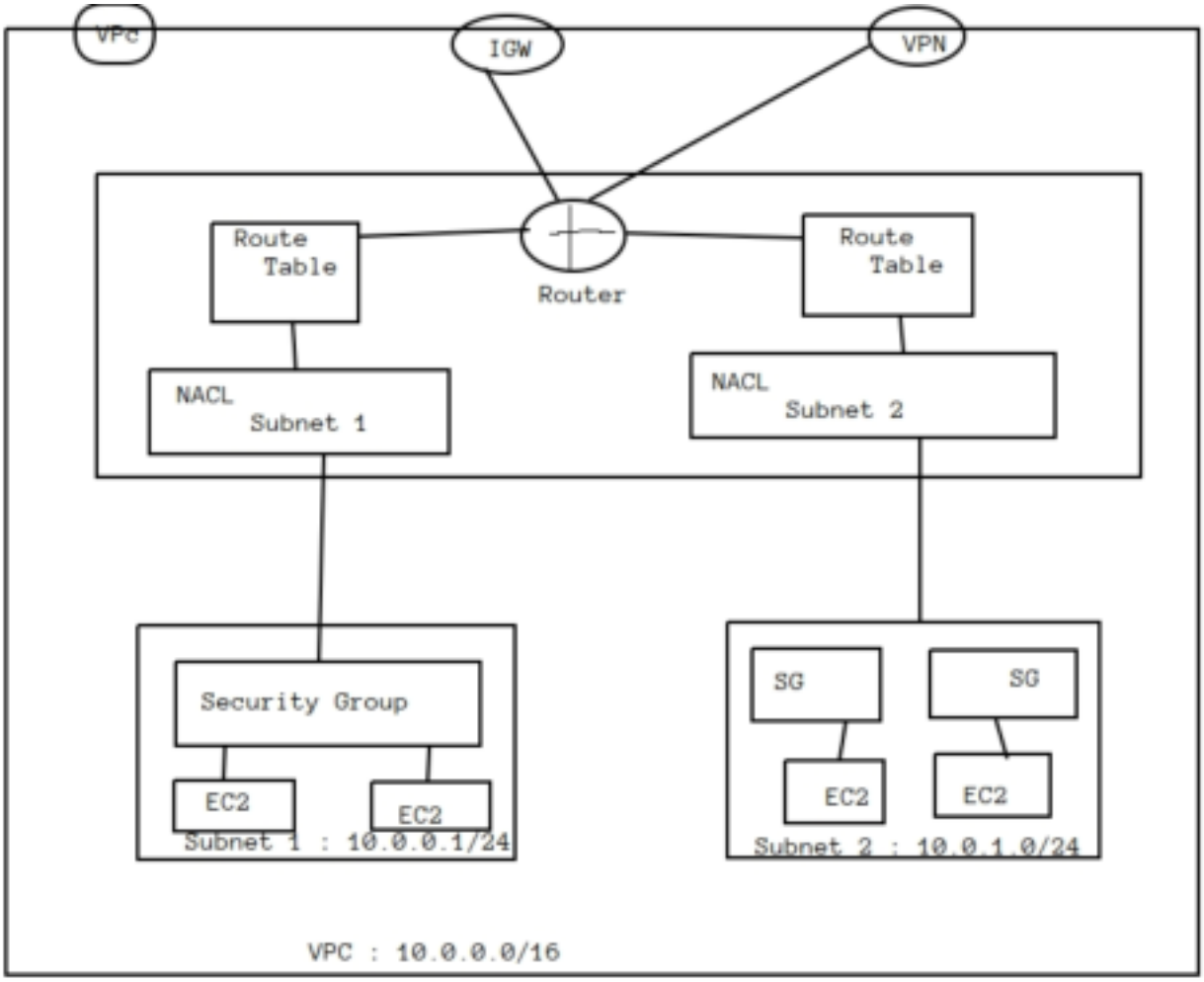
NAT Gateway : Also does PAT(Port Address translation)

• we can use a network address translation gateway to enable instances in a private subnet to connect to the internet or other aws services,but prevent the internet from initiating a connection with those instances
• We are charged for creating and using nat gateway in our account.NAT gateway hourly usage and data purchase rates apply.Amazon ec2 charges for data transfer also apply
• To create a nat gateway,we must specify the public subnet in which nat gateway reside
• We must also specify an elastic ip address to associate with nat gateway when we create it

• No need to assign public ip's to our private instances
• After we have created a nat gateway we must update the route table associated with one or more of our private subnets to point internet bound traffic to the nat gateway
• This enables instances in your private subnets to communicate with the internet
• Deleting a nat gateway,disassociates its elastic ip addr,but does not releases the address from your account

Security Groups :

• It is a virtual firewall works at ENI(Elastic Network Interface) level
• Upto 5 security gropus per ec2 instances interface can be applied
• Can only have permit rules,cannot have deny rule
• Stateful(if imbound allowed then automatically outbound is also allowed and vice versa) : return traffic is allowed then inbound traffic is also allowed,even if there are no rules to allow it



Network ACL :

• It is a function performed on the implied router
• NACL is an optional layer of security for our vpc that acts as a firewall for controlling traffic in and out of one or more subnets
• Our vpc automatically comes with a modifiable default network acl.By default, it allows all inbound and outbound ipv4 traffic and if applicable,ipv6 traffic
• We can create a custom network,acl and associate it with a subnet
• By default each custom network acl denies all inbound and outbound traffic untill we add rules
• Each subnet in your vpc must be associated with a network acl.if we dont explicitly associate a subnet with a network acl,the subnet is automatically associated with the default network acl
• we can associate a network acl with multiple subnet,however a subnet can be associated with only one network acl at a time.When we associate a network acl with a subnet,the previous association is removed
• A network acl contains a numbered list of rules that we evaluate in order,starting with the

lowest numbered rule
• The highest number that we can use for a rule is 32766.Recommended that we start by creating rules with rule numbered that are multiples of 100,so that we can insert new rules where you need later
• It functions at the subnet level
• NACl are stateless,outbound traffic for an allowed inbound traffic,must be explicitly allowed too
• We can have permit and deny rules in a NACL

Diff between Security Groups & NACL :

• Security group Operate at instance level and NACL oprates at subnet level
• SG support allows rules only and NACl permits allow as well as deny rules
• SG is stateful,return traffic is automatically allowed and NACl is stateless,return traffic must be explicitly allowed by rules
• SG applies to an instance only and NACl applies to all instances in this subnet

VPC Peering :

• A vpc peering connection is a networking connection between two vpc that enables us 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
• We can create a vpc peering connection between our own vpc, or with a vpc in another aws account.The vpc can be in diff region
• Transitive peering is not possible i.e if vpc-A peers with vpc-B and vpc-B peers with Vpc-C,but by default vpc-A is not peered with vpc-C

LABS :

Session 5 : Creating VPC,Subnets,Route Table,IGW

Session 6 : Access Internet Inside Private Subnet Using NAT Gateway

Session 7 : Establishing VPC to VPC Peering and Testing connection between two VPC's

Session 8 : Establishing connection between 2 VPC's in 2 Diff region

Session 9 : Network ACL Inside VPC

Session 10 : VPC Endpoint

VPC Endpoint : A vpc endpoint enables us to privately connect our vpc to supported aws services.Instances in our vpc do not require public ip address to communicate with resources in the services
• Endpoint is a virtual device

Session 11 : VPN Connection

Session 12 : Configuring NAT instance for Private Subnets & Internet Access

• We can use a NAT instance in a public subnet in our VPC to enable instances in the private subnet to initiate outbound IPv4 traffic to the internet or other AWS services,but prevent the instances from receiving inbound traffic initiated by someone on the internet

Note : NAT is not supported for ipv6 traffic -use an egress only internet gateway

Session 13 : Virtual Private Gateway,Customer Gateway & Site-to-Site VPN connection

• By default,instances that we launch into an Amazon VPC cant communicate with our own(our corporate or home network) Network.To enable the communication we have to establish site to site VPN connection

VPN Connection : A secure connection between our on-premises equipment and our VPC's

VPN Tunnel : An encrypted link where data can pass from the customer network to or from AWS.Each VPN connection include two VPN tunnels which we can simultaneously use for high availability

Customer Gateway : An AWS resource which provides information to AWS about our customer gateway device

Customer Gateway Device : A physical or software app on customer side

STEPS :

• Create two VPC's-One in Mumbai and another in Singapore(customer end)
• Create one linux machine in both the VPC,take RDP of it(Security Group-SSH,TCP,ICMP)
• Now go to mumbai region --Create Virtual Private Gateway
• Now Create customer gateway---Enter Public IP of Singapore EC2 instance
• Create Site to site VPN Connection,Add subnet of customer end
• Now go to route tables ----Route Propagation
• Site to site VPN----Download Configuration
• Now go to singapore region take access of ec2 using putty
• Then do the following :

Commands :

GO TO PUTTY,CONNECT TO OUR EC2 LOGIN AS-ec2-user

1. Commands for Installation of Openswan
   i. Change to root user: $ sudo su
  ii. Install openswan: $ yum install openswan -y
iii. In /etc/ipsec.conf
   uncomment following line if not already uncommented:
         include /etc/ipsec.d/*.conf
iv. Update /etc/sysctl.conf to have following
    net.ipv4.ip_forward = 1
     net.ipv4.conf.all.accept_redirects = 0
     net.ipv4.conf.all.send_redirects = 0
 v. Restart network service: $ service network restart

   2. Command for /etc/ipsec.d/aws-vpn.conf
   conn Tunnel1
   authby=secret
   auto=start
   left=%defaultroute
    leftid=Customer end Gateway VPN public IP
    right=AWS Virtual private gateway ID- public IP
    type=tunnel
    ikelifetime=8h
    keylife=1h
    phase2alg=aes128-sha1;modp1024
    ike=aes128-sha1;modp1024
    keyingtries=%forever
    keyexchange=ike
    leftsubnet=Customer end VPN CIDR
    rightsubnet=AWS end VPN CIDR
    dpddelay=10
    dpdtimeout=30

dpdaction=restart_by_peer

3. Contents for /etc/ipsec.d/aws-vpn.secrets
customer_public_ip aws_vgw_public_ip: PSK "shared secret"

4. Commands to enable/start ipsec service
$ chkconfig ipsec on
$ service ipsec start
$ service ipsec status

END .....