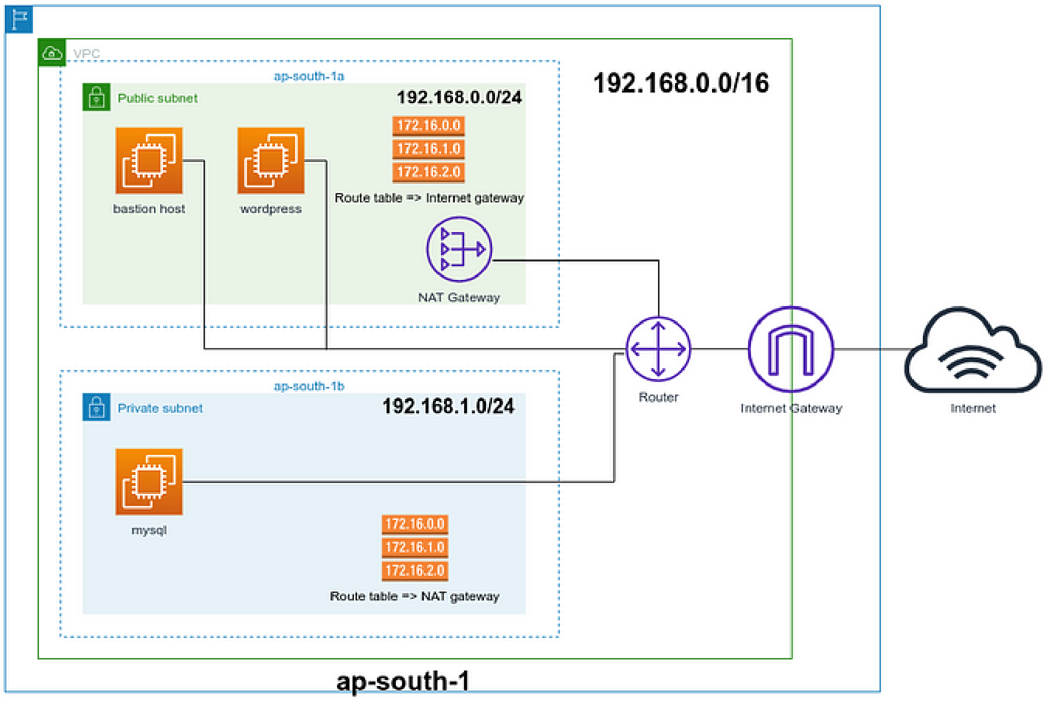
A **Virtual Private Cloud** (VPC) is a secure, isolated private cloud hosted within a public cloud like AWS, Azure, Google etc. In VPC users can get all the computing resources isolated form other users, users can host websites, and do anything else they could do in an ordinary private cloud, but the private cloud is hosted remotely by a public cloud provider.



Components of Virtual Private Cloud

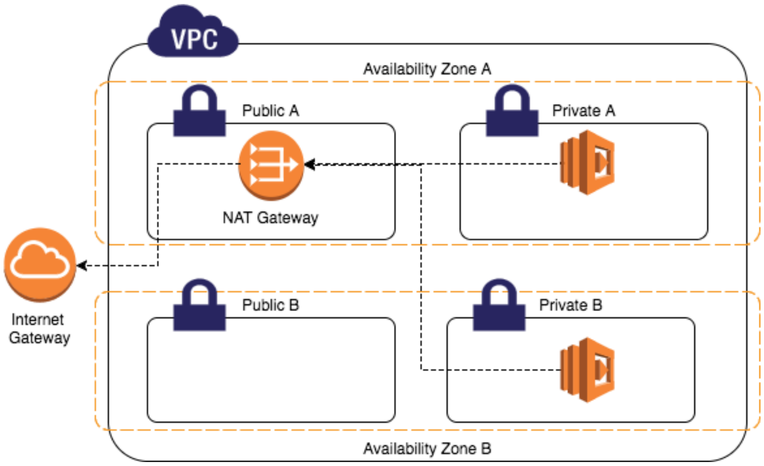
Subnets in VPC are the logical portion of VPC. This allows users to isolate resources and control access to those resources. Subnets can be either **public or private**, with public subnets having access to the internet and private subnets not having access to the internet. In a VPC, user must specify a range of IPv4 addresses for the VPC in the form of a Classless Inter-Domain Routing (CIDR) block; for example, 10.0.0.0/16, which is the primary CIDR block for your VPC. After creating a VPC, VPC of 10.0.0.0/16 is divided into multiple subnets across different Availability Zones.

In Private Subnet traffic originated from the instances are not allowed directly to the internet instead it will be routed through NAT Gateway. Once the subnet is created, the user can then add resources to the subnet, such as EC2 instances or RDS databases.

Route Table contains a set of rules, called routes, that are used to determine where network traffic from your subnet or gateway is directed. Two types of Route Tables one is Main Route Table that automatically comes when you create VPC and another is Custom Route Table that you create for your VPC. Main Route Table controls the routing for all subnets that are not explicitly associated with any other route table.

Internet gateway is a horizontally scaled, redundant, and highly available VPC component that allows communication between VPC and the internet. IGW is a target for internet-bound traffic from VPC Route Table, IGW performs network address translation (NAT) for instances that have been assigned public IPv4 addresses. ***IGW needs to be associated with public subnet***

Network Address Translation (NAT) gateway that enables instances in a private subnet to connect to the internet or other AWS services but prevents the internet from initiating a connection with those instances. NAT Gateway should always be launched in Public Subent.



Security group acts as a virtual firewall for your instance to control inbound and outbound traffic. When you launch an instance in a VPC, you can assign up to five security groups to the instance. Security groups act at the instance level, not the subnet level. Each instance in a subnet in VPC can be assigned to a different set of security groups.

Create VPC→ Create IGW → Create Subnet → Route table and assign RT to Public VPC → Edit RT and assign 0.0.0.0/0 to access internet & assign created IGW and Save changes → In RT Subnet association → Save Association → Create EC2 instance (after selecting AMI, instance type and key pair) → Edit Network settings and specify the resources you have created and choose Public IP →Enable Auto-assign public IP → Security Group (Type SSH and Source type Anywhare) → Launch Instance → Click on instance ID (to SSH into EC2 instance from internet) →copy public IP → Go to terminal on Desktop→ Change permission chmod –R 400 keypair Name → paste IP address on terminal

Video for reference: https://www.youtube.com/watch?v=43tIX7901Gs

Step1. Go to AWS management console and navigate to the VPC console.
Step2. Click your vpc's option
Step3.Click create VPC , assign a CIDR 10.1.0.0/16 and click create VPC.
Step4. Create two subnets, one for public access and the other for private access. The public subnet will contain the NAT instance, so that the private subnet can talk to internet.

Step5. Click the subnet option in the left pane and create two subnets, one at a time, using the create subnet option. Create two subnets in two different availability zones. For public subnet , use 10.1.1.0/24 subnet and for private 10.1.2.0/24 subnet. Make sure you select the appropriate VPC id. In our case, VPC with 10.1.0.0/16 CIDR block.

Step6. Go to Internet Gateway option in the left pane, and create one using the create option. Once created, right click it and associate it with your VPC.

Step7. Go to route tables option on the left pane , click create route table, select your VPC and click create.

Step8. Now we have to associate the route table to the subnet which has to be made public. Select the route table and associate it with your subnet which has to be made public. Route tables entry should have **0.0.0.0/0** as destination and internet gateway as target. Now you will be able to connect the ec2 instances launched in the public subnet (10.1.1.0/24) using the public ip or an elstic ip.

Step9. La