**Basic VPC Setting Up in AWS**

In this project we will learn how we can set up our own Virtual Private Cloud in AWS.

Till now we were using the default VPC Provided By AWS. To create EC2 Instance, Now we will Create our own VPC and create EC2 Instance in that VPC.

Steps to create VPC: -

* Create PVC
* IP Address
* Subnet
* INT Gateway
* Route Table

**Architecture:-**

A screenshot of a computer

Description automatically generated

Services Used are EC2, Subnet, VPC, Internet Gateway, Route Table.

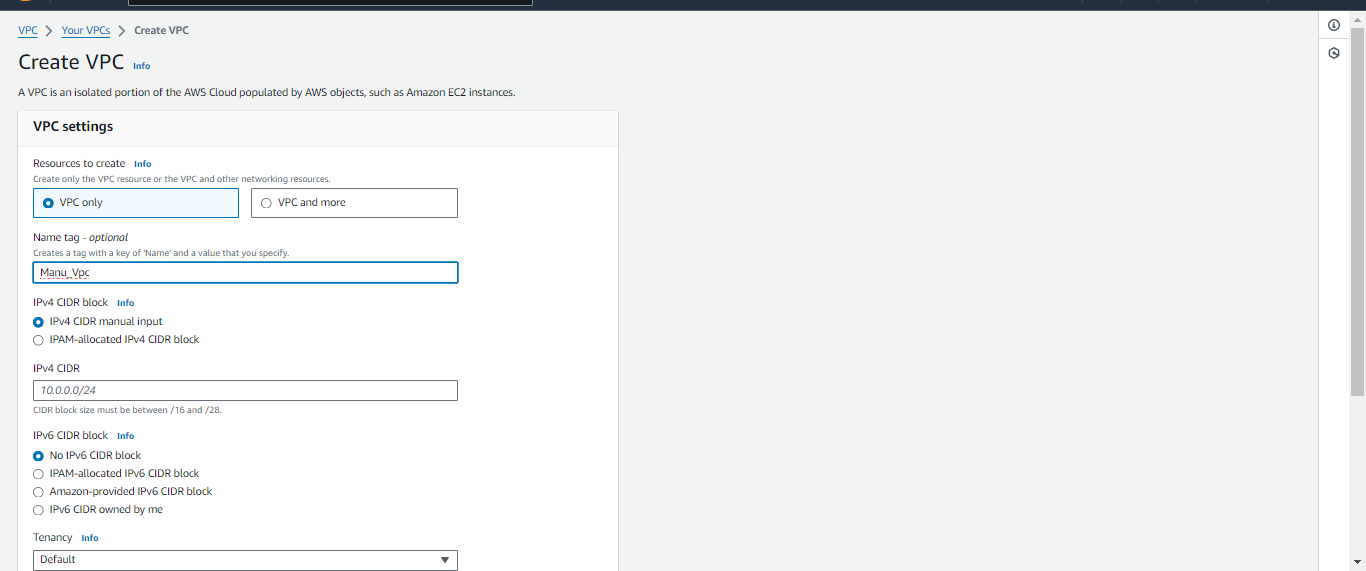
1. **CREATE VPC**

Navigate to VPC from AWS Management Console.

A screenshot of a computer

Description automatically generated

Give Name for your VPC.



1. **IP Address**

When we are working with VPC Mostly we use Private IP Address for communication within VPC, but Sometimes we use Public IP Address as well.

Private IP Address

Class A 🡪 10.0.0.0 to 10.255.255.255 /8

Class B 🡪 172.16.0.0 to 172.32.255.255 /16

Class C 🡪 192.168.0.0 to 192.168.255.255 /24

We are using Class C

A screenshot of a computer

Description automatically generated

Click on Create VPC

A screenshot of a computer

Description automatically generated

VPC is ready.

1. **Subnets**

Now we must create Subnets,

Navigate to Subnet from VPC Dashboard.

A screenshot of a computer

Description automatically generated

Click on Create Subnet

Select the above created VPC and give Name for Subnets

A screenshot of a computer

Description automatically generated

Choose the AZ where we need to create the Subnet. And enter the IPv4 CIDR Block

Then click on **create subnet.**

A screenshot of a computer

Description automatically generated

Now we can create EC2 Instance within this subnet easily. But it will be only available in one AZ. Because we created subnet only in One AZ as of now.(AP-south-1a).

So, we will create Subnets in other Availability Zones. Steps are same as above only difference we need to create it in diff rent AZ than above.

We can’t create another subnet using 192.168.0.0/24 as of now because we already assigned all the Ips 192.168.0.0 to Subnet-1.

Now What to do………………!!!!!!!??????

**Solution is :-**

* We can add one more IP series in VPC CIDR. As of now we have 192.168.0.0/24 we need to add 192.168.1.0/24. **But the Problem is when we are doing EDIT CIDR we can add only 5 IP Series to a VPC.**
* **Another Solution is** we will create multiple subnets using single IP address range using subnet mask.

Delete the created Subnet.

Now we will create Subnet by dividing The IP range using Subnet Mask.

Navigate to Subnets🡪Create Subnet🡪 choose VPC🡪 Give Name

A screenshot of a computer

Description automatically generated

Note that we have given CIDR as 192.168.0.0/25

A white rectangular object with a black border

Description automatically generated

Click on Create Subnet.

Now create the second subnet in different AZ(AP-south-1b)

Now use CIDR as 192.168.0.128/25 this means that IP Address starts from 192.168.0.128 for Second subnet.

A screenshot of a computer

Description automatically generated

Created Subnets can be visible in subnets Dashboard.

A screenshot of a computer

Description automatically generated

Only 126 hosts per subnet as two IP addresses of each subnet is reserved.

One IP reserved for router/network id. Another IP reserved for broadcast/DNS.

Now we can create EC2 Instances in Both AZ.

Now we will Create two EC2 Instances in 2 Subnets, (as we are aware How to Create EC2 Instances not covered in this Documents).

Before Creating Second EC2 Instance in Subnet 2 Enable auto assign IP Settings of Subnet 2, by navigating VPC🡪Subnet🡪choose subnet🡪 Edit Subnet Settings.

A screenshot of a computer

Description automatically generated

Now Create the EC2 Instance

Two EC2 Instances are created in Two Subnets.

A screenshot of a computer

Description automatically generated

Accessing created Instance using public IP address.

A computer screen with white text

Description automatically generated

I tried to access both the EC2 instance using respective Public IP address Not able to access the EC2 instance because our VPC is not having Internet Connection. Therefore, the above error message.

Solution is we need to add Internet Gateway to VPC.

1. **Internet Gateway**

Navigate to AWS🡪 VPC Dashboard🡪 Internet Gateway

Click on Create Internet Gateway

A screenshot of a computer

Description automatically generated

Give Name then Click on Create Internet Gateway.

A screenshot of a computer

Description automatically generated

Now the Internet Gateway is created but the state is in Detached Mode.

We will attach this to our VPC. Click on Actions🡪attach to VPC🡪Select the VPC🡪 click on Attach IG

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Now internet Gateway is attached.

Now try to access the EC2 Instance.

A computer screen with text on it

Description automatically generated

Still, we are not Able to access our EC2 Instance, because we are not yet routed the network traffic to our EC2 instance.

1. **Route Table**

Navigate to VPC Dashboard🡪Route tables.

A screenshot of a computer

Description automatically generated

Here we can see the first route table created for our VPC we just need to add a route for internet gateway (For any traffic coming from EC2 instances forward it to Internet Gateway).

Select the Route table🡪routes🡪 Edit routes. We need to add one route here.

A screenshot of a computer

Description automatically generated

Click on Save Changes.

The route is active.

A screenshot of a computer

Description automatically generated

**Now try accessing the EC2 Instances.**

A screenshot of a computer screen

Description automatically generated  
  
Now we can access EC2 Instances in both Subnets.

We got internet connectivity. To our EC2 Instances.

**Some Basic Questions**

* **Can we Access the EC2 in Subnet 2 from EC2 in Subnet 1 using private addresses.**

**Answer is YES,** two subnets relate to the router so they can communicate with each other using private IP addresses.

* **Can we Access EC2 instance inside a Subnet which don’t have any IP address**

**Answer is NO**