**Knowledge Transfer( Public and Private Subnet Setup )**

**Dated : 08/08/2023**

**\*\*Challenge :\*\***

Hey geeks, today I want to give a task that establish a Public and Private Subnet Setup which involves configuring two distinct subnets with differing connectivity requirements—public access through an Internet Gateway and controlled private access through a NAT Gateway.

==============================================================================

**\*\*Task :\*\***

Ticket: Give the subnets Private and Public Internet Connections.

==============================================================================

**\*\*Solution :\*\***

Finally I have got the solution by clearly defining the purpose of each subnet: public for internet-facing resources and private for internal services. Configure separate route tables for each subnet, directing public subnet traffic to the Internet Gateway and private subnet traffic to the NAT Gateway.

==============================================================================

**\*\*Pre-requisites:\*\***

• Have a good knowledge about VPC

• Know the all operations of subnets

• Know IG and NAT deeply

==============================================================================

**\*\*Objective:\*\***

The objective is to establish a clear and secure Public and Private Subnet Setup, enabling controlled internet access via an Internet Gateway and secure outbound connectivity through a NAT Gateway while mitigating configuration complexities

.==============================================================================

**\*\*Description:\*\***

A Public and Private Subnet Setup is a network architecture design within a Virtual Private Cloud (VPC) that establishes two distinct subnets, each with specific connectivity characteristics. In this configuration, the Public Subnet is designed to host resources that require direct access to the internet, such as web servers or external-facing applications. This subnet is associated with an Internet Gateway, allowing instances within it to have public IP addresses and bidirectional communication with the broader internet.

The Public and Private Subnet Setup offers several advantages, including enhanced security by isolating sensitive resources, improved control over outbound communication from the Private Subnet, and efficient use of resources by segregating public-facing and internal workloads. This architecture allows organizations to strike a balance between accessibility and security, making it a fundamental approach in cloud networking for building versatile and well-protected environments.

==============================================================================

**\*\*Steps:\*\***

# Login to your AWS Account and search the VPC Service and come to VPC Dashboard

**Step 1 ( Creating a VPC )**

# Firstly on VPC Dashboard you will see Virtual Private Cloud Section you will see Your VPC option Click it and follow the down steps....

* In your VPC click create VPC
* Name your VPC
* Then in IPv4 CIDR give the IP Code
* Lastly click create VPC

**Step 2 ( Creating a Subnet )**

# After creating VPC go to subnets in Virtual Private Cloud Section for creating subnets…

# There will be default subnets don’t touch it or delete it…

* Click create subnet
* Select the VPC in VPC ID

# Then down there will be subnet section there you can create the subnet …

* Give subnet name
* Then give the availability zone

# Note : Always create 2 subnet of same name in different zone so that if anything happen to one zone you will have backup of another zone…

* Then give the IPv4 CIDR block

# While giving the CIDR change one character in the IP address Example if first subnet IP is 10.0.0.0/24

Then for second give 10.0.1.0/24 likewise change till the subnets are over…

# After creating the main subnets lastly create a public subnet named as per your choice but commonly public subnet is named as BastionHost…

# This will help you to understand which Public and which is Private after that…

* Click create subnet

**Step 3 (Configure Route Tables )**

# After creating subnets now we will create the route table…

# There will be default route tables don’t touch it or delete it…

* Then click create route table
* Name the first route table as igrt i.e. internet gate way route table and
* Then give the VPC
* Then click create route table

# Now again repeat the step and create the second route table for NAT gatway

* Name the second route table as natrt i.e. Network Address Translation gate way route table
* Then give the VPC
* Then click create route table

**Step 4 ( Create Internet and NAT Gateway )**

# Now the next step is to create internet and NAT gateway

* Click on internet gateway
* Click on create internet gateway
* Name the gateway
* Then click on create internet gateway

# After creating the gateway upside you will see Attach VPC click on it if not then follow the steps…

* Click on actions
* Then click on attach VPC
* Then in available VPC click your created VPC
* After that click Attach Internet Gateway

# Now your Internet Gateway is created next is creating of NAT Gateway

# Go to NAT Gateway in Virtual Private Cloud above peering connection

* Click on NAT gateway
* Name the gateway
* Now select the anyone private subnet

# Note : Don’t select the bastionhost because it is public subnet select only private subnet

# That’s why always create a public subnet which you can recognize easily

# Don’t change the connectivity type be it in public…

* Then click Allocate Elastic IP

# Elastic IP is to stable the private IP and it will not change time to time…

# Note : Remember to Delete the Elastic IP in 5 mints otherwise AWS will charge it…

* Then click on create NAT gateway

**Step 5 ( Giving Network to IG and NAT gateway through Route Table )**

# Now go to Route Table again and follow the steps…

* Click on the internet gateway route
* Then click actions
* Then click on edit route
* Then click on add routeThen in destination select 0.0.0.0/0

# By selecting this means user can access it from anywhere

* Then in target select Internet Gateway

# After selecting it will show the IG created select it

* Then finally click on save changes

# Now the next step is to give the internet to Public subnet

* click on actions
* Then click on edit subnet association

# Then it will all the available subnet but only select the public subnet which is bastionhost

* Then click on save associations

# Now again go to route table and repeat the steps for NAT gate way…

* Click on the NAT gateway route
* Then click actions
* Then click on edit route
* Then click on add route
* Then in destination select 0.0.0.0/0

# By selecting this means user can access it from anywhere

* Then in target select NAT Gateway

# After selecting it will show the NAT created select it

* Then finally click on save changes

# Now the next step is to give the internet to Private subnets

* Click on actions
* Then click on edit subnet association

# Then it will all the available subnet here only select the private subnets

* Then click on save associations

Step 6 ( Creating EC2 instances to check the IPs )

# This is the Final Step to check the Setup is created correctly or not..

# Go to EC2 from the search bar and leave the VPC

# Now we are going to create 2 Instance one for Public and another Private…

**-----CREATING PUBLIC INSTANCE-----**

* Click on launch instance in instances

# First we are going to create Public Instance

* Give the instance name

# Select the AMI and Instance type as Free Tier..

* Then generate the key pair

# Note : Always generate new key pair

# Now do changes in network setting as follows...

* In network setting click on edit
* Now select the VPC you have created

# After selecting VPC it will show the subnets form the list select the public one

* Then enable the auto-assign public IP

# This will indicate that the this instance is PUBLIC

* Then click on launch instance

# Now your PUBLIC INSTANCE is created

**-----CREATING PRIVATE INSTANCE-----**

* Click on launch instance in instances

# Now we are going to create Private Instance

* Give the instance name

# Select the AMI and Instance type as Free Tier..

* Then generate the key pair

# Note : Always generate new key pair don’t use the public key pair

# Now do changes in network setting as follows...

* In network setting click on edit
* Now select the VPC you have created

# After selecting VPC it will show the subnets form the list select the private one

* Then disable the auto-assign public IP

# This will indicate that the this instance is PRIVATE

* Then click on launch instance

# Now your PRIVATE INSTANCE is created

**Step 7 ( Checking the IPs of Public and Private Instances )**

# Now click the Public Instance ID and open the summary you will see both PUBLIC IP4V ADDRESS AND PRIVATE IP4V ADDRESS this means the public subnet is successfully launched…

# Now open the Private Instance ID and open the summary you will see only PRIVATE IP4V ADDRESS this means the private subnet is successfully launched…

**##### Your PUBLIC AND PRIVATE SUBNET SETUP is completed here #####**

# NOTE : AFTER DOING THE PRACTICAL DELETE ALL THE SUBNETS, VPCs, INTERNET GATEWAY,NAT GATEWAY, ELASTIC IP, ROUTE TABLE AND INSTANCES.

==============================================================================

**\*\*Explanation:\*\***

A Public and Private Subnet Setup is a network configuration strategy commonly employed in cloud environments to establish a well-defined and secure networking environment within a Virtual Private Cloud (VPC). In this setup, the VPC is divided into two distinct subnets, each catering to specific connectivity requirements and security considerations.

The Public Subnet, as the name suggests, is designed to host resources that need direct internet accessibility. This could include web servers, load balancers, or any services that interact with external users. Instances in the Public Subnet are assigned public IP addresses, enabling them to communicate with the broader internet via an Internet Gateway. While offering ease of access, this subnet demands careful security measures through the configuration of security groups and access control lists to regulate inbound and outbound traffic, mitigating potential risks.

On the other hand, the Private Subnet is engineered to house sensitive or internal resources that don't necessitate direct internet connections. Examples could range from databases and application servers to internal APIs. Instances within the Private Subnet do not possess public IP addresses, rendering them less exposed to external threats. Instead of granting direct internet access, outbound communication is facilitated through a Network Address Translation (NAT) Gateway or NAT instance located in the Public Subnet. This approach ensures secure outbound connectivity while maintaining the isolation of private resources. Just like the Public Subnet, security groups and network ACLs govern the traffic flow within the Private Subnet, bolstering its resilience against potential breaches.