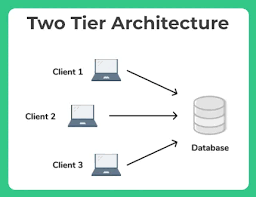
**Implementing 2-Tier Architecture in VPC**

**Two-tier Architecture**

In this lab we will implement the two-tier architecture using Virtual Private Cloud Services of AWS.



Above Diagram represents the two-tier architecture, usually we use two tier architecture to include layer of security or to hide confidential data from user.

Consider the Scenario of Web Server and Data Base server of a website. Database server must be hidden from normal user in this case we use two tier architecture in which we make web server to accessible to any one and Database server is only accessible to admin in laymen terms we can say that data base server is private whereas web server is public.

Here we don’t want Database server to be accessible by anyone over the internet, instead users will access webserver over the internet and webserver will manipulate database server. (webserver makes changes to DB server for only authorized requests).

**How to Implement this in the AWS Cloud…?**

**Solution** is AWS VPC multi-tier Architecture.

We will create a VPC with multiple AZ. And we will create a Public Subnet and a private subnet in each Availability Zone. We use public subnet for web server and private subnet for Database Server. We create subnets in two AZ because we need high Availability.

Architecture:-

A screenshot of a computer

Description automatically generated

Architecture Explanation:-

**STEPS:-**

1. **Creating VPC and 4 Subnets ( 2 Private and 2 Public Subnets)**

As we already know how to create VPC and Subnets I have already created them.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Here we can see I have created 4 Subnets out of which two subnets are public and having auto public IP assign enabled.

A screenshot of a computer

Description automatically generated

Two Private subnets are created in two different AZ. Similarly, two public subnets are created in two different AZ.

1. **Creation Of EC2 Instance in Public Subnet 1 (subnet in AP-South-1A)**

We name this instance as Web\_server\_1A.

Choose the subnet.

A screenshot of a computer

Description automatically generated

Create Security Group

A screenshot of a computer

Description automatically generated

Click on Launch Instance.

Instance created And accessing the instance.

A screenshot of a computer

Description automatically generated

Successfully accessed the EC2 Instance.

1. **Creating and accessing EC2 instance in Private Subnet 1 (Private Subnet in AP-South-1a)**

A screenshot of a computer

Description automatically generated

Auto assign public IP is disabled as it is Private Instance.

A screenshot of a computer

Description automatically generated

Private EC2 Instance is Created.

Try to access this instance using private IP.

A computer screen with white text

Description automatically generated

Now How to access this private instance.

**The first way is EC2 Instance Connect Endpoint (new feature).**

Navigate to VPC 🡪endpoints🡪 Create End Point.

A screenshot of a computer

Description automatically generated

Give name, select our VPC, Private Subnet, and default Security group.

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

Click on Create End Point.

A screenshot of a computer

Description automatically generated

Created Endpoint.

Now Navigate to EC2 Dashboard and Select the EC2 Instance. click on Connect.

A screenshot of a computer

Description automatically generated

Now Click on Connect.

Here we don’t need to enter password because we already logged in as root user.

A screenshot of a computer

Description automatically generated

Now Successfully connected.

**The Second way is to connect to EC2 Instance using Bastion Host.**

In this Scenario I will connect to Public instance(Bastion Host) first then we will access private instance from bastion host using private IP address.

A black screen with a black background

Description automatically generated

We already in bastion host, connecting to private ec2 instance.

As of now we don’t have the key file inside bastion host we are copying the key file from our system to bastion host.

A black screen with white text

Description automatically generated

Using the above command.

Now got Bastion Host and verify whether we have the Key file or Not.

A screenshot of a computer

Description automatically generated

Successfully copied the key file

Now accessing the private instance.

A computer screen shot of a computer screen

Description automatically generated

Permission denied because of permission error of .pem file

A computer screen with white text

Description automatically generated

.pem file should have read only access

A computer screen with numbers and symbols

Description automatically generated

Now we have only Read only access.

Trying again to access the EC2 Instance.

A computer screen with white text

Description automatically generated

HERE NOTE that the ip address got changed to private IP address. Indicates that we are in Private EC2 Instance.

Use Exit to Return back to Bastion Host.

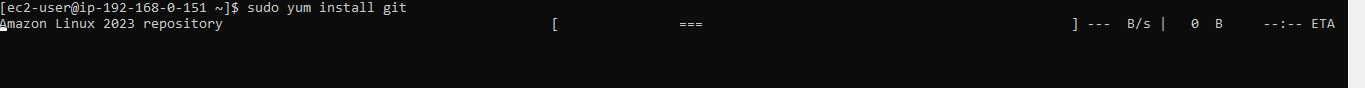
This is How we can access the private EC2 Instance. We must provide .pem file to the private EC2 instance.

But **Problem** is Private EC2 Instance don’t have internet Connection.

Trying ping in Private EC2 Instance.

A black screen with white text

Description automatically generated



We don’t have the internet so we can’t download any OS, DB Server, no updates possible we need internet connection.

**Here NOTE that we want Outbound internet in Private EC2, we don’t need inbound internet**.

Means we can access the internet from Private EC2, reverse is not possible that is Internet cannot access the Private EC2 Instance.

**NAT GATEWAY** comes to Rescue us from the situation.

NAT Gateway is a component which will provide outbound internet to private EC2 instance. By connecting to the Internet gateway.

NAT Gateway always resides in Public Subnet.

NAT Gateway can’t reside in private subnet because **the route table of private subnet don’t have route to internet.**

Second is **Internet gateway can only communicate with resources having public IP address only**, we created EC2 instance in Private subnet without Public IP.

NAT Gateway is placed as shown in above architecture diagram.

1. **Creation of NAT Gateway**

Navigate to VPC🡪NAT Gateway.

A screenshot of a computer

Description automatically generated

Click on Create NAT Gateway.

A screenshot of a computer

Description automatically generated

Allocate Elastic IP address and click on Create NAT Gateway.

**A screenshot of a computer

Description automatically generated**

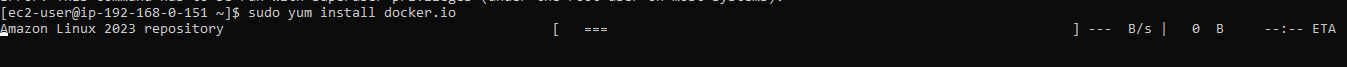
A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

NAT Gateway is available.



Still the Internet is still not available because we have not added route table.

Navigate to main route table.

A screenshot of a computer

Description automatically generated

Added Internet Gateway 0.0.0.0/0 in Main Route table which is used by private subnet.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

Now we are getting outbound internet to the Private EC2 instances as well.

A screen shot of a computer

Description automatically generated

**SUCCESSFULLY FINISHED LAB.**

**Summary:-**

In this Lab we created two tier architecture using VPC. We created 4 subnets and created EC2 instances in one private and one public subnet, created internet gateway for public subnet so that it can access internet and verified whether private subnet have internet access or not, later we setup NAT gateway for outbound internet access to the private EC2 instances.