**CREATE AN EC2 INSTANCES:**

Whenever we create an EC2 instance, we must create a key pair to access this instance.

Once this been created, we must SSH into this instance to access remote computer/instance securely.

**Key-pair:** Amazon **AWS** uses **keys** to encrypt and decrypt login information. At the basic level, a sender uses a public **key** to encrypt data, which its receiver then decrypts using another private **key**. These two **keys**, public and private, are known as a **key pair.**

**Public key stored by AWS and private key is stored by user. Its been created once we create EC2 instance first time and later it can be used with other instances as well.**

**We are normally allowed to connect to EC2 instance (only for Amazon Linux2) with an option “EC2 Instance Connect (browser-based SSH connection)”. This is because of in our default created security group it allows traffic from all IPs. (If we delete inbound rule, it will not allow)**

To access this instance (in windows **PowerShell**), along with public ip of instance we have to use this pem file using below command.

***ssh -i myFirstKeyPair.pem*** [***ec2-user@13.233.141.17***](mailto:ec2-user@13.233.141.17)

For first time login, we will get a warning

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

@ WARNING: UNPROTECTED PRIVATE KEY FILE! @

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

Permissions for 'myFirstKeyPair.pem' are too open.

To fix this issue, we must change the permission of .pem file from file properties.

Delete all the other users and create a new user for NTNET ashvansh. Give full permission to it.

Once we connect to EC2 instance, we can use it as a linux system and use it accordingly. For example, we want to install Apache server.

First change the permission as root user:

***sudo su***

su: switch user

sudo su: run switch user with root as user

To update the machine (Linux instance)

***yum update -y***

To install httpd

***yum install -y httpd .x86\_64***

To start a service:

***systemctl start httpd.service***

To Configure Apache to Start on Boot.

We configure Apache service at boot, so when a server gets a reboot, they **start automatically.**

There could be various reasons for server reboot, as Unexpected due to hardware/kernel issue

***systemctl enable httpd.service***

To access Apache server

***curl localhost:80***

*To add any content for the home page of apache server, we can add it as :*

***echo ”Hello word” > /var/www/html/index.html***

**CREATING APPLICATION LOAD BALANCER:**

We can create a load balancer to divide the load on different instances.

This should be created on the port on which we are having our outer traffic as port 80 in our example.

We can choose old security group or create a new one.

Create a target group for this ALB for health check and other uses.

We have to attach an Instance to this, which’s load we want to divide.

Once this LB created, we can use its DNS name instead of directly accessing the public IP of our instances. This DNS name need to access like any other website URL (no need to give any other port, IP details further).

**Route 53**

We need out DNS to hit the services and at this point we already have an instance, a LB for it and a target group.

First create a hosted zone: It will ask for Domain name so need to register a Domain name first.

While launching Auto Scaling Group, we can give our own configuration. Below mentioned data can be passed as User data under Advanced Details as text. It will start a tomcat server.

#!/bin/bash

#install httpd

yum update -y

yum install -y httpd.x86\_64

systemctl start httd.service

systemctl enable httpd.service

echo "Hello ashvani from my second EC2 instance from $(hostname -f)" > /var/www/html/index.html

#!/bin/bash

yum update -y  
yum install -y httpd.x86\_64  
systemctl start httpd.service  
systemctl enable httpd.service  
cd /var/www/html  
echo "Hello Ashvani, this is an EC2 instance running on $(hostname -f)" > /var/www/html/index.html

Questions:

What is t2.micro instances?