Conception Phase

A. Modi | 13-06-2023

Description

For this course I am going to do the first task of launching a webpage online. The main task is to setup a cloud architecture first and deploy it. The solution should come with many powerful services that makes the webpage scalable and highly available.

Main Concept

We are using amazon web service to launch our webpage online. Amazon web service is a powerful cloud computing service packed with ton of features. We are using a EC2 instance to run our webpage online. EC2 stands for Elastic Compute Cloud, which is virtual server used to run various applications. EC2 is a emerging solution for businesses to use cloud computing on pay-as you basis. In EC2 instance we get predefined templates for our instance known as AMI and various configuration (including memory and storage).

The next thing we are using is EC2 auto scaling as the one key part of our task is to autoscale the backend if more visitors come across. EC2 Auto Scaling instance provides more instances to handle increase in traffic. It creates a number of instances across different Availability zones to the condition we define. AWS Auto Scaling is a free service, but extra service capacity is priced according to use.

Elastic load balancer accepts the traffic from the users and forward it to different availability zones. It always directs traffic to healthy instance better performance. Load balancer can be configured in a way that it can accept more traffic by specifying the listeners. A listener is a process that checks for connection request. It can be easily configured through port number for connections from clients to load balancer. Using a load balancer increase the availability and fault tolerance of the application. But adding the elastic load balancer significantly latency by 1ms. For reducing the latency between the EC2 instance we are connecting our resources in same availability zone and same region. All the Availability zones within a region are connected to each other in low latency links.

To automate the whole the task of cloud solution we are using terraform as it helps us to automate the task of provisioning of cloud infrastructure over AWS cloud formation. Terraform is a open source infrastructure as a code service created by harshicorp. With it there is no longer a need to go to console and set the infrastructure manually. Terraform has many notable benefits it can be deployed on multiple cloud providers. Terraform supports built-in functions that can be called and used within your code. One of the other benefits of it is we can reuse and share our terraform file with others. The core workflow consist of three stages the initialisation phase in which we define infrastructure in configurable files. The next step is plan in this step terraform create an execution plan of the infrastructure we create and finally on our approval it applies the proposed plan in correct order. The html page is accessed as the user data from the bash script every time the instance starts. The detailed UML diagram of our concept is given below to give you a better insight of our architecture.

users

Admin / IAM
User

Internet

HTTPS/HTTPS
SSH / TCP

Elastic load
Balancer

Auto Scaling group

Security group

Avaliability zone1

Avaliability zone 2

EC2 instance

EC2 instance

Terraform