MODULE 4

Nayeer Naushad

21052338

**Amazon Elastic Compute Cloud (Amazon EC2):** A web service that provides secure, resizable compute capacity in the cloud. Think of it as renting a computer in the cloud.

**Amazon Simple Storage Service (Amazon S3):** A service provided by Amazon Web Services (AWS) that stores data for users in the cloud.

**Domain Name System (DNS):**A naming system for computers, devices, and resources connected to a network.

**S3 Bucket:** A container of objects (such as images, audio files, video files, documents, and so on) in Amazon S3.

**Domain Name:** A label that identifies a network of computers under centralized control.

**Amazon Route 53:** The AWS DNS web service.

**Virtual Private Cloud (VPC):** A virtual network dedicated to your AWS account. It is logically isolated from other virtual networks in the AWS Cloud. All your AWS services can be launched from a VPC. It is useful for protecting your data and managing who can access your network.

**JavaScript Object Notation (JSON):** A syntax for storing and exchanging data.

**Dynamic Website:** A website that changes based on user interactions; often built using Python, JavaScript, PHP, or ASP with Hypertext Markup Language (HTML).

**Static Website:** A website that does not change based on user interactions; typically built using HTML and Cascading Style Sheets (CSS).

**LAB 1: Launching an EC2 instance**

* Launched an EC2 instance from the Compute menu.
* Named the instance as Web Server 1’
* Used the default AMI Amazon Linux
  + Amazon Machine Image determines the OS that will run on the EC2 instance
* Used the default instance type t2.micro
  + The Instance Type defines the hardware resources assigned to the instance. This instance type has 1 virtual central processing unit (CPU) and 1 GiB of memory.
* Selected the keypair name ‘vockey’. This keypair will allow us to access the instance through SSH
* Used the default VPC and Subnet settings
  + We can have multiple networks for various purposed like developing, testing and deployment
* Created a firewall/security group named ‘Web Server’
  + A security group acts as a virtual firewall that controls the traffic for one or more instances.When you launch an instance, you associate one or more security groups with the instance. You add rules to each security group that allow traffic to or from its associated instances. You can modify the rules for a security group at any time. The new rules are automatically applied to all instances that are associated with the security group
* Used the default storage settings
* Configured a script under Advanced settings to run on the instance when it launches
  + This bash script will run with root user permissions on the guest OS of the instance. It will run automatically when the instance launches for the first time. This bash script does the following:
    - Updates the server
    - Installs an Apache web server
    - Configured the web browser to automatically start on boot
    - Activates the web server
    - Creates a simple web page
* Launched the instance
* The instance is running on a public IPv4 address: 54.197.69.2
* The IP address was unable to load because the security is not allowing inbound traffic on port 80, (HTTP web requests)
* Network and Security > Security Groups > Web Server > Inbound Traffic > Edit Inbound Rules > Add rule
  + Type: HTTP, Source: Anywhere IPv4
* Now, we are able to access the IPv4 address.
  + It displays a simple HTML page with the text ‘Hello World!’

**LAB 2: Creating an S3 bucket**

* Services > Storage > S3
* Created a bucket named ‘myawsbucket1004’ in region ‘Asia Pacific (Mumbai)’
* myawsbucket1004 > Permissions > Bucket Policy > Edit
* Made a custom policy for my bucket (provided in the lab instructions)
* myawsbucket1004 > Objects > Upload > Add Files > index.html (provided in the lab instructions)
* myawsbucket1004 > Properties > Static website hosting > Edit > Enable
* Accessed the static website