Web App Design

[Venkata Nitin Dantu]

INDEX

1. Project Overview

This project provides an opportunity to demonstrate the solution design skills that are learned throughout the course. Assignment is to design and deploy a solution for the Social Research Organization is a (fictitious) nonprofit organization

2. Introduction

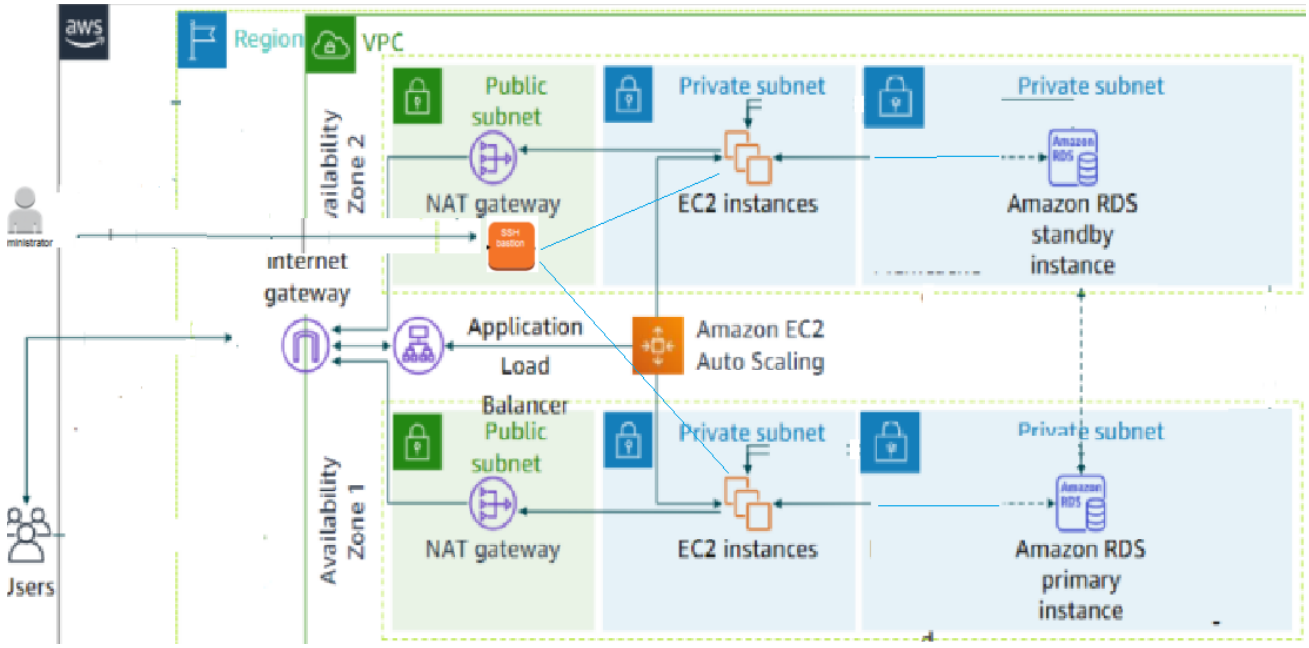
Example Social Research Organization is a (fictitious) nonprofit organization that provides a website for social science researchers to obtain global development statistics. For example, visitors to the site can look up various data, such as the life expectancy for any country in the world over the past 10 years.

Shirley approached your team to make sure that her current design follows best practices. She wants to make sure that she has a robust and secure website. One of your colleagues started the process of migrating the site to a more secure implementation, but they were reassigned to another project. Your tasks are to complete the implementation, make sure that the website is secure, and confirm that the website returns data from the query page

3. Solution requirements

Provide secure hosting of the MySQL database
Provide secure access for an administrative user
Provide anonymous access to web users
Run the website on a t2.micro EC2 instance, and provide Secure Shell (SSH) access to administrators
Provide high availability to the website through a load balancer
Store database connection information in the AWS Systems Manager Parameter Store
Provide automatic scaling that uses a launch template

4. Architecture



5. Design Decisions

Issues with the current architecture are , unable to expand capacity automatically for the increased traffic, which results in the unresponsive site and monolithic architecture results in tight coupling. We have proposed below design changes to improve performance, security and failure tolerance of the system

5.1 Enable scalability (elasticity)

To meet the demand automatically we create auto scaling group and attache it the load balancer so that we demand increases system automatically scales out EC2 instances and when there is less demand it scales in. This saves the cost as we do not need to guess the capcity and over spend on unwanted additional servers.

5.2 Choose the right database solution

As the organization is a non profit organization we do not need to maintain a team to handle routine database tasks, such as provisioning, patching, backup, recovery, failure detection, and repair. All these tasks are handled by AWS. So we have decided to use AWS RDS managed service and as the organization already using MySQL we shall continue using the same database engine on RDS.

5.3 Avoid single points of failure(HA)

To make system higly available we have decided to use Elastic Load Balancing (ELB) that automatically distributes incoming application traffic across multiple targets one or more Availability Zones (AZs). It checks the health of the webservers and diverts the traffic aromatically if any of the servers not available because of failure. This increases the availability time of the system and automatically recover from failure.

To achieve loose coupling we separate the web server and database layers and use load balancer

5.4 Secure your entire infrastructure

To improve security we separate the webserver layer and database layers in separate private subnet that are not availble to public and we provide access to the webservers to administraters using a bastion host with assymetric key encryption

We create roles so that the database password and connections string can be access from system manger store by specific roles that are attached to the EC2 instance

We create separate security groups for webservers, dbservers so that others can not access these infrastructures

Users can ananomusly access the website as we use a NAT gateway that masks the private IP of the webservers and allow trafic to reach the webservers from internet.