

Hands-On AWS Experience

AWS Fundamentals Specialization at Coursera:

Created a simple employee directory web application, a basic CRUD (Create, Read, Update, Delete) application.

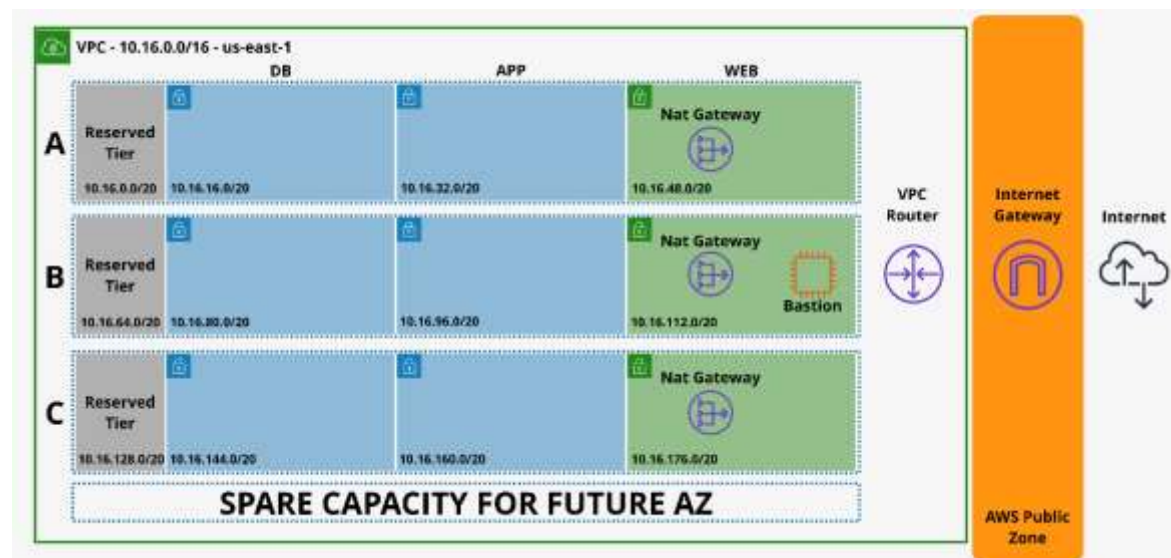
This application was built in a VPC utilizing multiple AZ's, an IGW for external access, and public and private subnets. The application's backend code was hosted on EC2, the employee data was stored in DynamoDB and images for the employees was stored using S3. I configured Amazon CloudWatch for monitoring the solution. To ensure that the application is scalable and fault-tolerant I added an Elastic Load Balancer and Amazon EC2 Auto Scaling, so that way the solution could scale out with demand and scale in when demand decreases. For security and identity, I used IAM. The root account was configured to use MFA (best practice) and a secondary admin account was created for use during the project.

Solutions Architect exam study:

Deployed EC2 instances running HTTPD to then connect to and deploy, configure, and examine how ELB, ASG, Route53, S3, and EBS services work.

Advanced Networking exam study:

Created multiple free tier accounts and then created an organization to manage them all under 1 roof, using assumed roles for the admin accounts. I secured the root and an admin user account using MFA. In my VPC, created multiple subnets to imitate traditional DB, App, and Web tiers spanning multiple AZ's to provide HA. Deployed an Internet Gateway, multiple NAT Gateways & configured custom route tables to provide internet connectivity to the public (Web) & private (DB, App) subnets.



Cloud Resume Challenge:

The goal is to create a static, S3 website to display your resume & have a working visitor counter.

To accomplish this, I used Route 53 to register my own domain (curtisfarley.net) & Cloudfront to provide SSL security (SSL cert created using AWS Certificate Manager) to the website using a custom DNS domain name. The website uses a template borrowed elsewhere, but I still needed to edit it for my own information, so that exposed me to both HTML and CSS. Also on the website is a piece of Javascript which manages the visitor counter.

This visitor counter is linked to a DynamoDB that houses the count. The data is updated and retrieved using a Lambda function running a simple Python script. The Lambda is called via an HTTP API from the embedded Javascript on the page. Both the Python and the Javascript are borrowed as the main goal of this challenge is the infrastructure, not the detailed coding, although I do understand what each piece of code is doing, as there was some trial and error to get them work as desired.

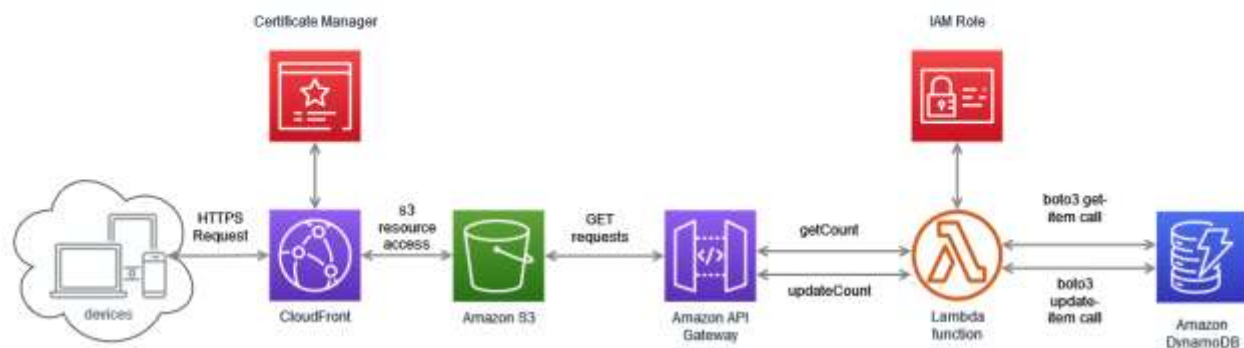
Additional parts of the challenge include using some kind of IAC (AWS SAM or Terraform) to automate the deployment of the infrastructure, and using GitHub as part of a CI/CD pipeline for code updates. Both of these I have yet to complete as my time has circled back to exam study. I have not, however, forgotten these parts & I still find & read articles/documents on SAM or Terraform so that when I am ready to go back, I am more educated on how to do the automation.

The link to the resume webpage is: <https://cfresume.curtisfarley.net>

I also created a blog about my experiences with the project which is here: <https://dev.to/curtfarley>

To get the full scope of the challenge, follow this link: <https://cloudresumechallenge.dev/docs/the-challenge/>

Cloud Resume Challenge



While none of this is as complete as using or learning these things in an actual enterprise environment, I just don't have that luxury. I have to find and do these types of things on my own. I do thoroughly enjoy learning AWS skills, and find the challenges I run across a welcome reprieve from my daily boredom. My goal with this document is to show that I do possess a level of knowledge about many different areas and that I am very good at self-directed study/training and successfully completing tasks with that new knowledge.