Blog Hosting

Title: Project 4 - Cost Efficiency
Date: October 27, 2018
Author(s): Mark Siegmund *<msiegmund@gmail.com>*
Yerden Zhursinbek *<yerden.zhursinbek.129@my.csun.edu>*
Nicholas Yoon *<nicholas.yoon.804@my.csun.edu>*
Aubrey Nigoza *<aubrey.nigoza.34@my.csun.edu>*
Reviewer(s): John Philip Vinuya *<johnphilip.vinuya.778@my.csun.edu>*
Approver(s): Aubrey Nigoza *<aubrey.nigoza.34@my.csun.edu>*
Revision Number: 5
Status: Approved

Executive Summary:

Our current blog and automation is run exclusively off AWS using the Elastic Container Service to manage our cluster, which consists of 6 instances. Primary charges from AWS largely consist of EC2 charges (documented by the hour) and make up a substantial amount of the monthly bill, ~$120+ with EC2 comprising ~$80+ of that. S3 charges are present, but cost mitigation should be applied.

Our new proposed solution will cost $5.69 per month. The system will use the combination of Cloudflare and AWS services to keep the cost down while maintaining high availability and fast and secured access to the blog website.

Goals:

- Cost effective hosting solution
- High availability
- Secured website
- Complete objectives based on Project 4 requirements

Non-goals:

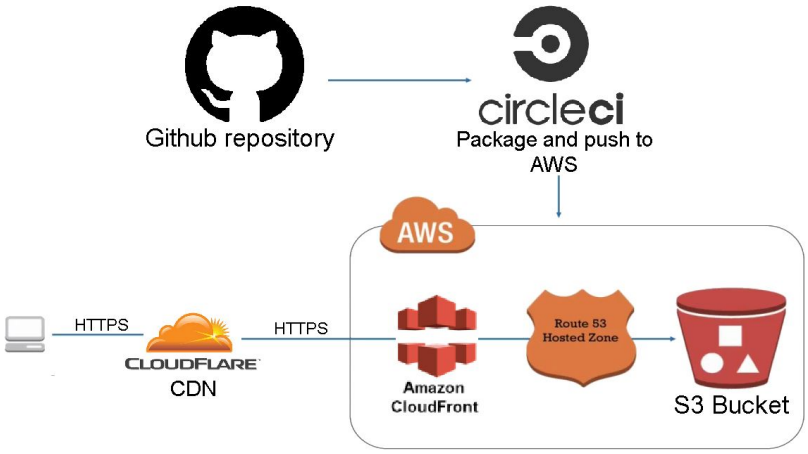
- Provide dedicated server for hosting the blog site

Background:

The blog site is the culmination of Projects 0-3 with AWS resources procured using Terraform and webserver configuration achieved with different iterations using Ansible, Packer, and ultimately Docker to employ a cluster infrastructure. Continuous deployment and integration was achieved using CircleCi, which pulls from our Github repository to package and deploy the blog's content to Amazon ECR and S3, where Lambda automates container deployment.

In this cost efficient approach, automation will still be in used. CircleCi will be used to build, test and deploy the website to S3.

High-Level Design:



CircleCi will build, test and deploy the website on every commit in the Github repository. CircleCi will deploy the static html files generated into S3. AWS Cloudfront will be used to serve the content of the S3 bucket using https. Cloudflare will use AWS Cloudfront as the source. In this way, most of the bandwidth going out to the website visitor will be thru cloudflare. Website visitors will access the website thru Cloudflare's CDN thus saving on the cost of Cloudfront.

Detailed Design:

CircleCi:
- Build, test and deploy the static html files built to an S3 Bucket.

S3:
- Static website hosting will be turned on.
- Index.html will be set as the homepage
- Set to Public
- Setup CORS to allow access via CNAME
- Encryption Enabled

Cloudfront:
- Viewer Protocol Policy will be set to redirect HTTP to HTTPS
- SSL provided by Cloudfront will be used
- Cloudfront will use https protocol to access S3 bucket.

Cloudflare:
- Cloudflare will be set to use Full SSL. This will allow https access to cloudfront and https access between the website visitors and Cloudflare.
- Setup with Free account
- Native support for HTTP2.0

Route53:
- 1 hosted zone
- www -> cloudflare
- blog -> cloudflare
- apex -> cloudflare

Metrics used for cost estimate:

- 1.8 MB frontpage
- 10.5 MB total blog post site
- 60KB per week addition
- 13 objects requested on frontpage load
- 30 days = 1 month
- 30,000 website requests per month (1,000 request per day)
- 390,000 object requests per month
- 54,000 MB -> data out without cache

Service Type	Components	Region	Component Price	Service Price
Amazon S3 Service (US East (N. Virginia))				$ 0.19
	S3 Standard Storage:	US East (N. Virginia)	$ 0.03	
	S3 Standard Other Requests:	US East (N. Virginia)	$ 0.16	
Amazon Route 53 Service				$ 0.90
	Hosted Zones:	Global	$ 0.50	
	Standard Queries:	Global	$ 0.40	
Amazon CloudFront Service				$ 4.60
	Data Transfer Out:	Global	$ 4.60	
Data Transfer Out to CloudFront				$ -
	US East (N. Virginia) Region:	Global	$ -	
AWS Support (Basic)				$ -
	Support for all AWS services:		$ -	
Cloudflare Free Tier				$ -
		Total Monthly Payment:		$ 5.69

Alternatives Considered: (alternatives considered and rejected; include pros and cons of each)

- Using different CDN to host cloud services
- Moving production instance to separate hosting service
- Reducing the number of instances used
- Using EC2 instances

Security Concerns:

- Access to the website content from the User all the way to the S3 bucket will be encrypted using SSL.
- TNO is implemented by encryption S3 storage
- Basic DDOS protection achieved via Cloudflare free tier plan
- Reduced liability by reducing the surface attack the team will be responsible for because of the “serverless setup”.