National Institute of Technology, Warangal

Department of Electronics and Communication Engineering

Cloud Computing (EC 4542)

Project

Serverless Static Website on AWS

(Research Project Blog)

VENKATNARAYAN GNANAGURUPARAN
Roll Number: 174266
Reg Number: 831777
Section: B

INDEX

INTRODUCTION

Cloud hosting enables applications and websites accessible using cloud resources. It is different from traditional hosting, where the applications and websites are deployed on a single server. Cloud hosting has a network of connected virtual and physical cloud servers that hosts the application or website, ensuring greater flexibility and scalability.

Cloud Servers can be created in seconds, easily scaled up, and deleted at ease. The cloud services are charged only for the Cloud Server utilization. These servers may also be easily scaled if needed. It allows users to maintain and configure their set-up without having to request outside help each time minor changes or upgrades are needed.

A Static website reduces complexities and focuses on improving efficiency and performance. It does not rely on databases. It mostly involves basic HTML, JavaScript, and CSS to produce light websites. These websites have no back-end systems, client-server requests, or database queries which allows these sites to have a faster performance and lesser loading time. Static websites provide more security than a dynamic website as they do not have any database to exploit. The HTML files require less size as a result, site hosting becomes more efficient.

A serverless approach for web development enables developers to build websites quickly, as developers can focus on the code and release it immediately. Serverless architectures usually have access points on a global scale, they can handle users from every corner of the world using appropriate load balancers and distribution networks.

A serverless website for a research project was developed using HTML, CSS and AWS. The information of the research project in the website is not part of this cloud computing project.

A website for the research project can encourage others to contribute, give feedback, or participate in the research activity that is carried out. The researchers and other interest groups can gain knowledge on key components, and findings/outcomes which can contribute to their work, while providing the author the recognition and also improves reputation for the valuable contribution in the industry.

AWS SERVICES UTILIZED

- Route 53
- CloudFront
- S3 (Simple Storage Service)
- ACM (AWS Certificate Manager)
- SNS (Simple Notification Service)

ARCHITECTURE

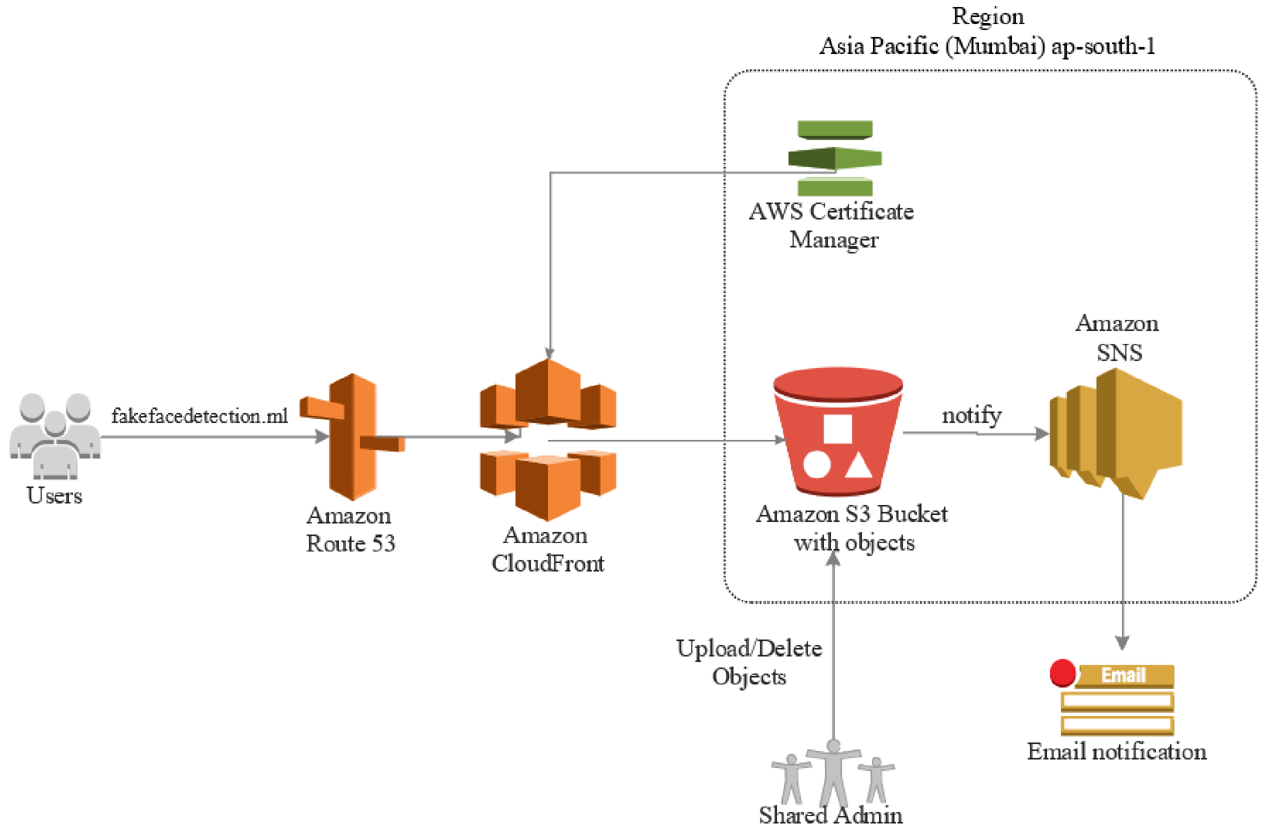


Figure 1 Research Blog Website Architecture

Route 53

Amazon Route 53 is a highly available and scalable cloud Domain Name System (DNS) web service. Amazon Route 53 effectively connects user requests to infrastructure running in AWS – such as Amazon EC2 instances, Elastic Load Balancing load balancers, or Amazon S3 buckets – and can also be used to route users to infrastructure outside of AWS. The Domain name (“fakefacedetection.ml”) was registered with Freenom.
In this Architecture, Route 53 was utilized to connect the Users with CloudFront.

CloudFront

CloudFront is a content delivery network (CDN) service that securely delivers data to users globally with low latency, high transfer speeds by caching the websites, within a developer-friendly environment. CloudFront offers field level encryption and HTTPS support that is integrated with Route 53 to protect against multiple types of attacks including network and application layer DDoS attacks.
CloudFront was used, as it works seamlessly with S3 or with any custom HTTP origin.

ACM (AWS Certificate Manager)

This service allows developers to provision, manage, and deploy public and private Secure Sockets Layer/Transport Layer Security (SSL/TLS) certificates for use with AWS services and internal connected resources. SSL protects website from phishing scams, data breaches, and many other threats. It builds a secure environment for both visitors and site developers. In this architecture AWS Certificate Manager was used to request a certificate, and deploy it on ACM-integrated Amazon CloudFront distribution.

S3 (Simple Storage Service)

S3 was used in this architecture to host the static HTML website with index document support and error document support. "index.html" was the default page that was displayed and "error.html" was used to display in the event of a partially invalid URL. All the objects necessary for the static website were uploaded on S3.

SNS (Simple Notification Service)

It is a fully managed messaging service for both application-to-application (A2A) and application-to-person (A2P) communication. It notifies the admin through E-Mail if any object was uploaded or deleted from the S3 bucket.

PIPELINE

User - Website

1. User accesses the "fakefacedetection.ml" website by typing the domain name.
2. Route 53 will route the user to CloudFront which is a content delivery network service.
3. CloudFront retrieves the content from the S3 bucket and distributes it.
4. ACM provides the SSL certification for the website to CloudFront.
5. The Website content is accessible to the User.

Admin/Developer - Website

1. Any update or deletion of objects in S3 bucket by the admin/developers.
2. S3 triggers the SNS about update of objects in bucket.
3. SNS sends an E-Mail to the admin regarding the update and deletion of Objects in S3.

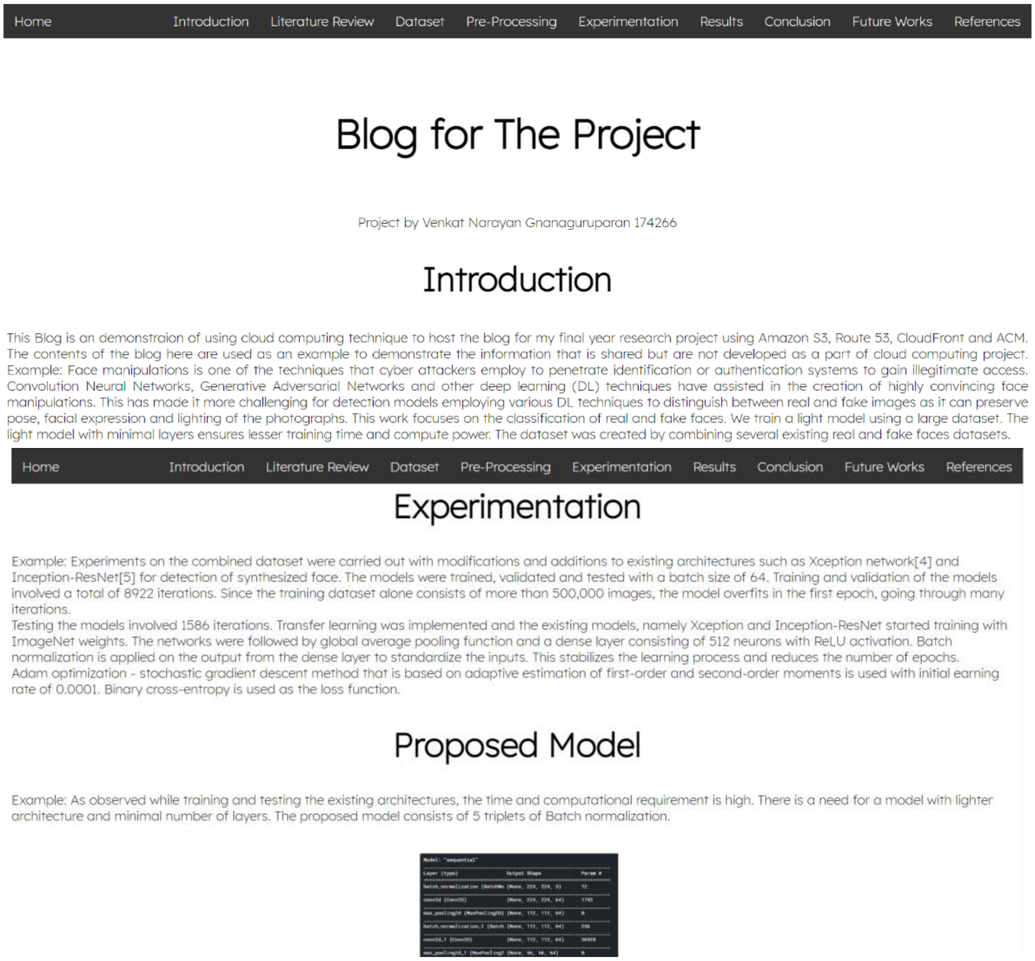
WEBSITE

The website was developed using HTML and CSS. Developing static websites provides improved security over dynamic websites but offers less functionality. It has improved performance for end users compared to sites with multiple pages. Cost of these websites is less as it only utilizes cloud storage, as opposed to a hosted environment.

This website has a single page design which contributes to an intuitive user journey. With no additional pages, users have a linear navigation flow – which provides information about the project.

Studies show that having a single page can lead to increased conversions than multi-page sites because users begin the process sooner and move through it more quickly, with nowhere to get distracted by another offer.

Having a straightforward navigation design is that there is usually only one action to be taken by the user at a time.



Blog for The Project

Project by Venkat Narayan Gnanaguruparan 174266

Introduction

This Blog is an demonstraion of using cloud computing technique to host the blog for my final year research project using Amazon S3, Route 53, CloudFront and ACM. The contents of the blog here are used as an example to demonstrate the information that is shared but are not developed as a part of cloud computing project. Example: Face manipulations is one of the techniques that cyber attackers employ to penetrate identification or authentication systems to gain illegitimate access. Convolution Neural Networks, Generative Adversarial Networks and other deep learning (DL) techniques have assisted in the creation of highly convincing face manipulations. This has made it more challenging for detection models employing various DL techniques to distinguish between real and fake images as it can preserve pose, facial expression and lighting of the photographs. This work focuses on the classification of real and fake faces. We train a light model using a large dataset. The light model with minimal layers ensures lesser training time and compute power. The dataset was created by combining several existing real and fake faces datasets.

Experimentation

Example: Experiments on the combined dataset were carried out with modifications and additions to existing architectures such as Xception network[4] and Inception-ResNet[5] for detection of synthesized face. The models were trained, validated and tested with a batch size of 64. Training and validation of the models involved a total of 8922 iterations. Since the training dataset alone consists of more than 500,000 images, the model overfits in the first epoch, going through many iterations.
Testing the models involved 1586 iterations. Transfer learning was implemented and the existing models, namely Xception and Inception-ResNet started training with ImageNet weights. The networks were followed by global average pooling function and a dense layer consisting of 512 neurons with ReLU activation. Batch normalization is applied on the output from the dense layer to standardize the inputs. This stabilizes the learning process and reduces the number of epochs.
Adam optimization - stochastic gradient descent method that is based on adaptive estimation of first-order and second-order moments is used with initial earning rate of 0.0001. Binary cross-entropy is used as the loss function.

Proposed Model

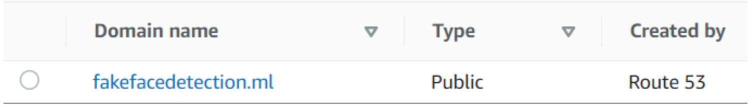
Example: As observed while training and testing the existing architectures, the time and computational requirement is high. There is a need for a model with lighter architecture and minimal number of layers. The proposed model consists of 5 triplets of Batch normalization.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
batch_normalization (BatchNo (None, 224, 224, 3)       12
_________________________________________________________________
conv2d (Conv2D)              (None, 224, 224, 64)      1792
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 112, 112, 64)      0
_________________________________________________________________
batch_normalization_1 (Batch (None, 112, 112, 64)      256
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 112, 112, 64)      36928
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 56, 56, 64)        0
```

Figure 2 Single one Page Static Website

IMPLEMENTATION

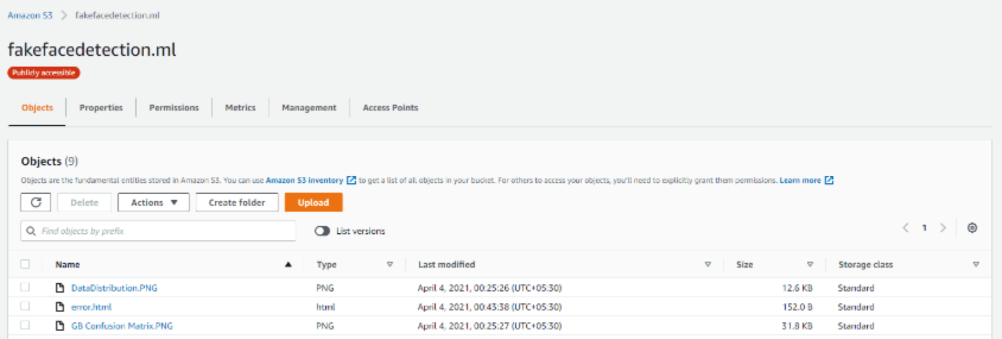
Route 53 was configured by creating a Hosted zone for DNS management with a suitable domain name. A hosted zone is a container for records, and records contain information about how you want to route traffic for the specified domain and its subdomains. A hosted zone and the corresponding domain have the same name.



	Domain name	Type	Created by
○	fakefacedetection.ml	Public	Route 53

Figure 3 Snip of Configured Route 53

Static website for the research project was developed with HTML and CSS. A new S3 bucket was created and the files of the static website was uploaded to S3 bucket as objects. The bucket was then configured for static website hosting by updating the bucket policies. An object is a file and any metadata that describes a file.



Amazon S3 > fakefacedetection.ml

fakefacedetection.ml

Publicly accessible

Objects (9)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

□	Name	Type	Last modified	Size	Storage class
□	DataDistribution.PNG	PNG	April 4, 2021, 00:25:26 (UTC+05:30)	12.6 KB	Standard
□	error.html	html	April 4, 2021, 00:43:38 (UTC+05:30)	152.0 B	Standard
□	GB Confusion Matrix.PNG	PNG	April 4, 2021, 00:25:27 (UTC+05:30)	31.8 KB	Standard

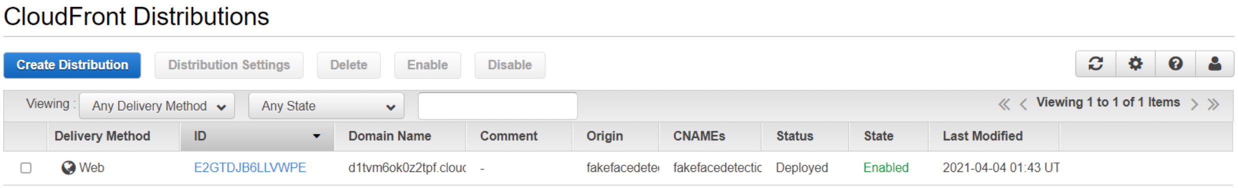
Figure 4 Snip of Configured and Updated S3 Bucket

SSL certificate for the registered domain is obtained from ACM Certificate and a record is created in Route 53.

Domain N	Record Na	Record Ty	Record Value	
fakefaced	_e19b4f8f	CNAME	_	.acm-validations.aws.

Figure 5 ACM Validation record created in Route 53

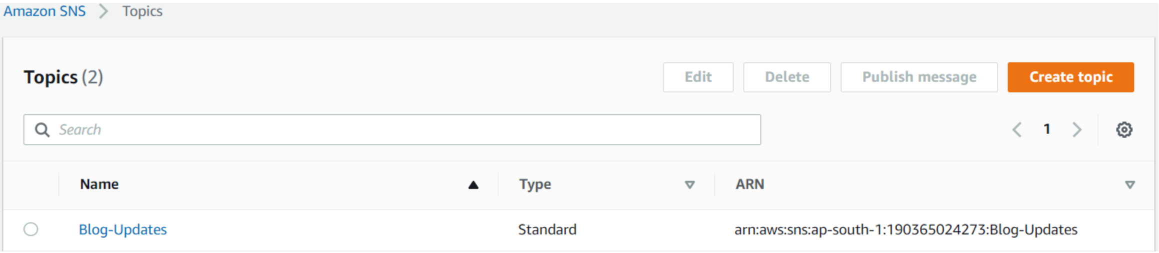
Created a CloudFront distribution and configured it with custom certificate from ACM, URL of the S3 bucket that holds the contents of the website and Route 53 with the domain name. After few minutes the distribution will be created and deployed.


CloudFront Distributions

	Delivery Method	ID	Domain Name	Comment	Origin	CNAMEs	Status	State	Last Modified
□	Web	E2GTDJB6LLVWPE	d1tvm6ok0z2tpf.clouc	-	fakefacedete	fakefacedetectic	Deployed	Enabled	2021-04-04 01:43 UT

Figure 6 Snip of the Configured CloudFront Distribution

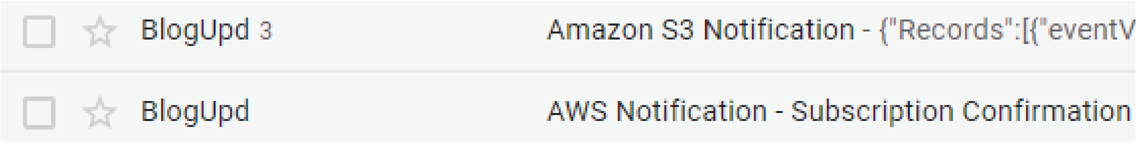
Create a SNS Topic and subscribe to the SNS Topic with the admin's E-Mail id as the endpoint, select the notifications for events that are necessary (i.e., Uploading and Deleting files from Bucket).



	Name	Type	ARN
○	Blog-Updates	Standard	arn:aws:sns:ap-south-1:190365024273:Blog-Updates

Figure 7 Confiured SNS

The SNS Topic will notify the admin through mails, if any object was created or deleted in the bucket.



□ ☆	BlogUpd 3	Amazon S3 Notification - {"Records":[{"eventV
□ ☆	BlogUpd	AWS Notification - Subscription Confirmation

Figure 7 E-Mails from SNS

ALTERNATIVE METHODS FOR HOSTING STATIC WEBSITES

WordPress, Amazon EC2 and S3

WordPress is a popular content management system due to its simplicity and is considered an inexpensive way for businesses to get started online. WordPress can be slow at times due to the added plugins, oversaturated codebases and other features that can slow down the website.

For static websites, the cost is reduced by using CloudFront and S3 rather than EC2 which charges every hour it runs. EC2 instances need to keep running to deliver content for static website. These websites don’t have the SSL certification from ACM. On S3, there is no risk of over-provisioning and hence, no need to manage disk utilization.

AWS Amplify

AWS Amplify provides fully managed hosting for static websites and web apps. Amplify hosting solution uses CloudFront and S3 to deliver the site contents via the AWS content delivery network (CDN) It requires knowledge of APIs and shell scripting.

However, if any objects in the S3 were created or deleted, it will not notify the admin as it lacks SNS feature and these websites does not have the SSL certification from ACM.

AWS LightSail

Amazon LightSail offers more than 10 images with ready to run popular software. The image with the pre-installed Nginx web server, PHP, and MySQL could be used to host a static website on Amazon LightSail. However, this would require a server even for a static website unlike the architecture developed in this project.

Google Cloud Storage and Cloud Load Balancing

The hosting environment in the Google cloud is optimized for high-end WordPress and WooCommerce based websites. Its price starts from 25$/month. Support fee is quite hefty, Amazon S3is more durable than cloud storage.

Other Services

Netlify, Surge, Render, GitHub Pages, Firebase and Cloudflare offers web hosting services which are user friendly but are not flexible with architecture. The architecture is abstracted in such services and there is a need for additional support and services from providers increasing the cost for hosting a static website. It would be relatively expensive and unjustifiable to host a single one page website on these services.

CONCLUSION

A Serverless static website was deployed with HTTPS on cloud with a suitable architecture, using services offered by AWS i.e. Route 53, CloudFront, S3 (Simple Storage Service), ACM (AWS Certificate Manager) and SNS (Simple Notification Service).

Serverless static website that was created for the research project can be accessed using the link https://fakefacedetection.ml

REFERENCES

- CloudFront Using Amazon S3 Buckets for Your Origin
  https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/DownloadDistS3AndCustomOrigins.html#concept_S3Origin

- S3 Website Hosting
  https://docs.aws.amazon.com/AmazonS3/latest/userguide/WebsiteHosting.html

- S3 Website Endpoints
  https://docs.aws.amazon.com/AmazonS3/latest/userguide/WebsiteEndpoints.html

- SNS Configuration
  https://docs.aws.amazon.com/sns/latest/dg/sns-configuring.html

- ACM for Certificate
  https://docs.aws.amazon.com/acm/latest/userguide/setup.html

- Configuring Amazon Route 53 as your DNS service
  https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/dns-configuring.html

- Amazon S3 vs. Google Cloud Storage vs. Azure pricing, features
  https://searchstorage.techtarget.com/feature/AWS-vs-Azure-vs-Google-pricing-and-features-compared

- Hosting a static website | Cloud Storage | Google Cloud
  https://cloud.google.com/storage/docs/hosting-static-website

- Hosting a Static Website on an Amazon LightSail Instance
  https://cloudbriefly.com/post/hosting-a-static-website-on-amazon-lightsail/

- Building a static serverless website using S3 and CloudFront
  https://sanderknape.com/2020/02/building-a-static-serverless-website-using-s3-cloudfront/