

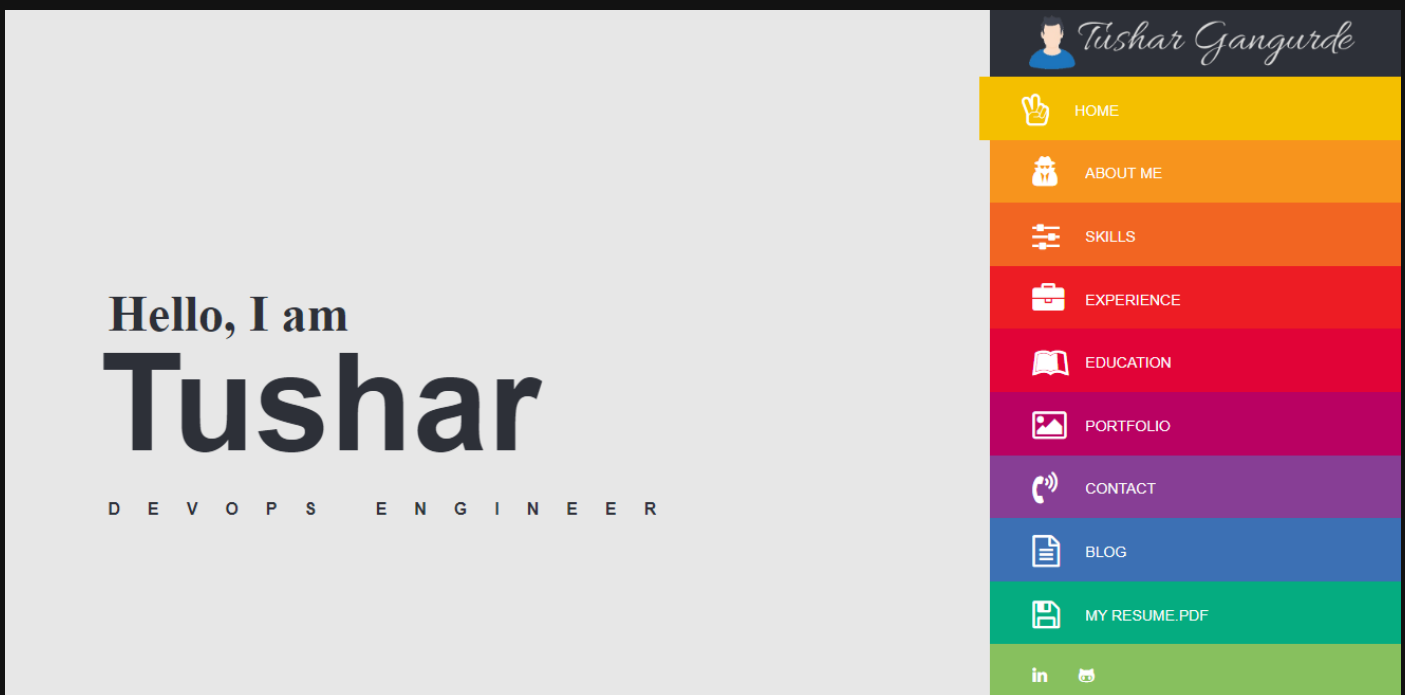
Project: Host a Static Website (RESUME) on AWS with S3, Cloud-Front, Route53

Pre-requisites:-

1. AWS Account
2. GitHub Account
3. Custom Domain
4. Basic Knowledge About CloudFront, S3, Route53, AWS Certificate Manager

Steps:

1. Create your own Resume/Portfolio Website or Clone the the sample repository created for this project.



2. Create a New S3 Bucket & Upload your Resume Website Files.
3. Enable S3 Static Website Hosting Feature
4. Connect your Domain to Route53 by Creating a Hosted Zone.

Follow For More
www.linkedin.com/in/devops-learning

5. Obtain a SSL Certificate
6. Create a CloudFront Distribution & Connect it with your S3 Bucket
7. Set Bucket Policy to Allow Cloudfront to access S3
8. Connect CloudFront to Route53 to redirect Traffic
9. Finally Visit your Website to See the Resume

Step 1:-

If you have not any Project Please use Below GitHub Repository it contains a Sample Resume Website



Note: Before Proceeding you should have your custom Domain you can buy a new domain or you can get a free domain from Freenom.com

Step 2:

Let's Create a new S3 Bucket and upload our Resume Files Go to your AWS Console & Open S3 & click on Create Bucket

The screenshot shows the 'General configuration' section of the AWS S3 'Create Bucket' wizard. It includes a 'Bucket name' field with the value 'resume-tg', a note stating 'Bucket name must be globally unique and must not contain spaces or uppercase letters. See rules for bucket naming', an 'AWS Region' dropdown menu set to 'Asia Pacific (Mumbai) ap-south-1', and a 'Copy settings from existing bucket - optional' section with a 'Choose bucket' button.

Follow For More
www.linkedin.com/in/devops-learning

Provide Bucket Name & choose Your Region

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

☒ **Block all public access**
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

For Security reasons block Public Access we are going to use cloudFront for Serve the Website

Default encryption

Automatically encrypt new objects stored in this bucket. [Learn more](#)

Server-side encryption

☐ Disable
☒ Enable

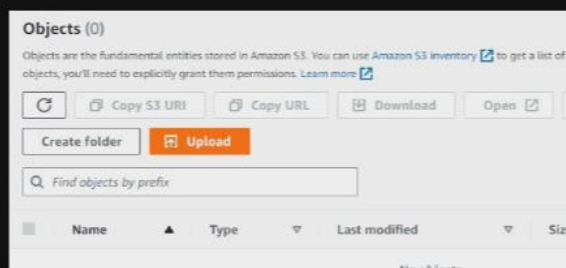
Encryption key type

To upload an object with a customer-provided encryption key (SSE-C), use the AWS CLI, AWS SDK, or Amazon S3 REST API.

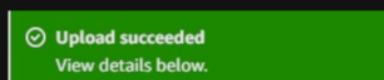
☒ Amazon S3-managed keys (SSE-S3)
An encryption key that Amazon S3 creates, manages, and uses for you. [Learn more](#)

We are using Server-Side encryption with Amazon S3-Manged Keys to reduce overhead of key management

Upload your Resume Website Files



Finally Upload your Website Files

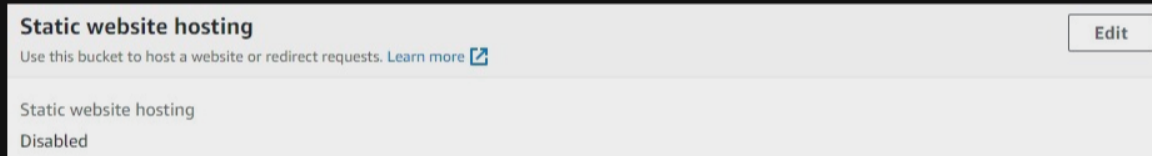


You can see I have Successfully Uploaded My Website Files

Follow For More
www.linkedin.com/in/devops-learning

Step 3:-

Now Let's Enable S3 Static Website Hosting Feature
Go to your Bucket -> Properties -> Static Website Hosting
as shown in below image edit & enable feature



Provide Your Index Document for me its index.html
After that click on Save Changes.

Step 4:-

Let's Connect our Domain to Route53
Go to Route53 in AWS -> Hosted Zone
Create a new Hosted Zone



Provide Your Domain Name & Click on Create Hosted Zone

Now Let's Connect our domain to Route53
Open your Hosted Zone You can see 4 NS Records , we
need to add that in our Domain Fields.
Go to your Domain Management & add 4 NameServers
as shown in the below image



Follow For More
www.linkedin.com/in/devops-learning

You have successfully created your domain to Route53

Step 5:-

Let's Obtain a SSL Certificate from AWS Certificate Manager
Go to your AWS Console -> Certificate Manager
Click on List Certificates -> Request
You can see below screen click on next

Certificate type [Info](#)

ACM certificates can be used to establish secure communications access across the internet or within an internal network. Choose the type of certificate for ACM to provide.

☒ Request a public certificate
Request a public SSL/TLS certificate from Amazon. By default, public certificates are trusted by browsers and operating systems.

☐ Request a private certificate
No private CAs available for issuance.

Requesting a private certificate requires the creation of a private certificate authority (CA). To create a private CA, visit [AWS Private Certificate Authority](#)

Provide your domain name let all the configurations as it is and click on Request

You can see your Request status as "Pending Validation"
Open that Certificate & click on created records in Route53 as shown in the below image

Domains (1)						Create records in Route 53	Export to CSV
< 1 >							
Domain	Status	Renewal status	Type	CNAME name	CNAME value		
tg-resume.tk	Pending validation	-	CNAME	_b05ea0aa68a4d2928233fb74aa4eb595.tg-resume.tk.	_cecf6641c3797cdb3282cd9c76b6d526.fyfbsdpvtv.acm-validations.aws.		

After that click on Create Records

<input checked="" type="checkbox"/>	Domain	Validation status	Type	CNAME name	CNAME value	Is domain in Route 53?
<input checked="" type="checkbox"/>	tg-resume.tk	Pending validation	CNAME	_b05ea0aa68a4d2928233fb74aa4eb595.tg-resume.tk.	_cecf6641c3797cdb3282cd9c76b6d526.fyfbsdpvtv.acm-validations.aws.	Yes
						Cancel Create records

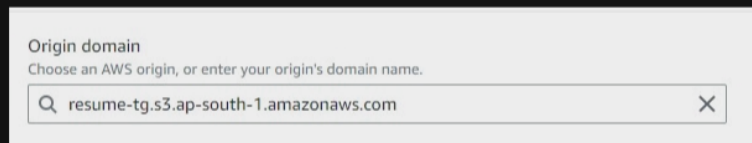
After few minutes you can see your Certificate status as "Issued"

Follow For More
www.linkedin.com/in/devops-learning

Step 6:-

Let's Create a CloudFront distribution with S3 bucket origin and SSL Certificate

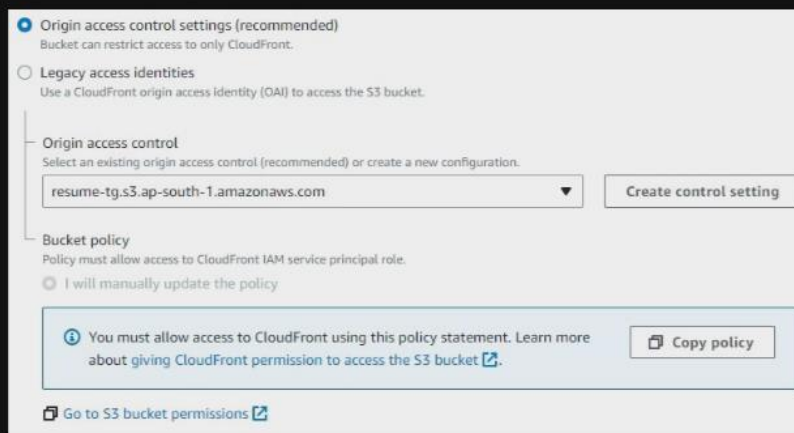
Go to your AWS Console -> CloudFront -> Create Distribution
Provide the Origin Domain as your S3 Bucket



Origin domain

Choose an AWS origin, or enter your origin's domain name.

Provide the Origin Access as shown in image & select your Bucket



☒ Origin access control settings (recommended)
Bucket can restrict access to only CloudFront.

☐ Legacy access identities
Use a CloudFront origin access identity (OAI) to access the S3 bucket.

Origin access control
Select an existing origin access control (recommended) or create a new configuration.

Bucket policy
Policy must allow access to CloudFront IAM service principal role.

☒ I will manually update the policy

☒ You must allow access to CloudFront using this policy statement. [Learn more](#) about giving CloudFront permission to access the S3 bucket [🔗](#).

☒ [Go to S3 bucket permissions](#) [🔗](#)

Follow For More
www.linkedin.com/in/devops-learning

Provide the Viewer protocol policy as shown in the image as we are going to redirect the traffic to HTTPS

Viewer

Viewer protocol policy

☐ HTTP and HTTPS

☒ Redirect HTTP to HTTPS

☐ HTTPS only

Add your Alternate Domain Name

Alternate domain name (CNAME) - optional

Add the custom domain names that you use in URLs for the files served by this distribution.

tg-resume.tk Remove

Add your SSL Certificate

Custom SSL certificate - optional

Associate a certificate from AWS Certificate Manager. The certificate must be in the US East (N. Virginia) Region (us-east-1).

tg-resume.tk (8ef29c23-5934-44e9-9c26-4802905d72d6) ▼ ↺

☒ tg-resume.tk ↗ Request certificate ↗

Legacy clients support - \$600/month prorated charge applies. Most customers do not need this. CloudFront allocates dedicated IP addresses at each CloudFront edge location to serve your content over HTTPS.

☐ Enabled

Security policy

The security policy determines the SSL or TLS protocol and the specific ciphers that CloudFront uses for HTTPS connections with viewers (clients).

☒ TLSv1.2_2021 (recommended)

☐ TLSv1.2_2019

☐ TLSv1.2_2018

☐ TLSv1.1_2016

☐ TLSv1_2016

☐ TLSv1

Finally click on Create Distribution

Follow For More
www.linkedin.com/in/devops-learning

Step 7:

Let's Set Bucket Policy to allow cloudfront to access s3
Go to Buckets -> Open the Bucket we have created
Go to Permission -> Edit Bucket Policy

```
{
  "Version": "2008-10-17",
  "Id": "PolicyForCloudFrontPrivateContent",
  "Statement": [
    {
      "Sid": "AllowCloudFrontServicePrincipal",
      "Effect": "Allow",
      "Principal": {
        "Service": "cloudfront.amazonaws.com"
      },
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::resume-tg/*",
      "Condition": {
        "StringEquals": {
          "AWS:SourceArn": "arn:aws:cloudfront::669087208571:distribution/E1X7VVUG2MRP6G"
        }
      }
    }
  ]
}
```

Provide your bucket details you can see at the top

Provide Cloud Distribution Details that we created

As shown in the image give the same policy edit the provided info as per your bucket and cloudfront distribution
After that click on Save Changes

Step 8:

Now Let's Connect our CloudFront to Route53
Go to Route53 -> Hosted Zone -> Open hosted zone that we have created
After that click on Create Record
Record Type -> A
Alias -> should be enable
Route Traffic to -> Alias to cloudfront distribution
choose distribution that we have created
Finally click on Create Records

Record name [Info](#)

subdomain tg-resume.tk

Keep blank to create a record for the root domain.

☒ Alias

Record type [Info](#)

A - Routes traffic to an IPv4 address and some AWS resources

Route traffic to [Info](#)

Alias to CloudFront distribution

US East (N. Virginia)

An alias to a CloudFront distribution and another record in the same hosted zone are global and available only in US East (N. Virginia).

Q d1v0ei54vtv52h.cloudfront.net X

Routing policy [Info](#)

Simple routing

Evaluate target health

☐ No

Add another record

Refer the above Image for Configurations

Step 9:-

Let's add a error page so that if any error request occurs users should redirect to error page

Go to CloudFront -> Open the CloudFront Distribution that we have created

Open Error Pages click on Create custom error response

HTTP error code

Customize the custom error response when the origin sends this error code.

403: Forbidden

Error caching minimum TTL

Enter the error caching minimum time to live (TTL), in seconds.

10

Customize error response

Send a custom error response instead of the error received from the origin.

☐ No

☒ Yes

Response page path

Enter the path to the custom error response page.

/error.html

HTTP Response code

Choose the HTTP status code to return to the viewer. CloudFront can return a different status code to the viewer than what it received from the origin.

403: Forbidden

After that click on Save Changes if we anything like URL/random_string we can see the error Page



Now you can visit your Website & its Completely hosted by the AWS.

Follow For More

www.linkedin.com/in/devops-learning