**Building a CI/CD Pipeline with AWS CodePipeline to Deploy a Static Website on S3**

**Introduction**

AWS CodePipeline provides a native AWS continuous deployment pipeline to manage web application deployments from the source code repository to the deployment of our web application.

In this lab, we will walk through setting up AWS CodePipeline to continuously deploy our web application to S3. We will walk through the prerequisites to set up our pipeline by setting up our source code repository with AWS CodeCommit and creating an S3 bucket to host static web applications. Finally, we will set up our continuous code pipeline with AWS CodePipeline and deploy our web application to S3.

**Solution**

Log in to the live AWS environment using the credentials provided. Make sure you're in the N. Virginia (us-east-1) region throughout the lab.

Please find the index.html and Mount.jpg files used for this lab in the [GitHub repository](https://github.com/natonic/Developer-Tools-Deep-Dive/tree/master/Labs/PipelineToStaticS3).

**Create a CodeCommit Repository from the Management Console**

1. Navigate to CodeCommit.
2. Under *Repositories*, click **Create repository**.
3. In *Repository name*, enter "sample-static-site".
4. Under *Tags*, click **Add**.
5. Set the following values:
   * *Key:* **CreatedBy**
   * *Value:* Your name
6. Click **Create**.

**Add an index.html and JPG File to the Repository**

1. Scroll to the bottom of the page and click **Add file > Upload file**.
2. Click **Choose file** and select the index.html file downloaded from [GitHub](https://github.com/natonic/Developer-Tools-Deep-Dive/tree/master/Labs/PipelineToStaticS3).
3. In *Commit changes to master*, set the following values:
   * *Author name*: Your name
   * *Email address*: Your email
   * *Commit message*: **Landing page**
4. Click **Commit changes**.
5. Click **Repositories**.
6. Click **sample-static-site** to open the repository page.
7. Click **Add file** > **Upload file**.
8. Click **Choose file** and select the Mount.jpg file downloaded from [GitHub](https://github.com/natonic/Developer-Tools-Deep-Dive/tree/master/Labs/PipelineToStaticS3).
9. In the *Commit changes to master* section, set the following values:
   * *Author name*: Your name
   * *Email address*: Your email
   * *Commit message:* **Photo**
10. Click **Commit changes**.

**Create an S3 Bucket and Configure It To Host a Static Website**

**Create an S3 Bucket**

1. Navigate to S3.
2. Click **Create bucket**.
3. Set the following values:
   * *Bucket name:* Unique bucket name
   * *Region:* **US East (N. Virginia)**
4. Click **Next** > **Next**.
5. Deselect **Block all public access**.
6. Select **I acknowledge that the current settings might result in this bucket and the objects within becoming public**.
7. Click **Add tag**.
8. Set the following values:
   * *Key:* **CreatedBy**
   * *Value:* Your name
9. Leave the rest as their defaults and click **Create bucket**.

**Enable Static Website Hosting**

1. Click the newly created bucket to open it and select the *Properties* tab.
2. Scroll to *Static website hosting* at the bottom and click **Edit**.
3. In *Static website hosting*, select **Enable**.
4. Set the following values:
   * *Index document:* **Index.html**
   * *Error document:* **error.html**
5. Click **Save changes**.

**Create an S3 Bucket Policy**

1. Select the *Permissions* tab.
2. Scroll down to *Bucket policy* and click **Edit**.
3. Copy the bucket ARN number above the policy editor.
4. Click **Policy Generator**.
5. In *Select Type of Policy*, select **S3 Bucket Policy**.
6. In *Add Statement(s)*, set the following values:
   * *Effect:* **Allow**
   * *Principal:* \*
   * *AWS Service:* **Amazon S3**
   * *Actions:* **GetObject**
   * *Amazon Resource Name (ARN):* **<BUCKET\_ARN\_NUMBER>/\***
7. Click **Add Statement**.
8. Click **Generate Policy**.
9. Copy the generated policy to your clipboard.
10. Return to the bucket policy editor in S3 and paste in the policy.
11. Click **Save changes**.

**Create a Pipeline in AWS CodePipeline That Deploys a Static Website**

1. Navigate to CodePipeline.
2. Under *Pipelines* click **Create pipeline**.
3. In *Pipeline name*, enter "sample-static-site-pipeline" and click **Next**.
4. On the *Source* page, set the following values:
   * *Source provider:* **AWS CodeCommit**
   * *Repository name:* **sample-static-site**
   * *Branch name:* **main**
5. In *Change detection options*, select **AWS CodePipeline** and click **Next**.
6. Click **Skip build stage** > **Skip**.
7. In *Deploy* on the next page, select **Amazon S3** as the deploy provider.
8. Set the following values:
   * *Region:* **US East (N. Virginia)**
   * *Bucket:* Your bucket name
9. Click **Extract file before deploy**.
10. Leave the rest as their defaults and click **Next**.
11. Click **Create pipeline**.
12. Once the deploy stage completes, click the Amazon S3 URL under *Deploy* to verify deployment.
13. Select the *Properties* tab and scroll down to **Static website hosting** at the bottom of the page.
14. Click the bucket website endpoint URL to view the static website.

**Conclusion**

Congratulations — you've completed this hands-on lab

