Overview

Create an automated software delivery pipeline using CodeCommit, CodeBuild, CodeDeploy and CodePipeline.

Tasks

Java project setup

During one of the lecture hands-on segments we used GitHub and AWS CodeBuild to generate a build artifact for a very simple Java application. Let's expand on that exercise a bit in this class project. Instead of hosting the application source code in GitHub let's try using AWS CodeCommit, Amazon's managed git repository service. Also, let's use AWS CodePipeline to manage the build process.

Okay, let's start begin building the pipeline!

Create a CloudFormation template

We always create infrastructure resources using code whenever possible and code repositories are no different. You should build a CloudFormation template to create all of the resources for this project including an S3 artifact bucket, CodeCommit repository, CodeBuild project, and CodePipeline pipeline. Instead of starting from scratch, you can use the following template as a starting point: https://seis665-public.s3.amazonaws.com/app-codebuild-template.json

Create a CodeCommit repository

One of the first resources you need to add to the CloudFormation template is a CodeCommit repository. You will use this as a git repository to host the Java application source code. You can use the following documentation to understand how to configure the repository resource:

- https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-resource-codecommit-repository.html

Give the repository a simple name like java-project. When CloudFormation creates the CodeCommit repository, it can automatically populate the repository with code specified by the Code property. In other words, CloudFormation will copy an archive file from a specified bucket and expand the contents of the file in the repository. This is a convenient way to bootstrap the files in a repository. The Code property will reference a source bucket. The value of this property is the actual bucket name, not a URL. A zipfile containing the source code for the Java project is located in the following S3 location: https://seis665-public.s3.amazonaws.com/java-project.zip

Modify the CodeBuild project

The CodeBuild project resource in the template you copied was configured to use a GitHub repository, not CodeCommit. You will need to modify this resource to work properly with CodeCommit. Here is CodeBuild resource CloudFormation documentation:

- https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-resource-codebuild-project.html#cfn-codebuild-project-source

There are a number of changes you need to make to the template to get the CodeBuild project connected to CodeCommit. Here are some hints:

- Since we are no longer using GitHub there's no reason to pass a projectUrl parameter into the template. We can pass the CodeCommit repository information to the CodeBuild project because both resources are defined in the same template.
    - The Source property in the CodeBuild project tells the CodeBuild service where to find the application source code.
    - The location of the CodeCommit repository is going to reference the CloneUrlHttp attribute.
- The ReportBuildStatus property is not supported by CodeBuild projects using CodeCommit repositories and can be removed.
- CodeCommit doesn't use webhooks to trigger CodeBuild projects.
- The CodePipeline project resource uses a custom IAM role (AppBuildRole) to access other AWS services and resources. For example, the role allows the project to retrieve and store objects in an S3 artifact bucket. The role does not currently allow CodePipeline to pull source code from a CodeCommit repository. You can update the project role to allow this by adding the following policy statement to the list of existing statements in the role:

```
{
  "Sid": "CodeCommitPolicy",
  "Effect": "Allow",
  "Action": [
    "codecommit:GitPull"
  ],
  "Resource": [
    "*"
  ]
}
```

Test your template changes

Now is a good time to test the changes you have made to the CloudFormation template. Create a new stack using your template and wait for it to create all of the resources. Access the

CodeCommit web console to verify that a java-project repository exists and it contains a set of application source files. One of those files should be the buildspec.yml file.
Next, go to the CodeBuild web console and verify that a build project exists. It should be configured to use your java-project repository as a build source. Go ahead and start a build to test the build process. Take a look at the build logs as the project is building.
Once you see a successful build and a new artifact uploaded into the S3 artifact bucket you are ready to continue working on the next stage of the project. If your build doesn't work, make note of the error message and try to figure out what is going on using your favorite search engine or troubleshooting sites like StackOverflow.

Add a pipeline resource

Our current stack built out a CodeBuild project which is able to retrieve source code from CodeCommit, build it, and generate an output artifact in S3. The build process we are simulating is super simple and generally real-world processes are much more complex. What if we had to combine a number of build processes together? We could do that by creating a pipeline.

Let's add a CodePipeline pipeline resource to the template. You could either start creating this resource from scratch or you could copy code from one of the hands-on templates that we used during the lecture: https://seis665-public.s3.amazonaws.com/app-codepipeline-template.json

Here's the documentation for the AWS::CodePipeline::Pipeline resource:

- https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-resource-codepipeline-pipeline.html

The build pipeline will have two stages. The first stage should be called Source and it will be configured to pull source code from the CodeCommit repository. The second stage should be called Build and it will take the source code and build it using CodeBuild.
Here are some hints:

- Most of the work in this part of the project involves setting up the source stage. If you copied the pipeline resource from the lecture template, you will have to change the source from GitHub to CodeCommit. This means that you have to change the Owner, Provider, and Configuration attributes. What values should you use for CodeCommit? The easiest way to figure this out is by looking at the following AWS documentation:
    - https://docs.aws.amazon.com/codepipeline/latest/userguide/reference-pipeline-structure.html#action-requirements
- The pipeline configuration references a special IAM service role (CodePipelineServiceRole) to access other AWS services. Here's the configuration code I recommend using for the service role:

```
"CodePipelineServiceRole": {
  "Type": "AWS::IAM::Role",
  "Properties": {
      "AssumeRolePolicyDocument": {
          "Statement": [
              {
                  "Action": [
                      "sts:AssumeRole"
                  ],
                  "Effect": "Allow",
                  "Principal": {
                      "Service": [
                          "codepipeline.amazonaws.com"
                      ]
                  }
              }
          ]
      },
      "Path": "/service-role/",
      "Policies": [
          {
              "PolicyDocument": {
                  "Statement": [
                      {
                          "Effect": "Allow",
                          "Resource": [
                              {
                                  "Fn::GetAtt": [
                                      "ArtifactBucket",
                                      "Arn"
                                  ]
                              },
                              {"Fn::Join": [
                                  "",
                                  [
                                      {
                                          "Fn::GetAtt": [
                                              "ArtifactBucket",
                                              "Arn"
                                          ]
                                      },
                                      "/*"
                                  ]
                              ]}
                          ],
                          "Action": [
                              "s3:PutObject",
                              "s3:GetObject",
                              "s3:GetBucketAcl",
                              "s3:GetBucketLocation"
                          ]
                      },
                      {
                          "Action": [
                              "codecommit:CancelUploadArchive",
                              "codecommit:GetBranch",
                              "codecommit:GetCommit",
```

```
                                "codecommit:GetUploadArchiveStatus",
                                "codecommit:UploadArchive"
                            ],
                            "Resource": [
                                { "Fn::GetAtt": [ "JavaRepository", "Arn" ] }
                            ],
                            "Effect": "Allow"
                        },
                        {
                            "Action": [
                                "codebuild:BatchGetBuilds",
                                "codebuild:StartBuild"
                            ],
                            "Resource": [
                                {    "Fn::GetAtt":    [    "AppBuildProject",
"Arn" ] }
                            ],
                            "Effect": "Allow"
                        }
                    ],
                    "Version": "2012-10-17"
                },
                "PolicyName": "ec2codedeploy"
            }
        ]
    }
}
```

Update your CloudFormation stack with the new template code and try to trigger the pipeline (release change). You should see the source and build stages successfully complete if everything is configured correctly. If you encounter any failure, look at the log messages to try to determine the cause of the error. Oftentimes an error in the pipeline is caused by a permissions problem.

Congratulations! You have created a very basic software delivery pipeline.

Submit your work

Create a new GitHub repository. Please push your template files to this repository and submit this GitHub repository's URL on Canvas.

Terminate AWS resources

Remember to terminate all the resources created in this project.