Jenkins project basic info

In this project jenkins integrate with AWS code build and Aws code deploy using s3 artifacts.

Prerequisites:

Before proceeding integration with Jenkins with Aws code build,Aws code deploy,S3 plugins are installed into Jenkins server.

Steps to be followed in the aws management console

1.An Aws account with the necessary permissions to create and configure code build and code deploy applications

2.A jenkins server up and running with the necessary plugins installed for aws integration

3.code build and code deploy application configured into aws console

4.create a two roles one is having s3 full access,ec2 full access and administration access is attached to jenkins server and the other having code deploy access is attached to deployment servers 1 and 2

**Create the resources**

* Amazon S3 bucket—Stores the GitHub repository files and the CodeBuild artifact application file that CodeDeploy uses.
* IAM S3 bucket policy—Allows the Jenkins server access to the S3 bucket.
* JenkinsRole—An IAM role and instance profile for the Amazon EC2 instance for use as a Jenkins server. This role allows Jenkins on the EC2 instance to access the S3 bucket to write files and access to create CodeDeploy deployments.
* CodeDeploy application and CodeDeploy deployment group.
* CodeDeploy service role—An IAM role to enable CodeDeploy to read the tags applied to the instances or the EC2 Auto Scaling group names associated with the instances.
* CodeDeployRole—An IAM role and instance profile for the EC2 instances of CodeDeploy. This role has permissions to write files to the S3 bucket created by this template and to create deployments in CodeDeploy.
* CodeBuildRole—An IAM role to be used by CodeBuild to access the S3 bucket and create the build projects.
* Jenkins server—An EC2 instance running Jenkins.
* CodeBuild project—This is configured with the S3 bucket and S3 artifact.
* Auto Scaling group—Contains EC2 instances running Apache and the CodeDeploy agent fronted by an Elastic Load Balancer.
* Auto Scaling launch configurations—For use by the Auto Scaling group.
* Security groups—For the Jenkins server, the load balancer, and the CodeDeploy EC2 instances.

1. To create the CloudFormation stack (for example in the AWS Frankfurt Region) click

## .

## Conclusion

In this post, I outlined how you can use a Jenkins open-source automation server to deploy CodeBuild artifacts with CodeDeploy. I showed you how to construct a functioning CI/CD pipeline with these tools. I walked you through how to build the deployment infrastructure and automatically deploy application version changes from GitHub to your production environment.

