> ⓘ AWS Mainframe Modernization is currently in preview. It is subject to change. We do not recommend using it for production workloads.

Enterprise Build Tools Tutorial

This tutorial demonstrates how to use the AWS Mainframe Modernization Micro Focus Build Tools to compile the BankDemo sample application source code from Amazon S3 and then export the compiled code back to Amazon S3.

AWS CodeBuild is a fully managed continuous integration service that compiles source code, runs tests, and produces software packages that are ready to deploy. With CodeBuild, you can use prepackaged build environments, or you can create custom build environments that use your own build tools. This demo scenario uses the second option. It consists of a CodeBuild build environment using a pre-packaged Docker image that contains the Micro Focus Build Tools.

Topics

- Prerequisite (#tutorial-build-prerequisites)
- Step 1: Compile the BankDemo source code using CodeBuild (#tutorial-build-step1)

Prerequisite

Before you start this tutorial, complete the following prerequisite.

- Install the AWS CLI. For more information, see Installing or updating the latest version of the AWS CLI (https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html) in the *https://docs.aws.amazon.com/cli/latest/userguide/*.

Step 1: Compile the BankDemo source code using CodeBuild

Follow these steps to compile the BankDemo source code using CodeBuild:

1. Create two Amazon S3 buckets following the instructions in Creating, configuring, and working with Amazon S3 buckets (https://docs.aws.amazon.com/AmazonS3/latest/userguide/creating-buckets-s3.html) in the *Amazon S3 User Guide*.

   - Create the input bucket, where the source code stays:

     ```
     aws s3api create-bucket --bucket codebuild-regionId-accountId-input-bucket
     --region regionId
     ```

     where `regionId` is the AWS Region of the bucket and `accountId` is your AWS account ID.

   - Create the output bucket, which will be the destination for the compiled code:

```
aws s3api create-bucket --bucket codebuild-regionId-accountId-output-bucket
--region regionId
```

where `regionId` is the AWS Region of the bucket and `accountId` is your AWS account ID.

2. Create a file named `buildspec.yml` at the local root directory of your source code folder. This is a collection of build commands and related settings, in YAML format, that CodeBuild uses to run a build. See Build specification reference for CodeBuild (https://docs.aws.amazon.com/codebuild/latest/userguide/build-spec-ref.html) in the *AWS CodeBuild User Guide* for more details. Add the following content at the end of the file:

```
version: 0.2
phases:
  install:
    runtime-versions:
      python: 3.7
  pre_build:
    commands:
      - echo Installing source dependencies...
  build:
    commands:
      - echo Build started on `date`
      - . /opt/microfocus/EnterpriseDeveloper/bin/cobsetenv
      - /opt/microfocus/EnterpriseDeveloper/remotedev/ant/apache-ant-1.9.9/bin/ant
-lib /opt/microfocus/EnterpriseDeveloper/lib/mfant.jar -f build.xml
-Dbasedir=$CODEBUILD_SRC_DIR/Sources -Dloaddir=$CODEBUILD_SRC_DIR/target
  post_build:
    commands:
      - echo Build completed on `date`
artifacts:
  files:
    - $CODEBUILD_SRC_DIR/target/**
```

3. Upload the BankDemo sample application source code to the input bucket:

```
aws s3 cp LocalPath s3://codebuild-regionId-accountId-input-bucket/ --recursive
```

where `LocalPath` is the directory with the source code, `regionId` is the AWS Region of the bucket, and `accountId` is your AWS account ID.

4. Create an IAM policy (https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies.html) that grants permissions to interact with the input and output buckets, as well as with the Amazon CloudWatch logs (https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/WhatIsCloudWatchLogs.html) that CodeBuild generates:

```
aws iam create-policy \
--policy-name BankdemoCodeBuildRolePolicy \
--policy-document \
  '{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "logs:CreateLogGroup",
                "logs:CreateLogStream",
                "logs:PutLogEvents"
            ],
            "Resource": [
                "arn:aws:logs:regionId:accountId:log-group:/aws/codebuild
/codebuild-bankdemo-project",
                "arn:aws:logs:regionId:accountId:log-group:/aws/codebuild
/codebuild-bankdemo-project:*"
            ],
            "Effect": "Allow"
        },
        {
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:GetObjectVersion",
                "s3:GetBucketAcl",
                "s3:GetBucketLocation",
                "s3:List*"
            ],
            "Resource": [
                "arn:aws:s3:::codebuild-regionId-accountId-input-bucket",
                "arn:aws:s3:::codebuild-regionId-accountId-input-bucket/*",
                "arn:aws:s3:::codebuild-regionId-accountId-output-bucket",
                "arn:aws:s3:::codebuild-regionId-accountId-output-bucket/*"
            ],
            "Effect": "Allow"
        }
    ]
}'
```

ⓘNote

If you are storing the Docker image for the Micro Focus Build Tools at Amazon Elastic Container Registry, you have to add additional permissions to the preceding policy:

```
{
```

```
            "Action": [
                "ecr:*"
            ],
            "Resource": [
                "MICRO_FOCUS_IMAGE_ARN"
            ],
            "Effect": "Allow"
    },          {
    {
            "Action": [
                "ecr:GetAuthorizationToken"
            ],
            "Resource": [
                "*"
            ],
            "Effect": "Allow"
    },
```

5. Create a new IAM role (https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles.html) that will grant CodeBuild permissions to interact with AWS resources on your behalf, after you associate the IAM policy that you created before:

```
aws iam create-role \
    --role-name BankdemoCodeBuildServiceRole \
    --assume-role-policy-document \
        '{
          "Version": "2012-10-17",
          "Statement": [
            {
              "Effect": "Allow",
              "Principal": {
                "Service": "codebuild.amazonaws.com"
              },
              "Action": "sts:AssumeRole"
            }
          ]
        }'
```

6. Attach the BankdemoCodeBuildRolePolicy IAM policy to the BankdemoCodeBuildServiceRole IAM role:

```
aws iam attach-role-policy \
    --region us-east-1 \
```

```
        --role-name BankdemoCodeBuildServiceRole \
        --policy-arn arn:aws:iam::accountId:policy/BankdemoCodeBuildRolePolicy
```

7. Create the CodeBuild project using the following command:

```
 aws codebuild create-project --name codebuild-bankdemo-project \
    --source \
      '{
        "type": "S3",
        "location": "codebuild-*regionId-accountId-input-bucket/"
      }' \
    --artifacts \
      '{
        "type": "S3",
        "location": "codebuild-regionId-accountId-output-bucket",
        "path": "builds/",
        "namespaceType": "NONE",
        "name": "bankdemo-output.zip",
        "packaging": "ZIP"
      }' \
    --environment \
      '{
        "type": "LINUX_CONTAINER",
        "image": "PATH_TO_MICRO_FOCUS_BUILD_TOOLS_DOCKER_IMAGE>",
        "computeType": "BUILD_GENERAL1_SMALL",
        "imagePullCredentialsType": "SERVICE_ROLE"
      }' \
    --service-role arn:aws:iam::accountId:role/BankdemoCodeBuildServiceRole
```

8. Start the build with the following command:

```
aws codebuild start-build --project-name codebuild-bankdemo-project
```

This command will trigger the build, which will run asynchronously. The output of the command is a JSON that includes the attribute id, which is a reference to the CodeBuild build id that you just initiated. You can view the status of the build with the command:

```
aws codebuild batch-get-builds --ids codebuild-bankdemo-project:buildId
```

When the current phase is COMPLETED, it means that your build finished successfully, and your compiled artifacts are ready on Amazon S3. You can also check the build progress on the AWS console ⤷ (https://console.aws.amazon.com/codesuite/codebuild/projects/codebuild-bankdemo-project/history) . In the console, you can also see detailed logs about the build execution. See View detailed build information (https://docs.aws.amazon.com/codebuild/latest/userguide/getting-started-build-log-console.html) for more

information.

9. Finally, download the output artifacts from Amazon S3 using the command:

```
aws s3 cp s3://codebuild-regionId-accountId-output-bucket/builds/bankdemo-
output.zip bankdemo-output.zip
```