



NextWork.org

Package an App with CodeBuild



Sadeesha Perera



AWS CodeBuild



Sadeesha Perera
[linkedin.com/sadeesha-perera](https://www.linkedin.com/sadeesha-perera)

NextWork.org

Introducing AWS CodeBuild!

What it does & how it's useful

AWS CodeBuild is a fully managed build service in the cloud. Developers and teams use AWS CodeBuild because it compiles the source code, runs unit tests, and produces artifacts that are ready to deploy. CodeBuild eliminates the need to provision, manage, and scale your own build servers.

How I'm using it in today's project

I'm using AWS CodeBuild in this project to fetch, compile, and package my web app project into a WAR file.

This project took me...

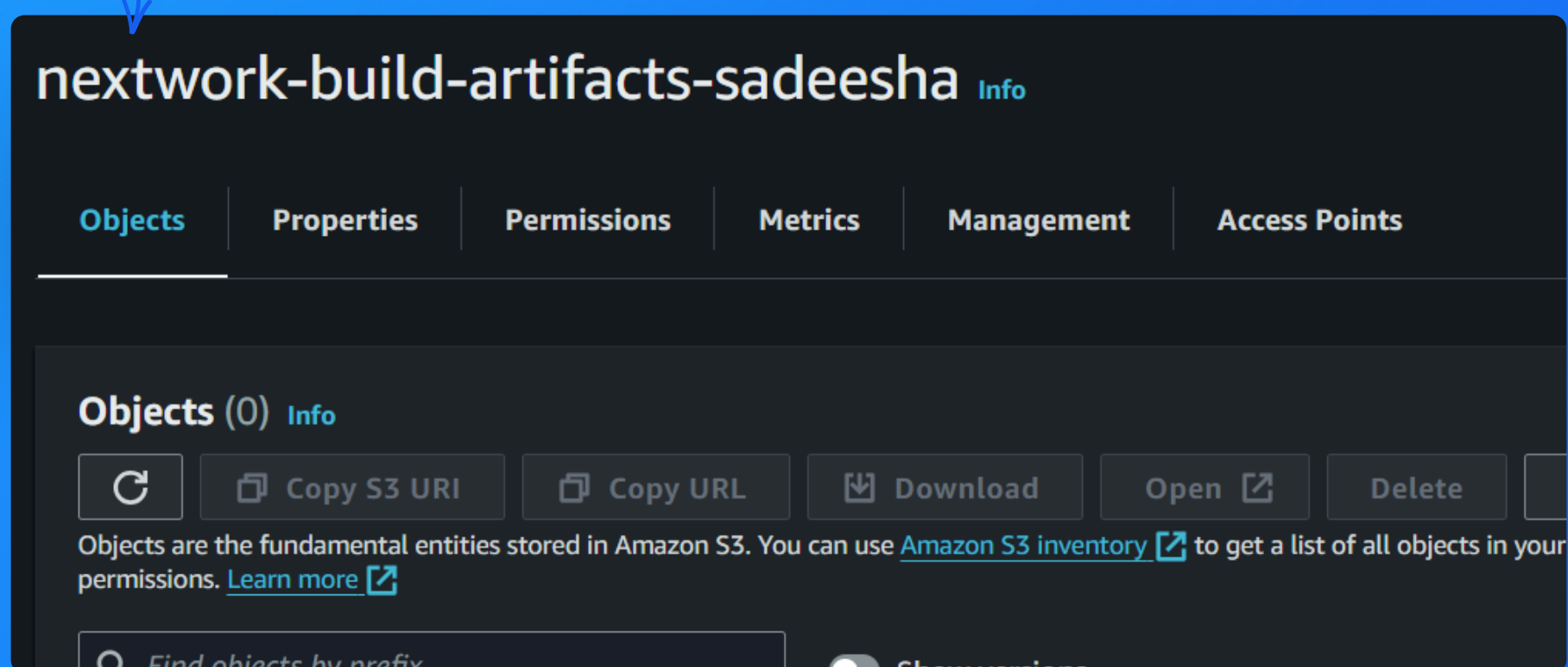
It took me 30 minutes to complete this project, and documentation took me 30 minutes as well. In total i spent 1 hour building and documenting this project.



Set up an S3 bucket

- I started my project by creating an S3 bucket because this bucket will later catch an important build artifact that gets created from the build process I'm setting up with CodeBuild.
- The key artifact that this S3 bucket will capture is called a WAR file(Web Application Resource).
- This file is important because it ensures that any server that will host my web app will have all of the resources/tools it needs to successfully run my application.

My S3 bucket!





Set up a CodeBuild project

When creating a project in CodeBuild, there were 5 key configurations I set up:

1. Source, which means the source of the project's code or files. I selected AWS CodeCommit, as that is my cloud repository for my project's files.
2. Environment, which means the set of resources/configurations required to build my web app. I selected an EC2 instance running Amazon Linux2 and Java Coretto 8 as the runtime.
3. Buildspec, which are the commands to run in my local computer terminal during the build process. I selected "use a buildspec file."
4. Artifacts, which means where CodeBuild should store the build artifact that are produced in the build process I selected the S3 bucket I created.
5. Logs, which means whether I like to keep records of every single build process that CodBuild does. I selected CloudWatch logs.

My completed project ready for the first build!

The screenshot shows the AWS CodeBuild console interface. At the top, a green notification bar states "Project created" and "You have successfully created the following project: nextwork-web-build". Below this, the breadcrumb navigation is "Developer Tools > CodeBuild > Build projects > nextwork-web-build". The main heading is "nextwork-web-build". Below the heading are several buttons: "Actions", "Create trigger", "Edit", "Clone", "Debug build", "Start build with overrides", and "Start build". A "Configuration" section is visible, containing a table with the following details:

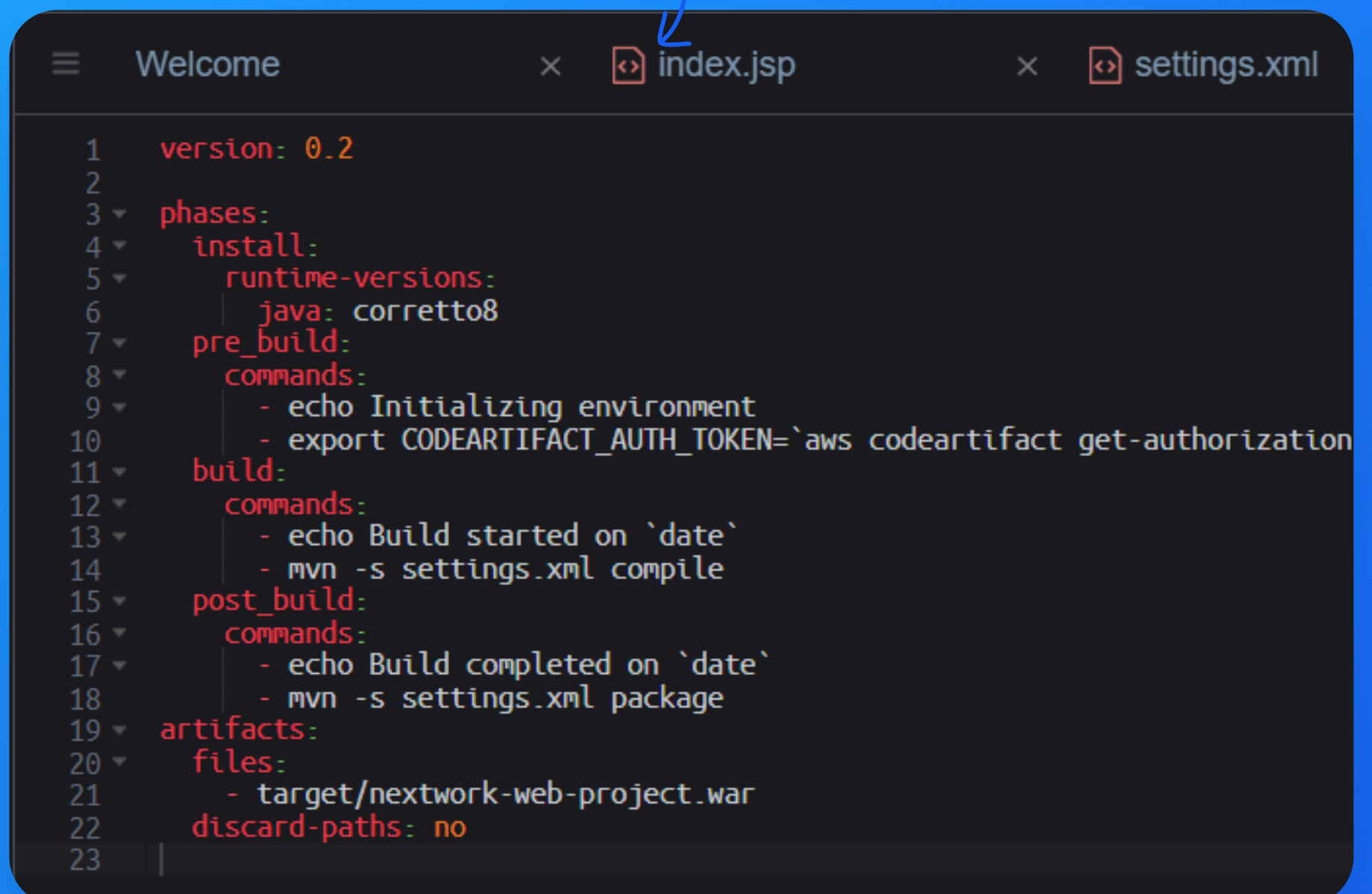
Configuration			
Source provider	Primary repository	Artifacts upload location	Service role
AWS CodeCommit	nextwork-web-project	nextwork-build-artifacts-sadeesha	arn:aws:iam::992382852391:role/service-role/codebuild-nextwork-web-build-service-role



Create a buildspec.yml file

- I created a buildspec.yml file at the root of my code repository.
- This file contains four phases that tells our build environment what commands to run. These four phases are:
 - a. Install - i.e. install these dependencies before you start compiling (in this project it's Java Coretto 8).
 - b. pre_build - i.e. before Maven starts building, retrieve an access token to our CodeArtifact repository so we can fetch dependencies later.
 - c. Build i.e. to build our web app project, run these commands.
 - d. Post_build i.e. after building our web app project, package up the machine code using the settings in the setting.xml file.

A peek into my buildspec.yml



```
1  version: 0.2
2
3  phases:
4    install:
5      runtime-versions:
6        java: corretto8
7    pre_build:
8      commands:
9        - echo Initializing environment
10       - export CODEARTIFACT_AUTH_TOKEN=`aws codeartifact get-authorization
11    build:
12      commands:
13        - echo Build started on `date`
14        - mvn -s settings.xml compile
15    post_build:
16      commands:
17        - echo Build completed on `date`
18        - mvn -s settings.xml package
19    artifacts:
20      files:
21        - target/nextwork-web-project.war
22    discard-paths: no
23
```




Edit CodeBuild's IAM role

- Before I start building my web app project (exciting!), I modified my CodeBuild project's service role first. This role was first created when I set up my CodeBuild project(I checked the settings for creating a new service role).
- I attached a new policy called codeartifact-nextworkconsumer-policy to my CodeBuild project's IAM role. This means my CodeBuild project now has access to the packages/dependencies that it will later compile.

Updating permission policies for my CodeBuild project's IAM role.

Permissions policies (3) Info

You can attach up to 10 managed policies.

Search

Filter

All t

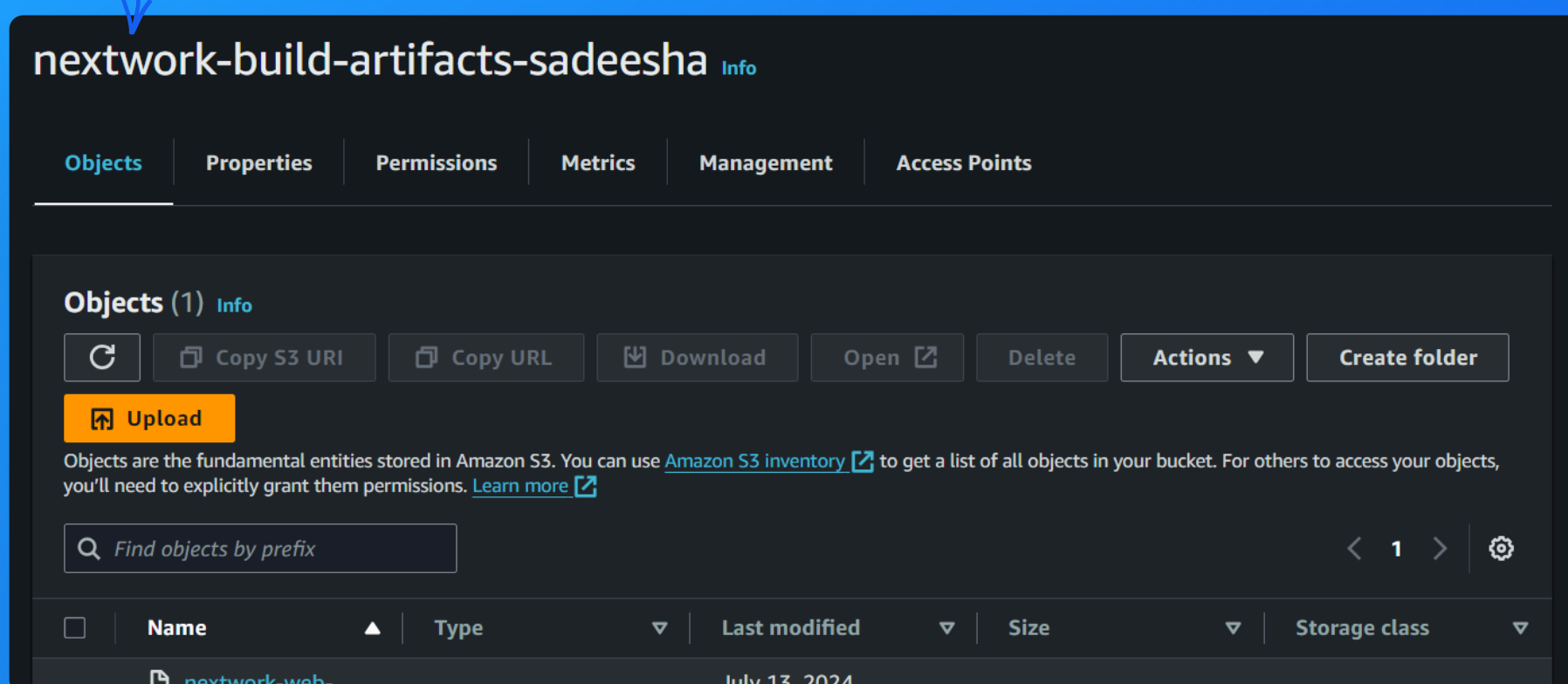
<input type="checkbox"/>	Policy name ↗	Type
<input type="checkbox"/>	<div><div>+</div>codeartifactpolicy</div>	Customer ma
<input type="checkbox"/>	<div><div>+</div>CodeBuildBasePolicy-nextwork-w...</div>	Customer ma
<input type="checkbox"/>	<div><div>+</div>CodeBuildCloudWatchLogsPolicy-...</div>	Customer ma



My first project build 💪

- To build my project, all I had to do was select the Start build button in my CodeBuild console.
- The build process in CodeBuild took about 5 minutes.
- Once the build is complete, I checked my S3 bucket that was set up at the start of this project to catch the build artifact that get produces from my build process. I saw the zip file i.e. the WAR file (a bundled up package of all the files/resources a s erver will need to run my web app), which verified that the build was completed successfully.

My completed project ready for the first build!





My key learnings

- 1 The build process, or 'building' , means is the process of taking my web app's code and translating it into machine code that servers (like computers or EC2 instances) can understand and run.
- 2 The buildspec.yml file is to configure the commands that CodeBuild will run line by line to build your web app
- 3 Even though CodeBuild creates a new service role for my build environment, I still have to modify the role's permission policies because, the new service role gave CodeBuild the minimum permissions it needs to build my web app, which by default does not include access to my CodeArtifact repository
- 4 One thing I didn't expect was how fast the build was completed after setting everything up.

Errors I was ran into 😊

Status

❌ Failed

Initiator

Sadeesha-IAM-Admin

Build ARN

arn:aws:codebuild:us-east-1:992382852391:build/nextwork-web-build:ea8f8b4b-7d88-495b-8885-9cb096d20b0e

Resolved source version

-

Start time

Jul 13, 2024 11:32 PM (UTC+5:30)

End time

Jul 13, 2024 11:32 PM (UTC+5:30)

Build number

2

Build logs

Phase details

Reports

Environment variables

Build details

Resource utilization

Showing the last 7 lines of the build log. [View entire log](#)

Tail logs

No previous logs

1 [Container] 2024/07/13 18:02:42.843750 Running on CodeBuild On-demand

2 [Container] 2024/07/13 18:02:42.843766 Waiting for agent ping

3 [Container] 2024/07/13 18:02:43.149862 Waiting for DOWNLOAD_SOURCE

4 [Container] 2024/07/13 18:02:50.233883 Phase is DOWNLOAD_SOURCE

5 [Container] 2024/07/13 18:02:50.235350 CODEBUILD_SRC_DIR=/codebuild/output/src3399461/src/git-codecommit.us-east-1.amazonaws.com/v1/repos/nextwork-web-project

6 [Container] 2024/07/13 18:02:50.241700 Phase completed: DOWNLOAD_SOURCE. Status: FAILED

- I mistakenly place buildspec.yml file in different folder. So I have to push code and build it again 😞



Everyone should be in a job they love. *yes!*

Check out community.nextwork.org for more free projects

