



Dependencies and CodeArtifact



Prerana P

The screenshot shows the AWS CodeArtifact interface. On the left, there's a sidebar with navigation links for Source, Artifacts, Repositories, and Pipelines. Under 'Repositories', 'nextwork-packages' is selected. The main area displays a table of packages:

Package name	Namespace	Format	Latest version	Latest publish date	Publish	Upstream
backport-util-concurrent	backport-util-concurrent	maven	3.1	Just now	Block	Allow
classworlds	classworlds	maven	1.1	1 minute ago	Block	Allow
google	com.google	maven	1	Just now	Block	Allow
jboss	com.jboss	maven	2.0.1	Just now	Block	Allow
google-collections	com.google.collections	maven	1.0	Just now	Block	Allow
commons-collections	commons-collections	maven	1.0	1 minute ago	Block	Allow
commons-*	commons-*	maven	**	**	Block	Allow



Introducing today's project!

What is AWS CodeArtifact?

AWS CodeArtifact is a fully managed artifact repository service that allows you to securely store, share, and manage software packages used in your development projects. It supports popular package formats like Maven, npm, and Python, enabling teams

How I used CodeArtifact in this project

In today's project, I used AWS CodeArtifact to manage and store the dependencies for my web app. I connected my project to CodeArtifact by configuring the settings.xml file in Maven to authenticate and access the repository. After compiling the web app

One thing I didn't expect in this project was...

while connecting to the code artifact by the given export statement it was failing later ran the compile statement as

```
CODEARTIFACT_AUTH_TOKEN=$(aws codeartifact get-authorization-token --domain nextwork --domain-owner 585008081047 --region ap-south-1)
```

This project took me...

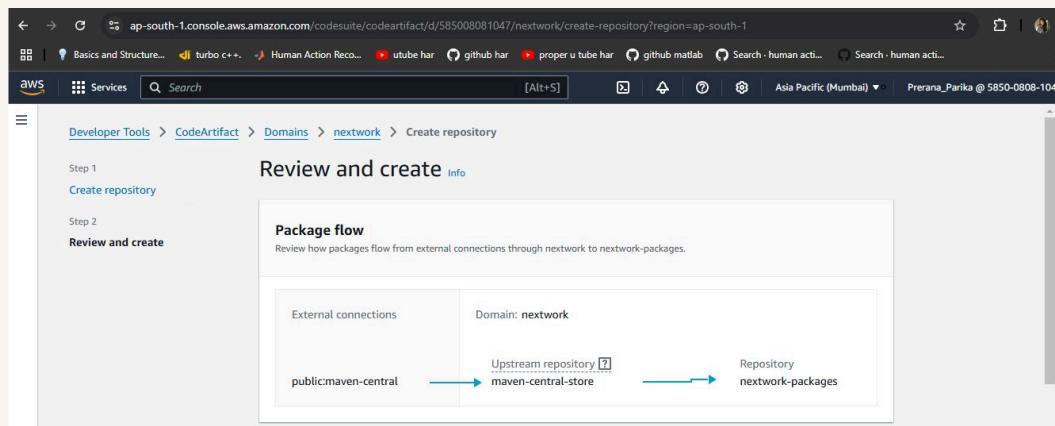
it took around 40min

My project has three artifact repositories

The local repository is a version-controlled directory on your local computer where you store and manage files for a project. It contains all the files, folders, and the version history of the project, typically maintained by a version control system

The upstream repository is the primary or original repository, typically hosted on a remote platform like GitHub, GitLab, or Bitbucket, from which a local repository or a forked repository is cloned. It serves as the central source of truth for the p

The public repository is a version-controlled repository that is accessible to anyone on the internet, allowing others to view, clone, and contribute to the project without requiring special permissions. It is often hosted on platforms like GitHub, G





Connecting my project with CodeArtifact

I connected my web app project (via my Cloud9 IDE) to CodeArtifact to securely manage and store project dependencies in a central, scalable repository. This integration allows me to pull, share, and manage libraries efficiently while ensuring consist

I created a new file, settings.xml, in my web app

`settings.xml` is a configuration file used by Apache Maven, a build automation tool, to define project-specific or user-specific settings for builds. It contains configurations for things like repositories, proxies, authentication details, and build phases.

The code configures Maven to authenticate with AWS CodeArtifact using an environment token. It sets the CodeArtifact repository as the default for dependencies and mirrors all repositories through it, ensuring seamless interaction with your CodeArtif

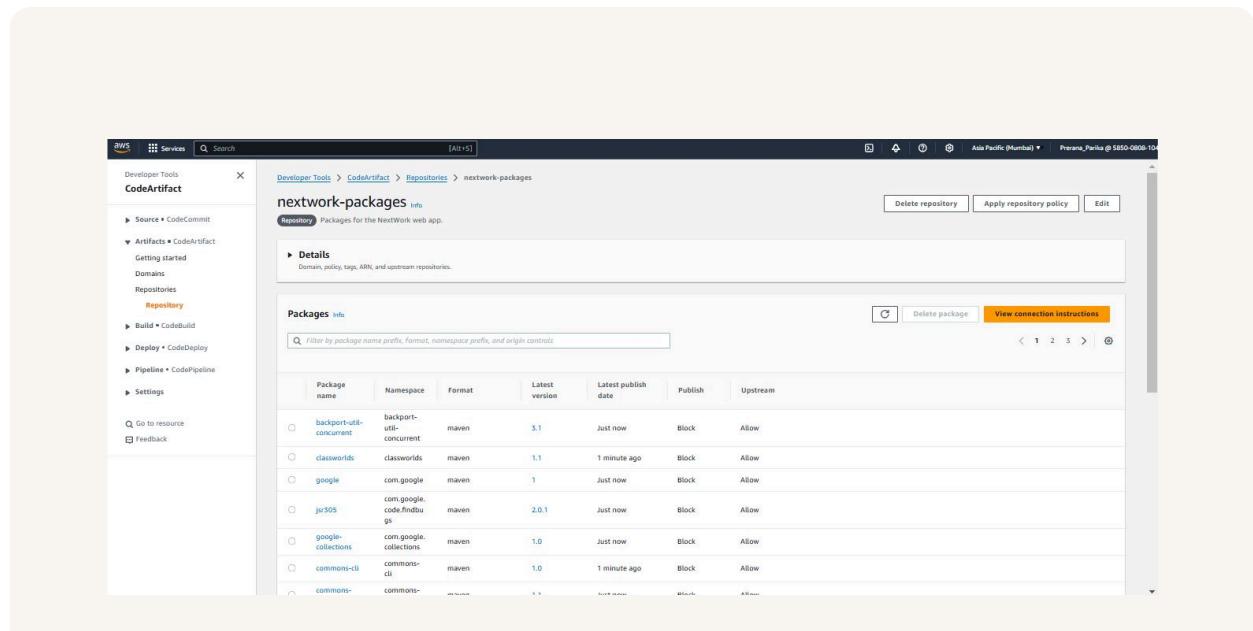
Testing the connection

To test the connection between Cloud9 and CodeArtifact, I compiled my web app

Compiling means the process of converting source code written in a programming language into machine code or bytecode that can be executed by a computer. It checks for syntax errors and creates an executable or intermediary file, enabling the program

Success!

After compiling, I checked the CodeArtifact repository. I saw that the compiled dependencies were successfully pushed and stored there, allowing me to verify that the necessary packages were available for use in the project, ensuring smooth builds an



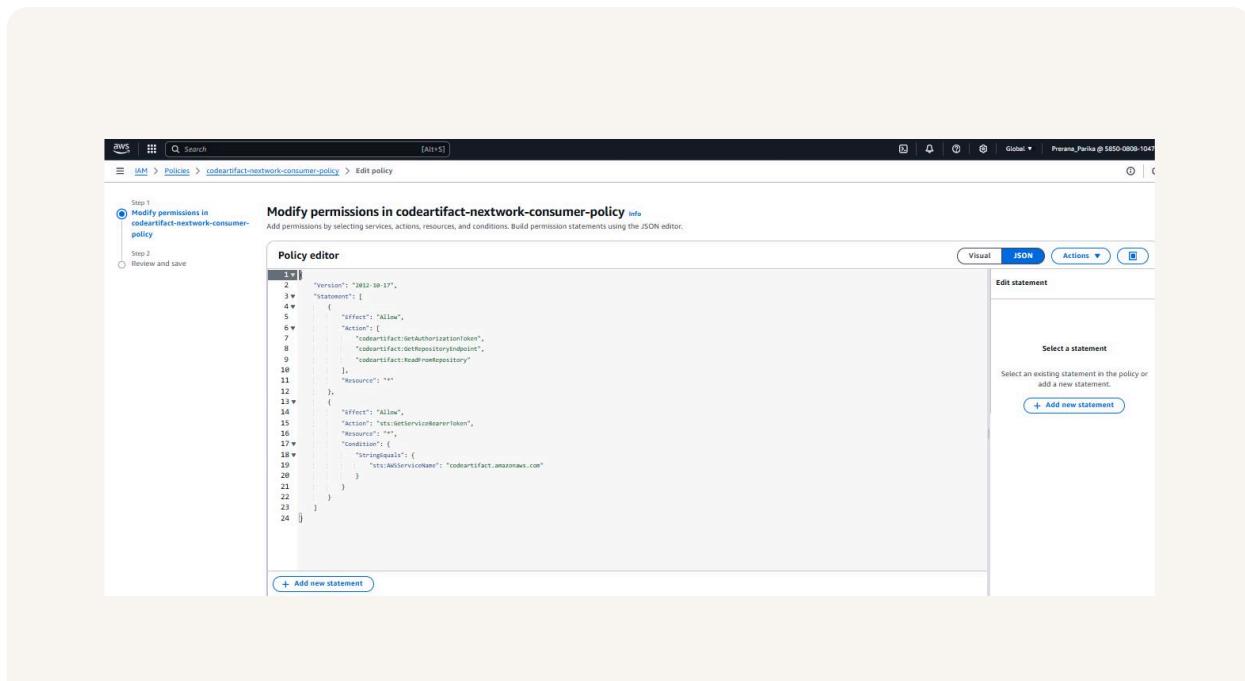
Create IAM policies

The importance of IAM policies

I also created an IAM policy because it defines permissions for the service to interact with CodeArtifact. It allows retrieving an authorization token, locating the repository endpoint, and reading packages from the repository, ensuring secure and co

I defined my IAM policy using JSON

This policy will allow the service to perform three actions: get an authorization token (codeartifact:GetAuthorizationToken), locate the repository endpoint (codeartifact:GetRepositoryEndpoint), and read/download packages from the repository (codeart





NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for
more projects

