



# Secure Packages with CodeArtifact

EA

eanitaokele@gmail.com

Filter by package name prefix, format, namespace prefix, and origin controls							
	Package name	Namespace	Format	Latest version	Latest publish date	Publish	Upstream
<input type="radio"/>	backport-util-concurrent	backport-util-concurrent	maven	3.1	36 minutes ago	Block	Allow
<input type="radio"/>	classworlds	classworlds	maven	1.1	36 minutes ago	Block	Allow
<input type="radio"/>	google	com.google	maven	1	36 minutes ago	Block	Allow
<input type="radio"/>	jsr305	com.google.code.findbugs	maven	2.0.1	36 minutes ago	Block	Allow
<input type="radio"/>	google-collections	com.google.collections	maven	1.0	36 minutes ago	Block	Allow
<input type="radio"/>	commons-cli	commons-cli	maven	1.0	36 minutes ago	Block	Allow
<input type="radio"/>	commons-logging-api	commons-logging	maven	1.1	36 minutes ago	Block	Allow
<input type="radio"/>	junit	junit	maven	3.8.2	36 minutes ago	Block	Allow
<input type="radio"/>	log4j	log4j	maven	1.2.12	36 minutes ago	Block	Allow
<input type="radio"/>	apache	org.apache	maven	13	36 minutes ago	Block	Allow
<input type="radio"/>	maven	org.apache.maven	maven	2.2.1	36 minutes ago	Block	Allow
<input type="radio"/>	maven-artifact	org.apache.maven	maven	2.2.1	36 minutes ago	Block	Allow

# Introducing Today's Project!

In this project, I would demonstrate how to set up CodeArtifact as my CI/CD pipeline's artifact repository. I'm doing this project to show the importance and value of having an artifact repository as it's a huge time saver.

## Key tools and concepts

Services I used were CodeArtifact, IAM, Amazon EC2, GitHub and VSCode. Key concepts I learnt include using Artifact repositories, connecting Maven with CodeArtifact, setting up IAM permissions to give EC2 instance permission to access CodeArtifact.

## Project reflection

This project took me approximately 4 hours. The most challenging part was creating a new IAM policy. It was most rewarding to see the packages in CodeArtifact after the connection was set up.

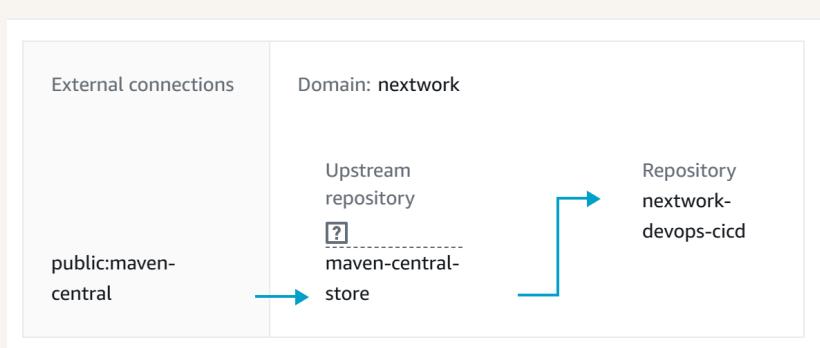
This project is part three of a series of DevOps projects where I'm building a CI/CD pipeline! I'll be working on the next project very soon.

# CodeArtifact Repository

CodeArtifact is an artifact repository service, used to create repositories for storing web app packages and dependencies. Engineering teams use artifact repositories for security, control and reliability.

A domain is like a folder that holds together multiple repositories under the same project/organization. They're helpful for setting up permissions or settings for multiple CodeArtifact repositories in one go. My domain is called nextwork.

A CodeArtifact repository can have an upstream repository, which means a public/backup source of packages that the primary repository can access when it does not have what is needed. My repository's upstream repository is Maven Central Store.



# CodeArtifact Security

## Issue

To access CodeArtifact, we need an authentication token that allows the EC2 instance to access CodeArtifact for 12 hours. I ran into an error when retrieving a token because EC2 instance doesn't have default permissions to access other AWS services.

## Resolution

To resolve the error with my security token, I had to set up an IAM policy that grants access to CodeArtifact and then IAM roles that was attached to the EC2 instance. This resolved the error because the EC2 instance now has access to the repository.

It's security best practice to use IAM roles because they are more secure and scalable than hardcoded credentials. Hardcoded credentials are much more vulnerable to security attacks and getting misused resulting in errors, losses and delays.

# The JSON policy attached to my role

The JSON policy I set up grants access to CodeArtifact. Specifically, it grants access to three key actions related to CodeArtifact: retrieving an authorization token, finding the repository endpoint and viewing the packages inside the repository.

```
Create policy

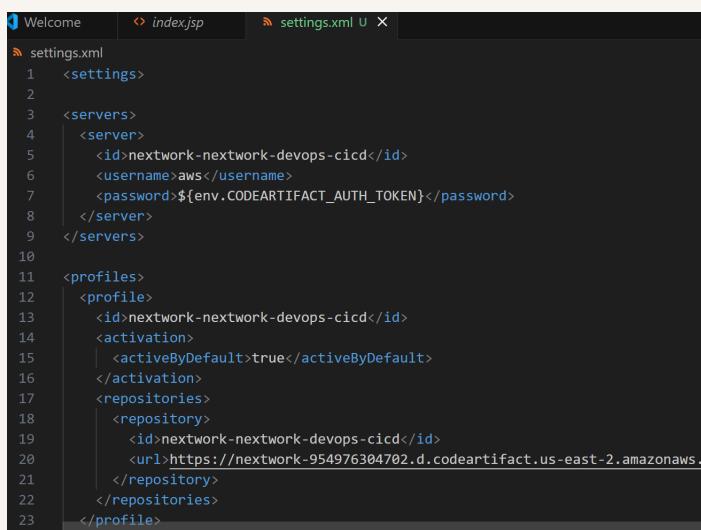
1 ▼ {
2     "Version": "2012-10-17",
3     "Statement": [
4         {
5             "Effect": "Allow",
6             "Action": [
7                 "codeartifact:GetAuthorizationToken",
8                 "codeartifact:GetRepositoryEndpoint",
9                 "codeartifact:ReadFromRepository"
10            ],
11            "Resource": "*"
12        },
13        {
14            "Effect": "Allow",
15            "Action": "sts:GetServiceBearerToken",
16            "Resource": "*",
17            "Condition": {
18                "StringEquals": {
19                    "sts:AWSServiceName": "codeartifact.amazonaws.com"
20                }
21            }
22        }
23    ]
24 }
```

# Maven and CodeArtifact

To test the connection between Maven and CodeArtifact, I compiled my web app using `settings.xml`

The `settings.xml` file configures Maven to use the CodeArtifact repository and supplies Maven with the name and authentication token to get access to the CodeArtifact repository and also sets up a profile section incase we have multiple repositories.

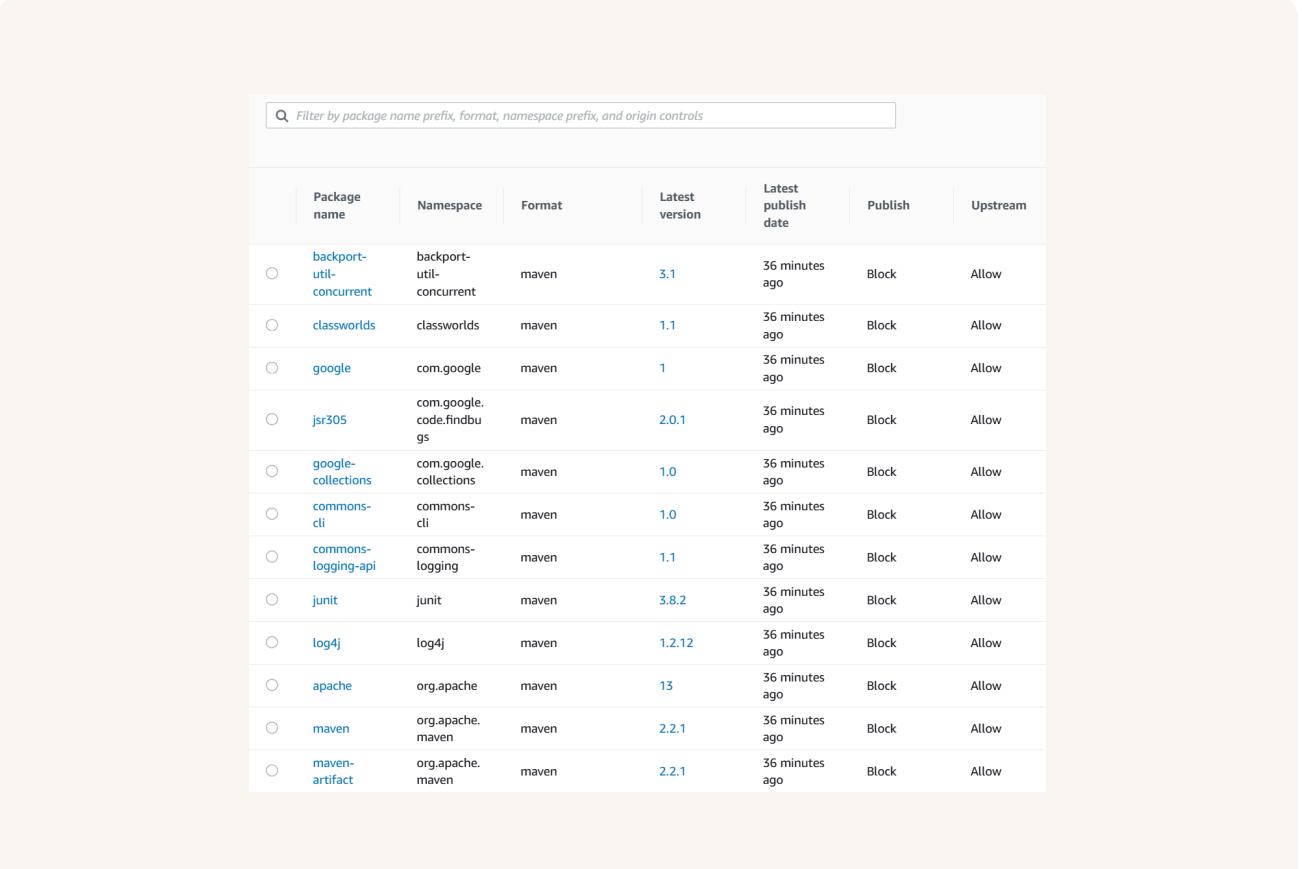
Compiling means a process of translating a project's code into a language that computers can understand and run. Maven is the compiler for this project.



```
1 <settings>
2
3 <servers>
4   <server>
5     <id>nextwork-nextwork-devops-cicd</id>
6     <username>aws</username>
7     <password>${env.CODEARTIFACT_AUTH_TOKEN}</password>
8   </server>
9 </servers>
10
11 <profiles>
12   <profile>
13     <id>nextwork-nextwork-devops-cicd</id>
14     <activation>
15       <activeByDefault>true</activeByDefault>
16     </activation>
17     <repositories>
18       <repository>
19         <id>nextwork-nextwork-devops-cicd</id>
20         <url>https://nextwork-954976304702.d.codeartifact.us-east-2.amazonaws.com/maven</url>
21       </repository>
22     </repositories>
23   </profile>
```

# Verify Connection

After compiling, I checked the CodeArtifact repository and I noticed 4 pages of packages inside. This shows that the web app dependencies was successfully stored in an Artifact repository.



The screenshot shows a table of packages in a CodeArtifact repository. The columns are: Package name, Namespace, Format, Latest version, Latest publish date, Publish, and Upstream. The packages listed are:

	Package name	Namespace	Format	Latest version	Latest publish date	Publish	Upstream
○	backport-util-concurrent	backport-util-concurrent	maven	3.1	36 minutes ago	Block	Allow
○	classworlds	classworlds	maven	1.1	36 minutes ago	Block	Allow
○	google	com.google	maven	1	36 minutes ago	Block	Allow
○	jsr305	com.google.code.findbugs	maven	2.0.1	36 minutes ago	Block	Allow
○	google-collections	com.google.collections	maven	1.0	36 minutes ago	Block	Allow
○	commons-cli	commons-cli	maven	1.0	36 minutes ago	Block	Allow
○	commons-logging-api	commons-logging	maven	1.1	36 minutes ago	Block	Allow
○	junit	junit	maven	3.8.2	36 minutes ago	Block	Allow
○	log4j	log4j	maven	1.2.12	36 minutes ago	Block	Allow
○	apache	org.apache	maven	13	36 minutes ago	Block	Allow
○	maven	org.apache.maven	maven	2.2.1	36 minutes ago	Block	Allow
○	maven-artifact	org.apache.maven	maven	2.2.1	36 minutes ago	Block	Allow



NextWork.org

# **Everyone should be in a job they love.**

Check out nextwork.org for  
more projects

