



nextwork.org

Secure Packages with CodeArtifact



Saish Nar

Packages <small>Info</small>						
	Package name	Namespace	Format	Latest version	Latest publish date	Publish
○	backport-util-concurrent	backport-util-concurrent	maven	3.1	2 minutes ago	Block
○	classworlds	classworlds	maven	1.1	2 minutes ago	Block
○	google	com.google	maven	1	1 minute ago	Block
○	jsr305	com.google.code.findbugs	maven	2.0.1	2 minutes ago	Block
○	google-collections	com.google.collections	maven	1.0	1 minute ago	Block
○	commons-cli	commons-cli	maven	1.0	2 minutes ago	Block
○	commons-logging-api	commons-logging	maven	1.1	1 minute ago	Block
○	junit	junit	maven	3.8.2	1 minute ago	Block
○	log4j	log4j	maven	1.2.12	1 minute ago	Block
○	apache	org.apache	maven	13	2 minutes ago	Block
○	maven	org.apache.maven	maven	2.2.1	2 minutes ago	Block
○	maven-artifact	org.apache.maven	maven	2.2.1	2 minutes ago	Block
○	maven-artifact-manager	org.apache.maven	maven	2.2.1	2 minutes ago	Block
○	maven-core	org.apache.maven	maven	2.2.1	2 minutes ago	Block
○	maven-error-diagnostics	org.apache.maven	maven	2.2.1	2 minutes ago	Block
○	maven-model	org.apache.maven	maven	2.2.1	2 minutes ago	Block



Introducing Today's Project!

In this project, I will demonstrate how to set up CodeArtifact as our CI/CD pipeline's artifact repository! I am doing this project to learn the importance and value of having an artifact repository. It's a huge time saver.

Key tools and concepts

Services I used in this project include CodeArtifact, IAM, EC2, GitHub and VS Code. Key concepts I learnt include using artifact repo's, connecting Maven with CodeArtifact, setting up IAM permissions to give instance permission to access CodeArtifact

Project reflection

This project took me approximately 2 hours to complete. The most challenging part was to compile the settings.xml file, but it finally worked. It was rewarding to learn the extended services of AWS and it was a fun time spent to complete this project

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 in next 24 hours to further explore AWS services and how it can help the development of my WebApp.

Saish Nar
NextWork Student

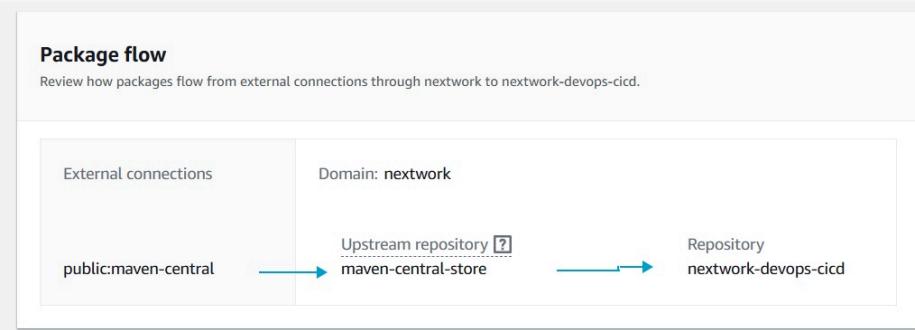
NextWork.org

CodeArtifact Repository

CodeArtifact is an artifact repository service, which means I can use it to create repositories to store my webapp's 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 or under the same organization. They're helpful for setting up permission settings for multiple CodeArtifact repositories in one go. My domain is nextwork.

A CodeArtifact repository can have an upstream repository, which means a public source of packages that Maven can visit if they can't find it in my local CodeArtifact repository. My repository's upstream repository is Maven Central Repository.





CodeArtifact Security

Issue

To access CodeArtiaact I need an authentication token that allows my EC2 instance to access CodeArtifact for 12 hours. I ran into an error when retrieving a token because my EC2 instance does not have permission to access my AWS resources by default

Resolution

To resolve the error with my security token, I set up an IAM Policy that grants access to CodeArtifact, Then I can attach role to EC2.This resolved the error because my EC2 instance now has access to request an authorization token from my repository

It's security best practice to use IAM roles because IAM roles are applied via AWS and do not require and hardcoding of credentials, which is much more secure and encrypted.



Saish Nar
NextWork Student

NextWork.org

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), viewing the packages inside the repository.

The screenshot shows the AWS Policy Editor interface with the JSON tab selected. The policy document is as follows:

```
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 my CodeArtifact repository. It supplies Maven with the name and authentication token to get access to the CodeArtifact repository, and also sets up a profile section in case I ever have multiple repo's.

Compiling means the process of translating my web app source code into something that machines can run. Maven is my compiler, so I asked maven to compile my web app code! Maven will put together all of the packages that web app needs in order to run.

```
src > main > webapp > settings.xml
 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-891377396307.d.codeartifact.ap-south-1.amazonaws.com/maven/nextwork-devops-cicd/</url>
21 |     </repository>
22 |   </repositories>
23 | </profile>
24 </profiles>
25
26 <mirrors>
27 | <mirror>
28 |   <id>nextwork-nextwork-devops-cicd</id>
29 |   <name>nextwork-nextwork-devops-cicd</name>
30 |   <url>https://nextwork-891377396307.d.codeartifact.ap-south-1.amazonaws.com/maven/nextwork-devops-cicd/</url>
31 |   <mirrorOf>*</mirrorOf>
32 | </mirror>
33 </mirrors>
```



Saish Nar
NextWork Student

NextWork.org

Verify Connection

After compiling, I checked my CodeArtifact repository and I noticed 4 pages of packages inside! This tells that I've successfully stored my web app's dependencies in an artifact repository.

Packages info

Filter by package name prefix, format, namespace prefix, and origin controls

Package name	Namespace	Format	Latest version	Latest publish date	Publish
backport-util-concurrent	backport-util-concurrent	maven	3.1	2 minutes ago	Block
classworlds	classworlds	maven	1.1	2 minutes ago	Block
google	com.google	maven	1	1 minute ago	Block
jsr305	com.google.code.findbugs	maven	2.0.1	2 minutes ago	Block
google-collections	com.google.collections	maven	1.0	1 minute ago	Block
commons-cli	commons-cli	maven	1.0	2 minutes ago	Block
commons-logging-api	commons-logging	maven	1.1	1 minute ago	Block
junit	junit	maven	3.8.2	1 minute ago	Block
log4j	log4j	maven	1.2.12	1 minute ago	Block
apache	org.apache	maven	13	2 minutes ago	Block
maven	org.apache.maven	maven	2.2.1	2 minutes ago	Block
maven-artifact	org.apache.maven	maven	2.2.1	2 minutes ago	Block
maven-artifact-manager	org.apache.maven	maven	2.2.1	2 minutes ago	Block
maven-core	org.apache.maven	maven	2.2.1	2 minutes ago	Block
maven-error-diagnostics	org.apache.maven	maven	2.2.1	2 minutes ago	Block
maven-model	org.apache.maven	maven	2.2.1	2 minutes ago	Block



NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for
more projects

