[ENGLISH]

Hello, Cloud Gurus,

and welcome to this lecture,

which is going to introduce CodeArtifact.

I will start off with, what is CodeArtifact?

What are artifacts?

We'll look at integrating CodeArtifact

with public software repositories,

and my exam tips.

So what is CodeArtifact?

Well, it's an artifact repository

that makes it really easy for developers

to find the software packages that they need

to build their software.

And an artifact repository allows you

to securely store, publish, and share artifacts

relating to your application.

So it's mainly used to store, publish,

and share software packages that are used

in your software development process.

And a package is simply a bundle of software.

And this includes open source software

from public repositories,

as well as software developed in-house.

So CodeArtifact is a central repository

that can be used by all your developers

to obtain the correct versions

of the software packages required for their projects.

It integrates with public repositories.

For example, the npm Registry,

for Node.js or JavaScript packages,

the Python Package Index for Python packages,

and Maven Central for commonly used libraries.

And don't worry,

you don't need to memorize

any of these public repositories,

they're just examples.

And of course,

CodeArtifact also integrates

with your CID system as well.

For example, CodeBuild can use CodeArtifact

to grab the correct version of the software needed

to build your application.

So what do we actually mean by artifacts?

Well, it's more than just a software,

and it includes documentation,

compiled applications, deployable packages,

and libraries as well.

Now integration with public repositories

is a really important feature

to understand for the exam.

So by integrating with a public repository,

this makes third party software available for use.

For example, open-source packages from public repositories,

like the npm Registry.

And this allows you to curate

an approved set of packages,

that you can then make available to your organization.

And often, this means the version

that our engineering team is willing to support,

which may not always be the latest version.

Instead, it's normally the version

that's been tested in your organization

and is known to be compatible

with everything else in your environment.

And having everything available in one place,

is an efficient way for developers

to find the approved packages,

and also publish and share their own packages

by adding them to the repository.

Now for the exam,

you will need to understand at a high level,

how to CodeArtifact with an external repository.

So essentially, what you do is create a domain.

And a domain is where you group your repositories

in order to manage them.

Next, you create a repository in your domain,

and this is where your packages will be stored.

And in this example, we've called it my-repo.

Next, you create an upstream repository.

And in this example, we've called it npm-store.

And this is going to connect

to the external public repository.

So you add an external connection

to the public repository,

and this allows you to pull packages down

from that external repository,

and everything will be encrypted in transit,

and at rest.

Next, you associate the npm-store repository

as an upstream repository to the my-repo repository.

And then let's say you've got a developer

who needs to install a package.

They can use the npm CLI to pull in the package

from the public repo,

and then it will be available

for them to use from my-repo.

And in this example,

we're pulling in a package called Express,

which is just a Node.js framework,

used to develop web and mobile applications.

And don't worry

if this all sounds a little bit convoluted,

you only need to understand it at a very high level.

And after this lesson, we've got a demo,

so you will get to see how it all works.

So for the exam,

remember that CodeArtifact is an artifact repository

that makes it easy for developers

to find the software packages that they need

to build their software.

A package is simply a bundle of software used

in your software development process.

And CodeArtifact enables developers

to find approved packages

and also publish their own.

But the main thing that I want you to take away,

is that you can connect

to an external public repository.

And in order to do that,

you just create an upstream repository,

with an external connection,

and that is used to pull the packages

from your external public repository.

But of course, the best way to learn CodeArtifact

is to get your hands dirty.

So if you're ready to do that,

I will see you in the next lecture.

Thank you.

[SPANISH]

Hola, gurús de la nube,

y bienvenidos a esta conferencia,

que va a presentar CodeArtifact .

Comenzaré con, ¿qué es CodeArtifact ?

¿Qué son los artefactos?

Veremos la integración de CodeArtifact

con repositorios públicos de software,

y mis consejos para el examen.

Entonces, ¿qué es CodeArtifact ?

Bueno, es un depósito de artefactos.

eso hace que sea muy fácil para los desarrolladores

para encontrar los paquetes de software que necesitan

para construir su software.

Y un repositorio de artefactos le permite

para almacenar, publicar y compartir artefactos de forma segura

relacionado con su aplicación.

Por lo tanto, se utiliza principalmente para almacenar, publicar,

y compartir paquetes de software que se utilizan

en su proceso de desarrollo de software.

Y un paquete es simplemente un paquete de software.

Y esto incluye software de código abierto.

de repositorios públicos,

así como software desarrollado internamente.

Entonces CodeArtifact es un repositorio central

que pueden usar todos sus desarrolladores

para obtener las versiones correctas

de los paquetes de software necesarios para sus proyectos.

Se integra con repositorios públicos.

Por ejemplo, el Registro npm ,

para paquetes Node.js o JavaScript,

el índice de paquetes de Python para paquetes de Python,

y Maven Central para bibliotecas de uso común.

Y no te preocupes,

no necesitas memorizar

cualquiera de estos repositorios públicos,

son solo ejemplos.

Y por supuesto,

CodeArtifact también integra

con su sistema CID también.

Por ejemplo, CodeBuild puede usar CodeArtifact

para obtener la versión correcta del software necesario

para construir su aplicación.

Entonces, ¿qué entendemos realmente por artefactos?

Bueno, es más que un simple software,

e incluye documentación,

compiladas , paquetes desplegables,

y bibliotecas también.

Ahora integración con repositorios públicos

es una característica muy importante

entender para el examen.

Entonces, al integrarse con un repositorio público,

esto hace que el software de terceros esté disponible para su uso.

Por ejemplo, paquetes de código abierto de repositorios públicos,

como el Registro npm .

Y esto le permite curar

un conjunto aprobado de paquetes,

que luego puede poner a disposición de su organización.

Y a menudo, esto significa que la versión

que nuestro equipo de ingeniería está dispuesto a apoyar,

que no siempre puede ser la última versión.

En cambio, normalmente es la versión

que ha sido probado en su organización

y se sabe que es compatible

con todo lo demás en su entorno.

Y tener todo disponible en un solo lugar,

es una manera eficiente para los desarrolladores

para encontrar los paquetes aprobados,

y también publicar y compartir sus propios paquetes

agregándolos al repositorio.

Ahora para el examen,

necesitarás entender a un alto nivel,

cómo CodeArtifact con un repositorio externo.

Básicamente, lo que haces es crear un dominio.

Y un dominio es donde agrupas tus repositorios

para gestionarlos.

A continuación, crea un repositorio en su dominio,

y aquí es donde se almacenarán sus paquetes.

Y en este ejemplo, lo hemos llamado my-repo.

A continuación, crea un repositorio ascendente.

Y en este ejemplo, lo llamamos npm -store.

Y esto se va a conectar

al repositorio público externo.

Entonces agregas una conexión externa

al depósito público,

y esto le permite bajar los paquetes

de ese repositorio externo,

y todo será encriptado en tránsito,

y en reposo.

A continuación, asocia el repositorio npm -store

como un repositorio ascendente al repositorio my-repo.

Y luego digamos que tienes un desarrollador

que necesita instalar un paquete.

Pueden usar la CLI de npm para extraer el paquete

del repositorio público,

y entonces estará disponible

para que lo usen desde my-repo.

Y en este ejemplo,

estamos sacando un paquete llamado Express,

que es solo un marco Node.js,

Se utiliza para desarrollar aplicaciones web y móviles.

y no te preocupes

si todo esto suena un poco complicado,

necesitas entenderlo a un nivel muy alto.

Y después de esta lección, tenemos una demostración,

así podrás ver cómo funciona todo.

Así que para el examen,

recuerda que CodeArtifact es un repositorio de artefactos

que hace que sea fácil para los desarrolladores

para encontrar los paquetes de software que necesitan

para construir su software.

Un paquete es simplemente un paquete de software utilizado

en su proceso de desarrollo de software.

Y CodeArtifact permite a los desarrolladores

para encontrar paquetes aprobados

y también publicar los suyos propios.

Pero lo principal que quiero que te lleves,

es que se puede conectar

a un repositorio público externo.

Y para hacer eso,

crea un repositorio ascendente,

con una conexión externa,

y que sirve para jalar los paquetes

de su repositorio público externo.

Pero claro, la mejor forma de aprender CodeArtifact

es ensuciarse las manos.

Así que si estás listo para hacer eso,

Te veré en la próxima conferencia.

Gracias.