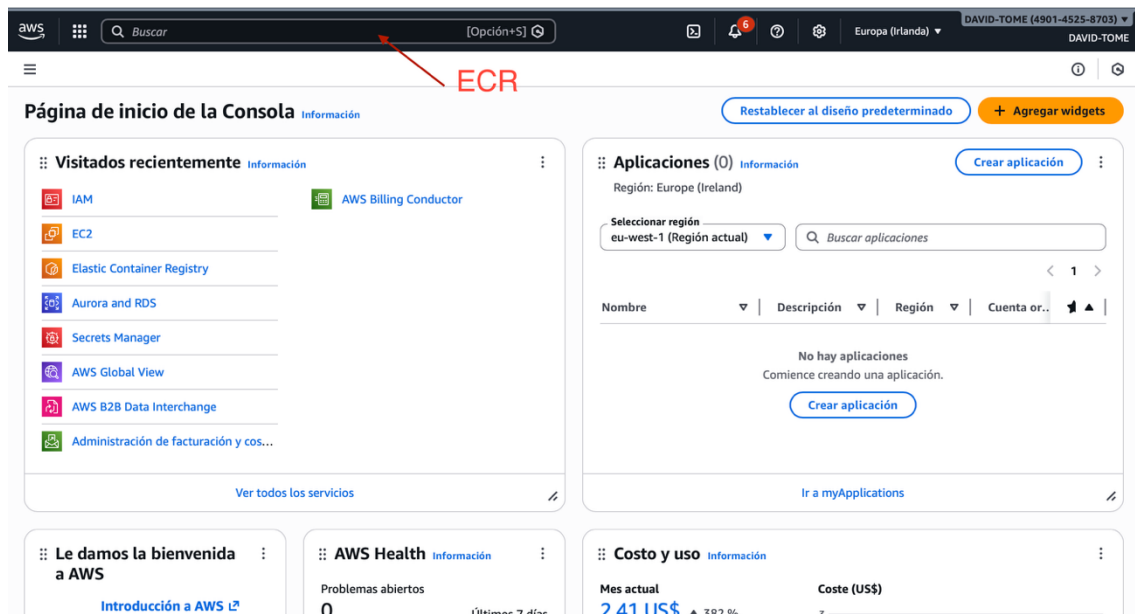


Paso 4 AWS

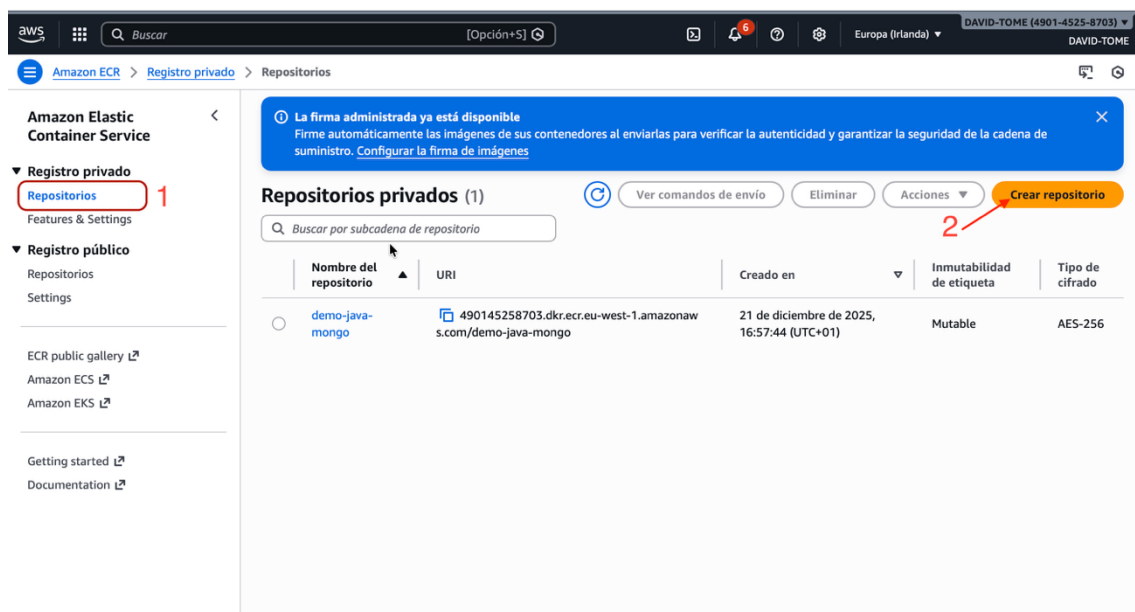
Lo primero que tienes que hacer es crearte una cuenta de AWS. Una vez que la tengas sigue los siguientes pasos:

1º Iniciamos sesión por consola desde la Web.

2º En el buscador superior escribimos “ECR”.



En el menú lateral damos a Repositorios y después al botón de “crear repositorio”



Le damos un nombre distinto al que tengamos en otras variables para diferenciar. Dejamos marcada la opción de “Mutable” y damos a crear el repositorio.

Crear repositorio privado

Configuración general

Nombre del repositorio
Introduzca un nombre conciso. Los repositorios admiten espacios de nombres, que puede usar para agrupar repositorios similares.
490145258703.dkr.ecr.eu-west-1.amazonaws.com/ecr-manual 1
10 de 256 caracteres máximo (2 mínimo). El nombre debe comenzar con una letra y solo puede contener letras minúsculas, números y caracteres especiales _./.

Configuración de etiquetas de imagen Información

Mutabilidad de la etiqueta de imagen
Elija la configuración de mutabilidad de las etiquetas.
☒ **Mutable** 2
Las etiquetas de imagen se pueden sobrescribir.
☐ **Immutable**
Las etiquetas de imagen no se pueden sobrescribir.

Exclusiones de etiquetas mutables
Las etiquetas que coincidan con estos filtros serán inmutables (no se podrán sobrescribir). El uso de comodines (*) coincidirá con cero o más caracteres de la etiqueta de la imagen.
Agregar filtro
Los filtros únicamente deben contener letras, números y caracteres especiales (_.-). Cada filtro está limitado a 128 caracteres, 2 comodines (*) y puede agregar hasta 5 filtros en la lista de exclusiones.

Configuración de cifrado Información

Una vez creado nos aparece esto:

Aquí obtenemos el valor de la variable, en mi caso: ECR_REPOSITORY= “ecr-manual”.

<input type="radio"/> ecr-manual	490145258703.dkr.ecr.eu-west-1.amazonaws.com/ecr-manual	30 de diciembre de 2025, 14:10:40 (UTC+01)	Mutable	AES-256
----------------------------------	---	--	---------	---------

Ahora vamos a obtener nuestra variable AWS_REGION.

En el menú principal de nuestra consola de AWS vemos en la parte superior una localización, en mi caso “Europa-Irlanda”. Desplegamos esa opción.

Página de inicio de la Consola Información

Restablecer al diseño predeterminado + Agregar widgets

Visitados recientemente Información

- Elastic Container Registry
- IAM
- EC2
- Aurora and RDS
- Secrets Manager
- AWS Global View
- AWS B2B Data Interchange
- Administración de facturación y cos...

Ver todos los servicios

Aplicaciones (0) Información

Región: Europe (Ireland)

Seleccionar región: eu-west-1 (Región actual)

Buscar aplicaciones

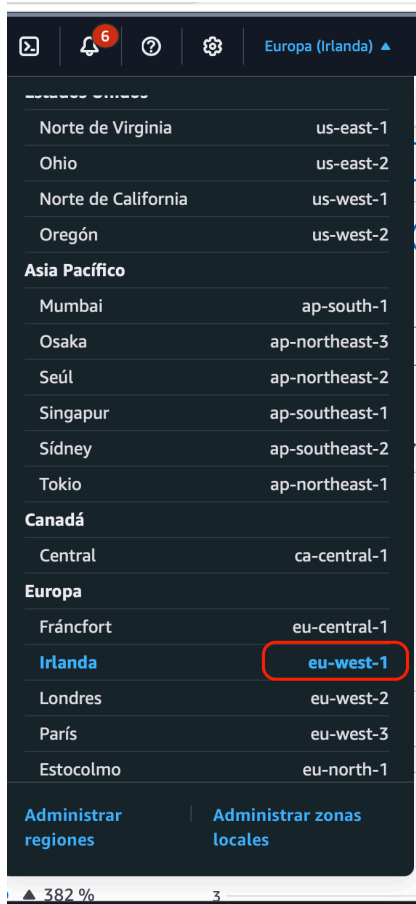
No hay aplicaciones. Comience creando una aplicación.

Crear aplicación

Ir a myApplications

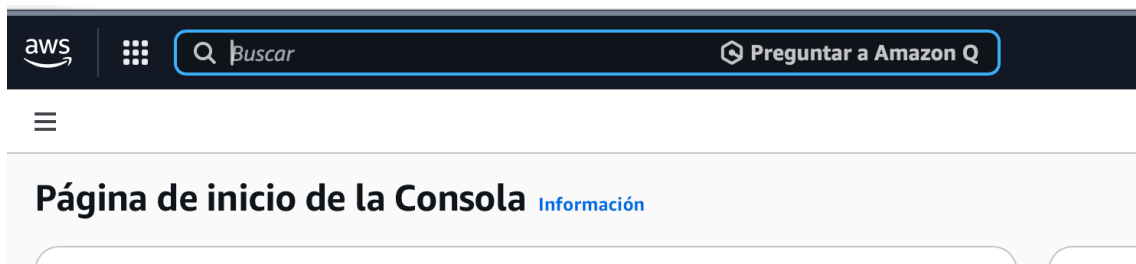
Vemos que nuestra opción su valor es “eu-west-1”, es decir, nuestro secrets en git será:

`AWS_REGION = eu-west-1`

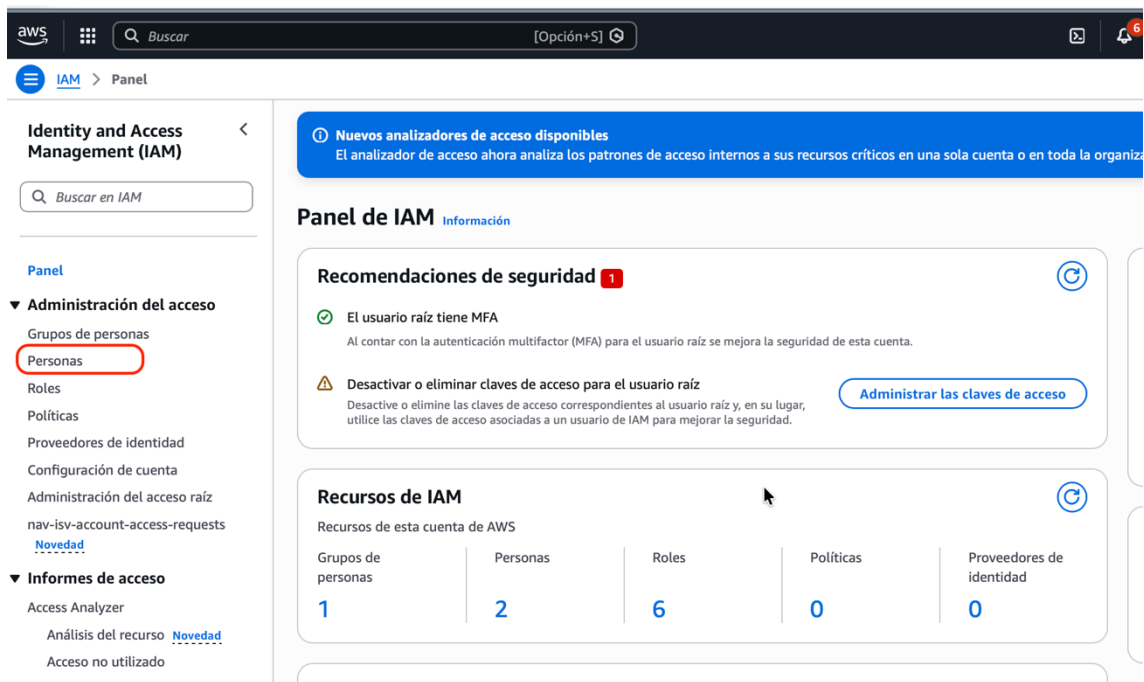


Ya tenemos el nombre del repositorio y la región. Vamos a por los otros 2 que nos quedan.

En el menú superior escribimos. “IAM”.



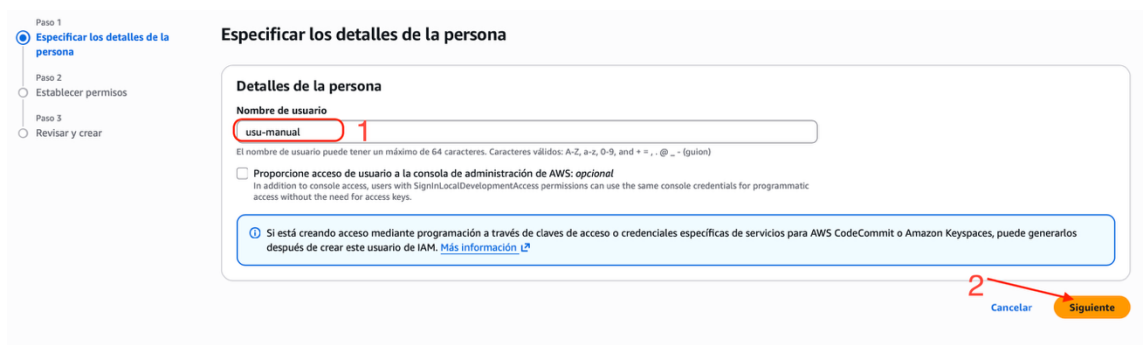
Se nos abre esa ventana, en el menú lateral damos a personas.



Pulsamos a crear persona



Le damos un nombre al usuario y luego pulsamos en siguiente.



En la siguiente ventana seleccionamos la opción de “adjuntar políticas directamente”.

Buscamos esta: AmazonEC2ContainerRegistryPowerUser y la marcamos.

Después damos a siguiente → crear persona.

Establecer permisos
Agregue una persona a un grupo existente o cree uno nuevo. El uso de grupos es una práctica recomendada para administrar los permisos de usuario según las funciones laborales. [Más información](#)

Opciones de permisos

- ☐ Agregar persona al grupo
Agregue la persona a un grupo existente o cree uno nuevo. Le recomendamos que utilice grupos para administrar los permisos de usuario según las funciones laborales.
- ☐ Copiar permisos
Copie todas las suscripciones a grupos, las políticas administradas adjuntas y las políticas insertadas de una persona existente.
- ☒ **Adjuntar políticas directamente**
Adjunte una política administrada a una persona de manera directa. Como práctica recomendada, le sugerimos, en cambio, adjuntar políticas a un grupo. A continuación, agregue la persona al grupo adecuado.

Políticas de permisos (1440)
Elija una o varias políticas para asociarlas a la nueva persona. [Crear política](#)

2 1 **Filtrar por Tipo** Todos los tipos 1 coincidencia

3 <input type="checkbox"/>	Nombre de la política	Tipo	Entidades asociadas
<input checked="" type="checkbox"/>	AmazonEC2ContainerRegistryPowerUser	Administrada por AWS	0

También agrega este otro permiso: AmazonEC2ContainerRegistryReadOnly y AmazonEC2FullAccess

Políticas de permisos (3) [Eliminar](#) [Agregar permisos](#)

Los permisos se definen mediante políticas asociadas a la persona directamente o a través de grupos.

Filtrar por Tipo Todos los tipos

<input type="checkbox"/>	Nombre de la política	Tipo	Adjuntado a través de
<input checked="" type="checkbox"/>	AmazonEC2ContainerRegistryPowerUser	Administrada por AWS	Directamente
<input checked="" type="checkbox"/>	AmazonEC2ContainerRegistryReadOnly	Administrada por AWS	Directamente
<input checked="" type="checkbox"/>	AmazonEC2FullAccess	Administrada por AWS	Directamente

Ya lo tenemos creado, ahora entramos en el.

Personas (2) [Información](#) [Eliminar](#) [Crear persona](#)

Un usuario de IAM es una identidad con credenciales válidas a largo plazo que se utiliza para interactuar con AWS en una cuenta.

Buscar

<input type="checkbox"/>	Nombre de usuario	Ruta	Grupos	Última actividad	MFA	Antigüedad de	Último inicio de sesión	ID de clave
<input checked="" type="checkbox"/>	tfg-cicd-user	/	1	✓ hace 18 horas	-	-	-	Active - AKI
<input type="checkbox"/>	usu-manual	/	0	-	-	-	-	-

Cuando se nos abra el usuario, vamos a la pestaña de credenciales de seguridad

usu-manual [Información](#) [Eliminar](#)

Resumen

ARN arn:aws:iam::490145258703:user/usu-manual	Acceso a la consola Desactivada	Clave de acceso 1 Crear clave de acceso
Creado December 30, 2025, 14:53 (UTC+01:00)	Último inicio de sesión en la consola -	

[Permisos](#) [Grupos](#) [Etiquetas](#) **Credenciales de seguridad** [Último acceso](#)

Políticas de permisos (1) [Eliminar](#) [Agregar permisos](#)

Los permisos se definen mediante políticas asociadas a la persona directamente o a través de grupos.

Filtrar por Tipo [Todos los tipos](#)

<input type="checkbox"/> Nombre de la política i	<input type="checkbox"/> Tipo	<input type="checkbox"/> Adjuntado a través de i
<input type="checkbox"/> AmazonEC2ContainerRegistryPowerUser	Administrada por AWS	Directamente

Bajamos hasta que veamos esta opción y pulsamos en “crear clave de acceso”

Claves de acceso (0) [Crear clave de acceso](#)

Utilice las claves de acceso para enviar llamadas mediante programación a AWS desde AWS CLI, Herramientas de AWS para PowerShell, AWS SDK o llamadas directas a la API de AWS. Puede tener un máximo de dos claves de acceso (activas o inactivas) a la vez. [Más información](#)

No hay claves de acceso. Como práctica recomendada, evite el uso de credenciales a largo plazo, como las claves de acceso. En su lugar, utilice herramientas que proporcionen credenciales a corto plazo. [Más información](#)

[Crear clave de acceso](#)

Seleccionamos la opción “interfaz de línea de comandos (CLI)” → siguiente.--> crear clave de acceso.

[IAM](#) > [Personas](#) > [usu-manual](#) > Crear clave de acceso

☐ Establecer el valor de etiqueta de descripción

Paso 3

☐ Recuperar claves de acceso

Caso de uso

☒ **Interfaz de línea de comandos (CLI)**
Tiene previsto utilizar esta clave de acceso para permitir que la AWS CLI obtenga acceso a su cuenta de AWS.

☐ **Código local**
Tiene previsto utilizar esta clave de acceso para habilitar el código de aplicación en un entorno de desarrollo local para obtener acceso a su cuenta de AWS.

☐ **Aplicación ejecutada en un servicio de computación de AWS**
Tiene previsto utilizar esta clave de acceso para permitir que el código de aplicación que se ejecuta en un servicio de computación de AWS como Amazon EC2, Amazon ECS o AWS Lambda obtenga acceso a su cuenta de AWS.

☐ **Servicio de terceros**
Tiene previsto utilizar esta clave de acceso para habilitar el acceso a una aplicación o servicio de terceros que supervise o administre sus recursos de AWS.

☐ **Aplicación ejecutada fuera de AWS**
Planea usar esta clave de acceso para autenticar las cargas de trabajo que se ejecutan en su centro de datos u otra infraestructura externa a AWS que necesitan acceder a sus recursos de AWS.

☐ **Otros**
Su caso de uso no aparece aquí.

Alternativas recomendadas

- Use AWS CLI V2 and the `aws login` command to use your existing console credentials in the CLI. [Más información](#)
- Use AWS CloudShell, a browser-based CLI, to run commands. [Más información](#)

Confirmación

☐ Entiendo la recomendación anterior y deseo proceder a la creación de una clave de acceso.

[Cancelar](#) [Siguiente](#)

Aquí vamos a encontrar las dos variables que nos faltan:

AWS_ACCESS_KEY_ID = AKIAXEHPDTH6O4WWOUV

AWS_SECRET_ACCESS_KEY = “CLAVE DE ACCESO SECRETA”

Ojo!!! AWS_SECRET_ACCESS_KEY solo se ve una vez.

Este es el único momento en el que se puede ver o descargar la clave de acceso secreta. No podrá recuperarla posteriormente. Sin embargo, puede crear una nueva clave de acceso en cualquier momento.

Paso 1
● Prácticas recomendadas y alternativas para la clave de acceso

Paso 2 - opcional
● Establecer el valor de etiqueta de descripción

Paso 3
● **Recuperar claves de acceso**

Recuperar claves de acceso Información

Clave de acceso
Si pierde u olvida la clave de acceso secreta, no podrá recuperarla. En su lugar, cree una nueva clave de acceso y deje inactiva la antigua.

Clave de acceso | Clave de acceso secreta

1 AKIAEHXPDTG04WWOUIV 2 ***** [Mostrar](#)

Prácticas recomendadas para la clave de acceso

- Nunca almacene la clave de acceso en texto plano, en un repositorio de código o en el código.
- Desactive o elimine la clave de acceso cuando ya no sea necesaria.
- Habilite los permisos con privilegios mínimos.
- Rote con regularidad las claves de acceso.

Para obtener más información sobre cómo administrar las claves de acceso, consulte las [prácticas recomendadas para administrar las claves de acceso de AWS](#).

[Descargar archivo.csv](#) [Listo](#)

Para obtener la 5ª variable en este paso, en la esquina superior derecha, en nuestro nombre, aparece nuestro identificador, aparece entre -, algo así:

David (4971-6543-xxxx), copia ese número sin los guiones, esta variable se va a llamar:

AWS_ACCOUNT_ID= 49716543xxxx

Ya tendríamos las 5 variables secretas que necesitamos para completar el paso 4:

AWS_ACCESS_KEY_ID

AWS_SECRET_ACCESS_KEY

AWS_REGION

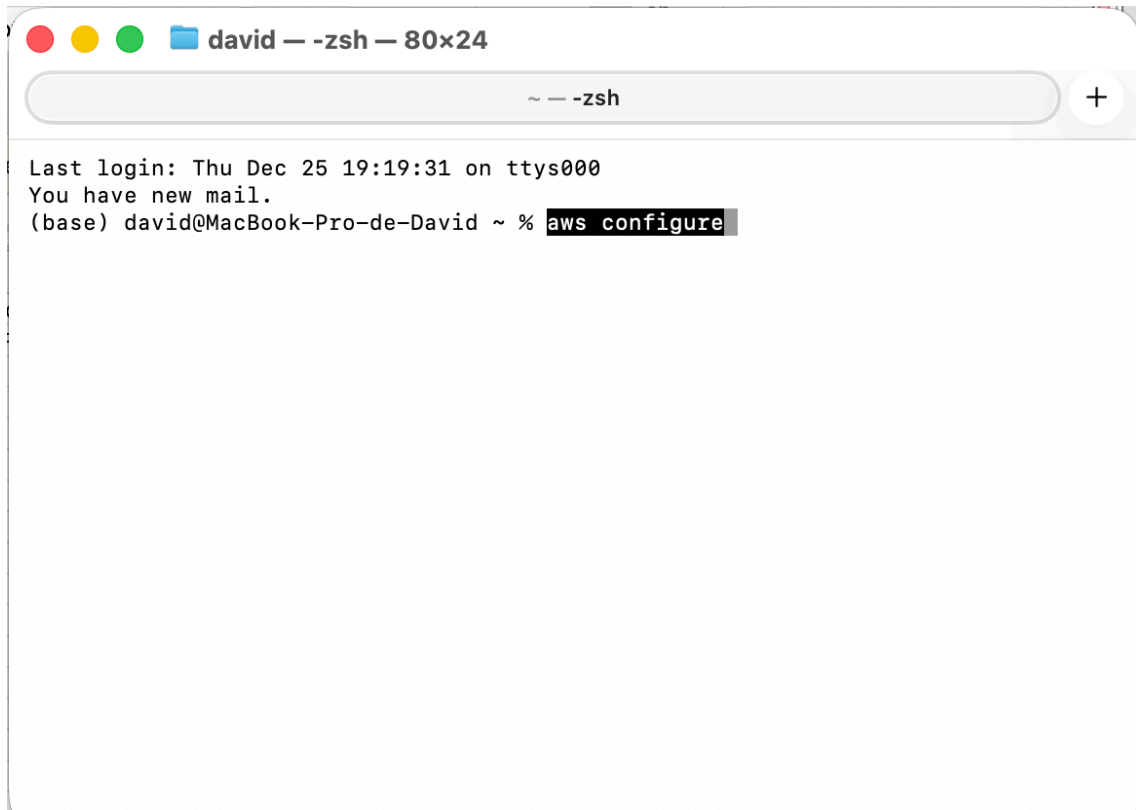
ECR_REPOSITORY

AWS_ACCOUNT_ID

Ahora vamos a configurar nuestro PC con las credenciales de este usuario para que nos reconozca.

Abrimos nuestro terminal y escribimos el comando:

“Aws configure”

A screenshot of a macOS terminal window. The title bar shows 'david — -zsh — 80x24'. The terminal content shows the last login time, a notification about new mail, and the prompt '(base) david@MacBook-Pro-de-David ~ %' with the command 'aws configure' entered and highlighted by a cursor.

```
david — -zsh — 80x24
~ — -zsh
Last login: Thu Dec 25 19:19:31 on ttys000
You have new mail.
(base) david@MacBook-Pro-de-David ~ % aws configure
```

Rellenamos estos 4 campos:

AWS Access Key ID [*****T7UG]:

AWS Secret Access Key [*****XRD8]:

Default region name [eu-west-1]:

Default output format [json]:


```
david — -zsh — 80x24
~ — -zsh
Last login: Thu Dec 25 19:19:31 on ttys000
You have new mail.
(base) david@MacBook-Pro-de-David ~ % aws configure
AWS Access Key ID [*****T7UG]: AKIAXEHXPDTHRHE4ZPEV
AWS Secret Access Key [*****XRD8]: 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
Default region name [eu-west-1]: eu-west-1
Default output format [json]: json
(base) david@MacBook-Pro-de-David ~ %
```

Ahora insertalas en tu git como variables secretas y después haz un commit

git commit --allow-empty -m "creación de variables ECR"

git push

Si hemos escrito bien las variables en nuestro git, nos debería salir lo siguiente:

The screenshot shows the GitHub Actions interface for a workflow named "creación de variables ECR #6". The workflow is triggered by a push to the main branch. The status is "Success" with a total duration of 1m 7s. The workflow consists of four jobs: build_and_test (12s), sonar (40s), build_and_push_ecr (48s), and deploy_ec2. The workflow file is named "generated-ci.yml" and is located in the ".github/workflows" directory.

Si falla, puede ser que haya algún espacio al crear las variables secretas, volverlas a poner de nuevo.

Si entramos en nuestra consola de AWS y vamos al apartado de ECR y entramos en nuestro repositorio, veremos que se nos ha creado nuestra imagen

The screenshot shows the AWS ECR console for the "ecr-manual" repository. The "Imágenes" tab is selected, showing a list of images. The first image is "latest" with a SHA hash of "65af88e0bd7b8295686c9aec1c2a3ea114d8041a". The image was created on December 30, 2025, at 15:18:02 (UTC+01) and has a size of 147.18 MB. The console also shows the repository tags and the "Ver comandos de envío" button.

Ahora vamos a nuestra App y comprobamos el paso 4, metiendo el nombre del repositorio y la etiqueta. Si sigue poniendo KO en la comprobación y hemos seguido todos los pasos correctamente, reinicia la App y vuelve a comprobarlo. Ahora debería salir esto

Paso 4 – Comprobar imagen en ECR

Ver guía de este paso

Se comprueba que existe una imagen Docker en tu repositorio de Amazon ECR con el **nombre** y el **tag** indicados.

Repositorio ECR

ecr-manual

Debe coincidir con el nombre del repositorio en ECR (sin URL).

Tag de la imagen

latest

Comprobar

Estado: OK

Imagen encontrada en ECR. Digest: sha256:f62634e88954a3fc4e15a5ba2cedc391028d9f6c963d729014406229a206699b

[← Volver al paso 3](#)

[Ir al paso 5 →](#)