

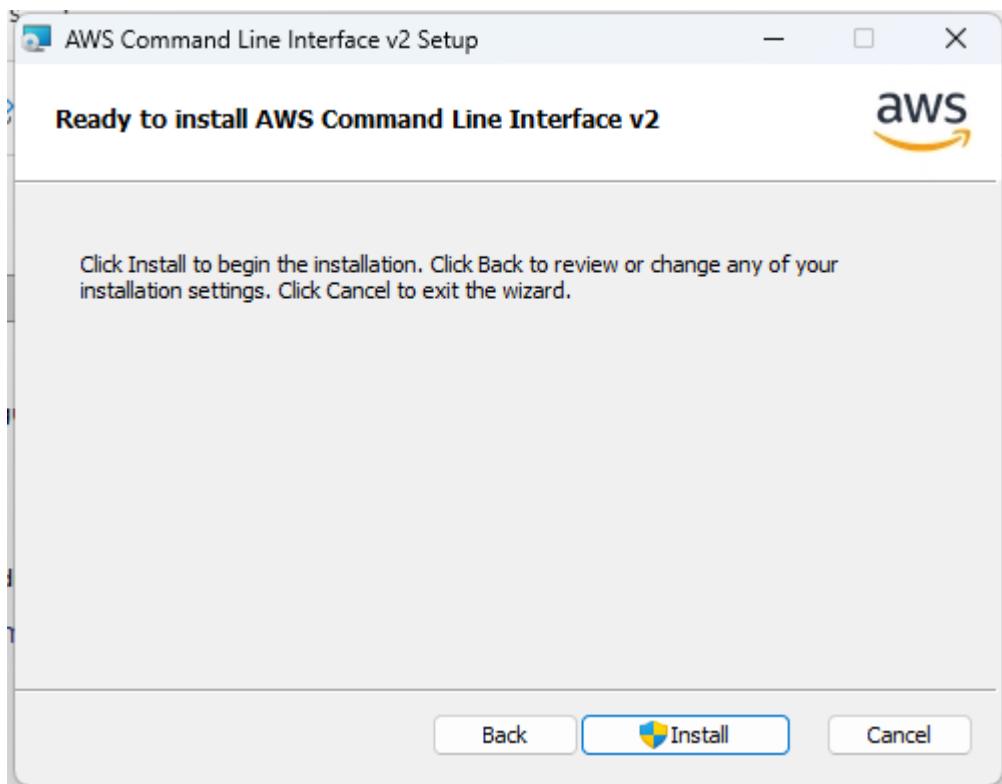
PR_07.2 Dani Gayol Rodríguez

PR_07.2 Dani Gayol Rodríguez.....	1
AWS CLI	1
1.) Instalación de AWS CLI	2
2.) Configuración AWS CLI	2
¿Como lo haríamos nosotros?	4
3.) Uso de AWS CLI.....	6

AWS CLI

1.) Instalación de AWS CLI

El primer paso es descargar el instalador que hay en el enlace del enunciado



Una vez instalado, entramos en PowerShell en modo administrador y escribimos el siguiente comando

```
PS C:\WINDOWS\system32> Set-ExecutionPolicy Unrestricted
Cambio de directiva de ejecución
La directiva de ejecución te ayuda a protegerte de scripts en los que no confías. Si cambias dicha directiva, podrías exponerte a los riesgos de seguridad descritos en el tema de la Ayuda about_Execution_Policies en https://go.microsoft.com/fwlink/?LinkId=135170. ¿Quieres cambiar la directiva de ejecución?
[S] Sí [O] No
[5] Sí a todo [N] No a todo [U] Suspender [?] Ayuda (el valor predeterminado es "N"): si
[5] Sí [O] Sí a todo [N] No a todo [U] Suspender [?] Ayuda (el valor predeterminado es "N"): o
```

Y finalmente, hacemos lo siguiente para comprobar la versión

```
PS C:\Users\Mañana> aws --version
aws-cli/2.32.19 Python/3.13.11 Windows/11 exe/AMD64
```

2.) Configuración AWS CLI

Primero de todo, tenemos que entrar en “IAM” y una vez dentro, ir a la sección de “Personas” y crear un nuevo usuario



Identity and Access Management (IAM)

 Buscar en IAM

Panel

▼ Administración del acceso

Grupos de personas

Personas

Roles

Políticas

Proveedores de identidad

Configuración de cuenta

Administración del acceso raíz

nav-isv-account-access-requests

Novedad

Personas (0) Información

Un usuario de IAM es una identidad con credenciales válidas a largo plazo que se puede usar para acceder a los servicios de AWS.

 Buscar

Nombre de usuario



Ruta



Detalles de la persona

Nombre de usuario

 mycli

El nombre de usuario puede tener un máximo de 64 caracteres. Caracteres válidos: A-Z, a-z, 0-9, and + = , . @ _ - (guion).

Proporcione acceso de usuario a la consola de administración de AWS: *opcional*

In addition to console access, users with SignInLocalDevelopmentAccess permissions can use the same console credentials for programmatic access without the need for access keys.

Contraseña de la consola

Contraseña generada automáticamente

Puede ver la contraseña después de crear la persona.

Contraseña personalizada

Ingrese una contraseña personalizada para la persona.

 Mostrar contraseña

Las personas deben crear una nueva contraseña en el siguiente inicio de sesión (recomendado).

Las personas obtienen automáticamente la IAMUserChangePassword política para poder cambiar su propia contraseña.

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 (1/1446)

Elija una o varias políticas para asociarlas a la nueva persona.

Filtrar por Tipo			
<input type="text"/> Buscar	Todos los tipos	< 1 2 3 4 5 6 7 ... 73 >	
<input type="checkbox"/> Nombre de la política <input type="checkbox"/>	Tipo	<input type="checkbox"/> Entidades asociadas	<input type="checkbox"/>
<input type="checkbox"/> AccessAnalyzerServiceRolePolicy	Administrada por AWS	0	<input type="checkbox"/>
<input type="checkbox"/> AccountManagementFromVercel	Administrada por AWS	0	<input type="checkbox"/>
<input checked="" type="checkbox"/> AdministratorAccess	Administrada por AWS: función de trabajo	1	<input type="checkbox"/>

En este caso, no tengo permisos para crear usuarios, por lo tanto, no lo puedo hacer

No se ha creado la persona.
User arnawssts:716757242964assumed-role/voclabs/user3935192=Daniel_Gayol_Rodr_guez is not authorized to perform: iam:CreateUser on resource: arn:aws:iam::716757242964:user/mycli because no identity-based policy allows the iam:CreateUser action

¿Como lo haríamos nosotros?

Nosotros al tener un laboratorio, tenemos que hacerlo de una manera distinta, para ello usamos el comando “aws configure” y ponemos lo siguiente;

```
C:\Users\Mañana>aws configure
AWS Access Key ID [None]:
AWS Secret Access Key [None]:
Default region name [None]: us-east-1
Default output format [None]: json
```

Una vez hecho esto, buscamos la carpeta “.aws”

```
C:\Users\Mañana>dir
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: 1494-2731

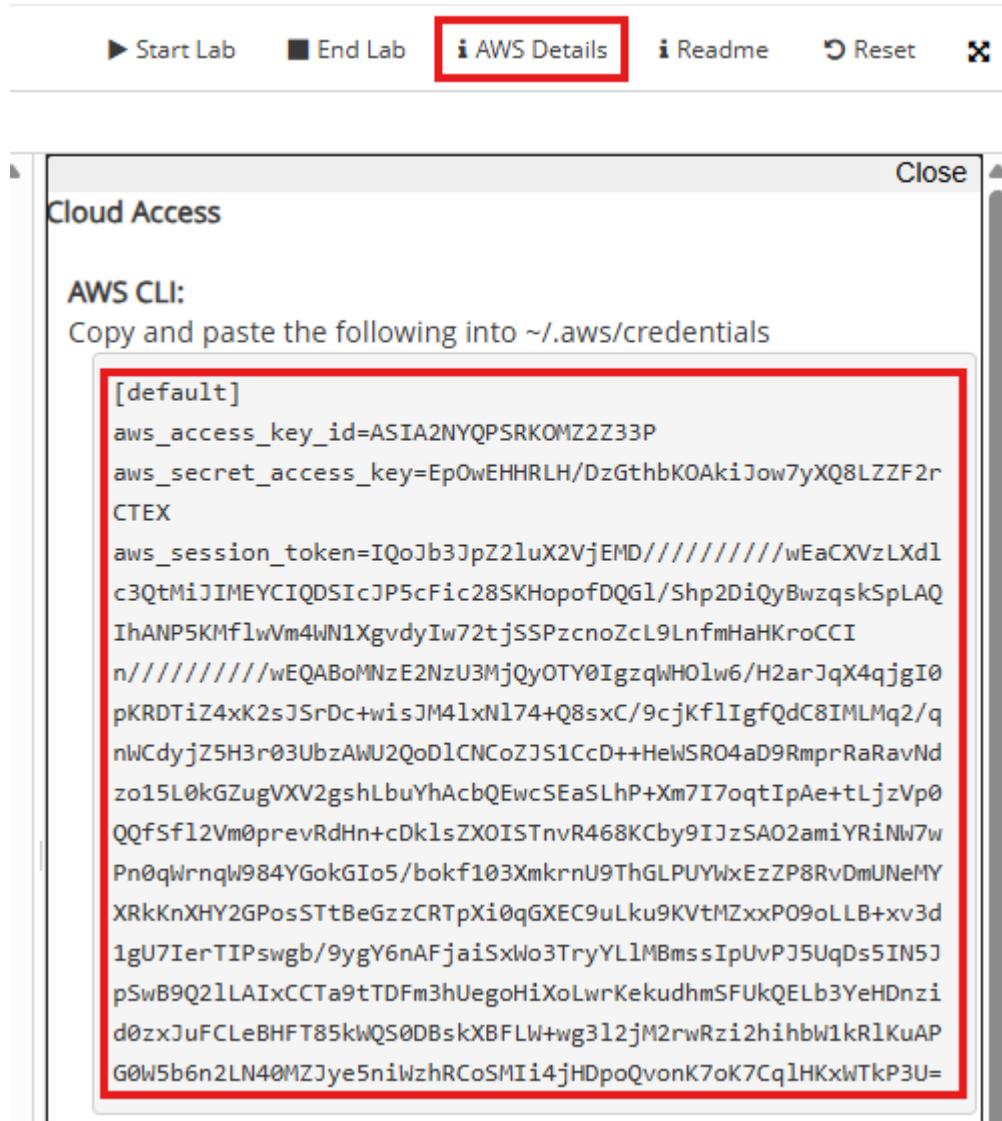
Directorio de C:\Users\Mañana

08/01/2026  09:33    <DIR>          .
15/09/2025  17:48    <DIR>          ..
13/11/2025  10:52    <DIR>          .anaconda
08/01/2026  09:33    <DIR>          .aws
08/01/2026  08:58            9.685 .bash_history
21/11/2025  12:23    <DIR>          .conda
13/11/2025  10:53            25  .condarc
13/11/2025  10:51    <DIR>          .continuum
```

Después, nos vamos a esa carpeta y ponemos el comando “code credentials”

```
C:\Users\Mañana>cd .aws  
C:\Users\Mañana\.aws>code credentials
```

Ahora nos dirigimos al laboratorio de AWS y hacemos lo siguiente;



Finalmente, pegamos el texto en el archivo que abrimos al hacer “code credential” y lo guardamos

The screenshot shows a terminal window with the title bar "credentials X". The command "C: > Users > Mañana > .aws > credentials" is entered. The file content is displayed, showing the AWS CLI configuration text with the first line "1 [default]" and the "aws_access_key_id" line highlighted with a red rectangle.

```
1 [default]  
2 aws_access_key_id=ASIA2NYQPSRKOMZ2Z33P  
3 aws_secret_access_key=Ep0wEHHRLH/DzGthbKOAKiJow7yXQ8LZZF2rCTEX  
4 aws_session_token=IQoJb3JpZ2luX2VjEMD//////////wEaCXVzLXd1c3QtMijIMEYC
```

Este proceso tenemos que repetirlo cada vez que arranquemos el laboratorio ya que se reinicia cada 4 horas

3.) Uso de AWS CLI

Lista de todas las instancias de EC2

```
C:\Users\Mañana\.aws>aws ec2 describe-instances --filters "Name=instance-type,Values=t2.micro"
{
    "Reservations": [
        {
            "ReservationId": "r-066c7cd9b0f499bd7",
            "OwnerId": "716757242964",
            "Groups": [],
            "Instances": [
                {
                    "Architecture": "x86_64",
                    "BlockDeviceMappings": [
                        {
                            "DeviceName": "/dev/sda1",
                            "Ebs": {
                                "AttachTime": "2025-11-24T10:02:29+00:00",
                                "DeleteOnTermination": true,
                                "Status": "attached",
                                "VolumeId": "vol-0f77deab530a832d0"
                            }
                        }
                    ],
                    "ClientToken": "1f8e5ba6-96ec-489a-8446-2856f3ba980f",
                    "EbsOptimized": false,
                    "EnaSupport": true,
                    "Hypervisor": "xen",
                    "NetworkInterfaces": [
                        {
                            "Association": {
                                "IpOwnerId": "716757242964",
                                "PublicDnsName": "ec2-54-86-196-253.compute-1.amazonaws.com",
-- Más -- |
```

Extraer el “ImageId” sin todo lo demás

```
C:\Users\Manana\.aws>aws ec2 describe-images --filters "Name=name,Values=amzn2-ami-hvm-2.0.202*x86_64-gp2" --query "Images[0].ImageId"
"ami-0156001f0548e90b1"
```

Para que la salida no aparezca entre comillas hacemos lo siguiente;

```
C:\Users\Mañana\.aws>aws ec2 describe-images --filters "Name=name,Values=amzn2-ami-hvm-2.0.202*x86_64-gp2" --query "Images[0].ImageId" --output text  
ami-0156001f0548e90b1
```