



Lab - Protección de datos mediante encriptación

Presentación realizada por Brendon Buriol, Paulo Sena, Ignivé Amaro y Valeria Cantoni



Objetivos

Después de completar esta práctica de laboratorio, debería poder:

- Crear una clave de cifrado de AWS KMS
- Instalar la CLI de cifrado de AWS
- Cifrar texto sin formato
- Descifrar texto cifrado

Tarea 1: Crear una clave de AWS KMS

Crear una clave KMS simétrica y le otorgó la propiedad de esa clave al rol voclabs IAM que se creó previamente para este laboratorio.

En el primer paso, buscaremos dentro de Servicios, las siglas “KMS”. Ingresaremos a KMS y luego presionamos sobre Create key.



- En Key type, seleccione Symmetric (Simétrica). El cifrado Simétrico usa la misma clave para cifrar y descifrar datos, lo que hace que sea fácil y eficiente de usar.
- Luego en Key usage selecciona Encrypt and decrypt para que puedas cifrar y descifrar la información fácilmente.

The screenshot shows the AWS Management Console interface for creating a new key. The top navigation bar includes the AWS logo, a 'Services' menu, a search bar, and the user's account information (Oregon, voclabs/user2855755=Igniv__Ana_Amaro_Alvez @ 8437-1960-282). The left sidebar shows the 'KMS' service and the 'Customer managed keys' path. The main content area is titled 'Configure key' and displays a five-step process: Step 1 (Configure key), Step 2 (Add labels), Step 3 (Define key administrative permissions), Step 4 (Define key usage permissions), and Step 5 (Review). The 'Configure key' step is active, showing two sections: 'Key type' and 'Key usage'. In the 'Key type' section, 'Symmetric' is selected with a radio button, and its description is 'A single key used for encrypting and decrypting data or generating and verifying HMAC codes'. The 'Asymmetric' option is also visible with its description. In the 'Key usage' section, 'Encrypt and decrypt' is selected with a radio button, and its description is 'Use the key only to encrypt and decrypt data.'. The 'Generate and verify MAC' option is also visible with its description.

aws Services Search [Alt+S] Oregon voclabs/user2855755=Igniv__Ana_Amaro_Alvez @ 8437-1960-282

VPC

KMS > Customer managed keys > Create key

Step 1
Configure key

Step 2
Add labels

Step 3
Define key administrative permissions

Step 4
Define key usage permissions

Step 5
Review

Configure key

Key type [Help me choose](#)

☒ Symmetric
A single key used for encrypting and decrypting data or generating and verifying HMAC codes

☐ Asymmetric
A public and private key pair used for encrypting and decrypting data or signing and verifying messages

Key usage [Help me choose](#)

☒ Encrypt and decrypt
Use the key only to encrypt and decrypt data.

☐ Generate and verify MAC
Use the key only to generate and verify hash-based message authentication codes (HMAC).

En la página Add labels (Agregar etiquetas) configure lo siguiente:

- Alias: MyKMSKey
- Description (Descripción): Key used to encrypt and decrypt data files.

Después seleccionaremos Next

KMS > Customer managed keys > Create key

Step 1
[Configure key](#)

Step 2
Add labels

Step 3
[Define key administrative permissions](#)

Step 4
Define key usage permissions

Step 5
Review

Add labels

Alias
You can change the alias at any time. [Learn more](#)

Alias
MyKMSKey

Description - optional
You can change the description at any time.

Description
Key used to encrypt and decrypt data files

En la página Define key administrative permissions (Definir permisos administrativos clave), en la sección Key administrators (Administradores de claves), busque y seleccione la casilla para voclabs y luego seleccione Next (Siguiente).

Step 2

[Add labels](#)

Step 3

Define key administrative permissions

Step 4

Define key usage permissions

Step 5

Review

Key administrators (1/13)

Choose the IAM users and roles who can administer this key through the KMS API. You may need to add additional permissions for the users or roles to administer this key from this console. [Learn more](#)

< 1 2 >

	Name	Path	Type
<input type="checkbox"/>	vocareum	/	Role
<input checked="" type="checkbox"/>	voclabs	/	Role
<input type="checkbox"/>	vocstartsoft	/	Role

Key deletion

☒ Allow key administrators to delete this key.

Cancel

Previous

Next

- En la página Define key usage permissions (Definir permisos de uso de claves), en la página This account (Esta cuenta), busque y seleccione la casilla para voclabs y luego seleccione Next (Siguiente).

Step 1
[Configure key](#)

Step 2
[Add labels](#)

Step 3
[Define key administrative permissions](#)

Step 4
Define key usage permissions

Step 5
Review

Define key usage permissions

Key users (1/13)
Select the IAM users and roles that can use the KMS key in cryptographic operations. [Learn more](#)

< 1 2 >

<input type="checkbox"/>	Name	Path	Type
<input type="checkbox"/>	vocareum	/	Role
<input checked="" type="checkbox"/>	voclabs	/	Role
<input type="checkbox"/>	vocstartsoft	/	Role

Other AWS accounts

Specify the AWS accounts that can use this key. Administrators of the accounts you specify are responsible for managing the permissions that allow their IAM users and roles to use this key. [Learn more](#)

Add another AWS account

- Revise la configuración y luego seleccione Finish (Finalizar).

KMS > [Customer managed keys](#) > Create key

Step 1
[Configure key](#)

Step 2
[Add labels](#)

Step 3
[Define key administrative permissions](#)

Step 4
[Define key usage permissions](#)

Step 5
Review

Review

Key configuration

Key type Symmetric	Key spec SYMMETRIC_DEFAULT	Key usage Encrypt and decrypt
Origin AWS KMS	Regionality Single-Region key	

i You cannot change the key configuration after the key is created.

Alias and description

Alias MyKMSKey	Description Key used to encrypt and decrypt data files
-------------------	---

- Revise la configuración y luego seleccione Finish (Finalizar).

Tags

Key	Value
No data No tags to display	

Key policy

To change this policy, return to previous steps or edit the text here.

```
1 {
2   "Id": "key-consolepolicy-3",
3   "Version": "2012-10-17",
4   "Statement": [
5     {
6       "Sid": "Enable IAM User Permissions",
7       "Effect": "Allow",
8       "Principal": {
9         "AWS": "arn:aws:iam::843719602821:root"
10      },
11       "Action": "kms:*",
12       "Resource": "*"
13     },
14   ]
15 }
```


Elija el enlace de MyKMSKey , que acaba de crear, y copie el valor de ARN (nombre de recurso de Amazon) en un editor de texto.

Utilizará este ARN copiado más adelante en el laboratorio.

[KMS](#) > [Customer managed keys](#) > Key ID: 664cbfc3-82b4-409e-a6fe-3f305630b02f

664cbfc3-82b4-409e-a6fe-3f305630b02f


Key actions ▼Edit

General configuration

Alias

MyKMSKey

ARN

 arn:aws:kms:us-west-2:843719602821:key/664cbfc3-82b4-409e-a6fe-3f305630b02f

Status

Enabled

Description

Key used to encrypt and decrypt data files

Creation date

Nov 10, 2023 19:04 GMT-3

Regionality

Single Region

Key policy

Cryptographic configuration

Tags

Key rotation

Aliases

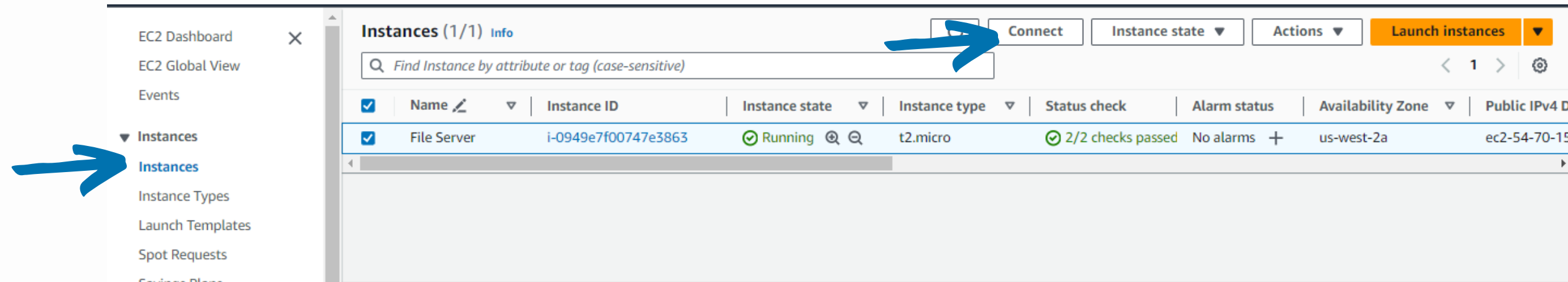
Key policy

Switch to policy view

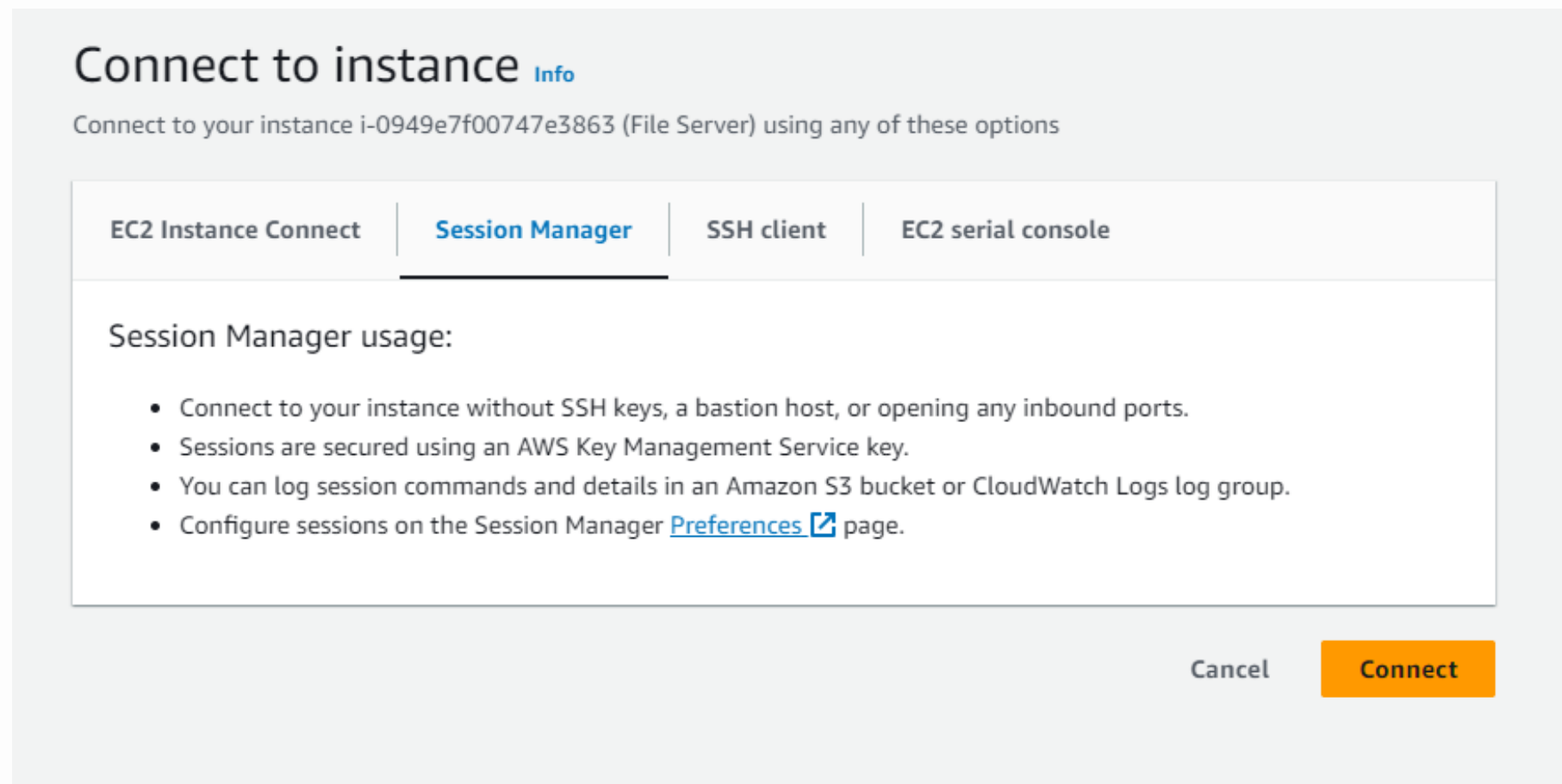


Tarea 2: Configurar la instancia del servidor de archivos

- En la consola, ingrese EC2 en la barra de búsqueda y luego seleccione EC2.
- En la lista Instances (Instancias), seleccione la casilla a su lado para la instancia de File Server (Servidor de archivos) y luego seleccione Connect (Conectar).



- En la pestaña Session Manager, elija Connect (Conectar).



- Para cambiar el directorio principal y crear el archivo de credenciales de AWS, ejecute los siguientes comandos:

```
cd ~  
aws configure
```

1. Cuando se le solicite, configure los siguientes ajustes:
 - AWS Access Key ID (ID de clave de acceso de AWS): Ingrese 1 y luego presione Intro.
 - AWS Secret Access Key ID (ID de clave de acceso secreto de AWS): Ingrese 1 y luego presione Intro.
 - Default region name (Nombre de región predeterminada): Copie y pegue la región proporcionada en la página AWS Details (Detalles de AWS) de Vocareum.
 - Default output format (Formato predeterminado de salida): Presione Intro.
2. Se creó archivo de configuración de AWS y lo actualizará en un paso posterior. Las entradas anteriores de 1 son marcadores de posición temporales.

Session ID:
user2855755=Igniv__Ana_Amaro_Alvez-
0656705288a32d459

Instance ID: i-0949e7f00747e3863

Terminate

```
sh-4.2$ cd ~  
sh-4.2$ aws configure  
AWS Access Key ID [None]: 1  
AWS Secret Access Key [None]: 1  
Default region name [None]: us-west-2  
Default output format [None]:  
sh-4.2$
```

26o9M0gRok3L2+NAePcEt+49X4PTw2IkyaRoGnB8eVJW5lsK7MRikbGAut6
yjpMSO7rAar/jpO2pBSTIY6ORLGGwORMyZDla2/Xsj1C71K5vwG9zk0oUOL
lMThzhqtQqC2wrsSA

```
vi ~/.aws/credentials
```

En el archivo `~/.aws/credentials`,
pulse dd múltiples veces para
borrar los contenidos del archivo

Pegue el bloque de código que copió de Vocareum

```
[default]
aws_access_key_id=ASIA4I4M3OKCRDXNSJPE
aws_secret_access_key=ZCYUjsOMoJr5mmA8BdtI3jwfcoNvApWD1e+m0ekk
aws_session_token=FwoGZXIvYXZEM////////wEaDOIDfXwTUcDGoCUHOCLIAy/ygFbjZcegkFUK9Bdd6uIu26o9M0gRok3L2+NAePcEt+49X4PTw2IkyaRoGnB8eVJW5lsK7MRikbGAut6
eR94WtxlgumSfx75VlrngfnlAPbbBgIT/i1tS/35oOucdPJC4Z1EhBWAMhTyG/5mpGqqIBro9qd8eeKBkNRODjtbE8yjPMS07rAar/jpO2pBSTIY6ORLGGwORMyZDla2/Xsj1C71K5vwG9zk0oUOL
ah/AmNeZzmuyzz8oOE0SHygvvpA5o789OzY7NKKrHuqoGMi03oJQKctQKi9KyUVCVvpSngqTVYuBH3Hnd+sH5yfXbK3lMThzhqtQqC2wrsSA
~
```

Para ver el contenido actualizado del archivo, ejecute el comando `cat ~/.aws/credentials`

```
sh-4.2$ cat ~/.aws/credentials
[default]
aws_access_key_id=ASIA4I4M3OKCRDXNSJPE
aws_secret_access_key=ZCYUjsOMoJr5mmA8BdtI3jwfcoNvApWD1e+m0ekk
aws_session_token=FwoGZXIvYXdzEM////////wEaDOIDfXwTUcDGoCUHOCLIAy/ygFbjZcegkFUK9Bdd6uIu26o9M0gRok3L2+NaePcEt+49X4PTw2IkyaRoGnB8eVJW5lsK7MRikbGAut6
er94WtxlgumSfx75VlrngfnlAPbbBgIT/i1ts/35oOucdPJC4ZlEhBWAMHTyG/5mpGqqIBro9qd8eeKBkNR0DjtbE8yjpMSO7rAar/jpO2pBSTIY6ORLGGwORMyZDla2/Xsj1C71K5vwG9zk0oUOL
ah/AmNeZzmuyzz8oOE0SHygvpa5o7890zY7NKKrHuqoGMi03oJQKctQKi9KyUVCVVpSngqTVYyBH3Hnd+sh5yfxbK3lMThzhqtQqC2wrsSA=
sh-4.2$
```

Para instalar la CLI de AWS Encryption y establecer su ruta, ejecute los siguientes comandos:


```
pip3 install aws-encryption-sdk-cli
export PATH=$PATH:/home/ssm-user/.local/bin
```

```
sh-4.2$ pip3 install aws-encryption-sdk-cli
Defaulting to user installation because normal site-packages is not writeable
Collecting aws-encryption-sdk-cli
  Downloading aws_encryption_sdk_cli-4.1.0-py2.py3-none-any.whl (44 kB)
    |████████████████████████████████████████| 44 kB 3.0 MB/s
Collecting base64io>=1.0.1
  Downloading base64io-1.0.3-py2.py3-none-any.whl (17 kB)
Collecting attrs>=17.1.0
  Downloading attrs-23.1.0-py3-none-any.whl (61 kB)
    |████████████████████████████████████████| 61 kB 11.8 MB/s
Collecting aws-encryption-sdk~=3.1
  Downloading aws_encryption_sdk-3.1.1-py2.py3-none-any.whl (99 kB)
    |████████████████████████████████████████| 99 kB 13.7 MB/s
Requirement already satisfied: setuptools in /usr/lib/python3.7/site-packages (from aws-encryption-sdk-cli) (49.1.3)
Collecting importlib-metadata; python_version < "3.8"
  Downloading importlib_metadata-6.7.0-py3-none-any.whl (22 kB)
Collecting wrapt>=1.10.11
  Downloading wrapt-1.16.0-cp37-cp37m-manylinux_2_5_x86_64.manylinux1_x86_64.manylinux2014_x86_64.whl (77 kB)
    |████████████████████████████████████████| 77 kB 11.0 MB/s
Collecting cryptography>=2.5.0
  Downloading cryptography-41.0.5-cp37-abi3-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (4.4 MB)
    |████████████████████████████████████████| 4.4 MB 39.0 MB/s
Collecting boto3>=1.10.0
  Downloading boto3-1.28.84-py3-none-any.whl (135 kB)
    |████████████████████████████████████████| 135 kB 46.9 MB/s
Collecting zipp>=0.5
  Downloading zipp-3.15.0-py3-none-any.whl (6.8 kB)
Collecting typing-extensions>=3.6.4; python_version < "3.8"
  Downloading typing_extensions-4.7.1-py3-none-any.whl (33 kB)
Collecting cffi>=1.12
  Downloading cffi-1.15.1-cp37-cp37m-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (427 kB)
    |████████████████████████████████████████| 427 kB 44.0 MB/s
Collecting botocore<1.32.0,>=1.31.84
```

```
Collecting botocore<1.32.0,>=1.31.84
  Downloading botocore-1.31.84-py3-none-any.whl (11.3 MB)
    |████████████████████████████████████████| 11.3 MB 33.5 MB/s
Collecting jmespath<2.0.0,>=0.7.1
  Downloading jmespath-1.0.1-py3-none-any.whl (20 kB)
Collecting s3transfer<0.8.0,>=0.7.0
  Downloading s3transfer-0.7.0-py3-none-any.whl (79 kB)
    |████████████████████████████████████████| 79 kB 14.5 MB/s
Collecting pycparser
  Downloading pycparser-2.21-py2.py3-none-any.whl (118 kB)
    |████████████████████████████████████████| 118 kB 44.8 MB/s
Collecting urllib3<1.27,>=1.25.4; python_version < "3.10"
  Downloading urllib3-1.26.18-py2.py3-none-any.whl (143 kB)
    |████████████████████████████████████████| 143 kB 46.0 MB/s
Collecting python-dateutil<3.0.0,>=2.1
  Downloading python_dateutil-2.8.2-py2.py3-none-any.whl (247 kB)
    |████████████████████████████████████████| 247 kB 46.6 MB/s
Collecting six>=1.5
  Downloading six-1.16.0-py2.py3-none-any.whl (11 kB)
Installing collected packages: base64io, zipp, typing-extensions, importlib-metadata, attrs, wrapt, pycparser, cffi, cryptography, urllib3, jmespath, six, python-dateutil, botocore, s3transfer, boto3, aws-encryption-sdk, aws-encryption-sdk-cli
  WARNING: The script aws-encryption-cli is installed in '/home/ssm-user/.local/bin' which is not on PATH.
  Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
Successfully installed attrs-23.1.0 aws-encryption-sdk-3.1.1 aws-encryption-sdk-cli-4.1.0 base64io-1.0.3 boto3-1.28.84 botocore-1.31.84 cffi-1.15.1 cryptography-41.0.5 importlib-metadata-6.7.0 jmespath-1.0.1 pycparser-2.21 python-dateutil-2.8.2 s3transfer-0.7.0 six-1.16.0 typing-extensions-4.7.1 urllib3-1.26.18 wrapt-1.16.0 zipp-3.15.0
sh-4.2$ export PATH=$PATH:/home/ssm-user/.local/bin
```


Resumen de la tarea 2

En esta tarea, configuró el archivo de credenciales de AWS, que proporciona la capacidad de usar la clave de AWS KMS que creó anteriormente. Luego instaló la CLI de AWS Encryption, para poder ejecutar comandos de cifrado.



Tarea 3: Cifrar y descifrar datos

En esta tarea, creará un archivo de texto con información confidencial ficticia. Luego, usará el cifrado para asegurar los contenidos del archivo. Luego, descifrará los datos y verá los contenidos del archivo.

Para crear el archivo de texto, ejecute los siguientes comandos:

```
touch secret1.txt secret2.txt secret3.txt  
echo 'TOP SECRET 1!!!' > secret1.txt
```

```
sh-4.2$ export PATH=$PATH:/home/ssm-user/.local/bin  
sh: export: `secret1.txt': not a valid identifier  
sh: export: `secret2.txt': not a valid identifier  
sh: export: `secret3.txt': not a valid identifier  
sh-4.2$ echo 'TOP SECRET 1!!!' > secret1.txt
```

Para ver los contenidos del archivo secret1.txt, ejecute el siguiente comando: `cat secret1.txt`

```
sh-4.2$ cat secret1.txt  
TOP SECRET 1!!!  
sh-4.2$
```

Para crear un directorio en el que crear el archivo cifrado, ejecute el siguiente comando:

```
mkdir output
```

Copie y pegue el siguiente comando en un editor de texto: `keyArn=(KMS ARN)`

```
sh-4.2$ mkdir output
sh-4.2$ keyArn=(2:843719602821:key/664cbfc3-82b4-409e-a6fe-3f305630b02f)
```


Para cifrar el archivo secret1.txt, ejecute el siguiente comando:

```
aws-encryption-cli --encriptar \  
    --entrada secret1.txt \  
    --wrapping-keys clave = $keyArn \  
    --salida-metadatos ~/metadatos \  
    --propósito del contexto de cifrado = prueba \  
    --política de compromiso requerir-cifrar-requerir-descifrar \  
    --salida ~/salida/.
```

La siguiente información describe lo que hace este comando:

- La primera línea cifra los contenidos del archivo. El comando usa el parámetro `--encrypt` para especificar la operación y el parámetro `--input` para indicar el archivo a cifrar.
- El parámetro `--wrapping-keys`, y su atributo requerido `key`, le indican al comando que use la clave de AWS KMS que está representada por el ARN de clave.
- El parámetro `--metadata-output` se usa para especificar un archivo de texto para los metadatos acerca de la operación de cifrado.
- Como práctica recomendada, el comando usa el parámetro `--encryption-context` para especificar un contexto de parámetro.
- El parámetro `--commitment-policy` se usa para especificar que la característica de seguridad de la confirmación de claves se debe usar para cifrar y descifrar.
- El valor del parámetro `--output`, `~/output/`, indica al comando que escriba el archivo de destino en el directorio de destino.

```
for SECRET in $(cat /dev/urandom | tr -dc 'a-z0-9' | fold -n 32 | xargs printf '%s\n'); do  
sh-4.2$ mkdir output  
sh-4.2$ keyArn=(KMS ARN)  
sh-4.2$ keyArn=arn:aws:kms:us-west-2:195227589964:key/2f03de7b-fc30-43f4-91ca-d596ad287718  
sh-4.2$ aws-encryption-cli --encrypt \  
> --input secret1.txt \  
> --wrapping-keys key=$keyArn \  
> --metadata-output ~/metadata \  
> --encryption-context purpose=test \  
> --commitment-policy require-encrypt-require-decrypt \  
> --output ~/output/.  
sh-4.2$ echo $?  
0
```

Para determinar si el comando se realizó correctamente, ejecute el siguiente comando: `echo $?`

Si el comando se realizó correctamente, el valor de \$? es 0. Si el comando falló, el valor no es cero.

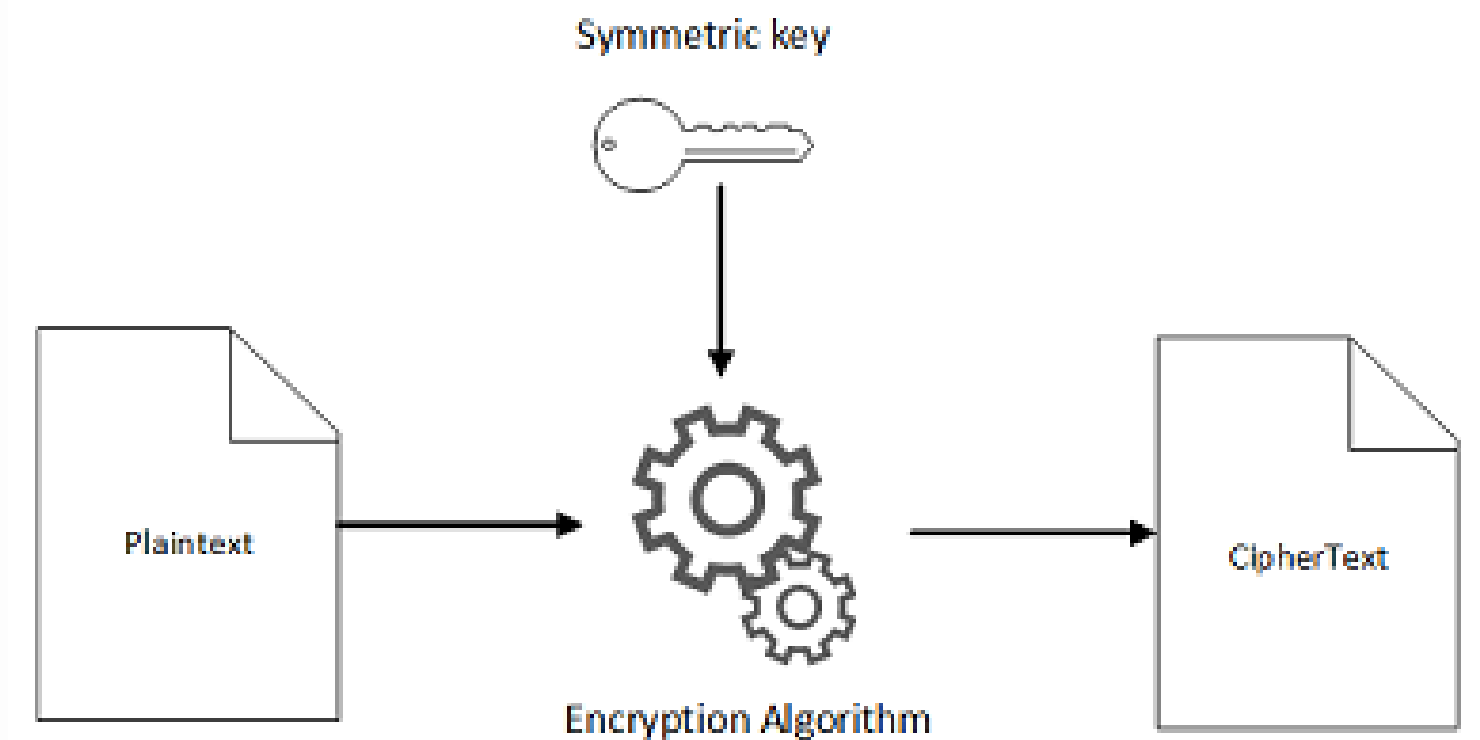
Para ver la ubicación del archivo recién cifrado, ejecute el siguiente comando: `ls output`

Para ver los contenidos del archivo recién cifrado, ejecute el siguiente comando:

```
cd output
cat secret1.txt.encrypted
```

```
sh-4.2$ echo $?
0
sh-4.2$ ls output
secret1.txt.encrypted
sh-4.2$ cd output
sh-4.2$ cat secret1.txt.encrypted
xS[
  YS?d:~?ni?h^es?naws-crypto-public-keyDaruTYNmt0kVlXWAQMGCzM4N986vAp8DCZHHkF8tZZ9dR8atRm6Sn1BM7M5hHbhQLQ==purp
osetestaws-kmsKarn:aws:kms:us-west-2:777936295668:key/91e2a04a-ee47-42fb-b99d-6c3fad177d21?xaom>sE?B,V?_i?
?mT?L?JE<bP-]Z?tf?????cI?5
?Pd?j?l?Al~?g0e0(?B?D?Bm?AN?"
r????v?hWB?#??e?<J??sh-4.2$
```

El siguiente diagrama muestra cómo funciona el cifrado con las claves y algoritmos simétricos.. Una clave y un algoritmo simétricos y usan para convertir un mensaje de texto simple en texto cifrado.



- A continuación, descifrará el archivo **secret1.txt.encrypted**
- Para descifrar el archivo, ejecute los siguientes comandos:

```
aws-encryption-cli --decrypt \  
    --input secret1.txt.encrypted \  
    --wrapping-keys wrapping-key=$keyArn \  
    --commitment-policy require-encrypt-require-decrypt \  
    --encryption-context purpose=test \  
    --metadata-output ~/metadata \  
    --max-encrypted-data-keys 1 \  
    --buffer \  
    --output .
```

La siguiente información describe lo que hace este comando:

- La primera línea cifra los contenidos del archivo. El comando usa el parámetro `--encrypt` para especificar la operación y el parámetro `--input` para indicar el archivo a cifrar.
- El parámetro `--wrapping-keys`, y su atributo requerido `key`, le indican al comando que use la clave de AWS KMS que está representada por el ARN de clave.
- El parámetro `--metadata-output` se usa para especificar un archivo de texto para los metadatos acerca de la operación de cifrado.
- Como práctica recomendada, el comando usa el parámetro `--encryption-context` para especificar un contexto de parámetro.
- El parámetro `--commitment-policy` se usa para especificar que la característica de seguridad de la confirmación de claves se debe usar para cifrar y descifrar.
- El valor del parámetro `--output`, `~/output/.`, indica al comando que escriba el archivo de destino en el directorio de destino.

- Para determinar si el comando se realizó correctamente, ejecute el siguiente comando: `echo $?`
Si el comando se realizó correctamente, el valor de \$? es 0. Si el comando falló, el valor no es cero.

- Para ver la ubicación del archivo recién cifrado, ejecute el siguiente comando: `ls output`

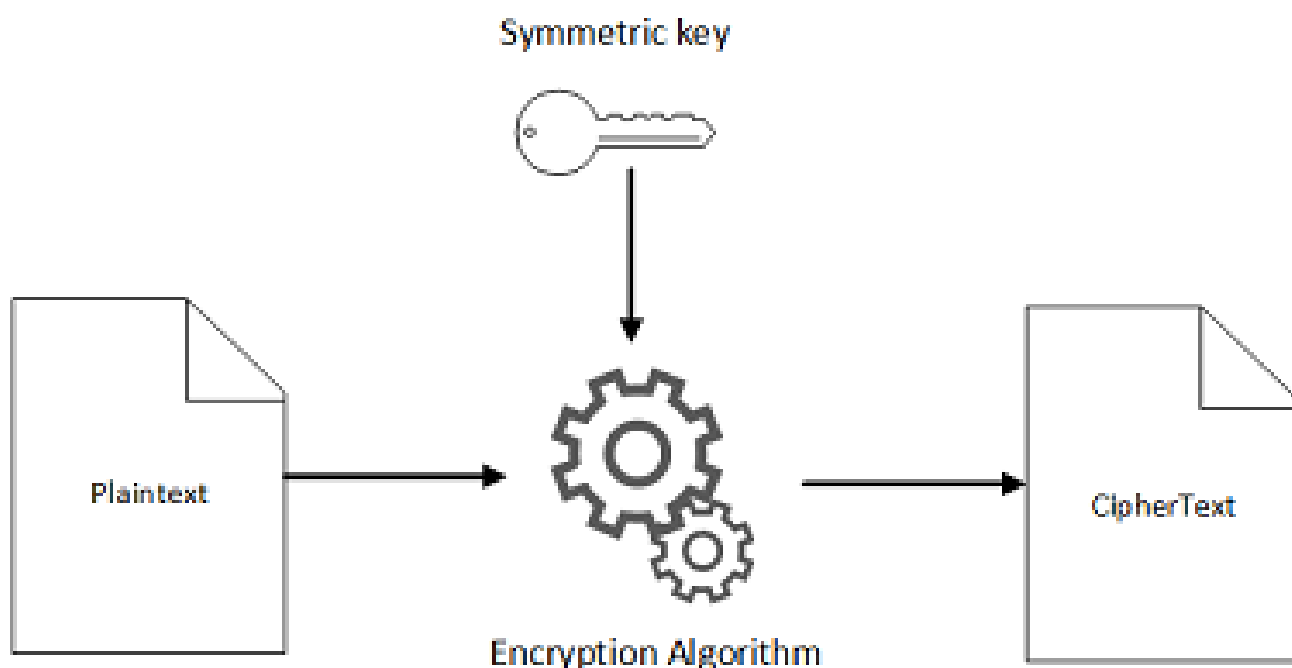
- El resultado debería verse de la siguiente manera: `secret1.txt.encrypted`

- Para ver los contenidos del archivo recién cifrado, ejecute el siguiente comando:
`cd output`
`cat secret1.txt.encrypted`

El proceso de cifrado y descifrado toma los datos en texto simple, que se puede leer y comprender, y manipula su forma para crear texto cifrado, que es lo que está viendo ahora.

Cuando los datos se transforman en texto cifrado, el texto simple no estará disponible hasta que se descifre.

El siguiente diagrama muestra cómo funciona el cifrado con las claves y algoritmos simétricos.. Una clave y un algoritmo simétricos y usan para convertir un mensaje de texto simple en texto cifrado.



```
#? ItN? sh-4.2$ aws-encryption-cli --decrypt \
> --input secret1.txt.encrypted \
> --wrapping-keys key=$keyArn \
> --commitment-policy require-encrypt-require-decrypt \
> --encryption-context purpose=test \
> --metadata-output ~/metadata \
> --max-encrypted-data-keys 1 \
> --buffer \
> --output .
sh-4.2$ ls
secret1.txt.encrypted secret1.txt.encrypted.decrypted
sh-4.2$ cat secret1.txt.encrypted.decrypted
TOP SECRET 1!!!
sh-4.2$
```

```
sh-4.2$ export PATH=$PATH:/home/ssm-user/.local/bin
sh-4.2$ touch secret1.txt secret2.txt secret3.txt
sh-4.2$ echo 'TOP SECRET 1!!!!' > secret1.txt
sh-4.2$ cat secret1.txt
TOP SECRET 1!!!!
sh-4.2$ mkdir output
sh-4.2$ keyArn=(KMS ARN)
sh-4.2$ keyArn=arn:aws:kms:us-west-2:195227589964:key/2f03de7b-fc30-43f4-
sh-4.2$ aws-encryption-cli --encrypt \
> --input secret1.txt \
> --wrapping-keys key=$keyArn \
> --metadata-output ~/metadata \
> --encryption-context purpose=test \
> --commitment-policy require-encrypt-require-decrypt \
> --output ~/output/.
sh-4.2$ echo $?
0
sh-4.2$ ls output
secret1.txt.encrypted
sh-4.2$ cd output
sh-4.2$ cat secret1.txt.encrypted
xkE0aws-crypto-public-keyDAjHNE0klM/Af+oFbUMtDadOn
0o0m0ha~He.0287718xKp00000I0AOj0fyg00
}000g^!0(000,$0w0H%
000E0SL 0E0tt,8>0000,YA0loL0{ 0000
$BQ0J00
0000000000000000/10m000e8000n0 y0svg0000
#0ItN00sh-4.2$ aws-encryption-cli --decrypt \
> --input secret1.txt.encrypted \
> --wrapping-keys key=$keyArn \
> --commitment-policy require-encrypt-require-decrypt \
> --encryption-context purpose=test \
> --metadata-output ~/metadata \
> --max-encrypted-data-keys 1 \
> --buffer \
> --output .
sh-4.2$ ls
secret1.txt.encrypted secret1.txt.encrypted.decrypted
sh-4.2$ cat secret1.txt.encrypted.decrypted
TOP SECRET 1!!!!
sh-4.2$
```