



## 278-[SF]-Lab - Protección de datos mediante encriptación

### Datos Generales:

**Nombre:** Tomás Alfredo Villaseca Constantinescu

**País:** Chile

**Fecha:** 23/09/2023

**Contacto:** [tomas.villaseca.c@gmail.com](mailto:tomas.villaseca.c@gmail.com)

Después de completar este laboratorio, podrá realizar lo siguiente:

- Crear una clave de cifrado de AWS KMS
- Instalar la CLI de AWS Encryption
- Cifrar datos de texto simple
- Descifrar texto cifrado

Entorno del laboratorio → Una instancia EC2 pre-configurada “FileServer”

- Rol de IAM adjuntado a FileServer
- Rol de IAM → Permite conectarse a la EC2 con Sessions Manager.

# Tarea 1: Crear una clave de AWS KMS

**AWS Key Management Service (AWS KMS)** = Servicio que permite realizar operaciones de cifrado mediante el uso de claves criptográficas.



AWS KMS

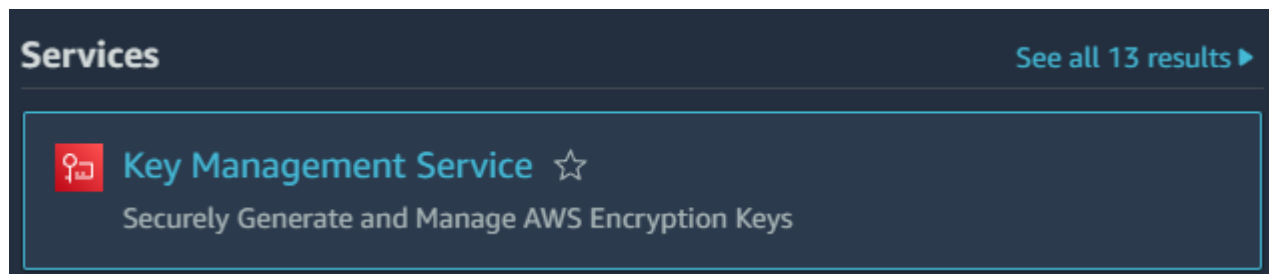
- Claves criptográficas = cadena aleatoria de dígitos utilizada para cifrar (clave de cifrado) y descifrar (clave de descifrado) datos.
- Crear, gestionar y utilizar claves criptográficas.
- Puede elegir los niveles específicos de control de acceso que necesite para sus claves.

Cifrado = Proceso de conversión de datos de un formato legible a un formato codificado.

Descifrado = Proceso de conversión de datos de un formato codificado a un formato legible.

Con AWS KMS, puede crear y administrar claves criptográficas y controlar su uso a lo largo de una amplia variedad de servicios de AWS y en sus aplicaciones.

**Paso 1:** AWS Management Console → Search → AWS KMS



**Paso 2:** Seleccionar “Create Key” → Configure Key

- Key type = Symmetric
- Key Usage = Encrypt and decrypt

You can create a key by clicking the button below.

Create a key

# 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).

### Paso 3: Configure Key → Add Labels

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

## 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

#### Paso 4: Configure Key → Define key administrative permissions

- Key administrators → Seleccionar casilla “voclabs”

### Define key administrative permissions

**Key administrators (1/14)**  
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 >

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

#### Paso 5: Configure Key → Define key usage permissions

- Key Users → Seleccionar casilla “voclabs”

### Define key usage permissions

**Key users (1/14)**  
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"/>	robomaker_students	/	Role
<input type="checkbox"/>	vocareum	/	Role
<input checked="" type="checkbox"/>	voclabs	/	Role
<input type="checkbox"/>	vocstartsoft	/	Role

**Paso 6:** Configure Key → Review → Verificar que está correcta la configuración → Finish

Review

Key configuration

Key type

Symmetric

Key spec

SYMMETRIC\_DEFAULT

Key usage

Encrypt and decrypt

Origin

AWS KMS

Regionality

Single-Region key

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.

Customer managed keys (1/1)

Filter keys by properties or tags

☒

Aliases

☒

MyKMSKey

25853553-31c4-4c15-8a33-a...

Enabled

Symmetric

**Paso 7:** Seleccionar MYKMS Key → Seleccionar Key ID → Copiar ARN para más tarde.

- ARN = `arn:aws:kms:us-west-2:229153271158:key/25853553-31c4-4c15-8a33-a35db6f50752`
- ARN (Amazon Resource Name) → Identifican de forma exclusiva recursos de AWS.

General configuration

Alias

MyKMSKey

ARN

arn:aws:kms:us-west-2:229153271158:key/25853553-31c4-4c15-8a33-a35db6f50752

Status

Enabled

Description

Key used to encrypt and decrypt data files.

Creation date

Sep 23, 2023 16:21 GMT-3

Regionality

Single Region

En esta tarea se creó una clave simétrica AWS KMS y se le entregó la propiedad de esa clave al rol voclabs IAM que fue pre-creado para este laboratorio.

5

## Tarea 2: Configurar la instancia de servidor de archivo

Para utilizar su Key de AWS KMS, configurará las credenciales de AWS en la instancia EC2 FileServer.



- Instalará la AWS Encryption CLI, que puede utilizar para ejecutar comandos de cifrado y descifrado.

**Paso 1:** AWS Management Console → EC2 → Instances → FileServer → Connect


- Connect to instance → Session Manager → Connect

**Instances (1/1)** [Info](#)

Find instance by attribute or tag (case-sensitive)

<input checked="" type="checkbox"/>	Name ▾	Instance ID	Instance state ▾
<input checked="" type="checkbox"/>	File Server	i-0d1cf6e3e94d9189f	<span>Running</span>  

[Connect](#) [Instance state ▾](#) [Actions ▾](#) [Launch instances ▾](#)


< 1 > 

### Connect to instance [Info](#)

Connect to your instance i-0d1cf6e3e94d9189f (File Server) using any of these options

[EC2 Instance Connect](#) | [Session Manager](#) | [SSH client](#) | [EC2 serial console](#)

Session Manager usage:

- Connect to your instance without SSH keys, a bastion host, or opening any inbound ports.
- Sessions are secured using an AWS Key Management Service key.
- You can log session commands and details in an Amazon S3 bucket or CloudWatch Logs log group.
- Configure sessions on the Session Manager [Preferences](#)  page.

[Cancel](#) [Connect](#)

## Paso 2: Cambiar al directorio Home:

Session ID: user2741130=Tom\_\_sVillaseca-00f20417e58660cd6

Instance ID: i-0d1cf6e3e94d9189f

```
sh-4.2$ cd ~
sh-4.2$ pwd
/home/ssm-user
sh-4.2$
```

## Paso 3: Ingresar comando “aws configure” para crear un archivo de credenciales AWS:

- **AWS Access Key ID** = Ingresar “1”
- **AWS Secret Access Key**: Ingresar “1”
- **Default region name**: Ingresar “us-west-2” (Región entregada por el Lab)
- **Default output format**: presionar “Enter”

Session ID: user2741130=Tom\_\_sVillaseca-00f20417e58660cd6

Instance ID: i-0d1cf6e3e94d9189f

```
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$
```

## Paso 4: Para abrir el archivo de credenciales AWS → Comando “vi ~/.aws/credentials”

- Copiar código de AWS CLI (entregado por el LAB) en el archivo de credenciales AWS.
- :wq para guardar y salir.

```
sh-4.2$ vi ~/.aws/credentials
```

Session ID: user2741130=Tom\_\_sVillaseca-00f20417e58660cd6

Instance ID: i-0d1cf6e3e94d9189f

```
[default]
aws_access_key_id=ASIATKWUZNV3O5PIXXZK
aws_secret_access_key=hbScc7iTZI2D0pnxkZitHpyIe//+h29G9jluwHjq
aws_session_token=FwoGZXIvYXdzEE0aDOsE82wALsdL5FmH9SLAAW1hWQOMTU1sgQds8XRSO8.
GKjVJvJsI1xVCS9dIikpjdK6a7ccimVc5DOXFHdsyC4x6C+YEEzmyUWNggjpoiyeO7Q7jcwec6Bo
~
~
```



**Paso 5:** Para ver el contenido actualizado del archivo de credenciales AWS

- Ingresar comando “cat ~/.aws/credentials”

Session ID: user2741130=Tom\_\_sVillaseca-00f20417e58660cd6

Instance ID: i-0d1cf6e3e94d9189f

```
sh-4.2$ cat ~/.aws/credentials
[default]
aws_access_key_id=ASIATKWUZNV3O5PIXXZK
aws_secret_access_key=hb8cc7iTZi2DOpnxkZitHpyIe//+h29G9jluwHjq
aws_session_token=FwoGZXIvYXdzEE0aDOsE82wALsdL5FmH9SLAAWlhWQOMTU1sgQds8XRSO8ic
GKjVJvJsI1xVCS9dIikpjdK6a7ccimVc5DOXFHDsyC4x6C+YEEzmyUWNggjpoiyeO7Q7jcwec6BoYs
sh-4.2$
```

**Paso 6:** Para instalar AWS Encryption CLI y exportar su ruta:

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

Session ID: user2741130=Tom\_\_sVillaseca-00f20417e58660cd6

Instance ID: i-0d1cf6e3e94d9189f

```
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 3.1 MB/s
```

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



## Tarea 3: Cifrar y descifrar datos

En esta tarea, creará un archivo de texto con datos confidenciales simulados.

- Se utilizará el cifrado para proteger el contenido del archivo.
- Descifrará los datos y visualizará el contenido del archivo.

**Paso 1:** Crear archivos de texto e ingresar uno de los archivos con datos confidenciales simulados.

Session ID: user2741130=Tom\_\_sVillaseca-0686345a7732c5988

```
sh-4.2$ touch secret1.txt secret2.txt secret3.txt
sh-4.2$ ls
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$
```

**Paso 2:** Crear un directorio “output” en donde se creará el output del cifrado de archivos.

Session ID: user2741130=Tom\_\_sVillaseca-0686345a7732c5988

```
sh-4.2$ mkdir output
sh-4.2$ ls
output  secret1.txt  secret2.txt  secret3.txt
sh-4.2$
```

**Paso 3:** Ingresar el comando “keyARN=(KMS ARN)”

- Reemplazar (KMS ARN) en el comando por el ARN de la Key creada anteriormente.
- Guarda el ARN de una Key en la variable \$keyArn.

Session ID: user2741130=Tom\_\_sVillaseca-0686345a7732c5988

Instance ID: i-0d1cf6e3e94d9189f

```
sh-4.2$ keyArn=arn:aws:kms:us-west-2:229153271158:key/25853553-31c4-4c15-8a33-a35db6f50752
sh-4.2$
```

**Paso 4:** Para cifrar el archivo secret1.txt ingresar el siguiente comando:

```
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/.
```

- **--encrypt** = cifra los contenidos del archivo
- **--input** = indicar el archivo a cifrar.
- **--wrapping-keys** = indican al comando que use la Key especificada representado por el \$keyArn
- **--metadata-output** = para especificar un archivo de texto para los metadatos acerca de la operación de cifrado.
- **--encryption-context** = para especificar un contexto de parámetro.
- **--commitment-policy** = para especificar que la característica de seguridad de la confirmación de claves se debe usar para cifrar y descifrar.
- **--output** = indica el directorio de destino para el archivo cifrado.

Session ID: user2741130=Tom\_\_sVillaseca-0686345a7732c5988

Instance ID: i-0d1cf6e3e94d9189f

```
sh-4.2$ aws-encryption-cli --encrypt \  
>     --input secret1.txt \  
>     --wrapping-keys key=$keyArn \  
>     --metadata-output ~/metadata \  
>     --encryption-context purpose=test \  
>     --commitment-policy require-encrypt-require-decrypt \  
>     --output ~/output/.  
2023-09-23 19:55:37,384 - MainThread - aws_encryption_sdk_cli - WARNING - Overw  
sh-4.2$
```

**Paso 5:** Para verificar que el comando de cifrado fue realizado correctamente:

- Si valor de \$? es 0 → Comando fue ejecutado correctamente

Session ID: user2741130=Tom\_\_s\

```
sh-4.2$ echo $?  
0  
sh-4.2$
```

**Paso 6:** Verificar que el output del comando de cifrado fue creado en el directorio designado:

Session ID: user2741130=Tom\_\_s'

```
sh-4.2$ ls output
secret1.txt.encrypted
sh-4.2$
```

**Paso 7:** Revisar que el contenido de secret1.txt.encrypted efectivamente se encuentra cifrado.

Session ID: user2741130=Tom\_\_sVillaseca-0686345a7732c5988

Instance ID: i-0d1cf6e3e94d9189f

```
sh-4.2$ pwd
/home/ssm-user/output
sh-4.2$ cat secret1.txt.encrypted
x@??}d???[?EiJx![?+?#j???Jy??maws-crypto-public-keyDArW5Rp7lELtTgPLiAOxuOs+NbXPDLK
0o0m0h?? ?He.0 ?o??(??x?ohPi????c???A?.;e?w??^?~0| *?H??
??v58??[?;?g
M??E???!???5 vT 5??B?se?? ???? ????`K?L2N?nA???D{E?p?Q? ? *?EK_nyrn??
????y?mg/QKB??I?D?????w????LY? ????k?L??tT??!kg0e1???lT&???r????s?N?p ????.
sh-4.2$
```

**Paso 8:** Para descifrar el archivo secret1.txt.encrypted utilizar el siguiente comando:

```
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 .
```

- **--decrypt** = descifra los contenidos del archivo

**Paso 9:** Verificar que el output del comando de descifrado fue creado en el directorio designado:

Session ID: user2741130=Tom\_\_sVillaseca-0686345a7732c5988

Instance ID: i-0d1cf6e3e94d9189f

```
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$
```

**Paso 9:** Revisar que el contenido del archivo secret1.txt.encrypted.decrypted efectivamente se encuentra descifrado.

Session ID: user2741130=Tom\_\_sVillaseca-0686345a7732c5988

Instance ID: i-0d1cf6e3e94d9189f

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

Se aprendió a cifrar datos de texto plano en texto cifrado ejecutando el comando **--encrypt**.

Se aprendió a descifrar con éxito el texto cifrado para convertirlo en el texto plano original legible ejecutando el comando **--decrypt**.

Laboratorio Completado

