# S3: Laboratorio: Protección de datos mediante encriptación

## Información general sobre el laboratorio

Criptografía es la conversión de información comunicada a un código secreto que mantiene la información confidencial y privada. Las funciones incluyen la autenticación, la integridad de datos y el no repudio. La función central de la criptografía es el *cifrado*, que transforma los datos en una forma ilegible.

El cifrado asegura la privacidad al mantener la información oculta de las personas para las que no está destinada la información. *Descifrado*, lo opuesto del cifrado, transforma los datos cifrados en datos una vez más, no tendrán sentido hasta que se hayan descifrado correctamente.

En este laboratorio, se conectará a un servidor de archivos que está alojado en una instancia de Amazon Elastic Compute Cloud (Amazon EC2). Configurará la interfaz de línea de comando (CL) de AWS Encryption en la instancia Creará una clave de cifrado usando AWS Key Management Service (AWS KMS). La clave se usará para cifrar y descifrar datos. A continuación, creará múltiples archivos de texto que de forma predeterminada, no están cifrados. Luego usará la clave de AWS KMS para cifrar los archivos y verlos mientras se cifran. Finalizará el laboratorio al descifrar los mismos archivos y ver los contenidos.

### **OBJETIVOS**

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

## Tarea 1: Crear una clave de AWS KMS

En esta tarea, creará una clave AWS KMS que luego usará para cifrar y descifrar datos.

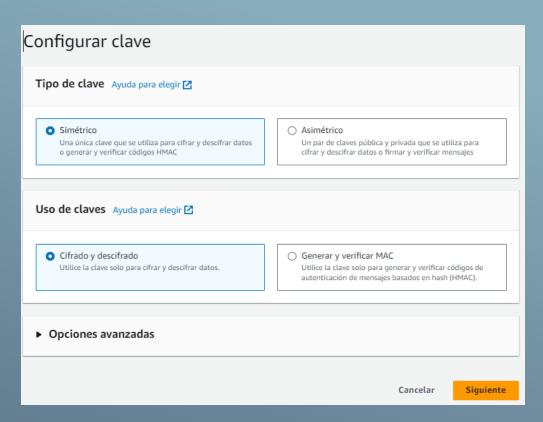
Con AWS KMS, puede crear y administrar claves criptográficas y controlar su uso a lo largo de una amplia variedad de servicios de AWS y en sus aplicaciones. AWS KMS es un servicio seguro y resiliente que usa módulos de seguridad de hardware (HSM) que están validados por el Estándar de procesamiento de información federal (FIPS) Publicación 140-2, o están en el proceso de ser validados, para proteger sus claves.

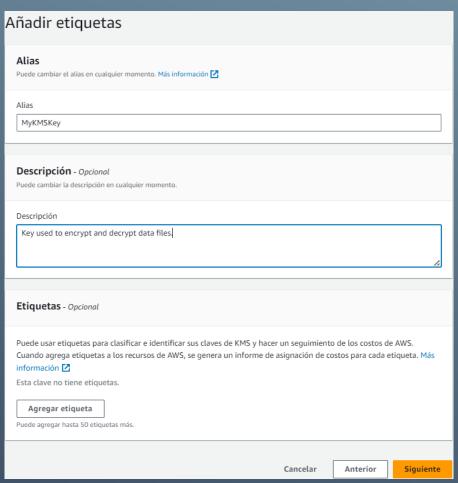
- 6. En la consola, ingrese **KMS** en la barra de búsqueda y luego seleccione **Key Management Service**.
- 7. Seleccione **Create a key** (Crear una clave).
- 8. En **Key type**, seleccione **Symmetric** (Simétrica) y luego seleccione **Siguiente**. El cifrado *Simétrico* usa la misma clave para cifrar y descifrar datos, lo que hace que sea fácil y eficiente de usar. El cifrado *Asimétrico* usa una clave pública para cifrar datos y una clave privada para descifrar información.
- 9. En la página Add labels (Agregar etiquetas) configure lo siguiente:
- Alias: MyKMSKey
- **Description** (Descripción): Key used to encrypt and decrypt data files.
- 10. Seleccione \*\*Next \*\*(Siguiente).
- 11. 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).
- 12. 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).
- 13. Revise la configuración y luego seleccione Finish (Finalizar).
- 14. Copie el enlace para **MyKMSKey**, que acaba de crear, y copie el valor **ARN** (Amazon Resource Name) a un editor de texto.

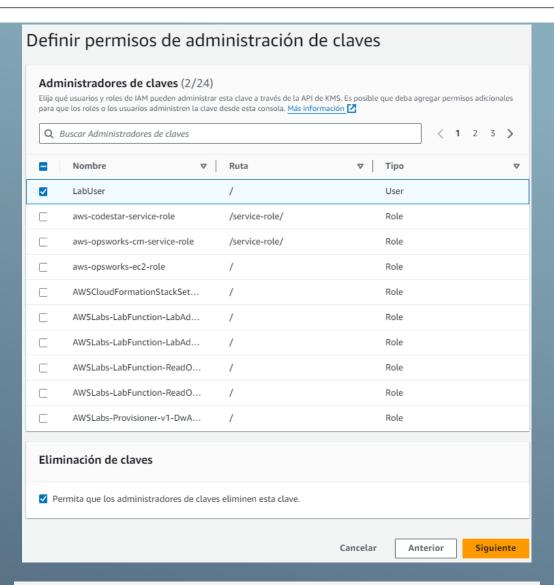
Utilizará este ARN copiado más adelante en el laboratorio.

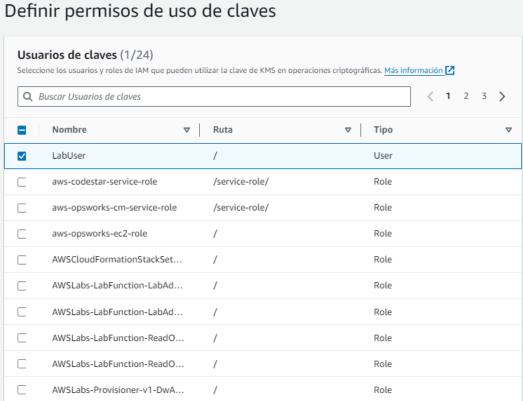
#### **RESUMEN DE LA TAREA 1**

En esta tarea, creó una clave AWS KMS simétrica y le otorgó la propiedad de esa clave al rol **voclabs** IAM que se creó previamente para este laboratorio.









# Revisar Configuración de la clave Tipo de clave Especificación de la clave Uso de claves Simétrico SYMMETRIC\_DEFAULT Cifrado y descifrado Origen Regionalidad AWS KMS Clave de una sola región (1) No puede cambiar la configuración de la clave una vez creada la clave. Alias y descripción Alias Descripción MyKMSKey Key used to encrypt and decrypt data files. Etiquetas Clave Valor No hay datos No hay etiquetas para mostrar

# Tarea 2: Configurar la instancia de servidor de archivo

Antes de que pueda cifrar y descifrar datos, debe configurar algunas cosas. Para usar su clave AWS KMS, configurará credenciales de AWS en la instancia de EC2 de **File Server** (Archivo de servidor). Después, instalará la CLI de AWS Encryption (aws-encryption-cli), que puede usar para cifrar y descifrar comandos.

- 15. En la consola, ingrese EC2 en la barra de búsqueda y luego seleccione EC2.
- 16. 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).
- 17. En la pestaña **Session Manager**, elija **Connect** (Conectar).
- 18. Para cambiar el directorio principal y crear el archivo de credenciales de AWS, ejecute los siguientes comandos:

cd ~

aws configure

- 19. 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** (Detailes de AWS) de Vocareum.
- **Sugerencia** Es posible que tenga que presionar Ctrl+Shift+V para pegar en Session Manager.
- **Default output format** (Formato predeterminado de salida): Presione Intro.

Se creó archivo de configuración de AWS y lo actualizará en un paso posterior. Las entradas anteriores de 1 son marcadores de posición temporales.

- 20. Navegue hasta la página de la consola de Vocareum y seleccione el botón **AWS Details** (Detailes de AWS).
- 21. Junto a AWS CLI, seleccione Show (Mostrar).
- 22. Copie y pegue el bloque de código, que comienza con [default] (predeterminado), en un editor de texto.
- 23. Vuelva a la pestaña del navegador en la que inició sesión en el Servidor de archivos.
- 24. Para abrir el archivo de credenciales de AWS, ejecute el siguiente comando:

vi ~/.aws/credentials

```
ID de instancia: i-Oc3cfc8f4472386c0

079b805c557b80b6c

sh-4.2$ cd ~
sh-4.2$ aws configure

AWS Access Key ID [None]: 1

Default region name [None]: us-west-2a

Default output format [None]: sh-4.2$
```

- 25. En el archivo ~/.aws/credentials, pulse dd múltiples veces para borrar los contenidos del archivo.
- 26. Pegue el bloque de código que copió de Vocareum. El archivo de credenciales de AWS debe ser similar a lo siguiente:

```
ID de sesión: 37ece71c-2d18-43aa-badc-7e1a03f5f73d-

04dd80d8f431069c6

Idefault

aws_access_key_id = AKIAXSSTRVON6CWZKNJX

aws_secret_access_key = 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
```

- 27. Para guardar y cerrar el archivo, presione Escape, escriba :wq y luego presione Intro.
- 28. Para ver los contenidos actualizados del archivo, ejecute el siguiente comando: cat ~/.aws/credentials

Ahora instalará la CLI de AWS Encryption y exportará su ruta. Al hacerlo, podrá ejecutar los comandos para cifrar y descifrar datos.

29. 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

```
ID de instancia: i-0eae67100f535d0d0
 ID de sesión: 37ece71c-2d18-43aa-badc-7e1a03f5f73d-
 04dd80d8f431069c6
sh-4.2$ cat ~/.aws/credentials
[default]
aws_access_key_id = AKIAXSSTRVQN6CWZKNJX
aws secret access key = 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
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 2.5 MB/s
Collecting base64io>=1.0.1
  Downloading base64io-1.0.3-py2.py3-none-any.whl (17 kB)
Collecting aws-encryption-sdk~
  Downloading aws_encryption_sdk-3.1.1-py2.py3-none-any.whl (99 kB)
                                           99 kB 7.4 MB/s
Requirement already satisfied:
Collecting attrs>=17.1.0
                                  setuptools in /usr/lib/python3.7/site-packages (from aws-encryption-sdk-cli) (49.1.3)
  Downloading attrs-23.2.0-py3-none-any.whl (60 kB)
                                            60 kB 12.5 MB/s
Collecting boto3>=1.10.0
  Downloading boto3-1.33.13-py3-none-any.whl (139 kB)
                                          | 139 kB 41.2 MB/s
Collecting wrapt>=1.10.11
  Downloading wrapt-1.16.0-cp37-cp37m-manylinux_2_5_x86_64.manylinux1_x86_64.manylinux_2_17_x86_64.manylinux2014_x86_64.whl (77 kB)
                                           77 kB 5.6 MB/s
Collecting cryptography>=2.5.0
  Downloading cryptography-42.0.2-cp37-abi3-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (4.6 MB)
                                         | 4.6 MB 38.6 MB/s
            importlib-metadata; python_version < "3.8"</pre>
Collecting
  Downloading importlib_metadata-6.7.0-py3-none-any.whl (22 kB)
Collecting jmespath<2.0.0,>=0.7.1
Downloading jmespath-1.0.1-py3-none-any.whl (20 kB) Collecting botocore<1.34.0,>=1.33.13
  Downloading botocore-1.33.13-py3-none-any.whl (11.8 MB)
                                          | 11.8 MB 19.1 MB/s
Collecting
            s3transfer<0.9.0
  Downloading s3transfer-0.8.2-py3-none-any.whl (82 kB)
Collectin
  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 urllib3<1.27,>=1.25.4; python_version < "3.10"
Downloading urllib3-1.26.18-py2.py3-none-any.whl (143 kB)
```

### **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.

- 30. Para crear el archivo de texto, ejecute los siguientes comandos:
  - touch secret1.txt secret2.txt secret3.txt echo 'TOP SECRET 1!!!' > secret1.txt
- 31. Para ver los contenidos del archivo secret1.txt, ejecute el siguiente comando: cat secret1.txt
- 32. Para crear un directorio en el que crear el archivo cifrado, ejecute el siguiente comando: mkdir output
- 33. Copie y pegue el siguiente comando en un editor de texto: keyArn=(KMS ARN)
- 34. En este editor de texto, reemplace (KMS ARN) con el AWS KMS ARN que copió en la tarea
- 35. Ejecute el comando actualizado en el terminal del Servidor de archivos.

Este comando guarda el ARN de una clave de AWS KMS en la variable \$keyArn. Cuando cifra usando una clave de AWS KMS, puede identificarla usando una ID de clave, el ARN de clave, el nombre de alias, o el ARN de alias.

36. Para cifrar el archivo **secret1.txt**, ejecute 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/.

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.

Cuando el comando encrypt se lleva a cabo correctamente, no arroja ningún resultado.

37. 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

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

ID de sesión: 37ece71c-2d18-43aa-badc-7e1a03f5f73d- ID de instancia: i-0eae67100f535d0d0 04dd80d8f431069c6

```
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$
```

El resultado debería verse de la siguiente manera:

ID de sesión: 37ece71c-2d18-43aa-badc-7e1a03f5f73d-04dd80d8f431069c6 ID de instancia: i-0eae67100f535d0d0

```
sh-4.2% ls
metadata output secret1.txt secret1.txt.encrypted.decrypted secret1.txt.encrypto secret2.txt secret3.txt
sh-4.2%
```

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

40. Presione Intro.

A continuación, descifrará el archivo secret1.txt.encrypted.

41. Para descifrar el archivo, ejecute los siguientes comandos:

```
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.
- 42. Para ver la ubicación del nuevo archivo, ejecute el siguiente comando:

Ls

- El archivo **secret1.txt.encrypted.decrypted** contiene los contenidos descifrados del archivo **secret1.txt.encrypted**.
  - 43. Para ver los contenidos del archivo descifrado, ejecute el siguiente comando: cat secret1.txt.encrypted.decrypted

Después del descifrado correcto, podrá ver los contenidos en *texto simple* originales de **secret1.txt**.

### **RESUMEN DE LA TAREA 3**

En esta tarea, aprendió cómo cifrar datos de texto simple en texto cifrado al ejecutar el comando —encrypt. Luego descifró correctamente el texto cifrado de vuelta a los datos de texto simple originales y legibles.