Guía del usuario

AWS Criptografía de pagos

AWS Criptografía de pagos: Guía del usuario

Table of Contents

¿Qué es la Criptografía de pagos de AWS?

La Criptografía de pagos de AWS es un servicio de AWS gestionado que proporciona acceso a las funciones criptográficas y a la gestión de claves que se utilizan en el procesamiento de pagos de conformidad con los estándares del sector de las tarjetas de pago (Payment Card Industry, PCI), sin necesidad de adquirir instancias de HSM de pago dedicadas. AWS La Criptografía de pagos ofrece a los clientes que realizan funciones de pago, como los adquirentes, los facilitadores de pagos, las redes, los conmutadores, los procesadores y los bancos, la posibilidad de acercar sus operaciones criptográficas de pago a las aplicaciones en la nube y minimizar la dependencia de los centros de datos auxiliares o las instalaciones de coubicación que contienen HSM de pago dedicados.

El servicio está diseñado para cumplir con las normas aplicables de la industria, incluidas PCI PIN, PCI P2PE y PCI DSS, y aprovecha el hardware que cuenta con la certificación PCI PTS HSM V3 y FIPS 140-2 de nivel 3. Está diseñado para soportar una baja latencia y altos niveles de tiempo de actividad y resiliencia. AWS La criptografía de pago es totalmente elástica y elimina muchos de los requisitos operativos de los HSM en las instalaciones, como la necesidad de aprovisionar hardware, gestionar de forma segura el material clave y mantener copias de seguridad de emergencia en instalaciones seguras. AWS La criptografía de pagos también ofrece la opción de compartir claves con sus socios electrónicamente, eliminando la necesidad de compartir componentes de texto claro en papel.

Puede usar la API del plano de control de Criptografía de pagos de AWS para crear y administrar claves.

Puede usar la API del plano de datos de la Criptografía de pagos de AWS para usar claves de cifrado para el procesamiento de transacciones relacionadas con los pagos y las operaciones criptográficas asociadas.

La Criptografía de pagos de AWS ofrece características importantes que puede utilizar para administrar sus claves:

- Cree y gestione claves de la Criptografía de pagos de AWS simétricas y asimétricas, incluidas las claves TDES, AES y RSA, y especifique su propósito, por ejemplo, para la generación de CVV o la obtención de claves DUKPT.
- Almacene automáticamente sus claves de la Criptografía de pagos de AWS de forma segura, protegidas por módulos de seguridad de hardware (Hardware Security Modules, HSM), al tiempo que aplica la separación de claves entre casos de uso.

- Cree, elimine, enumere y actualice alias, que son "nombres amistosos" que pueden utilizarse para acceder o controlar el acceso a sus claves de la Criptografía de pagos de AWS.
- Etiquete sus claves de la Criptografía de pagos de AWS para su identificación, agrupación, automatización, control de acceso y seguimiento de costos.
- Importe y exporte claves simétricas entre la Criptografía de pagos de AWS y su HSM (o terceros) utilizando claves de cifrado (KEK) siguiendo la TR-31 (Especificación de bloques de claves para el intercambio seguro e interoperable de claves).
- Importe y exporte claves de cifrado simétricas (KEK) entre la Criptografía de pagos de AWS y otros sistemas que utilicen pares de claves asimétricas siguiendo medios electrónicos como el TR-34 (Método de distribución de claves simétricas utilizando técnicas asimétricas).

Puede utilizar sus claves de la Criptografía de pagos de AWS en operaciones criptográficas, como:

- Cifrar, descifrar y volver a cifrar datos con claves simétricas o asimétricas de la Criptografía de pagos de AWS.
- Traducir de forma segura los datos sensibles (como los PIN de los titulares de tarjetas) entre claves de cifrado sin exponer el texto en claro, de acuerdo con las normas PCI sobre PIN.
- Generar o validar los datos del titular de la tarjeta, como el CVV, el CVV2 o el ARQC.
- Generar y validar los pines del titular de la tarjeta.
- Generar o validar las firmas MAC.

Conceptos

Conozca los términos y conceptos básicos utilizados en la Criptografía de pagos de AWS y cómo puede utilizarlos para ayudarle a proteger sus datos.

Alias

> Un nombre fácil de utilizar que se encuentra asociado a una clave de Criptografía de pagos de AWS. El alias se puede utilizar de forma indistinta con clave ARN en muchas de las operaciones de la Criptografía de pagos de AWS. Los alias permiten rotar o cambiar las claves sin que ello afecte al código de su aplicación. El nombre de alias es una cadena de hasta 256 caracteres. Identifica de forma inequívoca una clave de la Criptografía de pagos de AWS asociada dentro de una cuenta y región. En la Criptografía de pagos de AWS, los nombres de alias siempre empiezan por `alias/`.

El formato de un nombre de alias es el siguiente:

```
alias/<alias-name>
```

Por ejemplo:

```
alias/sampleAlias2
```

ARN de clave

El ARN de la clave es el nombre de recurso de Amazon (ARN) de una entrada de clave en la Criptografía de pagos de AWS. Es un identificador único y completo para la clave de la Criptografía de pagos de AWS. Un ARN clave incluye Cuenta de AWS, una región y un ID generado aleatoriamente. El ARN no está relacionado ni se deriva del material de la clave. Como se asignan automáticamente durante las operaciones de creación o importación, estos valores no son idempotentes. Si se importa la misma clave varias veces, se obtendrán varios ARN de clave con su propio ciclo de vida.

El formato de un ARN de clave es el siguiente:

```
arn:<partition>:payment-cryptography:<region>:<account-id>:alias/<alias-name>
```

A continuación, se muestra un ARN de clave de ejemplo.

```
arn:aws:payment-cryptography:us-east-2:111122223333:key/kwapwa6qaifllw2h
```

Claves de Criptografía de pagos de AWS

Las claves de Criptografía de pagos de AWS se utilizan para todas las funciones criptográficas. Las claves se generan directamente mediante el comando crear clave o se añaden al sistema al llamar a la importación de claves. El origen de una clave se puede determinar revisando el atributo keyOrigin. AWS La Criptografía de pagos también admite claves derivadas o intermedias utilizadas durante las operaciones criptográficas, como las utilizadas por DUKPT.

Estas claves tienen atributos inmutables y mutables definidos en el momento de su creación. Los atributos, como el algoritmo, la longitud y el uso, se definen en el momento de la creación y no se pueden cambiar. Otros, como la fecha de entrada en vigor o la fecha de caducidad, se pueden modificar. Consultar la Referencia de la API de Criptografía de pagos de AWS para obtener una lista completa de los atributos clave de la Criptografía de pagos de AWS.

Las claves de Criptografía de pagos de AWS tienen tipos de claves, definidos principalmente en la norma ANSI X9 TR 31, que restringen su uso a los fines previstos, tal como se especifica en el requisito 19 del PCI PIN v3.1.

Los atributos se vinculan a las claves mediante bloques de claves cuando se almacenan, se comparten con otras cuentas o se exportan, tal y como se especifica en el requisito 18-3 de PCI PIN v3.1.

Las claves se identifican en la plataforma de Criptografía de pagos de AWS mediante un valor único conocido como nombre de recurso clave de Amazon (ARN).

> **Note**
>
> La clave ARN se genera cuando se crea o importa inicialmente una clave en el servicio de la Criptografía de pagos de AWS. Por lo tanto, si añade el mismo material clave varias veces utilizando la funcionalidad de importación de claves, el mismo material clave se encontrará bajo varias claves pero cada una con un ciclo de vida clave diferente.

Terminología del sector

Temas

Tipos de claves comunes

AWK

Una clave de trabajo del adquirente (AcquirerWworking Key, AWK) es una clave que se utiliza normalmente para intercambiar datos entre un adquirente o el procesador del adquirente y una red (como Visa o Mastercard). Históricamente, AWK utiliza el 3DES para el cifrado y se representa como TR31_P0_PIN_ENCRYPTION_KEY.

BDK

Una clave de derivación base (Base Derivation Key, BDK) es una clave funcional que se utiliza para derivar claves posteriores y se utiliza normalmente como parte del proceso PCI PIN y PCI P2PE DUKPT. Se denomina TR31_B0_BASE_DERIVATION_KEY.

CMK

Una clave maestra de tarjeta (Card Master Key, CMK) es una o más claves específicas de una tarjeta que normalmente se derivan de una Clave maestra del emisor, PAN y PSN y, por lo general, son claves 3DES. Estas claves se almacenan en el chip EMV durante la personalización. Entre los ejemplos de CMK se incluyen las teclas AC, SMI y SMC.

CMK-AC

Una clave de criptograma de aplicación (AC) se utiliza como parte de las transacciones EMV para generar el criptograma de la transacción y es un tipo de clave maestra de tarjeta.

CMK-SMI

Una clave de integridad de mensajería segura (Secure Messaging Integrity, SMI) se utiliza como parte de EMV para verificar la integridad de las cargas útiles enviadas a la tarjeta mediante MAC, como los scripts de actualización de PIN. Es un tipo de clave maestra de la tarjeta.

CMK-SMC

Como parte del EMV, se utiliza una clave de confidencialidad segura de la mensajería (Secure Messaging Confidentiality, SMC) para cifrar los datos enviados a la tarjeta, como las actualizaciones del PIN. Es un tipo de clave maestra de la tarjeta.

CVK

Una clave de verificación de tarjeta (Card Verification Key, CVK) es una clave que se utiliza para generar valores CVV, CVV2 y similares mediante un algoritmo definido, así como para validar una entrada. Se denomina TR31_C0_CARD_VERIFICATION_KEY.

IMK

Una clave maestra del emisor (Issuer Master Key, IMK) es una clave maestra que se utiliza como parte de la personalización de la tarjeta con chip EMV. Normalmente, habrá 3 IMK: una para las claves AC (criptograma), SMI (clave maestra de script para integridad/firma) y SMC (clave maestra de script para confidencialidad/cifrado).

IK

Una clave inicial (IK) es la primera clave utilizada en el proceso DUKPT y se deriva de la clave de derivación base (BDK). No se procesa ninguna transacción en esta clave, pero se usa para derivar claves futuras que se usarán para las transacciones. El método de derivación para crear una IK se definió en X9. 24-1:2017. Cuando se utiliza un BDK TDES, el estándar aplicable es el X9. 24-1:2009 y el IK se sustituye por la clave de cifrado con PIN inicial (IPEK).

IPEK

Una clave de cifrado con PIN inicial (Initial PIN Encryption Key, IPEK) es la clave inicial que se utiliza en el proceso DUKPT y se deriva de la clave de derivación básica (Base Derivation Key, BDK). No se procesa ninguna transacción en esta clave, pero se usa para derivar claves futuras que se usarán para las transacciones. IPEK es un nombre inapropiado, ya que esta clave también se puede utilizar para derivar claves de cifrado de datos y de Mac. El método de derivación para crear un IPEK se definió en X9. 24-1:2009. Cuando se utiliza un BDK de AES, el estándar aplicable es el X9. 24-1:2017 y el IPEK se sustituye por la clave inicial (IK).

IWK

Una clave de trabajo del emisor (Issuer working key, IWK) es una clave que se utiliza normalmente para intercambiar datos entre un emisor/procesador del emisor y una red (como Visa o Mastercard). Históricamente, IWK utiliza el 3DES para el cifrado y se representa como TR31_P0_PIN_ENCRYPTION_KEY.

KEK

Una clave de cifrado clave (Key Encryption Key, KEK) es una clave que se utiliza para cifrar otras claves, ya sea para su transmisión o almacenamiento. Las claves destinadas a proteger otras claves suelen tener el valor TR31_K0_KEY_ENCRYPTION_KEY según el estándar KeyUsage. *TR-31*

PEK

Una clave de cifrado PIN (PIN Encryption Key, PEK) es un tipo de clave funcional que se utiliza para cifrar los PIN, ya sea para su almacenamiento o transmisión entre dos partes. IWK y AWK son dos ejemplos de usos específicos de las claves de cifrado con PIN. Estas claves se representan como TR31_P0_PIN_ENCRYPTION_KEY.

PVK

Una clave de verificación de PIN (PIN Verification Key, PVK) es un tipo de clave de trabajo que se utiliza para generar valores de verificación de PIN, como la PVV. Los dos tipos más comunes son TR31_V1_IBM3624_PIN_VERIFICATION_KEY que se utiliza para generar valores de compensación IBM3624 y TR31_V2_VISA_PIN_VERIFICATION_KEY que se utiliza para los valores de verificación Visa/ABA.

Otros términos

ARQC

El criptograma de solicitud de autorización (Authorization Request Cryptogram, ARQC) es un criptograma generado en el momento de la transacción mediante una tarjeta con chip estándar EMV (o una implementación sin contacto equivalente). Por lo general, un ARQC se genera mediante una tarjeta con chip y se envía al emisor o a su agente para su verificación en el momento de la transacción.

DUKPT

La clave única derivada por transacción (Derived Unique Key Per Transaction, DUKPT) es un estándar de administración de claves que se utiliza normalmente para definir el uso de claves de cifrado de un solo uso en puntos de venta o POI físicos. Históricamente, DUKPT utiliza el 3DES para el cifrado. El estándar industrial para el DUKPT se define en el ANSI X9.24-3-2017.

EMV

EMV (originalmente Europay, Mastercard y Visa) es un organismo técnico que trabaja con las partes interesadas en los pagos para crear estándares y tecnologías de pago interoperables. Un ejemplo de norma es el de las tarjetas con chip o sin contacto y los terminales de pago con los que interactúan, incluida la criptografía utilizada. La derivación de claves EMV se refiere a los métodos que permiten generar claves únicas para cada tarjeta de pago a partir de un conjunto inicial de claves, como una `IMK`

HSM

Un módulo de seguridad de hardware (Hardware Security Module, HSM) es un dispositivo físico que protege las operaciones criptográficas (por ejemplo, el cifrado, el descifrado y las firmas digitales), así como las claves subyacentes que se utilizan para estas operaciones.

KCV

El valor de comprobación de claves (Key Check Value, KCV) se refiere a una variedad de métodos de suma de comprobación que se utilizan principalmente para comparar claves entre sí sin tener acceso al material de las claves propiamente dichas. Los KCV también se han utilizado para validar la integridad (especialmente cuando se intercambian claves), aunque esta función ahora se incluye como parte de los formatos de bloques de claves, como `TR-31`. En el caso de las claves TDES, el KCV se calcula cifrando 8 bytes, cada uno con un valor igual a cero, con la clave que hay que comprobar y reteniendo los 3 bytes más importantes del resultado cifrado. En

el caso de las claves AES, el KCV se calcula mediante un algoritmo CMAC en el que los datos de entrada son 16 bytes de cero y se retienen los 3 bytes de orden superior del resultado cifrado.

KDH

Un host de distribución de claves (Key Distribution Host, KDH) es un dispositivo o sistema que envía claves en un proceso de intercambio de claves como el TR-34. Al enviar claves desde AWS Payment Cryptography, se considera KDH.

KIF

Un servicio de inyección de claves (Key Injection Facility, KIF) es un servicio seguro que se utiliza para inicializar los terminales de pago e incluso cargarlos con claves de cifrado.

KRD

Un dispositivo receptor de claves (Key Receiving Device, KRD) es un dispositivo que recibe claves en un proceso de intercambio de claves como el TR-34. Al enviar claves para la criptografía AWS de pagos, se considera el KRD.

KSN

Un número de serie clave (Key Serial Number, KSN) es un valor que se utiliza como entrada en el cifrado o descifrado DUKPT para crear claves de cifrado únicas por transacción. Por lo general, el KSN consta de un identificador BDK, un identificador de terminal semi-exclusivo y un contador de transacciones que se incrementa con cada transición procesada en un terminal de pago determinado.

PAN

El número de cuenta principal (Primary Account Number, PAN) es un identificador único para una cuenta, como una tarjeta de crédito o débito. Suele tener entre 13 y 19 dígitos. Los primeros 6 a 8 dígitos identifican la red y el banco emisor.

Bloqueo de PIN

Un bloque de datos que contiene un PIN durante el procesamiento o la transmisión, así como otros elementos de datos. Los formatos de bloque de PIN estandarizan el contenido del bloque de PIN y la forma en que se puede procesar para recuperar el PIN. La mayoría de los bloques de PIN están compuestos por el PIN, la longitud del PIN y, con frecuencia, contienen parte o todo el PAN. AWS La criptografía de pagos es compatible con los formatos ISO 9564-1 0, 1, 3 y 4. El formato 4 es obligatorio para las claves AES. Al verificar o traducir los PIN, es necesario especificar el bloque de PIN de los datos entrantes o salientes.

POI

El punto de interacción (Point of Interaction, POI), también utilizado con frecuencia como sinónimo de punto de venta (Point of Sale, POS), es el dispositivo de hardware con el que el titular de la tarjeta interactúa para presentar su credencial de pago. Un ejemplo de POI es la terminal física de un establecimiento comercial. Para ver la lista de terminales POI PCI PTS certificados, consulte el sitio web de PCI.

PSN

El número de secuencia PAN (PAN Sequence Number, PSN) es un valor numérico que se utiliza para diferenciar varias tarjetas emitidas con el mismo PAN.

Clave pública

Cuando se utilizan cifrados asimétricos (RSA), la clave pública es el componente público de un par de claves pública-privada. La clave pública se puede compartir y distribuir a entidades que necesitan cifrar datos para el propietario del par de claves público-privado. Para las operaciones de firma digital, la clave pública se utiliza a fin de verificar la firma.

Clave privada

Cuando se utilizan cifrados asimétricos (RSA), la clave privada es el componente privado de un par de claves pública-privada. La clave privada se utiliza para descifrar los datos o crear firmas digitales. Al igual que las claves de criptografía AWS de pago simétricas, los HSM crean claves privadas de forma segura. Solo se descifran en la memoria volátil del HSM y únicamente durante el tiempo necesario para procesar su solicitud criptográfica.

PVV

El valor de verificación del PIN (PIN Verification Value, PVV) es un valor derivado algorítmicamente de una serie de entradas, como el número de la tarjeta y el PIN, que genera un valor que se puede utilizar para su posterior validación. Uno de estos esquemas se conoce como Visa PVV (también conocido como método ABA), aunque se utiliza para los PIN de cualquier red.

RSA Wrap/Unwrap

La envoltura RSA utiliza una clave asimétrica para envolver una clave simétrica (como una clave TDES) para su transmisión a otro sistema. Solo el sistema con la clave privada coincidente puede descifrar la carga útil y cargar la clave simétrica. Por el contrario, RSA unwrap descifrará de forma segura una clave cifrada con RSA y, a continuación, la cargará en la criptografía de pagos. AWS El empaquetado RSA es un método de bajo nivel para intercambiar claves y no transmite las

claves en formato de bloque de claves ni utiliza la firma de carga útil por parte de la parte que las envía. Se deben considerar controles alternativos para determinar la procedencia y comprobar que los atributos clave no están mutados.

El TR-34 también utiliza RSA internamente, pero es un formato independiente y no es interoperable.

TR-31

TR-31 (definido formalmente como ANSI X9 TR 31) es un formato de bloques clave definido por el Instituto Nacional de Normalización de los Estados Unidos (American National Standards Institute, ANSI) para permitir la definición de los atributos clave en la misma estructura de datos que los propios datos clave. El formato de bloque de teclas TR-31 define un conjunto de atributos clave que están vinculados a la clave para que se mantengan unidos. AWS La criptografía de pagos utiliza términos estandarizados TR-31 siempre que es posible para garantizar una separación y un propósito adecuados de las claves. TR-31 ha sido sustituida por la norma ANSI X9.143-2022.

TR-34

El TR-34 es una implementación de la norma ANSI X9.24-2 que describe un protocolo para distribuir de forma segura claves simétricas (como el 3DES y el AES) mediante técnicas asimétricas (como el RSA). AWS La criptografía de pagos utiliza los métodos TR-34 para permitir la importación y exportación seguras de las claves.

Servicios relacionados

AWS Key Management Service

El servicio de gestión de claves de AWS (KMS de AWS) es un servicio gestionado que le facilita la creación y el control de las claves criptográficas que se utilizan para proteger sus datos. AWS KMS utiliza módulos de seguridad de hardware (HSM) para proteger y validar sus claves KMS de AWS.

AWS CloudHSM

AWS CloudHSM proporciona a los clientes instancias HSM dedicadas de propósito general en la Nube AWS. AWS CloudHSM puede proporcionar diversas funciones criptográficas como la creación de claves, la firma de datos o el cifrado y descifrado de datos.

Para obtener más información

- Para conocer los términos y conceptos utilizados en la Criptografía de pagos de AWS, consulte Conceptos de la criptografía de pagos de AWS.
- Para obtener información sobre la API del plano de control de la Criptografía de pagos de AWS, consulte Referencia de la API del plano de control de Criptografía de pagos de AWS.
- Para obtener información sobre la API del plano de datos de la Criptografía de pagos de AWS, consulte Referencia de la API del plano de datos de Criptografía de pagos de AWS.
- Para obtener información técnica detallada sobre cómo la Criptografía de pagos de AWS utiliza la criptografía y asegura las claves de la Criptografía de pagos de AWS, consulte Detalles criptográficos.

Puntos finales para AWS Payment Cryptography

Para conectarse mediante programación AWS Payment Cryptography, utilice un punto final, la URL del punto de entrada al servicio. AWS Los SDK y las herramientas de línea de comandos utilizan automáticamente el punto final predeterminado para el servicio en Región de AWS función del contexto regional de la solicitud, por lo que normalmente no es necesario establecer estos valores de forma explícita. Cuando sea necesario, puedes especificar un punto final diferente para tus solicitudes de API.

Puntos de conexión del plano de control

Nombre de la región	Región	Punto de conexión	Protocolo
Este de EE. UU. (Norte de Virginia)	`us-east-1`	controlplane.payment-cryptography.us-east-1.amazonaws.com	HTTPS
Este de EE. UU. (Ohio)	`us-east-2`	plano de control. payment-cryptography.us-east-2.amazonaws.com	HTTPS
Oeste de EE. UU. (Oregón)	`us-west-2`	controlplane.payment-cryptography.us-west-2.amazonaws.com	HTTPS

Puntos de conexión del plano de datos

Nombre de la región	Región	Punto de conexión	Protocolo
Este de EE. UU. (Norte de Virginia)	`us-east-1`	dataplane.payment-cryptography.us-east-1.amazonaws.com	HTTPS
Este de EE. UU. (Ohio)	`us-east-2`	plano de datos. payment-cryptography.us-east-2.amazonaws.com	HTTPS
Oeste de EE. UU. (Oregón)	`us-west-2`	dataplane.payment-cryptography.us-west-2.amazonaws.com	HTTPS

Introducción a la Criptografía de pagos de AWS

Para empezar con la Criptografía de pagos de AWS, primero deberá crear claves y, después, utilizarlas en diversas operaciones criptográficas. El siguiente tutorial presenta un caso de uso sencillo para generar una clave que se utilizará para generar/verificar los valores del CVV2. Para probar otros ejemplos y explorar los patrones de implementación en AWS, pruebe el siguiente Taller de criptografía de pagos de AWS o explore nuestro proyecto de muestra disponible en Github

En este tutorial, se explica cómo crear una clave única y cómo realizar operaciones criptográficas con ella. Después, borra la clave si ya no la desea, completando el ciclo de vida de la clave.

Temas

Requisitos previos

Antes de comenzar, asegúrese de que:

- Tiene permiso para acceder al servicio. Para obtener más información, consulte Políticas de IAM.
- Ha instalado AWS CLI. También puedes usar SDK de AWS o API de AWS para acceder a la Criptografía de pagos de AWS, pero en las instrucciones de este tutorial se utiliza AWS CLI.

Paso 1: crear una clave

El primer paso es crear una clave. Para este tutorial, debe crear una clave CVK 3DES de doble longitud (2KEY TDES) para generar y verificar los valores CVV/CVV2.

```
$ aws payment-cryptography create-key \
    --exportable
```

```
    --key-attributes KeyAlgorithm=TDES_2KEY,KeyUsage=TR31_C0_CARD_VERIFICATION_KEY,\
        KeyClass=SYMMETRIC_KEY,\
        KeyModesOfUse='{Generate=true,Verify=true}'
```

La respuesta refleja los parámetros de la solicitud, incluyendo un ARN para las llamadas posteriores y un valor de verificación clave (KCV).

```
{
    "Key": {
        "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
tqv5yij6wtxx64pi",
        "KeyAttributes": {
            "KeyUsage": "TR31_C0_CARD_VERIFICATION_KEY",
            "KeyClass": "SYMMETRIC_KEY",
            "KeyAlgorithm": "TDES_2KEY",
            "KeyModesOfUse": {
                "Encrypt": false,
                "Decrypt": false,
                "Wrap": false,
                "Unwrap": false,
                "Generate": true,
                "Sign": false,
                "Verify": true,
                "DeriveKey": false,
                "NoRestrictions": false
            }
        },
        "KeyCheckValue": "CADDA1",
        "KeyCheckValueAlgorithm": "ANSI_X9_24",
        "Enabled": true,
        "Exportable": true,
        "KeyState": "CREATE_COMPLETE",
        "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
        "CreateTimestamp": "2023-06-05T06:41:46.648000-07:00",
        "UsageStartTimestamp": "2023-06-05T06:41:46.626000-07:00"
    }
}
```

Tome nota de `KeyArn` que representa la clave, por ejemplo arn:aws:payment-cryptography:us-east-2:111122223333:key/tqv5yij6wtxx64pi. Lo necesitará en el siguiente paso.

Paso 2: Generar un valor CVV2 con la clave

En este paso, se genera un CVV2 para una PAN y fecha de caducidad determinada utilizando la clave del paso 1.

```
$  aws payment-cryptography-data generate-card-validation-data \
    --key-identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/
tqv5yij6wtxx64pi \
    --primary-account-number=171234567890123 \
    --generation-attributes CardVerificationValue2={CardExpiryDate=0123}
```

```
{
    "CardDataGenerationKeyCheckValue": "CADDA1",
    "CardDataGenerationKeyIdentifier": "arn:aws:payment-cryptography:us-
east-2:111122223333:key/tqv5yij6wtxx64pi",
    "CardDataType": "CARD_VERIFICATION_VALUE_2",
    "CardDataValue": "144"
}
```

Tome nota de `cardDataValue`, en este caso el número de 3 dígitos 144. Lo necesitará en el siguiente paso.

Paso 3: Verificar el valor generado en el paso 2

En este ejemplo, usted valida el CVV2 del paso 2 con la clave que creó en el paso 1.

Ejecute el siguiente comando para validar el CVV2.

```
$  aws payment-cryptography-data  verify-card-validation-data \
    --key-identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/
tqv5yij6wtxx64pi \
    --primary-account-number=171234567890123 \
    --verification-attributes CardVerificationValue2={CardExpiryDate=0123} \
    --validation-data 144
```

```
{
    "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
tqv5yij6wtxx64pi",
    "KeyCheckValue": "CADDA1"
```

```
}
```

El servicio devuelve una respuesta HTTP de 200 para indicar que ha validado el CVV2.

Paso 4: Realizar una prueba negativa

En este paso, se crea una prueba negativa en la que el CVV2 no es correcto y no se valida. Usted intenta validar un CVV2 incorrecto con la clave que creó en el paso 1. Esta es una operación esperada, por ejemplo, si el titular de la tarjeta inserta un CVV2 incorrecto al finalizar la compra.

```
$  aws payment-cryptography-data  verify-card-validation-data \
    --key-identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/
tqv5yij6wtxx64pi \
    --primary-account-number=171234567890123 \
    --verification-attributes CardVerificationValue2={CardExpiryDate=0123} \
    --validation-data 999
```

```
Card validation data verification failed.
```

El servicio devuelve una respuesta HTTP de 400 con el mensaje «Fallo en la verificación de los datos de validación de la tarjeta» y el motivo INVALID_VALIDATION_DATA.

Paso 5: Eliminación (opcional)

Ahora puede eliminar la clave que creó en el paso 1. Para minimizar los cambios irrecuperables, el periodo de eliminación de claves predeterminado es de siete de días.

```
$ aws payment-cryptography delete-key \
    --key-identifier=arn:aws:payment-cryptography:us-east-2:111122223333:key/
tqv5yij6wtxx64pi
```

```
{
    "Key": {
        "CreateTimestamp": "2022-10-27T08:27:51.795000-07:00",
        "DeletePendingTimestamp": "2022-11-03T13:37:12.114000-07:00",
        "Enabled": true,
        "Exportable": true,
        "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
tqv5yij6wtxx64pi",
```

```
        "KeyAttributes": {
            "KeyAlgorithm": "TDES_3KEY",
            "KeyClass": "SYMMETRIC_KEY",
            "KeyModesOfUse": {
                "Decrypt": true,
                "DeriveKey": false,
                "Encrypt": true,
                "Generate": false,
                "NoRestrictions": false,
                "Sign": false,
                "Unwrap": true,
                "Verify": false,
                "Wrap": true
            },
            "KeyUsage": "TR31_P0_PIN_ENCRYPTION_KEY"
        },
        "KeyCheckValue": "CADDA1",
        "KeyCheckValueAlgorithm": "ANSI_X9_24",
        "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
        "KeyState": "DELETE_PENDING",
        "UsageStartTimestamp": "2022-10-27T08:27:51.753000-07:00"
    }
}
```


Tome nota de dos campos en la salida. El `deletePendingTimestamp` se fija por defecto para los siete días posteriores. El KeyState está establecido en `DELETE_PENDING`. Puede cancelar esta eliminación en cualquier momento antes de la hora de eliminación programada llamando al `restore-key`.

Administración de claves

Para empezar con la criptografía AWS de pagos, querrás crear una clave de criptografía AWS de pagos.

En los temas de esta sección se explica cómo crear y administrar diversos tipos de claves de criptografía de AWS pagos, desde su creación hasta su eliminación. Incluye temas sobre la creación, edición y visualización de claves, el etiquetado de claves, la creación de alias de claves, así como la habilitación y deshabilitación de claves.

Temas

Generación de claves

Puede crear claves de criptografía de AWS pagos mediante la operación de la CreateKey API. Durante este proceso, especificará varios atributos de la clave o del resultado resultante, como el algoritmo clave (por ejemplo, TDES_3KEY), las operaciones permitidas (por ejemplo, TDES_3KEY) (por ejemplo, TR31_P0_PIN_ENCRYPTION_KEY), las operaciones permitidas KeyUsage (por ejemplo, cifrar, firmar) y si es exportable. No puede cambiar estas propiedades una vez creada la clave de criptografía de pagos. AWS

Ejemplos

- Creación de una clave asimétrica (RSA)
- Generar una clave de valor de verificación de PIN (PVV)

Generación de una clave 2KEY TDES

Example

Este comando genera una clave 2KEY TDES con el fin de generar y verificar valores CVV o CVV2. La respuesta devuelve los parámetros de la solicitud, incluyendo un ARN para llamadas posteriores, así como un valor de comprobación de la clave (KCV).

```
$ aws payment-cryptography create-key --exportable --key-attributes
 KeyAlgorithm=TDES_2KEY,\
KeyUsage=TR31_C0_CARD_VERIFICATION_KEY,KeyClass=SYMMETRIC_KEY, \
KeyModesOfUse='{Generate=true,Verify=true}'
```

```
                    {
    "Key": {
        "CreateTimestamp": "2022-10-26T16:04:11.642000-07:00",
        "Enabled": true,
        "Exportable": true,
        "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
hjprdg5o4jtgs5tw",
        "KeyAttributes": {
            "KeyAlgorithm": "TDES_2KEY",
            "KeyClass": "SYMMETRIC_KEY",
            "KeyModesOfUse": {
                "Decrypt": false,
                "DeriveKey": false,
                "Encrypt": false,
                "Generate": true,
                "NoRestrictions": false,
                "Sign": false,
                "Unwrap": false,
                "Verify": true,
                "Wrap": false
            },
            "KeyUsage": "TR31_C0_CARD_VERIFICATION_KEY"
        },
```

```
        "KeyCheckValue": "B72F",
        "KeyCheckValueAlgorithm": "ANSI_X9_24",
        "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
        "KeyState": "CREATE_COMPLETE",
        "UsageStartTimestamp": "2022-10-26T16:04:11.559000-07:00"
    }
}
```

Generación de una clave de cifrado de PIN

Example Generación de una clave de cifrado de PIN (PEK)

Este comando genera una clave 3KEY TDES con el propósito de encriptar valores PIN (conocida como clave de cifrado de PIN). Esta clave puede utilizarse para asegurar el almacenamiento de PIN o para descifrar PIN proporcionados durante un intento de verificación, por ejemplo, durante una transacción. La respuesta devuelve los parámetros de la solicitud, incluyendo un ARN para llamadas posteriores, así como un valor de comprobación de la clave (KCV).

```
$ aws payment-cryptography create-key --exportable --key-attributes \
                        KeyAlgorithm=TDES_3KEY,KeyUsage=TR31_P0_PIN_ENCRYPTION_KEY, \
                        KeyClass=SYMMETRIC_KEY,/

 KeyModesOfUse='{Encrypt=true,Decrypt=true,Wrap=true,Unwrap=true}'
```

```
{
    "Key": {
        "CreateTimestamp": "2022-10-27T08:27:51.795000-07:00",
        "Enabled": true,
        "Exportable": true,
        "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
kwapwa6qaifllw2h",
        "KeyAttributes": {
            "KeyAlgorithm": "TDES_3KEY",
            "KeyClass": "SYMMETRIC_KEY",
            "KeyModesOfUse": {
                "Decrypt": true,
                "DeriveKey": false,
                "Encrypt": true,
```

```
                "Generate": false,
                "NoRestrictions": false,
                "Sign": false,
                "Unwrap": true,
                "Verify": false,
                "Wrap": true
            },
            "KeyUsage": "TR31_P0_PIN_ENCRYPTION_KEY"
        },
        "KeyCheckValue": "9CA6",
        "KeyCheckValueAlgorithm": "ANSI_X9_24",
        "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
        "KeyState": "CREATE_COMPLETE",
        "UsageStartTimestamp": "2022-10-27T08:27:51.753000-07:00"
    }
}
```

Creación de una clave asimétrica (RSA)

Example

En este ejemplo, generaremos un nuevo par de claves asimétricas de RSA de 2048 bits. Se generará una nueva clave privada así como la clave pública correspondiente. La clave pública se puede recuperar mediante la getPublicCertificateAPI.

```
$ aws payment-cryptography create-key --exportable \
--key-attributes
 KeyAlgorithm=RSA_2048,KeyUsage=TR31_D1_ASYMMETRIC_KEY_FOR_DATA_ENCRYPTION, \
KeyClass=ASYMMETRIC_KEY_PAIR,KeyModesOfUse='{Encrypt=true,
 Decrypt=True,Wrap=True,Unwrap=True}'
```

```
{
    "Key": {
        "CreateTimestamp": "2022-11-15T11:15:42.358000-08:00",
        "Enabled": true,
        "Exportable": true,
        "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
nsq2i3mbg6sn775f",
        "KeyAttributes": {
            "KeyAlgorithm": "RSA_2048",
```

```
            "KeyClass": "ASYMMETRIC_KEY_PAIR",
            "KeyModesOfUse": {
                "Decrypt": true,
                "DeriveKey": false,
                "Encrypt": true,
                "Generate": false,
                "NoRestrictions": false,
                "Sign": false,
                "Unwrap": true,
                "Verify": false,
                "Wrap": true
            },
            "KeyUsage": "TR31_D1_ASYMMETRIC_KEY_FOR_DATA_ENCRYPTION"
        },
        "KeyCheckValue": "40AD487F",
        "KeyCheckValueAlgorithm": "CMAC",
        "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
        "KeyState": "CREATE_COMPLETE",
        "UsageStartTimestamp": "2022-11-15T11:15:42.182000-08:00"
    }
}
```

Generar una clave de valor de verificación de PIN (PVV)

Example

Este comando genera una clave 3KEY TDES con el fin de generar valores PVV (conocidos como valor de verificación de PIN). Puede utilizar esta clave para generar un valor PVV que pueda compararse con un PVV calculado posteriormente. La respuesta devuelve los parámetros de la solicitud, incluyendo un ARN para llamadas posteriores, así como un valor de comprobación de la clave (KCV).

```
$ aws payment-cryptography create-key --exportable/
--key-attributes KeyAlgorithm=TDES_3KEY,KeyUsage=TR31_V2_VISA_PIN_VERIFICATION_KEY,/
KeyClass=SYMMETRIC_KEY,KeyModesOfUse='{Generate=true,Verify=true}'
```

```
{
    "Key": {
        "CreateTimestamp": "2022-10-27T10:22:59.668000-07:00",
        "Enabled": true,
```

```
        "Exportable": true,
        "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
j4u4cmnzkelhc6yb",
        "KeyAttributes": {
            "KeyAlgorithm": "TDES_3KEY",
            "KeyClass": "SYMMETRIC_KEY",
            "KeyModesOfUse": {
                "Decrypt": false,
                "DeriveKey": false,
                "Encrypt": false,
                "Generate": true,
                "NoRestrictions": false,
                "Sign": false,
                "Unwrap": false,
                "Verify": true,
                "Wrap": false
            },
            "KeyUsage": "TR31_V2_VISA_PIN_VERIFICATION_KEY"
        },
        "KeyCheckValue": "5132",
        "KeyCheckValueAlgorithm": "ANSI_X9_24",
        "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
        "KeyState": "CREATE_COMPLETE",
        "UsageStartTimestamp": "2022-10-27T10:22:59.614000-07:00"
    }
}
```

Enumerar claves

Listar claves presenta una lista de claves accesibles a la persona que llama en esta cuenta y región.

Example

```
$ aws payment-cryptography list-keys
```

```
 {"Keys": [
    {
        "CreateTimestamp": "2022-10-12T10:58:28.920000-07:00",
        "Enabled": false,
        "Exportable": true,
```

```
        "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
alsuwfxug3pgy6xh",
        "KeyAttributes": {
            "KeyAlgorithm": "TDES_3KEY",
            "KeyClass": "SYMMETRIC_KEY",
            "KeyModesOfUse": {
                "Decrypt": true,
                "DeriveKey": false,
                "Encrypt": true,
                "Generate": false,
                "NoRestrictions": false,
                "Sign": false,
                "Unwrap": true,
                "Verify": false,
                "Wrap": true
            },
            "KeyUsage": "TR31_P1_PIN_GENERATION_KEY"
        },
        "KeyCheckValue": "369D",
        "KeyCheckValueAlgorithm": "ANSI_X9_24",
        "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
        "KeyState": "CREATE_COMPLETE",
        "UsageStopTimestamp": "2022-10-27T14:19:42.488000-07:00"
    }
    ]
}
```


Habilitación y deshabilitación de claves

Puede deshabilitar y volver a activar las claves de criptografía AWS de pagos. Al crear una clave, se habilita de forma predeterminada. Si desactiva una clave, no se puede utilizar en ninguna operación criptográfica hasta que la vuelva a activar. Los comandos de inicio o parada de uso tienen efecto inmediato, por lo que se recomienda que revise el uso antes de realizar un cambio de este tipo. También puede establecer un cambio (iniciar o detener el uso) para que surta efecto en el futuro utilizando el parámetro `timestamp` opcional.

Como es temporal y se deshace fácilmente, deshabilitar una clave de criptografía de AWS pago es una alternativa más segura que eliminar una clave de criptografía de AWS pago, una acción destructiva e irreversible. Si está pensando en eliminar una clave de criptografía de AWS pagos, desactívela primero y asegúrese de que no necesitará utilizarla para cifrar o descifrar datos en el futuro.

Temas

Iniciar el uso de claves

El uso de claves debe estar habilitado para poder utilizar una clave para operaciones criptográficas. Si una clave no está habilitada, puede utilizar esta operación para hacerla utilizable. El campo `UsageStartTimestamp` representará cuándo se activó o activará la clave. Esto será en el pasado para un token habilitado, y en el futuro si está pendiente de activación.

Example

En este ejemplo, se solicita la habilitación de una llave para su uso. La respuesta incluye la información de la llave y el indicador de habilitación ha pasado a verdadero. Esto también se reflejará en el objeto de respuesta lista-claves.

```
$ aws payment-cryptography start-key-usage --key-identifier "arn:aws:payment-
cryptography:us-east-2:111122223333:key/alsuwfxug3pgy6xh"
```

```
{
      "Key": {
      "CreateTimestamp": "2022-10-12T10:58:28.920000-07:00",
      "Enabled": true,
      "Exportable": true,
      "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
alsuwfxug3pgy6xh",
      "KeyAttributes": {
          "KeyAlgorithm": "TDES_3KEY",
          "KeyClass": "SYMMETRIC_KEY",
          "KeyModesOfUse": {
              "Decrypt": true,
              "DeriveKey": false,
              "Encrypt": true,
              "Generate": false,
              "NoRestrictions": false,
              "Sign": false,
              "Unwrap": true,
              "Verify": false,
              "Wrap": true
```

```
          },
          "KeyUsage": "TR31_P1_PIN_GENERATION_KEY"
      },
      "KeyCheckValue": "369D",
      "KeyCheckValueAlgorithm": "ANSI_X9_24",
      "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
      "KeyState": "CREATE_COMPLETE",
      "UsageStartTimestamp": "2022-10-27T14:09:59.468000-07:00"
  }
}
```

Detener el uso de claves

Si ya no tiene previsto utilizar una clave, puede detener el uso de la clave para evitar que se realicen más operaciones criptográficas. Esta operación no es permanente, por lo que puede revertirla utilizando iniciar uso de clave. También puede configurar una clave para que se desactive en el futuro. El campo `UsageStopTimestamp` representará cuándo la clave se deshabilitó o deshabilitará.

Example

En este ejemplo, se solicita detener el uso de la llave en el futuro. Tras la ejecución, esta clave no podrá utilizarse para operaciones criptográficas a menos que se vuelva a habilitar mediante iniciar uso de clave. La respuesta incluye la información de la clave y que el indicador de habilitación haya pasado a falso. Esto también se reflejará en el objeto de respuesta lista-claves.

```
$ aws payment-cryptography stop-key-usage --key-identifier "arn:aws:payment-
cryptography:us-east-2:111122223333:key/alsuwfxug3pgy6xh"
```

```
{
  "Key": {
      "CreateTimestamp": "2022-10-12T10:58:28.920000-07:00",
      "Enabled": false,
      "Exportable": true,
      "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
alsuwfxug3pgy6xh",
      "KeyAttributes": {
          "KeyAlgorithm": "TDES_3KEY",
          "KeyClass": "SYMMETRIC_KEY",
          "KeyModesOfUse": {
              "Decrypt": true,
              "DeriveKey": false,
```

```
            "Encrypt": true,
            "Generate": false,
            "NoRestrictions": false,
            "Sign": false,
            "Unwrap": true,
            "Verify": false,
            "Wrap": true
        },
        "KeyUsage": "TR31_P1_PIN_GENERATION_KEY"
    },
    "KeyCheckValue": "369D",
    "KeyCheckValueAlgorithm": "ANSI_X9_24",
    "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
    "KeyState": "CREATE_COMPLETE",
    "UsageStopTimestamp": "2022-10-27T14:09:59.468000-07:00"
  }
}
```


Eliminación de claves

Al eliminar una clave de criptografía de AWS pago, se eliminan el material de la clave y todos los metadatos asociados a la clave y es irreversible, a menos que haya una copia de la clave disponible fuera de la criptografía de AWS pago. Una vez que se elimina una clave, ya no pueden descifrar los datos que se habían cifrado con ella, lo que significa que los datos pueden volverse irrecuperables. Sólo debe eliminar una clave cuando esté seguro de que ya no necesita utilizarla y de que no hay terceros que la estén utilizando. Si no está seguro, considere la posibilidad de desactivar la clave en lugar de eliminarla. Puede volver a activar una clave desactivada si necesita volver a utilizarla más adelante, pero no podrá recuperar una clave de criptografía de AWS pagos eliminada a menos que pueda volver a importarla desde otra fuente.

Antes de eliminar una clave, asegúrate de que ya no la necesitas. AWS La criptografía de pagos no almacena los resultados de las operaciones criptográficas, como ocurre con el CVV2, y no puede determinar si se necesita una clave para cualquier material criptográfico persistente.

AWS La criptografía de pagos nunca elimina las claves que pertenecen a las AWS cuentas activas, a menos que se programe explícitamente su eliminación y caduque el período de espera obligatorio.

Sin embargo, puede optar por eliminar una clave de criptografía de AWS pago por uno o varios de los siguientes motivos:

- Para completar el ciclo de vida de una clave que ya no necesita

- Para evitar los gastos de administración asociados con el mantenimiento de las claves de criptografía AWS de pago no utilizadas

> **Note**
>
> Si cierras o eliminas la tuya Cuenta de AWS, tu clave de criptografía de AWS pago quedará inaccesible. No necesita programar la eliminación de su clave de criptografía de AWS pago aparte del cierre de la cuenta.

AWS La criptografía de pagos registra una entrada en su AWS CloudTrailregistro cuando programa la eliminación de la clave criptográfica de AWS pagos y cuando se elimina realmente la clave de criptografía de AWS pagos.

Acerca del período de espera

Dado que eliminar una clave es irreversible, la criptografía de AWS pagos requiere que establezcas un período de espera de entre 3 y 180 días. El periodo de espera predeterminado es de siete días.

Sin embargo, el período de espera real puede ser hasta 24 horas más largo que el programado. Para obtener la fecha y la hora reales en las que se eliminará la clave de criptografía de AWS pago, utilice las siguientes operaciones. GetKey Asegúrese de anotar la zona horaria.

Durante el período de espera, el estado de la clave AWS de criptografía de pago y el estado de la clave es Pendiente de eliminación.

> **Note**
>
> Una clave AWS de criptografía de pago pendiente de eliminación no se puede utilizar en ninguna operación criptográfica.

Una vez finalizado el período de espera, la criptografía de AWS pago elimina la clave de criptografía de AWS pago, sus alias y todos los metadatos de criptografía de pago relacionados. AWS

Utilice el período de espera para asegurarse de que no necesitará la clave de criptografía de AWS pago ahora o en el futuro. Si se da cuenta de que necesita la clave durante el periodo de espera, puede cancelar la eliminación de la clave antes de que finalice el periodo de espera. Una vez que finaliza el periodo de espera, no puede cancelar la eliminación de claves y el servicio elimina la clave.

Example

En este ejemplo, se solicita la eliminación de una clave. Además de la información clave básica, hay dos campos importantes: el estado de la clave se ha cambiado a DELETE_PENDING y deletePendingTimestamp representa el momento en el que está previsto eliminar la clave en ese momento.

```
$ aws payment-cryptography delete-key \
                    --key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/kwapwa6qaifllw2h
```

```
  {
    "Key": {
        "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
kwapwa6qaifllw2h",
        "KeyAttributes": {
            "KeyUsage": "TR31_V2_VISA_PIN_VERIFICATION_KEY",
            "KeyClass": "SYMMETRIC_KEY",
            "KeyAlgorithm": "TDES_3KEY",
            "KeyModesOfUse": {
                "Encrypt": false,
                "Decrypt": false,
                "Wrap": false,
                "Unwrap": false,
                "Generate": true,
                "Sign": false,
                "Verify": true,
                "DeriveKey": false,
                "NoRestrictions": false
            }
        },
        "KeyCheckValue": "",
        "KeyCheckValueAlgorithm": "ANSI_X9_24",
        "Enabled": false,
        "Exportable": true,
        "KeyState": "DELETE_PENDING",
        "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
        "CreateTimestamp": "2023-06-05T12:01:29.969000-07:00",
        "UsageStopTimestamp": "2023-06-05T14:31:13.399000-07:00",
        "DeletePendingTimestamp": "2023-06-12T14:58:32.865000-07:00"
```

```
    }
}
```

Example

En este ejemplo, se cancela un borrado pendiente. Una vez completada con éxito, la clave ya no se borrará según la programación anterior. La respuesta contiene la información básica de la clave; además, han cambiado dos campos relevantes: `KeyState` y `deletePendingTimestamp`. `KeyState` se devuelve a un valor de CREATE_COMPLETE, mientras que `DeletePendingTimestamp` se elimina.

```
$ aws payment-cryptography restore-key --key-identifier arn:aws:payment-
cryptography:us-east-2:111122223333:key/kwapwa6qaifllw2h
```

```
{
    "Key": {
        "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
kwapwa6qaifllw2h",
        "KeyAttributes": {
            "KeyUsage": "TR31_V2_VISA_PIN_VERIFICATION_KEY",
            "KeyClass": "SYMMETRIC_KEY",
            "KeyAlgorithm": "TDES_3KEY",
            "KeyModesOfUse": {
                "Encrypt": false,
                "Decrypt": false,
                "Wrap": false,
                "Unwrap": false,
                "Generate": true,
                "Sign": false,
                "Verify": true,
                "DeriveKey": false,
                "NoRestrictions": false
            }
        },
        "KeyCheckValue": "",
        "KeyCheckValueAlgorithm": "ANSI_X9_24",
        "Enabled": false,
        "Exportable": true,
        "KeyState": "CREATE_COMPLETE",
        "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
        "CreateTimestamp": "2023-06-08T12:01:29.969000-07:00",
```

```
        "UsageStopTimestamp": "2023-06-08T14:31:13.399000-07:00"
    }
}
```

Importar y exportar claves

AWS Las claves de criptografía de pago se pueden importar desde otras soluciones o exportar a otras soluciones (como otros HSM). Es un caso de uso común intercambiar claves con proveedores de servicios utilizando la funcionalidad de importación y exportación. Como servicio en la nube, la criptografía de AWS pagos adopta un enfoque electrónico moderno para la administración de claves y, al mismo tiempo, le ayuda a mantener el cumplimiento y los controles aplicables. El objetivo a largo plazo es abandonar los componentes de claves basados en papel y adoptar medios electrónicos de intercambio de claves basados en estándares.

Intercambio de claves de cifrado (KEK)

AWS La criptografía de pagos fomenta el uso de la criptografía de clave pública (RSA) para el intercambio inicial de claves mediante la consolidada norma ANSI X9.24 TR-34. Los nombres comunes para este tipo de clave inicial son la clave de cifrado de clave (KEK), la clave maestra de zona (ZMK) y la clave maestra de control de zona (ZCMK). Si sus sistemas o socios aún no son compatibles con el TR-34, también puede considerar la posibilidad de utilizar RSA Wrap/Unwrap.

Si tiene necesidad de seguir procesando componentes de claves en papel hasta que todos los socios admitan el intercambio electrónico de claves, puede considerar la posibilidad de retener un HSM fuera de línea para este fin.

> **Note**
>
> Si desea importar sus propias llaves de prueba, consulte el proyecto de ejemplo en Github. Para obtener instrucciones sobre cómo importar o exportar claves de otras plataformas, consulte la guía del usuario de dichas plataformas.

Intercambio de claves de trabajo (WK)

AWS La criptografía de pagos utiliza la norma industrial pertinente (ANSI X9.24 TR 31-2018) para intercambiar claves de trabajo. TR-31 asume que se ha intercambiado previamente una KEK.

Esto es coherente con el requisito de PCI PIN de vincular criptográficamente el material de la clave a su tipo de clave y uso en todo momento. Las claves de trabajo tienen varios nombres, incluyendo claves de trabajo del adquirente, claves de trabajo del emisor, BDK, IPEK, etc.

Temas

Importar claves

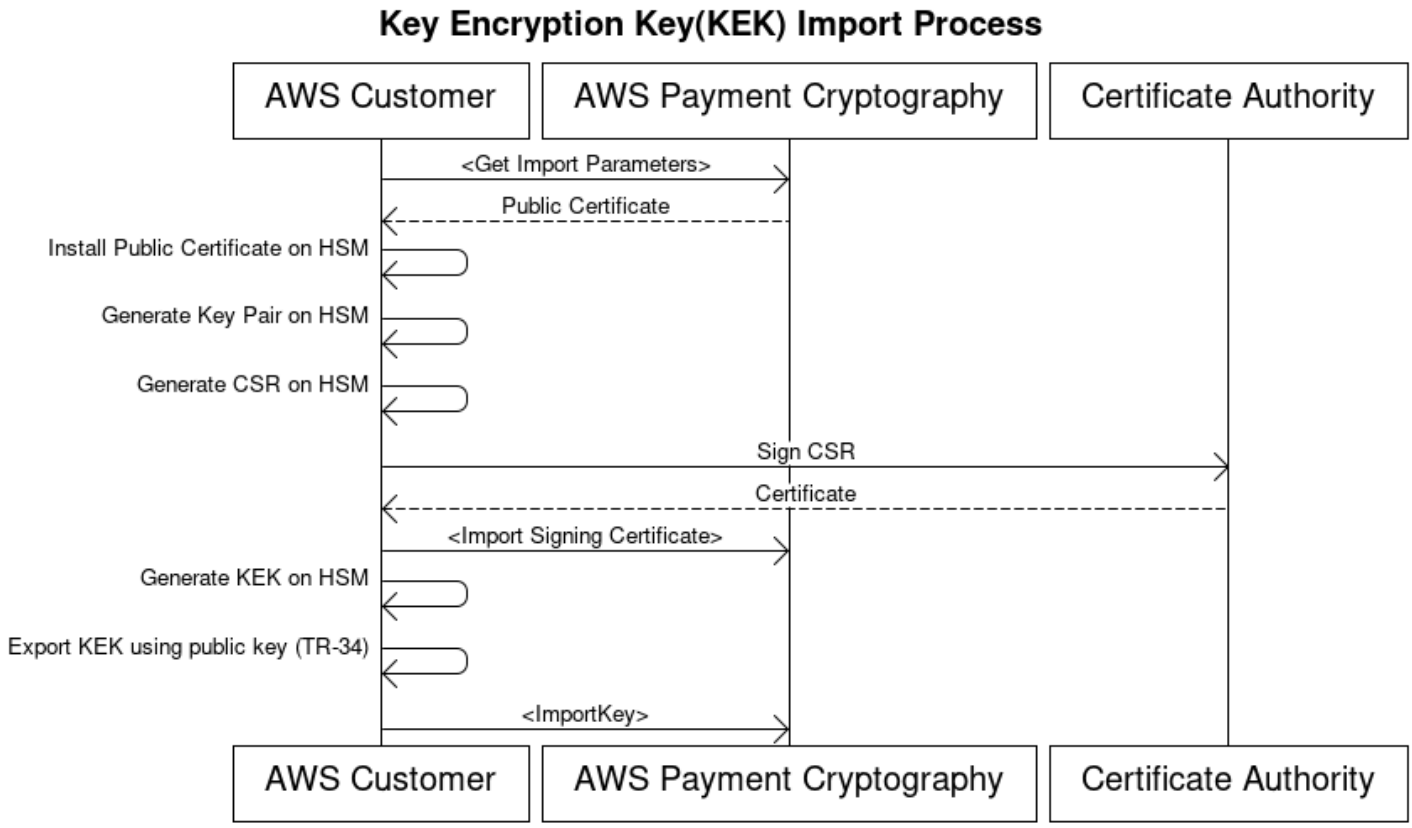
> **Important**
>
> Los ejemplos pueden requerir la versión más reciente de la AWS CLI V2. Antes de empezar, asegúrese de haber actualizado a la versión más reciente.

Temas

Importar claves simétricas

Temas

Importar claves utilizando técnicas asimétricas (TR-34)



Descripción general: TR-34 utiliza la criptografía asimétrica RSA para encriptar claves simétricas para su intercambio, así como para garantizar el origen de los datos (firma). Esto garantiza tanto la confidencialidad (encriptación) como la integridad (firma) de la clave envuelta.

Si desea importar sus propias claves, consulte el proyecto de ejemplo en Github. Para obtener instrucciones sobre cómo importar o exportar claves de otras plataformas, consulte la guía del usuario de dichas plataformas.

1. Llamar al comando de importación inicializar

   Llamar a `get-parameters-for-import` para inicializar el proceso de importación. Esta API generará un par de claves a efectos de importación de claves, firmará la clave y devolverá el certificado y la raíz del certificado. En última instancia, la clave que se vaya a exportar deberá cifrarse utilizando esta clave. En la terminología TR-34, esto se conoce como certificado KRD. Tenga en cuenta que estos certificados son de corta duración y sólo están pensados para este propósito.

2. Instalar el certificado público en el sistema fuente de claves

   Con muchos HSM, puede que necesite instalar, cargar o confiar el certificado público generado en el paso 1 para poder exportar claves utilizándolo.

3. Genere una clave pública y proporcione la raíz del certificado a AWS Payment Cryptography

   Para garantizar la integridad de la carga útil transmitida, ésta es firmada por la parte remitente (conocida como host de distribución de claves o KDH). La parte remitente querrá generar una clave pública para este fin y, a continuación, crear un certificado de clave pública (X509) que pueda devolverse a AWS Payment Cryptography. AWS Private CA es una opción para generar certificados, pero no hay restricciones en cuanto a la autoridad de certificación utilizada.

   Una vez que tenga el certificado, querrá cargar el certificado raíz en AWS Payment Cryptography mediante el `importKey` comando and KeyMaterialType of `ROOT_PUBLIC_KEY_CERTIFICATE` y KeyUsageType of`TR31_S0_ASYMMETRIC_KEY_FOR_DIGITAL_SIGNATURE`.

4. Exportar la clave desde el sistema fuente

   Muchos HSM y sistemas relacionados admiten la posibilidad de exportar claves mediante la norma TR-34. Deberá especificar la clave pública del paso 1 como el certificado KRD (cifrado) y la clave del paso 3 como el certificado KDH (firma). Para realizar la importación a AWS Payment Cryptography, deberá especificar que el formato sea el TR-34.2012, formato de dos pasadas, distinto del CMS, que también puede denominarse formato TR-34 Diebold.

5. Llamar a la clave de importación

   Como último paso, llamarás a la API ImportKey con un de. KeyMaterialType `TR34_KEY_BLOCK` El `certificate-authority-public-key-identifier` será el keyARN de la CA raíz importada en el paso 3, `key-material` será material clave envuelto del paso 4 y `signing-key-certificate` es el certificado hoja del paso 3. También deberá proporcionar el token de importación del paso 1.

6. Utilizar la clave importada para operaciones criptográficas o importaciones posteriores

   Si la importada KeyUsage era TR31_K0_KEY_ENCRYPTION_KEY, esta clave se puede usar para importaciones de claves posteriores mediante TR-31. Si el tipo de clave era cualquier otro (como TR31_D0_SYMMETRIC_DATA_ENCRYPTION_KEY), entonces la clave puede utilizarse directamente para operaciones criptográficas.

Importe las claves mediante técnicas asimétricas (RSA Unwrap)

Descripción general: La criptografía AWS de pagos admite el empaquetado y desempaquetado con RSA para el intercambio de claves cuando el TR-34 no es factible. Al igual que en el TR-34, esta técnica utiliza la criptografía asimétrica RSA para cifrar las claves simétricas que se van a intercambiar. Sin embargo, a diferencia del TR-34, este método no tiene la carga firmada por la parte que lo envía. Además, esta técnica de empaquetado RSA no mantiene la integridad de los metadatos clave durante la transferencia al no incluir los bloques clave.

> **Note**
>
> La envoltura RSA se puede utilizar para importar o exportar claves TDES y AES-128.

1. Llamar al comando de importación inicializar

   Llame `get-parameters-for-import` para inicializar el proceso de importación con un tipo de material clave llamado KEY_CRYPTOGRAM. WrappingKeyAlgorithm puede ser RSA_2048 al intercambiar claves TDES. Se pueden usar RSA_3072 o RSA_4096 al intercambiar claves TDES o AES-128. Esta API generará un par de claves para importarlas, firmará la clave con una raíz de certificado y devolverá tanto el certificado como la raíz del certificado. En última instancia, la clave que se vaya a exportar deberá cifrarse utilizando esta clave. Tenga en cuenta que estos certificados son de corta duración y sólo están pensados para este propósito.

```
$ aws payment-cryptography get-parameters-for-import --key-material-type
 KEY_CRYPTOGRAM --wrapping-key-algorithm RSA_4096
```

```
{
    "ImportToken": "import-token-bwxli6ocftypneu5",
    "ParametersValidUntilTimestamp": 1698245002.065,
    "WrappingKeyCertificateChain": "LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0....",
    "WrappingKeyCertificate": "LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0....",
    "WrappingKeyAlgorithm": "RSA_4096"
}
```

2. Instalar el certificado público en el sistema fuente de claves

   En el caso de muchos HSM, es posible que tengas que instalar, cargar o confiar en el certificado público (o en su raíz) generado en el paso 1 para poder exportar las claves que lo utilizan.

3. Exportar la clave desde el sistema fuente

   Muchos HSM y sistemas relacionados admiten la posibilidad de exportar claves mediante el empaquetado RSA. Deberá especificar la clave pública del paso 1 como el certificado (de cifrado) (). WrappingKeyCertificate Si necesitas la cadena de confianza, la encontrarás en el campo WrappingKeyCertificateChain de respuesta del paso #1. Al exportar la clave desde su HSM, querrá especificar que el formato sea RSA, modo de relleno = PKCS #1 v2.2 OAEP (con SHA 256 o SHA 512).

4. Llamar a la clave de importación

   Como último paso, llamarás a la API ImportKey con un de. KeyMaterialType `KeyMaterial` Necesitarás el token de importación del paso 1 y el `key-material` (el material clave empaquetado) del paso 3. Deberá proporcionar los parámetros clave (como el uso de claves), ya que RSA Wrap no utiliza bloques clave.

```
$ cat import-key-cryptogram.json
              {
                  "KeyMaterial": {
                      "KeyCryptogram": {
                          "Exportable": true,
                          "ImportToken": "import-token-bwxli6ocftypneu5",
                          "KeyAttributes": {
                              "KeyAlgorithm": "AES_128",
                              "KeyClass": "SYMMETRIC_KEY",
                              "KeyModesOfUse": {
                                  "Decrypt": true,
                                  "DeriveKey": false,
                                  "Encrypt": true,
                                  "Generate": false,
                                  "NoRestrictions": false,
                                  "Sign": false,
                                  "Unwrap": true,
                                  "Verify": false,
                                  "Wrap": true
                              },
                              "KeyUsage": "TR31_K0_KEY_ENCRYPTION_KEY"
                          },
```

```
                        "WrappedKeyCryptogram": "18874746731....",
                        "WrappingSpec": "RSA_OAEP_SHA_256"
                    }
                }
            }
```

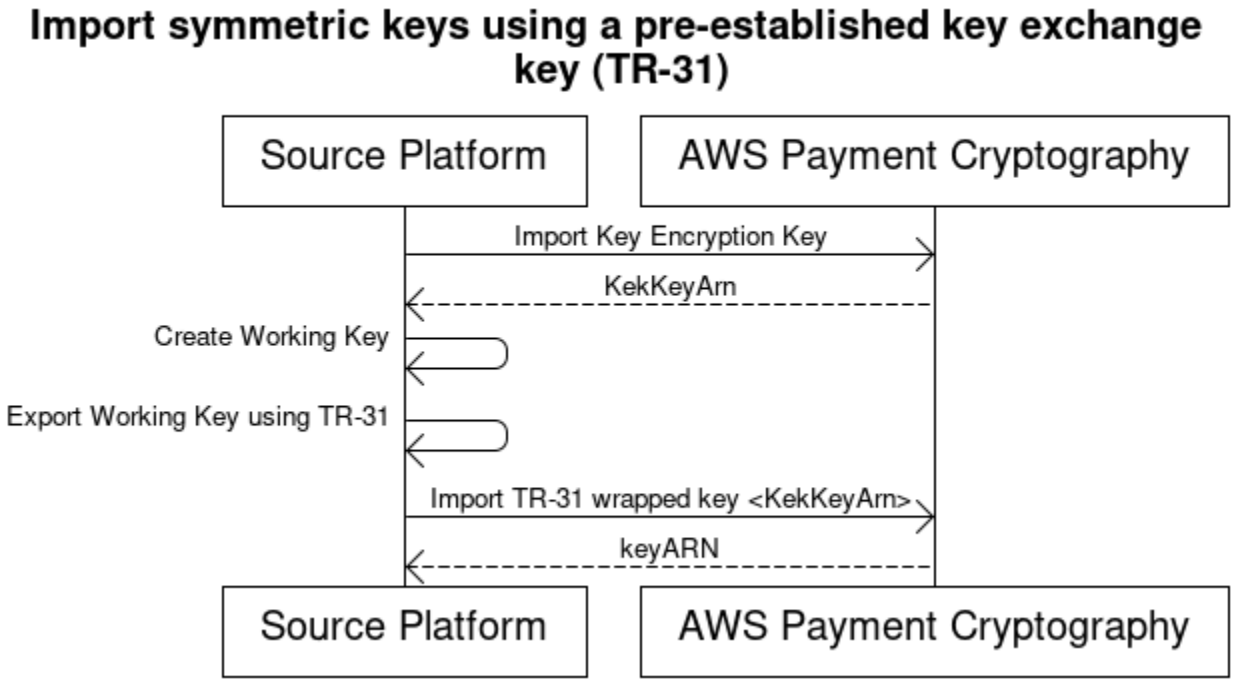
```
$ aws payment-cryptography import-key --cli-input-json file://import-key-
cryptogram.json
```

```
{
      "Key": {
          "KeyOrigin": "EXTERNAL",
          "Exportable": true,
          "KeyCheckValue": "DA1ACF",
          "UsageStartTimestamp": 1697643478.92,
          "Enabled": true,
          "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
kwapwa6qaifllw2h",
          "CreateTimestamp": 1697643478.92,
          "KeyState": "CREATE_COMPLETE",
          "KeyAttributes": {
              "KeyAlgorithm": "AES_128",
              "KeyModesOfUse": {
                  "Encrypt": true,
                  "Unwrap": true,
                  "Verify": false,
                  "DeriveKey": false,
                  "Decrypt": true,
                  "NoRestrictions": false,
                  "Sign": false,
                  "Wrap": true,
                  "Generate": false
              },
              "KeyUsage": "TR31_K0_KEY_ENCRYPTION_KEY",
              "KeyClass": "SYMMETRIC_KEY"
          },
          "KeyCheckValueAlgorithm": "CMAC"
      }
  }
```

5. Utilizar la clave importada para operaciones criptográficas o importaciones posteriores

   Si la importada KeyUsage era TR31_K0_KEY_ENCRYPTION_KEY, entonces esta clave se puede usar para importaciones de claves posteriores mediante TR-31. Si el tipo de clave era cualquier otro (como TR31_D0_SYMMETRIC_DATA_ENCRYPTION_KEY), entonces la clave puede utilizarse directamente para operaciones criptográficas.

Importar claves simétricas utilizando una clave de intercambio preestablecida (TR-31)



Cuando los socios intercambian varias claves (o para permitir la rotación de claves), lo habitual es intercambiar primero una clave de cifrado de clave (KEK) inicial mediante técnicas como componentes de clave en papel o, en el caso de la criptografía de AWS pagos, mediante el TR-34.

Una vez establecida una KEK, puede utilizarla para transportar las siguientes claves (incluidas otras KEK). AWS La criptografía de pagos admite este tipo de intercambio de claves mediante la norma ANSI TR-31, que es ampliamente utilizada y respaldada por los proveedores de HSM.

1. Importar clave de cifrado (KEK)

   Se supone que ya ha importado su KEK y tiene a su disposición el keyARN (o keyAlias).

2. Crear la clave en la plataforma de origen

   Si la clave aún no existe, cree la clave en la plataforma de origen. A la inversa, puede crear la clave en AWS Payment Cryptography y utilizar el comando `export` en su lugar.

3. Exportar la clave desde la plataforma de origen

   Al exportar, asegúrese de que especifica el formato de exportación como TR-31. La plataforma fuente también le preguntará por la clave a exportar y la clave de encriptación a utilizar.

4. AWS Importa a Payment Cryptography

   Al ejecutar el comando ImportKey, WrappingKeyIdentifier debe ser el keyArn (o alias) de la clave de cifrado y el resultado de la WrappedKeyBlock plataforma de origen.

   Example

```
$ aws payment-cryptography import-key \
            --key-material="Tr31KeyBlock={WrappingKeyIdentifier="arn:aws:payment-
cryptography:us-east-2:111122223333:key/ov6icy4ryas4zcza",\

 WrappedKeyBlock="D0112B0AX00E00002E0A3D58252CB67564853373D1EBCC1E23B2ADE7B15E967CC27B85D599
```

```
      {
        "Key": {
            "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
kwapwa6qaifllw2h",
            "KeyAttributes": {
                "KeyUsage": "TR31_D0_SYMMETRIC_DATA_ENCRYPTION_KEY",
                "KeyClass": "SYMMETRIC_KEY",
                "KeyAlgorithm": "AES_128",
                "KeyModesOfUse": {
                    "Encrypt": true,
                    "Decrypt": true,
                    "Wrap": true,
                    "Unwrap": true,
                    "Generate": false,
                    "Sign": false,
                    "Verify": false,
                    "DeriveKey": false,
                    "NoRestrictions": false
                }
```

```
            },
            "KeyCheckValue": "0A3674",
            "KeyCheckValueAlgorithm": "CMAC",
            "Enabled": true,
            "Exportable": true,
            "KeyState": "CREATE_COMPLETE",
            "KeyOrigin": "EXTERNAL",
            "CreateTimestamp": "2023-06-02T07:38:14.913000-07:00",
            "UsageStartTimestamp": "2023-06-02T07:38:14.857000-07:00"
        }
    }
```

Importar claves asimétricas (RSA)

Importar claves públicas RSA

AWS La criptografía de pagos permite importar claves RSA públicas en forma de certificados X.509. Para importar un certificado, deberá importar primero su certificado raíz. Todos los certificados deben estar vigentes en el momento de la importación. El certificado deberá estar en formato PEM y codificado en base64.

1. Importe el certificado raíz a la criptografía de pagos AWS

   Example

```
$ aws payment-cryptography import-key \
  --key-material='{"RootCertificatePublicKey":{"KeyAttributes":
{"KeyAlgorithm":"RSA_2048", \
  "KeyClass":"PUBLIC_KEY", "KeyModesOfUse":{"Verify":
 true},"KeyUsage":"TR31_S0_ASYMMETRIC_KEY_FOR_DIGITAL_SIGNATURE"}, \

 "PublicKeyCertificate":"LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSURKVENDQWcyZ0F3SUJBZ0lCWkR
```

```
  {
    "Key": {
        "CreateTimestamp": "2023-08-08T18:52:01.023000+00:00",
        "Enabled": true,
        "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
zabouwe3574jysdl",
        "KeyAttributes": {
```

```
            "KeyAlgorithm": "RSA_2048",
            "KeyClass": "PUBLIC_KEY",
            "KeyModesOfUse": {
                "Decrypt": false,
                "DeriveKey": false,
                "Encrypt": false,
                "Generate": false,
                "NoRestrictions": false,
                "Sign": false,
                "Unwrap": false,
                "Verify": true,
                "Wrap": false
            },
            "KeyUsage": "TR31_S0_ASYMMETRIC_KEY_FOR_DIGITAL_SIGNATURE"
        },
        "KeyOrigin": "EXTERNAL",
        "KeyState": "CREATE_COMPLETE",
        "UsageStartTimestamp": "2023-08-08T18:52:01.023000+00:00"
    }
}
```

2. Importe un certificado de clave pública a la criptografía AWS de pagos

   Ahora puede importar una clave pública. Existen dos opciones para importar claves públicas. `TR31_S0_ASYMMETRIC_KEY_FOR_DIGITAL_SIGNATURE` puede utilizarse si el propósito de la clave es verificar firmas (por ejemplo, al importar utilizando TR-34). `TR31_D1_ASYMMETRIC_KEY_FOR_DATA_ENCRYPTION` puede utilizarse cuando se encriptan datos destinados a ser utilizados con otro sistema.

   Example

```
$ aws payment-cryptography import-key \
    --key-material='{"TrustedCertificatePublicKey":
{"CertificateAuthorityPublicKeyIdentifier":"arn:aws:payment-cryptography:us-
east-2:111122223333:key/zabouwe3574jysdl", \
    "KeyAttributes":
{"KeyAlgorithm":"RSA_2048","KeyClass":"PUBLIC_KEY","KeyModesOfUse":
{"Verify":true},"KeyUsage":"TR31_S0_ASYMMETRIC_KEY_FOR_DIGITAL_SIGNATURE"},\
    "PublicKeyCertificate":"LS0tLS1CRUdJTiB..."}}'
```

```
{
```

```
  "Key": {
      "CreateTimestamp": "2023-08-08T18:55:46.815000+00:00",
      "Enabled": true,
      "KeyArn": "arn:aws:payment-cryptography:us-
east-2:111122223333:key/4kd6xud22e64wcbk",
      "KeyAttributes": {
          "KeyAlgorithm": "RSA_4096",
          "KeyClass": "PUBLIC_KEY",
          "KeyModesOfUse": {
              "Decrypt": false,
              "DeriveKey": false,
              "Encrypt": false,
              "Generate": false,
              "NoRestrictions": false,
              "Sign": false,
              "Unwrap": false,
              "Verify": true,
              "Wrap": false
          },
          "KeyUsage": "TR31_S0_ASYMMETRIC_KEY_FOR_DIGITAL_SIGNATURE"
      },
      "KeyOrigin": "EXTERNAL",
      "KeyState": "CREATE_COMPLETE",
      "UsageStartTimestamp": "2023-08-08T18:55:46.815000+00:00"
  }
}
```

Exportar claves

Temas

- Esportar claves simétricas
- Exportación de claves asimétricas (RSA)

Esportar claves simétricas

> **Important**
>
> Los ejemplos pueden requerir la versión más reciente de la AWS CLI V2. Antes de empezar, asegúrese de haber actualizado a la versión más reciente.

Temas

Exportar claves utilizando técnicas asimétricas (TR-34)

Descripción general: TR-34 utiliza la criptografía asimétrica RSA para encriptar claves simétricas para su intercambio, así como para garantizar el origen de los datos (firma). Esto garantiza tanto la confidencialidad (encriptación) como la integridad (firma) de la clave envuelta. Al exportar, la criptografía de AWS pagos se convierte en el host de distribución de claves (KDH) y el sistema de destino se convierte en el dispositivo receptor de claves (KRD).

1. Llamar al comando inicializar exportación

   Llamar a `get-parameters-for-export` para inicializar el proceso de exportación. Esta API generará un par de claves a efectos de exportación de claves, firmará la clave y devolverá el certificado y la raíz del certificado. En última instancia, la clave privada generada por este comando se utilizará para firmar la carga útil de la exportación. En la terminología TR-34, esto se conoce como certificado de firma KDH. Tenga en cuenta que estos certificados son de corta duración y sólo están pensados para este propósito. El parámetro ParametersValidUntilTimestamp especifica su duración.

   NOTA: Todos los certificados se devuelven en un formato codificado en base64

   Example

```
$ aws payment-cryptography get-parameters-for-export \
                    --signing-key-algorithm RSA_2048 --key-material-type
 TR34_KEY_BLOCK
```

```
{
               "SigningKeyCertificate":
 "LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUV2RENDQXFTZ0F3SUJBZ0lRZFAzSzNHNEFKT0I4WTNpTmUvYl
               "SigningKeyCertificateChain":
 "LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUY0VENDQThtZ0F3SUJBZ0lSQUtlN2piaHFKZjJPd3FGUWI5c3
               "SigningKeyAlgorithm": "RSA_2048",
```

```
            "ExportToken": "export-token-au7pvkbsq4mbup6i",
            "ParametersValidUntilTimestamp": "2023-06-13T15:40:24.036000-07:00"
}
```

2. Importe AWS el certificado de criptografía de pago al sistema receptor

   Importar la cadena de certificados proporcionada en el paso 1 a su sistema receptor según sea necesario.

3. Genere un key pair, cree un certificado público y proporcione la raíz del certificado a AWS Payment Cryptography

   Para garantizar la confidencialidad de la carga útil transmitida, ésta es encriptada por la parte remitente (conocida como host de distribución de claves o KDH). La parte receptora (normalmente su HSM o el HSM de sus socios) querrá generar una clave pública para este fin y, a continuación, crear un certificado de clave pública (x.509) que pueda devolverse a Payment Cryptography. AWS AWS Private CA es una opción para generar certificados, pero no hay restricciones en cuanto a la autoridad de certificación utilizada.

   Una vez que tenga el certificado, querrá cargar el certificado raíz en AWS Payment Cryptography mediante el `ImportKey` comando and KeyMaterialType of `ROOT_PUBLIC_KEY_CERTIFICATE` y KeyUsageType of`TR31_S0_ASYMMETRIC_KEY_FOR_DIGITAL_SIGNATURE`.

   El KeyUsageType de este certificado es TR31_S0_ASYMMETRIC_KEY_FOR_DIGITAL_SIGNATURE porque es la clave raíz y se utiliza para firmar el certificado hoja. Los certificados Leaf para la importación o exportación no se importan a Payment Cryptography, sino que se transmiten en línea. AWS

   > **Note**
   >
   > Si el certificado raíz se importó previamente, este paso puede omitirse.

4. Llamar a clave de exportación

   Como último paso, llamarás a la ExportKey API con un de. KeyMaterialType `TR34_KEY_BLOCK` `certificate-authority-public-key-identifier` será el keyARN de la CA raíz importada en el paso 3, `WrappingKeyCertificate` será el certificado hoja del paso 3 y `export-key-identifier` es el keyARN (o alias) a exportar. También deberá proporcionar el token de exportación del paso 1.

Exporte las claves mediante técnicas asimétricas (RSA Wrap)

Descripción general: La criptografía de AWS pagos admite el empaquetado y desempaquetado con RSA para el intercambio de claves cuando la contraparte no ofrece la opción TR-34. Al igual que en el TR-34, esta técnica utiliza la criptografía asimétrica RSA para cifrar las claves simétricas para su intercambio. Sin embargo, a diferencia del TR-34, este método no tiene la carga firmada por la parte que lo envía. Además, esta técnica de empaquetado RSA no incluye bloques clave que se utilizan para mantener la integridad de los metadatos clave durante el transporte.

> **Note**
>
> La envoltura RSA se puede utilizar para exportar claves TDES y AES-128.

1. Genere una clave RSA y un certificado en el sistema receptor

   Cree (o identifique) una clave RSA que se utilizará para recibir la clave empaquetada. AWS La criptografía de pagos espera claves en formato de certificado X.509. El certificado debe estar firmado por un certificado raíz que se importe (o se pueda importar) a AWS Payment Cryptography.
2. Instale un certificado público raíz en la criptografía AWS de pagos

```
$ aws payment-cryptography  import-key --key-material='{"RootCertificatePublicKey":
{"KeyAttributes":{"KeyAlgorithm":"RSA_4096","KeyClass":"PUBLIC_KEY","KeyModesOfUse":
{"Verify":
 true},"KeyUsage":"TR31_S0_ASYMMETRIC_KEY_FOR_DIGITAL_SIGNATURE"},"PublicKeyCertificate":"LS
```

```
    {
        "Key": {
        "CreateTimestamp": "2023-09-14T10:50:32.365000-07:00",
        "Enabled": true,
        "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
nsq2i3mbg6sn775f",
        "KeyAttributes": {
        "KeyAlgorithm": "RSA_4096",
        "KeyClass": "PUBLIC_KEY",
        "KeyModesOfUse": {
        "Decrypt": false,
        "DeriveKey": false,
        "Encrypt": false,
```

```
        "Generate": false,
        "NoRestrictions": false,
        "Sign": false,
        "Unwrap": false,
        "Verify": true,
        "Wrap": false
        },
        "KeyUsage": "TR31_S0_ASYMMETRIC_KEY_FOR_DIGITAL_SIGNATURE"
        },
        "KeyOrigin": "EXTERNAL",
        "KeyState": "CREATE_COMPLETE",
        "UsageStartTimestamp": "2023-09-14T10:50:32.365000-07:00“
        }
       }
```

3. Clave de exportación de llamadas

A continuación, debe indicar a AWS Payment Cryptography que exporte su clave utilizando su certificado principal. Especificará el ARN del certificado raíz importado anteriormente, el certificado hoja que se utilizará para la exportación y la clave simétrica que se exportará. El resultado será una versión empaquetada (cifrada) binaria y codificada en hexadecimal de su clave simétrica.

```
$ cat export-key.json
```

```
  {
      "ExportKeyIdentifier": "arn:aws:payment-cryptography:us-
east-2:111122223333:key/tqv5yij6wtxx64pi",
      "KeyMaterial": {
          "KeyCryptogram": {
              "CertificateAuthorityPublicKeyIdentifier": "arn:aws:payment-
cryptography:us-east-2:111122223333:key/zabouwe3574jysdl",
              "WrappingKeyCertificate": "LS0tLS1CRUdJTiBD...",
              "WrappingSpec": "RSA_OAEP_SHA_256"
          }
      }
  }
```

```
$ aws payment-cryptography export-key --cli-input-json file://export-key.json
```

```
{
      "WrappedKey": {
          "KeyMaterial":
 "18874746731E9E1C4562E4116D1C2477063FCB08454D757D81854AEAEE0A52B1F9D303FA29C02DC82AE7785353
          "WrappedKeyMaterialFormat": "KEY_CRYPTOGRAM"
      }
  }
```

4. Importe la clave al sistema receptor

   Muchos HSM y sistemas relacionados admiten la posibilidad de importar claves mediante RSA unwrap (incluida la criptografía de AWS pagos). Para ello, especifique la clave pública del paso 1 como certificado (de cifrado) y el formato debe especificarse como RSA, modo de relleno = PKCS #1 v2.2 OAEP (con SHA 256). La terminología exacta puede variar según el HSM.

   > **Note**
   >
   > AWS La criptografía de pagos genera la clave empaquetada en HexBinary. Es posible que necesite convertir el formato antes de importarlo si su sistema requiere una representación binaria diferente, como base64.

Exportar claves simétricas utilizando una clave de intercambio preestablecida (TR-31)

Cuando los socios intercambian varias claves (o para permitir la rotación de claves), lo habitual es intercambiar primero una clave de cifrado de clave (KEK) inicial mediante técnicas como componentes de clave en papel o, en el caso de la criptografía de AWS pagos, mediante el TR-34. Una vez establecida una KEK, puede utilizarla para transportar las siguientes claves (incluidas otras KEK). AWS La criptografía de pagos admite este tipo de intercambio de claves mediante la norma ANSI TR-31, que es ampliamente utilizada y respaldada por los proveedores de HSM.

1. Intercambiar la clave de cifrado (KEK)

   Se supone que ya ha intercambiado su KEK y dispone del keyARN (o keyAlias).

2. Cree una clave sobre criptografía de pagos AWS

   Si la clave no existe todavía, créela. Por el contrario, puede crear la clave en el otro sistema y utilizar en su lugar el comando importar.

3. Exporte la clave de AWS Payment Cryptography

   Al exportar, el formato será TR-31. Al llamar a la API, especificará la clave a exportar y la clave de envoltura a utilizar.

```
$ aws payment-cryptography export-key --key-material='{"Tr31KeyBlock":
 {"WrappingKeyIdentifier": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
ov6icy4ryas4zcza"}}' --export-key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/5rplquuwozodpwsp
```

```
{
             "WrappedKey": {
                 "KeyCheckValue": "73C263",
                 "KeyCheckValueAlgorithm": "ANSI_X9_24",
                 "KeyMaterial":
 "D0144K0AB00E0000A24D3ACF3005F30A6E31D533E07F2E1B17A2A003B338B1E79E5B3AD4FBF7850FACF9A37844
                 "WrappedKeyMaterialFormat": "TR31_KEY_BLOCK"
             }
     }
```

4. Importar a su sistema

   Usted o su socio utilizarán la implementación de la clave de importación en su sistema para importar la clave.

Exporte las claves iniciales de DUKPT (IPEK/IK)

Cuando se utiliza DUKPT, se puede generar una única clave de derivación base (BDK) para una flota de terminales. Sin embargo, los terminales nunca tienen acceso a esa BDK original, sino que a cada uno se le inyecta una clave de terminal inicial única, conocida como IPEK o clave inicial (IK). Cada IPEK es una clave derivada del BDK y se pretende que sea única por terminal, pero se deriva del BDK original. Los datos de derivación para este cálculo se conocen como número de serie clave (KSN). Según X9.24, en el caso del TDES, el KSN de 10 bytes normalmente consta de 24 bits para el identificador del conjunto de claves, 19 bits para el identificador del terminal y 21 bits para el contador de transacciones. En el caso del AES, el KSN de 12 bytes normalmente consta de 32 bits para el ID del BDK, 32 bits para el identificador de derivación (ID) y 32 bits para el contador de transacciones.

AWS La criptografía de pagos proporciona un mecanismo para generar y exportar estas claves iniciales. Una vez generadas, estas claves se pueden exportar mediante los métodos de

empaquetado TR-31, TR-34 y RSA. Las claves IPEK no se conservan y no se pueden utilizar para operaciones posteriores de criptografía de pagos AWS

AWS La criptografía de pagos no impone la división entre las dos primeras partes de la KSN. Si desea almacenar el identificador de derivación junto con el BDK, puede utilizar la función de AWS etiquetas para este fin.

> **Note**
>
> La parte del contador del KSN (32 bits para el AES DUKPT) no se utiliza para la derivación de IPEK/IK. Por lo tanto, una entrada de 12345678901234560001 y 12345678901234569999 generará el mismo IPEK.

```
$ aws payment-cryptography export-key --key-material='{"Tr31KeyBlock":
 {"WrappingKeyIdentifier": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
ov6icy4ryas4zcza"}}' --export-key-identifier arn:aws:payment-
cryptography:us-east-2:111122223333:key/tqv5yij6wtxx64pi --export-attributes
 'ExportDukptInitialKey={KeySerialNumber=12345678901234560001}'
```

```
{
    "WrappedKey": {
        "KeyCheckValue": "73C263",
        "KeyCheckValueAlgorithm": "ANSI_X9_24",
        "KeyMaterial":
 "B0096B1TX00S000038A8A06588B9011F0D5EEF1CCAECFA6962647A89195B7A98BDA65DDE7C57FEA507559AF2A5D60
        "WrappedKeyMaterialFormat": "TR31_KEY_BLOCK"
    }
}
```

Exportación de claves asimétricas (RSA)

Llamar a `get-public-key-certificate` para exportar una clave pública en forma de certificado. Esta API exportará el certificado así como su certificado raíz codificado en formato base64.

NOTA: Esta API no es idempotente; las llamadas posteriores pueden dar como resultado certificados diferentes aunque la clave subyacente sea la misma.

Example

```
$ aws payment-cryptography get-public-key-certificate \
```

```
        –key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/5dza7xqd6soanjtb
```

```
{
        "KeyCertificate": "LS0tLS1CRUdJTi...",
        "KeyCertificateChain": "LS0tLS1CRUdJT...“
}
```

Uso de alias

Un alias es un nombre descriptivo para una clave de criptografía AWS de pagos. Por ejemplo, un alias le permite referirse a una clave como `alias/test-key` en lugar de `arn:aws:payment-cryptography:us-east-2:111122223333:key/kwapwa6qaifllw2h`.

Puede utilizar un alias para identificar una clave en la mayoría de las operaciones de gestión de claves (plano de control), y en operaciones criptográficas (plano de datos).

También puedes permitir y denegar el acceso a la clave de criptografía de AWS pagos en función de sus alias sin editar las políticas ni gestionar las subvenciones. Esta característica forma parte de la compatibilidad del servicio con el control de acceso basado en atributos (ABAC).

Gran parte de la potencia de los alias proviene de su capacidad de cambiar la clave asociada a un alias en cualquier momento. Los alias pueden hacer que su código sea más fácil de escribir y mantener. Por ejemplo, supongamos que utiliza un alias para hacer referencia a una clave de criptografía de AWS pago concreta y desea cambiarla AWS . En ese caso, simplemente asocie el alias con una clave diferente. No necesita cambiar el código ni la configuración de su aplicación.

Los alias también facilitan la reutilización del mismo código en diferentes Regiones de AWS. Cree alias con el mismo nombre en varias regiones y asocie cada alias a una clave de criptografía de AWS pagos en su región. Cuando el código se ejecuta en cada región, el alias hace referencia a la clave de criptografía de AWS pagos asociada en esa región.

Puedes crear un alias para una clave de criptografía de AWS pagos mediante la `CreateAlias` API.

La API AWS de criptografía de pagos proporciona un control total de los alias de cada cuenta y región. La API incluye operaciones para crear un alias (CreateAlias), ver los nombres de los alias y el keyARN vinculado (alias de lista), cambiar la clave de criptografía de AWS pagos asociada a un alias (update-alias) y eliminar un alias (delete-alias).

Temas

Acerca de los alias

AWS Descubre cómo funcionan los alias en la criptografía de pagos.

Un alias es un recurso independiente AWS

Un alias no es propiedad de una clave de criptografía de AWS pagos. Las acciones que realice en el alias no afectan a su clave asociada. Puede crear un alias para una clave de criptografía de AWS pago y, a continuación, actualizar el alias para que se asocie a una clave de criptografía de AWS pago diferente. Incluso puedes eliminar el alias sin que ello afecte a la clave de criptografía de AWS pagos asociada. Si elimina una clave de AWS Payment Cryptography, todos los alias asociados a esa clave quedarán sin asignar.

Si especificas un alias como recurso en una política de IAM, la política se refiere al alias y no a la clave de criptografía de AWS pagos asociada.

Cada alias tiene un nombre coloquial

Cuando cree un alias, especifique el nombre del alias precedido por `alias/`. Por ejemplo, `alias/test_1234`

Cada alias está asociado a una clave de criptografía AWS de pagos a la vez

El alias y su clave AWS de criptografía de pagos deben estar en la misma cuenta y región.

Una clave de criptografía de AWS pago se puede asociar a más de un alias al mismo tiempo, pero cada alias solo se puede asignar a una única clave

Por ejemplo, esta salida `list-aliases` muestra que el alias `alias/sampleAlias1` está asociado con exactamente una clave de AWS Payment Cryptography de destino, que está representada por la propiedad de `KeyArn`.

```
$ aws payment-cryptography list-aliases
```

```
  {
    "Aliases": [
        {
            "AliasName": "alias/sampleAlias1",
            "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
kwapwa6qaifllw2h"
        }
    ]
}
```

Se pueden asociar varios alias a la misma clave de criptografía de pagos AWS

Por ejemplo, puede asociar los alias `alias/sampleAlias1;` y `alias/sampleAlias2` con la misma clave.

```
$ aws payment-cryptography list-aliases
```

```
{
        "Aliases": [
            {
                "AliasName": "alias/sampleAlias1",
                "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
kwapwa6qaifllw2h"
            },
            {
                "AliasName": "alias/sampleAlias2",
                "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
kwapwa6qaifllw2h"
            }
        ]
    }
```

El alias debe ser único para una cuenta y región determinadas.

Por ejemplo, solo puede tener un alias `alias/sampleAlias1` en cada cuenta y región. Los alias distinguen entre mayúsculas y minúsculas, pero le recomendamos que no utilice alias que sólo difieran en las mayúsculas, ya que pueden dar lugar a errores. No puede cambiar un nombre de alias. Sin embargo, puede eliminar el alias y crear un nuevo alias con el nombre deseado.

Puede crear alias con el mismo nombre en diferentes regiones

Por ejemplo, puede tener un alias `alias/sampleAlias2` en el Este de EE. UU. (Norte de Virginia) y un alias `alias/sampleAlias2` en el Oeste de EE. UU. (Oregón). Cada alias estaría asociado a una clave de criptografía AWS de pagos en su región. Si su código se refiere a un nombre de alias como `alias/finance-key`, puede ejecutarlo en varias regiones. En cada región, utiliza un alias/sampleAlias2 diferente. Para obtener más detalles, consulte Usar alias en las aplicaciones.

Puede cambiar la clave de criptografía de AWS pago asociada a un alias

Puede utilizar la `UpdateAlias` operación para asociar un alias a una clave de criptografía AWS de pagos diferente. Por ejemplo, si el `alias/sampleAlias2` alias está asociado a la clave de criptografía de `arn:aws:payment-cryptography:us-east-2:111122223333:key/kwapwa6qaifllw2h` AWS pagos, puede actualizarlo para que esté asociado a la `arn:aws:payment-cryptography:us-east-2:111122223333:key/tqv5yij6wtxx64pi` clave.

> **Warning**
>
> AWS La criptografía de pagos no valida que la clave antigua y la nueva tengan los mismos atributos, como el uso de claves. La actualización con un tipo de clave diferente puede dar lugar a problemas en su aplicación.

Algunas claves no tienen alias

Un alias es una característica opcional y no todas las claves tendrán alias a menos que elija operar su entorno de esta manera. Las claves pueden asociarse con alias utilizando el comando `create-alias`. Además, puede usar la operación update-alias para cambiar la clave de AWS Payment Cryptography asociada a un alias y la operación delete-alias para eliminar un alias. Como resultado, es posible que algunas claves de criptografía de AWS pagos tengan varios alias y que otras no tengan ninguno.

Asignación de una clave a un alias

Puede asignar una clave (representada por un ARN) a uno o más alias utilizando el comando `create-alias`. Este comando no es idempotente; para actualizar un alias, utilice el comando update-alias.

```
$ aws payment-cryptography create-alias --alias-name alias/sampleAlias1 \
```

```
            --key-arn arn:aws:payment-cryptography:us-east-2:111122223333:key/
kwapwa6qaifllw2h
```

```
{
    "Alias": {
        "AliasName": "alias/alias/sampleAlias1",
        "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
kwapwa6qaifllw2h"
    }
}
```

Usar alias en las aplicaciones

Puedes usar un alias para representar una clave de criptografía AWS de pagos en el código de tu aplicación. El `key-identifier` parámetro en las operaciones de datos de criptografía de AWS pagos, así como en otras operaciones como List Keys, acepta un alias o un alias ARN.

```
$ aws payment-cryptography-data generate-card-validation-data --key-identifier alias/
BIN_123456_CVK --primary-account-number=171234567890123 --generation-attributes
 CardVerificationValue2={CardExpiryDate=0123}
```

Cuando utilice un ARN de alias, recuerde que la asignación de alias a una clave de criptografía de AWS pago se define en la cuenta propietaria de la clave de criptografía de AWS pago y puede diferir en cada región.

Uno de los usos más potentes de los alias es en aplicaciones que se ejecutan en múltiples Regiones de AWS.

Puede crear una versión diferente de la aplicación en cada región o utilizar un diccionario, una configuración o una sentencia de cambio para seleccionar la clave criptográfica de AWS pagos adecuada para cada región. Pero puede ser más fácil crear un alias con el mismo nombre de alias en cada región. El nombre del alias distingue entre mayúsculas y minúsculas.

API relacionadas

Etiquetas

Las etiquetas son pares de claves y valores que actúan como metadatos para organizar las claves de criptografía AWS de pagos. Pueden utilizarse para identificar claves de forma flexible o agrupar una o varias claves.

Obtener claves

Una clave AWS de criptografía de pago representa una sola unidad de material criptográfico y solo se puede utilizar para las operaciones criptográficas de este servicio. La GetKeys API toma KeyIdentifier como entrada y devuelve los atributos inmutables y mutables de la clave, pero no contiene ningún material criptográfico.

Example

```
$ aws payment-cryptography get-key --key-identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/kwapwa6qaifllw2h
```

```
{
  "Key": {
      "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
kwapwa6qaifllw2h",
      "KeyAttributes": {
          "KeyUsage": "TR31_D0_SYMMETRIC_DATA_ENCRYPTION_KEY",
          "KeyClass": "SYMMETRIC_KEY",
          "KeyAlgorithm": "AES_128",
          "KeyModesOfUse": {
              "Encrypt": true,
              "Decrypt": true,
              "Wrap": true,
              "Unwrap": true,
              "Generate": false,
              "Sign": false,
              "Verify": false,
              "DeriveKey": false,
```

```
            "NoRestrictions": false
        }
    },
    "KeyCheckValue": "0A3674",
    "KeyCheckValueAlgorithm": "CMAC",
    "Enabled": true,
    "Exportable": true,
    "KeyState": "CREATE_COMPLETE",
    "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
    "CreateTimestamp": "2023-06-02T07:38:14.913000-07:00",
    "UsageStartTimestamp": "2023-06-02T07:38:14.857000-07:00"
  }
}
```

Obtener la clave pública o certificado asociado a un par de claves

Obtener clave pública o certificado devuelve la clave pública indicada por el `KeyArn`. Puede ser la parte de clave pública de un par de claves generado en la criptografía de AWS pago o una clave pública importada anteriormente. El caso de uso más común es proporcionar la clave pública a un servicio externo que cifrará datos. Luego, esos datos se pueden pasar a una aplicación que utilice la criptografía de AWS pago y se pueden descifrar con la clave privada protegida en la criptografía de pago. AWS

El servicio devuelve las claves públicas como un certificado público. El resultado de la API contiene la CA y el certificado de clave pública. Ambos elementos de datos están codificados en base64.

> **Note**
>
> El certificado público devuelto está pensado para ser de corta duración y no pretende ser idempotente. Es posible que reciba un certificado diferente en cada llamada a la API, aunque la clave pública en sí no cambie.

Example

```
$ aws payment-cryptography get-public-key-certificate  --key-identifier
 arn:aws:payment-cryptography:us-east-2:111122223333:key/nsq2i3mbg6sn775f
```

```
{
  "KeyCertificate":
 "LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUV2VENDQXFXZ0F3SUJBZ0lSQUo1OWd2VkpDd3dlYldMNldYZEpYY
  "KeyCertificateChain":
 "LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUY0VENDQThtZ0F3SUJBZ0lSQUtlN2piaHFKZjJPd3FGUWI5c3VuC
}
```

Etiquetado de claves

En la criptografía de AWS pagos, puede añadir etiquetas a una clave de criptografía de AWS pagos al crear una clave, y etiquetar o desetiquetar las claves existentes, a menos que estén pendientes de ser eliminadas. Las etiquetas son opcionales, pero pueden ser muy útiles.

Para obtener información general sobre las etiquetas, incluidas las prácticas recomendadas, las estrategias de etiquetado y el formato y la sintaxis de las etiquetas, consulte los recursos de etiquetado AWS en la. Referencia general de Amazon Web Services

Temas

- Acerca de las etiquetas en la criptografía de pagos AWS
- Visualización de etiquetas clave en la consola
- Administración de etiquetas de clave con operaciones de la API
- Control del acceso a las etiquetas
- Uso de etiquetas para controlar el acceso a las claves

Acerca de las etiquetas en la criptografía de pagos AWS

Una etiqueta es una etiqueta de metadatos opcional que puede asignar (o AWS puede asignar) a un AWS recurso. Cada etiqueta consta de una clave de etiqueta y a valor de etiqueta, que distinguen entre mayúsculas y minúsculas. El valor de la etiqueta puede ser una cadena vacía (nula). Cada etiqueta de un recurso debe tener una clave de etiqueta diferente, pero puedes añadir la misma

etiqueta a varios AWS recursos. Cada recurso puede tener un máximo de 50 etiquetas creadas por el usuario.

No incluya información confidencial en la clave ni en el valor de la etiqueta. Muchas personas pueden acceder a las etiquetas Servicios de AWS, incluida la facturación.

En la criptografía de AWS pagos, puede añadir etiquetas a una clave al crearla y etiquetar o desetiquetar las claves existentes, a menos que estén pendientes de ser eliminadas. No puede etiquetar alias. Las etiquetas son opcionales, pero pueden ser muy útiles.

Por ejemplo, puede añadir una `"Project"="Alpha"` etiqueta a todas las claves de criptografía de AWS pagos y a los buckets de Amazon S3 que utilice para el proyecto Alpha. Otro ejemplo es añadir una etiqueta `"BIN"="20130622"` a todas las claves asociadas a un número de identificación bancaria (BIN) específico.

```
  [
  {
    "Key": "Project",
    "Value": "Alpha"
  },
  {
    "Key": "BIN",
    "Value": "20130622"
  }
]
```

Para obtener información general sobre las etiquetas, incluidos el formato y la sintaxis, consulte AWS los recursos de etiquetado en. Referencia general de Amazon Web Services

Las etiquetas le ayudan a hacer lo siguiente:

- Identifique y organice sus AWS recursos. Muchos AWS servicios admiten el etiquetado, por lo que puede asignar la misma etiqueta a los recursos de diferentes servicios para indicar que los recursos están relacionados. Por ejemplo, puede asignar la misma etiqueta a una clave de criptografía de AWS pagos y a un volumen o secreto de Amazon Elastic Block Store (Amazon EBS). AWS Secrets Manager También puede utilizar etiquetas para identificar claves para la automatización.
- Realice un seguimiento de sus costes. AWS Cuando agrega etiquetas a sus AWS recursos, AWS genera un informe de asignación de costos con el uso y los costos agregados por etiquetas. Puede

usar esta función para realizar un seguimiento de los costos de criptografía de AWS pagos de un proyecto, aplicación o centro de costos.

Para obtener más información sobre el uso de etiquetas para la asignación de costos, consulte Uso de etiquetas de asignación de costos en la Guía del usuario de AWS Billing . Para obtener información sobre las reglas que se aplican a las claves y los valores de las etiquetas, consulte Restricciones de las etiquetas definidas por el usuario en la Guía del usuario AWS Billing .

- Controle el acceso a sus AWS recursos. Permitir y denegar el acceso a las claves en función de sus etiquetas forma parte del apoyo de la criptografía de AWS pagos al control de acceso basado en atributos (ABAC). Para obtener más información sobre el control de acceso a AWS Payment Cryptography basado en etiquetas, consulte Autorización basada en las etiquetas de AWS Payment Cryptography. Para obtener más información general sobre el uso de etiquetas para controlar el acceso a AWS los recursos, consulte Control del acceso a los AWS recursos mediante etiquetas de recursos en la Guía del usuario de IAM.

AWS La criptografía de pagos escribe una entrada en su AWS CloudTrail registro cuando utiliza las operaciones TagResource UntagResource, o ListTagsForResource .

Visualización de etiquetas clave en la consola

Para ver las etiquetas en la consola, necesita permiso de etiquetado en la clave desde una política de IAM que incluya la clave. Necesita estos permisos además de los permisos para ver las claves en la consola.

Administración de etiquetas de clave con operaciones de la API

Puede utilizar la API de AWS Payment Cryptography para agregar, eliminar y enumerar etiquetas para las claves que administre. En estos ejemplos, se utiliza la AWS Command Line Interface (AWS CLI), pero puede usar cualquier lenguaje de programación admitido. No puedes etiquetar Claves administradas por AWS.

Para agregar, editar, ver y eliminar etiquetas de una clave, debe tener los permisos necesarios. Para obtener más detalles, consulte Control del acceso a las etiquetas.

Temas

- CreateKey: Añadir etiquetas a una clave nueva
- TagResource: Añada o cambie las etiquetas de una clave
- ListResourceTags: Obtenga las etiquetas de una clave

- UntagResource: Eliminar etiquetas de una clave

CreateKey: Añadir etiquetas a una clave nueva

Puede añadir etiquetas cuando cree una llave. Para especificar las etiquetas, utilice el `Tags` parámetro de la CreateKeyoperación.

Para agregar etiquetas al crear una clave, la persona que llama debe tener el permiso `payment-cryptography:TagResource` en una política de IAM. Como mínimo, el permiso debe cubrir todas las claves de la cuenta y la región. Para obtener más detalles, consulte Control del acceso a las etiquetas.

El valor del parámetro `Tags` de `CreateKey` es una colección de pares de claves y valores de etiqueta que distinguen mayúsculas y minúsculas. Cada etiqueta de una clave debe tener un nombre de etiqueta diferente. El valor de etiqueta puede ser una cadena vacía o nula.

Por ejemplo, el siguiente AWS CLI comando crea una clave de cifrado simétrica con una `Project:Alpha` etiqueta. Cuando especifique más de un par de clave-valor, utilice un espacio para separar cada par.

```
$ aws payment-cryptography create-key --exportable --key-attributes
 KeyAlgorithm=TDES_2KEY, \
       KeyUsage=TR31_C0_CARD_VERIFICATION_KEY,KeyClass=SYMMETRIC_KEY, \
       KeyModesOfUse='{Generate=true,Verify=true}' \
       --tags '[{"Key":"Project","Value":"Alpha"},{"Key":"BIN","Value":"123456"}]'
```

Cuando este comando se ejecuta correctamente, devuelve un objeto `Key` con información sobre la nueva clave. Sin embargo, `Key` no incluye etiquetas. Para obtener las etiquetas, utilice la ListResourceTagsoperación.

TagResource: Añada o cambie las etiquetas de una clave

La TagResourceoperación añade una o más etiquetas a una clave. No puede usar esta operación para agregar o editar etiquetas en una Cuenta de AWS diferente.

Para agregar una etiqueta, especifique una clave de etiqueta nueva y un valor de la etiqueta. Para editar una etiqueta, especifique una clave de etiqueta existente y un nuevo valor de etiqueta. Cada etiqueta de una clave debe tener una clave de etiqueta distinta. El valor de etiqueta puede ser una cadena vacía o nula.

Por ejemplo, el siguiente comando agrega las etiquetas **`UseCase`** y **`BIN`** a una clave de ejemplo.

```
$ aws payment-cryptography tag-resource --resource-arn arn:aws:payment-
cryptography:us-east-2:111122223333:key/kwapwa6qaifllw2h --tags
 '[{"Key":"UseCase","Value":"Acquiring"},{"Key":"BIN","Value":"123456"}]'
```

Si este comando se realiza correctamente, no devuelve ningún resultado. Para ver las etiquetas de una tecla, utilice la ListResourceTagsoperación.

También pueden utilizar TagResource para cambiar los valores de una etiqueta existente. Para sustituir los valores de etiqueta, especifique la misma clave de etiqueta con distintos valores. Las etiquetas no listadas en un comando de modificación no se cambian ni se eliminan.

Por ejemplo, este comando cambia el valor de la etiqueta `Project` de `Alpha` a `Noe`.

El comando devolverá http/200 sin contenido. Para ver los cambios, utilice `ListTagsForResource`

```
$ aws payment-cryptography tag-resource --resource-arn arn:aws:payment-cryptography:us-
east-2:111122223333:key/kwapwa6qaifllw2h \
        --tags '[{"Key":"Project","Value":"Noe"}]'
```

ListResourceTags: Obtenga las etiquetas de una clave

La ListResourceTagsoperación obtiene las etiquetas de una clave. El parámetro `ResourceArn` (keyArn o keyAlias) es obligatorio. No puede usar esta operación para ver las etiquetas de claves en una Cuenta de AWS diferente.

Por ejemplo, el comando siguiente obtiene las etiquetas para una clave de ejemplo.

```
$ aws payment-cryptography list-tags-for-resource --resource-arn arn:aws:payment-
cryptography:us-east-2:111122223333:key/kwapwa6qaifllw2h

  {
    "Tags": [
        {
            "Key": "BIN",
            "Value": "20151120"
        },
        {
            "Key": "Project",
            "Value": "Production"
        }
```

```
    ]
}
```

UntagResource: Eliminar etiquetas de una clave

La UntagResourceoperación elimina las etiquetas de una clave. Para identificar las etiquetas que desea eliminar, especifique las claves de etiqueta. No puede usar esta operación para eliminar etiquetas de claves una Cuenta de AWS diferente.

Cuando tiene éxito, la operación `UntagResource` no devuelve ningún resultado. Además, si la clave de etiqueta especificada no se encuentra en la clave, no arroja una excepción ni devuelve una respuesta. Para confirmar que la operación ha funcionado, utilice la ListResourceTagsoperación.

Por ejemplo, este comando elimina la etiqueta **`Purpose`** y todos sus valores de la clave especificada.

```
$ aws payment-cryptography untag-resource \
        --resource-arn arn:aws:payment-cryptography:us-east-2:111122223333:key/
kwapwa6qaifllw2h --tag-keys Project
```

Control del acceso a las etiquetas

Para agregar, ver y eliminar etiquetas mediante el uso de la API, las entidades principales necesitan permisos de etiquetado en la política de IAM.

También puede limitar estos permisos mediante el uso de claves de condición AWS globales para las etiquetas. En la criptografía de AWS pagos, estas condiciones pueden controlar el acceso a las operaciones de etiquetado, como TagResourcey. UntagResource

Para obtener más información y políticas de ejemplo, consulte Control del acceso en función de las claves de etiquetaen la Guía del usuario de IAM.

Los permisos para crear y administrar etiquetas funcionan de la siguiente manera.

criptografía de pagos: TagResource

Permite a las entidades principales agregar o editar etiquetas. Para agregar etiquetas al crear una clave, la entidad principal debe tener permiso en una política de IAM que no esté restringida a determinadas claves.

criptografía de pago: ListTagsForResource

Permite a las entidades principales ver etiquetas en claves.

criptografía de pago: UntagResource

Permite a las entidades principales eliminar etiquetas de las claves.

Permisos de etiquetas en políticas

Puede proporcionar permisos de etiquetas en una política de claves o una política de IAM. Por ejemplo, la siguiente política de claves de ejemplo ofrece permiso de etiquetar a los usuarios seleccionados en la clave. Da permiso a todos los usuarios que pueden asumir los roles de administrador o desarrollador de ejemplo para ver etiquetas.

```
{
  "Version": "2012-10-17",
  "Id": "example-key-policy",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {"AWS": "arn:aws:iam::111122223333:root"},
      "Action": "payment-cryptography:*",
      "Resource": "*"
    },
    {
      "Sid": "Allow all tagging permissions",
      "Effect": "Allow",
      "Principal": {"AWS": [
        "arn:aws:iam::111122223333:user/LeadAdmin",
        "arn:aws:iam::111122223333:user/SupportLead"
      ]},
      "Action": [
          "payment-cryptography:TagResource",
          "payment-cryptography:ListTagsForResource",
          "payment-cryptography:UntagResource"
      ],
      "Resource": "*"
    },
    {
      "Sid": "Allow roles to view tags",
      "Effect": "Allow",
      "Principal": {"AWS": [
        "arn:aws:iam::111122223333:role/Administrator",
        "arn:aws:iam::111122223333:role/Developer"
```

```
      ]},
      "Action": "payment-cryptography:ListResourceTags",
      "Resource": "*"
    }
  ]
}
```

Para conceder permiso de etiquetado de entidades principales en varias claves, puede usar una política de IAM. Para que esta política sea efectiva, la política de claves de cada clave debe permitir a la cuenta utilizar políticas de IAM para controlar el acceso a clave.

Por ejemplo, la siguiente política de IAM permite a las entidades principales crear claves. También les permite crear y administrar etiquetas en todas las claves de la cuenta especificada. Esta combinación permite a los directores utilizar el parámetro tags de la CreateKeyoperación para añadir etiquetas a una clave mientras la crean.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "IAMPolicyCreateKeys",
      "Effect": "Allow",
      "Action": "payment-cryptography:CreateKey",
      "Resource": "*"
    },
    {
      "Sid": "IAMPolicyTags",
      "Effect": "Allow",
      "Action": [
        "payment-cryptography:TagResource",
        "payment-cryptography:UntagResource",
        "payment-cryptography:ListTagsForResource"
      ],
      "Resource": "arn:aws:payment-cryptography:*:111122223333:key/*"
    }
  ]
}
```

Limitar los permisos de etiqueta

Puede limitar los permisos de etiquetado mediante condiciones de política. Las siguientes condiciones de política se pueden aplicar a los permisos `payment-cryptography:TagResource`

y `payment-cryptography:UntagResource`. Por ejemplo, puede utilizar la condición `aws:RequestTag/`*`tag-key`* para permitir que una entidad principal agregue solo etiquetas particulares, o impedir que una entidad principal agregue etiquetas con claves de etiqueta concretas.

- leyes: RequestTag
- aws:ResourceTag/tag-key (solo políticas de IAM)
- AWS: TagKeys

Como práctica recomendada cuando utilice etiquetas para controlar el acceso a claves, utilice la clave de condición `aws:RequestTag/`*`tag-key`* o `aws:TagKeys` para determinar qué etiquetas (o claves de etiqueta) están permitidas.

Por ejemplo, la siguiente política IAM es similar a la anterior. Sin embargo, esta política permite a las entidades principales crear etiquetas (`TagResource`) y eliminar etiquetas `UntagResource` solo para etiquetas con una clave de etiqueta `Project`.

Como `TagResource` las `UntagResource` solicitudes pueden incluir varias etiquetas, debe especificar un operador `ForAllValues` o `ForAnyValue` configurarlo con la TagKeys condición aws:. El operador `ForAnyValue` requiere que al menos una de las claves de etiqueta de la solicitud coincida con una de las claves de etiqueta de la política. El operador `ForAllValues` requiere que todas las claves de etiqueta de la solicitud coincidan con una de las claves de etiqueta de la política. El `ForAllValues` operador también retorna `true` si no hay etiquetas en la solicitud, pero TagResource no lo UntagResource hace si no se especifica ninguna etiqueta. Para obtener información detallada sobre los operadores de conjunto, consulte Usar varias claves y valores en la Guía del usuario de IAM.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "IAMPolicyCreateKey",
      "Effect": "Allow",
      "Action": "payment-cryptography:CreateKey",
      "Resource": "*"
    },
    {
      "Sid": "IAMPolicyViewAllTags",
      "Effect": "Allow",
      "Action": "payment-cryptography:ListResourceTags",
```

```
      "Resource": "arn:aws:payment-cryptography:*:111122223333:key/*"
    },
    {
      "Sid": "IAMPolicyManageTags",
      "Effect": "Allow",
      "Action": [
        "payment-cryptography:TagResource",
        "payment-cryptography:UntagResource"
      ],
      "Resource": "arn:aws:payment-cryptography:*:111122223333:key/*",
      "Condition": {
          "ForAllValues:StringEquals": {"aws:TagKeys": "Project"}
      }
    }
  ]
}
```

Uso de etiquetas para controlar el acceso a las claves

Puede controlar el acceso a la criptografía de AWS pagos en función de las etiquetas de la clave. Por ejemplo, puede escribir una política de IAM que permita a las entidades principales habilitar y desactivar solo las claves que tienen una etiqueta concreta. O bien, puede utilizar una política de IAM para evitar que las entidades principales utilicen claves en operaciones criptográficas, a menos que la clave tenga una etiqueta concreta.

Esta función forma parte del soporte de criptografía de AWS pagos para el control de acceso basado en atributos (ABAC). Para obtener información sobre el uso de etiquetas para controlar el acceso a AWS los recursos, consulta ¿Para qué sirve el ABAC? AWS y Cómo controlar el acceso a AWS los recursos mediante etiquetas de recursos en la Guía del usuario de IAM.

> **Note**

AWS La criptografía de pagos admite la clave contextual de condición global aws:ResourceTag/tag-key, que permite controlar el acceso a las claves en función de las etiquetas de la clave. Dado que varias claves pueden tener la misma etiqueta, esta función le permite aplicar el permiso a un conjunto seleccionado de claves. También puede cambiar fácilmente las claves del conjunto cambiando sus etiquetas.

En la criptografía AWS de pagos, la clave de `aws:ResourceTag/`*`tag-key`* condición solo se admite en las políticas de IAM. No se admite en las políticas clave, que se aplican solo a una clave, ni en las operaciones que no utilizan una clave en particular, como las operaciones ListKeyso ListAliases.

Controlar el acceso con etiquetas proporciona una forma sencilla, escalable y flexible de administrar los permisos. Sin embargo, si no está diseñado y administrado correctamente, puede permitir o denegar el acceso a sus claves inadvertidamente. Si utiliza etiquetas para controlar el acceso, tenga en cuenta las siguientes prácticas.

- Utilice etiquetas para reforzar la práctica recomendada de acceso menos privilegiado. Proporcione a las entidades principales de IAM solo los permisos que necesitan y únicamente en las claves de que deben usar o administrar. Por ejemplo, utilice etiquetas para etiquetar las claves utilizadas en un proyecto. A continuación, dé permiso al equipo del proyecto para usar solo claves con la etiqueta de proyecto.
- Tenga cuidado al dar a las entidades principales los permisos `payment-cryptography:TagResource` y `payment-cryptography:UntagResource` que les permiten agregar, editar y eliminar etiquetas. Cuando utiliza etiquetas para controlar el acceso a las claves, cambiar una etiqueta puede dar permiso a las entidades principales para usar claves que de otro modo no tenían permiso para usar. También puede denegar el acceso a las claves que otras entidades principales requieren para realizar sus trabajos. Los administradores de claves que no tienen permiso para cambiar políticas de claves o crear concesiones pueden controlar el acceso a claves si tienen permiso para administrar etiquetas.

  Siempre que sea posible, utilice una condición de política, como `aws:RequestTag/`*`tag-key`* o `aws:TagKeys` para limitar los permisos de etiquetado de una entidad principal a determinadas etiquetas o patrones de etiquetas en determinadas claves.
- Revisa los elementos principales Cuenta de AWS que actualmente tienen permisos para etiquetar y desetiquetar y ajústalos si es necesario. Las políticas de IAM pueden habilitar permisos de etiqueta y desetiqueta en todas las claves. Por ejemplo, la política Admin permite a las entidades principales administradas etiquetar, desetiquetar y generar un lista de etiquetas en todas las claves.
- Antes de establecer una política que dependa de una etiqueta, revisa las etiquetas de las claves de tu. Cuenta de AWS Asegúrese de que su política solo se aplique a las etiquetas que desea incluir. Usa CloudTrail registros y CloudWatch alarmas para avisarte de los cambios en las etiquetas que puedan afectar al acceso a tus llaves.

- Las condiciones de política basadas en etiquetas utilizan la coincidencia de patrones; no están vinculadas a una instancia concreta de una etiqueta. Una política que utiliza claves de condición basadas en etiquetas afecta a todas las etiquetas nuevas y existentes que coincidan con el patrón. Si elimina y vuelve a crear una etiqueta que coincida con una condición de política, la condición se aplica a la nueva etiqueta, igual que a la anterior.

Por ejemplo, tomemos el siguiente ejemplo de política de IAM. Permite a las entidades principales llamar a las operaciones de Descifrado sólo en claves de su cuenta que sean de la región Este de EE. UU. (Norte de Virginia) y tengan una etiqueta `"Project"="Alpha"`. Puede adjuntar esta política a roles del ejemplo de proyecto Alpha.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "IAMPolicyWithResourceTag",
      "Effect": "Allow",
      "Action": [
        "payment-cryptography:DecryptData"
      ],
      "Resource": "arn:aws::us-east-1:111122223333:key/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Project": "Alpha"
        }
      }
    }
  ]
}
```

La siguiente política de IAM de ejemplo permite a la entidad principal utilizar la clave en la cuenta para operaciones criptográficas. Pero prohíbe a las entidades principales usar estas operaciones criptográficas en claves con una etiqueta `"Type"="Reserved"` o sin etiqueta `"Type"`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "IAMAllowCryptographicOperations",
      "Effect": "Allow",
```

```
      "Action": [
        "payment-cryptography:EncryptData",
        "payment-cryptography:DecryptData",
        "payment-cryptography:ReEncrypt*"
      ],
      "Resource": "arn:aws:payment-cryptography:*:111122223333:key/*"
    },
    {
      "Sid": "IAMDenyOnTag",
      "Effect": "Deny",
      "Action": [
        "payment-cryptography:EncryptData",
        "payment-cryptography:DecryptData",
        "payment-cryptography:ReEncrypt*"
      ],
      "Resource": "arn:aws:payment-cryptography:*:111122223333:key/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Type": "Reserved"
        }
      }
    },
    {
      "Sid": "IAMDenyNoTag",
      "Effect": "Deny",
      "Action": [
        "payment-cryptography:EncryptData",
        "payment-cryptography:DecryptData",
        "payment-cryptography:ReEncrypt*"
      ],
      "Resource": "arn:aws:kms:*:111122223333:key/*",
      "Condition": {
        "Null": {
          "aws:ResourceTag/Type": "true"
        }
      }
    }
  ]
}
```

Comprender los atributos clave de la clave AWS de criptografía de pagos

Un principio de la gestión adecuada de claves es que éstas tengan un alcance apropiado y sólo puedan utilizarse para operaciones permitidas. Como tal, ciertas claves sólo pueden crearse con ciertos modos de uso. Siempre que sea posible, esto se alinea con los modos de uso disponibles definidos por TR-31.

Si bien la criptografía de AWS pagos le impedirá crear claves no válidas, aquí encontrará combinaciones válidas para su comodidad.

Claves simétricas

- TR31_B0_BASE_DERIVATION_KEY
  - Algoritmos de clave permitidos: TDES_2KEY, TDES_3KEY, AES_128, AES_192, AES_256
  - Combinación permitida de modos de uso clave: { DeriveKey = true}, { NoRestrictions = true}
- TR31_C0_CARD_VERIFICATION_KEY
  - Algoritmos de clave permitidos: TDES_2KEY, TDES_3KEY, AES_128, AES_192, AES_256
  - Combinación permitida de modos de uso clave: {Generate = true}, {Verify = true}, {Generate = true, Verify= true}, { NoRestrictions = true}
- TR31_D0_SYMMETRIC_DATA_ENCRYPTION_KEY
  - Algoritmos de clave permitidos: TDES_2KEY, TDES_3KEY, AES_128, AES_192, AES_256
  - Combinación de modos de uso clave permitida: {Encrypt = true, Decrypt = true, Wrap = true, Unwrap = true}, {Encrypt = true}, {Decrypt = true, Unwrap = true}, {= true} NoRestrictions
- TR31_E0_EMV_MKEY_APP_CRYPTOGRAMS
  - Algoritmos de clave permitidos: TDES_2KEY, TDES_3KEY, AES_128, AES_192, AES_256
  - Combinación permitida de modos de uso clave: { DeriveKey = true}, {= true} NoRestrictions
- TR31_E1_EMV_MKEY_CONFIDENTIALITY
  - Algoritmos de clave permitidos: TDES_2KEY, TDES_3KEY, AES_128, AES_192, AES_256
  - Combinación permitida de modos de uso clave: { DeriveKey = true}, { NoRestrictions = true}
- TR31_E2_EMV_MKEY_INTEGRITY
  - Algoritmos de clave permitidos: TDES_2KEY, TDES_3KEY, AES_128, AES_192, AES_256
  - Combinación permitida de modos de uso clave: { DeriveKey = true}, { NoRestrictions = true}

- TR31_E4_EMV_MKEY_DYNAMIC_NUMBERS
  - Algoritmos de clave permitidos: TDES_2KEY, TDES_3KEY, AES_128, AES_192, AES_256
  - Combinación permitida de modos de uso clave: { DeriveKey = true}, { NoRestrictions = true}
- TR31_E5_EMV_MKEY_CARD_PERSONALIZATION
  - Algoritmos de clave permitidos: TDES_2KEY, TDES_3KEY, AES_128, AES_192, AES_256
  - Combinación permitida de modos de uso clave: { DeriveKey = true}, { NoRestrictions = true}
- TR31_E6_EMV_MKEY_OTHER
  - Algoritmos de clave permitidos: TDES_2KEY, TDES_3KEY, AES_128, AES_192, AES_256
  - Combinación permitida de modos de uso clave: { DeriveKey = true}, { NoRestrictions = true}
- TR31_K0_KEY_ENCRYPTION_KEY
  - Algoritmos de clave permitidos: TDES_2KEY, TDES_3KEY, AES_128, AES_192, AES_256
  - Combinación permitida de modos de uso clave: {Encrypt = true, Decrypt = true, Wrap = true, Unwrap = true}, {Encrypt = true, Wrap = true}, {= true} NoRestrictions
- TR31_M3_ISO_9797_3_MAC_KEY
  - Algoritmos de clave permitidos: TDES_2KEY, TDES_3KEY, AES_128, AES_192, AES_256
  - Combinación permitida de modos de uso clave: {Generate = true}, {Verify = true}, {Generate = true, Verify= true}, {= true} NoRestrictions
- TR31_M6_ISO_9797_5_CMAC_KEY
  - Algoritmos de clave permitidos: TDES_2KEY, TDES_3KEY, AES_128, AES_192, AES_256
  - Combinación permitida de modos de uso clave: {Generate = true}, {Verify = true}, {Generate = true, Verify= true}, {= true} NoRestrictions
- TR31_M7_HMAC_KEY
  - Algoritmos de clave permitidos: TDES_2KEY, TDES_3KEY, AES_128, AES_192, AES_256
  - Combinación permitida de modos de uso clave: {Generate = true}, {Verify = true}, {Generate = true, Verify= true}, {= true} NoRestrictions
- TR31_P0_PIN_ENCRYPTION_KEY
  - Algoritmos de clave permitidos: TDES_2KEY, TDES_3KEY, AES_128, AES_192, AES_256
  - Combinación de modos de uso clave permitida: {Encrypt = true, Decrypt = true, Wrap = true, Unwrap = true}, {Encrypt = true}, {Decrypt = true, Unwrap = true}, {= true} NoRestrictions
- TR31_V1_IBM3624_PIN_VERIFICATION_KEY
  - Algoritmos de clave permitidos: TDES_2KEY, TDES_3KEY, AES_128, AES_192, AES_256

  - Combinación permitida de modos de uso clave: {Generate = true}, {Verify = true}, {Generate = true, Verify= true}, {= true} NoRestrictions
- TR31_V2_VISA_PIN_VERIFICATION_KEY
  - Algoritmos de clave permitidos: TDES_2KEY, TDES_3KEY, AES_128, AES_192, AES_256
  - Combinación permitida de modos de uso clave: {Generate = true}, {Verify = true}, {Generate = true, Verify= true}, {= true} NoRestrictions

Claves asimétricas

- TR31_D1_ASYMMETRIC_KEY_FOR_DATA_ENCRYPTION
  - Algoritmos de clave permitidos: RSA_2048, RSA_3072, RSA_4096
  - Combinación permitida de modos de uso clave: {Encrypt = true, Decrypt = true, Wrap = true, Unwrap = true}, {Encrypt = true, Wrap = true}, {Decrypt = true, Unwrap = true}, {NoRestrictions = true}
  - NOTA: {Encrypt = true, Wrap = true} es la única opción válida al importar una clave pública destinada a cifrar datos o empaquetar una clave
- TR31_S0_ASYMMETRIC_KEY_FOR_DIGITAL_SIGNATURE
  - Algoritmos de clave permitidos: RSA_2048, RSA_3072, RSA_4096
  - Combinación permitida de modos de uso de las teclas: {Sign = true}, {Verify = true}
  - NOTA: {Verify = true} es la única opción válida al importar una clave destinada a la firma, como un certificado raíz, un certificado intermedio o un certificado de firma para el TR-34.

Operaciones de datos

Una vez establecida una clave de criptografía de AWS pago, se puede utilizar para realizar operaciones criptográficas. Las diferentes operaciones realizan diferentes tipos de actividad, desde el cifrado y el hash hasta algoritmos específicos del dominio, como la generación de CVV2.

Los datos cifrados no se pueden descifrar sin la clave de descifrado correspondiente (la clave simétrica o la clave privada, según el tipo de cifrado). De la misma forma, los algoritmos de hash y los algoritmos específicos de dominio no se pueden verificar sin la clave simétrica o la clave pública.

Para obtener información sobre los tipos de clave válidos para operaciones específicas, consulte Claves válidas para operaciones criptográficas

> **Note**
>
> Recomendamos utilizar datos de prueba en un entorno que no sea de producción. El uso de claves y datos de producción (PAN, ID de BDK, etc.) en un entorno ajeno a la producción puede afectar al ámbito de cumplimiento, por ejemplo, en el caso de PCI DSS y PCI P2PE.

Temas

Cifrar, descifrar y volver a cifrar datos

Los métodos de cifrado y descifrado se pueden utilizar para cifrar o descifrar datos mediante una variedad de técnicas simétricas y asimétricas, incluidas TDES, AES y RSA. Estos métodos también admiten claves derivadas mediante las técnicas DUKPT y EMV. Para los casos de uso en los que desee proteger los datos con una clave nueva sin exponer los datos subyacentes, también se puede utilizar el ReEncrypt comando.

> **Note**
>
> Al utilizar las funciones de cifrado y descifrado, se supone que todas las entradas están en HexBinary; por ejemplo, un valor de 1 se introduce como 31 (hexadecimal) y una t minúscula se representa como 74 (hexadecimal). Todas las salidas también están en hexBinary.

Para obtener más información sobre todas las opciones disponibles, consulte la Guía de la API para cifrar, descifrar y volver a cifrar.

Temas

Cifrar datos

La `Encrypt Data` API se utiliza para cifrar datos mediante claves de cifrado de datos simétricas y asimétricas, así como claves derivadas de DUKPT y EMV. Se admiten varios algoritmos y variaciones, incluidos `TDES`, `RSA` y `AES`.

Las entradas principales son la clave de cifrado utilizada para cifrar los datos, los datos de texto simple en formato HexBinary que se van a cifrar y los atributos de cifrado, como el vector de inicialización y el modo para los cifrados por bloques, como el TDES. Los datos en texto plano deben estar en múltiplos de 8 bytes para`TDES`, 16 bytes para `AES` y la longitud de la clave en el caso de. `RSA` Las entradas clave simétricas (TDES, AES, DUKPT, EMV) deben rellenarse en los casos en que los datos de entrada no cumplan estos requisitos. La siguiente tabla muestra la longitud máxima del texto sin formato para cada tipo de clave y el tipo de relleno que se define para las claves RSA. `EncryptionAttributes`

Tipo de relleno	RSA_2048	RSA_3072	RSA_4096
`OAEP SHA1`	428	684	940
`OAEP SHA256`	380	636	892
`OAEP SHA512`	252	508	764

Tipo de relleno	RSA_2048	RSA_3072	RSA_4096
PKCS1	488	744	1 000
None	488	744	1 000

Las salidas primarias incluyen los datos encriptados como texto cifrado en formato hexBinario y el valor de la suma de comprobación de la clave de encriptación. Para obtener más información sobre todas las opciones disponibles, consulte la Guía de API para Encrypt.

Ejemplos

- Cifrar datos utilizando la clave simétrica AES
- Cifrar los datos con la clave DUKPT
- Cifre los datos mediante una clave simétrica derivada de EMV
- Cifrado de los datos utilizando una clave de RSA

Cifrar datos utilizando la clave simétrica AES

> **Note**
>
> En todos los ejemplos se asume que la clave correspondiente ya existe. Las claves se pueden crear mediante la CreateKeyoperación o importar mediante la ImportKeyoperación.

Example

En este ejemplo, cifraremos los datos en texto plano mediante una clave simétrica que se creó mediante la CreateKeyoperación o se importó mediante la operación. ImportKey Para esta operación, la clave debe estar configurada en `Encrypt` y KeyModesOfUse KeyUsage establecida en. `TR31_D0_SYMMETRIC_DATA_ENCRYPTION_KEY` Consulte Claves para operaciones criptográficas para ver más opciones.

```
$ aws payment-cryptography-data encrypt-data --key-identifier arn:aws:payment-
cryptography:us-east-2:111122223333:key/tqv5yij6wtxx64pi --plain-text
 313233343132333431323334313233343 --encryption-attributes 'Symmetric={Mode=CBC}'
```

```
{
    "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
tqv5yij6wtxx64pi",
    "KeyCheckValue": "71D7AE",
    "CipherText": "33612AB9D6929C3A828EB6030082B2BD"
}
```

Cifrar los datos con la clave DUKPT

Example

En este ejemplo, cifraremos los datos en texto plano mediante una clave DUKPT. AWS Soportes de criptografía de pagos y claves DUKPT. `TDES AES` Para esta operación, la clave debe estar configurada en `DeriveKey` y KeyModesOfUse KeyUsage configurada en. `TR31_B0_BASE_DERIVATION_KEY` Consulte Claves para operaciones criptográficas para ver más opciones.

```
$ aws payment-cryptography-data encrypt-data --key-identifier
 arn:aws:payment-cryptography:us-east-2:111122223333:key/tqv5yij6wtxx64pi
 --plain-text 31323334313233343132333431323334 --encryption-attributes
 'Dukpt={KeySerialNumber=FFFF9876543210E00001}'
```

```
{
    "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
tqv5yij6wtxx64pi",
    "KeyCheckValue": "71D7AE",
    "CipherText": "33612AB9D6929C3A828EB6030082B2BD"
}
```

Cifre los datos mediante una clave simétrica derivada de EMV

Example

En este ejemplo, cifraremos los datos de texto no cifrado mediante una clave simétrica derivada de EMV que ya se ha creado. Puede utilizar un comando como este para enviar datos a una tarjeta EMV. Para esta operación, la clave debe estar KeyModesOfUse configurada en `Derive` y KeyUsage establecida en `TR31_E1_EMV_MKEY_CONFIDENTIALITY` o`TR31_E6_EMV_MKEY_OTHER`. Consulte Claves para operaciones criptográficas para obtener más información.

```
$ aws payment-cryptography-data encrypt-data --key-identifier
 arn:aws:payment-cryptography:us-east-2:111122223333:key/tqv5yij6wtxx64pi
 --plain-text 33612AB9D6929C3A828EB6030082B2BD --encryption-attributes
 'Emv={MajorKeyDerivationMode=EMV_OPTION_A,PanSequenceNumber=27,PrimaryAccountNumber=1000000000
 InitializationVector=1500000000000999,Mode=CBC}'
```

```
{
    "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
tqv5yij6wtxx64pi",
    "KeyCheckValue": "71D7AE",
    "CipherText": "33612AB9D6929C3A828EB6030082B2BD"
}
```

Cifrado de los datos utilizando una clave de RSA

Example

En este ejemplo, cifraremos los datos en texto plano mediante una clave pública RSA que se importó mediante la operación. ImportKey Para esta operación, la clave debe estar configurada en `Encrypt` y KeyModesOfUse KeyUsage establecida en. `TR31_D1_ASYMMETRIC_KEY_FOR_DATA_ENCRYPTION` Consulte Claves para operaciones criptográficas para ver más opciones.

Para PKCS #7 u otros esquemas de relleno no admitidos actualmente, solicítelos antes de llamar al servicio y seleccione sin relleno omitiendo el indicador de relleno 'Asymmetric={}'

```
$ aws payment-cryptography-data encrypt-data --key-identifier
 arn:aws:payment-cryptography:us-east-2:111122223333:key/thfezpmsalcfwmsg
 --plain-text 313233343132333431323334313233343132333431323334 --encryption-attributes
 'Asymmetric={PaddingType=OAEP_SHA256}'
```

```
{
 "CipherText":
 "12DF6A2F64CC566D124900D68E8AFEAA794CA819876E258564D525001D00AC93047A83FB13 \
 E73F06329A100704FA484A15A49F06A7A2E55A241D276491AA91F6D2D8590C60CDE57A642BC64A897F4832A3930
 \
 0FAEC7981102CA0F7370BFBF757F271EF0BB2516007AB111060A9633D1736A9158042D30C5AE11F8C5473EC70F067
 \
 72590DEA1638E2B41FAE6FB1662258596072B13F8E2F62F5D9FAF92C12BB70F42F2ECDCF56AADF0E311D4118FE3591
 \
 FB672998CCE9D00FFFE05D2CD154E3120C5443C8CF9131C7A6A6C05F5723B8F5C07A4003A5A6173E1B425E2B5E42AD
 \
 7A2966734309387C9938B029AFB20828ACFC6D00CD1539234A4A8D9B94CDD4F23A",
 "KeyArn": "arn:aws:payment-cryptography:us-east-1:529027455495:key/5dza7xqd6soanjtb",
 "KeyCheckValue": "FF9DE9CE"
}
```

Descifrado de datos

La `Decrypt Data` API se utiliza para descifrar datos mediante claves de cifrado de datos simétricas y asimétricas, así como claves derivadas de DUKPT y EMV. Se admiten varios algoritmos y variaciones, incluidos `TDES`, `RSA` y `AES`.

Las entradas principales son la clave de descifrado utilizada para descifrar los datos, los datos del texto cifrado en formato hexBinario que deben descifrarse y los atributos de descifrado, como el vector de inicialización, el modo como los cifradores de bloques, etc. Las salidas principales incluyen los datos descifrados como texto plano en formato hexBinario y el valor de la suma de comprobación de la clave de descifrado. Para obtener más información sobre todas las opciones disponibles, consulte la Guía de API para descifrar.

Ejemplos

- Descifrar los datos mediante la clave simétrica AES
- Descifrar los datos con la clave DUKPT
- Descifre los datos mediante una clave simétrica derivada de EMV
- Descifrar datos con una clave de RSA

Descifrar los datos mediante la clave simétrica AES

Example

En este ejemplo, descifraremos los datos de texto cifrado mediante una clave simétrica. En este ejemplo se muestra una AES clave, pero también TDES_2KEY se admiten. TDES_3KEY Para esta operación, la clave debe estar KeyModesOfUse configurada en `Decrypt` y KeyUsage establecida en`TR31_D0_SYMMETRIC_DATA_ENCRYPTION_KEY`. Consulte Claves para operaciones criptográficas para ver más opciones.

```
$ aws payment-cryptography-data decrypt-data --key-identifier arn:aws:payment-
cryptography:us-east-2:111122223333:key/tqv5yij6wtxx64pi --cipher-text
 33612AB9D6929C3A828EB6030082B2BD --decryption-attributes 'Symmetric={Mode=CBC}'
```

```
{
    "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
tqv5yij6wtxx64pi",
    "KeyCheckValue": "71D7AE",
    "PlainText": "31323334313233343132333431323334"
}
```

Descifrar los datos con la clave DUKPT

> **Note**
>
> El uso del descifrado de datos con DUKPT para las transacciones P2PE puede devolver a su aplicación el PAN de la tarjeta de crédito y otros datos del titular de la tarjeta que deberán tenerse en cuenta a la hora de determinar su alcance PCI DSS.

Example

En este ejemplo, descifraremos los datos de texto cifrado mediante una clave DUKPT que se creó mediante la CreateKeyoperación o se importó mediante la operación. ImportKey Para esta operación, la clave debe estar configurada en y KeyModesOfUse establecida en. `DeriveKey`

KeyUsage TR31_B0_BASE_DERIVATION_KEY Consulte Claves para operaciones criptográficas para ver más opciones. Cuando se utiliza DUKPT, como algoritmo TDES, la longitud de los datos del texto cifrado debe ser un múltiplo de 16 bytes. Para el algoritmo AES, la longitud de los datos del texto cifrado debe ser un múltiplo de 32 bytes.

```
$ aws payment-cryptography-data decrypt-data --key-identifier
 arn:aws:payment-cryptography:us-east-2:111122223333:key/tqv5yij6wtxx64pi
 --cipher-text 33612AB9D6929C3A828EB6030082B2BD --decryption-attributes
 'Dukpt={KeySerialNumber=FFFF9876543210E00001}'
```

```
{
    "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
tqv5yij6wtxx64pi",
    "KeyCheckValue": "71D7AE",
    "PlainText": "31323334313233343132333431323334"
}
```

Descifre los datos mediante una clave simétrica derivada de EMV

Example

En este ejemplo, descifraremos los datos de texto cifrado mediante una clave simétrica derivada de EMV que se creó mediante la operación o se importó mediante la operación. CreateKeyImportKey Para esta operación, la clave debe estar establecida en y KeyModesOfUse establecida en o. Derive KeyUsage TR31_E1_EMV_MKEY_CONFIDENTIALITY TR31_E6_EMV_MKEY_OTHER Consulte Claves para operaciones criptográficas para obtener más información.

```
$ aws payment-cryptography-data decrypt-data --key-identifier
 arn:aws:payment-cryptography:us-east-2:111122223333:key/tqv5yij6wtxx64pi
 --cipher-text 33612AB9D6929C3A828EB6030082B2BD --decryption-attributes
 'Emv={MajorKeyDerivationMode=EMV_OPTION_A,PanSequenceNumber=27,PrimaryAccountNumber=1000000000
 InitializationVector=1500000000000999,Mode=CBC}'
```

```
{
```

```
"KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/tqv5yij6wtxx64pi",
"KeyCheckValue": "71D7AE",
"PlainText": "31323334313233343132333431323334"
}
```

Descifrar datos con una clave de RSA

Example

En este ejemplo, descifraremos los datos de texto cifrado mediante un par de claves RSA que se creó mediante la operación. CreateKey Para esta operación, la clave debe estar configurada como `Decrypt` habilitada y KeyModesOfUse configurada como. KeyUsage `TR31_D1_ASYMMETRIC_KEY_FOR_DATA_ENCRYPTION` Consulte Claves para operaciones criptográficas para ver más opciones.

Para PKCS #7 u otros esquemas de relleno no admitidos actualmente, seleccione sin relleno omitiendo el indicador de relleno 'Asymmetric={}' y elimine el relleno después de llamar al servicio.

```
$ aws payment-cryptography-data decrypt-data \
        --key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/5dza7xqd6soanjtb --cipher-text
 8F4C1CAFE7A5DEF9A40BEDE7F2A264635C... \
        --decryption-attributes 'Asymmetric={PaddingType=OAEP_SHA256}'
```

```
{
    "KeyArn": "arn:aws:payment-cryptography:us-
east-1:529027455495:key/5dza7xqd6soanjtb",
    "KeyCheckValue": "FF9DE9CE",
    "PlainText": "31323334313233343132333431323334"
}
```

Generación y verificación de datos de tarjetas

La función generar y verificar los datos de la tarjeta incorpora datos derivados de los datos de la tarjeta, por ejemplo, CVV, CVV2, CVC y DCVV.

Temas

- Generar datos de tarjetas

- Comprobación de datos de tarjetas

Generar datos de tarjetas

La API `Generate Card Data` se utiliza para generar datos de tarjetas mediante algoritmos como CVV, CVV2 o CVV2 dinámico. Para ver qué claves se pueden usar para este comando, consulte la sección Claves válidas para operaciones criptográficas.

Example

En este ejemplo, generaremos un CVV/CVV2 para un PAN determinado con entradas de PAN y la fecha de caducidad de la tarjeta. La fecha de caducidad de la tarjeta puede estar en formato MMAA o AAMM, pero debe coincidir con todos los usos posteriores para que la validación funcione correctamente. Esto supone que se ha generado una clave de verificación de la tarjeta.

```
$ aws payment-cryptography-data generate-card-validation-data --key-
identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/
tqv5yij6wtxx64pig --primary-account-number=171234567890123 --generation-attributes
 CardVerificationValue2={CardExpiryDate=0123}
```

```
    {
        "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
tqv5yij6wtxx64pi",
        "KeyCheckValue": "CADDA1",
        "ValidationData": "801"
    }
```

Comprobación de datos de tarjetas

`Verify Card Data` se utiliza para verificar los datos que se han creado mediante algoritmos de pago que se basan en principios de cifrado, como `DISCOVER_DYNAMIC_CARD_VERIFICATION_CODE`.

Los valores de entrada suelen proporcionarse como parte de una transacción entrante a un emisor o a un socio de plataforma de apoyo. Para verificar un criptograma ARQC (utilizado para tarjetas con chips EMV), consulte Verificar el ARQC.

Si se verifica el valor, la API devolverá http/200. Si no se verifica el valor, devolverá http/400.

Example

En este ejemplo, validaremos un CVV/CVV2 para un PAN determinado. Por lo general, el titular de la tarjeta o el usuario proporcionan el CVV2 durante el momento de la transacción para su validación. Para validar su entrada, se proporcionarán los siguientes valores durante el tiempo de ejecución: la clave que se utilizará para la validación (CVK)PAN, la fecha de caducidad de la tarjeta y la introducción del CVV2. El formato de caducidad de la tarjeta debe coincidir con el utilizado en la generación del valor inicial.

```
$ aws payment-cryptography-data verify-card-validation-data --key-identifier
 arn:aws:payment-cryptography:us-east-2:111122223333:key/tqv5yij6wtxx64pi
 --primary-account-number=171234567890123 --verification-attributes
 CardVerificationValue2={CardExpiryDate=0123} --validation-data 801
```

```
    {
        "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
tqv5yij6wtxx64pi",
        "KeyCheckValue": "CADDA1"
    }
```

Generar, traducir y verificar los datos del PIN

Las funciones de datos PIN permiten generar códigos PIN al azar, valores de verificación de números PIN (PVV) y validar los pines entrantes cifrados comparándolos con valores PVV o compensaciones de PIN.

La traducción de pines permite convertir un pin de una clave funcional a otra sin exponer el pin en texto claro, tal y como se especifica en el requisito 1 del PIN PCI.

> **Note**
>
> Como la generación y la validación del PIN suelen ser funciones del emisor y la traducción del PIN es una función típica del adquirente, recomendamos que considere el acceso con menos privilegios y establezca las políticas adecuadas para el caso de uso de su sistema.

Temas

Traducir datos PIN

Las funciones de traducir datos PIN se utilizan para traducir los datos PIN cifrados de un conjunto de claves a otro sin que los datos cifrados salgan del HSM. Se utiliza para el cifrado P2PE, en el que las claves de trabajo deberían cambiar, pero el sistema de procesamiento no necesita descifrar los datos o no está autorizado a hacerlo. Las entradas principales son los datos cifrados, la clave de cifrado utilizada para cifrar los datos y los parámetros utilizados para generar los valores de entrada. El otro conjunto de entradas son los parámetros de salida solicitados, como la clave que se utilizará para cifrar la salida y los parámetros que se utilizarán para crear esa salida. Las salidas principales son un conjunto de datos recién cifrado, así como los parámetros utilizados para generarlo.

> **Note**
>
> Los tipos de claves AES solo admiten bloques de pines con ISO formato 4.

Temas

PIN de PEK a DUKPT

Example

En este ejemplo, traduciremos un PIN del cifrado PEK TDES mediante un bloque de PIN ISO 0 a un bloque de PIN AES ISO 4 mediante el algoritmo DUKPT. Normalmente, esto puede hacerse a la inversa, es decir, un terminal de pago cifra un PIN según la norma ISO 4 y, a continuación, puede volver a traducirlo a TDES para su posterior procesamiento.

```
$ aws payment-cryptography-data translate-pin-data --encrypted-pin-block
 "AC17DC148BDA645E" --incoming-translation-
```

```
attributes=IsoFormat0='{PrimaryAccountNumber=171234567890123}' --incoming-
key-identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/
ivi5ksfsuplneuyt  --outgoing-key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/4pmyquwjs3yj4vwe --outgoing-translation-attributes
 IsoFormat4="{PrimaryAccountNumber=171234567890123}" --outgoing-dukpt-attributes
 KeySerialNumber="FFFF9876543210E00008"
```

```
    {
            "PinBlock": "1F4209C670E49F83E75CC72E81B787D9",
            "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
ivi5ksfsuplneuyt  --outgoing-key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/4pmyquwjs3yj4vwe",
            "KeyCheckValue": "7CC9E2"
        }
```

PIN de DUKPT a AWK

Example

En este ejemplo, traduciremos un PIN de un PIN cifrado con AES DUKPT a un PIN cifrado con un AWK. Funcionalmente es lo contrario del ejemplo anterior.

```
$ aws payment-cryptography-data translate-pin-data --encrypted-pin-
block "1F4209C670E49F83E75CC72E81B787D9" --outgoing-translation-
attributes=IsoFormat0='{PrimaryAccountNumber=171234567890123}' --outgoing-
key-identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/
ivi5ksfsuplneuyt   --incoming-key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/4pmyquwjs3yj4vwe --incoming-translation-attributes
 IsoFormat4="{PrimaryAccountNumber=171234567890123}" --incoming-dukpt-attributes
 KeySerialNumber="FFFF9876543210E00008"
```

```
        {
            "PinBlock": "AC17DC148BDA645E",
            "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
ivi5ksfsuplneuyt",
            "KeyCheckValue": "FE23D3"
        }
```

Generar datos PIN

Las funciones de generación de datos PIN se utilizan para generar valores relacionados con el PIN, como el PVV y las compensaciones de bloques de pines que se utilizan para validar la introducción de los PIN por parte de los usuarios durante el tiempo de transacción o autorización. Esta API también puede generar un nuevo pin al azar mediante varios algoritmos.

Example

En este ejemplo, generaremos un nuevo pin (aleatorio) mediante el esquema de pines de Visa, donde las salidas serán cifradas `PIN block` (. PinData PinBlock) y un PVV (pinData.Offset). Las entradas clave son `PAN`, `Pin Verification Key`, `Pin Encryption Key` y `PIN block format`.

```
$ aws payment-cryptography-data generate-pin-data --generation-key-identifier
 arn:aws:payment-cryptography:us-east-2:111122223333:key/37y2tsl45p5zjbh2 --encryption-
key-identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/ivi5ksfsuplneuyt
 --primary-account-number 171234567890123 --pin-block-format ISO_FORMAT_0 --generation-
attributes VisaPin={PinVerificationKeyIndex=1}
```

```
{
        "GenerationKeyArn": "arn:aws:payment-cryptography:us-
east-2:111122223333:key/37y2tsl45p5zjbh2",
        "GenerationKeyCheckValue": "7F2363",
        "EncryptionKeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
ivi5ksfsuplneuyt",
        "EncryptionKeyCheckValue": "7CC9E2",
        "EncryptedPinBlock": "AC17DC148BDA645E",
        "PinData": {
            "VerificationValue": "5507"
        }
    }
```

Comprobación de datos PIN

Las funciones de comprobación de datos PIN se utilizan para comprobar si un PIN es correcto. Por lo general, esto implica comparar el valor del PIN previamente almacenado con el que ingresó el titular

de la tarjeta en un POI. Estas funciones comparan dos valores sin exponer el valor subyacente de ninguna de las fuentes.

Validar el PIN cifrado

Example

En este ejemplo, validaremos un PIN para un PAN determinado. El titular de la tarjeta o el usuario suelen proporcionar el PIN durante el momento de la transacción para su validación y se compara con el valor registrado (se proporciona como un valor cifrado). Para validar esta entrada, también se proporcionarán los siguientes valores en tiempo de ejecución: la clave utilizada para cifrar el pin de entrada (que suele ser `IWK`) PAN y el valor con el que realizar la verificación (`PVV` o `PIN offset`).

Si la criptografía de AWS pago puede validar el PIN, se devuelve un http/200. Si el pin no está validado, devolverá un http/400.

```
$ aws payment-cryptography-data verify-pin-data --verification-key-identifier
 arn:aws:payment-cryptography:us-east-2:111122223333:key/37y2tsl45p5zjbh2 --encryption-
key-identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/ivi5ksfsuplneuyt
 --primary-account-number 171234567890123 --pin-block-format ISO_FORMAT_0  --
verification-attributes VisaPin="{PinVerificationKeyIndex=1,VerificationValue=5507}" --
encrypted-pin-block AC17DC148BDA645E
```

```
        {
            "VerificationKeyArn": "arn:aws:payment-cryptography:us-
east-2:111122223333:key/37y2tsl45p5zjbh2",
            "VerificationKeyCheckValue": "7F2363",
            "EncryptionKeyArn": "arn:aws:payment-cryptography:us-
east-2:111122223333:key/ivi5ksfsuplneuyt",
            "EncryptionKeyCheckValue": "7CC9E2",

        }
```

Verificar el criptograma de solicitud de autenticación (ARQC)

La API de verificación del criptograma de solicitud de autenticación se utiliza para verificar el ARQC. La generación del ARQC está fuera del alcance de la criptografía de AWS pagos y, por lo general,

se realiza en una tarjeta con chip EMV (o un equivalente digital, como una billetera móvil) durante el tiempo de autorización de la transacción. Un ARQC es único para cada transacción y su objetivo es mostrar criptográficamente la validez de la tarjeta y garantizar que los datos de la transacción coincidan exactamente con los de la transacción actual (esperada).

AWS La criptografía de pagos ofrece una variedad de opciones para validar el ARQC y generar valores ARPC opcionales, incluidos los definidos en la EMV 4.4, libro 2, y otros esquemas utilizados por Visa y Mastercard. Para obtener una lista completa de todas las opciones disponibles, consulte la sección de la VerifyCardValidationData Guía de API.

Los criptogramas ARQC suelen requerir las siguientes entradas (aunque esto puede variar según la implementación):

- PAN: especificado en el campo PrimaryAccountNumber
- Número de secuencia PAN (PSN): especificado en el campo PanSequenceNumber
- Método de derivación de claves, como la clave de sesión común (CSK), especificado en el SessionKeyDerivationAttributes
- Modo de derivación de clave maestra (como la opción A de EMV): especificado en el MajorKeyDerivationMode
- Datos de la transacción: una cadena de varios datos de transacciones, terminales y tarjetas, como el importe y la fecha, especificados en el campo TransactionData
- Clave maestra del emisor: la clave maestra utilizada para derivar la clave de criptograma (AC) utilizada para proteger las transacciones individuales y especificada en el campo KeyIdentifier

Temas

Creación de datos de transacciones

El contenido (y el orden) exactos del campo de datos de la transacción varían según la implementación y el esquema de red, pero los campos mínimos recomendados (y la secuencia de concatenación) se definen en la sección 8.1.1 del libro 2 de EMV 4.4: Selección de datos. Si los tres primeros campos son importe (17.00), otro importe (0,00) y país de compra, los datos de la transacción comenzarán de la siguiente manera:

- 000000001700 - cantidad: 12 posiciones implican un decimal de dos dígitos
- 000000000000 - otra cantidad: 12 posiciones implican un decimal de dos dígitos
- 0124: código de país de cuatro dígitos
- Datos de transacción de salida (parciales): 00000000170000000000000000124

Relleno de datos de transacciones

Los datos de las transacciones deben rellenarse antes de enviarlos al servicio. La mayoría de los esquemas utilizan el relleno ISO 9797 Método 2, en el que se añade una cadena hexadecimal 80 seguida de 00 hasta que el campo es un múltiplo del tamaño del bloque de cifrado; 8 bytes o 16 caracteres para TDES y 16 bytes o 32 caracteres para AES. La alternativa (método 1) no es tan común pero utiliza sólo 00 como caracteres de relleno.

Relleno ISO 9797 Método 1

Sin relleno:
00000000170000000000000008400080008000084016051700000000093800000B03011203 (74 caracteres o 37 bytes)

Con relleno:
00000000170000000000000008400080008000084016051700000000093800000B03011203000000 (80 caracteres o 40 bytes)

Relleno ISO 9797 Método 2

Sin relleno:
00000000170000000000000008400080008000084016051700000000093800000B1F220103000000 (80 caracteres o 40 bytes)

Sin relleno:
00000000170000000000000008400080008000084016051700000000093800000B1F220103000000800000 (88 caracteres o 44 bytes)

Ejemplos

Visa CVN10

Example

En este ejemplo, validaremos un ARQC generado con Visa CVN10.

Si la criptografía AWS de pagos puede validar el ARQC, se devuelve un http/200. Si el arqc no se valida, devolverá una respuesta http/400.

```
$ aws payment-cryptography-data verify-auth-request-cryptogram --auth-request-
cryptogram D791093C8A921769 \
--key-identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/
pw3s6nl62t5ushfk \
--major-key-derivation-mode EMV_OPTION_A \
--transaction-data
 00000000170000000000000008400080008000084016051700000000093800000B03011203000000 \
--session-key-derivation-attributes='{"Visa":{"PanSequenceNumber":"01" \
,"PrimaryAccountNumber":"9137631040001422"}}'
```

```
{
  "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/pw3s6nl62t5ushfk",
  "KeyCheckValue": "08D7B4"
}
```

Visa CVN18 y Visa CVN22

Example

En este ejemplo, validaremos un ARQC generado con Visa CVN18 o CVN22. Las operaciones criptográficas son las mismas entre el CVN18 y el CVN22, pero los datos contenidos en los datos de las transacciones varían. En comparación con el CVN10, se genera un criptograma completamente diferente incluso con las mismas entradas.

Si la criptografía AWS de pagos puede validar el ARQC, se devuelve un http/200. Si el arqc no se valida, devolverá un http/400.

```
$ aws payment-cryptography-data verify-auth-request-cryptogram \
--auth-request-cryptogram 61EDCC708B4C97B4
--key-identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/
pw3s6nl62t5ushfk \
--major-key-derivation-mode EMV_OPTION_A
--transaction-data
 00000000170000000000000008400080008000084016051700000000093800000B1F22010300000000000
 \
00000000000000000000000000000000000000000000800000000000000000
--session-key-derivation-attributes='{"EmvCommon":
{"ApplicationTransactionCounter":"000B", \
"PanSequenceNumber":"01","PrimaryAccountNumber":"9137631040001422"}}'
```

```
{
  "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/pw3s6nl62t5ushfk",
  "KeyCheckValue": "08D7B4"
}
```

Generar y verificar MAC

Los Códigos de autenticación de mensajes (Message Authentication Codes. MAC) se utilizan normalmente para autentificar la integridad de un mensaje (independientemente de si se ha modificado). Los hashes criptográficos, como el HMAC (Código de autenticación de mensajes basado en hash), el CBC-MAC y el CMAC (Código de autenticación de mensajes basado en cifrado), proporcionan además una seguridad adicional para el remitente del MAC al utilizar la criptografía. El HMAC se basa en funciones de hash, mientras que el CMAC se basa en cifrados por bloques.

Todos los algoritmos MAC de este servicio combinan una función hash criptográfica y una clave secreta compartida. Reciben un mensaje y una clave secreta, como el material clave de una llave, y devuelven una etiqueta o mac únicos. Si cambia incluso un carácter del mensaje, o si la clave secreta cambia, la etiqueta resultante es totalmente diferente. Al requerir una clave secreta, los MAC criptográficos también proporcionan autenticación; es imposible generar una mac idéntica sin la clave secreta. Los MAC criptográficos a veces se llaman firmas simétricas, porque funcionan como firmas digitales, pero utilizan una única clave para la firma y la verificación.

La Criptografía de pagos de AWS admite varios tipos de MAC:

ALGORITMO 1 ISO9797

- Denotado por `KeyUsage` del ISO9797_ALGORITHM1

ALGORITMO 3 ISO9797 (MAC minorista)

- Denotado por `KeyUsage` del ISO9797_ALGORITHM3

ALGORITMO 5 ISO9797 (CMAC)

- Denotado por `KeyUsage` del TR31_M6_ISO_9797_5_CMAC_KEY

HMAC

- Denotado por `KeyUsage` del TR31_M7_HMAC_KEY incluyendo HMAC_SHA224, HMAC_SHA256, HMAC_SHA384 y HMAC_SHA512

Temas

Generar MAC

La API Generar MAC se utiliza para autentificar los datos relacionados con las tarjetas, como los datos de seguimiento de una banda magnética, mediante valores de datos conocidos para generar un MAC (código de autenticación de mensajes) para la validación de los datos entre las partes remitentes y receptoras. Los datos utilizados para generar el MAC incluyen los datos de los mensajes, la clave de cifrado MAC secreta y el algoritmo MAC para generar un valor MAC único para la transmisión. La parte receptora del MAC utiliza los mismos datos del mensaje MAC, la misma clave de cifrado MAC y el mismo algoritmo para reproducir otro valor MAC con fines de comparación y autenticación de datos. Aunque cambie un carácter del mensaje o la clave MAC utilizada para la verificación no sea idéntica, el valor MAC resultante será diferente. La API admite las claves de cifrado DUPKT MAC, HMAC y EMV MAC para esta operación.

El valor de entrada para `message-data` debe ser un dato hexBinary.

En este ejemplo, generaremos un HMAC (Hash-Based Message Authentication Code, Código de autenticación de mensajes basado en hash) para la autenticación de los datos de la tarjeta mediante el algoritmo HMAC `HMAC_SHA256` y la clave de cifrado HMAC. La clave debe tener KeyUsage establecido en `TR31_M7_HMAC_KEY` y KeyModesOfUse en `Generate`. La clave MAC puede crearse con la Criptografía de pagos de AWS llamando a CreateKey o importarse llamando a ImportKey.

Example

```
$ aws payment-cryptography-data generate-mac \
    --key-identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/
qnobl5lghrzunce6 \
    --message-data
 "3b3130383834393030313037333934313533d32343038323236303030373030303f33" \
    --generation-attributes Algorithm=HMAC_SHA256
```

```
{
    "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
qnobl5lghrzunce6,
```

```
    "KeyCheckValue": "2976E7",
    "Mac": "ED87F26E961C6D0DDB78DA5038AA2BDDEA0DCE03E5B5E96BDDD494F4A7AA470C"
}
```

Verificar MAC

Verifique que la MAC API se utiliza para verificar MAC (Message Authentication Code, Código de autenticación de mensajes) para la autenticación de datos relacionados con tarjetas. Debe utilizar la misma clave de cifrado utilizada durante la generación del MAC para reproducir el valor MAC para la autenticación. La clave de encriptación MAC puede crearse con la Criptografía de pagos de AWS llamando a CreateKey o importarse llamando a ImportKey. La API admite las claves de cifrado DUPKT MAC, HMAC y EMV MAC para esta operación.

Si el valor se verifica, el parámetro de respuesta `MacDataVerificationSuccessful` devolverá `Http/200`, en caso contrario `Http/400` con un mensaje indicando que `Mac verification failed`.

En este ejemplo, verificaremos un código HMAC (Código de autenticación de mensajes basado en hash, Hash-Based Message Authentication Code) para la autenticación de los datos de la tarjeta utilizando el algoritmo HMAC `HMAC_SHA256` y la clave de encriptación HMAC. La clave debe tener KeyUsage establecido en `TR31_M7_HMAC_KEY` y KeyModesOfUse en `Verify`.

Example

```
$ aws payment-cryptography-data verify-mac \
     --key-identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/
qnobl5lghrzunce6 \
     --message-data
 "3b343038383439303031303733393431353d323430383232363030303730303f33" \
     --verification-attributes='Algorithm=HMAC_SHA256' \
     --mac ED87F26E961C6D0DDB78DA5038AA2BDDEA0DCE03E5B5E96BDDD494F4A7AA470C
```

```
{
    "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
qnobl5lghrzunce6,
    "KeyCheckValue": "2976E7",
}
```

Claves válidas para las operaciones criptográficas

Algunas claves solo se pueden utilizar para determinadas operaciones. Además, algunas operaciones pueden limitar los modos de uso de las claves. Consulte la tabla siguiente para las combinaciones permitidas.

> **Note**
>
> Ciertas combinaciones, si bien están permitidas, pueden crear situaciones inutilizables, como generar códigos CVV (`generate`) pero luego no pueden verificarlos (`verify`).

Temas

GenerateCardData

Punto de conexión de la API	Operación criptográfica o algoritmo	Uso permitido de claves	Algoritmo clave permitido	Combinación permitida de modos de uso clave
GenerateCardData	• AMEX_CARD_SECURITY_CODE_VERSION_1 • AMEX_CARD_SECURITY_CODE_VERSION_2	TR31_C0_CARD_VERIFICATION_KEY	• TDES_2KEY • TDES_3KEY	{ Generate = true },{ Generate = true, Verify = true }
GenerateCardData	• CARD_VERIFICATION_VALUE_1 • CARD_VERIFICATION_VALUE_2	TR31_C0_CARD_VERIFICATION_KEY	• TDES_2KEY	{ Generate = true },{ Generate = true, Verify = true }
GenerateCardData	• CARDHOLDER_AUTHENTICATION_VERIFICATION_VALUE	TR31_E6_EMV_MKEY_OTHER	• TDES_2KEY	{ DeriveKey = verdadero}
GenerateCardData	• DYNAMIC_CARD_VERIFICATION_CODE	TR31_E4_EMV_MKEY_DYNAMIC_NUMBERS	• TDES_2KEY	{ DeriveKey = verdadero}
GenerateCardData	• DYNAMIC_CARD_VERIF	TR31_E6_EMV_MKEY_OTHER	• TDES_2KEY	{ DeriveKey = verdadero}

Punto de conexión de la API	Operación criptográfica o algoritmo	Uso permitido de claves	Algoritmo clave permitido	Combinación permitida de modos de uso clave
	ICATION_V ALUE			

VerifyCardData

Operación criptográfica o algoritmo	Uso permitido de claves	Algoritmo clave permitido	Combinación permitida de modos de uso clave
• AMEX_CARD _SECURITY _CODE_VER SION_1 • AMEX_CARD _SECURITY _CODE_VER SION_2	TR31_C0_C ARD_VERIF ICATION_KEY	• TDES_2KEY • TDES_3KEY	{ Generate = true }, { Generate = true, Verify = true }
• CARD_VERI FICATION_ VALUE_1 • CARD_VERI FICATION_ VALUE_2	TR31_C0_C ARD_VERIF ICATION_KEY	• TDES_2KEY	{ Generate = true }, { Generate = true, Verify = true }
• CARDHOLDE R_AUTHENT ICATION_V ERIFICATI ON_VALUE	TR31_E6_E MV_MKEY_OTHER	• TDES_2KEY	{ DeriveKey = verdadero}

Operación criptográfica o algoritmo	Uso permitido de claves	Algoritmo clave permitido	Combinación permitida de modos de uso clave
• DYNAMIC_CARD_VERIFICATION_CODE	TR31_E4_EMV_MKEY_DYNAMIC_NUMBERS	• TDES_2KEY	{ DeriveKey = verdadero}
• DYNAMIC_CARD_VERIFICATION_VALUE	TR31_E6_EMV_MKEY_OTHER	• TDES_2KEY	{ DeriveKey = verdadero}

GeneratePinData (para los regímenes VISA/ABA)

`VISA_PIN or VISA_PIN_VERIFICATION_VALUE`

Tipo de clave	Uso permitido de claves	Algoritmo clave permitido	Combinación permitida de modos de uso clave
Clave de cifrado de PIN	TR31_P0_PIN_ENCRYPTION_KEY	• TDES_2KEY • TDES_3KEY	• { Encrypt = true, Wrap = true } • { Encrypt = true, Decrypt = true, Wrap = true, Unwrap = true } • {= verdadero} NoRestrictions
Clave de generación de PIN	TR31_V2_VISA_PIN_VERIFICATION_KEY	• TDES_3KEY	• { Generate = true } • { Generate = true, Verify = true }

GeneratePinData (para**IBM3624**)

`IBM3624_PIN_OFFSET,IBM3624_NATURAL_PIN,IBM3624_RANDOM_PIN, IBM3624_PIN_FROM_OFFSET)`

Tipo de clave	Uso permitido de claves	Algoritmo clave permitido	Combinación permitida de modos de uso clave
Clave de cifrado de PIN	TR31_P0_PIN_ENCRYPTION_KEY	• TDES_2KEY • TDES_3KEY	Para IBM3624_NATURAL_PIN, IBM3624_RANDOM_PIN, IBM3624_PIN_FROM_OFFSET • { Encrypt = true, Wrap = true } • { Encrypt = true, Decrypt = true, Wrap = true, Unwrap = true } • { NoRestrictions = verdadero} Para IBM3624_PIN_OFFSET • { Encrypt = true, Unwrap = true } • { Encrypt = true, Decrypt = true, Wrap = true, Unwrap = true } • { NoRestrictions = verdadero}

Tipo de clave	Uso permitido de claves	Algoritmo clave permitido	Combinación permitida de modos de uso clave
Clave de generación de PIN	TR31_V1_IBM3624_PIN_VERIFICATION_KEY	• TDES_3KEY	• { Generate = true } • { Generate = true, Verify = true }

VerifyPinData (para los regímenes VISA/ABA)

`VISA_PIN`

Tipo de clave	Uso permitido de claves	Algoritmo clave permitido	Combinación permitida de modos de uso clave
Clave de cifrado de PIN	TR31_P0_PIN_ENCRYPTION_KEY	• TDES_2KEY • TDES_3KEY	• { Decrypt = true, Unwrap = true } • { Encrypt = true, Decrypt = true, Wrap = true, Unwrap = true } • {= verdadero} NoRestrictions
Clave de generación de PIN	TR31_V2_VISA_PIN_VERIFICATION_KEY	• TDES_3KEY	• { Verify = true } • { Generate = true, Verify = true }

VerifyPinData (para**IBM3624**)

`IBM3624_PIN_OFFSET,IBM3624_NATURAL_PIN,IBM3624_RANDOM_PIN, IBM3624_PIN_FROM_OFFSET)`

Tipo de clave	Uso permitido de claves	Algoritmo clave permitido	Combinación permitida de modos de uso clave
Clave de cifrado de PIN	TR31_P0_PIN_ENCRYPTION_KEY	• TDES_2KEY • TDES_3KEY	Para IBM3624_NATURAL_PIN, IBM3624_RANDOM_PIN, IBM3624_PIN_FROM_OFFSET • { Decrypt = true, Unwrap = true } • { Encrypt = true, Decrypt = true, Wrap = true, Unwrap = true } • { NoRestrictions = verdadero}
Clave de verificación de PIN	TR31_V1_IBM3624_PIN_VERIFICATION_KEY	• TDES_3KEY	• { Verify = true } • { Generate = true, Verify = true }

Descifrado de datos

Tipo de clave	Uso permitido de claves	Algoritmo clave permitido	Combinación permitida de modos de uso clave
DUKPT	TR31_B0_BASE_DERIVATION_KEY	• TDES_2KEY • AES_128 • AES_192 • AES_256	• { DeriveKey = verdadero} • { NoRestrictions = verdadero}

Tipo de clave	Uso permitido de claves	Algoritmo clave permitido	Combinación permitida de modos de uso clave
EMV	TR31_E1_EMV_MKEY_CONFIDENTIALITY TR31_E6_EMV_MKEY_OTHER	• TDES_2KEY	• { DeriveKey = verdadero}
RSA	TR31_D1_ASYMMETRIC_KEY_FOR_DATA_ENCRYPTION	• RSA_2048 • RSA_3072 • RSA_4096	• { Decrypt = true, Unwrap=true} • {Encrypt=true, Wrap=true,Decrypt = true, Unwrap=true}
Claves simétricas	TR31_D0_SYMMETRIC_DATA_ENCRYPTION_KEY	• TDES_2KEY • TDES_3KEY • AES_128 • AES_192 • AES_256	• {Decrypt = true, Unwrap=true} • {Encrypt=true, Wrap=true,Decrypt = true, Unwrap=true} • { NoRestrictions = verdadero}

Cifrado de datos

Tipo de clave	Uso permitido de claves	Algoritmo clave permitido	Combinación permitida de modos de uso clave
DUKPT	TR31_B0_BASE_DERIVATION_KEY	• TDES_2KEY • AES_128 • AES_192	• { DeriveKey = verdadero}

Tipo de clave	Uso permitido de claves	Algoritmo clave permitido	Combinación permitida de modos de uso clave
		• AES_256	• { NoRestrictions = verdadero}
EMV	TR31_E1_EMV_MKEY_CONFIDENTIALITY TR31_E6_EMV_MKEY_OTHER	• TDES_2KEY	• { DeriveKey = verdadero}
RSA	TR31_D1_ASYMMETRIC_KEY_FOR_DATA_ENCRYPTION	• RSA_2048 • RSA_3072 • RSA_4096	• { Encrypt = true, Wrap=true} • {Encrypt=true, Wrap=true,Decrypt = true, Unwrap=true}
Claves simétricas	TR31_D0_SYMMETRIC_DATA_ENCRYPTION_KEY	• TDES_2KEY • TDES_3KEY • AES_128 • AES_192 • AES_256	• {Encrypt = true, Wrap=true} • {Encrypt=true, Wrap=true,Decrypt = true, Unwrap=true} • { NoRestrictions = verdadero}

Traducir datos PIN

Dirección	Tipo de clave	Uso permitido de claves	Algoritmo clave permitido	Combinación permitida de modos de uso clave
Origen de datos de entrada	DUKPT	TR31_B0_BASE_DERIVATION_KEY	• TDES_2KEY • AES_128 • AES_192 • AES_256	• { DeriveKey = verdadero} • { NoRestrictions = verdadero}
Origen de datos de entrada	no DUKPT (PEK, AWK, IWK, etc.)	TR31_P0_PIN_ENCRYPTION_KEY	• TDES_2KEY • TDES_3KEY • AES_128 • AES_192 • AES_256	• { Decrypt = true, Unwrap = true } • { Encrypt = true, Decrypt = true, Wrap = true, Unwrap = true } • { NoRestrictions = verdadero}
Objetivo de datos de salida	DUKPT	TR31_B0_BASE_DERIVATION_KEY	• TDES_2KEY • AES_128 • AES_192 • AES_256	• { DeriveKey = verdadero} • { NoRestrictions = verdadero}
Objetivo de datos de salida	no DUKPT (PEK, IWK, AWK, etc)	TR31_P0_PIN_ENCRYPTION_KEY	• TDES_2KEY • TDES_3KEY • AES_128 • AES_192 • AES_256	• { Encrypt = true, Wrap = true } • { Encrypt = true, Decrypt

Dirección	Tipo de clave	Uso permitido de claves	Algoritmo clave permitido	Combinación permitida de modos de uso clave
				= true, Wrap = true, Unwrap = true } • { NoRestrictions = verdadero}

VerifyAuthRequestCryptogram

Uso permitido de claves	Opción EMV	Algoritmo clave permitido	Combinación permitida de modos de uso clave
• OPCIÓN A • OOCIÓN B	TR31_E0_EMV_MKEY_APP_CRYPTOGRAMS	• TDES_2KEY	• { DeriveKey = verdadero}

Tipos de claves sin utilizar

La criptografía de AWS pagos no utiliza actualmente los siguientes tipos de claves

- TR31_K1_KEY_BLOCK_PROTECTION_KEY
- TR31_P1_PIN_GENERATION_KEY
- TR31_K3_ASYMMETRIC_KEY_FOR_KEY_AGREEMENT

Seguridad en la criptografía de pagos AWS

La seguridad en la nube AWS es la máxima prioridad. Como AWS cliente, usted se beneficia de una arquitectura de centro de datos y red diseñada para cumplir con los requisitos de las organizaciones más sensibles a la seguridad.

La seguridad es una responsabilidad compartida entre usted AWS y usted. El modelo de responsabilidad compartida la describe como seguridad de la nube y seguridad en la nube:

- Seguridad de la nube:AWS es responsable de proteger la infraestructura que ejecuta AWS los servicios en la AWS nube. AWS también le proporciona servicios que puede utilizar de forma segura. Los auditores externos prueban y verifican periódicamente la eficacia de nuestra seguridad como parte de los AWS programas de de . Para obtener más información sobre los programas de conformidad que se aplican a la criptografía de AWS pagos, consulte Servicios de AWS en el ámbito de aplicación por programa de conformidad .
- Seguridad en la nube: su responsabilidad viene determinada por el AWS servicio que utilice. Usted también es responsable de otros factores, incluida la confidencialidad de los datos, los requisitos de la empresa y la legislación y los reglamentos aplicables.

Este tema le ayuda a entender cómo aplicar el modelo de responsabilidad compartida al utilizar la criptografía AWS de pagos. Le muestra cómo configurar la criptografía AWS de pagos para cumplir sus objetivos de seguridad y conformidad. También aprenderá a utilizar otros AWS servicios que le ayudan a supervisar y proteger sus recursos de criptografía AWS de pagos.

Temas

Protección de datos en la criptografía AWS de pagos

El modelo de se aplica a protección de datos en la criptografía de AWS pagos. Como se describe en este modelo, AWS es responsable de proteger la infraestructura global que ejecuta todos los Nube de AWS. Usted es responsable de mantener el control sobre el contenido alojado en esta

infraestructura. Usted también es responsable de las tareas de administración y configuración de seguridad para los Servicios de AWS que utiliza. Para obtener más información sobre la privacidad de los datos, consulte las Preguntas frecuentes sobre la privacidad de datos. Para obtener información sobre la protección de datos en Europa, consulte la publicación de blog sobre el Modelo de responsabilidad compartida de AWS y GDPR en el Blog de seguridad de AWS .

Con fines de protección de datos, le recomendamos que proteja Cuenta de AWS las credenciales y configure los usuarios individuales con AWS IAM Identity Center o AWS Identity and Access Management (IAM). De esta manera, solo se otorgan a cada usuario los permisos necesarios para cumplir sus obligaciones laborales. También recomendamos proteger sus datos de la siguiente manera:

- Utilice la autenticación multifactor (MFA) en cada cuenta.
- Utilice SSL/TLS para comunicarse con los recursos. AWS Se recomienda el uso de TLS 1.2 y recomendamos TLS 1.3.
- Configure la API y el registro de actividad de los usuarios con. AWS CloudTrail
- Utilice soluciones de AWS cifrado, junto con todos los controles de seguridad predeterminados Servicios de AWS.
- Utilice servicios de seguridad administrados avanzados, como Amazon Macie, que lo ayuden a detectar y proteger los datos confidenciales almacenados en Amazon S3.
- Si necesita módulos criptográficos validados por FIPS 140-2 para acceder a AWS través de una interfaz de línea de comandos o una API, utilice un punto final FIPS. Para obtener más información sobre los puntos de conexión de FIPS disponibles, consulte Estándar de procesamiento de la información federal (FIPS) 140-2.

Se recomienda encarecidamente no introducir nunca información confidencial o sensible, como, por ejemplo, direcciones de correo electrónico de clientes, en etiquetas o campos de formato libre, tales como el campo Nombre. Esto incluye cuando trabaja con criptografía de AWS pagos u otra forma Servicios de AWS mediante la consola, la API o los SDK. AWS CLI AWS Cualquier dato que ingrese en etiquetas o campos de formato libre utilizados para nombres se puede emplear para los registros de facturación o diagnóstico. Si proporciona una URL a un servidor externo, recomendamos encarecidamente que no incluya información de credenciales en la URL a fin de validar la solicitud para ese servidor.

AWS Payment Cryptography almacena y protege sus claves de cifrado de pagos para que estén altamente disponibles al mismo tiempo que le proporciona un control de acceso sólido y flexible.

Temas

Rotación del material de claves

Por defecto, AWS Payment Cryptography protege el material criptográfico de las claves de pago gestionadas por el servicio. Además, AWS Payment Cryptography ofrece opciones para importar material de claves creado fuera del servicio. Para obtener más detalles sobre las claves de pago y el material de claves, consulte los detalles de criptografía de AWS Payment Cryptography.

Cifrado de datos

Los datos de AWS Payment Cryptography consisten en claves de AWS Payment Cryptography, el material de claves criptográficas que representan y sus atributos de uso. El material de claves solo existe en texto sin formato dentro de los módulos de seguridad de hardware (HSM) de AWS Payment Cryptography y solo cuando estén en uso. De lo contrario, el material de la clave y los atributos se cifran y se almacenan en almacenamiento persistente duradero.

El material de claves que AWS Payment Cryptography genera o carga para las claves de pago nunca abandona el límite de los HSM de AWS Payment Cryptography sin cifrar. Puede ser exportado cifrado por las operaciones API de AWS Payment Cryptography.

Cifrado en reposo

AWS Payment Cryptography genera material de claves para claves de pago en los HSMs de la lista PCI PTS. Cuando no se utiliza, el material de claves se cifra mediante una clave de HSM y se escribe en un almacenamiento duradero y persistente. El material de las claves de Payment Cryptography y las claves de cifrado que protegen el material de claves nunca dejan los HSM en forma de texto sin formato.

El cifrado y la administración del material de claves para las claves de Payment Cryptography está completamente a cargo del servicio.

Para obtener más información, consulte el documento técnico Detalles criptográficos de AWS Key Management Service.

Cifrado en tránsito

El material de claves que AWS Payment Cryptography genera o carga para las claves de pago nunca se exporta o transmite en texto claro en las operaciones API de AWS Payment Cryptography. AWS Payment Cryptography utiliza identificadores de clave para representar las claves en las operaciones de API.

Sin embargo, algunas operaciones de la API de AWS Payment Cryptography exportan claves cifradas por una clave de intercambio de claves previamente compartida o asimétrica. Además, los clientes pueden usar las operaciones de la API para importar material de claves encriptadas para las claves de pago.

Todas las llamadas de la API de AWS Payment Cryptography deben firmarse y transmitirse mediante seguridad de la capa de transporte (TLS). AWS Payment Cryptography requiere versiones TLS y suites de cifrado definidas por PCI como “criptografía fuerte”. Todos los puntos de conexión del servicio admiten TLS 1.0-1.3 y TLS híbrido post-cuántico.

Para obtener más información, consulte el documento técnico Detalles criptográficos de AWS Key Management Service.

Privacidad del tráfico entre redes

AWS Payment Cryptography admite una consola de administración de AWS y un conjunto de operaciones de API que le permiten crear y administrar claves de pago y usarlas en operaciones criptográficas.

AWS Payment Cryptography admite dos opciones de conectividad de red desde su red privada a AWS.

- Una conexión de una conexión de VPN IPSec a través de Internet.
- AWS Direct Connect vincula su red interna con una ubicación de AWS Direct Connect a través de cable estándar Ethernet de fibra óptica.

Todas las llamadas de la API de Payment Cryptography deben firmarse y transmitirse mediante seguridad de la capa de transporte (TLS). Las llamadas también requieren un paquete de cifrado moderno que admita el secreto perfecto en el futuro. El tráfico a los módulos de seguridad de

hardware (HSM) que almacenan material de claves para las claves de pago solo se permite desde hosts de la API de AWS Payment Cryptography a través de la red interna de AWS.

Para conectarse directamente a la criptografía de pagos de AWS desde su nube privada virtual (VPC) sin enviar tráfico a través de la Internet pública, utilice los puntos de enlace de VPC, con tecnología de AWS. PrivateLink Para obtener más información, consulte Conexión a AWS Payment Cryptography a través de un punto de conexión de VPC.

AWS Payment Cryptography admite también una opción de intercambio híbrido postcuántico de claves para el protocolo de cifrado de red de seguridad de la capa de transporte (TLS). Puede utilizar esta opción de TLS cuando se conecte a los puntos de conexión de la API de AWS Payment Cryptography.

Resiliencia en la criptografía de pagos AWS

AWS La infraestructura global se basa en AWS regiones y zonas de disponibilidad. Las regiones proporcionan varias zonas de disponibilidad físicamente independientes y aisladas que se encuentran conectadas mediante redes con un alto nivel de rendimiento y redundancia, además de baja demora. Con las zonas de disponibilidad, puede diseñar y utilizar aplicaciones y bases de datos que realizan una conmutación por error automática entre las zonas sin interrupciones. Las zonas de disponibilidad tienen una mayor disponibilidad, tolerancia a errores y escalabilidad que las infraestructuras tradicionales de uno o varios centros de datos.

Para obtener más información sobre AWS las regiones y las zonas de disponibilidad, consulte Infraestructura AWS global.

Aislamiento regional

La criptografía de pagos de AWS es un servicio regional que está disponible en varias regiones.

El diseño regionalmente aislado de AWS Payment Cryptography garantiza que un problema de disponibilidad en una región de AWS no puede afectar al funcionamiento de AWS Payment Cryptography en ninguna otra región. AWS Payment Cryptography está diseñado para garantizar un tiempo de inactividad planificado cero, con todas las actualizaciones de software y operaciones de escalado realizadas de manera imperceptible y sin problemas.

El Acuerdo de nivel de servicio (SLA) de AWS Payment Cryptography incluye un compromiso de servicio del 99,99 % para todas las API de AWS Payment Cryptography. Para cumplir este compromiso, AWS Payment Cryptography garantiza que todos los datos y la información de

autorización necesarios para ejecutar una solicitud de la API estén disponibles en todos los hosts regionales que reciben la solicitud.

La infraestructura de AWS Payment Cryptography se replica en al menos tres zonas de disponibilidad (AZ) en cada región. Para garantizar que los errores de varios hosts no afecten al rendimiento, AWS Payment Cryptography está diseñado para atender el tráfico de clientes de cualquiera de las zonas de disponibilidad de una región.

Los cambios realizados en las propiedades o permisos de una clave de pagos se replican en todos los hosts de la región para garantizar que cualquier host de la región pueda procesar de manera correcta la solicitud posterior. Solicitudes de operaciones criptográficas mediante el uso de la clave de pagos se reenvían a una flota de módulos de seguridad de hardware (HSM) de AWS Payment Cryptography, cualquiera de los cuales puede realizar la operación con la clave de pagos.

Diseño de varios inquilinos

El diseño de varios inquilinos de AWS Payment Cryptography permite cumplir el SLA de disponibilidad y mantener altas tasas de solicitudes, al tiempo que protege la confidencialidad de las claves y los datos.

Se implementan varios mecanismos de cumplimiento de la integridad para garantizar que la clave de pagos especificada para la operación criptográfica sea siempre la que se utiliza.

El material de clave de texto sin formato para las claves de Payment Cryptography está ampliamente protegido. El material de clave se cifra en el HSM tan pronto como se crea y el material de clave cifrado se mueve de inmediato al almacenamiento seguro. La clave cifrada se recupera y se descifra dentro del HSM justo a tiempo para su uso. La clave de texto sin formato permanece en la memoria HSM solo durante el tiempo necesario para completar la operación criptográfica. El material de claves de texto sin formato nunca sale de los HSM; nunca se escribe en un almacenamiento persistente.

Para obtener más información acerca de los mecanismos que AWS Payment Cryptography utiliza para proteger sus claves, consulte Detalles criptográficos de AWS Payment Cryptography.

Seguridad de la infraestructura en AWS Payment Cryptography

Como servicio gestionado, AWS Payment Cryptography está protegido por los procedimientos de seguridad de red AWS global que se describen en el documento técnico Amazon Web Services: Overview of Security Processes.

Utiliza las llamadas a la API AWS publicadas para acceder a AWS Payment Cryptography través de la red. Los clientes deben ser compatibles con Transport Layer Security (TLS) 1.2 o una versión posterior. Los clientes también deben ser compatibles con conjuntos de cifrado con confidencialidad directa total (PFS) tales como Ephemeral Diffie-Hellman (DHE) o Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). La mayoría de los sistemas modernos como Java 7 y posteriores son compatibles con estos modos.

Además, las solicitudes deben estar firmadas mediante un ID de clave de acceso y una clave de acceso secreta que esté asociada a una entidad principal de IAM. También puede utilizar AWS Security Token Service (AWS STS) para generar credenciales de seguridad temporales para firmar solicitudes.

Aislamiento de hosts físicos

La seguridad de la infraestructura física que AWS Payment Cryptography utiliza está sujeto a los controles que se describen en la sección Seguridad Física y Ambiental de Amazon Web Services: Información general de los procesos de seguridad. Puede encontrar más detalles en los informes de conformidad y los resultados de auditoría de terceros enumerados en la sección anterior.

La criptografía de pagos de AWS es compatible con módulos de seguridad de hardware (HSM) específicos con certificación commercial-off-the-shelf PCI PTS HSM. El material de claves para las claves de AWS Payment Cryptography se almacena solo en memoria volátil en los HSM, y solo mientras la clave de Payment Cryptography está en uso. Los HSM se encuentran en bastidores de acceso controlado dentro de los centros de datos de Amazon que aplican un doble control para cualquier acceso físico. Para obtener información detallada acerca de la operación de los HSM de AWS Payment Cryptography, consulte Detalles criptográficos de AWS Payment Cryptography.

Prácticas recomendadas de seguridad para la criptografía de pagos AWS

AWS La criptografía de pagos admite muchas funciones de seguridad integradas o que puede implementar de forma opcional para mejorar la protección de sus claves de cifrado y garantizar que se utilicen para los fines previstos, incluidas las políticas de IAM, un amplio conjunto de claves condicionales de políticas para refinar sus políticas de claves y políticas de IAM y la aplicación integrada de las normas PCI PIN en relación con los bloques de claves.

> **Important**
>
> Las directrices generales proporcionadas no representan una solución de seguridad completa. Dado que no todas las mejores prácticas son adecuadas para todas las situaciones, no se pretende que sean prescriptivas.

- Uso de claves y modos de uso: La criptografía de AWS pagos sigue y aplica las restricciones de uso y uso de claves tal como se describe en la especificación de bloque de claves de intercambio seguro de claves interoperable ANSI X9 TR 31-2018 y de conformidad con el requisito de seguridad 18-3 del PIN PCI. Esto limita la capacidad de utilizar una sola clave para varios fines y vincula criptográficamente los metadatos de la clave (como las operaciones permitidas) al propio material de la clave. AWS La criptografía de pagos impone automáticamente estas restricciones, por ejemplo, no se puede utilizar una clave de cifrado clave (TR31_K0_KEY_ENCRYPTION_KEY) para descifrar datos. Consulte Comprender los atributos clave de la clave AWS de criptografía de pagos para obtener más detalles.
- Limitar el uso compartido de material de clave simétrica: sólo comparta material de claves simétricas (como claves de cifrado de pines o claves de cifrado de claves) con un máximo de otra entidad. Si es necesario transferir material confidencial a más entidades o socios, cree claves adicionales. AWS La criptografía de pagos nunca expone el material de clave simétrica ni el material de clave privada asimétrica de forma transparente.
- Utilizar alias o etiquetas para asociar claves a determinados casos de uso o socios: los alias pueden utilizarse para denotar fácilmente el caso de uso asociado a una clave, como alias/BIN_12345_CVK para denotar una clave de verificación de tarjeta asociada a BIN 12345. Para ofrecer más flexibilidad, considere la posibilidad de crear etiquetas como bin=12345, use_case=acquiring,country=us,partner=foo. Los alias y las etiquetas también se pueden utilizar para limitar el acceso como, por ejemplo, para aplicar controles de acceso entre los casos de uso emisor y adquirente.
- Practicar el acceso con privilegio mínimo: IAM puede utilizarse para limitar el acceso de producción a los sistemas en lugar de a los individuos, como prohibir a los usuarios individuales la creación de claves o la ejecución de operaciones criptográficas. IAM también puede utilizarse para limitar el acceso tanto a comandos como a claves que pueden no ser aplicables para su caso de uso, como limitar la capacidad de generar o validar pines para un adquirente. Otra forma de utilizar el acceso con privilegio mínimo es restringir las operaciones sensibles (como la importación de claves) a cuentas de servicio específicas. Para ver ejemplos, consulte AWS Ejemplos de políticas de criptografía de pagos basadas en la identidad.

Véase también

- Gestión de identidad y acceso para criptografía AWS de pagos
- Consulte Prácticas recomendadas en IAM en la Guía del usuario de IAM.

Validación de la conformidad en la Criptografía de pagos de AWS

Auditores externos evalúan la seguridad y la conformidad de la Criptografía de pagos de AWS como parte de varios programas de conformidad de AWS. Esto incluye SOC, PCI y otros.

La Criptografía de pagos de AWS ha sido evaluada para varias normas PCI además de la PCI DSS. Entre ellas se incluyen la seguridad PIN de la PCI (PIN de la PCI) y el cifrado punto a punto de la PCI (P2PE). Consulte AWS Artifact para ver los certificados y guías de cumplimiento disponibles.

Para obtener una lista de servicios de AWS en el ámbito de programas de conformidad específicos, consulte Servicios de AWS en el ámbito del programa de conformidad. Para obtener información general, consulte Programas de conformidad de AWS.

Puede descargar los informes de auditoría de terceros utilizando AWS Artifact. Para obtener más información, consulte Descarga de informes en AWS Artifact.

Su responsabilidad de conformidad al utilizar la Criptografía de pagos de AWS se determina en función de la sensibilidad de los datos, los objetivos de cumplimiento de su empresa y la legislación y los reglamentos correspondientes. AWS proporciona los siguientes recursos para ayudar con la conformidad:

- Guías de inicio rápido de seguridad y conformidad: estas guías de implementación analizan consideraciones sobre arquitectura y proporcionan los pasos para implementar los entornos de referencia centrados en la seguridad y la conformidad en AWS.
- Recursos de conformidad de AWS: este conjunto de manuales y guías podría aplicarse a su sector y ubicación.
- Evaluación de recursos con reglas en la Guía para desarrolladores de AWS Config: AWS Config evalúa en qué medida las configuraciones de sus recursos cumplen las prácticas internas, las directrices del sector y las normativas.
- AWS Security Hub: este servicio de AWS ofrece una vista integral de su estado de seguridad en AWS que le ayuda a comprobar la conformidad con las normas del sector de seguridad y las prácticas recomendadas.

Gestión de identidad y acceso para criptografía AWS de pagos

AWS Identity and Access Management (IAM) es una herramienta Servicio de AWS que ayuda al administrador a controlar de forma segura el acceso a los recursos. AWS Los administradores de IAM controlan quién puede autenticarse (iniciar sesión) y quién puede autorizarse (tener permisos) para usar los recursos de criptografía de AWS pagos. La IAM es una herramienta Servicio de AWS que puede utilizar sin coste adicional.

Temas

Público

La forma de usar AWS Identity and Access Management (IAM) varía según el trabajo que realices en criptografía de AWS pagos.

Usuario del servicio: si utiliza el servicio de criptografía de AWS pagos para realizar su trabajo, el administrador le proporcionará las credenciales y los permisos que necesita. A medida que vaya utilizando más funciones de criptografía de AWS pagos para realizar su trabajo, es posible que necesite permisos adicionales. Entender cómo se administra el acceso puede ayudarlo a solicitar los permisos correctos al administrador. Si no puede acceder a una característica en AWS Payment Cryptography, consulte Solución de problemas de AWS identidad y acceso a la criptografía de pagos.

Administrador de servicios: si estás a cargo de los recursos de criptografía de AWS pagos de tu empresa, es probable que tengas pleno acceso a la criptografía de AWS pagos. Su trabajo consiste en determinar a qué funciones y recursos AWS de criptografía de pagos deben acceder los usuarios del servicio. Luego, debe enviar solicitudes a su administrador de IAM para cambiar los permisos

de los usuarios de su servicio. Revise la información de esta página para conocer los conceptos básicos de IAM. Para obtener más información sobre cómo su empresa puede utilizar la IAM con la criptografía de AWS pagos, consulte. Cómo funciona la criptografía de AWS pagos con IAM

Administrador de IAM: si es administrador de IAM, puede que le interese obtener más información sobre cómo redactar políticas para gestionar el acceso a la criptografía de pagos. AWS Para ver ejemplos de políticas AWS de criptografía de pagos basadas en la identidad que puede utilizar en IAM, consulte. AWS Ejemplos de políticas de criptografía de pagos basadas en la identidad

Autenticación con identidades

La autenticación es la forma de iniciar sesión con sus AWS credenciales de identidad. Debe estar autenticado (con quien haya iniciado sesión AWS) como usuario de IAM o asumiendo una función de IAM. Usuario raíz de la cuenta de AWS

Puede iniciar sesión AWS como una identidad federada mediante las credenciales proporcionadas a través de una fuente de identidad. AWS IAM Identity Center Los usuarios (IAM Identity Center), la autenticación de inicio de sesión único de su empresa y sus credenciales de Google o Facebook son ejemplos de identidades federadas. Al iniciar sesión como una identidad federada, su administrador habrá configurado previamente la federación de identidades mediante roles de IAM. Cuando accedes AWS mediante la federación, asumes un rol de forma indirecta.

Según el tipo de usuario que sea, puede iniciar sesión en el portal AWS Management Console o en el de AWS acceso. Para obtener más información sobre cómo iniciar sesión AWS, consulte Cómo iniciar sesión Cuenta de AWS en su Guía del AWS Sign-In usuario.

Si accede AWS mediante programación, AWS proporciona un kit de desarrollo de software (SDK) y una interfaz de línea de comandos (CLI) para firmar criptográficamente sus solicitudes con sus credenciales. Si no utilizas AWS herramientas, debes firmar las solicitudes tú mismo. Para obtener más información sobre cómo usar el método recomendado para firmar las solicitudes usted mismo, consulte Firmar las solicitudes de la AWS API en la Guía del usuario de IAM.

Independientemente del método de autenticación que use, es posible que deba proporcionar información de seguridad adicional. Por ejemplo, le AWS recomienda que utilice la autenticación multifactor (MFA) para aumentar la seguridad de su cuenta. Para obtener más información, consulte Autenticación multifactor en la Guía del usuario de AWS IAM Identity Center y Uso de la autenticación multifactor (MFA) en AWSen la Guía del usuario de IAM.

Cuenta de AWS usuario root

Al crear una Cuenta de AWS, comienza con una identidad de inicio de sesión que tiene acceso completo a todos Servicios de AWS los recursos de la cuenta. Esta identidad se denomina usuario Cuenta de AWS raíz y se accede a ella iniciando sesión con la dirección de correo electrónico y la contraseña que utilizaste para crear la cuenta. Recomendamos encarecidamente que no utilice el usuario raíz para sus tareas diarias. Proteja las credenciales del usuario raíz y utilícelas solo para las tareas que solo el usuario raíz pueda realizar. Para ver la lista completa de las tareas que requieren que inicie sesión como usuario raíz, consulte Tareas que requieren credenciales de usuario raíz en la Guía del usuario de IAM.

Usuarios y grupos de IAM

Un usuario de IAM es una identidad propia Cuenta de AWS que tiene permisos específicos para una sola persona o aplicación. Siempre que sea posible, recomendamos emplear credenciales temporales, en lugar de crear usuarios de IAM que tengan credenciales de larga duración como contraseñas y claves de acceso. No obstante, si tiene casos de uso específicos que requieran credenciales de larga duración con usuarios de IAM, recomendamos rotar las claves de acceso. Para más información, consulte Rotar las claves de acceso periódicamente para casos de uso que requieran credenciales de larga duración en la Guía del usuario de IAM.

Un grupo de IAM es una identidad que especifica un conjunto de usuarios de IAM. No puede iniciar sesión como grupo. Puede usar los grupos para especificar permisos para varios usuarios a la vez. Los grupos facilitan la administración de los permisos de grandes conjuntos de usuarios. Por ejemplo, podría tener un grupo cuyo nombre fuese IAMAdmins y conceder permisos a dicho grupo para administrar los recursos de IAM.

Los usuarios son diferentes de los roles. Un usuario se asocia exclusivamente a una persona o aplicación, pero la intención es que cualquier usuario pueda asumir un rol que necesite. Los usuarios tienen credenciales permanentes a largo plazo y los roles proporcionan credenciales temporales. Para más información, consulte Cuándo crear un usuario de IAM (en lugar de un rol) en la Guía del usuario de IAM.

Roles de IAM

Un rol de IAM es una identidad dentro de usted Cuenta de AWS que tiene permisos específicos. Es similar a un usuario de IAM, pero no está asociado a una determinada persona. Puede asumir temporalmente una función de IAM en el AWS Management Console cambiando de función. Puede

asumir un rol llamando a una operación de AWS API AWS CLI o utilizando una URL personalizada. Para más información sobre los métodos para el uso de roles, consulte Uso de roles de IAM en la Guía del usuario de IAM.

Los roles de IAM con credenciales temporales son útiles en las siguientes situaciones:

- Acceso de usuario federado: para asignar permisos a una identidad federada, puede crear un rol y definir sus permisos. Cuando se autentica una identidad federada, se asocia la identidad al rol y se le conceden los permisos define el rol. Para obtener información acerca de roles para federación, consulte Creación de un rol para un proveedor de identidades de terceros en la Guía del usuario de IAM. Si utiliza IAM Identity Center, debe configurar un conjunto de permisos. IAM Identity Center correlaciona el conjunto de permisos con un rol en IAM para controlar a qué pueden acceder las identidades después de autenticarse. Para obtener información acerca de los conjuntos de permisos, consulte Conjuntos de permisos en la Guía del usuario de AWS Single Sign-On.
- Permisos de usuario de IAM temporales: un usuario de IAM puede asumir un rol de IAM para recibir temporalmente permisos distintos que le permitan realizar una tarea concreta.
- Acceso entre cuentas: puede utilizar un rol de IAM para permitir que alguien (una entidad principal de confianza) de otra cuenta acceda a los recursos de la cuenta. Los roles son la forma principal de conceder acceso entre cuentas. Sin embargo, con algunas Servicios de AWS, puedes adjuntar una política directamente a un recurso (en lugar de usar un rol como proxy). Para obtener información acerca de la diferencia entre los roles y las políticas basadas en recursos para el acceso entre cuentas, consulte Cómo los roles de IAM difieren de las políticas basadas en recursos en la Guía del usuario de IAM.
- Acceso entre servicios: algunos Servicios de AWS utilizan funciones en otros Servicios de AWS. Por ejemplo, cuando realiza una llamada en un servicio, es común que ese servicio ejecute aplicaciones en Amazon EC2 o almacene objetos en Amazon S3. Es posible que un servicio haga esto usando los permisos de la entidad principal, usando un rol de servicio o usando un rol vinculado al servicio.
  - Sesiones de acceso directo (FAS): cuando utilizas un usuario o un rol de IAM para realizar acciones en ellas AWS, se te considera director. Cuando utiliza algunos servicios, es posible que realice una acción que desencadene otra acción en un servicio diferente. El FAS utiliza los permisos del principal que llama Servicio de AWS y los solicita Servicio de AWS para realizar solicitudes a los servicios descendentes. Las solicitudes de FAS solo se realizan cuando un servicio recibe una solicitud que requiere interacciones con otros Servicios de AWS recursos para completarse. En este caso, debe tener permisos para realizar ambas acciones. Para

obtener información sobre las políticas a la hora de realizar solicitudes de FAS, consulte Reenviar sesiones de acceso.

- Rol de servicio: un rol de servicio es un rol de IAM que adopta un servicio para realizar acciones en su nombre. Un administrador de IAM puede crear, modificar y eliminar un rol de servicio desde IAM. Para obtener más información, consulte Creación de un rol para delegar permisos a un Servicio de AWS en la Guía del usuario de IAM.
- Función vinculada al servicio: una función vinculada a un servicio es un tipo de función de servicio que está vinculada a un. Servicio de AWS El servicio puede asumir el rol para realizar una acción en su nombre. Los roles vinculados al servicio aparecen en usted Cuenta de AWS y son propiedad del servicio. Un administrador de IAM puede ver, pero no editar, los permisos de los roles vinculados a servicios.

- Aplicaciones que se ejecutan en Amazon EC2: puede usar un rol de IAM para administrar las credenciales temporales de las aplicaciones que se ejecutan en una instancia EC2 y realizan AWS CLI solicitudes a la API. AWS Es preferible hacerlo de este modo a almacenar claves de acceso en la instancia de EC2. Para asignar un AWS rol a una instancia EC2 y ponerlo a disposición de todas sus aplicaciones, debe crear un perfil de instancia adjunto a la instancia. Un perfil de instancia contiene el rol y permite a los programas que se ejecutan en la instancia de EC2 obtener credenciales temporales. Para más información, consulte Uso de un rol de IAM para conceder permisos a aplicaciones que se ejecutan en instancias Amazon EC2 en la Guía del usuario de IAM.

Para obtener información sobre el uso de los roles de IAM, consulte Cuándo crear un rol de IAM (en lugar de un usuario) en la Guía del usuario de IAM.

Administración de acceso mediante políticas

El acceso se controla AWS creando políticas y adjuntándolas a AWS identidades o recursos. Una política es un objeto AWS que, cuando se asocia a una identidad o un recurso, define sus permisos. AWS evalúa estas políticas cuando un director (usuario, usuario raíz o sesión de rol) realiza una solicitud. Los permisos en las políticas determinan si la solicitud se permite o se deniega. La mayoría de las políticas se almacenan AWS como documentos JSON. Para obtener más información sobre la estructura y el contenido de los documentos de política JSON, consulte Información general de políticas JSON en la Guía del usuario de IAM.

Los administradores pueden usar las políticas de AWS JSON para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

De forma predeterminada, los usuarios y los roles no tienen permisos. Para conceder permiso a los usuarios para realizar acciones en los recursos que necesiten, un administrador de IAM puede crear políticas de IAM. A continuación, el administrador puede añadir las políticas de IAM a roles y los usuarios pueden asumirlos.

Las políticas de IAM definen permisos para una acción independientemente del método que se utilice para realizar la operación. Por ejemplo, suponga que dispone de una política que permite la acción `iam:GetRole`. Un usuario con esa política puede obtener información sobre el rol de la API AWS Management Console AWS CLI, la o la AWS API.

Políticas basadas en identidades

Las políticas basadas en identidad son documentos de políticas de permisos JSON que puede adjuntar a una identidad, como un usuario, un grupo de usuarios o un rol de IAM. Estas políticas controlan qué acciones pueden realizar los usuarios y los roles, en qué recursos y en qué condiciones. Para obtener más información sobre cómo crear una política basada en identidad, consulte Creación de políticas de IAM en la Guía del usuario de IAM.

Las políticas basadas en identidades pueden clasificarse además como políticas insertadas o políticas administradas. Las políticas insertadas se integran directamente en un único usuario, grupo o rol. Las políticas administradas son políticas independientes que puede adjuntar a varios usuarios, grupos y roles de su Cuenta de AWS empresa. Las políticas administradas incluyen políticas AWS administradas y políticas administradas por el cliente. Para más información sobre cómo elegir una política administrada o una política insertada, consulte Elegir entre políticas administradas y políticas insertadas en la Guía del usuario de IAM.

Políticas basadas en recursos

Las políticas basadas en recursos son documentos de política JSON que se asocian a un recurso. Ejemplos de políticas basadas en recursos son las políticas de confianza de roles de IAM y las políticas de bucket de Amazon S3. En los servicios que admiten políticas basadas en recursos, los administradores de servicios pueden utilizarlos para controlar el acceso a un recurso específico. Para el recurso al que se asocia la política, la política define qué acciones puede realizar una entidad principal especificada en ese recurso y en qué condiciones. Debe especificar una entidad principal en una política en función de recursos. Los principales pueden incluir cuentas, usuarios, roles, usuarios federados o. Servicios de AWS

Las políticas basadas en recursos son políticas insertadas que se encuentran en ese servicio. No puedes usar políticas AWS gestionadas de IAM en una política basada en recursos.

Listas de control de acceso (ACL)

Las listas de control de acceso (ACL) controlan qué entidades principales (miembros de cuentas, usuarios o roles) tienen permisos para acceder a un recurso. Las ACL son similares a las políticas basadas en recursos, aunque no utilizan el formato de documento de políticas JSON.

Amazon S3 y Amazon VPC son ejemplos de servicios que admiten las ACL. AWS WAF Para obtener más información sobre las ACL, consulte Información general de Lista de control de acceso (ACL) en la Guía para desarrolladores de Amazon Simple Storage Service.

Otros tipos de políticas

AWS admite tipos de políticas adicionales y menos comunes. Estos tipos de políticas pueden establecer el máximo de permisos que los tipos de políticas más frecuentes le conceden.

- Límites de permisos: un límite de permisos es una característica avanzada que le permite establecer los permisos máximos que una política basada en identidad puede conceder a una entidad de IAM (usuario o rol de IAM). Puede establecer un límite de permisos para una entidad. Los permisos resultantes son la intersección de las políticas basadas en la identidad de la entidad y los límites de permisos. Las políticas basadas en recursos que especifiquen el usuario o rol en el campo `Principal` no estarán restringidas por el límite de permisos. Una denegación explícita en cualquiera de estas políticas anulará el permiso. Para obtener más información sobre los límites de los permisos, consulte Límites de permisos para las entidades de IAM en la Guía del usuario de IAM.
- Políticas de control de servicios (SCP): las SCP son políticas de JSON que especifican los permisos máximos para una organización o unidad organizativa (OU). AWS Organizations AWS Organizations es un servicio para agrupar y gestionar de forma centralizada varios de los Cuentas de AWS que son propiedad de su empresa. Si habilita todas las características en una organización, entonces podrá aplicar políticas de control de servicio (SCP) a una o a todas sus cuentas. El SCP limita los permisos de las entidades en las cuentas de los miembros, incluidas las de cada una. Usuario raíz de la cuenta de AWS Para obtener más información acerca de Organizations y las SCP, consulte Funcionamiento de las SCP en la Guía del usuario de AWS Organizations.
- Políticas de sesión: las políticas de sesión son políticas avanzadas que se pasan como parámetro cuando se crea una sesión temporal mediante programación para un rol o un usuario federado. Los permisos de la sesión resultantes son la intersección de las políticas basadas en identidades del rol y las políticas de la sesión. Los permisos también pueden proceder de una política en

función de recursos. Una denegación explícita en cualquiera de estas políticas anulará el permiso. Para más información, consulte Políticas de sesión en la Guía del usuario de IAM.

Varios tipos de políticas

Cuando se aplican varios tipos de políticas a una solicitud, los permisos resultantes son más complicados de entender. Para saber cómo AWS determinar si se debe permitir una solicitud cuando se trata de varios tipos de políticas, consulte la lógica de evaluación de políticas en la Guía del usuario de IAM.

Cómo funciona la criptografía de AWS pagos con IAM

Antes de utilizar IAM para gestionar el acceso a la criptografía de AWS pagos, debe saber qué funciones de IAM están disponibles para su uso con la criptografía de pagos. AWS Para obtener una visión general de cómo funcionan la criptografía de AWS pagos y otros AWS servicios con IAM, consulte AWS Servicios que funcionan con IAM en la Guía del usuario de IAM.

Temas

- AWS Criptografía de pagos: políticas basadas en la identidad
- Autorización basada en las etiquetas de AWS Payment Cryptography

AWS Criptografía de pagos: políticas basadas en la identidad

Con las políticas basadas en identidades de IAM, puede especificar las acciones y los recursos permitidos o denegados, así como las condiciones en las que se permiten o deniegan las acciones. AWS La criptografía de pagos admite acciones, recursos y claves de condición específicos. Para obtener información sobre todos los elementos que utiliza en una política JSON, consulte Referencia de los elementos de las políticas JSON de IAM en la Guía del usuario de IAM.

Acciones

Los administradores pueden usar las políticas de AWS JSON para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

El elemento `Action` de una política JSON describe las acciones que puede utilizar para conceder o denegar el acceso en una política. Las acciones políticas suelen tener el mismo nombre que la

operación de AWS API asociada. Hay algunas excepciones, como acciones de solo permiso que no tienen una operación de API coincidente. También hay algunas operaciones que requieren varias acciones en una política. Estas acciones adicionales se denominan acciones dependientes.

Incluya acciones en una política para conceder permisos y así llevar a cabo la operación asociada.

Las acciones políticas en criptografía de AWS pagos utilizan el siguiente prefijo antes de la acción:. `payment-cryptography:` Por ejemplo, para conceder permiso a alguien para ejecutar una operación de la `VerifyCardData` API de criptografía de AWS pagos, debes incluir la `payment-cryptography:VerifyCardData` acción en su política. Las instrucciones de la política deben incluir un elemento `Action` o un elemento `NotAction`. AWS La criptografía de pagos define su propio conjunto de acciones que describen las tareas que puede realizar con este servicio.

Para especificar varias acciones en una única instrucción, sepárelas con comas del siguiente modo:

```
"Action": [
      "payment-cryptography:action1",
      "payment-cryptography:action2"
```

Puede utilizar caracteres comodín para especificar varias acciones (*). Por ejemplo, para especificar todas las acciones que comiencen con la palabra `List` (como, por ejemplo, `ListKeys` y `ListAliases`), incluya la siguiente acción:

```
"Action": "payment-cryptography:List*"
```

Para ver una lista de las acciones de criptografía de AWS pagos, consulte las acciones definidas por la criptografía de AWS pagos en la Guía del usuario de IAM.

Recursos

Los administradores pueden usar las políticas de AWS JSON para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

El elemento `Resource` de la política JSON especifica el objeto u objetos a los que se aplica la acción. Las instrucciones deben contener un elemento `Resource` o `NotResource`. Como práctica recomendada, especifique un recurso utilizando el Nombre de recurso de Amazon (ARN). Puede hacerlo para acciones que admitan un tipo de recurso específico, conocido como permisos de nivel de recurso.

Para las acciones que no admiten permisos de nivel de recurso, como las operaciones de descripción, utilice un carácter comodín (*) para indicar que la instrucción se aplica a todos los recursos.

```
"Resource": "*"
```

El recurso de clave de criptografía de pagos tiene el siguiente ARN:

```
arn:${Partition}:payment-cryptography:${Region}:${Account}:key/${keyARN}
```

Para obtener más información sobre el formato de los ARN, consulte Nombres de recursos de Amazon (ARN) y espacios de nombres de AWS servicio.

Por ejemplo, para especificar la instancia de `arn:aws:payment-cryptography:us-east-2:111122223333:key/kwapwa6qaifllw2h` en su instrucción, utilice el siguiente ARN:

```
"Resource": "arn:aws:payment-cryptography:us-east-2:111122223333:key/kwapwa6qaifllw2h"
```

Para especificar todas las claves que pertenecen a una cuenta específica, utilice el carácter comodín (*):

```
"Resource": "arn:aws:payment-cryptography:us-east-2:111122223333:key/*"
```

Algunas acciones AWS de criptografía de pagos, como las de creación de claves, no se pueden realizar en un recurso específico. En dichos casos, debe utilizar el carácter comodín (*).

```
"Resource": "*"
```

Para especificar varios recursos en una única instrucción, utilice una coma, tal y como se indica a continuación:

```
"Resource": [
      "resource1",
      "resource2"
```

Ejemplos

Para ver ejemplos de políticas de criptografía de AWS pagos basadas en la identidad, consulte. AWS Ejemplos de políticas de criptografía de pagos basadas en la identidad

Autorización basada en las etiquetas de AWS Payment Cryptography

AWS Ejemplos de políticas de criptografía de pagos basadas en la identidad

De forma predeterminada, los usuarios y los roles de IAM no tienen permiso para crear, ver ni modificar recursos de AWS Payment Cryptography. Tampoco pueden realizar tareas con la API AWS Management Console AWS CLI, o. AWS Un administrador de IAM debe crear políticas de IAM que concedan permisos a los usuarios y a los roles para realizar operaciones de la API concretas en los recursos especificados que necesiten. El administrador debe adjuntar esas políticas a los usuarios o grupos de IAM que necesiten esos permisos.

Para obtener información acerca de cómo crear una política basada en identidad de IAM con estos documentos de políticas JSON de ejemplo, consulte Creación de políticas en la pestaña JSON en la Guía del usuario de IAM.

Temas

- Prácticas recomendadas relativas a políticas
- Uso de la consola de AWS Payment Cryptography
- Permitir a los usuarios consultar sus propios permisos
- Posibilidad de acceder a todos los aspectos de la criptografía de pagos AWS
- Posibilidad de llamar a las API mediante claves específicas
- Capacidad para denegar específicamente un recurso

Prácticas recomendadas relativas a políticas

Las políticas basadas en la identidad determinan si alguien puede crear recursos de criptografía de AWS pagos de tu cuenta, acceder a ellos o eliminarlos. Estas acciones pueden generar costos adicionales para su Cuenta de AWS. Siga estas directrices y recomendaciones al crear o editar políticas basadas en identidades:

- Comience con las políticas AWS administradas y avance hacia los permisos con privilegios mínimos: para empezar a conceder permisos a sus usuarios y cargas de trabajo, utilice las

políticas AWS administradas que otorgan permisos para muchos casos de uso comunes. Están disponibles en su. Cuenta de AWS Le recomendamos que reduzca aún más los permisos definiendo políticas administradas por el AWS cliente que sean específicas para sus casos de uso. Con el fin de obtener más información, consulte las políticas administradas por AWS o las políticas administradas por AWS para funciones de trabajo en la Guía de usuario de IAM.

- Aplique permisos de privilegio mínimo: cuando establezca permisos con políticas de IAM, conceda solo los permisos necesarios para realizar una tarea. Para ello, debe definir las acciones que se pueden llevar a cabo en determinados recursos en condiciones específicas, también conocidos como permisos de privilegios mínimos. Con el fin de obtener más información sobre el uso de IAM para aplicar permisos, consulte Políticas y permisos en IAM en la Guía del usuario de IAM.
- Utilice condiciones en las políticas de IAM para restringir aún más el acceso: puede agregar una condición a sus políticas para limitar el acceso a las acciones y los recursos. Por ejemplo, puede escribir una condición de políticas para especificar que todas las solicitudes deben enviarse utilizando SSL. También puedes usar condiciones para conceder el acceso a las acciones del servicio si se utilizan a través de una acción específica Servicio de AWS, por ejemplo AWS CloudFormation. Para obtener más información, consulte Elementos de la política de JSON de IAM: Condición en la Guía del usuario de IAM.
- Utilice el analizador de acceso de IAM para validar las políticas de IAM con el fin de garantizar la seguridad y funcionalidad de los permisos: el analizador de acceso de IAM valida políticas nuevas y existentes para que respeten el lenguaje (JSON) de las políticas de IAM y las prácticas recomendadas de IAM. El analizador de acceso de IAM proporciona más de 100 verificaciones de políticas y recomendaciones procesables para ayudar a crear políticas seguras y funcionales. Para más información, consulte Política de validación de Analizador de acceso de IAM en la Guía de usuario de IAM.
- Requerir autenticación multifactor (MFA): si tiene un escenario que requiere usuarios de IAM o un usuario raíz en Cuenta de AWS su cuenta, active la MFA para mayor seguridad. Para solicitar la MFA cuando se invocan las operaciones de la API, agregue las condiciones de la MFA a sus políticas. Para más información, consulte Configuración del acceso a una API protegido por MFA en la Guía de usuario de IAM.

Para obtener más información sobre las prácticas recomendadas de IAM, consulte las Prácticas recomendadas de seguridad en IAM en la Guía del usuario de IAM.

Uso de la consola de AWS Payment Cryptography

Para acceder a la consola AWS de criptografía de pagos, debe tener un conjunto mínimo de permisos. Estos permisos deben permitirle enumerar y ver detalles sobre los recursos de criptografía de AWS pagos de su AWS cuenta. Si crea una política basada en identidad que sea más restrictiva que el mínimo de permisos necesarios, la consola no funcionará del modo esperado para las entidades (usuarios o roles de IAM) que tengan esa política.

Para garantizar que esas entidades puedan seguir utilizando la consola de criptografía de AWS pagos, adjunte también la siguiente política de AWS gestión a las entidades. Para obtener más información, consulte Adición de permisos a un usuario en la Guía del usuario de IAM.

No es necesario conceder permisos mínimos de consola a los usuarios que solo realicen llamadas a la AWS API AWS CLI o a la misma. En su lugar, permite acceso únicamente a las acciones que coincidan con la operación de API que intenta realizar.

Permitir a los usuarios consultar sus propios permisos

En este ejemplo, se muestra cómo podría crear una política que permita a los usuarios de IAM ver las políticas administradas e insertadas que se asocian a la identidad de sus usuarios. Esta política incluye permisos para completar esta acción en la consola o mediante programación mediante la API AWS CLI o AWS .

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
```

```
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Posibilidad de acceder a todos los aspectos de la criptografía de pagos AWS

> **Warning**
>
> Este ejemplo proporciona permisos amplios y no se recomienda. En su lugar, considere los modelos de acceso con menos privilegios.

En este ejemplo, desea conceder a un usuario de IAM de su AWS cuenta acceso a todas sus claves de criptografía de AWS pagos y la posibilidad de acceder a todas las API de criptografía de AWS pagos, incluidas ambas operaciones. ControlPlane DataPlane

```
  {
     "Version": "2012-10-17",
     "Statement": [
           {
              "Effect": "Allow",
              "Action": [
                 "payment-cryptography:*"
              ],
              "Resource": [
                 "*"
              ]
           }
```

```
      ]
   }
```

Posibilidad de llamar a las API mediante claves específicas

En este ejemplo, quiere conceder a un usuario de IAM de su AWS cuenta acceso a una de sus claves de criptografía de AWS pagos y, a continuación, utilizar este recurso en dos API, `arn:aws:payment-cryptography:us-east-2:111122223333:key/kwapwa6qaifllw2h` y. `GenerateCardData` `VerifyCardData` Por el contrario, el usuario de IAM no tendrá acceso para usar esta clave en otras operaciones, como `DeleteKey` o `ExportKey`

Los recursos pueden ser claves con el prefijo `key` o alias con el prefijo `alias`.

```
   {
      "Version": "2012-10-17",
      "Statement": [
            {
               "Effect": "Allow",
               "Action": [
                  "payment-cryptography:VerifyCardData",
                  "payment-cryptography:GenerateCardData"
               ],
               "Resource": [
                  "arn:aws:payment-cryptography:us-east-2:111122223333:key/
kwapwa6qaifllw2h"
               ]
            }
      ]
   }
```

Capacidad para denegar específicamente un recurso

> **Warning**
>
> Considere detenidamente las implicaciones de conceder un acceso comodín. En su lugar, considere un modelo de privilegio mínimo.

En este ejemplo, quiere permitir que un usuario de IAM de su AWS cuenta acceda a cualquiera de sus claves de criptografía de AWS pagos, pero quiere denegar los permisos a una clave específica.

El usuario tendrá acceso a `VerifyCardData` y `GenerateCardData` con todas las claves a excepción de la especificada en la instrucción de denegación.

```
  {
     "Version": "2012-10-17",
     "Statement": [
         {
             "Effect": "Allow",
             "Action": [
                 "payment-cryptography:VerifyCardData",
                 "payment-cryptography:GenerateCardData"
             ],
             "Resource": [
                 "arn:aws:payment-cryptography:us-east-2:111122223333:key/*"
             ]
         },
         {
             "Effect": "Deny",
             "Action": [
                 "payment-cryptography:GenerateCardData"
             ],
             "Resource": [
                 "arn:aws:payment-cryptography:us-east-2:111122223333:key/
arn:aws:payment-cryptography:us-east-2:111122223333:key/kwapwa6qaifllw2h"
             ]
         }
     ]
 }
```

Solución de problemas de AWS identidad y acceso a la criptografía de pagos

Se añadirán temas a esta sección a medida que se identifiquen los problemas relacionados con la IAM específicos de la criptografía AWS de pagos. Para obtener información general sobre la solución de problemas relacionados con la IAM, consulte la sección de solución de problemas de la Guía del usuario de IAM.

Supervisión de la Criptografía de pagos de AWS

La monitorización es una parte importante del mantenimiento de la fiabilidad, la disponibilidad y el rendimiento de la Criptografía de pagos de AWS y de las demás soluciones de AWS. AWS ofrece las siguientes herramientas de monitorización para vigilar la Criptografía de pagos de AWS, informar cuando algo no funciona y realizar acciones automáticas cuando proceda:

- Amazon CloudWatch monitorea los recursos de AWS y las aplicaciones que ejecuta en AWS en tiempo real. Puede recopilar métricas y realizar un seguimiento de las métricas, crear paneles personalizados y definir alarmas que le advierten o que toman medidas cuando una métrica determinada alcanza el umbral que se especifique. Por ejemplo, puede hacer que CloudWatch haga un seguimiento del uso de la CPU u otras métricas de las instancias de Amazon EC2 y lanzar nuevas instancias automáticamente cuando sea necesario. Para obtener más información, consulte la Guía del usuario de Amazon CloudWatch.
- Registros de Amazon CloudWatch le permite monitorear, almacenar y tener acceso a los archivos de registro desde instancias de Amazon EC2, CloudTrail u otras fuentes. CloudWatch Logs puede monitorear información en los registros y enviarle una notificación cuando se llega a determinados umbrales. También se pueden archivar los datos de los registros en un almacenamiento de larga duración. Para obtener más información, consulte la Guía del usuario de Amazon CloudWatch Logs.
- Amazon EventBridge puede utilizarse para automatizar los servicios de AWS y responder automáticamente a eventos del sistema, como problemas de disponibilidad de las aplicaciones o cambios en los recursos. Los eventos de los servicios de AWS se envían a EventBridge casi en tiempo real. Puede crear reglas sencillas para indicar qué eventos le resultan de interés, así como qué acciones automatizadas se van a realizar cuando un evento cumple una de las reglas. Para obtener más información, consulte la Guía del usuario de Amazon EventBridge.
- AWS CloudTrail captura llamadas a la API y eventos relacionados efectuados por su cuenta de AWS o en su nombre, y entrega los archivos de registro al bucket de Amazon S3 que se haya especificado. También pueden identificar qué usuarios y cuentas llamaron a AWS, la dirección IP de origen de las llamadas y el momento en que se hicieron. Para obtener más información, consulte la Guía del usuario de AWS CloudTrail.

> **Note**
>
> Los registros de AWS CloudTrail se admiten para las operaciones del plano de control, como CreateKey, pero no para las operaciones del plano de datos, como la Generación de datos de tarjetas

Registro de llamadas de la API de Criptografía de pagos de AWS mediante AWS CloudTrail

La Criptografía de pagos de AWS se integra con AWS CloudTrail, un servicio que proporciona un registro de las medidas adoptadas por un usuario, un rol o un servicio de AWS en Criptografía de pagos de AWS. CloudTrail captura las llamadas a la API de Criptografía de pagos de AWS como eventos. Las llamadas capturadas incluyen las llamadas desde la consola de la Criptología de pagos de AWS y las llamadas desde el código a las operaciones de la API de la Criptografía de pagos de AWS. Si crea un registro de seguimiento, puede habilitar la entrega continua de eventos de CloudTrail a un bucket de Amazon S3, incluidos los eventos para la Criptografía de pagos de AWS. Si no configura un registro de seguimiento, puede ver los eventos más recientes de la consola de CloudTrail en el Historial de eventos. Mediante la información que recopila CloudTrail, puede determinar la solicitud que se hizo a Criptografía de pagos de AWS, la dirección IP desde la que se hizo dicha solicitud, quién la hizo y cuándo, además de información adicional.

Para obtener más información acerca de CloudTrail, consulte la Guía del usuario de AWS CloudTrail.

> **Note**
>
> Actualmente, la integración con Cloudtrail solo se admite para las operaciones del plano de control.

Información sobre la Criptografía de pagos de AWS en CloudTrail

CloudTrail se habilita en su cuenta de AWS cuando la crea. Cuando se produce actividad en la Criptografía de pagos de AWS, esta se registra en un evento de CloudTrail junto con otros eventos de servicio de AWS en el Historial de eventos. Puede ver, buscar y descargar los últimos eventos de

la cuenta de AWS. Para obtener más información, consulte Ver eventos con el historial de eventos de CloudTrail.

Para mantener un registro continuo de los eventos de la cuenta de AWS, incluidos los eventos de la Criptografía de pagos de AWS, cree un registro de seguimiento. Un registro de seguimiento permite a CloudTrail enviar archivos de registro a un bucket de Amazon S3. De manera predeterminada, cuando se crea un registro de seguimiento en la consola, el registro de seguimiento se aplica a todas las regiones de AWS. El registro de seguimiento registra los eventos de todas las regiones de la partición de AWS y envía los archivos de registro al bucket de Amazon S3 especificado. También es posible configurar otros servicios de AWS para analizar en profundidad y actuar en función de los datos de eventos recopilados en los registros de CloudTrail. Para obtener más información, consulte los siguientes temas:

- Introducción a la creación de registros de seguimiento
- Servicios e integraciones compatibles con CloudTrail
- Configuración de notificaciones de Amazon SNS para CloudTrail
- Recepción de archivos de registro de CloudTrail de varias regiones
- Recepción de archivos de registro de CloudTrail de varias cuentas

Registros CloudTrail AWS Operaciones de Criptografía de pagos, como CreateKey, ImportKey, DeleteKey, ListKeys, TagResource, y todas las demás operaciones del plano de control.

Cada entrada de registro o evento contiene información sobre quién generó la solicitud. La información de identidad del usuario le ayuda a determinar lo siguiente:

- Si la solicitud se realizó con credenciales de usuario AWS Identity and Access Management (IAM) o credenciales de usuario raíz.
- Si la solicitud se realizó con credenciales de seguridad temporales de un rol o fue un usuario federado.
- Si la solicitud la realizó otro servicio de AWS.

Para obtener más información, consulte el elemento userIdentity de CloudTrail.

Comprender las entradas del archivo de registro de la Criptografía de pagos de AWS

Un registro de seguimiento es una configuración que permite la entrega de eventos como archivos de registros en un bucket de Amazon S3 que especifique. Los archivos log de CloudTrail pueden contener una o varias entradas de log. Un evento representa una solicitud específica realizada desde un origen y contiene información sobre la acción solicitada, la fecha y la hora de la acción, los parámetros de la solicitud, etc. Los archivos de registro de CloudTrail no rastrean el orden en la pila de las llamadas públicas a la API, por lo que estas no aparecen en ningún orden específico.

En el siguiente ejemplo, se muestra una entrada de registro de CloudTrail que ilustra la acción `CreateKey` de la Criptografía de pagos de AWS.

```
      {
    CloudTrailEvent: {
      tlsDetails= {
        TlsDetails: {
          cipherSuite=TLS_AES_128_GCM_SHA256,
          tlsVersion=TLSv1.3,
          clientProvidedHostHeader=pdx80.controlplane.paymentcryptography.us-
west-2.amazonaws.com
        }
      },
    requestParameters=CreateKeyInput (
      keyAttributes=KeyAttributes(
        KeyUsage=TR31_B0_BASE_DERIVATION_KEY,
        keyClass=SYMMETRIC_KEY,
        keyAlgorithm=AES_128,
        keyModesOfUse=KeyModesOfUse(
          encrypt=false,
          decrypt=false,
          wrap=false
          unwrap=false,
          generate=false,
          sign=false,
          verify=false,
          deriveKey=true,
          noRestrictions=false)
        ),
      keyCheckValueAlgorithm=null,
      exportable=true,
```

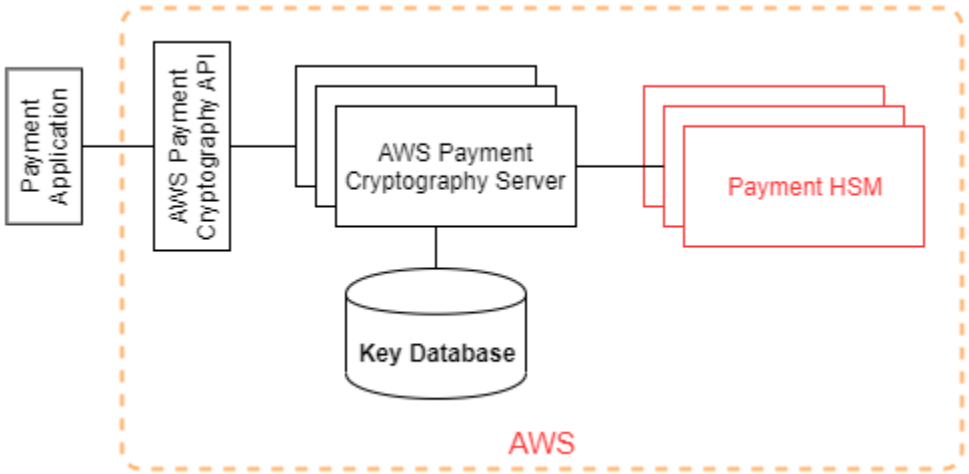
```
      enabled=true,
      tags=null),
    eventName=CreateKey,
    userAgent=Coral/Apache-HttpClient5,
    responseElements=CreateKeyOutput(
      key=Key(
        keyArn=arn:aws:payment-cryptography:us-
east-2:111122223333:key/5rplquuwozodpwsp,
          keyAttributes=KeyAttributes(
            KeyUsage=TR31_B0_BASE_DERIVATION_KEY,
            keyClass=SYMMETRIC_KEY,
            keyAlgorithm=AES_128,
            keyModesOfUse=KeyModesOfUse(
              encrypt=false,
              decrypt=false,
              wrap=false,
              unwrap=false,
              generate=false,
              sign=false,
              verify=false,
              deriveKey=true,
              noRestrictions=false)
            ),
          keyCheckValue=FE23D3,
          keyCheckValueAlgorithm=ANSI_X9_24,
          enabled=true,
          exportable=true,
          keyState=CREATE_COMPLETE,
          keyOrigin=AWS_PAYMENT_CRYPTOGRAPHY,
          createTimestamp=Sun May 21 18:58:32 UTC 2023,
          usageStartTimestamp=Sun May 21 18:58:32 UTC 2023,
          usageStopTimestamp=null,
          deletePendingTimestamp=null,
          deleteTimestamp=null)
        ),
      sourceIPAddress=192.158.1.38,
        userIdentity={
          UserIdentity: {
            arn=arn:aws:sts::111122223333:assumed-role/TestAssumeRole-us-west-2-PDX80/
ControlPlane-IntegTest-68211a2a-3e9d-42b7-86ac-c682520e0410,
            invokedBy=null,
            accessKeyId=,
            type=AssumedRole,
            sessionContext={
```

```
              SessionContext: {
                sessionIssuer={
                  SessionIssuer: {arn=arn:aws:iam::111122223333:role/TestAssumeRole-us-
west-2-PDX80,
                  type=Role,
                  accountId=111122223333,
                  userName=TestAssumeRole-us-west-2-PDX80,
                  principalId=}
                },
                attributes={
                  SessionContextAttributes: {
                    creationDate=Sun May 21 18:58:31 UTC 2023,
                    mfaAuthenticated=false
                  }
                },
                webIdFederationData=null
              }
            },
          username=null,
          principalId=:ControlPlane-User,
          accountId=111122223333,
          identityProvider=null
        }
      },
      eventTime=Sun May 21 18:58:32 UTC 2023,
      managementEvent=true,
      recipientAccountId=111122223333,
      awsRegion=us-west-2,
      requestID=151cdd67-4321-1234-9999-dce10d45c92e,
      eventVersion=1.08, eventType=AwsApiCall,
      readOnly=false,
      eventID=c69e3101-eac2-1b4d-b942-019919ad2faf,
      eventSource=payment-cryptography.amazonaws.com,
      eventCategory=Management,
      additionalEventData={
    }
  }
}
```

Detalles criptográficos

La Criptografía de pagos de AWS proporciona una interfaz web para generar y administrar claves criptográficas para las transacciones de pago. AWS La Criptografía de pagos ofrece servicios estándar de administración de claves y criptografía de transacciones de pago y herramientas que puede utilizar para la administración y la auditoría centralizadas. Esta documentación proporciona una descripción detallada de las operaciones criptográficas que puede utilizar en la Criptografía de pagos de AWS para ayudarle a evaluar las funciones que ofrece el servicio.

La Criptografía de pagos de AWS contiene varias interfaces (incluida una API de RESTful, a través de AWS CLI, AWS SDK y AWS Management Console) para solicitar operaciones criptográficas de una flota distribuida de módulos de seguridad de hardware validados por HSM PCI PTS.



La Criptografía de pagos de AWS es un servicio en niveles que consiste en hosts de la Criptografía de pagos de AWS orientados a la web y un nivel de HSM. La agrupación de estos hosts en niveles forma la pila de Criptografía de pagos de AWS. Todas las solicitudes a la Criptografía de pagos de AWS deben realizarse a través del protocolo de Transport Layer Security (TLS) y finalizar en un host de la Criptografía de pagos de AWS. Los hosts del servicio solo permiten el uso de TLS con un conjunto de cifrado que proporciona una confidencialidad total. El servicio autentifica y autoriza sus solicitudes mediante los mismos mecanismos de credenciales y políticas de IAM que están disponibles para todas las demás operaciones de la API de AWS.

Los servidores de Criptografía de pagos de AWS se conectan al HSM subyacente a través de una red privada no virtual. Las conexiones entre los componentes del servicio y el HSM están protegidas con TLS mutuos (mTLS) para la autenticación y el cifrado.

Objetivos de diseño

La Criptografía de pagos de AWS está diseñado para cumplir los siguientes requisitos:

- Confiable: el uso de las claves está protegido por las políticas de control de acceso que usted define y administra. No existe ningún mecanismo para exportar claves de Criptografía de pagos de AWS en texto plano. La confidencialidad de las claves criptográficas es crucial. Se requieren varios empleados de Amazon con acceso específico a roles para realizar los controles de acceso basados en quórum con el fin de llevar a cabo acciones administrativas en los HSM. Ningún empleado de Amazon tiene acceso a las claves principales (o maestras) ni a las copias de seguridad de HSM. Las claves principales no se pueden sincronizar con los HSM que no formen parte de una región de la Criptografía de pagos de AWS. Todas las demás claves están protegidas por las claves principales de HSM. Por lo tanto, las claves de la Criptografía de pagos de AWS de los clientes no se pueden utilizar fuera del servicio de Criptografía de pagos de AWS que opera en la cuenta del cliente.
- Baja latencia y alto rendimiento: la Criptografía de pagos de AWS proporciona operaciones criptográficas a nivel de latencia y rendimiento adecuadas para gestionar claves criptográficas de pago y procesar transacciones de pago.
- Durabilidad: La durabilidad de las claves criptográficas está diseñada para igualar a la de los servicios de mayor durabilidad en AWS. Una única clave criptográfica puede compartirse con un terminal de pago, una tarjeta con chip EMV u otro dispositivo criptográfico seguro (DSC) que se utilice durante muchos años.
- Regiones independientes: AWS proporciona regiones independientes para los clientes que necesiten restringir el acceso a los datos en diferentes regiones o necesiten cumplir requisitos de residencia de datos. El uso de claves se puede aislar dentro de una Región de AWS.
- Fuente segura de números aleatorios: debido a que la criptografía rigurosa depende de la generación de números aleatorios verdaderamente impredecibles, la Criptografía de pagos de AWS proporciona una fuente de alta calidad y con validación de números aleatorios. Toda la generación de claves para la Criptografía de pagos de AWS utiliza un HSM con certificación PCI PTS HSM y funciona en modo PCI.
- Auditoría: la Criptografía de pagos de AWS registra el uso y la gestión de las claves criptográficas en los registros de CloudTrail y en los registros de servicio disponibles a través de Amazon CloudWatch. Puede utilizar los registros de CloudTrail para inspeccionar el uso de sus claves criptográficas, incluido el uso de claves por parte de cuentas con las que haya compartido claves. AWS La criptografía de pagos es auditada por asesores externos según los estándares PCI

aplicables, la marca de la tarjeta y la seguridad de pagos regionales. Las Certificaciones y las guías de Responsabilidad compartida están disponibles en AWS Artifact.

- Elástico: la Criptografía de pagos de AWS se escala horizontalmente en función de su demanda. En lugar de predecir y reservar la capacidad de HSM, la Criptografía de pagos de AWS ofrece criptografía de pagos bajo demanda. AWS La Criptografía de pagos asume la responsabilidad de mantener la seguridad y el cumplimiento de HSM para proporcionar la capacidad suficiente para satisfacer los picos de demanda de los clientes.

Principios básicos

Los temas de este capítulo describen las primitivas criptográficas de la criptografía de AWS pagos y dónde se utilizan. También presentan los elementos básicos del servicio.

Temas

Primitivas criptográficas

AWS La criptografía de pagos utiliza algoritmos criptográficos estándar parametrizables para que las aplicaciones puedan implementar los algoritmos necesarios para su caso de uso. El conjunto de algoritmos criptográficos está definido por las normas PCI, ANSI X9, EMVco e ISO. Toda la criptografía se realiza mediante HSM normalizados de PCI PTS HSM que funcionan en modo PCI.

Entropía y generación de números aleatorios

AWS La generación de claves de criptografía de pagos se realiza en los HSM de criptografía de pagos. AWS Los HSM implementan un generador de números al azar que cumple con el requisito PCI PTS HSM para todos los tipos y parámetros de clave compatibles.

Operaciones de clave simétrica

Se admiten los algoritmos de clave simétrica y los valores de clave definidos en ANSI X9 TR 31, ANSI X9.24 y PCI PIN Anexo C:

- Funciones hash: algoritmos de las familias SHA2 y SHA3 con un tamaño de salida superior a 2551. Excepto para la retrocompatibilidad con terminales POI v3 anteriores al PTS PCI.
- Cifrado y descifrado: AES con tamaño de clave mayor o igual a 128 bits, o TDEA con tamaño de clave mayor o igual a 112 bits (2 claves o 3 claves).
- Códigos de autenticación de mensajes (Message Authentication Codes, MAC) CMAC o GMAC con AES, así como HMAC con una función hash aprobada y un tamaño de clave mayor o igual a 128.

AWS La criptografía de pagos utiliza el AES 256 para las claves principales del HSM, las claves de protección de datos y las claves de sesión TLS.

Operaciones con claves asimétricas

Se admiten los algoritmos de clave asimétrica y los valores de clave definidos en ANSI X9 TR 31, ANSI X9.24 y PCI PIN Anexo C:

- Esquemas de establecimiento de claves aprobados: como se describe en NIST SP800-56A (acuerdo de claves basado en ECC/FCC2), NIST SP800-56B (acuerdo de claves basado en IFC) y NIST SP800-38F (cifrado/envoltura de claves basado en AES).

AWS Los servidores de criptografía de pago solo permiten las conexiones al servicio mediante TLS con un conjunto de cifrado que proporciona un secreto total.

Almacenamiento de claves

AWS Las claves de criptografía de pago están protegidas por las claves principales del HSM AES 256 y se almacenan en bloques de claves ANSI X9 TR 31 en una base de datos cifrada. La base de datos se replica en la base de datos en memoria de los servidores de criptografía de pagos. AWS

Según el Anexo C de la Normativa de seguridad PIN de la PCI, las claves AES 256 son igual o más fuertes que:

- TDEA de 3 claves
- RSA de 15360 bits
- ECC de 512 bits
- DSA, DH y MQV 15360/512

Importación de claves simétricas

AWS La criptografía de pagos permite la importación de criptogramas y bloques de claves con claves simétricas o públicas con una clave de cifrado de clave simétrica (KEK) igual o más segura que la clave protegida para la importación.

Importación de claves con claves asimétricas

AWS La criptografía de pagos permite la importación de criptogramas y bloques de claves con claves simétricas o públicas protegidas por una clave de cifrado de clave privada (KEK) igual o más segura que la clave protegida para la importación. La clave pública proporcionada para el descifrado debe tener su autenticidad e integridad garantizadas por un certificado de una autoridad de confianza del cliente.

Las KEK públicas que proporciona AWS Payment Cryptography cuentan con la protección de autenticación e integridad de una autoridad de certificación (CA) que certifica el cumplimiento de las normas PCI PIN Security y del anexo A de la PCI P2PE.

Exportación de claves

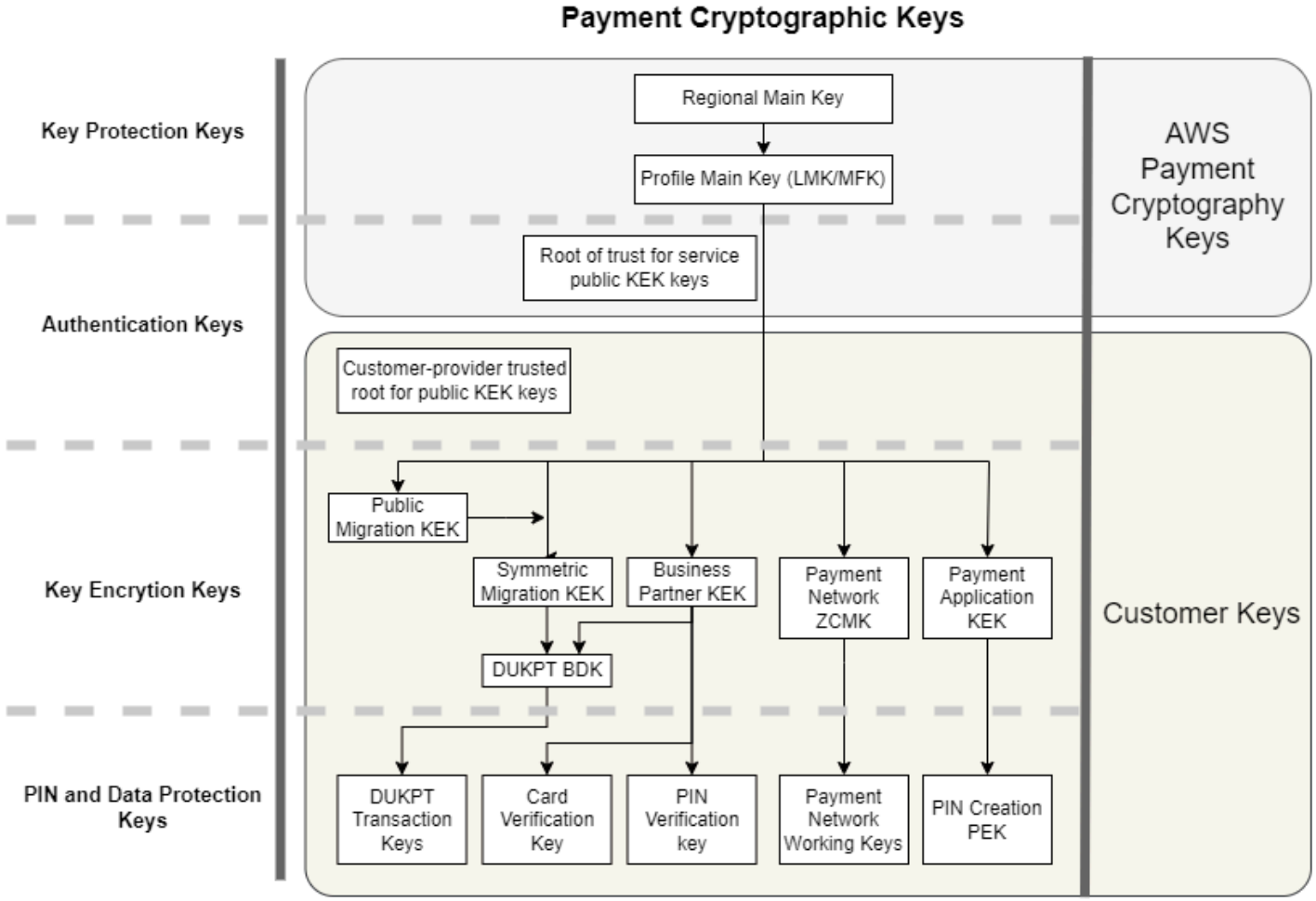
Las claves se pueden exportar y proteger con las claves adecuadas KeyUsage y que sean igual o más seguras que la clave que se va a exportar.

Protocolo de clave única derivada por transacción (DUKPT)

AWS La criptografía de pagos es compatible con las claves de derivación base (BDK) TDEA y AES, tal como se describe en la norma ANSI X9.24-3.

Jerarquía de claves

La jerarquía de claves de criptografía de AWS pagos garantiza que las claves estén siempre protegidas por claves tan sólidas o más sólidas que las claves que protegen.



AWS Las claves de criptografía de pago se utilizan para la protección de las claves dentro del servicio:

Clave	Descripción
Clave principal regional	Protege las imágenes virtuales del HSM, o perfiles, utilizadas para el procesamiento criptográfico. Esta clave sólo existe en el HSM y en las copias de seguridad.

Clave	Descripción
Clave principal de perfil	Clave de protección de claves de cliente de nivel superior, tradicionalmente denominad a Clave maestra local (Local Master Key, LMK) o Clave maestra de archivo (Master File Key, MFK) para las claves de cliente. Esta clave sólo existe en el HSM y en las copias de seguridad. Los perfiles definen distintas configuraciones del HSM según lo requieran las normas de seguridad para los casos de uso de pagos.
Raíz de confianza para las claves de cifrado de clave pública (KEK) de criptografía de AWS pagos	La clave pública raíz de confianza y el certifica do para autenticar y validar las claves públicas proporcionados por AWS Payment Cryptogra phy para la importación y exportación de claves mediante claves asimétricas.

Las claves de cliente se agrupan por claves utilizadas para proteger otras claves y claves que protegen datos relacionados con el pago. Estos son ejemplos de claves de cliente de ambos tipos:

Clave	Descripción
Raíz de confianza proporcionada por el cliente para claves públicas KEK	Clave pública y certificado proporcionados por usted como raíz de confianza para autenticar y validar las claves públicas que usted suministr a para la importación y exportación de claves utilizando claves asimétricas.
Claves de cifrado de claves (Key Encryption Key, KEK)	Las KEK se utilizan únicamente para cifrar otras claves para su intercambio entre almacenes de claves externos y AWS Payment Cryptography, socios comerciales, redes de pago o diferentes aplicaciones de su organizac ión.

Clave	Descripción
Clave derivada única por transacción (Derived Unique Key Per Transaction, DUKPT) clave derivada base (Base Derivation Key, BDK)	Las BDK se utilizan para crear claves únicas para cada terminal de pago y traducir las transacciones de múltiples terminales a una única clave de trabajo del banco adquirent e, o adquirente. La mejor práctica, exigida por el Cifrado punto a punto (P2PE) de la PCI, es que se utilicen BDK diferentes para distintos modelos de terminal, servicios de inyección o inicialización de claves u otro tipo de segmentación para limitar el impacto de comprometer una BDK.
Clave maestra de control de zona (Zone Control Master Key, ZCMK) de la red de pagos	Las ZCMK, también denominadas claves de zona o claves maestras de zona, son proporcio nadas por las redes de pago para establecer claves de trabajo iniciales.
Claves de transacción DUKPT	Los terminales de pago configurados para DUKPT obtienen una clave única para el terminal y la transacción. El HSM que recibe la transacción puede determinar la clave a partir del identificador del terminal y del número de secuencia de la transacción.
Claves de preparación de datos de tarjetas	Las claves maestras del emisor EMV, las claves y valores de verificación de la tarjeta EMV y las claves de protección del archivo de datos de personalización de la tarjeta se utilizan para crear datos de tarjetas individua les para su uso por un proveedor de personali zación de tarjetas. Estas claves y los datos criptográficos de validación también son utilizados por los bancos emisores, o emisores, para autenticar los datos de las tarjetas como parte de la autorización de las transacciones.

Clave	Descripción
Claves de preparación de datos de tarjetas	Las claves maestras del emisor EMV, las claves y valores de verificación de la tarjeta EMV y las claves de protección del archivo de datos de personalización de la tarjeta se utilizan para crear datos de tarjetas individua les para su uso por un proveedor de personali zación de tarjetas. Estas claves y los datos criptográficos de validación también son utilizados por los bancos emisores, o emisores, para autenticar los datos de las tarjetas como parte de la autorización de las transacciones.
Claves de trabajo de la red de pago	A menudo denominadas clave de trabajo del emisor o clave de trabajo del adquirente, son las claves que cifran las transacciones enviadas a las redes de pago o recibidas de ellas. Estas claves son rotadas con frecuencia por la red, a menudo diariamente o cada hora. Son las claves de cifrado del PIN (PEK) para las transacciones con PIN/débito.
Claves de cifrado del número de identificación personal (Personal Identification Number, PIN) (Personal Encryption Keys, PEK)	Las aplicaciones que crean o descifran bloques de PIN utilizan PEK para evitar el almacenam iento o la transmisión del PIN en texto claro.

Operaciones internas

Este tema describe los requisitos internos implementados por el servicio para asegurar las claves de los clientes y las operaciones criptográficas para un servicio de criptografía de pagos y gestión de claves distribuido y escalable a nivel mundial.

Especificaciones y ciclo de vida del HSM{

AWS La criptografía de pagos utiliza una flota de HSM disponibles en el mercado. Los HSM cuentan con la validación FIPS 140-2 de nivel 3 y también utilizan las versiones de firmware y la política

de seguridad que figuran en la lista de dispositivos PTS PCI aprobados por el consejo de normas de seguridad de la PCI{ut} como cumplimiento de la norma PCI HSM v3. La norma PCI PTS HSM incluye requisitos adicionales para la fabricación, el envío, la implementación, la gestión y la destrucción del hardware HSM que son importantes para la seguridad y el cumplimiento de los pagos pero que no están contemplados en FIPS 140.

Todos los HSM funcionan en modo PCI y están configurados con la política de seguridad PCI PTS HSM. Solo están habilitadas las funciones necesarias para admitir los casos de uso de la criptografía de AWS pagos. AWS La criptografía de pagos no permite imprimir, mostrar ni devolver los PIN de texto no cifrado.

Seguridad física del dispositivo HSM

El servicio solo puede utilizar los HSM que tengan claves de dispositivo firmadas por una autoridad certificadora (CA) de criptografía de AWS pagos por el fabricante antes de la entrega. La criptografía AWS de pagos es una entidad de certificación secundaria de la entidad emisora de certificados del fabricante y constituye la base de confianza de los certificados de fabricantes y dispositivos de los HSM. La CA del fabricante implementa la norma ANSI TR 34 y ha certificado el cumplimiento del anexo A de seguridad del PCI PIN y del anexo A de la PCI P2PE. El fabricante verifica que todos los HSM con claves de dispositivo firmadas por la CA de criptografía de AWS pagos se envíen al receptor designado de AWS.

Tal y como exige la seguridad PCI PIN, el fabricante suministra una lista de números de serie a través de un canal de comunicación distinto al del envío del HSM. Estos números de serie se comprueban en cada paso del proceso de instalación del HSM en los centros de datos de AWS. Por último, los operadores AWS de criptografía de pago validan la lista de HSM instalados comparándola con la lista del fabricante antes de añadir el número de serie a la lista de HSM autorizados a recibir claves de criptografía de pago. AWS

Los HSM están almacenados de forma segura o bajo doble control en todo momento, lo que incluye:

- El envío desde el fabricante a una instalación de montaje en bastidor de AWS.
- Durante el montaje en bastidor.
- Envío desde la instalación de montaje en bastidor a un centro de datos.
- Recepción e instalación en una sala de procesamiento segura de un centro de datos. Los bastidores HSM aplican un doble control con cerraduras de acceso controlado por tarjeta, sensores de puerta con alarma y cámaras.
- Durante las operaciones.

- Durante el desmantelamiento y la destrucción.

Para cada chain-of-custody HSM se mantiene y supervisa un sistema completo, con responsabilidad individual.

Inicialización de HSM

Un HSM solo se inicializa como parte del conjunto de criptografía de AWS pagos después de validar su identidad e integridad mediante los números de serie, las claves de dispositivo instaladas por el fabricante y la suma de verificación del firmware. Una vez validada la autenticidad e integridad de un HSM, se configura, incluyendo la habilitación del Modo PCI. A continuación, se establecen las claves principales de la región de criptografía de AWS pagos y las claves principales del perfil, y el HSM queda a disposición del servicio.

Servicio y reparación del HSM

Los HSM tienen componentes reparables que no requieren la violación del límite criptográfico del dispositivo. Estos componentes incluyen ventiladores de refrigeración, fuentes de alimentación y baterías. Si un HSM u otro dispositivo dentro del bastidor de HSM necesita servicio, se mantiene el control dual durante todo el periodo en que el bastidor está abierto.

Desactivación de HSM

El desmantelamiento se produce debido a un HSM end-of-life o a un fallo de éste. Los HSM se ponen lógicamente a cero antes de retirarlos de su rack, si son funcionales, y luego se destruyen dentro de las salas de procesamiento seguras de los centros de datos de AWS. Nunca se devuelven al fabricante para su reparación, ni se utilizan para otro fin, ni se sacan de otro modo de una sala de procesamiento segura antes de su destrucción.

Actualización del firmware del HSM

Las actualizaciones del firmware del HSM se aplican cuando es necesario mantener la alineación con las versiones incluidas en la lista PCI PTS HSM y FIPS 140-2 (o FIPS 140-3), si se trata de una actualización relacionada con la seguridad o si se determina que los clientes pueden beneficiarse de las funciones de una nueva versión. AWS Los HSM de criptografía de pagos utilizan un firmware que coincide con las versiones incluidas en el PCI PTS HSM. off-the-shelf Se valida la integridad de las nuevas versiones de firmware con las versiones de firmware certificadas PCI o FIPS y, a continuación, se comprueba su funcionalidad antes de implantarlas en todos los HSM.

Acceso del operador

Los operadores pueden tener acceso no consular a los HSM para la resolución de problemas en los raros casos en que la información recopilada de los HSM durante las operaciones normales sea insuficiente para identificar un problema o planificar un cambio. Se ejecutan los siguientes pasos:

- Se desarrollan y aprueban las actividades de resolución de problemas y se programa la sesión no consular.
- Se retira un HSM del servicio de procesamiento del cliente.
- Se eliminan las claves principales, bajo doble control.
- Se permite al operador el acceso no consular al HSM para realizar las actividades de resolución de problemas aprobadas, bajo control dual.
  - Tras la finalización de la sesión no consular, se realiza el proceso de aprovisionamiento inicial en el HSM, devolviendo el firmware y la configuración estándar y sincronizando la clave principal, antes de devolver el HSM al servicio de los clientes.
  - Los registros de la sesión se graban en el seguimiento de cambios.
  - La información obtenida de la sesión se utiliza para planificar futuros cambios.

Todos los registros de acceso ajenos a la consola se revisan para comprobar si cumplen con los procesos y si se producen posibles cambios en la supervisión del HSM, el proceso de gestión o la formación de los operadores non-console-access .

Administración de claves

Todos los HSM de una región se sincronizan con una Clave principal de región. Una Clave Principal de Región protege al menos una Clave principal de perfil. Una Clave Principal de Perfil protege claves de cliente.

Todas las claves principales son generadas por un HSM y distribuidas a mediante la distribución de claves simétricas utilizando técnicas asimétricas, alineadas con ANSI X9 TR 34 y PCI PIN Anexo A.

Temas

- Sincronización de la clave principal de perfil
- Rotación de la clave principal del perfil
- Protección
- Durabilidad
- Seguridad de las comunicaciones
- Gestión de las claves de los clientes
- Registro y monitorización

Generación

Las claves principales AES de 256 bits se generan en uno de los HSM aprovisionados para la flota de HSM de servicio, utilizando el generador de números al azar PCI PTS HSM.

Sincronización de la clave principal de región

Las claves principales de la región HSM son sincronizadas por el servicio en toda la flota regional con mecanismos definidos por ANSI X9 TR-34, que incluyen:

- Autenticación mutua mediante claves del host de distribución de claves (Key Distribution Host, KDH) y del dispositivo receptor de claves (Key Receiving Device, KRD) y certificados para proporcionar autenticación e integridad de las claves públicas.
- Los certificados están firmados por una autoridad de certificación (CA) que cumple los requisitos del Anexo A2 del PIN de la PCI, excepto para los algoritmos asimétricos y las intensidades de clave apropiadas para proteger las claves AES de 256 bits.
- Identificación y protección de claves para las claves simétricas distribuidas coherentes con ANSI X9 TR-34 y PCI PIN Anexo A1, excepto para los algoritmos asimétricos y las intensidades de clave apropiadas para proteger las claves AES de 256 bits.

Las claves principales de región se establecen para los HSM que han sido autenticados y aprovisionados para una región por:

- Se genera una clave principal en un HSM de la región. Ese HSM se designa como host de distribución de claves.
- Todos los HSM aprovisionados en la región generan tokens de autenticación KRD, que contienen la clave pública del HSM e información de autenticación no reproducible.

- Los tokens KRD se añaden a la lista de permitidos del KDH después de que éste valide la identidad y el permiso del HSM para recibir claves.
- El KDH produce un token de clave principal autenticable para cada HSM. Los tokens contienen la información de autenticación del KDH y la clave principal cifrada que sólo se puede cargar en el HSM para el que se ha creado.
- A cada HSM se le envía el token de clave principal creado para él. Tras validar la información de autenticación propia del HSM y la información de autenticación del KDH, la clave principal se descifra mediante la clave privada del KRD y se carga en la clave principal.

En caso de que un único HSM deba volver a sincronizarse con una región:

- Se vuelve a validar y se aprovisiona con firmware y configuración.
- Si es nuevo en la región:
  - El HSM genera un token de autenticación KRD.
  - El KDH añade el token a su lista de permitidos.
  - El KDH genera un token de clave principal para el HSM.
  - El HSM carga la clave principal.
  - El HSM se pone a disposición del servicio.

Esto garantiza que:

- Solo los HSM validados para el procesamiento AWS de criptografía de pagos en una región pueden recibir la clave maestra de esa región.
- Solo se puede distribuir una clave maestra de un HSM de criptografía de AWS pagos a un HSM de la flota.

Rotación de la clave principal de la región

Las claves principales de región se rotan al expirar el periodo de criptografía, en el improbable caso de que se sospeche que la clave está comprometida, o tras cambios en el servicio que se determine que afectan a la seguridad de la clave.

Se genera y distribuye una nueva clave principal de región como en la provisión inicial. Las claves principales de perfil guardadas deben traducirse a la nueva clave principal de región.

La rotación de la clave principal de región no afecta al procesamiento del cliente.

Sincronización de la clave principal de perfil

Las claves principales de perfil están protegidas por las claves principales de región. Esto restringe un perfil a una región específica.

Las claves principales de perfil se aprovisionan en consecuencia:

- Se genera una clave principal de perfil en un HSM que tenga sincronizada la clave principal de región.
- La clave principal del perfil se almacena y encripta con la configuración del perfil y otros contextos.
- El perfil es utilizado para las funciones criptográficas del cliente por cualquier HSM de la región con la clave principal de la región.

Rotación de la clave principal del perfil

Las claves principales de perfil se rotan al expirar el periodo criptográfico, tras sospecha de compromiso de la clave o tras cambios en el servicio que se determine que afectan a la seguridad de la clave.

Pasos de la rotación:

- Se genera una nueva clave principal de perfil y se distribuye como clave principal pendiente al igual que con la provisión inicial.
- Un proceso en segundo plano traduce el material de la clave de cliente de la clave principal de perfil establecida a la clave principal pendiente.
- Cuando todas las claves de cliente se han cifrado con la clave pendiente, ésta se promueve a clave principal de perfil.
- Un proceso en segundo plano borra el material de las claves de cliente protegido por la clave pendiente.

La rotación de la clave principal del perfil no afecta al procesamiento de los clientes.

Protección

Las claves sólo dependen de la jerarquía de claves para su protección. La protección de las claves principales es fundamental para evitar la pérdida o el compromiso de todas las claves de cliente.

Las claves principales de región son restaurables desde la copia de seguridad sólo para HSM autenticados y aprovisionados para el servicio. Estas claves sólo pueden almacenarse como tokens de clave principal mutuamente autenticables y cifrados de un KDH específico para un HSM específico.

Las claves principales de perfil se almacenan con la configuración del perfil y la información de contexto encriptada por región.

Las claves de cliente se almacenan en bloques de claves, protegidos por una clave maestra de perfil.

Todas las claves existen exclusivamente dentro de un HSM o se almacenan protegidas por otra clave de igual o mayor fuerza criptográfica.

Durabilidad

Las claves de cliente para la criptografía de las transacciones y las funciones empresariales deben estar disponibles incluso en situaciones extremas que suelen provocar interrupciones. AWS La criptografía de pagos utiliza un modelo de redundancia de varios niveles en todas las zonas y regiones de disponibilidad. AWS Los clientes que requieran una mayor disponibilidad y durabilidad para las operaciones criptográficas de pago que la proporcionada por el servicio deberán implementar arquitecturas multirregión.

Los tokens de autenticación del HSM y de la clave principal se guardan y pueden utilizarse para restaurar una clave principal o sincronizarse con una nueva clave principal, en caso de que deba restablecerse un HSM. Los tokens se archivan y sólo se utilizan bajo doble control cuando es necesario.

Seguridad de las comunicaciones

Externo

AWS Los terminales de la API de criptografía de pagos cumplen con los estándares de AWS seguridad, como el TLS igual o superior a 1.2 y la versión 4 de Signature para la autenticación e integridad de las solicitudes.

Las conexiones TLS entrantes se terminan en equilibradores de carga de red y se reenvían a los gestores de API a través de conexiones TLS internas.

Interno

Las comunicaciones internas entre componentes de servicio y entre componentes de servicio y otro servicio de AWS están protegidas por TLS utilizando criptografía fuerte.

Los HSM se encuentran en una red privada no virtual a la que sólo se puede acceder desde los componentes de servicio. Todas las conexiones entre los HSM y los componentes de servicio están protegidas con TLS mutuo (mTLS), igual o superior a TLS 1.2. Los certificados internos para TLS y mTLS son administrados por el Gestor de certificación de Amazon mediante una Autoridad de certificación privada de AWS. Las VPC internas y la red HSM se monitorizan para detectar actividades no aceptadas y cambios de configuración.

Gestión de las claves de los clientes

En AWS, la confianza de los clientes es nuestra principal prioridad. Usted mantiene el control total de las claves que cargue o cree en el servicio bajo su cuenta de AWS y la responsabilidad de configurar el acceso a las claves.

AWS La criptografía de pagos es totalmente responsable de la conformidad física del HSM y de la gestión de claves de las claves gestionadas por el servicio. Esto requiere la propiedad y administración de las claves principales de HSM y el almacenamiento de las claves de los clientes protegidas en la base de datos de claves de criptografía AWS de pagos.

Separación del espacio de claves del cliente

AWS La criptografía de pagos aplica políticas clave para todos los usos de las claves, incluida la limitación del capital a la cuenta propietaria de la clave, a menos que la clave se comparta explícitamente con otra cuenta.

Copia de seguridad y recuperación

Las claves y la información sobre claves de una región se respaldan en archivos cifrados por AWS. Los archivos requieren un doble control para restaurarlos. AWS

Bloques de claves

Todas las claves se almacenan en bloques de claves con formato ANSI X9 TR-31.

Las claves se pueden importar al servicio desde criptogramas u otros formatos de bloques de claves compatibles ImportKey con. Del mismo modo, las claves pueden exportarse, si son exportables, a otros formatos de bloques de claves o criptogramas soportados por perfiles de exportación de claves.

Uso de claves

El uso de claves está restringido a lo configurado KeyUsage por el servicio. El servicio fallará cualquier solicitud con un uso de clave, modo de uso o algoritmo inapropiados para la operación criptográfica solicitada.

Relaciones de intercambio de claves

PCI PIN Security y PCI P2PE exigen que las organizaciones que comparten claves que cifran PIN, incluida la KEK utilizada para compartir esas claves, no compartan esas claves con ninguna otra organización. Es una práctica recomendada que las claves simétricas se compartan sólo entre 2 partes, incluso dentro de la misma organización. Esto minimiza el impacto de presuntos compromisos de claves que obliguen a reemplazar las claves afectadas.

Incluso los casos empresariales que requieren compartir claves entre más de 2 partes, deben mantener el número de partes al mínimo.

AWS La criptografía de pagos proporciona etiquetas clave que se pueden usar para rastrear y hacer cumplir el uso de las claves dentro de esos requisitos.

Por ejemplo, KEK y BDK para diferentes instalaciones de inyección de claves pueden identificarse estableciendo un "KIF"="POSStation" para todas las claves compartidas con ese proveedor de servicios. Otro ejemplo sería etiquetar las claves compartidas con las redes de pago con «Network» = «PayCard». El etiquetado le permite crear controles de acceso y crear informes de auditoría para hacer cumplir y demostrar sus prácticas de gestión de claves.

Eliminación de claves

DeleteKey marca las claves de la base de datos para eliminarlas después de un período configurable por el cliente. Transcurrido este periodo, la llave se borra irremediablemente. Se trata de un mecanismo de seguridad para evitar el borrado accidental o malintencionado de una clave. Las claves marcadas para su eliminación no están disponibles para ninguna acción excepto: RestoreKey

Las llaves borradas permanecen en las copias de seguridad del servicio durante 7 días después de su borrado. No son restaurables durante este periodo.

Las claves pertenecientes a cuentas AWS cerradas se marcan para su borrado. Si la cuenta se reactiva antes de que se alcance el periodo de borrado, las claves marcadas para borrado se restauran, pero se desactivan. Deberá volver a habilitarlas para poder utilizarlas en operaciones criptográficas.

Uso compartido de claves

Las claves pueden compartirse con otras cuentas dentro o fuera de su organización utilizando el Gestor de acceso a recursos de AWS (https://docs.aws.amazon.com/ARG/index.html). Las claves pueden agruparse en un recurso compartido y luego compartirse con una cuenta o con usuarios

y roles específicos de IAM dentro de una cuenta. Usted especifica los permisos de uso para cada recurso compartido. Los permisos de uso compartido están restringidos por una política de recursos de claves. Una clave compartida no permitirá una acción restringida por su propia política. El permiso de uso compartido puede retirarse en cualquier momento.

Registro y monitorización

Los registros de servicios internos incluyen:

- CloudTrail registros de las llamadas al servicio de AWS realizadas por el servicio
- CloudWatch registros de ambos eventos registrados directamente en los CloudWatch registros o eventos de HSM
- Archivos de registro de HSM y sistemas de servicio
- Archivos de registro

Todas las fuentes de registros controlan y filtran la información confidencial, incluida la relativa a las claves. Los registros se revisan sistemáticamente para garantizar que no contienen información sensible de los clientes.

El acceso a los registros está restringido a las personas necesarias para completar las funciones de su trabajo.

Todos los registros se retienen de acuerdo con las políticas de conservación de registros de AWS.

Operaciones de clientes

AWS La criptografía de pagos es totalmente responsable del cumplimiento físico del HSM según las normas PCI. El servicio también proporciona un almacén de claves seguro y garantiza que las claves sólo puedan utilizarse para los fines permitidos por las normas PCI y especificados por usted durante su creación o importación. Usted es responsable de configurar los atributos clave y el acceso para aprovechar las capacidades de seguridad y cumplimiento del servicio.

Temas

- Rotar claves de

Generación de claves

Al crear las claves, debe establecer los atributos que el servicio utiliza para hacer cumplir el uso de la clave:

- Longitud de algoritmo y clave
- Uso
- Disponibilidad y vencimiento

Las etiquetas que se utilizan para el control de acceso basado en atributos (ABAC) se utilizan para limitar las claves para su uso con socios o aplicaciones específicos. Asegúrese de incluir políticas para limitar los roles permitidos para borrar o cambiar etiquetas.

Debe asegurarse de que las políticas que determinan los roles que pueden utilizar y gestionar la clave se establecen antes de la creación de la clave.

> **Note**
>
> Las políticas de IAM relativas a los CreateKey comandos se pueden utilizar para imponer y demostrar el doble control en la generación de claves.

Importación de claves

Al importar claves, el servicio establece los atributos para imponer el uso conforme de la clave utilizando la información criptográficamente vinculada del bloque de claves. El mecanismo para establecer el contexto de clave fundamental consiste en utilizar bloques de claves creados con el HSM de origen y protegidos por un KEK compartido o asimétrico. Esto se ajusta a los requisitos del PCI PIN y preserva el uso, el algoritmo y la solidez de las claves de la aplicación de origen.

En el momento de la importación, se deben establecer los atributos clave, las etiquetas y las políticas de control de acceso importantes, además de la información del bloque clave.

La importación de claves mediante criptogramas no transfiere los atributos clave de la aplicación de origen. Debe configurar los atributos de forma adecuada utilizando este mecanismo.

A menudo, las claves se intercambian utilizando componentes de texto claro, se transmiten por los custodios de las claves y, a continuación, se cargan con la ceremonia que implementa el doble control en una sala segura. Esto no es compatible directamente con la criptografía de AWS pagos. La API exportará una clave pública con un certificado que puede ser importado por su propio HSM para exportar un bloque de claves que sea importable por el servicio. Esto permite utilizar su propio HSM para cargar componentes de texto no cifrado.

Debe utilizar los valores de comprobación de claves (KCV) para verificar que las claves importadas coinciden con las claves de origen.

Las políticas de IAM de la ImportKey API se pueden utilizar para aplicar y demostrar un doble control en la importación de claves.

Exportación de claves

Para compartir claves con socios o aplicaciones en las instalaciones, es posible que sea necesario exportar las claves. El uso de bloques clave para las exportaciones mantiene el contexto fundamental de las claves con el material de las claves cifradas.

Las etiquetas de las claves pueden utilizarse para limitar la exportación de claves a KEK que compartan la misma etiqueta y el mismo valor.

AWS La criptografía de pagos no proporciona ni muestra un texto claro sobre los componentes clave. Esto requiere el acceso directo de los custodios de claves a los dispositivos criptográficos seguros (SCD) probados por la PCI PTS HSM o la ISO 13491 para su visualización o impresión. Puede establecer una KEK asimétrica o una KEK simétrica con su SCD para llevar a cabo la ceremonia de creación de componentes clave de texto claro bajo doble control.

Los valores de comprobación clave (Key Check Values, KCV) se deben utilizar para comprobar que los valores importados por el HSM de destino coinciden con las claves de origen.

Eliminación de claves

Puede usar la API de eliminación de claves para programar la eliminación de las claves tras el período de tiempo que usted configure. Antes de ese momento, las claves se pueden recuperar. Una vez que se eliminan las claves, se eliminan permanentemente del servicio.

Las políticas de IAM de la DeleteKey API se pueden utilizar para aplicar y demostrar un doble control a la hora de eliminar las claves.

Rotar claves de

El efecto de la rotación de claves puede implementarse utilizando alias de claves creando o importando una nueva clave y modificando después el alias de claves para que haga referencia a la clave nueva. La clave anterior se eliminaría o deshabilitaría, según sus prácticas de administración.

Cuotas para AWS Payment Cryptography

La cuenta de AWS tiene cuotas predeterminadas para cada servicios de AWS (estas cuotas se denominaban anteriormente "límites"). A menos que se indique otra cosa, cada cuota es específica de la región. Puede solicitar el aumento de algunas cuotas, pero otras no se pueden aumentar.

Nombre	Valor predeterminado	Ajusta	Descripción
Alias	Cada región admitida: 2000	Sí	El número máximo de alias que puede tener en esta cuenta en la Región actual.
Tasa combinada de solicitudes de plano de control	Cada región admitida: 5 por segundo	Sí	El número máximo de solicitudes de plano de control por segundo que puede realizar en esta cuenta en la Región actual. Esta cuota se aplica a todas las operaciones del plano de control combinadas.
Tasa combinada de solicitudes de planos de datos (asimétrica)	Cada región admitida: 20 por segundo	Sí	El número máximo de solicitudes por segundo para operaciones en el plano de datos con una clave asimétric a que puede realizar en esta cuenta en la Región actual. Esta cuota se aplica a todas las operaciones del plano de datos combinadas.

Nombre	Valor predeterminado	Ajusta	Descripción
Tasa combinada de solicitudes de planos de datos (simétrica)	Cada región admitida: 500 por segundo	Sí	El número máximo de solicitudes por segundo para operaciones de plano de datos con una clave simétrica que puede realizar en esta cuenta en la Región actual. Esta cuota se aplica a todas las operaciones del plano de datos combinadas.
Claves	Cada región admitida: 2000	Sí	El número máximo de claves que puede tener en esta cuenta en la Región actual, excluyendo las claves eliminadas.

Historial documental de la Guía del usuario de la Criptografía de pagos de AWS

En la siguiente tabla se describen las versiones de la documentación de la Criptografía de pagos de AWS.

Cambio	Descripción	Fecha
Lanzamiento de funciones	Se agregó información sobre las nuevas funciones relacionadas con la importación y exportación de claves mediante RSA y la exportación de claves DUKPT IPEK/IK.	15 de enero de 2024
Versión inicial	Publicación inicial de la Guía del usuario de la Criptografía de pagos de AWS	8 de junio de 2023

Las traducciones son generadas a través de traducción automática. En caso de conflicto entre la traducción y la version original de inglés, prevalecerá la version en inglés.