# Prácticas recomendadas de AWS Key Management Service

*Abril de 2017*

# Avisos

Este documento se suministra únicamente con fines informativos. Representa la oferta actual de productos y prácticas de AWS a partir de la fecha de publicación de este documento. Dichas prácticas y productos pueden modificarse sin previo aviso. Los clientes son responsables de realizar sus propias evaluaciones independientes de la información contenida en este documento y de cualquier uso de los productos o servicios de AWS, cada uno de los cuales se ofrece “tal cual”, sin garantía de ningún tipo, ya sea explícita o implícita. Este documento no genera ninguna garantía, declaración, compromiso contractual, condición ni certeza por parte de AWS, sus filiales, proveedores o licenciantes. Las responsabilidades y obligaciones de AWS respecto de sus clientes se controlan mediante los acuerdos de AWS y este documento no forma parte ni modifica ningún acuerdo entre AWS y sus clientes.

# Contenido

# Resumen

AWS Key Management Service (AWS KMS) es un servicio administrado que le permite concentrarse en las necesidades criptográficas de sus aplicaciones mientras que Amazon Web Services (AWS) administra la disponibilidad, la seguridad física, el control de acceso lógico y el mantenimiento de la infraestructura subyacente. Además, AWS KMS le permite auditar el uso de las claves por medio de registros de todas las llamadas a API realizadas en ellos para ayudarle a cumplir con los requisitos normativos y de conformidad.

Los clientes quieren saber cómo implementar AWS KMS de manera efectiva en su entorno. Este documento técnico explica cómo utilizar AWS KMS para cada capacidad descrita en el documento técnico de Perspectiva de seguridad del Marco de adopción de la nube de AWS (CAF), incluidas las diferencias entre los diversos tipos de claves maestras de cliente, el uso las políticas de claves de AWS KMS para garantizar privilegios mínimos, la auditoría del uso de las claves y la enumeración de algunos casos de uso que funcionan para proteger información confidencial dentro de AWS.

# Introducción

AWS Key Management Service (AWS KMS) es un servicio administrado que le permite crear y controlar fácilmente las claves de cifrado usadas para cifrar sus datos. AWS KMS usa los módulos de seguridad de hardware (HSM) para proteger la seguridad de sus claves[1]. Puede utilizar AWS KMS para proteger sus datos en los servicios de AWS y en sus aplicaciones. El documento técnico Detalles criptográficos de AWS Key Management Service describe el diseño y los controles realizados dentro del servicio para garantizar la seguridad y privacidad de sus datos[2].

El documento técnico Marco de adopción de la nube de AWS (CAF) proporciona una guía para coordinar las diferentes partes de las organizaciones que se trasladan a la informática en la nube.[3] La guía de AWS CAF se divide en áreas de enfoque que son pertinentes para la implementación de los sistemas de TI basados en la nube, que llamamos *perspectivas*. El documento técnico Perspectivas de seguridad de CAF organiza los principios que ayudarán a la transformación de la seguridad de su organización a través de cinco capacidades principales: Identity and Access Management, control de detección, seguridad de la infraestructura, protección de datos y respuesta ante incidentes[4].

Para cada capacidad de la perspectiva de seguridad de CAF, este documento técnico proporciona información detallada sobre cómo su organización debería utilizar AWS KMS para proteger información confidencial en una serie de distintos casos de uso y los medios para medir el progreso:

- **Identity and Access Management:** Permite crear varios mecanismos de control de acceso y administrar los permisos para cada uno de ellos.
- **Controles de detección:** Ofrece la capacidad de registro nativo y visibilidad en el servicio.
- **Seguridad de la infraestructura:** Proporciona la capacidad de configurar los controles de seguridad para que cumplan con sus requisitos.
- **Protección de los datos:** Ofrece la capacidad de mantener visibilidad y control sobre los datos.
- **Respuesta frente a incidencias:** Proporciona la capacidad de responder, administrar, reducir el daño y restaurar las operaciones durante y después de una incidencia.

# Identity and Access Management

La capacidad de Identity and Access Management proporciona una guía sobre cómo determinar los controles para la administración del acceso dentro de AWS KMS para proteger su infraestructura de acuerdo con las prácticas recomendadas establecidas y políticas internas.

# AWS KMS y políticas de IAM

Puede utilizar las políticas de AWS Identity and Access Management (IAM) junto con las políticas de claves para controlar el acceso a sus claves maestras de cliente (CMK) en AWS KMS. En esta sección se explica cómo utilizar IAM en el contexto de AWS KMS. No proporciona información detallada sobre el servicio de IAM. Para ver la documentación completa de IAM, consulte la Guía del usuario de AWS IAM[5].

Las políticas asociadas a las identidades de IAM (es decir, usuarios, grupos y roles) se denominan *políticas basadas en la identidad* (o *políticas de IAM*). Las políticas asociadas a los recursos externos a IAM se denominan *políticas basadas en los recursos*. En AWS KMS, debe asociar las políticas basadas en los recursos a sus claves maestras de cliente (CMK). Se denominan *políticas de claves*. Todas las KMS CMK tienen una política de claves y la debe utilizar para controlar el acceso a una CMK. Las políticas de IAM por sí solas no son suficientes para permitir el acceso a una CMK, aunque puede utilizarlas junto con una política de claves de CMK. Para ello, asegúrese de que la política de claves de CMK incluya la declaración de la política que habilita las políticas de IAM[6].

Al utilizar una política de IAM basada en la identidad, puede aplicar privilegios mínimos mediante la concesión de acceso pormenorizado a las llamadas a API de KMS dentro de una cuenta de AWS. Recuerde que las políticas de IAM se basan en una política de permiso denegado por defecto, a menos que conceda permisos de forma explícita a una entidad principal para realizar una acción.

# Políticas de claves

Las políticas de claves son la forma principal de controlar el acceso a las CMK en AWS KMS. Cada CMK tiene una política de claves asociada que define los permisos sobre el uso y la administración de la clave. La política predeterminada habilita a cualquier entidad principal que defina, así como también habilita al usuario raíz de la cuenta para que añada las políticas de IAM que remiten a la clave. Le recomendamos que edite la política CMK predeterminada para que coincida con las prácticas recomendadas de su organización para los privilegios mínimos. Para obtener acceso a un recurso cifrado, la entidad principal debe tener permisos para utilizar el recurso y para usar la clave de cifrado que protege el recurso. Si la entidad principal no tiene los permisos necesarios para cualquiera de esas acciones, se denegará la solicitud para utilizar el recurso cifrado.

También es posible restringir una CMK de modo que solo la puedan utilizar los servicios de AWS específicos mediante el uso de la declaración condicional dentro de la política de claves de CMK.`kms: ViaService` Para obtener más información, consulte la Guía para desarrolladores de AWS KMS[7].

Para crear y utilizar un volumen de Amazon Elastic Block Store (EBS), necesita permisos para utilizar Amazon EBS. La política de claves asociada a la CMK tendría que incluir un texto similar al siguiente:

```
{
   "Sid": "Allow for use of this Key",
   "Effect": "Allow",
   "Principal": {
     "AWS": "arn:aws:iam::111122223333:role/UserRole"
   },
   "Action": [
     "kms: GenerateDataKeyWithoutPlaintext",
     "kms:Decrypt"
   ],
   "Resource": "*"
 },
 {
   "Sid": "Allow for EC2 Use",
   "Effect": "Allow",
   "Principal": {
     "AWS": "arn:aws:iam::111122223333:role/UserRole"
   },
   "Action": [
     "kms:CreateGrant",
     "kms:ListGrants",
     "kms:RevokeGrant"
   ],
   "Resource": "*",
   "Condition": {
     "StringEquals": {
       "kms:ViaService": "ec2.us-west-2.amazonaws.com"
     }
   }
 }
```

En esta política de CMK, la primera declaración le proporciona a una entidad principal de IAM específica la posibilidad de generar una clave de datos y descifrarla de la CMK cuando sea necesario. Estas dos API son necesarias para cifrar el volumen de EBS mientras está asociado a una instancia de Amazon Elastic Compute Cloud (EC2).

La segunda declaración en esta política le proporciona a la entidad principal de IAM específica la posibilidad de crear, enumerar y revocar concesiones para Amazon EC2. Las concesiones se utilizan para delegar un subconjunto de permisos a los servicios de AWS, o a otras entidades principales, de modo que puedan utilizar las claves en su nombre. En este caso, la política de condición garantiza de forma explícita que solo Amazon EC2 puede utilizar las concesiones. Amazon EC2 las utilizará para volver a asociar un volumen de EBS cifrado de vuelta a una instancia si el volumen se separa a causa de una interrupción planificada o no planificada. Estos eventos se registrarán en AWS CloudTrail siempre y cuando sucedan para su auditoría.

A la hora de desarrollar una política de CMK, debe tener en cuenta cómo se evalúan las declaraciones de la política en AWS. Esto significa que si ha activado IAM para ayudar a controlar el acceso a una CMK, cuando AWS evalúa si una acción permitida se permite o se deniega, la política de CMK se combina con la política de IAM. Además, debe asegurarse de que el uso y la administración de una clave estén limitados a las partes necesarias.

### Separación de obligaciones/privilegios mínimos

Las políticas de claves especifican un recurso, acción, efecto, entidad principal y condiciones para conceder acceso a las CMK. Las políticas de claves le permiten incorporar permisos más pormenorizados para que las CMK apliquen privilegios mínimos. Por ejemplo, una aplicación podría realizar una llamada a API de KMS para cifrar datos; sin embargo, no existe ningún caso de uso para esa misma aplicación para descifrar datos. En ese caso de uso, una política de claves podría conceder acceso a la acción *kms:Encrypt* pero no a *kms:Decrypt* y disminuir las chances de una exposición. Además, AWS le permite separar los permisos de uso de los de administración asociados a la clave. Esto significa que una persona puede tener la capacidad de manipular la política de claves, pero podría no tener los permisos necesarios para utilizar la clave para las funciones criptográficas.

Dado que sus CMK se utilizan para proteger su información confidencial, debe trabajar con el fin de asegurarse de que las políticas de claves correspondientes sigan un modelo de privilegios mínimos. Esto abarca garantizar que **NO** incluye permisos kms:* en una política de IAM. Esta política concedería a la entidad principal los permisos administrativos y de uso de todas las CMK a las que tiene acceso. Del mismo modo, incluir los permisos kms:* para las entidades principales dentro de su política de claves les proporciona permisos administrativos y de uso sobre la CMK.

Es importante recordar que las políticas de denegación explícita tienen prioridad por sobre las políticas de denegación implícita. Cuando se utiliza NotPrincipal en la misma declaración de política que "Effect: Deny", los permisos especificados en la declaración de la política se deniegan de forma explícita para todas las entidades principales, *salvo* las especificadas. Una política KMS de nivel superior puede denegar de forma explícita el acceso a prácticamente todas las operaciones de KMS, salvo los roles que realmente las necesiten. Esta técnica ayuda a impedir que los usuarios no autorizados se concedan acceso a KMS.

## Uso compartido de claves entre cuentas

La delegación de los permisos a una CMK dentro de AWS KMS puede ocurrir cuando incluye la entidad principal raíz de una cuenta de confianza dentro de la política de claves de CMK. De este modo, la cuenta de confianza tiene la capacidad de delegar estos permisos a los roles y usuarios de IAM dentro de su propia cuenta por medio de las políticas de IAM. Si bien este enfoque puede simplificar la administración de la política de claves, también se basa en las cuentas de confianza para asegurarse de que los permisos delegados estén correctamente administrados. El otro enfoque sería administrar los permisos de forma explícita para todos los usuarios autorizados únicamente mediante la política de claves de KMS, que, a su vez, podría

hacer que la política de claves sea compleja y más difícil de administrar. Independientemente del enfoque que adopte, la confianza específica debería separarse por clave para garantizar que se atenga al modelo de privilegios mínimos.

## Concesiones de CMK

Los cambios de la política de claves siguen el mismo modelo de permisos que se usa para la edición de políticas en otras partes de AWS. Es decir, los usuarios tienen o no permiso para cambiar la política de claves. Los usuarios con el permiso `PutKeyPolicy` para una CMK pueden sustituir por completo la política de claves para una CMK por una política de claves diferente que elijan. Puede utilizar políticas de claves para permitir que otras entidades principales obtengan acceso a una CMK, pero las políticas de claves funcionan mejor para las asignaciones de permisos relativamente estáticas. Para facilitar una administración de permisos más pormenorizada, puede utilizar concesiones. Las concesiones son útiles cuando desea definir permisos temporales más enfocados para que otras entidades principales utilicen su CMK en su nombre cuando usted no realiza una llamada directa a la API.

Es importante tener en cuenta las concesiones por clave y los límites de las concesiones para una entidad principal por clave cuando diseña aplicaciones que utilizan concesiones para controlar el acceso a las claves. Asegúrese de que la entidad principal de retirada elimine la concesión después de utilizarla para evitar alcanzar estos límites.

## Contexto de cifrado

Además de limitar el permiso para las API de AWS KMS, AWS KMS también le ofrece la posibilidad de añadir una capa adicional de autenticación para sus llamadas a API de KMS a través del contexto de cifrado. El contexto de cifrado es un par clave-valor de datos adicionales que quiere asociar a la información protegida por AWS KMS. Posteriormente, esto se incorpora a los datos autenticados adicionales (AAD) del cifrado autenticado en textos cifrados con AWS KMS. Si envía el valor del contexto de cifrado en la operación de cifrado, se le exige pasarlo en la operación de descifrado correspondiente. Puede usar el contexto de cifrado dentro de sus políticas para aplicar controles más estrictos para los recursos cifrados. Debido a que el contexto de cifrado está registrado en CloudTrail, puede obtener más información sobre el uso de las claves desde una perspectiva de auditoría. Tenga en cuenta que el contexto de cifrado no está cifrado y podrá verlo dentro de los registros de CloudTrail. El contexto de cifrado no deben considerarse información confidencial ni debe exigir confidencialidad.

Los servicios de AWS que utilizan AWS KMS usan el contexto de cifrado para limitar el alcance de las claves. Por ejemplo, Amazon EBS envía el ID de volumen como el contexto de cifrado cuando cifra/descifra un volumen, y cuando toma una snapshot, el ID de esta se utiliza como el contexto. Si Amazon EBS no utilizara este contexto de cifrado, una instancia de EC2 podría descifrar cualquier volumen de EBS con esa CMK específica.

Un contexto de cifrado también se puede utilizar para las aplicaciones personalizadas que desarrolla y actúa como una capa adicional de control al garantizar que las llamadas de

descifrado se llevan a cabo correctamente solo si el contexto de cifrado coincide con lo que se haya pasado en la llamada de cifrado. Si el contexto de cifrado para una aplicación específica no cambia, puede incluir este contexto dentro de la clave de la política de claves de AWS KMS como una declaración condicional. Por ejemplo, si tiene una aplicación que requiere la capacidad de cifrar y descifrar datos, puede crear una política de claves en la CMK que se asegura de proporcionar los valores esperados. En la siguiente política, se trata de comprobar que el nombre de la aplicación "ExampleApp" y su versión actual "1.0.24" sean los valores que se pasan a AWS KMS durante las llamadas de cifrado y descifrado. Si se pasan valores diferentes, la llamada se denegará y no se realizará la acción de descifrar o cifrar.

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/RoleForExampleApp"
  },
  "Action": [
    "kms:Encrypt",
    "kms:Decrypt"
  ],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:EncryptionContext:AppName": "ExampleApp",
      "kms:EncryptionContext:Version": "1.0.24"
    }
  }
}
```

Este uso del contexto de cifrado ayudará a garantizar que solo las partes o aplicaciones autorizadas pueden acceder y utilizar las CMK. Ahora, la parte tendrá que contar con permisos de IAM para AWS KMS, una política de CMK que les permite utilizar la clave en la forma solicitada y, por último, conocer los valores del contexto de cifrado esperados.

## Multi-factor authentication

Para proporcionar una capa de seguridad adicional a través de acciones específicas, puede implementar una capa adicional de protección mediante multi-factor authentication (MFA) en llamadas a API de KMS fundamentales. Algunas de dichas llamadas son `PutKeyPolicy`, `ScheduleKeyDeletion`, `DeleteAlias` y `DeleteImportedKeyMaterial`. Esto

puede lograrse a través de una declaración condicional dentro de la política de claves que comprueba cuándo o si un dispositivo MFA se utilizó como parte de la autenticación.

Si alguien intenta realizar una de las acciones fundamentales de AWS KMS, la siguiente política de CMK validará que su MFA fue autenticada en los últimos 300 segundos, o 5 minutos, antes de realizar la acción.

```
{
  "Sid": "MFACriticalKMSEvents",
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:user/ExampleUser"
  },
  "Action": [
    "kms:DeleteAlias",
    "kms:DeleteImportedKeyMaterial",
    "kms:PutKeyPolicy",
    "kms:ScheduleKeyDeletion"
  ],
  "Resource": "*",
  "Condition":{
    "NumericLessThan ":{"aws: MultiFactorAuthAge":"300"}
  }
}
```

# Controles de detección

Los controles de detección se aseguran de que configure correctamente AWS KMS para registrar la información necesaria para obtener una mayor visibilidad de su entorno.

## Auditoría de CMK

AWS KMS está integrada con CloudTrail. Para auditar el uso de sus claves en AWS KMS, debe habilitar el registro de CloudTrail en su cuenta de AWS. Esto garantiza que todas las llamadas realizadas a API de KMS sobre las claves en su cuenta de AWS se registren automáticamente en los archivos que posteriormente se entregan al bucket de Amazon Simple Storage Service (S3) que especifique. Con la información recopilada por CloudTrail, podrá determinar qué solicitudes se realizaron, la dirección IP de origen desde la cual se realizó la solicitud, quién la realizó, cuándo se realizó, etcétera.

AWS KMS se integra de manera nativa con muchos otros servicios de AWS para que el monitoreo sea sencillo. Puede utilizar estos servicios de AWS, o su suite de herramientas de seguridad existente, para monitorear sus registro de CloudTrail en busca de acciones específicas como `ScheduleKeyDeletion, PutKeyPolicy, DeleteAlias,`

`DisableKey, DeleteImportedKeyMaterial` en su clave de KMS. Además, AWS KMS emite eventos de Amazon CloudWatch cuando expira el material de claves rotado, eliminado e importado de su CMK.

## Validación del uso de CMK

Además de capturar los datos de auditoría asociados con la administración y el uso de claves, debe asegurarse de que los datos que revisa concuerden con sus políticas y prácticas recomendadas establecidas. Un método consiste en monitorear y comprobar los registros de CloudTrail de forma continua a medida que van entrando. Otro método consiste en utilizar reglas de AWS Config. Al usar las reglas de AWS Config, se asegura de que muchos de los servicios de AWS se configuren de forma adecuada. Por ejemplo, con los volúmenes de EBS, puede utilizar la regla de AWS Config `ENCRYPTED_VOLUMES` para validar el cifrado de los volúmenes de EBS asociados.

### Etiquetas de las claves

Una CMK puede tener una etiqueta aplicada para distintos fines. El uso más común consiste en relacionar una CMK específica a una categoría de negocio (como un centro de costos, el nombre de una aplicación o un propietario). Las etiquetas se pueden usar para comprobar que se está utilizando la CMK correcta para una acción determinada. Por ejemplo, en los registros de CloudTrail, para una determinada acción de KMS, puede comprobar que la CMK que se está utilizando pertenece a la misma categoría de negocio que el recurso en donde se utiliza. Anteriormente, esto podría haber requerido una búsqueda dentro de un catálogo de recursos, pero ahora esta búsqueda externa no es necesaria gracias al etiquetado dentro de AWS KMS, así como en muchos de los otros servicios de AWS.

# Seguridad de la infraestructura

La capacidad de seguridad de la infraestructura le ofrece prácticas recomendadas acerca de cómo configurar AWS KMS para asegurarse de que dispone de una implementación ágil que puede crecer junto a su negocio mientras protege su información confidencial.

## Claves maestras de cliente

Dentro de AWS KMS, la jerarquía de claves comienza con una CMK. Una CMK puede utilizarse para cifrar directamente los bloques de datos hasta 4 KB o puede utilizarse para proteger las claves de datos, las cuales protegen los datos subyacentes de cualquier tamaño.

### CMK administradas por el cliente y administradas por AWS

Las CMK pueden desglosarse en dos tipos generales: administradas por el cliente y administradas por AWS. Una CMK administrada por AWS se crea cuando elige habilitar el cifrado del lado del servidor de un recurso de AWS en la CMK administrada por AWS para dicho servicio por primera vez (por ejemplo, SSE-KMS). La CMK administrada por AWS es

exclusiva de su cuenta de AWS y de la región en la que se utiliza. Una CMK administrada por AWS solo puede utilizarse para proteger recursos dentro del servicio de AWS específico para el que se crea. No proporciona el nivel de control pormenorizado que proporciona una CMK administrada por el cliente. Para obtener más control, una práctica recomendada es utilizar una CMK administrada por el cliente en todos los servicios compatibles de AWS y en sus aplicaciones. Una CMK administrada por el cliente se crea a petición suya y se debe configurar en función de su caso de uso explícito.

La siguiente tabla resume las principales diferencias y similitudes entre las CMK administradas por AWS y las CMK administradas por el cliente.

| | CMK administrada por AWS | CMK administrada por el cliente |
|---|---|---|
| **Creación** | AWS generada en nombre del cliente | Generada por el cliente |
| **Rotación** | Una vez cada tres años de forma automática | Una vez al año de forma automática a través de la confirmación o a pedido de forma manual |
| **Eliminación** | No se puede eliminar | Se puede eliminar |
| **Ámbito de uso** | Limitado a un servicio de AWS específico | Controlado a través de una política de KMS/IAM |
| **Política de acceso de clave** | Administrada por AWS | Administrada por el cliente |
| **Administración de acceso de usuarios** | Política de IAM | Política de IAM |

Para las CMK administradas por el cliente, dispone de dos opciones para crear el material de claves subyacente. Cuando elige crear una CMK con AWS KMS, puede permitir que KMS cree el material criptográfico para usted, o puede elegir importar su propio material de claves. Ambas opciones le proporcionan el mismo nivel de control y auditoría para el uso de la CMK dentro de su entorno. La posibilidad de importar su propio material criptográfico le permite hacer lo siguiente:

- Comprobar que ha generado el material de claves con su origen aprobado que cumple con sus requisitos de aleatoriedad.
- Utilice el material de claves de su propia infraestructura con los servicios de AWS y utilice AWS KMS para administrar el ciclo de vida de ese material de claves en AWS.
- Obtenga la posibilidad de establecer una fecha de vencimiento para el material de claves en AWS y eliminarlo manualmente; además, haga que vuelva a estar disponible en el futuro.
- Tenga la copia original del material de claves y manténgala fuera de AWS para aumentar su durabilidad y recuperación de desastres durante todo el ciclo de vida de dicho material.

La decisión de utilizar un material de claves importado o un material de claves generado por KMS dependerá de las políticas de su organización y los requisitos de conformidad.

## Creación y administración de claves

Debido a que AWS facilita la creación y administración de claves a través del uso de AWS KMS, le recomendamos que tenga un plan sobre cómo utilizar el servicio para controlar mejor el radio alrededor de las claves individuales. Anteriormente, es posible que haya utilizado la misma clave en diferentes regiones geográficas, entornos o incluso aplicaciones. Con AWS KMS, debe definir los niveles de clasificación de los datos y tener al menos una CMK por nivel. Por ejemplo, puede definir una CMK para datos clasificados como "Confidenciales", y así sucesivamente. De este modo, se garantiza que solo los usuarios autorizados tengan permisos para el material de claves que necesitan para completar su trabajo.

También debe decidir cómo desea administrar el uso de AWS KMS. La creación de las claves de KMS dentro de cada cuenta que requiere la capacidad para cifrar y descifrar datos confidenciales funciona mejor para la mayoría de los clientes, pero otra opción es compartir las CMK desde unas pocas cuentas centralizadas. Mantener las CMK en la misma cuenta que la mayoría de la infraestructura utiliza ayuda a los usuarios a aprovisionar y ejecutar los servicios de AWS que utilizan esas claves. Los servicios de AWS no permiten la búsqueda entre cuentas a menos que la entidad principal que realiza la búsqueda tenga permisos de lista* explícitos sobre los recursos propiedad de la cuenta externa. Esto también puede lograrse únicamente a través de la CLI o SDK y no a través de búsquedas basadas en la consola del servicio. Además, al almacenar las credenciales en las cuentas locales, posiblemente sea más fácil delegar permisos para las personas que conocen las entidades principales de IAM que requieren acceso a las CMK específicas. Si compartiera las claves a través de un modelo centralizado, los administradores de AWS KMS tendrían que conocer el nombre del recurso de Amazon (ARN) completo para todos los usuarios de las CMK a fin de garantizar los privilegios mínimos. De lo contrario, los administradores podrían conceder permisos por demás de permisivos sobre las claves.

Su organización también debe tener en cuenta la frecuencia de rotación de las CMK. Muchas organizaciones rotan las CMK todos los años. Esto es fácil de aplicar para las CMK administradas por el cliente con material de claves generado por KMS. Solo tiene que confirmar un calendario de rotación anual para su CMK. Cuando llega el momento de rotar la CMK, se crea una nueva clave de respaldo y se marca como la clave activa para todas las nuevas solicitudes para proteger la información. La clave de respaldo anterior permanece disponible para su uso para descifrar cualquier valor de los textos cifrados existentes que se cifraron con esta clave. Para rotar las CMK con mayor frecuencia, también puede llamar a `UpdateAlias` para señalar un alias a una nueva CMK, tal como se describe en la siguiente sección. El método `UpdateAlias` funciona tanto para las CMK administradas por el cliente como para las CMK con material de claves importado. AWS ha descubierto que la frecuencia de la rotación de claves depende en gran medida de las leyes, regulaciones y políticas corporativas.

## Alias de claves

Un alias de clave le permite abstraer a los usuarios de claves del ID de la clave subyacente específica de la región y el ARN de la clave. Las personas autorizadas pueden crear un alias de clave que les permita a sus aplicaciones utilizar una CMK específica independiente de la región o el calendario de rotación. Por lo tanto, las aplicaciones de múltiples regiones pueden utilizar el mismo alias de clave para hacer referencia a las claves de KMS en varias regiones sin tener que preocuparse por el ID de clave o el ARN de la clave. También puede activar la rotación manual de una CMK al dirigir un alias de clave determinado a una CMK diferente. De forma similar al modo en que el sistema de nombres de dominio (DNS) permite la abstracción de direcciones IP, un alias de clave hace lo mismo para el ID de la clave. Cuando crea un alias de clave, le recomendamos que determine un esquema de nombres que pueda aplicar en sus cuentas como *alias/<Environment>-<Function>-<Service Team>*.

Cabe señalar que los alias de CMK no se pueden utilizar en las políticas. Esto se debe a que el mapeo de los alias de las claves se pueden manipular fuera de la política, lo que permitiría una escalada de privilegio. Por lo tanto, los ID de clave deben utilizarse en las políticas de claves de KMS, las políticas de IAM y las concesiones de KMS.

# Uso de AWS KMS a escala

Como hemos indicado anteriormente, una práctica recomendada es utilizar al menos una CMK para una determinada clase de datos. Esto le ayudará a definir políticas que reducen los permisos para la clave y, por tanto, los datos a los usuarios autorizados. Puede optar por distribuir aún más los datos entre varias CMK para proporcionar controles de seguridad más sólidos dentro de una clasificación de datos determinada.

AWS recomienda utilizar el cifrado de sobre para escalar su implementación de KMS. El cifrado de sobre es la práctica de cifrar los datos de archivos de texto simple con una clave de datos única y luego cifrar la clave de datos con una clave de cifrado de claves (KEK). Dentro de AWS KMS, la CMK es la KEK. Puede cifrar su mensaje con la clave de datos y luego cifrar la clave de datos con la CMK. De esta manera, la clave de datos cifrada se pueden almacenar junto con el mensaje cifrado. Puede almacenar en caché la versión en archivo de texto simple de la clave de datos para un uso repetido, lo que reduce el número de solicitudes a AWS KMS. Además, el cifrado de sobre puede ayudar a diseñar su aplicación para la recuperación de desastres. Puede mover los datos cifrados tal como están entre las regiones y solo tiene que volver a cifrar las claves de datos con las CMK específicas de las regiones.

El equipo criptográfico de AWS ha lanzado un SDK de cifrado de AWS que facilita el uso de AWS KMS de manera eficiente. Este SDK implementa de manera transparente los detalles de bajo nivel para utilizar AWS KMS. También le proporciona a los desarrolladores opciones para proteger sus claves de datos después de su uso para asegurarse de que el desempeño de su aplicación no se vea afectado de forma significativa mediante el cifrado de los datos confidenciales.

# Protección de datos

La capacidad de protección de datos se encarga de algunos de los casos de uso de AWS comunes para utilizar AWS KMS dentro de su organización para proteger la información confidencial.

## Casos de uso de AWS KMS comunes

### Cifrado de datos de PCI a través de AWS KMS

Dado que los controles de seguridad y calidad en AWS KMS han sido validados y certificados para satisfacer los requisitos de la certificación PCI DSS de nivel 1, puede directamente cifrar los datos de número de cuenta principal (PAN) con una AWS KMS CMK. El uso de una CMK para cifrar datos directamente elimina parte de la carga de administrar bibliotecas de cifrado. Además, una CMK no se puede exportar desde AWS KMS, que alivia la preocupación por la clave de cifrado que se almacena de manera no segura. Como todas las solicitudes de KMS se registran en CloudTrail, el uso de la CMK puede auditarse al revisar los registros de CloudTrail. Es importante tener en cuenta el límite de solicitudes por segundo a la hora de diseñar aplicaciones que utilizan la CMK directamente para proteger los datos de la industria de las tarjetas de pago (PCI).

### Administración de datos secretos a través de AWS KMS y Amazon S3

Si bien AWS KMS principalmente ofrece funciones de administración de claves, puede utilizar AWS KMS y Amazon S3 para crear su propia solución de administración de datos secretos.

Cree un nuevo bucket de Amazon s3 para almacenar sus datos secretos. Implemente una política de bucket en el bucket para restringir el acceso únicamente a personas y servicios autorizados. Los datos secretos almacenados en el bucket utilizan un prefijo predefinido por archivo para permitir el control pormenorizado del acceso a los datos secretos. Cada dato secreto, cuando se lo coloca en el bucket de S3, se cifra con una clave de KMS específica administrada por el cliente. Además, debido a la naturaleza altamente confidencial de la información que se almacena en este bucket, se habilitan los registros de acceso de S3 o los eventos de datos de CloudTrail para fines de auditoría. Luego, cuando un usuario o servicio requiere el acceso al dato secreto, asumen una identidad dentro de AWS que tiene los permisos para usar el objeto en el bucket de S3 y la clave de KMS. Una aplicación que se ejecuta en una instancia EC2 utiliza un rol de instancia que tiene los permisos necesarios.

### Cifrado de variables de entorno para Lambda

Por defecto, al crear o actualizar las funciones Lambda que utilizan variables de entorno, estas variables se cifran con AWS KMS. Cuando se invoca la función Lambda, dichos valores se descifran y se ponen a disposición del código Lambda. Tiene la opción de usar la clave de KMS predeterminada para Lambda o especificar una CMK de su elección.

Para proteger aún más sus variables de entorno, deberá seleccionar la casilla de verificación "Habilitar auxiliares de cifrado". Al seleccionar esta opción, las variables de entorno también se

cifrarán individualmente a través de una CMK de su elección y, de esta manera, su función Lambda tendrá que específicamente descifrar cada variable de entorno cifrada que se necesita.

## Cifrado de datos en Parameter Store de Systems Manager

Amazon EC2 Systems Manager es una colección de funciones que puede ayudarle a automatizar las tareas de administración a escala. Para eficientemente almacenar y remitir datos de configuración confidenciales como contraseñas, claves de licencias y certificados, el Parameter Store le permite proteger información confidencial dentro de los parámetros de cadena segura.

Una cadena segura es cualquier dato confidencial que se debe almacenar y remitir de manera segura. Si tiene datos que no desea que los usuarios modifiquen o remitan como texto sin cifrar, como contraseñas de unión de dominio o claves de licencias, especifique esos valores con el tipo de datos de la cadena segura. Debe utilizar cadenas seguras en las siguientes situaciones:

- Desea utilizar los datos/parámetros en los servicios de AWS sin exponer los valores como texto sin cifrar en comandos, funciones, registros de agente o registros de CloudTrail.
- Desea controlar quién tiene acceso a los datos confidenciales.
- Desea tener la posibilidad de auditar los accesos a los datos confidenciales con CloudTrail.
- Desea disponer de un cifrado en el nivel de AWS para los datos confidenciales y desea utilizar sus propias claves de cifrado para administrar el acceso.

Al seleccionar esta opción cuando crea su parámetro, Systems Manager cifra ese valor cuando se transfiere a un comando y lo descifra cuando lo procesa en la instancia administrada. AWS KMS gestiona el cifrado y puede ser una clave de KMS predeterminada para Systems Manager o puede especificar una CMK por parámetro.

## Aplicación del cifrado de los datos en reposo en los servicios de AWS

Su organización podría requerir que el cifrado de todos los datos cumpla con una clasificación específica. Según el servicio específico, puede aplicar políticas de cifrado de datos a través de controles preventivos o de detección. Para algunos servicios como Amazon S3, una política puede impedir el almacenamiento de datos no cifrados. Para otros servicios, el mecanismo más eficiente es monitorear la creación de recursos de almacenamiento y comprobar si el cifrado está habilitado adecuadamente. En el caso de que se cree un almacenamiento sin cifrar, dispone de una serie de posibles respuestas que varían desde eliminar el recurso de almacenamiento hasta notificar a un administrador.

## Cifrado de datos en reposo con Amazon S3

Cuando se usa Amazon S3, es posible implementar una política de bucket de S3 que garantiza que todos los objetos que se están cargando estén cifrados. La política tiene un aspecto similar a lo siguiente:

```
{
   "Version":"2012-10-17",
   "Id":"PutObjPolicy",
   "Declaración":[{
         "Sid":"DenyUnEncryptedObjectUploads",
         "Effect":"Deny",
         "Principal":"*",
         "Action":"s3:PutObject",
         "Resource":"arn:aws:s3:::YourBucket/*",
         "Condition":{
            "StringNotEquals":{
               "s3:x-amz-server-side-encryption":"aws:kms"
            }
         }
      }
   ]
}
```

Tenga en cuenta que esto no hace que los objetos que ya están en el bucket se cifren. Esta política rechaza los intentos de añadir nuevos objetos en el bucket a menos que esos objetos estén cifrados. Los objetos que ya están en el bucket antes de que se aplique esta política permanecerán cifrados o no, según cómo fueron cargados la primera vez.

## Cifrado de datos en reposo con Amazon EBS

Puede crear imágenes de máquina de Amazon (AMI) que hacen uso de los volúmenes de arranque de EBS cifrados y utilizan las AMI para lanzar instancias EC2. Se cifran los datos almacenados, así como la ruta de transferencia de datos entre el volumen de EBS y la instancia EC2. Los datos se descifran en el hipervisor de dicha instancia en función de las necesidades, luego se almacenan en la memoria. Esta función ayuda a la seguridad, a la conformidad y a los esfuerzos de auditoría ya que le permite verificar que todos los datos que se almacenan en el volumen de EBS están cifrados, ya sea que estén almacenados en un volumen de arranque o en un volumen de datos. Además, ya que esta función hace uso de AWS KMS, puede hacer un seguimiento y auditar todos los usos de las claves de cifrado.

Existen dos métodos para garantizar que los volúmenes de EBS estén siempre cifrados. Puede verificar que la marca de cifrado como parte del contexto `CreateVolume` esté configurada en "true" a través de una política de IAM. Si la marca no es "true", la política de IAM puede impedir que una persona cree el volumen de EBS. El otro método consiste en monitorear la creación de

volúmenes de EBS. Si se crea un nuevo volumen de EBS, CloudTrail registrará un evento. Puede activarse una función Lambda mediante el evento de CloudTrail para comprobar si el volumen de EBS está cifrado o no, y también qué clave de KMS se utilizó para el cifrado.

Una función de AWS Lambda puede responder a la creación de un volumen sin cifrar de diferentes maneras. La función podría llamar a la API `CopyImage` con la opción cifrada para crear una nueva versión cifrada del volumen de EBS y asociarlo a la instancia y eliminar la versión anterior. Algunos clientes eligen eliminar automáticamente la instancia EC2 que tiene el volumen no cifrado. Otros optar por automáticamente poner en cuarentena a la instancia mediante la aplicación de grupos de seguridad que evitan la mayoría de las conexiones entrantes. También es fácil escribir una función Lambda que se publica en un tema de Amazon Simple Notification Service (SNS) y le advierte a los administradores que realicen una investigación manual y una intervención. Tenga en cuenta que la mayoría de las respuestas de aplicación pueden (y deben) lograrse de acuerdo al programa sin la intervención humana.

### Cifrado de datos en reposo con Amazon RDS

Amazon Relational Database Service (RDS) se basa en el cifrado de Amazon EBS para proporcionar un cifrado de disco completo para los volúmenes de base de datos. Cuando crea una instancia de base de datos cifrados con Amazon RDS, este crea un volumen de EBS cifrado en su nombre para almacenar la base de datos. Los datos almacenados en reposo en el volumen, las snapshots de base de datos, los backups automatizados y las réplicas de lectura se cifran con la KMS CMK que especificó cuando creó la instancia de base de datos.

Similar a Amazon EBS, puede configurar una función AWS Lambda para monitorear la creación de nuevas instancias RDS a través de la llamada a la API mediante CloudTrail.`CreateDBInstance` En el evento `CreateDBInstance,` asegúrese de que el parámetro `KmsKeyId` esté configurado con la CMK esperada.

# Respuesta frente a incidencias

La capacidad de respuesta frente a incidencias se centra en la capacidad de su organización de solucionar problemas que pueden tener que ver con AWS KMS.

## Automatización de seguridad de AWS KMS

Cuando monitorea sus CMK, si se detecta una acción específica, podría configurarse una función AWS Lambda para deshabilitar la CMK o tomar cualquier otra medida en respuesta frente a incidencias según se establezca en sus políticas de seguridad local. Sin intervención humana, una posible exposición podría quedar sin efecto en cuestión de minutos gracias al uso de las herramientas de automatización dentro de AWS.

# Eliminación y desactivación de las CMK

Si bien es posible eliminar las CMK, esto tiene consecuencias importantes en una organización. Primero debe considerar si es suficiente configurar el estado de la CMK a desactivado en las claves que ya no tiene la intención de utilizar. Esto impedirá cualquier uso futuro de la CMK. No obstante, la CMK seguirá estando disponible y se podrá volver a activar en el futuro si es necesario. AWS KMS sigue almacenando las claves desactivadas; por lo tanto, siguen incurriendo en cargos de almacenamiento recurrentes. Debe considerar firmemente la desactivación de las claves en lugar de eliminarlas hasta que se sienta seguro con su administración de datos cifrados.

Debe considerar con mucho cuidado la eliminación de una clave. Los datos no se pueden descifrar si se eliminó la CMK correspondiente. Además, una vez que se elimina una CMK, desaparece para siempre. AWS no tiene medios para recuperar una CMK eliminada cuando finalmente fue eliminada. Al igual que con otras operaciones fundamentales en AWS, debe aplicar una política que requiera MFA para la eliminación de una CMK.

Para ayudar a garantizar que no se elimina una CMK por error, KMS aplica un período de espera mínimo de siete días antes de que la CMK realmente se elimine. Puede optar por aumentar este período de espera hasta un valor máximo de 30 días. Durante el período de espera, la CMK sigue estando almacenada en KMS en un estado de "eliminación pendiente". No se puede utilizar para cifrar o descifrar operaciones. Cualquier intento por utilizar una clave con el estado de "eliminación pendiente" para el cifrado o descifrado se registrará en el CloudTrail. Puede establecer una alarma de Amazon CloudWatch para estos eventos en los registros de CloudTrail. Esto le da la oportunidad de cancelar el proceso de eliminación si es necesario. Hasta que haya caducado el período de espera, la CMK puede recuperarse del estado "eliminación pendiente" y restablecerse al estado desactivado o activado.

Por último, también cabe señalar que si utiliza una CMK con material de claves importado, puede eliminarlo de forma inmediata. Esto es diferente a eliminar una CMK directamente de varias maneras. Al realizar la acción `DeleteImportedKeyMaterial`, AWS KMS elimina el material de claves y el estado de la CMK cambia a importación pendiente. Cuando se elimina el material de claves, la CMK deja de ser utilizable al instante. No hay período de espera. Para activar nuevamente el uso de la CMK, debe volver a importar el mismo material de claves. La eliminación del material de claves afecta a la CMK de forma inmediata, pero las claves del cifrado de datos que se usan de forma activa por los servicios de AWS no se ven afectadas de forma inmediata.

Por ejemplo, supongamos que una CMK con su material importado se utilizó para cifrar un objeto a colocar en un bucket de S3 mediante SSE-KMS[8]. Justo antes de cargar el objeto en el bucket de S3, coloca el material importado en su CMK. Una vez cargado el objeto, puede eliminar su material de claves de esa CMK. El objeto seguirá en el bucket de S3 en un estado cifrado, pero nadie podrá acceder a él hasta que se vuelva a importar el mismo material de claves a la CMK. Obviamente, este flujo requiere una automatización precisa para importar y

eliminar el material de claves de una CMK, pero puede proporcionar un nivel adicional de control dentro de un entorno.

# Conclusión

AWS KMS le proporciona a su organización un servicio completamente administrado para controlar de manera centralizada las claves de cifrado. Su integración nativa con otros servicios de AWS hace que sea fácil para AWS KMS cifrar los datos que usted almacena y procesa.

Al tomarse el tiempo necesario para diseñar e implementar correctamente AWS KMS, puede garantizar que las claves de cifrado sean seguras y estén disponibles para las aplicaciones y sus usuarios autorizados. Además, puede mostrarle a sus auditores los registros detallados asociados a su uso de las claves.

# Colaboradores

Las siguientes personas y organizaciones han participado en la redacción de este documento:

- Matthew Bretan, Consultor de seguridad sénior, servicios profesionales de AWS
- Sree Pisharody, Gerente de productos sénior - Técnico, criptografía de AWS
- Ken Beer, Gerente sénior, desarrollo de software, criptografía de AWS
- Brian Wagner, Consultor de seguridad, servicios profesionales de AWS
- Eugene Yu, Consultor de administración, servicios profesionales de AWS
- Michael St.Onge, Arquitecto de seguridad global en la nube, servicios profesionales de AWS
- Balaji Palanisamy, Consultor sénior, servicios profesionales de AWS
- Jonathan Rault, Consultor sénior, servicios profesionales de AWS
- Reef Dsouza, Consultor, servicios profesionales de AWS
- Paco Hope, Consultor principal, servicios profesionales de AWS

# Revisiones del documento

Para obtener la versión más actualizada de este documento técnico, visite:

https://d0.awsstatic.com/whitepapers/KMS-Best-Practices.pdf

# Notas

[1] http://docs.aws.amazon.com/kms/latest/developerguide/overview.html

[2] https://d0.awsstatic.com/whitepapers/KMS-Cryptographic-Details.pdf

[3] https://d0.awsstatic.com/whitepapers/aws_cloud_adoption_framework.pdf

[4] https://d0.awsstatic.com/whitepapers/AWS_CAF_Security_Perspective.pdf

[5] http://docs.aws.amazon.com/IAM/latest/UserGuide/introduction.html

[6] http://docs.aws.amazon.com/kms/latest/developerguide/key-policies.html#key-policy-default-allow-root-enable-iam

[7] http://docs.aws.amazon.com/kms/latest/developerguide/policy-conditions.html#conditions-kms-via-service

[8] http://docs.aws.amazon.com/kms/latest/developerguide/services-s3.html#sse