

Explorar Azure Key Vault

La protección de las claves es esencial para proteger la identidad y los datos en la nube.

Azure Key Vault ayuda a proteger las claves criptográficas y los secretos que usan los servicios y aplicaciones en la nube. Key Vault agiliza el proceso de administración de claves y le permite mantener el control de claves que obtienen acceso a sus datos y los cifran. Los desarrolladores pueden crear claves para desarrollo y prueba en minutos y, a continuación, migrarlas a claves de producción. Los administradores de seguridad pueden conceder (y revocar) permisos a las claves según sea necesario.

Puedes usar Key Vault para crear múltiples contenedores seguros denominados almacenes. Los almacenes ayudan a reducir las posibilidades de que se produzca una pérdida accidental de información de seguridad mediante la centralización del almacenamiento de los secretos de aplicación. Los almacenes de claves también controlan y registran el acceso a todo lo que almacenan.

Azure Key Vault puede administrar la solicitud y renovación de certificados TLS. Proporciona características para una solución sólida para la administración del ciclo de vida de certificados.

Azure Key Vault ayuda a solucionar los siguientes problemas:

- **Administración de secretos.** Puede usar Azure Key Vault para almacenar de forma segura y controlar exhaustivamente el acceso a tokens, contraseñas, certificados, claves de API y otros secretos.
- **Administración de claves.** Azure Key Vault se usa como una solución de administración de claves basada, lo que facilita la creación y el control de las claves de cifrado que se usan para cifrar los datos.
- **Administración de certificados.** Azure Key Vault también es un servicio que permite aprovisionar, administrar e implementar fácilmente certificados SSL y TLS públicos y privados para su uso con Azure y los recursos internos conectados.
- **Almacenamiento de secretos respaldados por módulos de seguridad de hardware (HSM).** Las claves y los secretos se pueden proteger mediante software, o bien con dispositivos HSM validados por FIPS 140-2 nivel 2.

Azure Key Vault está diseñado para admitir secretos y claves de aplicación. Key Vault no está pensado como almacenamiento para las contraseñas de usuario.

En la tabla siguiente se enumeran los procedimientos recomendados de seguridad para usar Key Vault.

Procedimiento recomendado

Solución

Conceda acceso a usuarios, grupos y aplicaciones en un ámbito concreto.

use los roles predefinidos de RBAC. Por ejemplo, para conceder acceso a un usuario para administrar los almacenes de claves, se le asignaría el rol predefinido Colaborador de almacén de claves en un ámbito específico. En este caso, el ámbito sería una suscripción, un grupo de recursos o, simplemente, un almacén de claves específico. Si los roles predefinidos no se ajustan a sus necesidades, puede definir roles propios.

Controle a qué tienen acceso los usuarios.

El acceso a un almacén de claves se controla a través de dos interfaces independientes: plano de administración y plano de datos. Los controles de acceso del plano de administración y del plano de datos funcionan de forma independiente. Use RBAC para controlar a qué tienen acceso los usuarios. Por ejemplo, si desea conceder a una aplicación acceso para usar las claves de un almacén de claves, solo necesita conceder permisos de acceso al plano de datos mediante directivas de acceso de Key Vault y no se necesita acceso a ningún plano de administración para esta aplicación. Por el contrario, si quiere que un usuario pueda leer las propiedades y etiquetas del almacén, pero que no acceda a las claves, los secretos o los certificados, puede concederle acceso de lectura mediante RBAC y no se requiere acceso al plano de datos.

Almacene los certificados en el almacén de claves.

Azure Resource Manager puede implementar de manera segura los certificados almacenados en Azure Key Vault para las máquinas virtuales de Azure cuando estas se implementan. Al establecer directivas de acceso adecuadas para el almacén de claves, también controla quién obtiene acceso al certificado. Otra ventaja es que administra todos los certificados desde el mismo sitio en Azure Key Vault.

Asegúrese de que puede recuperar almacenes de claves u objetos de almacén de claves si se eliminan.

La eliminación de almacenes de claves u objetos de almacén de claves puede ser involuntaria o malintencionada. Habilite las características de protección de purga y eliminación temporal de Key Vault, especialmente para las claves que se usan para cifrar datos en reposo. La eliminación de estas claves es equivalente a la pérdida de datos, así que, si es necesario, puede recuperar almacenes eliminados y objetos de almacén. Practique las operaciones de recuperación de Key Vault de forma periódica.

Azure Key Vault se ofrece en dos niveles de servicio: Estándar y Premium

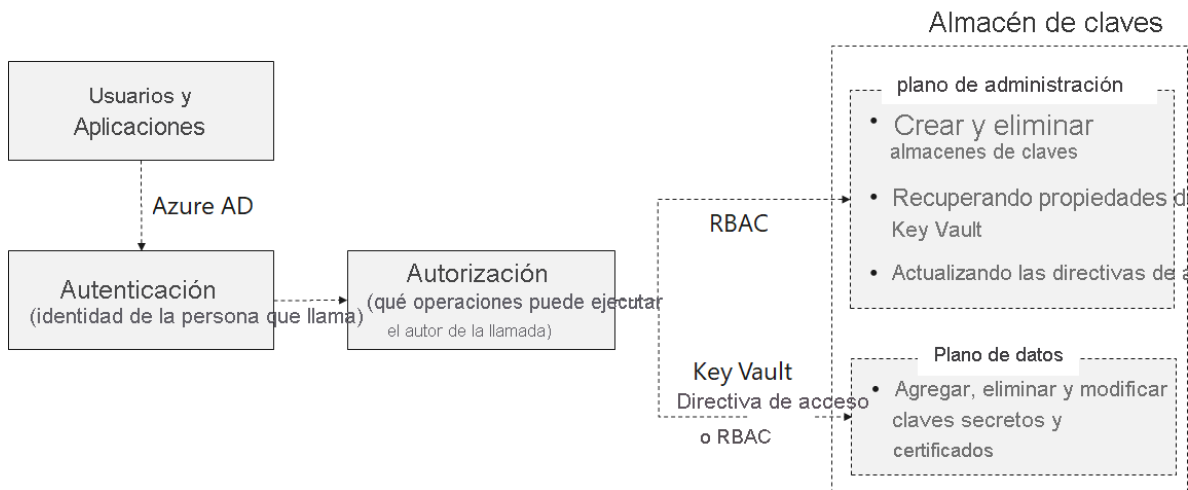
La principal diferencia entre Estándar y Premium es que **Premium admite claves protegidas con HSM**.

Importante

Si un usuario tiene permisos de colaborador (RBAC) en un plano de administración de Key Vault, se puede conceder a sí mismo acceso al plano de datos estableciendo la directiva de acceso al almacén de claves. Se recomienda controlar de forma estricta quién tiene acceso de colaborador a los almacenes de claves, con el fin de garantizar que las personas autorizadas son las únicas que pueden acceder a los almacenes de claves, las claves, los secretos y los certificados, y administrarlos.

Configuración del acceso a Key Vault

El acceso a un almacén de claves se controla a través de dos interfaces: el **plano de administración** y el **plano de datos**. El plano de administración es donde puede administrar el propio almacén de claves. Las operaciones en este plano incluyen crear y eliminar los almacenes de claves, recuperar las propiedades de un almacén de claves y actualizar las directivas de acceso. El plano de datos es donde se trabaja con los datos almacenados en un almacén de claves. Puede agregar, eliminar y modificar claves, secretos y certificados desde aquí.



Para obtener acceso a un almacén de claves en cualquier plano, todos los llamadores (usuarios o aplicaciones) deben tener una autorización y autenticación correctas. La autenticación establece la identidad del llamador. La autorización determina las operaciones que puede ejecutar el llamador.

Ambos planos usan Azure AD para la autenticación. Para la autorización, el plano de administración usa RBAC y el plano de datos puede usar **RBAC recién agregado** o una directiva de acceso de Key Vault.

Autenticación de Active Directory

Cuando se crea un almacén de claves en una suscripción de Azure, se asocia automáticamente con el inquilino de Azure AD de la suscripción. En ambos planos, todos los llamadores deben registrarse en este inquilino y autenticarse para acceder al almacén de claves. En ambos casos, las aplicaciones pueden acceder a Key Vault de dos maneras:

- **Acceso de usuario y aplicación.** la aplicación accede a Key Vault en nombre de un usuario que ha iniciado sesión. Los ejemplos de este tipo de acceso incluyen Azure PowerShell y Azure Portal. Se concede acceso de usuario de dos maneras. Pueden acceder a Key Vault desde cualquier aplicación o deben usar una aplicación específica (denominada identidad compuesta).
- **Acceso solo de aplicación.** la aplicación se ejecuta como un servicio de demonio o un trabajo en segundo plano. A la identidad de la aplicación se le concede acceso al almacén de claves.

Para ambos tipos de acceso, la aplicación se autentica con Azure AD. La aplicación utiliza cualquiera método de autenticación compatible según el tipo de aplicación. La aplicación adquiere un token para un recurso del plano para conceder acceso. El recurso es un punto de conexión en el plano de administración o de datos, según el entorno de Azure. La aplicación usa el token y envía la solicitud de una API de REST a Key Vault. Para más información, revise todo el flujo de autenticación.

Ventajas

El modelo de un único mecanismo de autenticación para ambos planos tiene varias ventajas:

- Las organizaciones pueden controlar de forma centralizada el acceso a todos sus almacenes de claves.
- Si un usuario abandona la organización, al instante pierde el acceso a todos los almacenes de claves de la organización.
- Las organizaciones pueden personalizar la autenticación mediante las opciones de Azure AD, como habilitar la autenticación multifactor para mayor seguridad.

Revisión de un ejemplo de Key Vault seguro

En este ejemplo, se desarrolla una aplicación que utiliza un certificado para SSL, Azure Storage para almacenar los datos y una clave RSA de 2048 bits para las operaciones de firma. La aplicación se ejecuta en una máquina virtual (VM) de Azure o en un conjunto de escalado de máquinas virtuales. Podemos usar un almacén de claves para almacenar los secretos de la aplicación. Podemos almacenar el certificado de arranque que usa la aplicación para autenticarse con Azure AD.

Se necesita acceso a los siguientes secretos y claves almacenados:

- **Certificado SSL:** se usa para SSL.
- **Clave de almacenamiento:** se usa para acceder a la cuenta de Storage.
- **Clave de 2048 bits RSA:** se usa para las operaciones de firma.
- **Certificado de arranque:** se usa para autenticarse con Azure AD. Una vez que se concede acceso, se puede recuperar la clave de almacenamiento y usar la clave RSA para la firma.

Es necesario definir los siguientes roles para especificar quién puede administrar, implementar y auditar la aplicación:

- **Equipo de seguridad:** el personal de TI de la oficina del director de seguridad o colaboradores similares. El equipo de seguridad es responsable de la protección adecuada

de los secretos. Los secretos pueden incluir los certificados SSL, las claves RSA utilizadas para la firma, las cadenas de conexión y las claves de la cuenta de almacenamiento.

- **Desarrolladores y operadores:** el personal que desarrolla la aplicación y la implementa en Azure. Los miembros de este equipo no forman parte del personal de seguridad. No deben tener acceso a información confidencial, como los certificados SSL y las claves RSA. Solo la aplicación que implementan debe tener acceso a información confidencial.
- **Auditores:** el rol es para colaboradores que no son miembros del personal de TI general o de desarrollo. Deben revisar el uso y mantenimiento de los certificados, las claves y los secretos para garantizar el cumplimiento de los estándares de seguridad.

Hay otro rol que está fuera del ámbito de nuestra aplicación: el **administrador de la suscripción (o grupo de recursos)**. El administrador de la suscripción configura los permisos de acceso iniciales del equipo de seguridad. Conceden acceso al equipo de seguridad mediante el uso de un grupo de recursos que tiene los recursos requeridos por la aplicación.

Equipo de seguridad

- Crear instancias de Key Vault.
- Activar el registro de Key Vault.
- Agregar claves y secretos.
- Crear copias de seguridad de las claves para la recuperación ante desastres.
- Establecer directivas de acceso de Key Vault para conceder permisos a usuarios y aplicaciones para operaciones concretas.
- Rotar periódicamente las claves y los secretos.

Desarrolladores y operadores

- Obtener referencias del equipo de seguridad para los certificados de arranque y SSL (huellas digitales), la clave de almacenamiento (identificador URI de secreto) y la clave RSA (identificador URI de clave) para la firma.
- Desarrollar e implementar una aplicación que acceda a las claves y los secretos mediante programación.

Auditores

- Revisar los registros de Key Vault para confirmar el uso adecuado de las claves y los secretos, así como el cumplimiento de los estándares de seguridad de datos.

En la tabla siguiente se resumen los permisos de acceso para los roles y la aplicación.

Rol	Permisos del plano de administración	Permisos del plano de datos
Equipo de seguridad	Colaborador de almacén de claves	Claves: back, create, delete, get, import, list, restore. Secretos: todas las operaciones
Desarrolladores y operadores	Permiso de implementación de Key Vault Nota: este permiso permite que las máquinas virtuales implementadas capturen secretos desde un almacén de claves.	None
Audidores	None	Claves: list Secretos: list. Nota: Este permiso permite a los auditores inspeccionar los atributos (etiquetas, fechas de activación y fechas de expiración) para las claves y los secretos que no se emiten en los registros.
Application	None	Claves: sign Secretos: get

Los tres roles de equipo necesitan tener acceso a otros recursos junto con los permisos de Key Vault. Para implementar máquinas virtuales (o la característica Web Apps de Azure App Service), los desarrolladores y operadores necesitan acceso de colaborador a esos tipos de recursos. Los auditores necesitan acceso de lectura a la cuenta de almacenamiento donde se almacenan los registros de Key Vault.

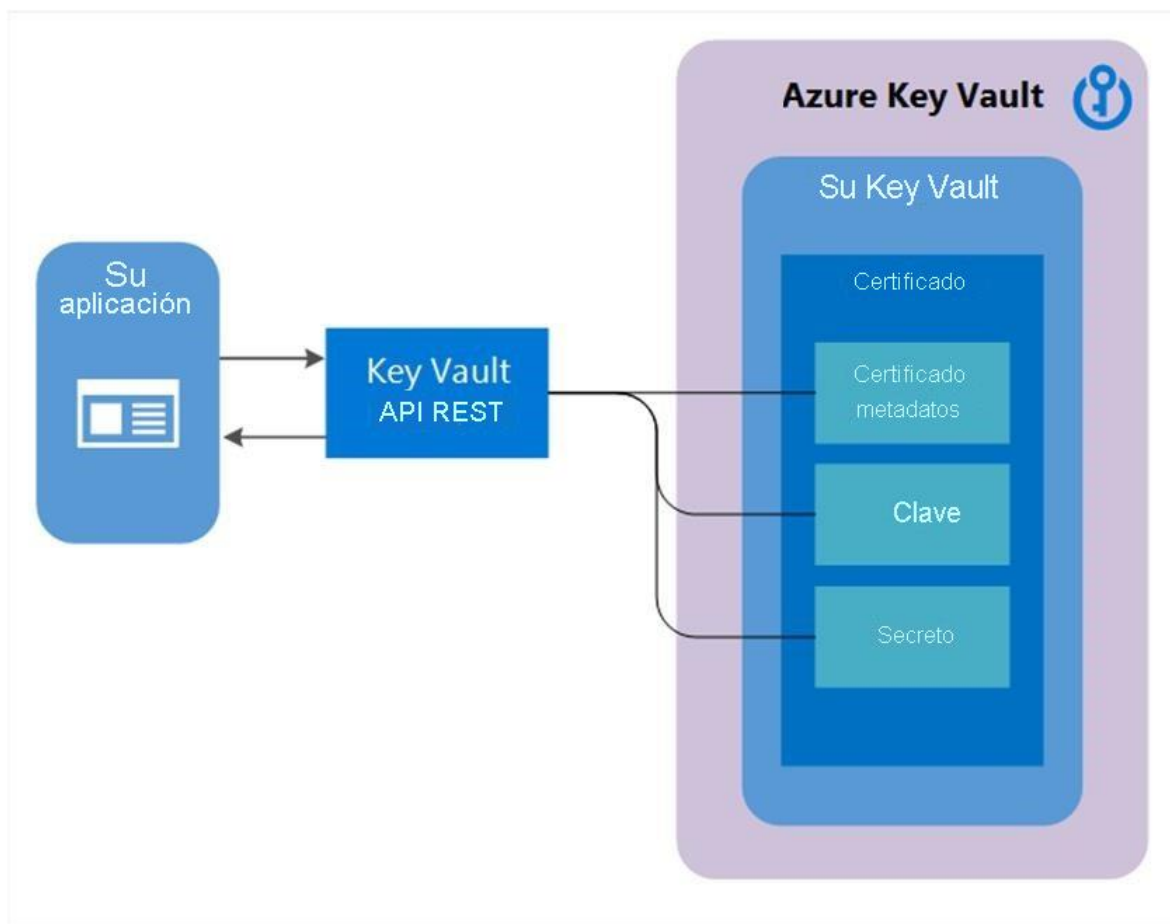
Implementación y administración de certificados de Key Vault

La compatibilidad con certificados de Key Vault proporciona la administración de los certificados x509 y habilita:

- El propietario de un certificado para crear un certificado a través de un proceso de creación de Key Vault o a través de la importación de un certificado existente. Incluye certificados autofirmados y generados por CA.
- El propietario de un certificado de Key Vault para implementar almacenamiento seguro y la administración de certificados X509 sin la interacción con material de clave privada.
- El propietario de un certificado para crear una directiva que indique a Key Vault cómo administrar el ciclo de vida de un certificado.
- Los propietarios de certificados para proporcionar información de contacto para la notificación de eventos del ciclo de vida de expiración y renovación de certificados.
- Renovación automática con emisores seleccionados: propietarios de certificados X509 y entidades de certificación asociados con Key Vault.

Cuando se crea un certificado de Key Vault, se crean también una clave direccionable y un secreto con el mismo nombre. La clave de Key Vault permite las operaciones de clave y el secreto de Key Vault permite la recuperación del valor del certificado como un secreto. Un certificado de Key Vault también contiene metadatos del certificado X.509 público.

El identificador y la versión de los certificados es similar al de las claves y los secretos. Una versión específica de una clave direccionable y el secreto creados con la versión del certificado de Key Vault está disponible en la respuesta del certificado de Key Vault.



Cuando se crea un certificado de Key Vault, este se puede recuperar desde el secreto direccionable con la clave privada en formato PFX o PEM. Sin embargo, la directiva utilizada para crear el certificado debe indicar que la clave es exportable. Si la directiva indica que no es exportable, la clave privada no forma parte del valor cuando se recupera como un secreto.

La clave direccionable se vuelve más pertinente con los certificados de Key Vault no exportables. Las operaciones de la clave de Key Vault direccionable se asignan desde el campo keyusage de la directiva del certificado de Key Vault usada para crear el certificado de Key Vault. Si expira un certificado de Key Vault, la clave y el secreto direccionables dejan de funcionar.

Se admiten dos tipos de clave con los certificados: RSA o RSA HSM. Exportable solo se permite con RSA y no es compatible con RSA HSM.

Directiva de certificados

Una directiva de certificado contiene información sobre cómo crear y administrar el ciclo de vida del certificado de Key Vault. Cuando se importa un certificado con clave privada en la instancia de Key Vault, se crea una directiva predeterminada mediante la lectura del certificado x509.

Cuando se crea un certificado de Key Vault desde el principio, debe proporcionarse una directiva. Esta directiva especifica cómo crear la versión del certificado de Key Vault o la siguiente versión del certificado de Key Vault. Una vez establecida una directiva, no es necesaria con las sucesivas

operaciones de creación para versiones futuras. Solo hay una instancia de una directiva para todas las versiones de un certificado de Key Vault.

En un nivel general, una directiva de certificado contiene la siguiente información:

- Propiedades del certificado X509. contiene el nombre de asunto, nombres alternativos de asunto y otras propiedades que se usan para crear una solicitud de certificado X509.
- Propiedades clave. Contiene los campos Tipo de clave, Longitud de la clave, Exportable y Volver a usar clave. Estos campos indican a Key Vault cómo generar una clave.
- Propiedades del secreto. Contiene propiedades del secreto como el tipo de contenido del secreto direccionable para generar el valor del secreto, para recuperar el certificado como secreto.
- Acciones de duración. Contiene acciones de duración para el certificado de Key Vault. Cada acción de vigencia contiene:
 - Desencadenador, que se especifica con días antes de la expiración o el porcentaje del intervalo de duración.
 - Acción, que especifica el tipo de acción: emailContacts o autoRenew.
- Emisor: contiene los parámetros sobre el emisor de certificado que se usarán para emitir certificados x509.
- Atributos de directiva: contiene atributos asociados a la directiva.

Emisor de certificados

Para poder crear un emisor de certificado en una instancia de Key Vault, se deben completar correctamente los dos pasos de requisitos previos siguientes:

1. Incorporación a proveedores de CA:
 - un administrador de la organización debe incorporar su empresa con al menos un proveedor de CA.
2. El administrador crea credenciales de solicitante para que Key Vault inscriba (y renueve) certificados SSL:
 - Proporciona la configuración que se va a usar para crear un objeto de emisor del proveedor en el almacén de claves.

Contactos de certificados

Los contactos de certificados contienen información de contacto para enviar notificaciones desencadenadas por los eventos de vigencia del certificado. La información de los contactos es compartida por todos los certificados del almacén de claves. Se envía una notificación a todos los contactos especificados para un evento de cualquier certificado del almacén de claves.

Si la directiva de un certificado se establece en renovación automática, se envía una notificación para los siguientes eventos:

- Antes de la renovación del certificado
- Después de la renovación de certificado e indicando si el certificado se renovó correctamente o si se produjo un error, requiriéndose la renovación manual del certificado
- Cuando llega el momento de renovar un certificado para una directiva de certificado que se establece para la renovación manual (solo correo electrónico)

Control de acceso a certificados

La instancia de Key Vault que contiene certificados administra el control de acceso para esos mismos certificados. La directiva de control de acceso para los certificados es distinta de la directiva de control de acceso para las claves y los secretos en la misma instancia de Key Vault. Los usuarios pueden crear uno o varios almacenes para almacenar los certificados, a fin de mantener una segmentación y administración de los certificados apropiadas para cada escenario.

Los siguientes permisos reflejan fielmente las operaciones permitidas en un objeto de secreto y pueden utilizarse, en función de cada entidad, en la entrada del control de acceso de secretos en un almacén de claves:

- Permisos para operaciones de administración de certificados:
 - get: obtener la versión actual del certificado o cualquier versión de un certificado.
 - list: enumerar los certificados actuales o las versiones de un certificado.
 - update: agregar un certificado.
 - create: crear un certificado de Key Vault.
 - import: importar el material del certificado en un certificado de Key Vault.
 - delete: eliminar un certificado, su directiva y todas sus versiones.
 - recover: recuperar un certificado eliminado.
 - backup: realizar una copia de seguridad de un certificado en un almacén de claves.
 - restore: restaurar la copia de seguridad de un certificado a un almacén de claves.
 - managecontacts: administrar contactos del certificado de Key Vault.
 - manageissuers: administrar autoridades/emisores de certificados de Key Vault.
 - getissuers: obtener las autoridades o los emisores de un certificado.
 - listissuers: enumerar las autoridades o los emisores de un certificado.
 - setissuers: crear o actualizar las autoridades o los emisores de un certificado de Key Vault.
 - deleteissuers: eliminar las autoridades o los emisores de un certificado de Key Vault.

- Permisos para operaciones con privilegios:
 - purge: purgar (eliminar permanentemente) un certificado eliminado.

Creación de claves de Key Vault

Las claves criptográficas de Key Vault se representan como objetos JSON Web Key (JWK). Hay dos tipos de claves, en función de cómo se crearon.

- **Claves débiles:** una clave procesada en software por Key Vault, pero se cifra en reposo mediante una clave del sistema que está en un módulo de seguridad de hardware (HSM). Los clientes pueden importar una clave RSA o EC existente (curva elíptica) o solicitar que Key Vault genere una.
- **Claves rígidas:** una clave procesada en un HSM (módulo de seguridad de hardware). Estas claves se protegen en uno de los espacios de seguridad de HSM de Key Vault (hay un espacio de seguridad en cada región geográfica para mantener el aislamiento). Los clientes pueden importar una clave RSA o EC, de forma temporal o exportándola desde un dispositivo HSM compatible. Los clientes también pueden solicitar que Key Vault genere una clave.

Operaciones con claves

Key Vault admite muchas operaciones en objetos de clave. Estas son algunas:

- **Crear:** permite a un cliente crear una clave en Key Vault. El valor de la clave lo genera y almacena Key Vault y no se entrega al cliente. Las claves asimétricas pueden crearse en Key Vault.
- **Import:** permite a un cliente importar una clave existente en Key Vault. Se pueden importar claves asimétricas en Key Vault mediante una serie de métodos de empaquetado diferentes dentro de una construcción JWK.
- **Actualizar:** permite a un cliente con los permisos suficientes modificar los metadatos (atributos de la clave) asociados con una clave almacenada previamente en Key Vault.
- **Eliminar:** permite a un cliente con permisos suficientes eliminar una clave de Key Vault

Operaciones criptográficas

Una vez creada una clave en Key Vault, se pueden realizar las siguientes operaciones criptográficas mediante la clave. Para obtener el mejor rendimiento de la aplicación, verifique que las operaciones se realizan localmente.

- **Firmar y verificar:** estrictamente, esta operación es "firmar un hash" o "verificar un hash", ya que Key Vault no admite la creación de un hash del contenido como parte de la creación de la firma. Las aplicaciones deben crear el hash de los datos que se van a firmar de modo local y, a continuación, solicitar que Key Vault firme el hash. La verificación de

valores hash firmados se admite como una operación conveniente para aplicaciones que no pueden acceder a material de clave [público].

- **Cifrado/Encapsulado de clave:** una clave almacenada en Key Vault se puede utilizar para proteger otra clave, normalmente una clave de cifrado de contenido (CEK) simétrica. Cuando la clave en Key Vault es asimétrica, se usa el cifrado de claves. Cuando la clave en Key Vault es simétrica, se usa el encapsulado de clave.
- **Cifrar y descifrar:** una clave almacenada en Key Vault puede utilizarse para cifrar o descifrar un único bloque de datos. El tamaño del bloque se determina en función del tipo de clave y del algoritmo de cifrado seleccionado. La operación de cifrado se proporciona por comodidad, para las aplicaciones que no pueden acceder a material de clave [público].

Plan de servicios de aplicaciones

Cada vez más organizaciones adoptan directivas de administración de secretos, donde los secretos se almacenan de forma centralizada con expectativas en torno a la expiración y el control de acceso. Azure Key Vault proporciona estas funcionalidades de administración a las aplicaciones de Azure, pero algunas aplicaciones no pueden adoptar fácilmente los cambios de código para empezar a integrarse con él. Las referencias de Key Vault son una manera de introducir la administración de secretos en la aplicación sin cambios de código.

Las aplicaciones hospedadas en App Service y Azure Functions ahora simplemente pueden definir una referencia a un secreto administrado en Key Vault como parte de la configuración de la aplicación. La identidad asignada por el sistema de la aplicación se usa para capturar de forma segura el secreto y ponerlo a disposición de la aplicación como variable de entorno. Esto significa que los equipos pueden simplemente reemplazar los secretos existentes almacenados en la configuración de la aplicación por referencias al mismo secreto en Key Vault, y la aplicación seguirá funcionando con normalidad.

Configurar una solución de generación de claves del módulo de seguridad de hardware

Para obtener una mayor seguridad, cuando utilice Azure Key Vault, puede importar o generar claves en módulos de seguridad de hardware (HSM) que no se salen nunca del límite de los HSM. Con frecuencia este escenario también se conoce como **Bring Your Own Key (BYOK)**. Los HSM tienen la validación FIPS 140-2 de nivel 2. Azure Key Vault usa la familia Thales nShield de HSM para proteger sus claves. (esta funcionalidad no está disponible para Azure China).

Generación y transferencia de una clave protegida con HSM a través de Internet:

- Genere la clave desde una estación de trabajo sin conexión, lo que reduce la superficie de ataque.
- La clave está cifrada con una Clave de intercambio de claves (KEK), que permanece cifrada hasta que se transfiere a los HSM de Azure Key Vault. Solo la versión cifrada de la clave deja la estación de trabajo original.
- El conjunto de herramientas establece las propiedades en su clave de inquilino que enlaza la clave con el espacio de seguridad de Azure Key Vault. Una vez que los HSM de Azure Key

Vault reciban y descifren la clave, solo estos HSM podrán usarla. La clave no se puede exportar. Este enlace lo exigen los HSM de Thales.

- La KEK que cifra la clave se genera dentro de los HSM de Azure Key Vault y no es exportable. Los HSM exigen que no pueda haber una versión sin cifrar de la KEK fuera de los HSM. Además, el conjunto de herramientas incluye la atestación desde Thales de que la KEK no es exportable y se generó dentro de un HSM genuino que fabricó Thales.
- El conjunto de herramientas incluye la atestación desde Thales de que el espacio de seguridad de Azure Key Vault también se generó en un HSM genuino que fabricó Thales.
- Microsoft usa KEK independientes y espacios de seguridad independientes en cada región geográfica. Esta separación garantiza que la clave puede utilizarse únicamente en centros de datos de la región en la que se ha cifrado. Por ejemplo, una clave de un cliente europeo no se puede utilizar en centros de datos de Norteamérica o Asia.

Si tiene acceso a HSM de Thales, tarjetas inteligentes y software de soporte, puede ver un ejercicio detallado en el vínculo anterior. Se recomienda revisar los pasos incluso si no puede realizar el ejercicio

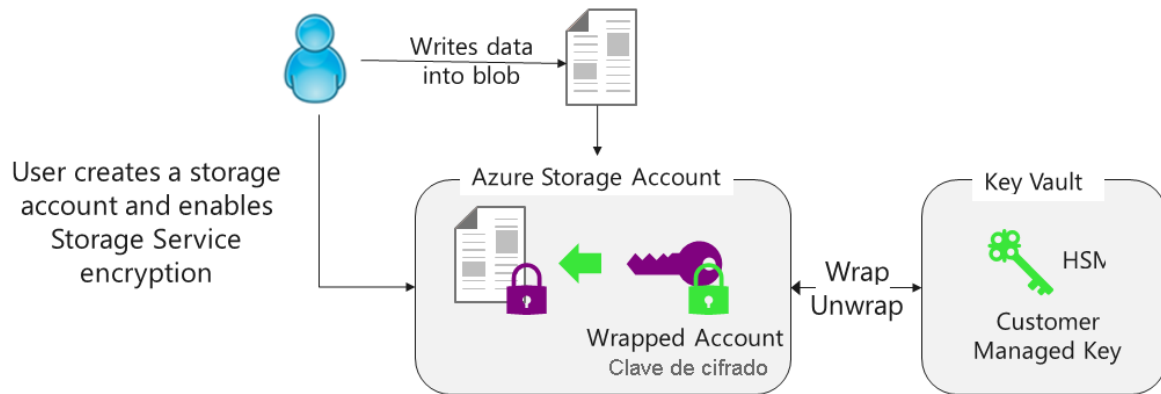
Administración de claves administradas por el cliente

Una vez que haya creado la instancia de Key Vault y la haya rellenado con claves y secretos. El siguiente paso consiste en configurar una estrategia de rotación para los valores que almacena como secretos de Key Vault. Los secretos se pueden rotar de varias maneras:

- Como parte de un proceso manual.
- Mediante programación con llamadas API REST
- A través de un script de Azure Automation.

Ejemplo de Storage Service Encryption con claves administradas por el cliente.

Este servicio usa Azure Key Vault, que proporciona almacenamiento seguro altamente disponible y escalable para claves criptográficas RSA con el respaldo de dispositivos HSM validados por FIPS 140-2 nivel 2 (módulos de seguridad de hardware). Key Vault simplifica el proceso de administración de claves y permite a los clientes mantener totalmente el control de las claves que se usan para cifrar los datos, administrar y auditar su uso de claves, con el fin de proteger los datos confidenciales como parte de sus necesidades normativas o de cumplimiento, compatibles con HIPAA y BAA.



Los clientes pueden generar o importar su clave RSA para Azure Key Vault y habilitar Storage Service Encryption. Azure Storage controla el cifrado y descifrado de forma totalmente transparente mediante el cifrado de sobre en el que se cifran los datos con una clave basada en AES, que a su vez está protegida mediante la clave administrada por el cliente almacenada en Azure Key Vault.

Los clientes pueden rotar su clave en Azure Key Vault según sus directivas de cumplimiento. Cuando rotan su clave, Azure Storage detecta la nueva versión de la clave y vuelve a cifrar la clave de cifrado de cuenta de esa cuenta de almacenamiento. La rotación de claves no da lugar a un nuevo cifrado de todos los datos y no se requiere ninguna otra acción del usuario.

Los clientes también pueden revocar el acceso a la cuenta de almacenamiento revocando el acceso en su clave en Azure Key Vault. Hay varias maneras para revocar el acceso a las claves. Para más información, consulte PowerShell de Azure Key Vault y la CLI de Azure Key Vault. La revocación del acceso bloqueará eficazmente el acceso a todos los blobs de la cuenta de almacenamiento, ya que Azure Storage no podrá acceder a la clave de cifrado de la cuenta.

Los clientes pueden habilitar esta característica en todos los tipos de redundancia disponibles de Azure Blob Storage, incluido Premium Storage, y pueden cambiar del uso de claves administradas por Microsoft al de claves administradas por el cliente. No hay ningún cargo adicional por habilitar esta característica.

Puede habilitar esta característica en cualquier cuenta de almacenamiento de Azure Resource Manager mediante Azure Portal, Azure PowerShell, la CLI de Azure o la API de proveedor de recursos de Microsoft Azure Storage.

Habilitación de secretos de Key Vault

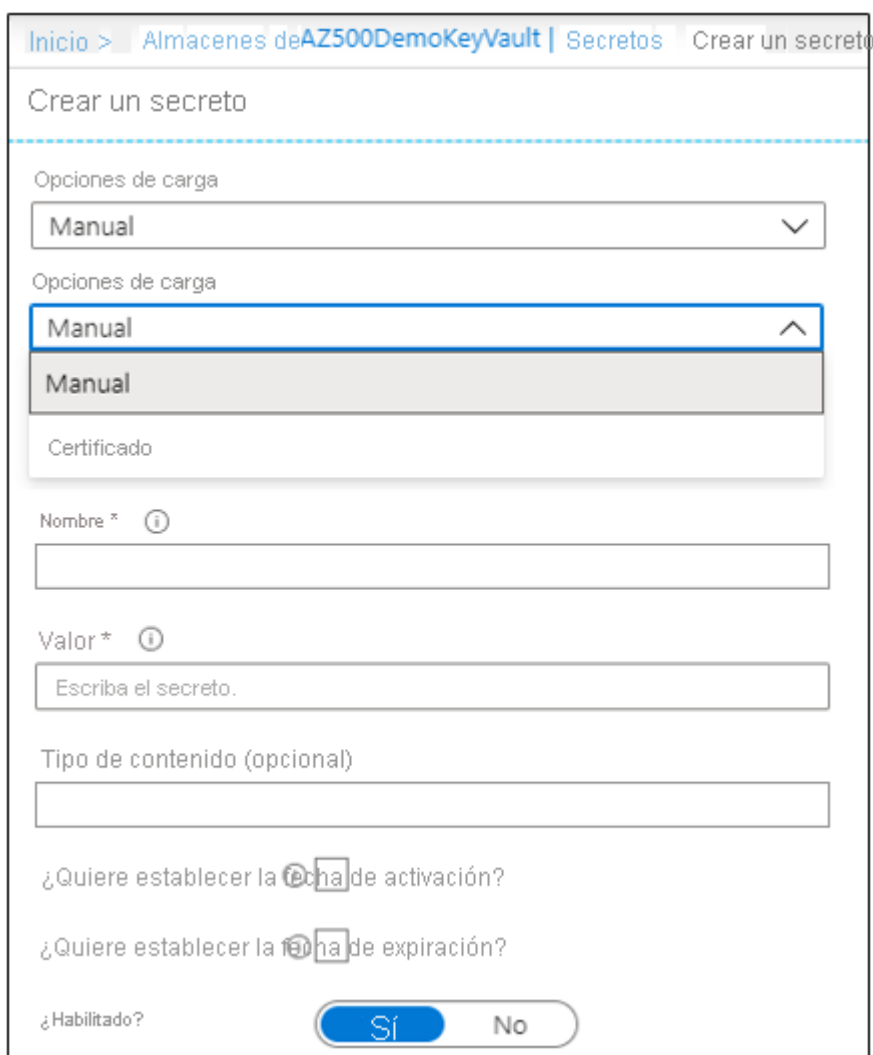
Key Vault proporciona un almacenamiento seguro de secretos, como contraseñas y cadenas de conexión de base de datos.

Desde la perspectiva del desarrollador, las API de Key Vault aceptan y devuelven los valores de secreto como cadenas. Internamente, Key Vault almacena y administra secretos como secuencias de octetos (bytes de 8 bits), con un tamaño máximo de 25 000 bytes. El servicio Key Vault no

proporciona semántica para los secretos. Simplemente acepta los datos, los cifra, los almacena y devuelve un identificador ("id.") de secreto. El identificador puede usarse para recuperar el secreto en un momento posterior.

Si se trata de información muy confidencial, los clientes deben considerar las capas adicionales de protección de datos. Un ejemplo es el cifrado de datos mediante una clave de protección independiente antes de su almacenamiento en Key Vault.

Key Vault también admite un campo contentType para secretos. Los clientes pueden especificar el tipo de contenido de un secreto para ayudar a interpretar los datos secretos cuando se recuperen. La longitud máxima de este campo es de 255 caracteres. No hay ningún valor predeterminado. El uso sugerido es como sugerencia para interpretar los datos secretos. Por ejemplo, una implementación puede almacenar contraseñas y certificados como secretos y, a continuación, utilizar este campo para diferenciarlos. No hay ningún valor predeterminado.



Inicio > Almacenes de AZ500DemoKeyVault | Secretos Crear un secreto

Crear un secreto

Opciones de carga

Manual

Opciones de carga

Manual

Manual

Certificado

Nombre * ⓘ

Valor * ⓘ

Escriba el secreto.

Tipo de contenido (opcional)

¿Quiere establecer la fecha de activación?

¿Quiere establecer la fecha de expiración?

¿Habilitado?

Si No

Como se mostró anteriormente, los valores de los secretos de Key Vault son:

- Par nombre-valor: **el nombre debe ser único en el almacén**

- El valor puede ser cualquier cadena UTF-8: tamaño máximo de 25 KB
- Creación manual o de certificados
- Fecha de activación
- Fecha de expiración

Cifrado

Todos los secretos de Key Vault se almacenan cifrados. Este cifrado es transparente y no requiere ninguna acción del usuario. El servicio Azure Key Vault cifra sus secretos cuando los agrega y los descifra automáticamente cuando los lee. La clave de cifrado es exclusiva de cada almacén de claves.

Administración de claves de cuenta de almacenamiento de Azure

Key Vault puede administrar las claves de la cuenta de almacenamiento de Azure:

- Internamente, Key Vault puede enumerar (sincronizar) las claves con una cuenta de almacenamiento de Azure.
- Key Vault vuelve a generar (rotar) las claves periódicamente.
- Los valores de clave nunca se devuelven como respuesta al autor de la llamada.
- Key Vault administra las claves de las cuentas de almacenamiento y de las cuentas de almacenamiento clásicas.

Control de acceso a la cuenta de almacenamiento

Los siguientes permisos pueden usarse al autorizar a una entidad de seguridad de aplicación o usuario para realizar operaciones en una cuenta de almacenamiento administrada:

Permisos para las operaciones de definición de SaS y cuenta de almacenamiento administrada:

- get: administra información sobre una cuenta de almacenamiento
- list: enumerar cuentas de almacenamiento administradas por una instancia de Key Vault
- update: actualizar una cuenta de almacenamiento
- delete: eliminar una cuenta de almacenamiento
- recover: recuperar una cuenta de almacenamiento eliminada
- backup: realizar una copia de seguridad de una cuenta de almacenamiento
- restore: restaurar la copia de seguridad de una cuenta de almacenamiento a una instancia de Key Vault
- set: crear o actualizar una cuenta de almacenamiento
- regeneratekey: regenerar un valor de clave especificado para una cuenta de almacenamiento

- getsas: obtener información sobre una definición de SAS de una cuenta de almacenamiento
- listsas: enumerar definiciones de SAS de almacenamiento para una cuenta de almacenamiento
- deletesas: eliminar una definición de SAS de una cuenta de almacenamiento
- setsas: crear o actualizar atributos o una nueva definición de SAS para una cuenta de almacenamiento

Permisos para las operaciones con privilegios

- purge: purgar (eliminar permanentemente) una cuenta de almacenamiento administrada

Configuración de la rotación de clave

Una vez que tenga claves y secretos almacenados en el almacén de claves, es muy importante pensar en una estrategia de rotación. Hay varias maneras de rotar los valores:

- Como parte de un proceso manual.
- Mediante programación usando llamadas API.
- A través de un script de Azure Automation.

En este diagrama se muestra cómo pueden usarse Event Grid y aplicaciones de funciones para automatizar el proceso.

1. Treinta días antes de la fecha de expiración de un secreto, Key Vault publica el evento de "expiración cercana" en Event Grid.
2. Event Grid comprueba las suscripciones del evento y usa HTTP POST para llamar al punto de conexión de la aplicación de funciones suscrita al evento.
3. La aplicación de funciones recibe la información del secreto, genera una nueva contraseña aleatoria y crea una versión del secreto con la contraseña nueva en Key Vault.
4. La aplicación de funciones actualiza SQL Server con la nueva contraseña.

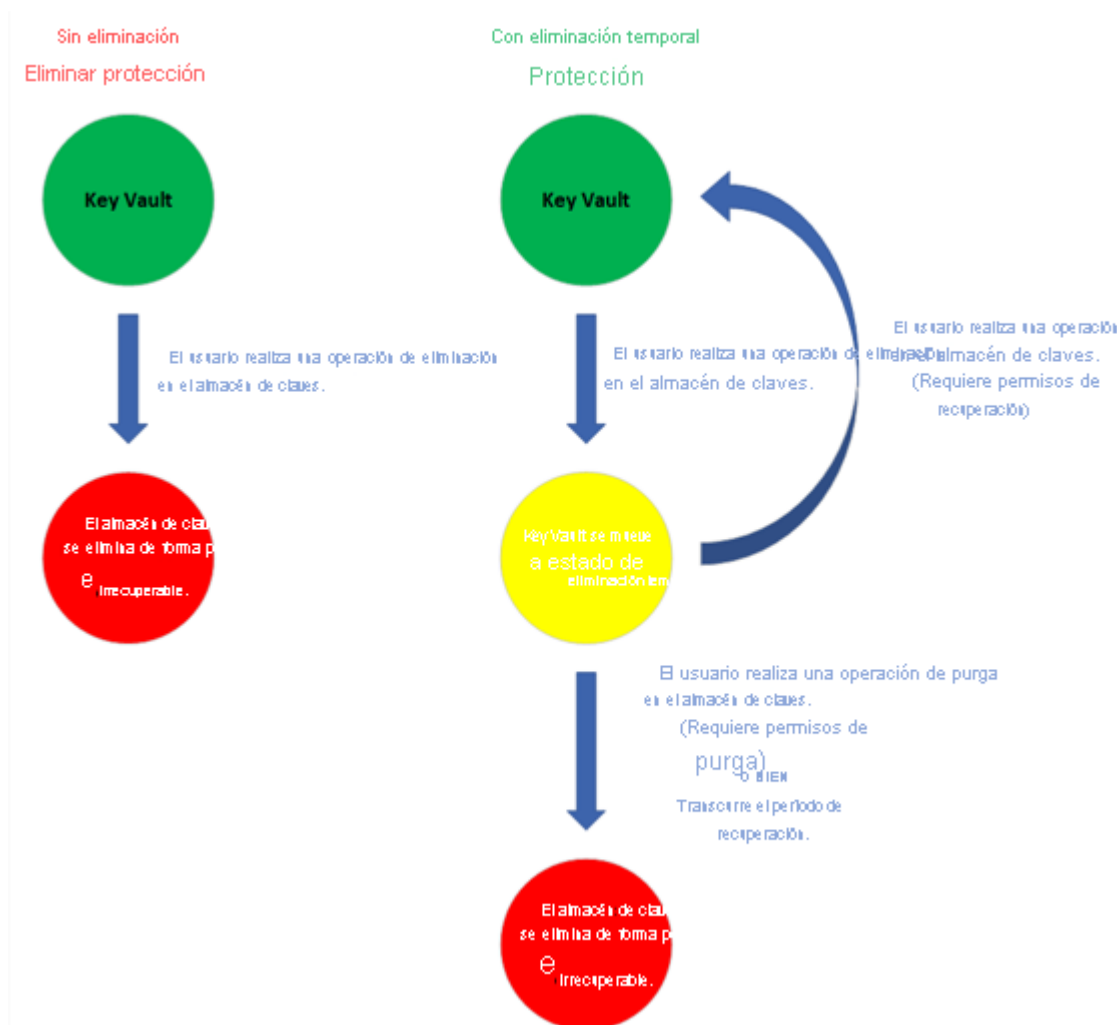
Administración de características de seguridad y recuperación de Key Vault

Siempre quiere asegurarse de que puede hacer una copia de seguridad de claves, certificados e incluso secretos de su instancia de Key Vault en caso de emergencia. Hay dos características principales para ello:

- **Eliminación temporal de Azure Key Vault**
- **Copia de seguridad de Key Vault**

Eliminación temporal de Key Vault

En este diagrama se muestra el flujo de proceso de eliminación de una clave con y sin protección de eliminación temporal.



Cuando se elimina un secreto de un almacén de claves sin protección contra la eliminación temporal, el secreto se elimina de forma permanente. Los usuarios pueden optar por no realizar la eliminación temporal durante la creación del almacén de claves. Sin embargo, Microsoft pronto habilitará la protección contra la eliminación temporal en todos los almacenes de claves para proteger los secretos de la eliminación accidental o malintencionada de un usuario. Los usuarios ya no podrán optar por realizar ni desactivar la eliminación temporal.

Copia de seguridad de Key Vault

En la captura de pantalla se muestra cómo descargar una copia de seguridad de una clave principal. Se puede usar un proceso similar para realizar una copia de seguridad de otros elementos en Key Vault.

Inicio > Almacenes de [new-primary-vault](#) | Clave



pruebas

Versiones



Nueva versión

Actualizar



Eliminar



Descargar copia de seguridad

Versión	Estado
VERSIÓN ACTUAL	
f3c1	✓ Habilitado
VERSIONES ANTERIORES	
0ee11	✓ Habilitado

Puede realizar una copia de seguridad de secretos, claves y certificados almacenados en el almacén de claves. Esta copia de seguridad está diseñada para proporcionarle una copia sin conexión de todos los secretos en el caso improbable de que pierda el acceso al almacén de claves.

Nota

Actualmente no hay ninguna manera de realizar una copia de seguridad de toda la instancia de Key Vault

Tutorial: Uso de Azure Key Vault con una máquina virtual en Python

Azure Key Vault le ayuda a proteger las claves, los secretos y los certificados, como las claves de API y las cadenas de conexión de base de datos.

En este tutorial, configurará una aplicación de Python para leer información de Azure Key Vault mediante identidades administradas para recursos de Azure. Aprenderá a:

- Creación de un Almacén de claves
- Almacenar un secreto en Key Vault
- Crear una máquina virtual Linux de Azure
- Habilitar una [identidad administrada](#) para la máquina virtual
- Conceder los permisos necesarios para que la aplicación de consola lea datos de Key Vault
- Recuperar un secreto del almacén de claves

Antes de empezar, lea los [conceptos básicos de Key Vault](#).

Si no tiene una suscripción a Azure, cree una [cuenta gratuita](#).

Requisitos previos

Para Windows, Mac y Linux:

- [Git](#)
- Este tutorial requiere que se ejecute localmente la CLI de Azure. Debe tener instalada la versión 2.0.4 de la CLI de Azure o una versión posterior. Ejecute `az --version` para encontrar la versión. Si necesita instalarla o actualizarla, consulte [Instalación de la CLI de Azure 2.0](#).

Inicio de sesión en Azure

Para iniciar sesión en Azure mediante la CLI de Azure, escriba:

Azure CLI

```
az login
```

Creación de un grupo de recursos y de un almacén de claves

En este inicio rápido se usa un almacén de claves de Azure creado previamente. Puede crear un almacén de claves siguiendo los pasos descritos en el [inicio rápido de CLI de Azure](#), [inicio rápido de Azure PowerShell](#) o [inicio rápido de Azure Portal](#).

Como alternativa, puede ejecutar simplemente los siguientes comandos de la CLI de Azure o de Azure PowerShell.

Importante

Cada almacén de claves debe tener un nombre único. Reemplace <nombre-almacén de claves-único> por el nombre del almacén de claves en los siguientes ejemplos.

- [CLI de Azure](#)
- [Azure PowerShell](#)

Azure CLI

```
az group create --name "myResourceGroup" -l "EastUS"
```

```
az keyvault create --name "<your-unique-keyvault-name>" -g "myResourceGroup"
```

Rellenado del almacén de claves con un secreto

Vamos a crear un secreto llamado `mySecret` cuyo valor sea ¡Correcto! . Un secreto puede ser una contraseña, una cadena de conexión SQL o cualquier otra información que necesite mantener segura y disponible para la aplicación.

Para agregar un secreto al almacén de claves recién creado, use el comando siguiente:

- [CLI de Azure](#)
- [Azure PowerShell](#)

Azure CLI

```
az keyvault secret set --vault-name "<your-unique-keyvault-name>" --name "mySecret" --value "Success!"
```

Creación de una máquina virtual

Create una máquina virtual llamada myVM, para lo que debe usar uno de los siguientes métodos:

Linux

[CLI de Azure](#)

[PowerShell](#)

[Azure Portal](#)

Windows

[CLI de Azure](#)

[PowerShell](#)

[Portal de Azure](#)

Para crear una máquina virtual Linux mediante la CLI de Azure, use el comando [az vm create](#). En el ejemplo siguiente se agrega una cuenta de usuario llamada *azureuser*. El parámetro `--generate-ssh-keys` se usa para generar automáticamente una clave SSH y colocarla en la ubicación de la clave predeterminada (`~/ssh`).

Azure CLI

Pruébalo

```
az vm create \
  --resource-group myResourceGroup \
  --name myVM \
  --image UbuntuLTS \
  --admin-username azureuser \
  --generate-ssh-keys
```

Anote el valor de `publicIpAddress` en la salida.

Asignación de una identidad a la máquina virtual

Cree una identidad asignada por el sistema para la máquina virtual mediante el comando [az vm identity assign](#) de la CLI de Azure:

Azure CLICopiar

```
az vm identity assign --name "myVM" --resource-group "myResourceGroup"
```

Tenga en cuenta la identidad asignada por el sistema que se muestra en el código siguiente. La salida del comando anterior sería:

Resultados

```
{  
  "systemAssignedIdentity": "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",  
  "userAssignedIdentities": {}  
}
```

Asignación de permisos a la identidad de máquina virtual

Ahora puede asignar los permisos de la identidad creada anteriormente al almacén de claves mediante la ejecución del comando siguiente:

Azure CLI

```
az keyvault set-policy --name "<your-unique-keyvault-name>" --object-id  
"<systemAssignedIdentity>" --secret-permissions get list
```

Inicio de sesión en la máquina virtual

Para iniciar sesión en la máquina virtual, siga las instrucciones que encontrará en el artículo en el que se explica la [conexión y el inicio de sesión en una máquina virtual Linux en Azure](#), o bien en el que se explica la [conexión y el inicio de sesión en una máquina virtual Windows en Azure](#).

Para iniciar sesión en una máquina virtual Linux, puede usar el comando ssh con el valor de <publicIpAddress> que se proporciona en el paso [Creación de una máquina virtual](#):

terminal

```
ssh azureuser@<PublicIpAddress>
```

Instalación de bibliotecas de Python en la máquina virtual

En la máquina virtual, instale las dos bibliotecas de Python que vamos a usar en el script de Python: azure-keyvault-secrets y azure.identity.

En una máquina virtual Linux, por ejemplo, se puede usar pip3 para instalarlas:

Bash

```
pip3 install azure-keyvault-secrets
```

```
pip3 install azure.identity
```

Creación y edición del script de Python de ejemplo

En la máquina virtual, cree un archivo de Python llamado `sample.py`. Posteriormente, edítelo para que contenga el siguiente código, reemplazando `<nombre-almacén de claves-único>` por el nombre de su almacén de claves:

Python

```
from azure.keyvault.secrets import SecretClient

from azure.identity import DefaultAzureCredential

keyVaultName = "<your-unique-keyvault-name>"
KVUri = f"https://{keyVaultName}.vault.azure.net"
secretName = "mySecret"

credential = DefaultAzureCredential()
client = SecretClient(vault_url=KVUri, credential=credential)
retrieved_secret = client.get_secret(secretName)

print(f"The value of secret '{secretName}' in '{keyVaultName}' is: '{retrieved_secret.value}'")
```

Ejecución de la aplicación de Python de ejemplo

Por último, ejecute `sample.py`. Si todo ha ido bien, debería devolver el valor de su secreto:

Bash

```
python3 sample.py
```

```
The value of secret 'mySecret' in '<your-unique-keyvault-name>' is: 'Success!'
```

Limpieza de recursos

Cuando ya no son necesarios, elimine la máquina virtual y el almacén de claves. Puede hacerlo rápidamente, solo debe eliminar el grupo de recursos al que pertenecen:

Azure CLI

```
az group delete -g myResourceGroup
```


Exploración del módulo de seguridad de hardware de Azure

Azure Dedicated HSM ofrece almacenamiento de claves criptográficas en Azure. Dedicated HSM cumple los requisitos de seguridad más estrictos. Es la solución idónea para los clientes que necesitan dispositivos validados con la certificación FIPS 140-2 nivel 3 y un control completo y exclusivo del dispositivo HSM.

Casos en los que está indicado

Azure Dedicated HSM es el más adecuado para escenarios de "migración mediante lift-and-shift" que requieren un acceso directo y exclusivo a los dispositivos HSM. Entre los ejemplos se incluyen:

- Migración de aplicaciones locales a Azure Virtual Machines
- Migración de aplicaciones de Amazon AWS EC2 a máquinas virtuales que usan el servicio AWS Cloud HSM Classic
- Ejecución de un software empaquetado como Apache/Ngnix SSL Offload, Oracle TDE y ADCS en Azure Virtual Machines

Casos en los que no está indicado

Azure Dedicated HSM no es idóneo para el siguiente tipo de escenario: Los servicios en la nube de Microsoft que admiten el cifrado con claves administradas por los clientes (como Azure Information Protection, Azure Disk Encryption, Azure Data Lake Storage, Azure Storage, Azure SQL Database y clave de cliente para Office 365) no están integrados con Azure Dedicated HSM.