

Revisión de la Plataforma de identidad de Microsoft

Plataforma de identidad de Microsoft es una evolución de la plataforma para desarrolladores de Azure Active Directory (Azure AD). Permite que los desarrolladores compilen aplicaciones que inician sesión de usuarios, obtienen tokens para llamar a API como Microsoft Graph o a API que los desarrolladores hayan creado. Consiste en un servicio de autenticación, bibliotecas de código abierto, registro y configuración de aplicaciones (a través de un portal para desarrolladores y una API de aplicación), documentación completa para desarrolladores, ejemplos de inicio rápido, ejemplos de código, tutoriales, guías paso a paso y otros contenidos para desarrolladores. La plataforma de identidad de Microsoft admite los protocolos estándar del sector como OAuth 2.0 y OpenID Connect.

Hasta ahora, la mayoría de los desarrolladores han trabajado con la plataforma Azure AD v1.0 para autenticar cuentas profesionales y educativas (aprovisionadas por Azure AD) mediante la solicitud de tokens del punto de conexión de Azure AD v1.0, con la Biblioteca de autenticación de Azure AD (ADAL), Azure Portal para el registro y la configuración de aplicaciones y Microsoft Graph API para la configuración de aplicaciones mediante programación.

Con la Plataforma de identidad de Microsoft (versión 2.0) unificada, puede escribir código una vez y autenticar cualquier identidad de Microsoft en la aplicación. Si hay varias plataformas, se recomienda usar la Biblioteca de autenticación de Microsoft (MSAL) de código abierto, que es totalmente compatible, con los puntos de conexión de la plataforma de identidad. MSAL es fácil de usar, ofrece experiencias increíbles de inicio de sesión único (SSO) para los usuarios, lo ayuda a alcanzar una alta confiabilidad y rendimiento y se desarrolla con el Ciclo de vida de desarrollo de seguridad (SDL) de Microsoft. Al llamar a las API, puede configurar la aplicación para que use el consentimiento incremental, lo que permite retrasar la solicitud de consentimiento para ámbitos más invasivos hasta que el uso de la aplicación lo garantice en el entorno de ejecución. MSAL también es compatible con Azure Active Directory B2C, por lo que los clientes usan las identidades de la cuenta de redes sociales, corporativa o local que prefieran para obtener el acceso mediante inicio de sesión único a sus aplicaciones y API.

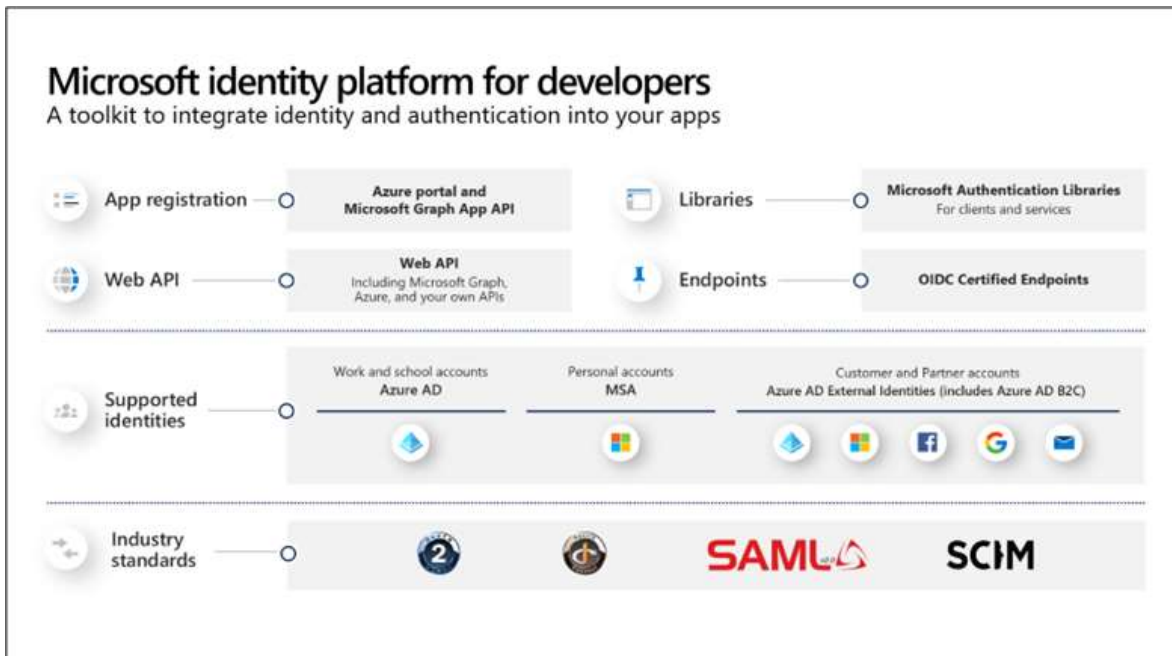
Con la Plataforma de identidad de Microsoft, se puede expandir el alcance a estos tipos de usuarios:

- Cuentas profesionales y educativas (cuentas aprovisionadas por Azure AD).
- Cuentas personales (como Outlook.com o Hotmail.com)
- Los clientes que traen su propio correo electrónico o identidad en redes sociales (como LinkedIn, Facebook, Google) mediante MSAL y Azure AD B2C

Puede usar Azure Portal para registrar y configurar la aplicación, además de usar Microsoft Graph API para la configuración de aplicaciones mediante programación.

Plataforma de identidad de Microsoft

En el diagrama siguiente se muestra la experiencia de identidad de Microsoft en un nivel alto, incluida la experiencia de registro de aplicaciones, los kits de desarrollo de software (SDK), los puntos de conexión y las identidades admitidas.



La plataforma de identidad de Microsoft tiene dos puntos de conexión (v1.0 y v2.0), pero, al desarrollar una aplicación nueva, se recomienda usar el punto de conexión v2.0 (valor predeterminado) para obtener las ventajas de las características y funcionalidades más recientes:

MSAL se puede usar en muchos escenarios de aplicación, incluidos:

- Aplicaciones de una sola página (JavaScript)
- Web app signing in users (Aplicación web que inicia sesión de los usuarios)
- Web application signing in a user and calling a web API on behalf of the user (Aplicación web que inicia sesión de un usuario y llama a una API web en nombre del usuario)
- Protecting a web API so only authenticated users can access it (Protección de una API web para que solo los usuarios autenticados puedan acceder a ella)
- API web que llama a otra API web de bajada en nombre del usuario que inició sesión.
- Desktop application calling a web API on behalf of the signed-in user (Aplicación de escritorio que llama a una API web en nombre del usuario que inició sesión)
- Aplicación móvil que llama a una API web en nombre del usuario que inició sesión de manera interactiva.
- Desktop/service daemon application calling web API on behalf of itself (Aplicación de demonio de escritorio/servicio que llama a una API web en su propio nombre)

Lenguajes y plataformas

Library	Plataformas y marcos compatibles
MSAL para Android	Android
MSAL Angular	Aplicaciones de una sola página con los marcos de trabajo Angular y Angular.js
MSAL para iOS y macOS	iOS y macOS
MSAL Go (versión preliminar)	Windows, macOS, Linux
Java de MSAL	Windows, macOS, Linux
MSAL.js	Marcos de trabajo JavaScript/TypeScript, como Vue.js, Ember.js, o Durandal.js
MSAL.NET	.NET Framework, .NET Core, Xamarin Android, Xamarin iOS, Plataforma universal de Windows
MSAL Node	Aplicaciones web con Express, aplicaciones de escritorio con Electron y aplicaciones de consola multiplataforma
Python de MSAL	Windows, macOS, Linux
MSAL React	Aplicaciones de una sola página con React y bibliotecas basadas en React (Next.js y Gatsby.js)

Migración de aplicaciones que usan ADAL a MSAL

La Biblioteca de autenticación de Active Directory (ADAL) se integra con el punto de conexión de Azure AD para desarrolladores (v1.0), donde MSAL se integra con la Plataforma de identidad de Microsoft. El punto de conexión v1.0 admite cuentas profesionales, pero no cuentas personales. El punto de conexión v2.0 es la unión de las cuentas personales y de las cuentas profesionales de Microsoft en un sistema de autenticación único. Con MSAL además puede obtener autenticaciones para Azure AD B2C.

Exploración de escenarios de aplicación de Azure AD

Cualquier aplicación que externalice la autenticación a Azure AD debe registrarse en un directorio. Este paso implica informar a Azure AD sobre la aplicación, incluido lo siguiente:

Escenarios de aplicación de Azure AD		
Front-end	Autenticación	Back-end
Una aplicación de página única son servidores front-end que se ejecutan en un explorador	Punto de conexión de autorización de Azure AD	API Web
Las aplicaciones web son aplicaciones que autentican a un usuario de un explorador web en una aplicación web	WS-Federation de Azure AD o punto de conexión SAML	Aplicación web
Las aplicaciones nativas son aplicaciones que llaman a una API web en nombre de un usuario	Punto de conexión de autorización y token de autenticación de Azure AD	API Web
Las aplicaciones de API web son aplicaciones web que necesitan obtener recursos de una API web	Punto de conexión de autorización y token de autenticación de Azure AD	Aplicación web y API web
Las aplicaciones de servicio a servicio son aplicaciones de servidor o de demonio que necesitan obtener recursos de una API web	Punto de conexión de autorización y token de autenticación de Azure AD	API Web

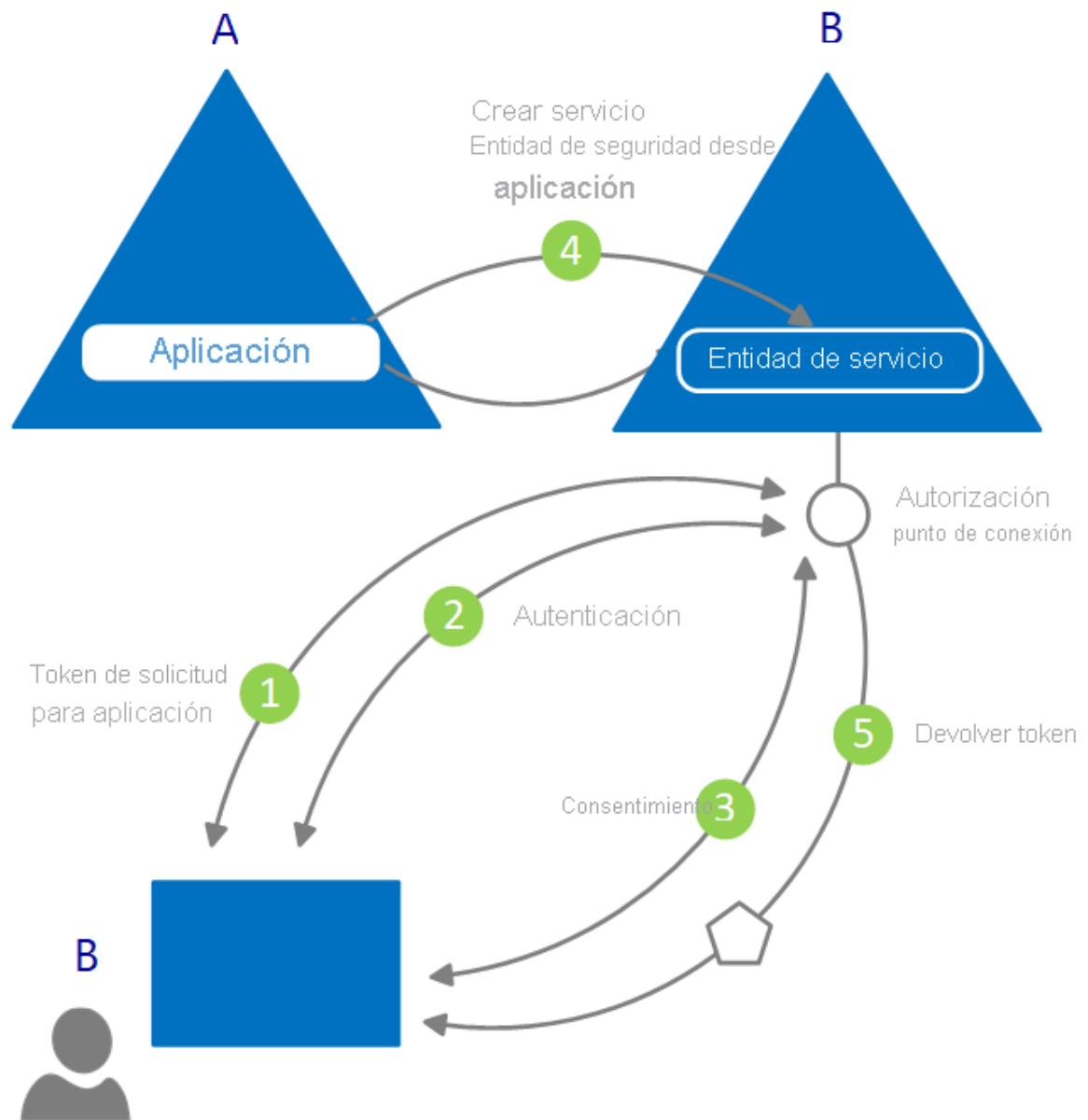
Azure AD representa aplicaciones siguiendo un modelo específico que se ha diseñado para satisfacer dos funciones principales:

- Identificar la aplicación según los protocolos de autenticación que admite. Esto implica enumerar todos los identificadores, las direcciones URL, los secretos y la información relacionada que Azure AD necesita en tiempo de autenticación. Aquí, Azure AD:
 - Contiene todos los datos necesarios para admitir la autenticación en tiempo de ejecución.

- Contiene todos los datos para decidir los recursos a los que puede requerir acceso una aplicación, si esta debe satisfacer una solicitud determinada y en qué circunstancias debe satisfacerla.
 - Proporciona la infraestructura para implementar el aprovisionamiento de aplicaciones tanto dentro del inquilino del desarrollador de la aplicación como para cualquier otro inquilino de Azure AD.
- Controlar el consentimiento del usuario durante el tiempo de solicitud del token y facilitar el aprovisionamiento dinámico de aplicaciones entre inquilinos. Aquí, Azure AD:
 - Permite a los usuarios y administradores conceder o denegar el consentimiento dinámicamente para que la aplicación acceda a recursos en su nombre.
 - Permite a los administradores decidir en última instancia qué se permite a las aplicaciones que hagan, qué usuarios pueden utilizar aplicaciones específicas y cómo se accede a los recursos de directorio.

En Azure AD, un objeto de aplicación describe una aplicación como entidad abstracta. Los desarrolladores trabajan con aplicaciones. En el momento de la implementación, Azure AD usa un objeto de aplicación específico como plano técnico para crear una entidad de servicio, que representa una instancia concreta de una aplicación en un directorio o inquilino. Es la entidad de servicio la que define lo que la aplicación puede hacer en un directorio de destino específico, quién puede usarla y a qué recursos tiene acceso, entre otras cosas. Azure AD crea una entidad de servicio desde un objeto de aplicación a través del consentimiento.

En el diagrama siguiente se muestra un flujo de aprovisionamiento de Azure AD simplificado que se controla por consentimiento.



En este flujo de aprovisionamiento:

1. Un usuario de B intenta iniciar sesión con la aplicación.
2. Azure AD obtiene las credenciales del usuario y las comprueba.
3. Azure AD solicita al usuario su consentimiento para que la aplicación pueda acceder al inquilino B.
4. Azure AD usa el objeto de aplicación en A como plano técnico para crear una entidad de servicio en B.
5. El usuario recibe el token solicitado.

Puede repetir este proceso tantas veces como quiera para otros inquilinos (C, D, etc.). El directorio A conserva el plano técnico para la aplicación (objeto de aplicación). Puede aplicarse a los usuarios y administradores del resto de inquilinos en los que la aplicación ha obtenido el consentimiento para conservar el control sobre lo que esta puede hacer a través del objeto de entidad de servicio correspondiente en cada inquilino.

Cuando una aplicación tiene permiso para acceder a los recursos de un inquilino (tras el registro o consentimiento), se crea un objeto de entidad de seguridad de servicio. La **entidad ServicePrincipal** de Microsoft Graph define el esquema para las propiedades de un objeto de entidad de servicio.

Registro de una aplicación con el registro de aplicaciones

Para obtener el funcionamiento más seguro, registre la aplicación con la Plataforma de identidad de Microsoft.

Para que la aplicación pueda obtener un token de la Plataforma de identidad de Microsoft, debe registrarse en Azure Portal. El registro integra la aplicación en la Plataforma de identidad de Microsoft y establece la información que usa para obtener tokens, incluida la siguiente:

- **Id. de aplicación:** identificador único asignado por la Plataforma de identidad de Microsoft.
- **URI o dirección URL de redireccionamiento:** uno o varios puntos de conexión en los que la aplicación recibirá respuestas de la Plataforma de identidad de Microsoft. (En el caso de las aplicaciones nativas y móviles, se trata de un URI asignado por la Plataforma de identidad de Microsoft).
- **Secreto de aplicación:** una contraseña o un par de claves pública/privada que la aplicación usa para autenticarse con la Plataforma de identidad de Microsoft. (No es necesario para las aplicaciones nativas o móviles).

[Inicio](#) [Registros de aplicaciones](#)

Registro de una aplicación

 Si va a crear una aplicación para los usuarios externos distribuida por Microsoft, debe registrarla como una aplicación propia para cumplir todas las directivas de seguridad, privacidad y cumplimiento. Lea nuestra [política de sobre la toma de decisiones](#)

* Nombre

Se trata del nombre para mostrar accesible por los usuarios para esta aplicación. Se puede cambiar posteriormente.

Tipos de cuenta admitidos

¿Quién puede usar esta aplicación o acceder a esta API?

- ☒ Solo las cuentas de este directorio organizativo (solo de Microsoft 365)
- ☐ Cuentas en cualquier directorio organizativo (cualquier directorio de Azure AD)
- ☐ Cuentas en cualquier directorio organizativo (cualquier directorio de Azure AD) y cuentas de Microsoft personales (como Skype o Xbox)
- ☐ Solo cuentas personales de Microsoft

[Ayudarme a elegir...](#)

URI de redireccionamiento (opcional)

Devolveremos la respuesta de autenticación a esta dirección URI después de autenticar correctamente al usuario. Este dato es opcional y se puede cambiar más tarde, pero se necesita un valor para la mayoría de los escenarios de autenticación.

Web

▼

p.ej. https://example.com/auth

Registre una aplicación en la que esté trabajando aquí. Integre aplicaciones de la galería y otras aplicaciones de fuera de su organización agregándolas desde

[Al continuar, acepta las directivas de la plataforma Microsoft.](#) 

Registrar

Obtención de un token de acceso

Al igual que la mayoría de los desarrolladores, probablemente use las bibliotecas de autenticación para administrar las interacciones de los tokens con la Plataforma de identidad de Microsoft. Las bibliotecas de autenticación resumen muchos detalles del protocolo, como la validación, la administración de cookies, el almacenamiento en caché de tokens y el mantenimiento de conexiones seguras, lejos del desarrollador y le permiten centrar el desarrollo en la aplicación. Microsoft publica bibliotecas cliente de código abierto y middleware de servidor.

Configuración de permisos de Microsoft Graph

Microsoft Graph expone permisos detallados que controlan el acceso que las aplicaciones tienen a recursos como los usuarios, los grupos y el correo. Como desarrollador, usted decide qué permisos se van a solicitar para Microsoft Graph. Cuando un usuario (o un administrador, en algunos casos) inicia sesión en la aplicación, se le da la oportunidad de dar su consentimiento a estos permisos. Si el usuario da su consentimiento, se concede a la aplicación acceso a los recursos y a las API que ha solicitado. En el caso de las aplicaciones en las que los usuarios no inician sesión,

un administrador puede otorgar su consentimiento previo a los permisos cuando se instala la aplicación.

Microsoft Graph tiene dos tipos de permisos:

- **Permisos delegados:** se utilizan en aplicaciones que tienen un usuario con la sesión iniciada. En estas aplicaciones, el usuario o un administrador da su consentimiento a los permisos que la aplicación solicita y esta puede actuar como el usuario que ha iniciado sesión al realizar llamadas a Microsoft Graph. Los usuarios que no son administradores pueden aceptar algunos permisos delegados, pero otros con mayores privilegios requieren el consentimiento del administrador.
- **Permisos de aplicación:** los usan las aplicaciones que se ejecutan sin la presencia de un usuario con la sesión iniciada; por ejemplo, las aplicaciones que se ejecutan como demonios o servicios en segundo plano. Los permisos de aplicación solo pueden ser aceptados por un administrador.

Los permisos efectivos son los que la aplicación tendrá al realizar solicitudes a Microsoft Graph. Es importante comprender la diferencia entre los permisos delegados y de aplicación que se conceden a la aplicación y sus permisos efectivos al realizar llamadas a Microsoft Graph.

En el caso de los permisos delegados, los permisos efectivos de la aplicación serán la intersección de los permisos delegados que se han concedido a la aplicación (a través de consentimiento) y los privilegios del usuario que ha iniciado sesión actualmente. La aplicación nunca puede tener más privilegios que el usuario que tiene la sesión iniciada. Dentro de las organizaciones, los privilegios del usuario que tiene la sesión iniciada pueden determinarse mediante directivas o pertenencia a uno o varios roles de administrador.


Por ejemplo, suponga que a su aplicación se le ha concedido el permiso delegado User.ReadWrite.All en Microsoft Graph. Este permiso concede a su aplicación de forma nominal un permiso para leer y actualizar el perfil de cada usuario de una organización. Si el usuario que inició sesión es un administrador global, la aplicación podrá actualizar el perfil de cada usuario de la organización. Sin embargo, si el usuario con la sesión iniciada no pertenece a un rol de administrador, la aplicación podrá actualizar solo el perfil del usuario que tiene la sesión iniciada. No podrá actualizar los perfiles de otros usuarios de la organización, porque el usuario para el que tiene permiso para actuar en su nombre no tiene tales privilegios. Para los permisos de aplicación, los permisos efectivos de la aplicación serán el nivel completo de privilegios que concede el permiso. Por ejemplo, una aplicación que tiene el permiso de aplicación User.ReadWrite.All puede actualizar el perfil de cada usuario de la organización.

Solicitar permisos de API

Selección de una API


API de Microsoft API usadas en mi organización Mis API

API de Microsoft más usadas




Microsoft Graph

Aproveche la gran cantidad de datos disponibles en Office 365, Enterprise Mobility + Security y Windows 10. Acceda a Azure AD, Excel, Intune, Outlook/Exchange, OneDrive, OneNote, SharePoint, Planner y muchos más mediante un único punto de conexión.




Azure Batch

Permite programar aplicaciones HPC y paralelas a gran escala en la nube




Azure Data Catalog

Ofrece acceso mediante programación a recursos de datos para registrar, anotar y buscar activos de datos




Azure Data Explorer

Consultas ad-hoc sobre terabytes de datos para crear soluciones de análisis complejas y prácticamente en tiempo real




Azure Data Explorer (con autenticación multifactor)

Consultas ad-hoc sobre terabytes de datos para crear soluciones de análisis complejas y prácticamente en tiempo real.



Azure Key Vault

Permite administrar los almacenes de claves, así como las claves, secretos y certificados correspondientes en los almacenes de claves

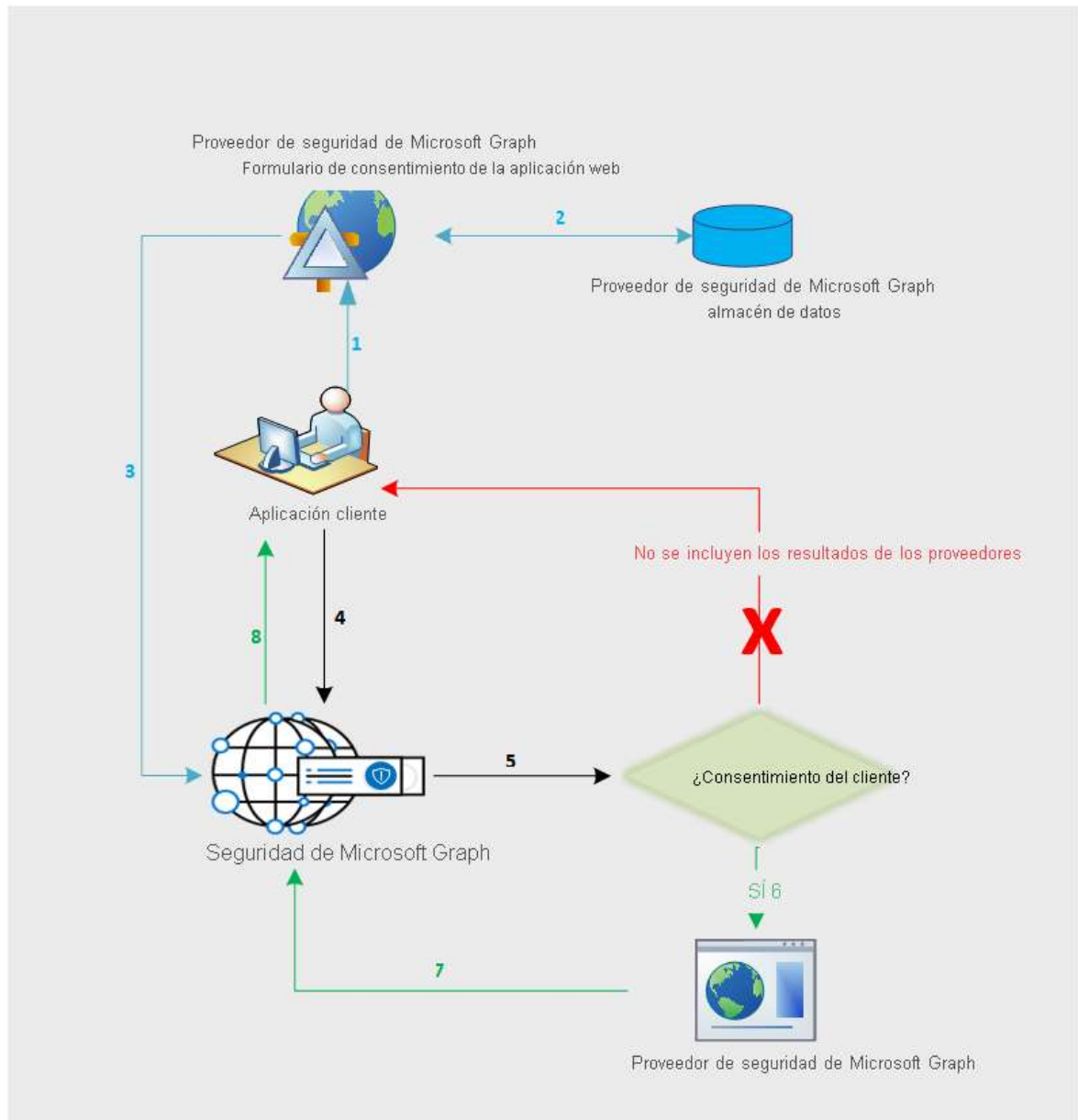


Azure Storage

Objeto seguro escalable de forma masiva y Data Lake Storage para datos no estructurados y semiestructurados

Microsoft Graph API

Puede usar la API Microsoft Graph Security para conectar productos, servicios y asociados de seguridad de Microsoft a fin de simplificar las operaciones de este campo y mejorar las funcionalidades de detección y protección frente a amenazas y de respuesta a estas. La API Microsoft Graph Security es un servicio intermediario (o agente) que proporciona una única interfaz de programación para conectar varios proveedores de Microsoft Graph Security (también denominados proveedores de seguridad o proveedores). La API Microsoft Graph Security federa las solicitudes para todos los proveedores del ecosistema de Microsoft Graph Security. Esto se basa en el consentimiento del proveedor de seguridad que la aplicación proporciona, como se muestra en el diagrama siguiente. El flujo de trabajo de consentimiento solo se aplica a los proveedores que no son de Microsoft.



A continuación, se muestra una descripción del flujo:

1. El usuario de la aplicación inicia sesión en la aplicación del proveedor para ver el formulario de consentimiento de este. La interfaz de usuario o la experiencia del formulario de consentimiento es propiedad del proveedor y solo se aplica a los proveedores que no son de Microsoft a fin de obtener el consentimiento explícito de sus clientes para enviar solicitudes a la API Microsoft Graph Security.
2. El consentimiento del cliente se almacena en el lado del proveedor.
3. El servicio de consentimiento del proveedor llama a la API Microsoft Graph Security para informar de la aprobación del consentimiento para el cliente respectivo.
4. La aplicación envía una solicitud a la API Microsoft Graph Security.

5. La API Microsoft Graph Security comprueba la información de consentimiento de este cliente asignada a varios proveedores.
6. La API Microsoft Graph Security llama a todos los proveedores a los que el cliente ha dado su consentimiento explícito a través de la experiencia de consentimiento de proveedores.
7. La respuesta se devuelve desde todos los proveedores con consentimiento para ese cliente.
8. La respuesta del conjunto de resultados se devuelve a la aplicación.
9. Si el cliente no ha dado su consentimiento a ningún proveedor, no se incluirá ningún resultado de esos proveedores en la respuesta.

La API Microsoft Graph Security facilita la conexión con las soluciones de seguridad de Microsoft y sus asociados. Permite comprender y enriquecer de forma más fácil el valor de estas soluciones. Para conectar fácilmente con la API Microsoft Graph Security, use uno de los enfoques siguientes, en función de sus requisitos:

Motivos para usar la API Microsoft Graph Security

- Escriba código: encuentre ejemplos de código en C#, Java y NodeJS, entre otros.
- Conéctese mediante scripts: encuentre ejemplos de PowerShell.
- Arrastre y coloque en flujos de trabajo y cuadernos de estrategias: use conectores de Microsoft Graph Security para Azure Logic Apps, Microsoft Flow y PowerApps.
- Introduzca datos en informes y paneles: use el conector de Microsoft Graph Security para Power BI.
- Conéctese con instancias de Jupyter Notebook: encuentre ejemplos de Jupyter Notebook.

Unificación y normalización del seguimiento de alertas

Conéctese una vez para integrar alertas de cualquier solución de seguridad integrada de Microsoft Graph y mantener sincronizados las asignaciones y el estado de las alertas en todas las soluciones. También puede transmitir alertas a las soluciones de administración de eventos e información de seguridad (SIEM), como Splunk, con los conectores de la API Microsoft Graph Security.

Correlación de las alertas de seguridad para mejorar la protección y la respuesta contra amenazas

Use un esquema de alertas unificado para que sea más fácil poner en correlación las alertas entre las distintas soluciones de seguridad. Esto no solo le permite recibir información de las alertas que requieren acción, sino que permite a los analistas de seguridad adaptar y enriquecer las alertas con información de los recursos y los usuarios, lo que permite una respuesta a las amenazas y una protección de los recursos más rápidas.

Actualización de las etiquetas, el estado y las asignaciones de las alertas

Etiquete las alertas con contexto adicional o inteligencia sobre amenazas para determinar la respuesta y la corrección. Asegúrese de que se capturen los comentarios sobre las alertas para obtener visibilidad en todos los flujos de trabajo. Mantenga sincronizados el estado y las asignaciones para que todas las soluciones integradas reflejen el estado actual. Use suscripciones de webhook para que se le notifiquen los cambios.

Desbloqueo del contexto de seguridad para impulsar la investigación

Profundice en el inventario relacionado relevante para la seguridad (como usuarios, hosts y aplicaciones) y, después, agregue contexto organizativo de otros proveedores de Microsoft Graph (Azure AD, Microsoft Intune, Microsoft 365) para reunir el contexto empresarial y de seguridad y mejorar la respuesta frente a amenazas.

Habilitar identidades administradas

Un desafío común al compilar aplicaciones en la nube consiste en el modo de administrar las credenciales del código para autenticar los servicios en la nube. Proteger las credenciales es una tarea esencial. Lo ideal sería que nunca aparecieran en las estaciones de trabajo de los desarrolladores y que no se controlaran en el código fuente. Azure Key Vault proporciona una manera segura de almacenar credenciales, secretos y otras claves pero el código tiene que autenticarse en Key Vault para recuperarlos.

Identidades administradas para recursos de Azure es el nuevo nombre del servicio anteriormente conocido como la característica Managed Service Identity (MSI) para recursos de Azure en Azure Active Directory (Azure AD) que resuelve el problema indicado anteriormente. Esta característica proporciona a los servicios de Azure una identidad de sistema administrada automáticamente en Azure AD. Puede usar esta identidad para autenticarse en cualquier servicio que admita la autenticación de Azure AD, incluido Key Vault, sin necesidad de credenciales en el código.

La característica Managed Identities for Azure Resources se incluye gratuitamente en Azure AD con las suscripciones de Azure. No hay ningún costo adicional.

Terminología

Los siguientes términos se usan en las identidades administradas para el conjunto de documentación de los recursos de Azure:

- **Id. de cliente:** un identificador único que genera Azure AD y que está asociado a una aplicación y entidad de servicio durante su aprovisionamiento inicial.
- **Id. de entidad de seguridad:** identificador de objeto del objeto de entidad de servicio de la identidad administrada que se usa para conceder acceso basado en roles a los recursos de Azure.
- **Azure Instance Metadata Service (IMDS) :** un punto de conexión de REST al que pueden acceder todas las máquinas virtuales IaaS creadas mediante Azure Resource Manager. El

punto de conexión está disponible en una dirección IP no enrutable conocida (169.254.169.254) a la que se puede acceder solo desde dentro de la máquina virtual.

Funcionamiento de las identidades administradas para recursos de Azure

Hay dos tipos de identidades administradas:

- **Una identidad administrada asignada por el sistema** se habilita directamente en una instancia de servicio de Azure. Cuando se habilita la identidad, Azure crea una identidad para la instancia del servicio en el inquilino de Azure AD de confianza de la suscripción de la instancia. Una vez creada la identidad, las credenciales se aprovisionan en la instancia. El ciclo de vida de una identidad administrada asignada por el sistema está vinculado directamente a la instancia de servicio de Azure en que está habilitada. Si se elimina la instancia, Azure limpia automáticamente las credenciales y la identidad en Azure AD.
- **Una identidad administrada asignada por el usuario** se crea como recurso independiente de Azure. Mediante un proceso de creación, Azure crea una identidad en el inquilino de Azure AD de confianza para la suscripción que se utiliza. Una vez creada la identidad, esta puede asignarse a una o varias instancias de servicio de Azure. El ciclo de vida de una identidad asignada por el usuario no se administra junto con el ciclo de vida de las instancias de servicio de Azure a las que se asigna.

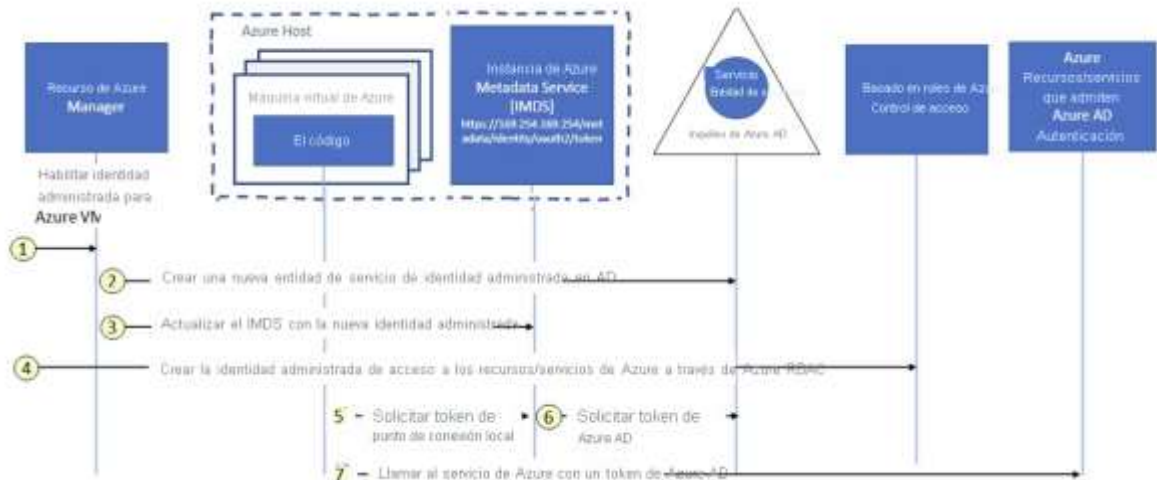
Internamente, las identidades administradas son entidades de servicio de un tipo especial, que se bloquean para que solo puedan usarse con recursos de Azure. Cuando se elimina la identidad administrada, se quita automáticamente la entidad de servicio correspondiente. Además, cuando se crea una identidad asignada por el usuario o asignada por el sistema, el proveedor de recursos de identidades administradas (MSRP) genera un certificado internamente para esa identidad.

El código puede usar una identidad administrada para solicitar tokens de acceso de los servicios que admiten la autenticación de Azure AD. Azure se encarga de rotar las credenciales que usa la instancia del servicio.

Rotación de credenciales

La rotación de credenciales la controla el proveedor de recursos que hospeda el recurso de Azure. La rotación predeterminada de la credencial se produce cada 46 días. Es responsabilidad del proveedor de recursos llamar para obtener nuevas credenciales, por lo que el proveedor de recursos puede tardar más de 46 días en hacerlo.

En el diagrama siguiente se muestra cómo funcionan las identidades de servicio administradas con máquinas virtuales (VM) de Azure:




Funcionamiento de una identidad administrada asignada por el sistema con una máquina virtual de Azure

1. Azure Resource Manager recibe una solicitud para habilitar la identidad administrada asignada por el sistema de una máquina virtual.
2. Entonces crea una entidad de servicio en Azure AD para la identidad de la máquina virtual. La entidad de servicio se crea en el inquilino de Azure AD que sea de confianza para la suscripción.
3. Azure Resource Manager configura la identidad de la máquina virtual mediante la actualización del punto de conexión de identidad de Azure Instance Metadata Service con el identificador de cliente y el certificado de la entidad de servicio.
4. Ahora que la máquina virtual tiene una identidad, se usa la información de la entidad de servicio para conceder a la máquina virtual acceso a los recursos de Azure. Para llamar a Azure Resource Manager, use el control de acceso basado en rol (RBAC) de Azure AD para asignar el rol apropiado a la entidad de servicio de la máquina virtual. Para llamar a Key Vault, conceda a su código acceso al secreto o a la clave específicos en Key Vault.
5. El código que se ejecuta en la máquina virtual puede solicitar un token del punto de conexión de Azure Instance Metadata Service, accesible únicamente desde dentro de la máquina virtual: <https://169.254.169.254/metadata/identity/oauth2/token>
 - El parámetro del recurso especifica el servicio al que se va a enviar el token. Para autenticarse en Azure Resource Manager, use `resource=https://management.azure.com/..`
 - El parámetro de versión de API especifica la versión de IMDS, use `api-version=2018-02-01` o posterior.
6. Se realiza una llamada a Azure AD para solicitar un token de acceso, tal y como se especifica en el paso 5, con el identificador de cliente y el certificado configurado en el paso 3. Azure AD devuelve un token de acceso JSON Web Token (JWT).

7. El código envía el token de acceso en una llamada a un servicio que admite la autenticación de Azure AD.

Implementación de certificados de aplicación web


Puede restringir el acceso a una aplicación de Azure App Service habilitando diferentes tipos de autenticación. Una manera de hacerlo consiste en solicitar un certificado de cliente cuando la solicitud de cliente se realice a través de TLS/SSL y validar el certificado. Este mecanismo se denomina autenticación mutua de TLS o autenticación de certificado de cliente. En este artículo se muestra cómo configurar la aplicación para que use la autenticación de certificados de cliente. Si accede a su sitio a través de HTTP y no de HTTPS, no recibirá ningún certificado de cliente. Por lo tanto, si la aplicación requiere certificados de cliente, no debe permitir solicitudes a la aplicación mediante HTTP.

 **HumanResources**
App Service

Configuración TLS/SSL

Enlaces

Certificados de clave privada (.pfx) Certificados de clave pública (.cer)



Configuración del protocolo


La configuración del protocolo es global y se aplica a todos los enlaces definidos por la aplicación.

Solo HTTPS: ⓘ

Desactivado Activado

Versión de TLS mínima ⓘ

1,0 1.1 1.2



Enlaces TLS/SSL

Los enlaces permiten especificar el certificado que se utilizará para responder a las solicitudes a un nombre de host específico a través de HTTPS. El enlace TLS/SSL requiere un certificado emitido para el nombre de host específico.

+

Agregar enlace TLS/SSL

☐

Nombre de host

Huella digital de certificado privado

Tipo de TLS/SSL

No hay ningún enlace TLS/SSL configurado para la aplicación.

Habilitación de certificados de cliente

Para configurar la aplicación de forma que requiera certificados de cliente, puede activar **Requerir el certificado entrante** mediante la selección de Configuración > Configuración general en Azure Portal, o bien establecer el valor `clientCertEnabled` de la aplicación en `true`.

Exclusión de rutas de acceso para que no requieran autenticación

Si habilita la autenticación mutua en su aplicación, todas las rutas de acceso situadas bajo la raíz de la aplicación necesitarán un certificado de cliente para obtener acceso. Si desea

permitir que ciertas rutas permanezcan abiertas para el acceso anónimo, puede definir rutas de exclusión al configurar la aplicación.

Se pueden configurar rutas de exclusión si se selecciona **Configuración > Configuración general** y se define una ruta de exclusión. En este ejemplo, todos los elementos bajo la ruta de acceso /public de la aplicación no solicitan un certificado de cliente

Acceso al certificado de cliente

En App Service, la terminación TLS de la solicitud tiene lugar en el equilibrador de carga de front-end. Al reenviar la solicitud al código de la aplicación con los certificados de cliente habilitados, App Service inserta un encabezado de solicitud X-ARR-ClientCert con el certificado de cliente. App Service no hace nada con este certificado de cliente aparte de reenviarlo a la aplicación. El código de la aplicación es responsable de validar el certificado de cliente.
