Patrón de identidad federada

Delegue la autenticación a un proveedor de identidad externo. Esto puede simplificar el desarrollo, minimizar el requisito de administración de usuarios y mejorar la experiencia del usuario de la aplicación.

Contexto y problema

Los usuarios normalmente necesitan trabajar con múltiples aplicaciones proporcionadas y alojadas por diferentes organizaciones con las que tienen una relación comercial. Es posible que se requiera que estos usuarios utilicen credenciales específicas (y diferentes) para cada uno. Esto puede:

- Provocar una experiencia de usuario inconexa. Los usuarios a menudo olvidan las credenciales de inicio de sesión cuando tienen muchas diferentes.
- Exponga las vulnerabilidades de seguridad. Cuando un usuario abandona la empresa, la cuenta debe darse de baja de inmediato. Es fácil pasar esto por alto en las grandes organizaciones.
- Gestión de usuarios complicada. Los administradores deben administrar las credenciales de todos los usuarios y realizar tareas adicionales, como proporcionar recordatorios de contraseñas.

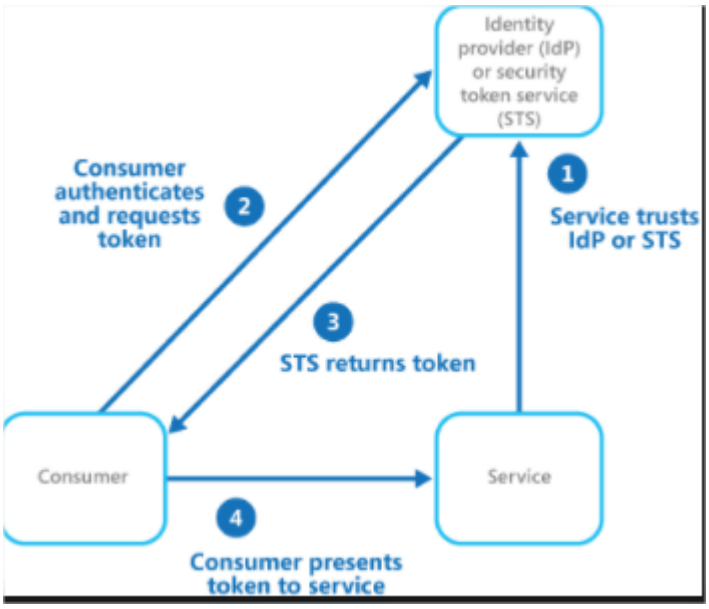
Los usuarios suelen preferir utilizar las mismas credenciales para todas estas aplicaciones.

Solución

Implemente un mecanismo de autenticación que pueda usar identidad federada. Separe la autenticación de usuario del código de la aplicación y delegue la autenticación a un proveedor de identidad de confianza. Esto puede simplificar el desarrollo y permitir que los usuarios se autentiquen utilizando una gama más amplia de **proveedores de identidad (IdP)** al tiempo que se minimizan los gastos administrativos. También le permite desvincular claramente la autenticación de la autorización.

Los proveedores de identidad de confianza incluyen directorios corporativos, servicios de federación locales, otros **servicios de token de seguridad (STS**) proporcionados por socios comerciales o proveedores de identidad social que pueden autenticar a los usuarios que tienen, por ejemplo, Microsoft, Google, Yahoo! O Facebook. cuenta.

La figura ilustra el patrón de identidad federada cuando una aplicación cliente necesita acceder a un servicio que requiere autenticación. La autenticación la realiza un IdP que trabaja en conjunto con un STS. El IdP emite tokens de seguridad que proporcionan información sobre el usuario autenticado. Esta información, a la que se hace referencia como notificaciones, incluye la identidad del usuario y también puede incluir otra información, como la pertenencia al rol y derechos de acceso más granulares.



Este modelo a menudo se denomina control de acceso basado en notificaciones. Las aplicaciones y los servicios autorizan el acceso a características y funcionalidades según las afirmaciones contenidas en el token. El servicio que requiere autenticación debe confiar en el IdP. La aplicación cliente se pone en contacto con el IdP que realiza la autenticación. Si la autenticación es exitosa, el IdP devuelve un token que contiene las afirmaciones que identifican al usuario al STS (tenga en cuenta que el IdP y el STS pueden ser el mismo servicio). El STS puede transformar y aumentar las reclamaciones en el token según reglas predefinidas, antes de devolverlo al cliente. La aplicación cliente puede luego pasar este token al servicio como prueba de su identidad.

Es posible que haya servicios de token de seguridad adicionales en la cadena de confianza. Por ejemplo, en el escenario que se describe más adelante, un STS local confía en otro STS que es responsable de acceder a un proveedor de identidad para autenticar al usuario. Este enfoque es común en escenarios empresariales donde hay un STS y un directorio local.

La autenticación federada proporciona una solución basada en estándares para el problema de confiar en las identidades en diversos dominios y puede admitir el inicio de sesión único. Se está volviendo más común en todos los tipos de aplicaciones, especialmente las aplicaciones alojadas en la nube, porque admite el inicio de sesión único sin requerir una conexión de red directa a los proveedores de identidad. El usuario no tiene que ingresar credenciales para cada aplicación. Esto aumenta la seguridad porque evita la creación de las credenciales necesarias para acceder a muchas aplicaciones diferentes y también oculta las credenciales del usuario a todos menos al proveedor de identidad original. Las aplicaciones solo ven la información de identidad autenticada contenida en el token.

La identidad federada también tiene la principal ventaja de que la administración de la identidad y las credenciales es responsabilidad del proveedor de identidad. No es necesario que la aplicación o el servicio proporcionen funciones de gestión de identidad. Además, en escenarios corporativos, el directorio corporativo no necesita conocer al usuario si confía en el proveedor de identidad. Esto elimina toda la sobrecarga administrativa de administrar la identidad del usuario dentro del directorio.

Problemas y consideraciones

Tenga en cuenta lo siguiente al diseñar aplicaciones que implementan la autenticación federada:

- La autenticación puede ser un único punto de falla. Si implementa su aplicación en varios centros de datos, considere implementar su mecanismo de administración de identidad en los mismos centros de datos para mantener la confiabilidad y disponibilidad de la aplicación.

- Las herramientas de autenticación permiten configurar el control de acceso en función de las reclamaciones de roles contenidas en el token de autenticación. Esto a menudo se denomina control de acceso basado en roles (RBAC) y puede permitir un nivel de control más granular sobre el acceso a funciones y recursos.

- A diferencia de un directorio corporativo, la autenticación basada en notificaciones que utiliza proveedores de identidad social no suele proporcionar información sobre el usuario autenticado que no sea una dirección de correo electrónico y quizás un nombre. Algunos proveedores de identidad social, como una cuenta de Microsoft, proporcionan solo un identificador único. La aplicación generalmente necesita mantener cierta información sobre los usuarios registrados y poder hacer coincidir esta información con el identificador contenido en las notificaciones del token. Por lo general, esto se hace mediante el registro cuando el usuario accede por primera vez a la aplicación, y luego se inyecta información en el token como reclamos adicionales después de cada autenticación.

- Si hay más de un proveedor de identidad configurado para el STS, debe detectar a qué proveedor de identidad se debe redirigir al usuario para la autenticación. Este proceso se llama descubrimiento del reino de origen. El STS podría hacer esto automáticamente basándose en una dirección de correo electrónico o nombre de usuario que el usuario proporciona, un subdominio de la aplicación a la que el usuario está accediendo, el alcance de la dirección IP del usuario o en el contenido de una cookie almacenada en el navegador. Por ejemplo, si el usuario ingresó una dirección de correo electrónico en el dominio de Microsoft, como usuario@live.com, el STS redirigirá al usuario a la página de inicio de sesión de la cuenta de Microsoft. En visitas posteriores, el STS podría usar una cookie para indicar que el último inicio de sesión fue con una cuenta de Microsoft. Si el descubrimiento automático no puede determinar el ámbito de origen,

Cuando usar este patrón

Este patrón es útil para escenarios como:

- Inicio de sesión único en la empresa . En este escenario, debe autenticar a los empleados para las aplicaciones corporativas que están alojadas en la nube fuera de los límites de seguridad corporativos, sin requerir que inicien sesión cada vez que visitan una aplicación. La experiencia del usuario es la misma que cuando se utilizan aplicaciones locales en las que se autentican al iniciar sesión en una red corporativa y, a partir de ese momento, tienen acceso a todas las aplicaciones relevantes sin necesidad de volver a iniciar sesión.

- Identidad federada con múltiples socios . En este escenario, debe autenticar tanto a los empleados corporativos como a los socios comerciales que no tienen cuentas en el directorio corporativo. Esto es común en aplicaciones de empresa a empresa, aplicaciones que se

integran con servicios de terceros y donde empresas con diferentes sistemas de TI han fusionado o compartido recursos.

- Identidad federada en aplicaciones SaaS . En este escenario, los proveedores de software independientes brindan un servicio listo para usar para múltiples clientes o inquilinos. Cada inquilino se autentica utilizando un proveedor de identidad adecuado. Por ejemplo, los usuarios comerciales usarán sus credenciales corporativas, mientras que los consumidores y clientes del inquilino usarán sus credenciales de identidad social.

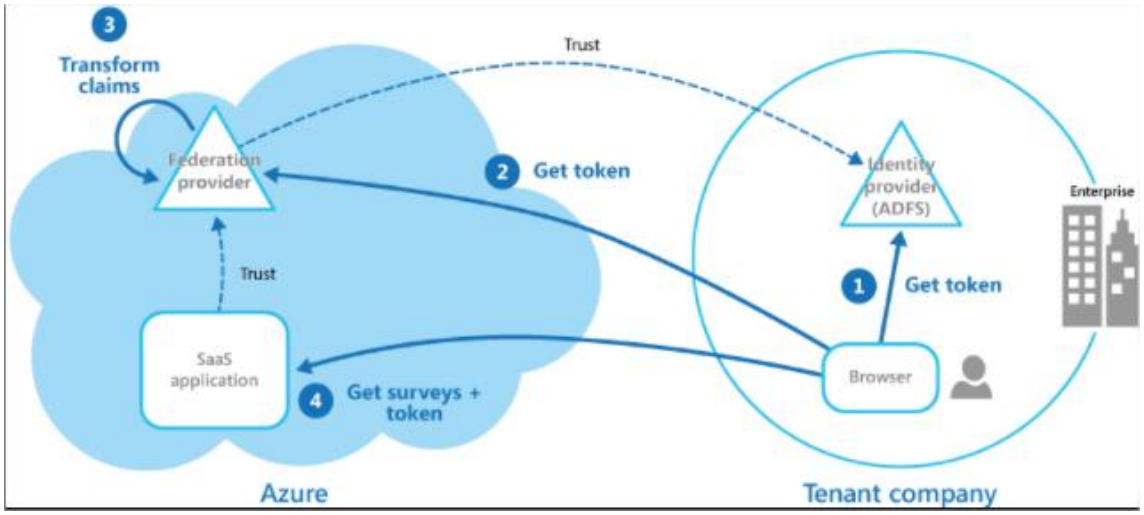
Es posible que este patrón no sea útil en las siguientes situaciones:

- Todos los usuarios de la aplicación pueden ser autenticados por un proveedor de identidad y no es necesario autenticarse con ningún otro proveedor de identidad. Esto es típico en aplicaciones comerciales que usan un directorio corporativo (accesible dentro de la aplicación) para la autenticación, mediante el uso de una VPN o (en un escenario alojado en la nube) a través de una conexión de red virtual entre el directorio local y la aplicación.

- La aplicación se creó originalmente utilizando un mecanismo de autenticación diferente, tal vez con tiendas de usuarios personalizadas, o no tiene la capacidad de manejar los estándares de negociación utilizados por las tecnologías basadas en reclamos. La adaptación de la autenticación basada en reclamaciones y el control de acceso en aplicaciones existentes puede ser compleja y probablemente no rentable.

Ejemplo

Una organización aloja una aplicación de software como servicio (SaaS) de varios inquilinos en Microsoft Azure. La aplicación incluye un sitio web que los inquilinos pueden usar para administrar la aplicación para sus propios usuarios. La aplicación permite a los inquilinos acceder al sitio web mediante el uso de una identidad federada generada por los Servicios de federación de Active Directory (AD FS) cuando un usuario es autenticado por el propio Active Directory de esa organización.



La figura muestra cómo los inquilinos se autentican con su propio proveedor de identidad (paso 1), en este caso AD FS. Después de autenticar correctamente a un inquilino, AD FS emite un token. El navegador del cliente reenvía este token al proveedor de federación de la aplicación SaaS, que confía en los tokens emitidos por el AD FS del inquilino, para recuperar un

token que es válido para el proveedor de federación de SaaS (paso 2). Si es necesario, el proveedor de federación de SaaS realiza una transformación de las notificaciones del token en notificaciones que la aplicación reconoce (paso 3) antes de devolver el nuevo token al navegador del cliente. La aplicación confía en los tokens emitidos por el proveedor de federación de SaaS y usa las notificaciones del token para aplicar las reglas de autorización (paso 4).

Los inquilinos no necesitarán recordar credenciales separadas para acceder a la aplicación, y un administrador de la empresa del inquilino puede configurar en su propio AD FS la lista de usuarios que pueden acceder a la aplicación.