Guía Detallada para Implementar Inicio de Sesión con GitLab

Introducción al Inicio de Sesión Social y OAuth 2.0

El inicio de sesión utilizando un proveedor social ofrece principalmente una experiencia de usuario con menos procesos de registro e inicio de sesión, que es un proceso normalmente engorroso. De esta manera los usuarios encuentran menos barreras para interactuar con una aplicación. Adicionalmente, al delegar la autenticación a un proveedor de identidad confiable como GitLab, Google o cualquier otro, se puede potencialmente mejorar la seguridad. El inicio de sesión social también permite, con el consentimiento del usuario, acceder a información básica de su perfil, lo que puede enriquecer la experiencia dentro de la aplicación.

OAuth 2.0 es un protocolo de autorización que posibilita que una aplicación acceda a recursos alojados en otra aplicación en nombre de un usuario, sin necesidad de que la primera conozca las credenciales de este último. En este modelo, existen roles definidos: el propietario del recurso (el usuario), la aplicación cliente (la aplicación a la cual queremos que el cliente se conecte), el servidor de autorización (GitLab en este contexto) y el servidor de recursos (también GitLab, que alberga la información del usuario). El flujo general implica que la aplicación cliente solicita autorización al usuario, quien la otorga o deniega. Si se otorga, el servidor de autorización emite un código de autorización a la aplicación cliente. Esta aplicación luego intercambia dicho código por un token de acceso. Finalmente, la aplicación cliente utiliza este token para solicitar los recursos protegidos al servidor de recursos.

Configuración de la Aplicación OAuth en GitLab

Para habilitar el inicio de sesión social utilizando GitLab, el primer paso es registrar tu aplicación como una "aplicación OAuth" dentro de tu cuenta de GitLab. Este proceso se inicia accediendo a la configuración de tu perfil en GitLab, generalmente a través del menú desplegable asociado a tu avatar en la esquina superior derecha, seleccionando la opción "Settings". Dentro del menú de la izquierda, se debe seleccionar la sección "Applications". A continuación, se hace clic en el botón "New application".

En el formulario de creación de la nueva aplicación, se deben completar los siguientes campos. En primer lugar, se debe asignar un nombre descriptivo a la aplicación, por ejemplo, "Inicio de sesión de Mi Aplicación". Seguidamente, se debe configurar la "Redirect URI", que es la URL a la cual GitLab redirigirá al usuario una vez que haya autorizado el acceso a tu aplicación. Es de vital importancia que esta URL sea precisa y coincida exactamente con la ruta de manejo de la respuesta de autenticación implementada en tu aplicación. Para aplicaciones web, esta suele ser una ruta en el backend. La casilla "Confidential" debe marcarse si tu aplicación posee un backend seguro capaz de almacenar el "Client Secret". Finalmente, se deben seleccionar los "Scopes" o permisos que tu aplicación necesita solicitar

a los usuarios de GitLab. Scopes comunes para el inicio de sesión social incluyen `read_user`, que permite acceder a la información básica del perfil del usuario, y `email`, que otorga acceso a su dirección de correo electrónico. Es fundamental solicitar únicamente los scopes que sean estrictamente necesarios para la funcionalidad de la aplicación. Una vez completado el formulario, se debe hacer clic en el botón "Submit".

Tras la creación exitosa de la aplicación, GitLab proporcionará un "Application ID", también conocido como "Client ID", y un "Secret", denominado "Client Secret". Es crucial almacenar estos valores de manera segura, especialmente el "Client Secret", ya que se utiliza para autenticar tu aplicación ante GitLab.

Implementación en Aplicaciones Web

El proceso de autenticación en aplicaciones web implica redirigir al usuario hacia GitLab para la autorización y posteriormente manejar la respuesta en el backend de la aplicación.

El flujo de autenticación OAuth 2.0 en aplicaciones web. El usuario interactúa con un botón "Iniciar sesión con GitLab" en la aplicación. Al hacer clic, el frontend o backend de la aplicación construye una URL de autorización y redirige al usuario a GitLab. GitLab presenta entonces una página de autorización, solicitando al usuario confirmar si desea otorgar acceso a la aplicación. Si el usuario aprueba, GitLab lo redirige de vuelta a la "Redirect URI" configurada, adjuntando un código de autorización (`code`) en la URL. El backend de la aplicación recibe esta solicitud que contiene el código. A continuación, el backend realiza una solicitud POST al endpoint de token de GitLab, incluyendo el código de autorización, el "Client ID" y el "Client Secret" para obtener un token de acceso. GitLab verifica estas credenciales y, si son válidas, devuelve un token de acceso y, opcionalmente, un token de actualización. Con el token de acceso, el backend puede solicitar la información del usuario a GitLab, dirigiéndose al endpoint de información del usuario. GitLab responde con la información solicitada, generalmente en formato JSON. Finalmente, basándose en esta información, el backend crea una nueva cuenta de usuario en la aplicación o inicia la sesión de un usuario existente, gestionando la sesión mediante mecanismos como cookies.

En la implementación del lado del cliente (Frontend), es necesario generar la URL de autorización que dirigirá al usuario al endpoint de autorización de GitLab. Esta URL debe incluir los siguientes parámetros: `client_id` (la "Application ID" de GitLab), `redirect_uri` (la URL de redireccionamiento configurada, codificada para su uso en URL), `response_type` (que debe ser `code` para el flujo de código de autorización), `scope` (los permisos solicitados, separados por espacios y codificados para URL), y `state` (un valor aleatorio opcional pero altamente recomendado, generado por tu aplicación y verificado al recibir la respuesta de GitLab para prevenir ataques CSRF). Cuando el usuario interactúa con el botón de inicio de sesión con GitLab, el frontend debe realizar una redirección del navegador a esta URL construida.

En la implementación del lado del servidor (Backend), se debe configurar una ruta para recibir la redirección de GitLab. Esta ruta recibirá el código de autorización (`code`) y el

parámetro `state` (si se envió) como parte de la URL. El siguiente paso es intercambiar este código de autorización por un token de acceso. Para ello, el backend debe enviar una solicitud POST al endpoint de token de GitLab (`https://gitlab.com/oauth/token`). Esta solicitud debe incluir en el cuerpo (generalmente con el formato `application/x-www-form-urlencoded`) los siguientes parámetros: `client_id` ("Application ID"), `client_secret` ("Client Secret"), `code` (el código de autorización recibido), `grant_type` (que debe ser `authorization_code`), y `redirect_uri` (la misma URL de redireccionamiento utilizada en el paso de autorización). Una vez obtenido el `access_token` en la respuesta de GitLab, se puede utilizar para solicitar la información del usuario al endpoint de la API de GitLab (`https://gitlab.com/api/v4/user`). Esto se realiza típicamente mediante una solicitud GET, incluyendo el `access_token` en el encabezado `Authorization` con el formato `Bearer <access_token>` . La respuesta de GitLab contendrá la información del usuario, como su ID, nombre de usuario, correo electrónico y nombre. Con esta información, el backend puede proceder a buscar un usuario existente en la base de datos de la aplicación o crear uno nuevo, asociando la cuenta de GitLab con la cuenta local. Finalmente, se debe iniciar la sesión del usuario en la aplicación. Durante todo este proceso, es crucial verificar el parámetro `state` recibido para prevenir ataques CSRF, manejar los posibles errores en cada etapa, registrar estos errores para facilitar la depuración e implementar medidas de seguridad adecuadas para proteger los tokens y la información del usuario.

Consideraciones de Seguridad

La seguridad es un aspecto primordial en la implementación del inicio de sesión social. Es esencial validar rigurosamente la URL de redireccionamiento configurada en GitLab para asegurar que las respuestas de autenticación se dirijan únicamente a ubicaciones de confianza. El "Client Secret" debe almacenarse de forma segura en el backend de la aplicación y nunca debe exponerse en el código del lado del cliente (frontend o aplicaciones móviles). Si aplica, se debe validar la firma de los tokens recibidos de GitLab para verificar su autenticidad e integridad. Para proteger contra ataques de falsificación de solicitudes entre sitios (CSRF), es crucial implementar y verificar el parámetro `state` en el flujo de autorización. Además, se debe tener un control estricto sobre los scopes solicitados, otorgando a la aplicación únicamente los permisos necesarios. Finalmente, es importante implementar mecanismos para revocar los tokens de acceso en caso de necesidad.

Referencias