

Versão: 2025-03

Autenticação

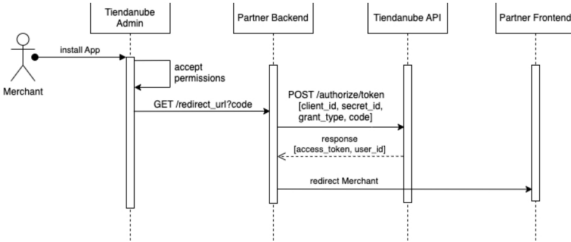
Fornecemos autorização e autenticação de usuários por meio de uma forma restrita do OAuth 2. Em resumo:

- O único tipo de concessão que aceitamos é o de "código de autorização".
- Nossos **tokens de acesso** não expiram. Eles se tornam inválidos somente depois que você obtém um novo ou se o usuário desinstala seu aplicativo.
- Juntamente com o token de acesso, fornecemos um ID `user_id`, que é o ID da loja. Este ID é fundamental: você precisa dele para fazer solicitações à nossa API. Ele também é útil para autenticar usuários do aplicativo em seu site (consulte Autenticação de usuário).

Ao criar um aplicativo, você será solicitado a preencher algumas informações sobre os URLs do seu aplicativo ([consulte URLs](#)). Você também precisa selecionar os escopos necessários para o seu aplicativo e definir um URL de redirecionamento, que é o URL para onde redirecionaremos o cliente após a instalação do aplicativo. Os escopos especificam os recursos para os quais o aplicativo solicita autorização dos usuários ([consulte Escopos](#)). O URL de redirecionamento é usado como parte do fluxo de autorização (veja abaixo). Assista a [este vídeo](#) onde mostramos como o processo de autorização funciona aqui no Tiendanube.

Fluxo de autorização

O fluxo de autorização é bastante padrão, exceto pela primeira etapa:



- O usuário, a partir do painel de administração do Tiendanube/Nuvemshop, clica em um botão para instalar seu aplicativo. Ou, alternativamente, ele acessa https://www.tiendanube.com/apps/app_id/authorize (se não estiver logado, será solicitado que faça login).
- Ele é redirecionado para uma página onde precisa autorizar os escopos necessários para o seu aplicativo (se ele já tiver feito isso, esta etapa é ignorada).
- Ele é redirecionado para o URL de redirecionamento do seu aplicativo com um código de autorização, que expira em 5 minutos.
- Usando as credenciais do seu aplicativo e o código de autorização, você pode obter um token de acesso fazendo uma solicitação POST para <https://www.tiendanube.com/apps/authorize/token>. (Não se esqueça de enviar também o token de acesso `grant_type=authorization_code`, veja o exemplo abaixo).

Nota: Qualquer um desses URLs no domínio [tiendanube.com](https://www.tiendanube.com) pode ser substituído por URLs idênticos no domínio [nuvemshop.com.br](https://www.nuvemshop.com.br).

Suponha que seu aplicativo tenha:

- `id = 123`
- URL de redirecionamento = <https://www.example.com/>
- `escopos = read_orders,write_products`
- `segredo_do_cliente = abcdef`

- A loja com ID 789 acessa <https://www.tiendanube.com/apps/123/authorize?state=csrf-code>
- O usuário aceita.
- Ele é redirecionado para <https://www.example.com/?code=xyz&state=csrf-code>.
- Então você faz:

```
curl -d '{"client_id": "123", "client_secret": "xxxxxxxx", "grant_type": "authorization_code", "code": "H", "Content-Type": "application/json"}' \
-X POST "https://www.tiendanube.com/apps/authorize/token"
```

e receber:

```
{
  "access_token": "61181d08b7e328d256736hdc671c3ce50b8af5",
  "token_type": "bearer",
  "scope": "read_orders,write_products",
  "user_id": "789"
}
```

Em algumas solicitações, você receberá a resposta com um `store_id` campo contendo os dados de `user_id`

Os URLs do seu aplicativo são importantes para proporcionar a melhor experiência possível ao lojista/consumidor ao usar nossa plataforma. Todos eles são obrigatórios.

URL, where we will redirect to the client after installing the application
https://google.com/tiendanube/callback
URL
https://google.com
Preferences URL
https://google.com/preferences
Privacy Policy URL
https://google.com/privacy-policy
URL, webhook store redact
https://google.com/webhook-store-redact
customers redact webhook url
https://google.com/webhook-customers-redact
URL, customers data request webhook
https://google.com/webhook-data-request
Support URL
https://google.com/support
Support Email
support@google.com

- URL para onde o usuário será redirecionado para o comerciante após a instalação do aplicativo:** URL para a qual o usuário será redirecionado após instalar o aplicativo. Esta é a sua URL de retorno de chamada. Você deve usá-la para obter o código de autorização e gerar o token de acesso, conforme explicado acima (consulte Fluxo de Autorização).
- URL:** Endereço do painel de administração que o usuário acessará para utilizar o aplicativo. Por exemplo, a primeira página que o usuário vê após fazer login em seu aplicativo.

- **URL de preferências:** URL da página do seu aplicativo onde o usuário pode configurar as opções do aplicativo. Alguns exemplos:
 - Pagamentos: configure o parcelamento para compras com cartão de crédito.
 - Envio: configure quais métodos de envio estarão disponíveis para os consumidores.
 - Marketing: modificar os scripts inseridos na página.
- **URL da Política de Privacidade:** este URL é exibido na página de detalhes do aplicativo na App Store. Funciona como uma ferramenta de transparência entre a empresa e o usuário.
- **Webhook de URL para Redação de Dados da Loja:** solicitações de exclusão de dados da loja. Este webhook será acionado somente se o lojista solicitar a exclusão dos dados da loja por meio do painel de administração.
- **Webhook de URL para Redação de Dados do Cliente:** solicitação de exclusão de dados do consumidor. Este webhook será acionado somente se o lojista solicitar a exclusão dos dados do cliente por meio do painel de administração.
- **Webhook de solicitação de dados do cliente por URL:** solicitação para receber o relatório de dados armazenados de um usuário.
- **URL de suporte:** URL para onde o usuário será redirecionado quando precisar de ajuda. Pode ser uma página de autoatendimento ou um formulário de contato. Se a URL de suporte estiver vazia, o lojista será redirecionado para o e-mail de suporte.
- **E-mail de suporte:** endereço de e-mail para contato com o suporte, onde os usuários do aplicativo podem entrar em contato.

Os aplicativos devem solicitar apenas os escopos de acesso de que precisam. Se um aplicativo precisar apenas ler produtos, ele deverá solicitar apenas o `read_products` escopo necessário.

Se você solicitar qualquer escopo de escrita, o escopo de leitura estará implícito.

Como os **webhooks** dependem de outros recursos, você só poderá registrar webhooks para os recursos que lhe foram concedidos para uso.

Os escopos disponíveis para a API são:

- `ler_conteúdo / escrever_conteúdo`
 - Página
- `ler_produtos / escrever_produtos`
 - Produto
 - Variante do produto
 - Imagem do produto
 - Categoria
- `ler_clientes / escrever_clientes`
 - Cliente
- `ler_pedidos / escrever_pedidos`
 - Ordem
- `ler_cupons / escrever_cupons`
 - Cupom
- `escrever_scripts`
 - Roteiro
- `escrever_envio`
 - Transportadora

do usuário

Se você precisa que os usuários do seu aplicativo possam fazer login no seu site, você pode reutilizar o fluxo de autorização para essa finalidade:

- O botão de login deve ser um link para [https://www.tiendanube.com/apps/\(app_id\)/authorize](https://www.tiendanube.com/apps/(app_id)/authorize).
- Você pode fazer login com o usuário que tiver o ID recebido com o novo token de acesso.
- No link, **você deve** adicionar um `state` parâmetro para proteção contra CSRF (por exemplo, um hash do cookie de sessão) e verificá-lo após o redirecionamento.

comuns

Abaixo estão alguns dos problemas mais comuns relacionados à autenticação.

Enviar dados como parâmetros em vez de como corpo do arquivo.

Ao realizar a requisição POST para obter o `access_token`, você deve enviar as informações no corpo da requisição, e não como parâmetros na URL.

Errado:

```
curl -X POST "https://www.tiendanube.com/apps/authorize/token?client_id=0000&client_secret=xxxxxxx&grant_type=authorization_code"
```

Correto:

```
curl -d '{"client_id": "0000", "client_secret": "xxxxxxx", "grant_type": "authorization_code", "code": "XXXXXX"}' -X POST "https://www.tiendanube.com/apps/authorize/token"
```

No Postman, certifique-se de que as informações estejam incluídas na aba **Corpo**, e não na aba **Parâmetros**, e que o tipo de dados seja **JSON**, e não **Texto/Javascript/HTML/XML**.

Cabeçalho de autenticação versus autorização

Lembre-se de que o cabeçalho deve ser de Autenticação, e não de Autorização. Usar o último resultará em um erro 401: não autorizado - token de acesso inválido.

Portador (malúsculo) vs portador (minúsculo)

A palavra **"bearer"** deve estar inteiramente em minúsculas. Usar uma palavra diferente resultará em um erro 401: não autorizado - token de acesso inválido.

Comando curl com código nulo

Quando a instalação do aplicativo é aceita e o usuário é redirecionado para a página onde retornamos o comando curl, o `code` valor pode aparecer como **nulo**. Isso ocorre porque essa página de destino gera um comando curl completo apenas durante a instalação inicial.

Para instalações subsequentes, o parâmetro `code` deve ser capturado diretamente `code` na URL de redirecionamento.

A instalação na loja solicita que você faça login na conta do parceiro.

Se, após aceitar a instalação do aplicativo, o fluxo levar a uma tela de login de parceiro, é provável que seja porque o URL de redirecionamento padrão ainda está sendo usado:

- [https://partners.tiendanube.com/applications/authentication/\(app_id\)](https://partners.tiendanube.com/applications/authentication/(app_id))

Este URL padrão aponta para uma página de destino acessível apenas pelo painel de administração do parceiro. Consequentemente, ao ser redirecionado para lá, o sistema solicita as credenciais do parceiro. Para corrigir isso, o parceiro deve atualizar o URL de redirecionamento para um URL personalizado. A partir desse momento, o redirecionamento funcionará corretamente e incluirá o `code` parâmetro conforme o esperado.

Última atualização em 22 de janeiro de 2026

Anterior
« [Controle de versões](#)

Próximo
[Recursos da API](#) »

