

Introduction

Authentication & Authorization



The Google Ads API Developer Series



Laura Chevalier

Developer Relations Engineer

Authentication & Authorization

Authentication - WHO you are

Authorization - WHAT you can do

What is OAuth2 and Why Do We Need It?

- Industry-standard authorization protocol
- All Google Ads API calls must be authorized through [OAuth2](#)
- OAuth2 enables your Google Ads API client app to access a user's Google Ads account without having to handle or store the user's login info



Required Credentials: Developer Token



Required Credentials: Developer Token

- Account agnostic credential to connect your app to the Google Ads API

The image shows a screenshot of the Google API Access and Developer Details pages. The 'API Access' section includes a table with the following data:

| Field | Value |
|-----------------|-------------------|
| Developer token | <DEVELOPER TOKEN> |
| Access level | Standard Access |

The 'Developer Details' section includes a text prompt: "Please tell us about yourself and how you plan on using the API. We encourage you to keep this information up to date for important service announcements." Below this is another table with the following data:

| Field | Value |
|-----------------------------|----------------------------|
| API contact email | <EMAIL ADDRESS> |
| Company name | Google |
| Company URL | http://www.google.com |
| Company type | Advertiser |
| Intended use | Internal tests DevRel Team |
| Principal place of business | United States |

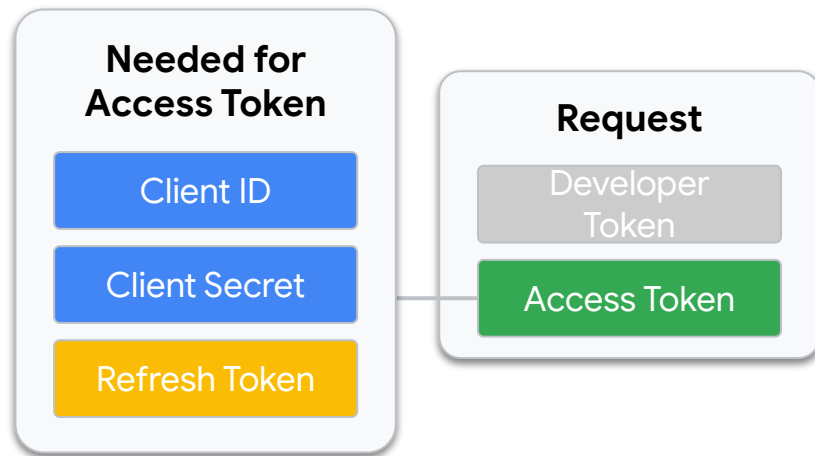
A 'TOOLS & SETTINGS' menu is overlaid on the right side of the screenshot, listing the following options:

- SETUP
- Account map
- Access and security
- Linked accounts
- Preferences
- API Center
- Google Merchant Center

To the right of the screenshot is a 'Request' box containing two buttons: a blue 'Developer Token' button and a grey 'Access Token' button. A line connects the 'Developer Token' button to the 'Developer token' field in the 'API Access' table.

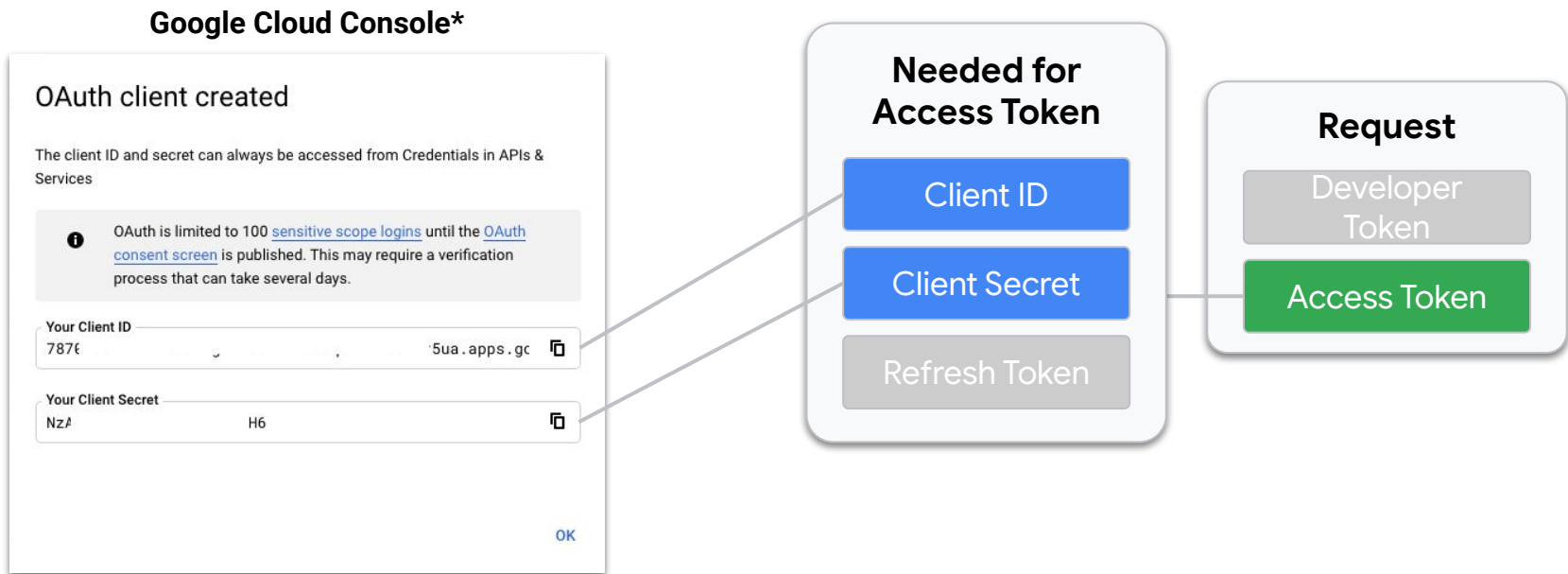
Required Credentials: Access Token

- Credential to grant temporary access to a specific account (typically last 1 hour)



Required Credentials: Client ID & Client Secret

- Credentials for a specific Google Cloud account

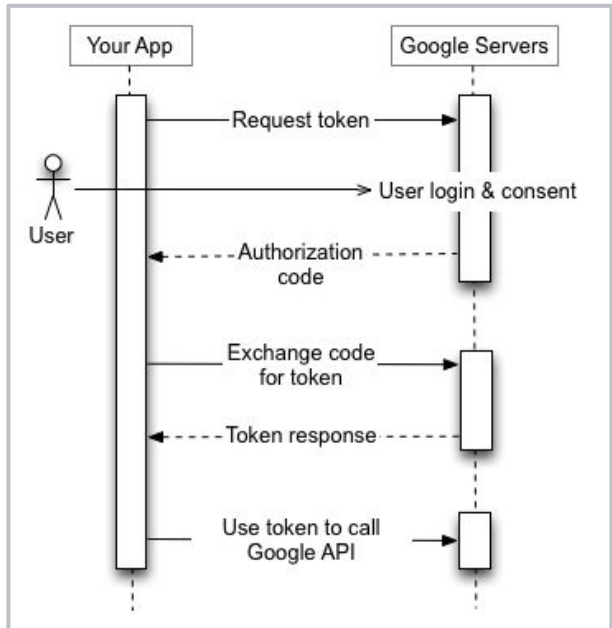


* The Google Ads API must be [enabled](#) on the Google Cloud Project

Required Credentials: Refresh Token

- Credential to generate new Access Tokens

Example Refresh Token Flow



Needed for Access Token

Client ID

Client Secret

Refresh Token

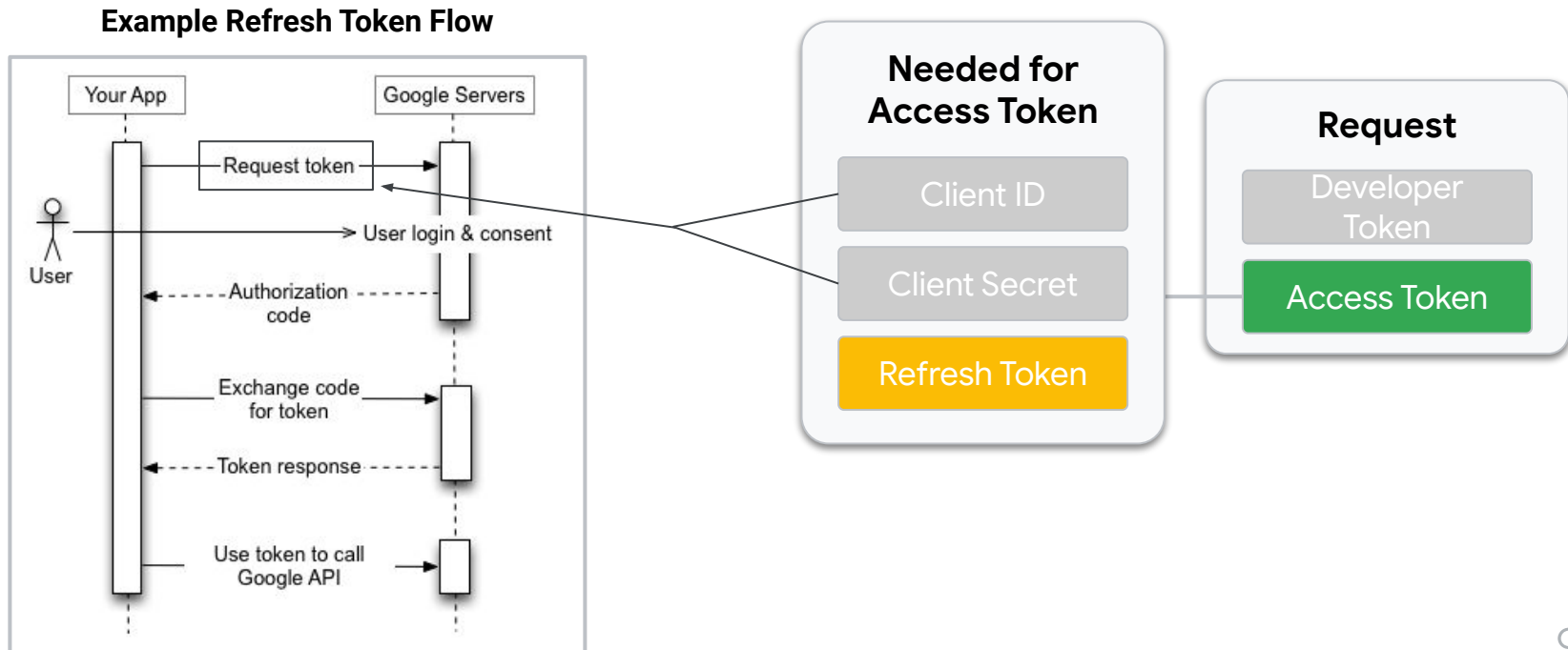
Request

Developer Token

Access Token

Required Credentials: Refresh Token

- Client ID and Client Secret also used to generate a refresh token



Steps to Make an Authenticated Request

1. Create a Developer Token

The image shows a screenshot of the Google API Access and Developer Details pages. The 'API Access' section displays the 'Developer token' as '<DEVELOPER TOKEN>' and the 'Access level' as 'Standard Access'. The 'Developer Details' section includes fields for 'API contact email', 'Company name', 'Company URL', 'Company type', 'Intended use', and 'Principal place of business'. A 'TOOLS & SETTINGS' menu is open, showing options like 'SETUP', 'Account map', 'Access and security', 'Linked accounts', 'Preferences', 'API Center', and 'Google Merchant Center'. A 'Request' box on the right highlights the 'Developer Token' and 'Access Token'.

| API Access | |
|-----------------|-------------------|
| Developer token | <DEVELOPER TOKEN> |
| Access level | Standard Access |

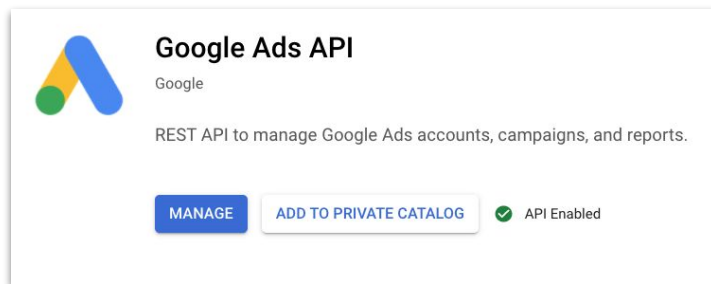
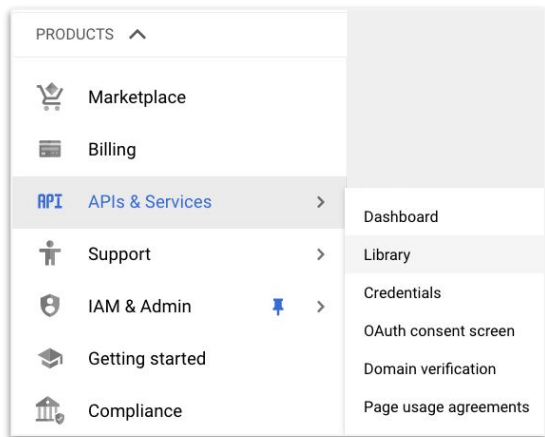
| Developer Details | |
|--|----------------------------|
| Please tell us about yourself and how you plan on using the API. We encourage you to keep this information up to date for important service announcements. | |
| API contact email | <EMAIL ADDRESS> |
| Company name | Google |
| Company URL | http://www.google.com |
| Company type | Advertiser |
| Intended use | Internal tests DevRel Team |
| Principal place of business | United States |

- TOOLS & SETTINGS
 - SETUP
 - Account map
 - Access and security
 - Linked accounts
 - Preferences
 - API Center
 - Google Merchant Center

| Request | |
|-----------------|--|
| Developer Token | |
| Access Token | |

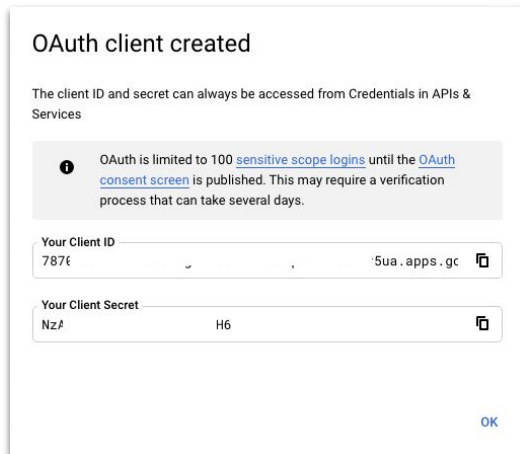
Steps to Make an Authenticated Request

1. Create a Developer Token
2. Obtain Client ID and Client Secret from configured Google Cloud Project
 - a. Create a [Google Cloud Project](#)
 - b. Enable the Google Ads API in your project in the [API Library](#)**



Steps to Make an Authenticated Request

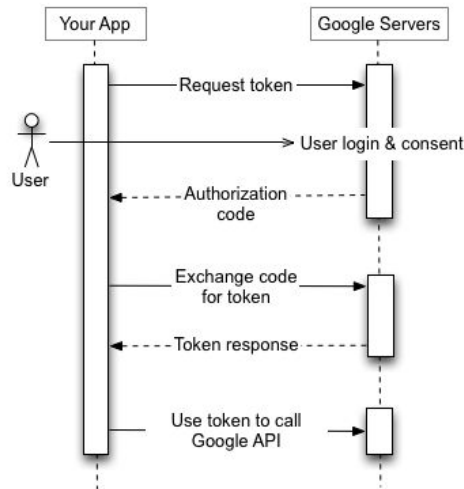
1. Create a Developer Token
2. Obtain Client ID and Client Secret from configured Google Cloud Project
 - a. Create a [Google Cloud Project](#)
 - b. Enable the Google Ads API in your project in the [API Library](#)
 - c. Create a Client ID and Client Secret**



Steps to Make an Authenticated Request

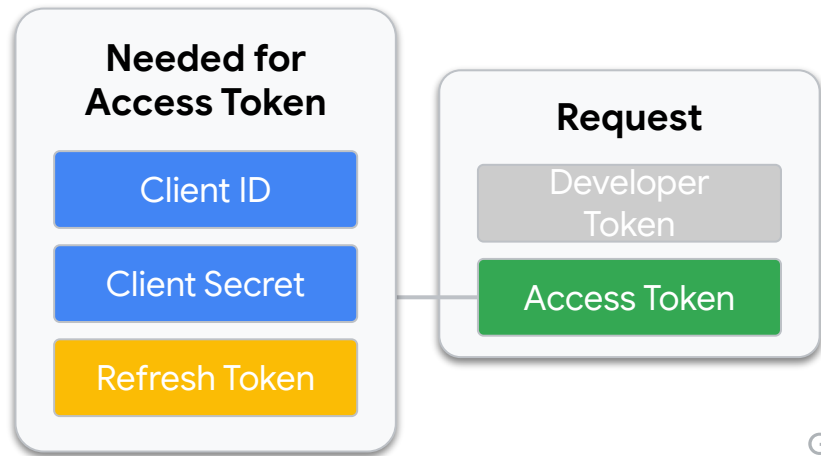
1. Create a Developer Token
2. Obtain Client ID and Client Secret from configured Google Cloud Project
 - a. Create a [Google Cloud Project](#)
 - b. Enable the Google Ads API in your project in the [API Library](#)
 - c. Create a Client ID and Client Secret

3. Generate a Refresh Token



Steps to Make an Authenticated Request

1. Create a Developer Token
2. Obtain Client ID and Client Secret from configured Google Cloud Project
 - a. Create a [Google Cloud Project](#)
 - b. Enable the Google Ads API in your project in the [API Library](#)
 - c. Create a Client ID and Client Secret
3. Generate a Refresh Token
- 4. Generate an Access Token**



Steps to Make an Authenticated Request

1. Create a Developer Token
2. Obtain Client ID and Client Secret from configured Google Cloud Project
 - a. Create a [Google Cloud Project](#)
 - b. Enable the Google Ads API in your project in the [API Library](#)
 - c. Create a Client ID and Client Secret
3. Generate a Refresh Token

4. Generate an Access Token

Note: client libraries handle this for you provided a valid refresh token

Steps to Make an Authenticated Request

1. Create a Developer Token
2. Obtain Client ID and Client Secret from configured Google Cloud Project
 - a. Create a [Google Cloud Project](#)
 - b. Enable the Google Ads API in your project in the [API Library](#)
 - c. Create a Client ID and Client Secret
3. Generate a Refresh Token

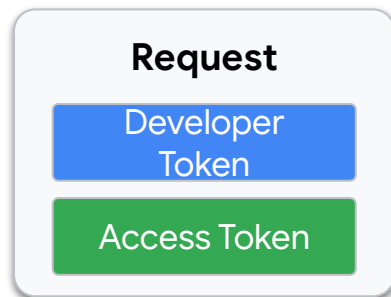
4. Generate an Access Token

cURL Request to Generate an Access Token

```
curl \  
  --data "grant_type=refresh_token" \  
  --data "client_id=${CLIENT_ID}" \  
  --data "client_secret=${CLIENT_SECRET}" \  
  --data "refresh_token=${REFRESH_TOKEN}" \  
  https://www.googleapis.com/oauth2/v3/token
```

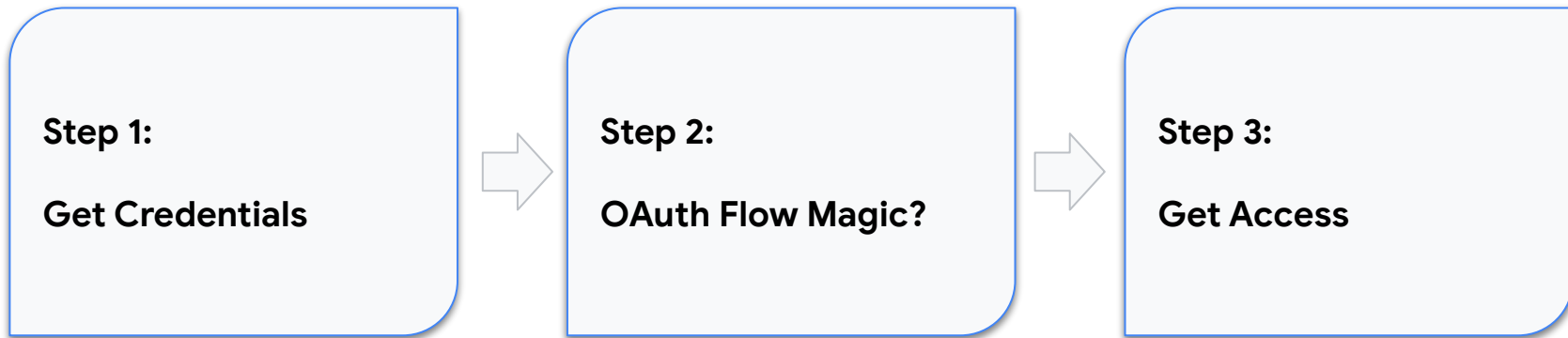

Steps to Make an Authenticated Request

1. Create a Developer Token
2. Obtain Client ID and Client Secret from configured Google Cloud Project
 - a. Create a [Google Cloud Project](#)
 - b. Enable the Google Ads API in your project in the [API Library](#)
 - c. Create a Client ID and Client Secret
3. Generate a Refresh Token
4. Generate an Access Token
- 5. Make an authenticated request**



Account Access

Wait... How is Account Access Actually Determined?

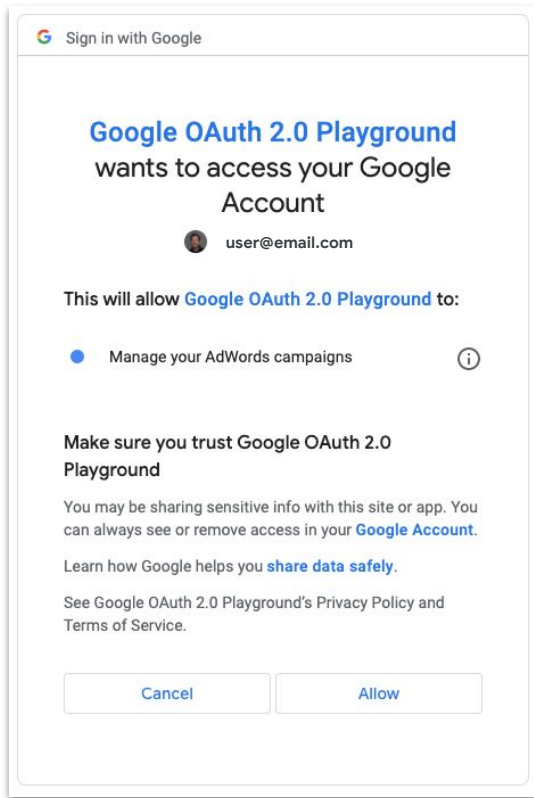
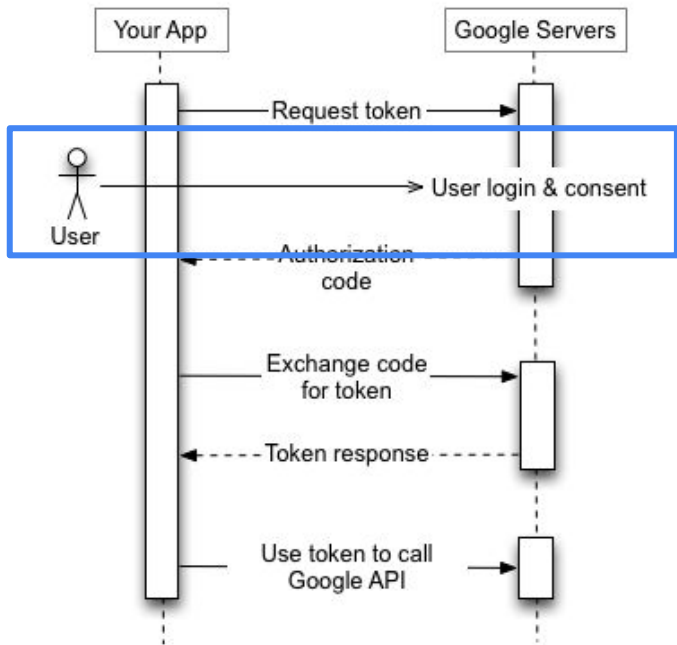


Wait... How is Account Access Actually Determined?

Step 2:

OAuth Flow Magic?

User Login & Consent




User Login & Consent

The permissions that an access token has are directly tied to the permissions that email has on the Google Ads account.

Sign in with Google

Google OAuth 2.0 Playground wants to access your Google Account

 user@email.com

This will allow **Google OAuth 2.0 Playground** to:

- Manage your AdWords campaigns

Make sure you trust **Google OAuth 2.0 Playground**

You may be sharing sensitive info with this site or app. You can always see or remove access in your [Google Account](#).

Learn how Google helps you [share data safely](#).

See Google OAuth 2.0 Playground's [Privacy Policy](#) and [Terms of Service](#).

Google Ads | Access and security

USERS MANAGERS SECURITY

This table shows the users that have access to this account. To manage users who manage billing, visit the "Payments contacts" section of "Billing & payments" settings.

| User | Last signed-in | Access level | Authentication method | Inviter user |
|-----------------|----------------|--------------|-----------------------|----------------|
| user@email.com | Sep 13, 2021 | Admin | 2-Step Verification | |
| user2@email.com | Jan 6, 2020 | Admin | 2-Step Verification | user@email.com |


TOOLS & SETTINGS

- SETUP
- Business data
- Policy manager
- Access and security
- Linked accounts
- Preferences
- Google Merchant Center

User Login & Consent

Sign in with Google

Google OAuth 2.0 Playground wants to access your Google Account

 user@email.com

This will allow **Google OAuth 2.0 Playground** to:

- Manage your AdWords campaigns i

Make sure you trust Google OAuth 2.0 Playground

You may be sharing sensitive info with this site or app. You can always see or remove access in your [Google Account](#).

Learn how Google helps you [share data safely](#).

See Google OAuth 2.0 Playground's [Privacy Policy](#) and [Terms of Service](#).

Google Ads | Access and security

SEARCH | REPORTS | TOOLS & SETTINGS

USERS | MANAGERS | SECURITY

This table shows the users that have access to this account. To manage users who manage billing, visit the "Payments contacts" section of "Billing & payments" settings.

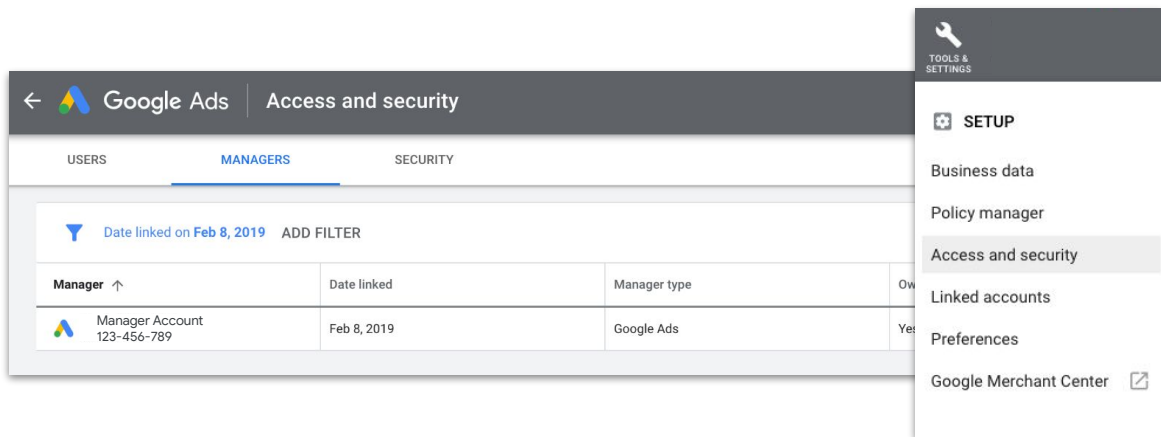
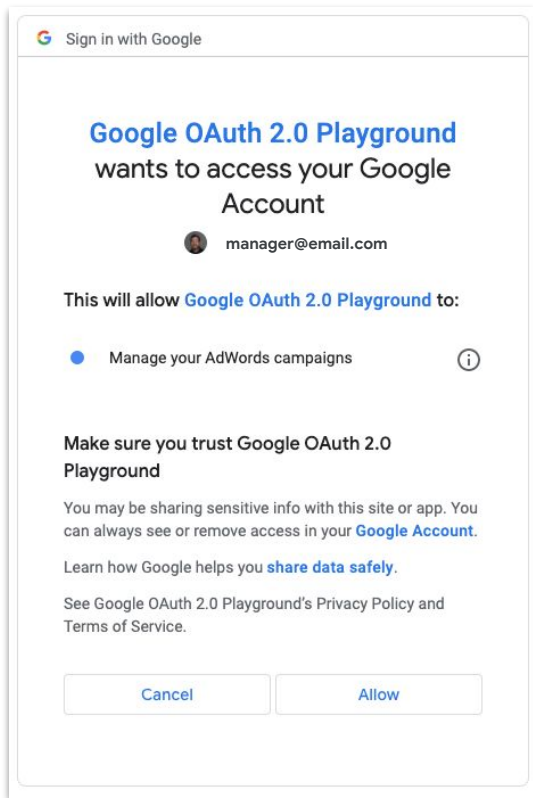
+ ▼ ADD FILTER

| User ↑ | Last signed-in | Access level | Authentication method | Inviter user |
|-----------------|----------------|--|-----------------------|----------------|
| user@email.com | Sep 13, 2021 | Admin | 2-Step Verification | |
| user2@email.com | Jan 6, 2020 | Admin | 2-Step Verification | user@email.com |

⚙️ **SETUP**

- Business data
- Policy manager
- Access and security**
- Linked accounts
- Preferences
- Google Merchant Center ↗️

User Login & Consent



- Users on Manager accounts can also authenticate to make requests on behalf of client accounts using the [login-customer-id](#) request header

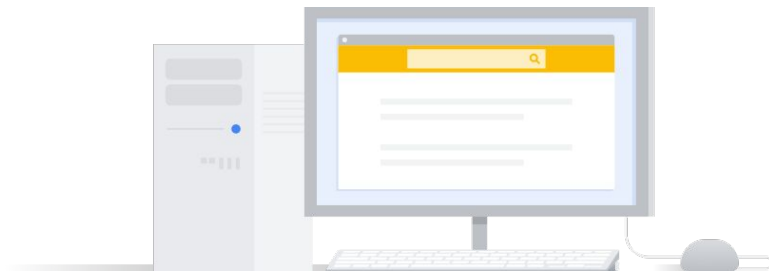
OAuth2 Flows

OAuth2 Flows

- Three types of OAuth2 “flows”
 - Desktop app flow (preferred)
 - Web app flow
 - Service account flow
- Client libraries provide utilities to make this easier for desktop and web app flows
 - [Desktop application flow](#)
 - [Web application flow](#)

Desktop App Flow (Recommended)

- Choose this app type if...
 - You're managing all of your Google Ads accounts using a single top level manager account
 - You're a first-time user, or want to get started quickly with the simplest setup
 - Your app will only authenticate Google users from one machine



Web App Flow

- Choose this app type if...
 - You want to authenticate as a user who can grant permission to your app to access their Google Ads account data
 - You want to easily generate multiple authorization credentials, to manage third-party accounts for example
 - Your app requires callback URLs, which are not supported in the desktop app flow

Quickstart Guide

The screenshot shows the Google Ads API Quickstart Guide page. The page has a green header with the Google Ads API logo and a search bar. Below the header is a navigation menu with links for Guides, Migration, Reference, Reports, Libraries & Examples, REST Interface, and Support & Policies. The left sidebar contains a list of product overview items, with 'Quickstart' selected. The main content area features a purple banner for migration workshops, a breadcrumb trail (Home > Products > Google Ads API), and a 'Was this helpful?' feedback widget. The main heading is 'Quickstart' with a square icon. The text explains that the guide uses a client library to retrieve Google Ads campaigns and provides instructions on enabling API access. It also includes sections for prerequisites, config parameters, and developer tokens.

Google Ads API

Guides Migration Reference Reports Libraries & Examples REST Interface Support & Policies

Product overview

- Quickstart
- Developer token
- Client library
- OAuth cloud project
- OAuth client library
- Refresh token
- First call
- Basic concepts
- Account management
- Ad management
- Assets
- Authentication
- Billing
- Campaign management
- Conversion management
- Extensions and feeds
- Planning service
- Reach forecasting
- Remarketing & audience targeting
- Reporting
- Targeting

Register for one of our Google Ads API Migration Workshops (October 5, 2021 - October 14, 2021)

Home > Products > Google Ads API

Was this helpful?

Quickstart

This guide uses a [client library](#) to retrieve Google Ads campaigns.

To enable API access, first obtain the required OAuth2 credentials. Then, add the authorization and account details to a config file. Once the config file is set up, you're ready to call the API.

Prerequisites

Config parameters

You'll need to supply the following information in the config file of your client library.

Developer token

Your developer token lets your app access the API, but the token is not restricted to only one Google Ads account. Once your token is approved, you can use the same token for requests against all your Google Ads accounts, even

Send feedback

Authentication Guide

The screenshot shows the Google Ads API Authentication Guide page. The page has a green header with the Google Ads API logo and a search bar. Below the header is a navigation menu with links for Guides, Migration, Reference, Reports, Libraries & Examples, REST Interface, and Support & Policies. The left sidebar contains a list of product overview items, with 'Authentication' selected. The main content area features a purple banner for migration workshops, a breadcrumb trail (Home > Products > Google Ads API), and a 'Was this helpful?' feedback widget. The main heading is 'OAuth2 in the Google Ads API' with a square icon. The text explains that all Google Ads API calls must be authorized through OAuth2 and provides a list of steps to complete authorization.

Google Ads API

Guides Migration Reference Reports Libraries & Examples REST Interface Support & Policies

Product overview

- Quickstart
- Basic concepts
- Account management
- Ad management
- Assets
- Authentication
- Cloud project
- Client library
- Service accounts
- Internals
- 2-step verification
- Playground
- Billing
- Campaign management
- Conversion management
- Extensions and feeds
- Planning service
- Reach forecasting
- Remarketing & audience targeting
- Reporting

Register for one of our Google Ads API Migration Workshops (October 5, 2021 - October 14, 2021)

Home > Products > Google Ads API

Was this helpful?

OAuth2 in the Google Ads API

All Google Ads API calls must be authorized through [OAuth2](#). OAuth2 enables your Google Ads API client app to access a user's Google Ads account without having to handle or store the user's login info.

The following steps must be completed to authorize requests to the Google Ads API.

1. [Configure a Cloud project for the Google Ads API](#)
2. [Configure a client library for OAuth in the Google Ads API](#)

Supporting materials are also available for related topics.

- [Using Service Accounts.](#)
- [Getting familiar with OAuth2 Internals for the Google Ads API.](#)
- [Using the OAuth Playground to generate refresh and access tokens.](#)

Send feedback

Thanks for Listening

- We welcome your feedback at googleadsapi-support@google.com
- Check out the full Authentication & Authorization [playlist](#) for related content
- See more episodes on our [YouTube channel](#)

