Getting Started

Before you can begin developing your Drive application, you need to download the Android and Google play Services SDKs and get an Android certificate, as described in the following sections.

Overview

Creating a new Android application that uses the Google Drive Android API requires several steps. Many of these steps only have to be perfomed once. The overall process of making a Drive-enabled Android application is as follows:

1. Install the Android SDK (//developer.android.com/sdk/index.html).
2. Download and configure the Google Play services SDK (//developer.android.com/google/play-services/setup.html), which includes the Google Drive Android API.
3. Register your application. To do this, you need to register a project in the Google API Console and get a signing certificate for your app.
4. Add the required settings (https://developers.google.com/drive/android/java-client#set_mime_types_in_the_app_manifest) to your application's manifest.
5. Add Drive services to your application. The easiest way to begin is by creating a file (#create_a_file_in_google_drive).

You may wish to begin by looking at some sample code, some of which is also included with the Google Play services SDK.

Install the Android SDK

As a prerequisite, you need to install the Android SDK. See Get the Android SDK (//developer.android.com/sdk/index.html).

Install and configure the Google Play services SDK

You need an Android project for your app before you can complete the steps in this section. If you haven't yet created an Android application, you can follow the tutorial Building Your First App (//developer.android.com/training/basics/firstapp/index.html).

The Google Drive Android API is distributed as part of the Google Play services SDK. You can download the Google Play services SDK via the Android SDK Manager. For detailed instructions, see the Google Play services SDK documentation (http://developer.android.com/google/play-services/setup.html).

Get an Android certificate and register your application

If you haven't already registered your application with the Google API Console, then set up a project and application in the API Console (https://console.developers.google.com/start/api?id=drive&credential=client_key). The system guides you through the process of choosing or creating a project and registering a new application, and it automatically activates the API for you.

If you've already registered your application with the API Console, then follow this procedure instead:

In a terminal, run the Keytool utility (https://developer.android.com/studio/publish/app-signing.html#signing-manually) to get the SHA1 fingerprint for your digitally signed `.apk` file's public certificate.

```
keytool -exportcert -alias androiddebugkey -keystore path-to-debug-or-production
```

Note: For the `debug.keystore`, the password is `android`. For Eclipse, the debug keystore is typically located at `~/.android/debug.keystore`.

The Keytool prints the fingerprint to the shell. For example:

```
$ keytool -exportcert -alias androiddebugkey -keystore ~/.android/debug.keystore
Enter keystore password: Type "android" if using debug.keystore
Alias name: androiddebugkey
Creation date: Aug 27, 2012
```

```
Entry type: PrivateKeyEntry
Certificate chain length: 1
Certificate[1]:
Owner: CN=Android Debug, O=Android, C=US
Issuer: CN=Android Debug, O=Android, C=US
Serial number: 503bd581
Valid from: Mon Aug 27 13:16:01 PDT 2012 until: Wed Aug 20 13:16:01 PDT 2042
Certificate fingerprints:
   MD5:  1B:2B:2D:37:E1:CE:06:8B:A0:F0:73:05:3C:A3:63:DD
   SHA1: D8:AA:43:97:59:EE:C5:95:26:6A:07:EE:1C:37:8E:F4:F0:C8:05:C8
   SHA256: F3:6F:98:51:9A:DF:C3:15:4E:48:4B:0F:91:E3:3C:6A:A0:97:DC:0A:3F:B2:D2:
   Signature algorithm name: SHA1withRSA
   Version: 3
```

Copy the SHA1 fingerprint, which is highlighted in the example above.

Important: When you prepare to release your app to your users, follow these steps again and create a new OAuth 2.0 client ID for your production app. For production apps, use your own private key to sign the production app's `.apk` file. For more information, see Signing your applications (https://developer.android.com/studio/publish/app-signing.html).

Next, create credentials appropriate to your project in the Google API Console:

1. Open the Credentials page (https://console.developers.google.com/apis/credentials) in the API Console.
2. Follow these steps if your application needs to submit authorized requests:
    a. Click **Create credentials > OAuth client ID**.
    b. Select **Android**.
    c. In the **Package name** field, enter your Android app's package name (//developer.android.com/guide/topics/manifest/manifest-element.html#package).
    d. Paste the SHA1 fingerprint into the form where requested.
    e. Click **Create**.

    Otherwise, follow the steps below, which are for applications that only need to make unauthorized API calls:

    a. Click **Create credentials > API key**.
    b. Select **Android key**.

c. Paste the SHA1 fingerprint into the form where requested.

d. Type your Android app's package name (//developer.android.com/guide/topics/manifest/manifest-element.html#package) into the form where requested.

e. Click **Create**.

Create a file in Google Drive

Now you're ready to authorize your application (https://developers.google.com/drive/android/auth#connecting_and_authorizing_the_google_drive_android_api), create a file (https://developers.google.com/drive/android/create-file), and test your app.

Except as otherwise noted, the content of this page is licensed under the Creative Commons Attribution 3.0 License (http://creativecommons.org/licenses/by/3.0/), and code samples are licensed under the Apache 2.0 License (http://www.apache.org/licenses/LICENSE-2.0). For details, see our Site Policies (https://developers.google.com/terms/site-policies). Java is a registered trademark of Oracle and/or its affiliates.

Last updated November 6, 2017.

Blog
Stay up-to-date on developing for G Suite

Google+ Community
Add us to your circles and connect with developers

GitHub
Explore our sample apps or fork them to build your own

Stack Overflow
Ask questions with the google-drive-sdk tag

Client Libraries
Download a client library to help you get started

Videos
Check out Drive API videos on YouTube