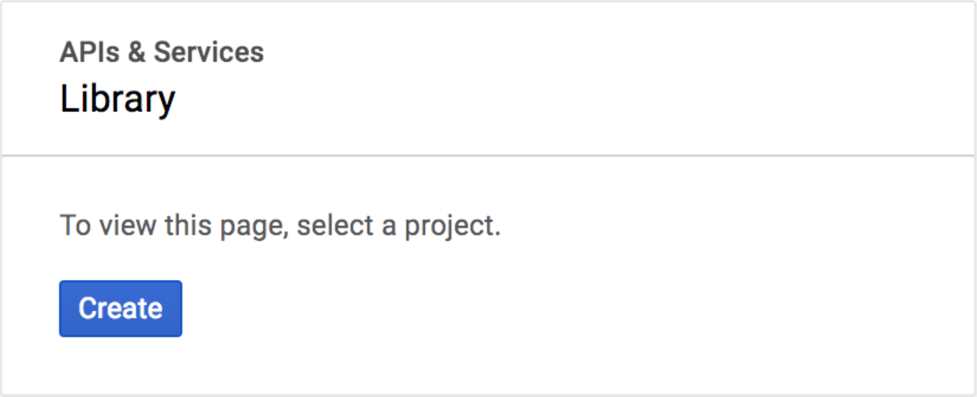
Setting up a Google Service Account with Binder

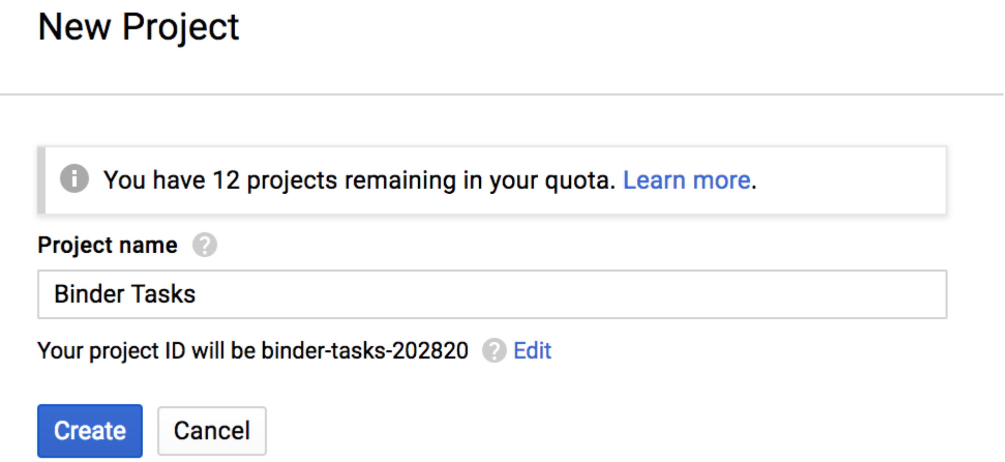
If you don’t already have a Google Account that you want to use with the service account, you can create one here: https://accounts.google.com/signup/v2/webcreateaccount?flowName=GlifWebSignIn&flowEntry=SignUp

1. **Enable Google Calendar API** – while signed in to the Google Account that you want to connect the calendar to, visit https://console.cloud.google.com/projectselector/apis/library/calendar-json.googleapis.com/ and click the blue “Create” button.



To view this page, select a project.

Name your project and click “Create” to finish creating your project.



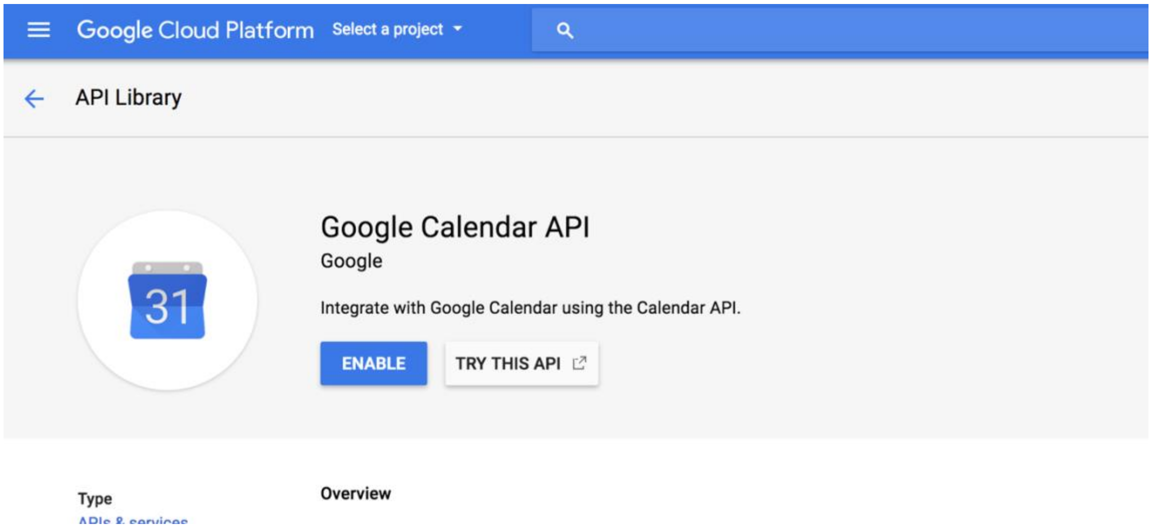
You have 12 projects remaining in your quota.

Project name: Binder Tasks

Your project ID will be binder-tasks-202820

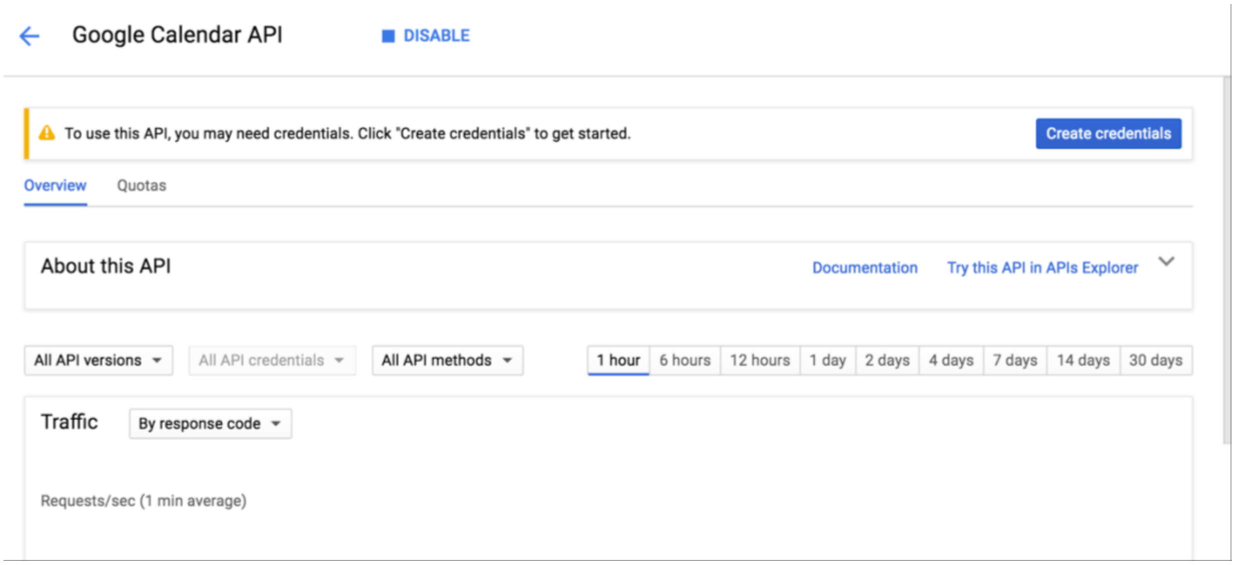
2. **Enable Google Calendar API** –visit https://console.cloud.google.com/apis/library/calendar-

json.googleapis.com/?q=calendar&id=84f291c9-2585-4af1-a78b-09c53a78202f and click the blue “ENABLE” button.



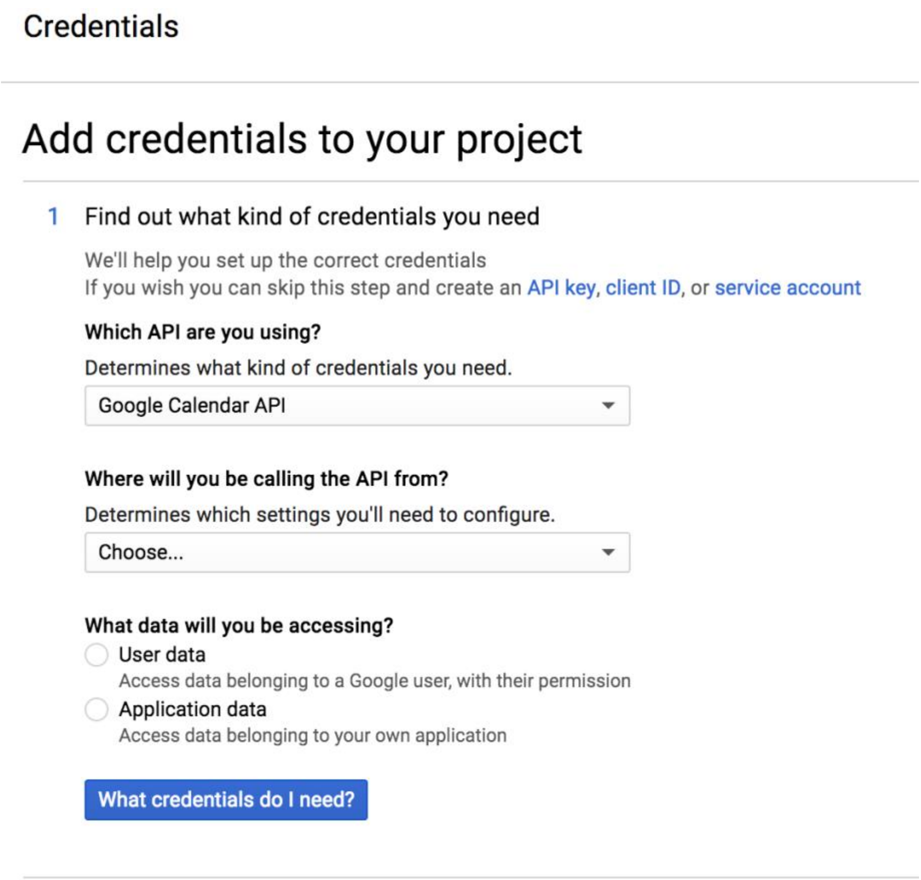
Integrate with Google Calendar using the Calendar API.

3. **Create a Service Account** – Click the blue “Create Credentials” button on the right side



To use this API, you may need credentials. Click "Create credentials" to get started.

Then, select “service account”.



Credentials

Add credentials to your project

1 Find out what kind of credentials you need

We'll help you set up the correct credentials
If you wish you can skip this step and create an API key, client ID, or service account

Which API are you using?

Determines what kind of credentials you need.

Google Calendar API

Where will you be calling the API from?

Determines which settings you'll need to configure.

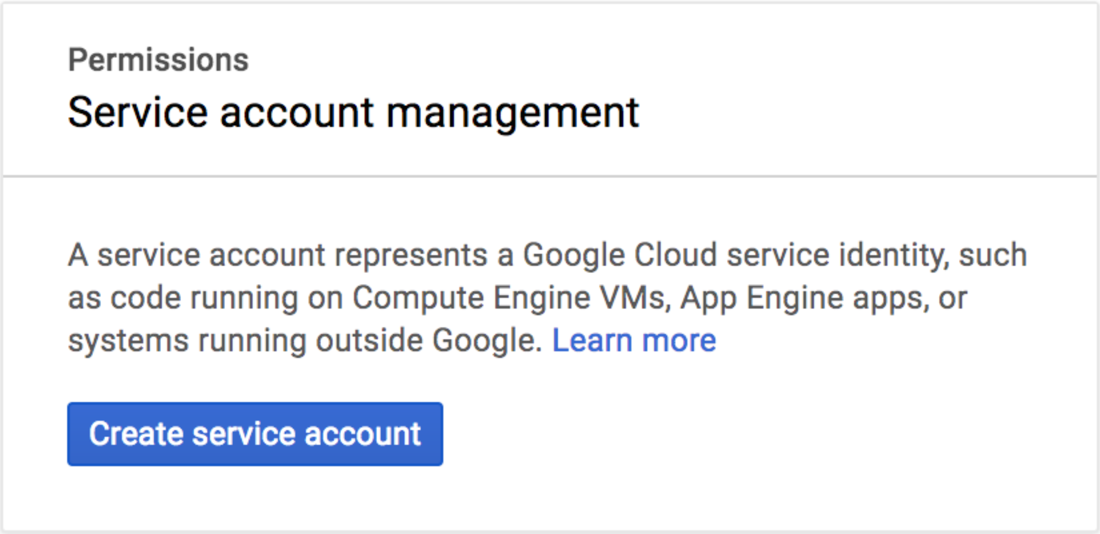
Choose...

What data will you be accessing?

○ User data
Access data belonging to a Google user, with their permission

○ Application data
Access data belonging to your own application

Then, click "Create Service Account".

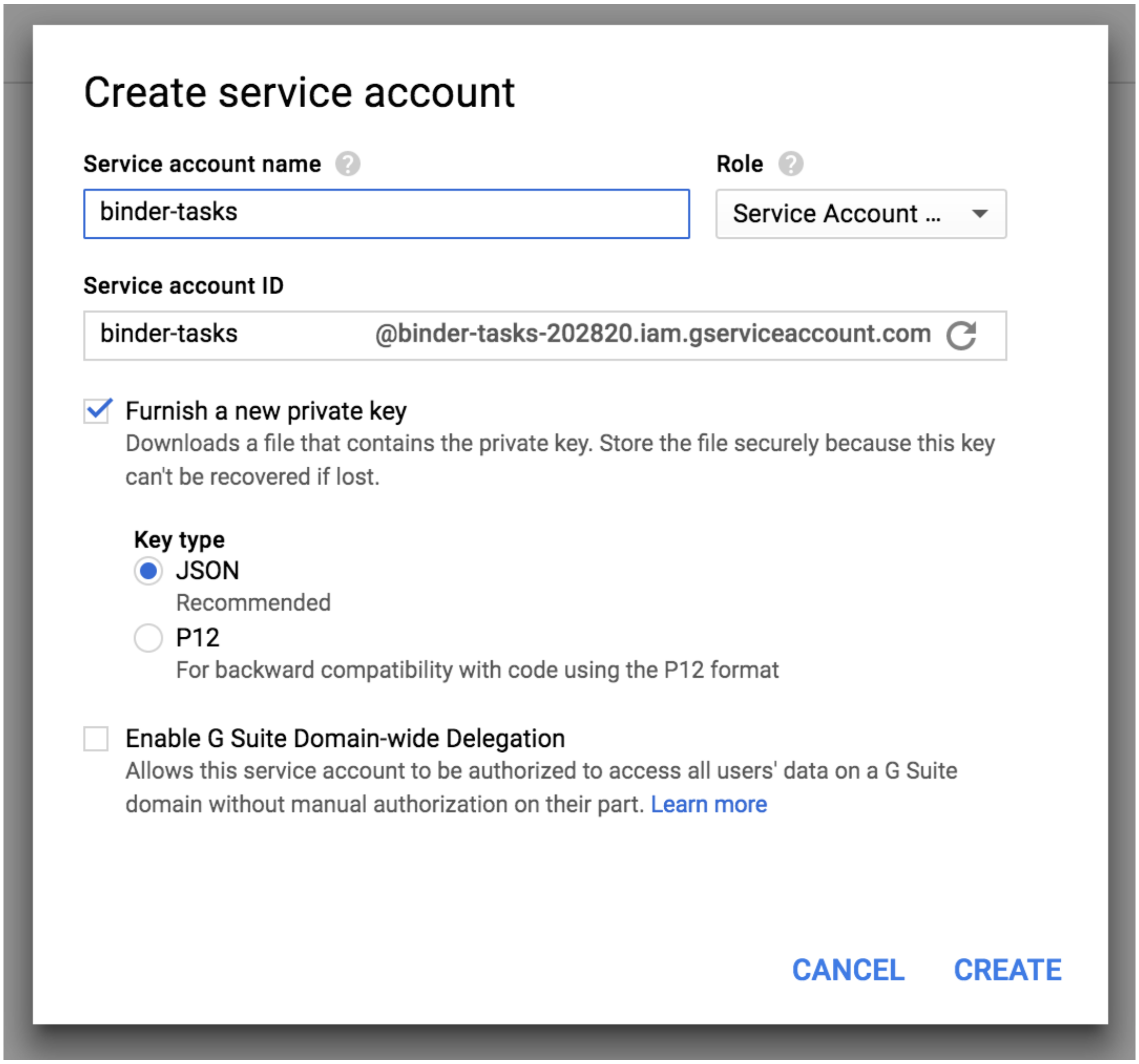


Permissions

Service account management

A service account represents a Google Cloud service identity, such as code running on Compute Engine VMs, App Engine apps, or systems running outside Google. Learn more

In the modal, create a name for your service account, select “Service Account Admin” as your role, and make sure that “Furnish a new private key” in JSON format is selected. A JSON file will be automatically downloaded for you.



Create service account

Service account name: binder-tasks

Role: Service Account ...

Service account ID: binder-tasks @binder-tasks-202820.iam.gserviceaccount.com

☑ Furnish a new private key
Downloads a file that contains the private key. Store the file securely because this key can't be recovered if lost.

Key type
● JSON
Recommended
○ P12
For backward compatibility with code using the P12 format

□ Enable G Suite Domain-wide Delegation
Allows this service account to be authorized to access all users' data on a G Suite domain without manual authorization on their part. Learn more

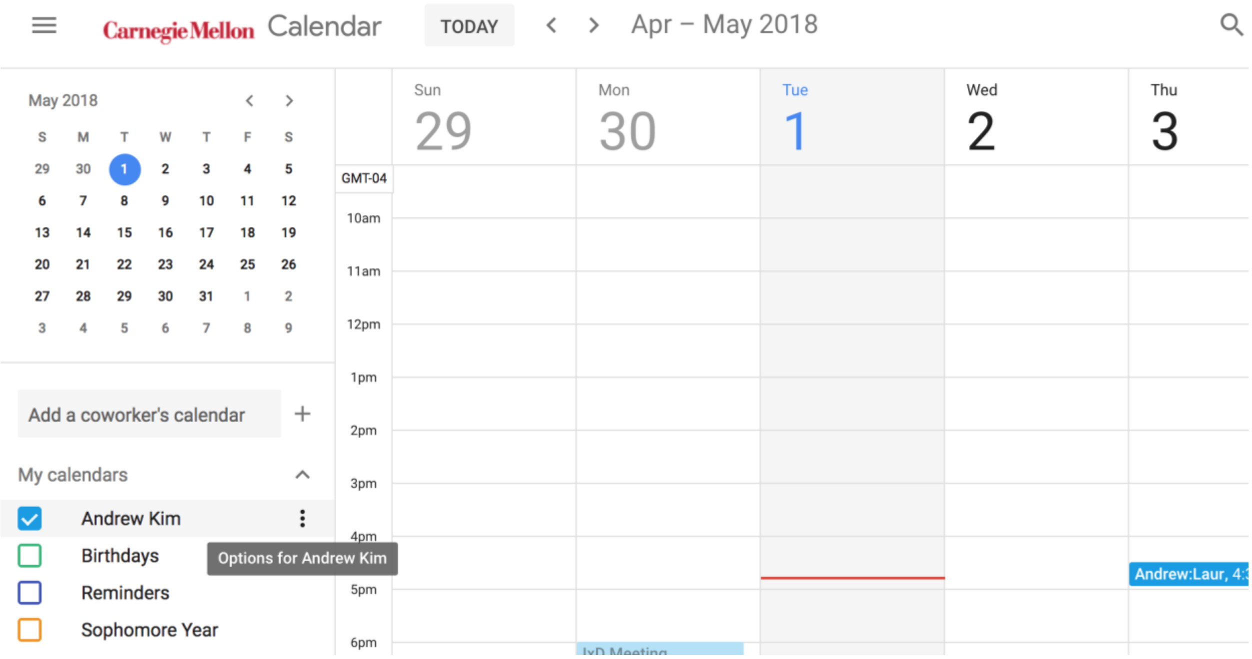
4. **Download the google-api-client gem -** `gem install google-api-client`

5. **Place the JSON file that was downloaded earlier in the lib folder. Name it “keyfile.json”.**

6. **Create a file called “google_calendar_helper.rb” in your lib folder with the following code.**

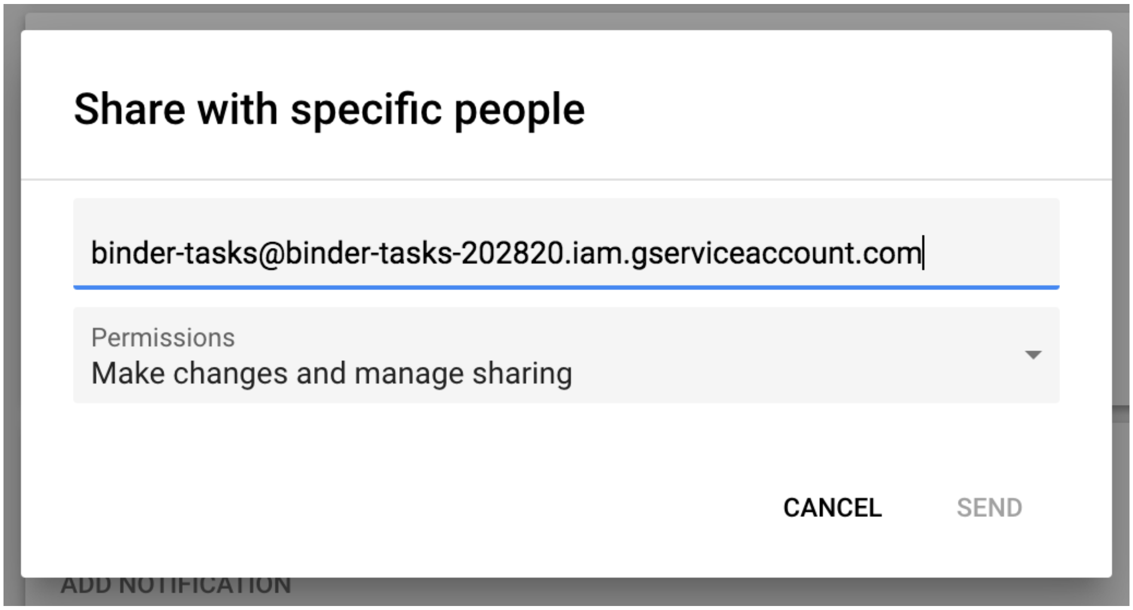
```
require 'googleauth'
require 'google/apis/calendar_v3'

class GoogleCalendarHelper
  attr_reader :calendar
  SCOPE = ["https://www.googleapis.com/auth/calendar"]
  CALENDAR_ID = 'YOUR-CALENDAR-ID-HERE'

  def initialize
    @calendar = Google::Apis::CalendarV3::CalendarService.new
    # get_application_default forces the use of this environment variable.
    # Set it to the location of your JSON credentials.
    ENV["GOOGLE_APPLICATION_CREDENTIALS"] = "lib/keyfile.json"
    @calendar.authorization = Google::Auth.get_application_default(SCOPE)
  end
end
```

7. **In the above code, replace “**YOUR-CALENDAR-ID-HERE” with the ID of the Google Calendar you want to connect. You can find the ID by clicking on the three dots next to a calendar, clicking “Settings and Sharing”, and scrolling down to find “Calendar ID”



S	M	T	W	T	F	S
29	30	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2
3	4	5	6	7	8	9

8. **Share the calendar with your service account** – stay on the “Settings and Sharing’ page and find the “Share with specific people” section. Share it with your service account (which should look something like “binder-tasks@binder-tasks-202820.iam.gserviceaccount.com”).Make sure that the account can “Make changes and manage sharing”.



Share with specific people

binder-tasks@binder-tasks-202820.iam.gserviceaccount.com

Permissions
Make changes and manage sharing

The Google Calendar account associated with the account you created should now be synced up with Binder!

--

This guide is a modification of the guide here http://cirefice.me/software%20development/integrating-rails-with-google-apis, credit goes to Christopher S. Cirefice.