

# Lepide Active Directory Self Service

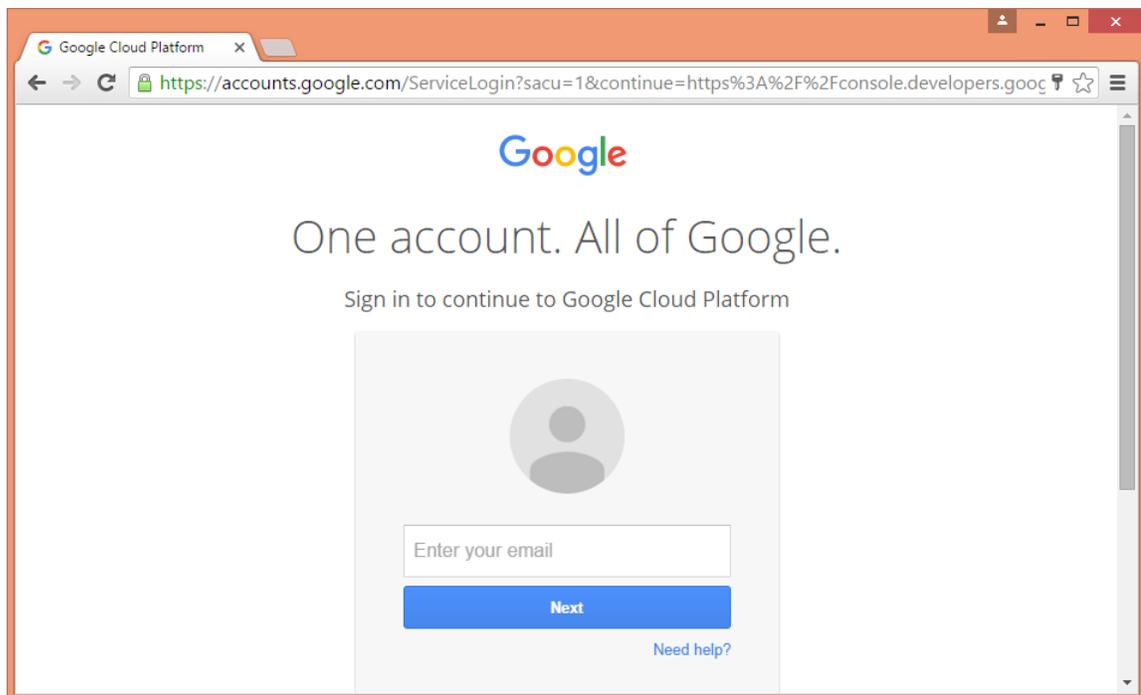


## How to generate P12 key

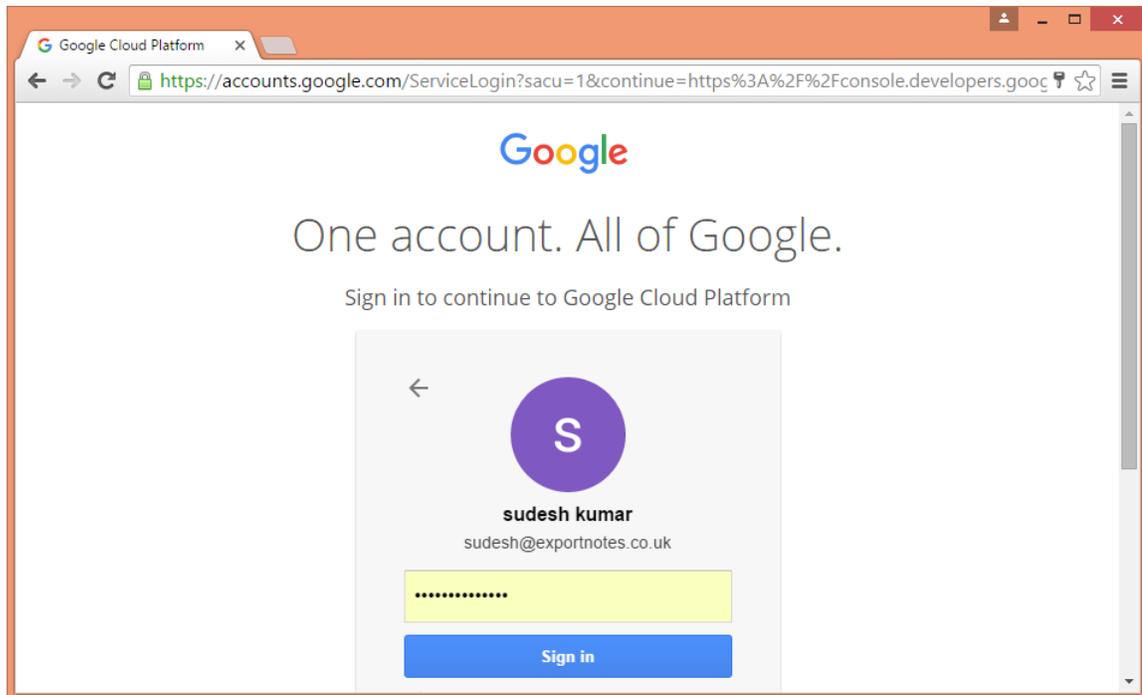
Lepide Active Directory Self Service allows Password Synchronization of Google Apps and IBM accounts. In order to enable Google Apps Password synchronization, you need to generate a P12 key by making certain settings in your Google service account.

Follow the below mentioned steps to generate the P12 key for Google App Service Account.

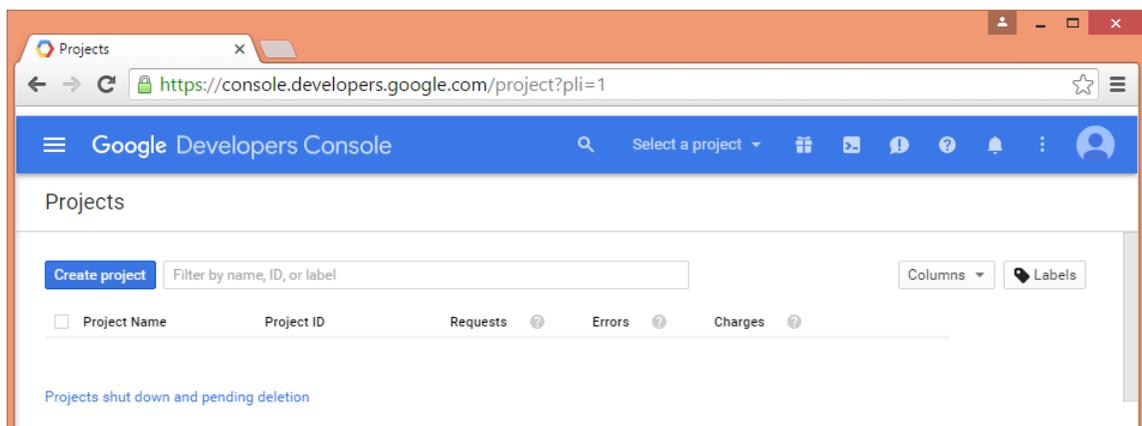
1. Open Google Developer Console Project for Google Cloud Platform - <https://console.developers.google.com/project> in Web browser, preferably Google Chrome.



2. Enter the email address of your Google Service Account and click "Next".



3. Enter the password of your Google Service Account and click "Sign In". It takes you to the Google Service Projects page.



4. Click "Create Project" to access "New Project" pop-up.

### New Project

**Project name** [?](#)

Your project ID will be based on your project name [?](#) [Edit](#)

[Show advanced options...](#)

5. The Project ID is assigned as per the project name. If you want to provide a manual project ID, click "Edit" link. You can also click "Show advanced options..." to access the advanced options.

### New Project

**Project name** [?](#)

**Project ID** [?](#)

[Hide advanced options...](#)

**App Engine location** [?](#)

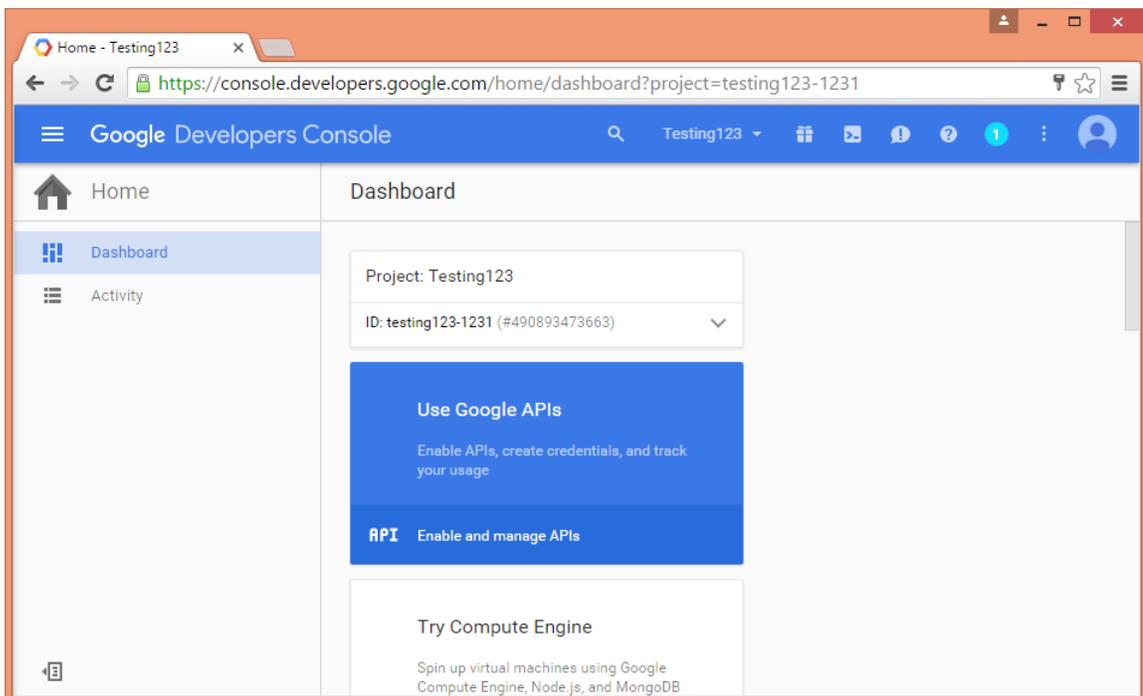
6. Enter the Project Name, Project ID, and select the advanced option. Click "Create" to create the project.

### New Project

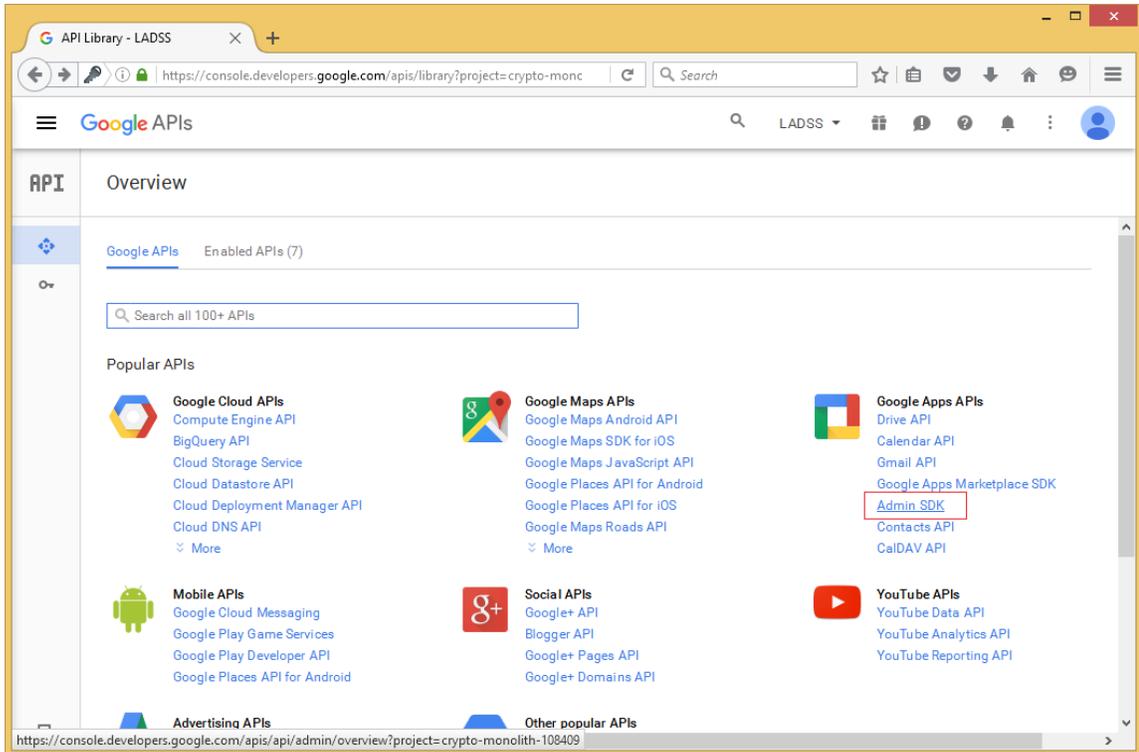
**Project name** ?

  
**Project ID** ? ↻  
[Hide advanced options...](#)  
**App Engine location** ? ▼

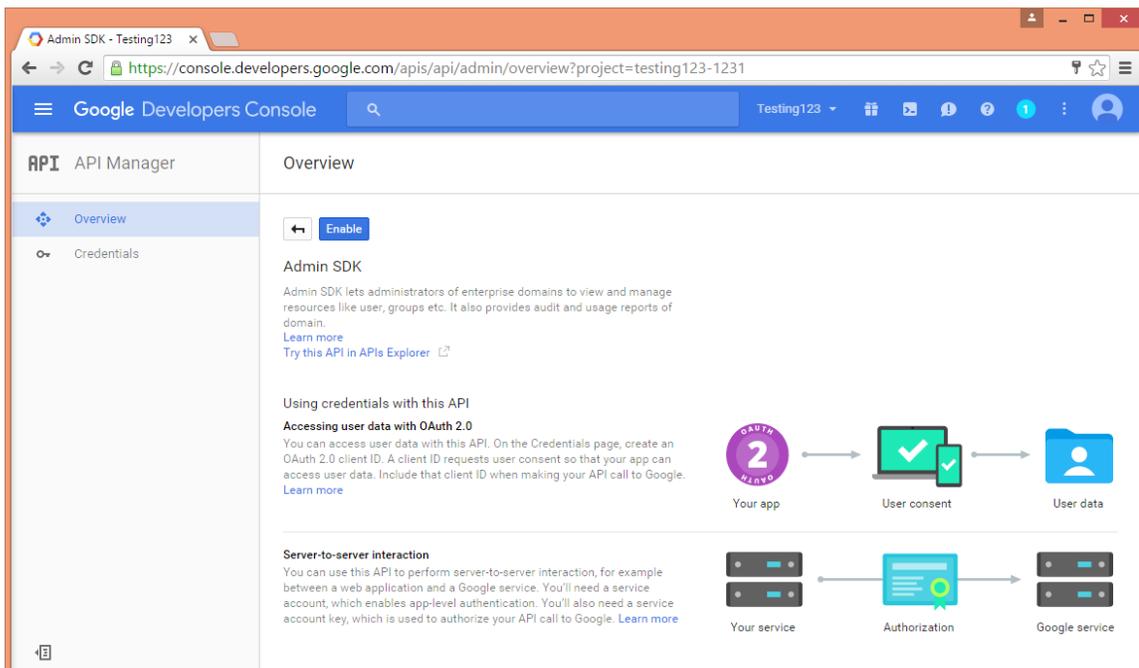
- Once created, the dashboard comes up. Click "Enable and Manage APIs" in "Use Google APIs" section. It displays the popular APIs.



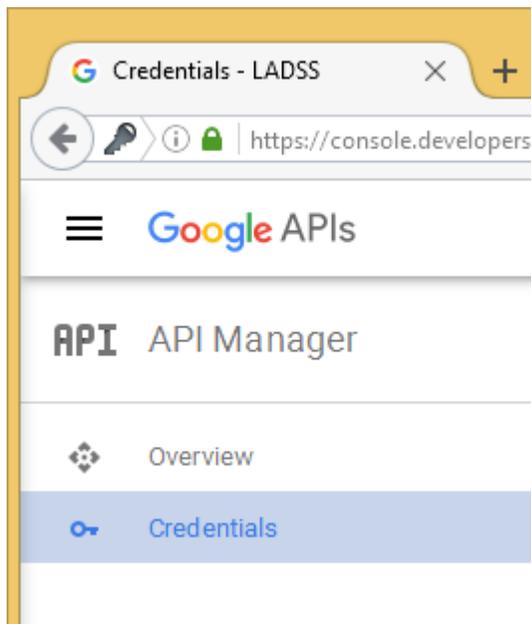
- Click 'Admin SDK' API link to access its settings.



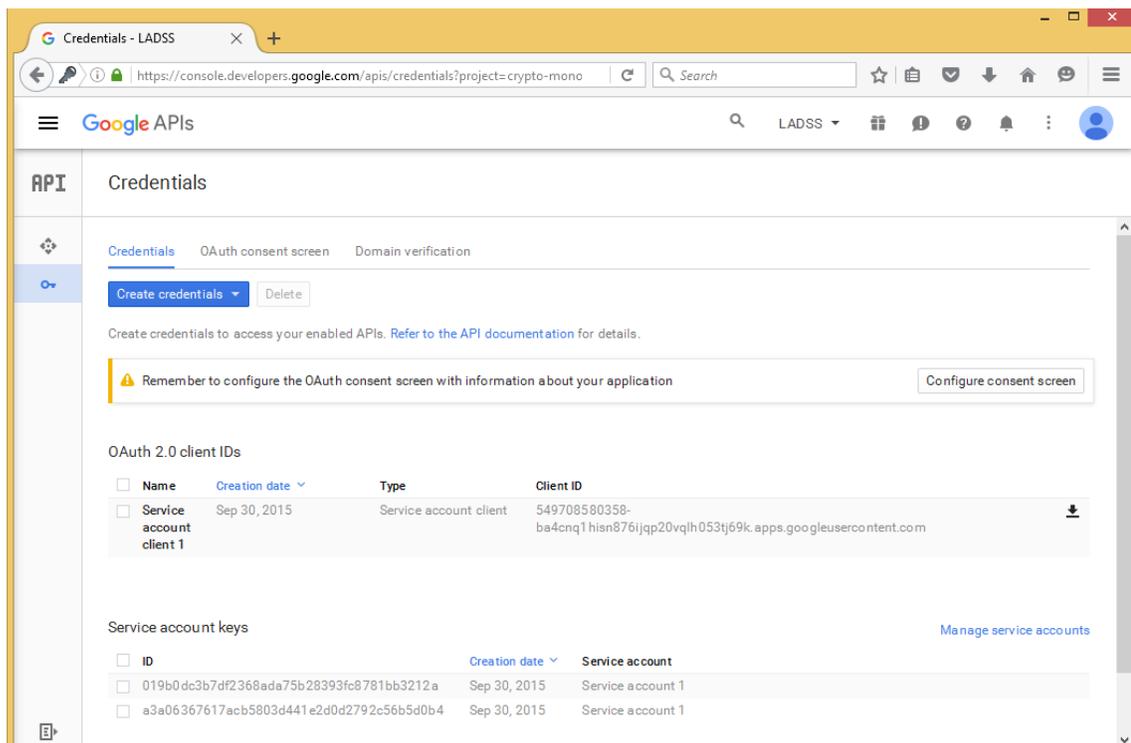
9. Click the **Enable** button to enable Admin SDK API for the service account.



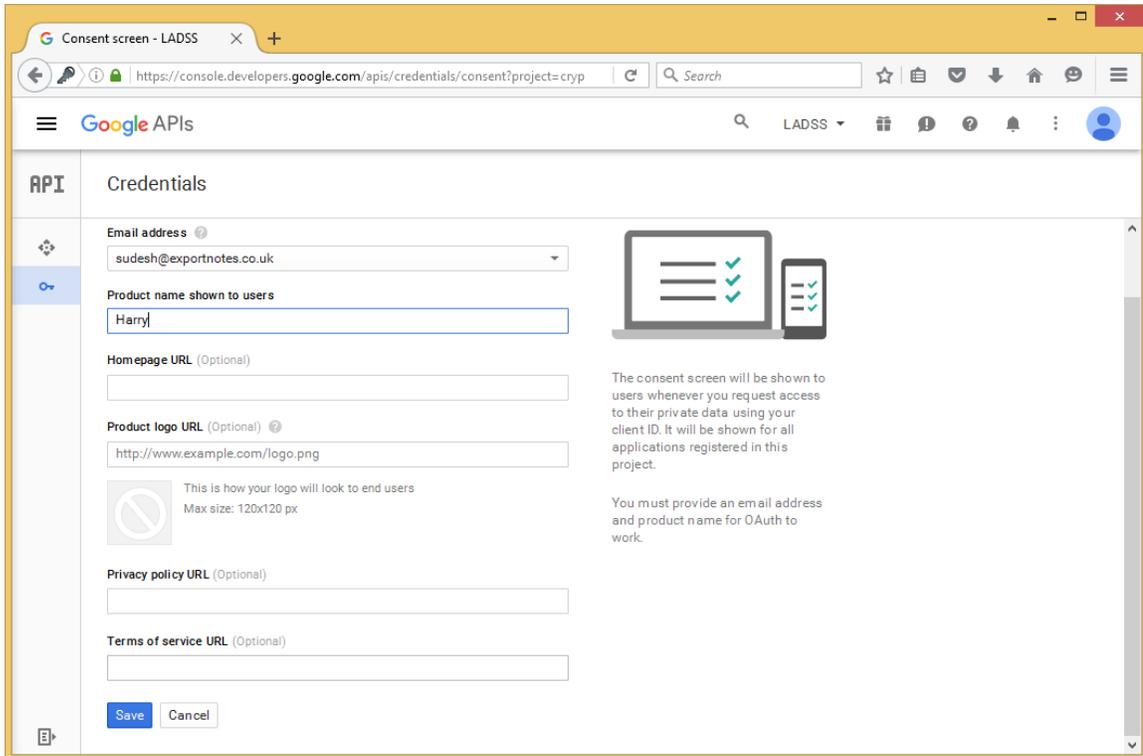
10. Now from the left pane of API Manager select credential.



11. The following page will open up. Click on the "OAuth consent screen" tab.



12. In the "OAuth consent screen" page, fill the form to complete the project registration. Click "Save".

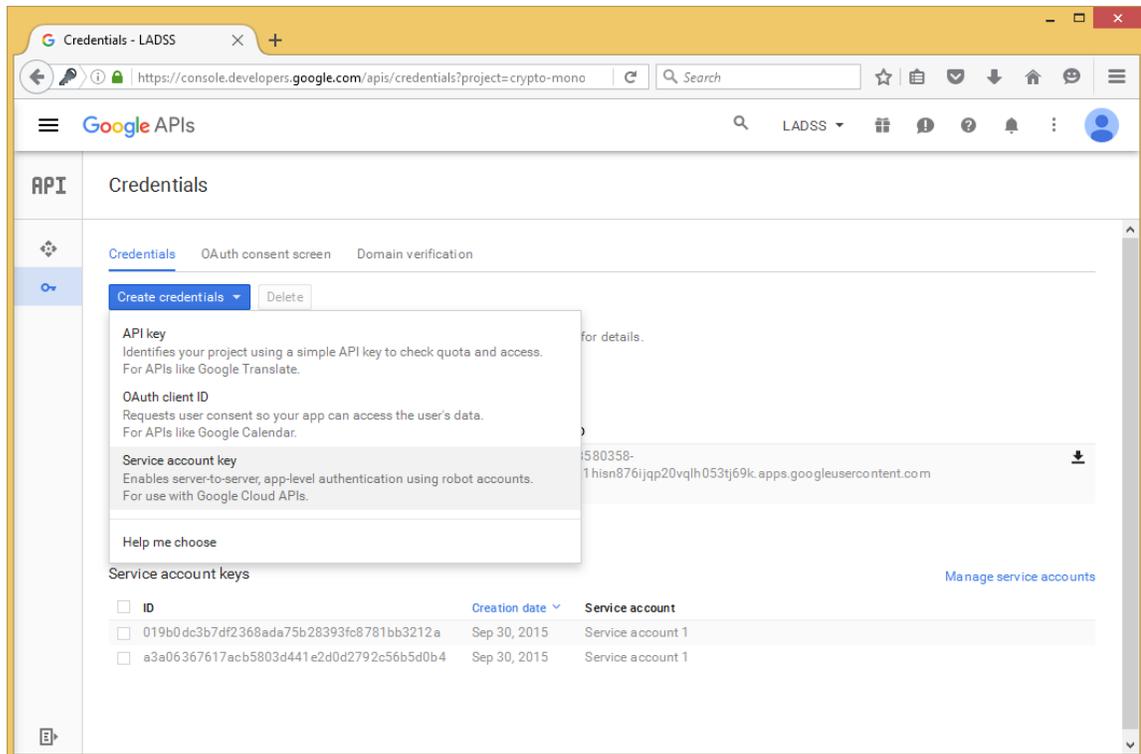


The screenshot shows the Google APIs console's OAuth consent screen registration page. The browser address bar shows the URL: `https://console.developers.google.com/apis/credentials/consent?project=cryp`. The page title is "Consent screen - LADSS". The left sidebar shows the "API" section with "Credentials" selected. The main content area contains the following fields:

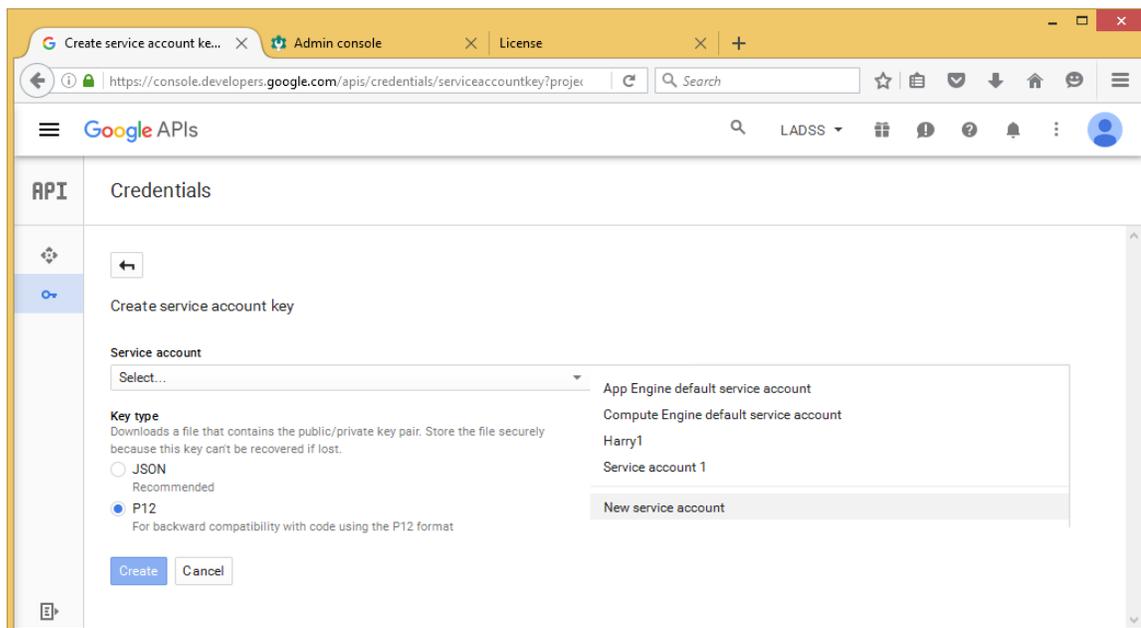
- Email address:** A dropdown menu with the value "sudesh@exportnotes.co.uk".
- Product name shown to users:** A text input field containing "Harry".
- Homepage URL (Optional):** An empty text input field.
- Product logo URL (Optional):** A text input field containing "http://www.example.com/logo.png".
- Product logo:** A placeholder image with a "no" symbol and the text "This is how your logo will look to end users. Max size: 120x120 px".
- Privacy policy URL (Optional):** An empty text input field.
- Terms of service URL (Optional):** An empty text input field.

At the bottom of the form are "Save" and "Cancel" buttons. On the right side of the form, there is an illustration of a laptop and a smartphone, and a text box explaining that the consent screen will be shown to users when requesting access to their private data. Below this, a note states: "You must provide an email address and product name for OAuth to work."

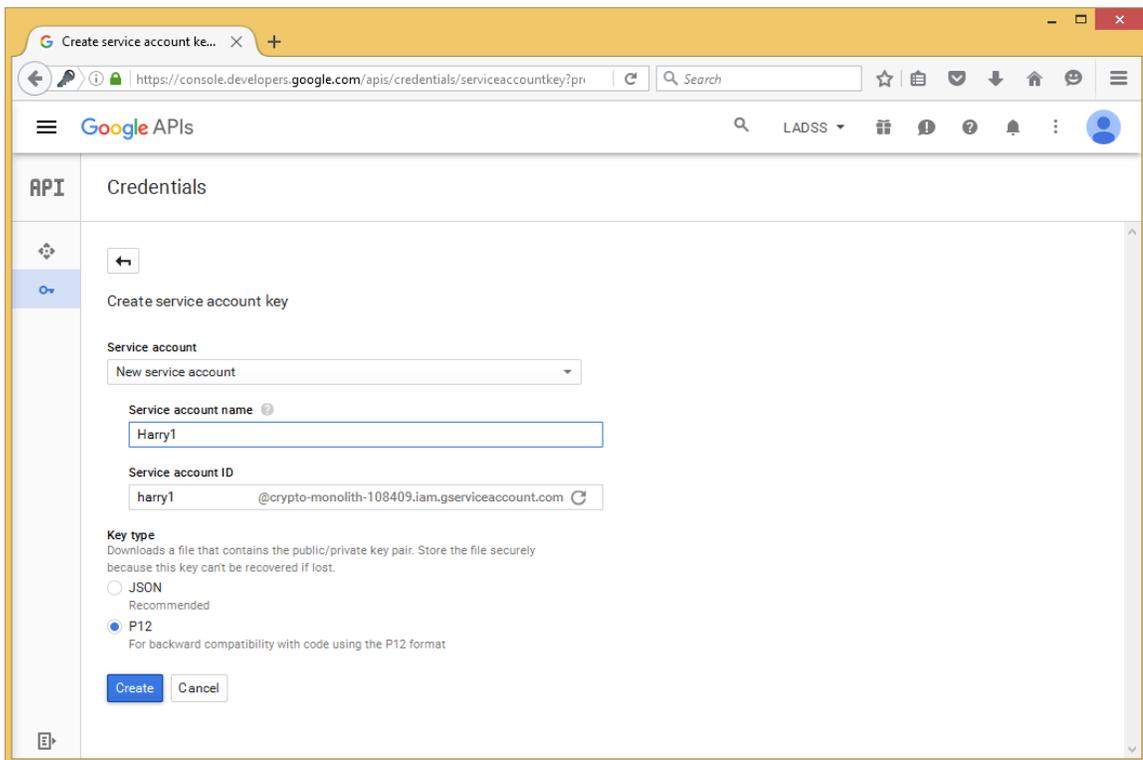
13. You will be redirected back to the Credentials tab. Now, click on "Create credential" and select service account key from the dropdown menu.



14. Select P12 as the key type and create a new service account by selecting it from the 'service account' dropdown menu.



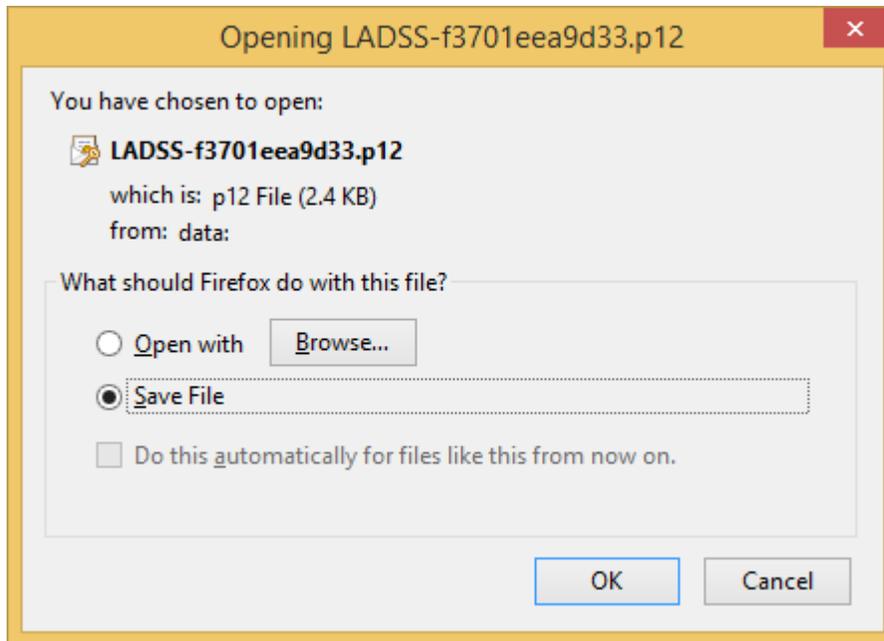
15. Enter the service account name and service account ID. Click "Create".



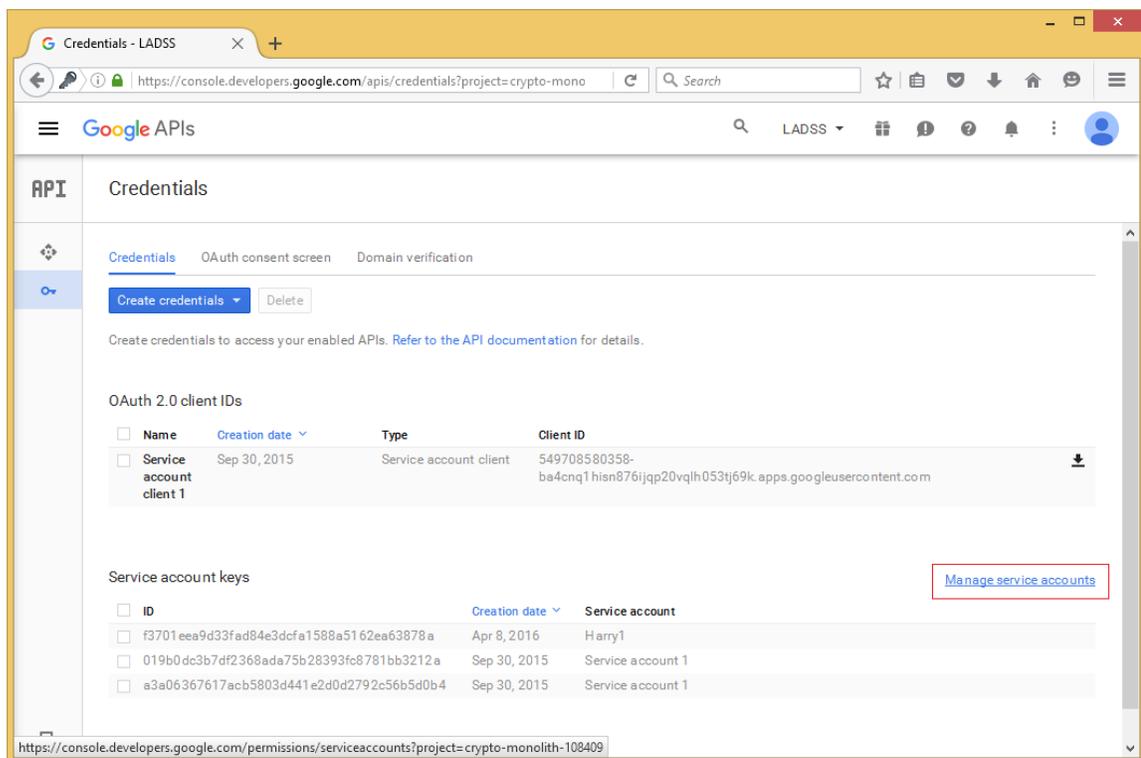
The screenshot shows the 'Create service account key' page in the Google Cloud Platform console. The page is titled 'Create service account key' and is part of the 'Credentials' section. The 'Service account' dropdown is set to 'New service account'. The 'Service account name' field contains 'Harry1'. The 'Service account ID' field contains 'harry1' and '@crypto-monolith-108409.iam.gserviceaccount.com'. The 'Key type' section has two options: 'JSON' (Recommended) and 'P12' (selected). The 'P12' option is selected with a blue radio button. Below the 'Key type' section are 'Create' and 'Cancel' buttons.

16. As soon as the service account is created, a P12 key file is generated. Save the generated p12 file to a safe location.

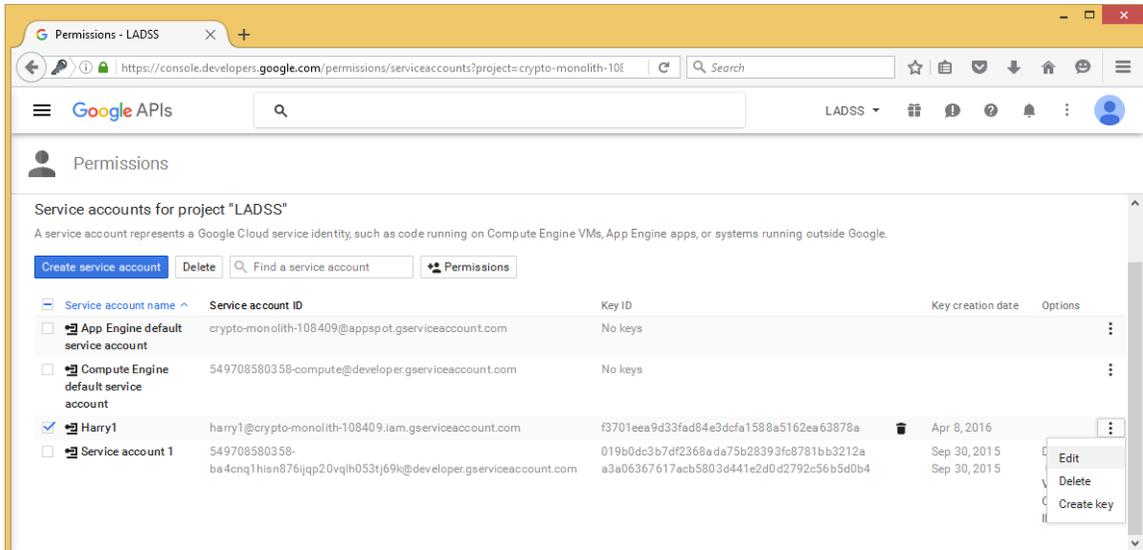
**Note:** This is the P12 key file which you need to browse while adding Google Apps for password synchronization.



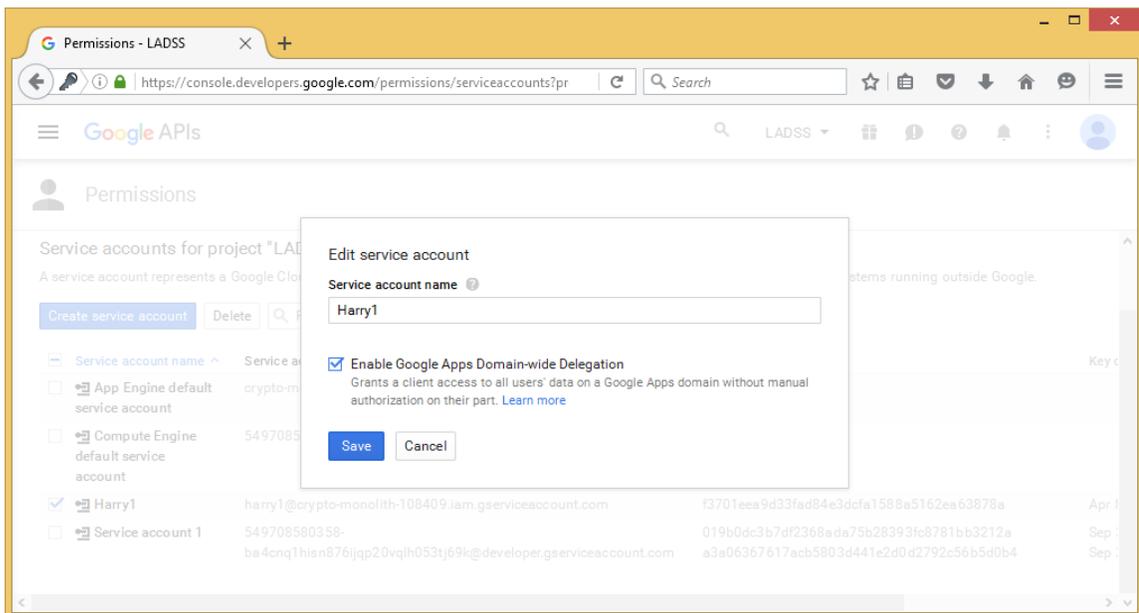
17. You will be redirected back to the Credentials tab. Now, click on "Manage service accounts".



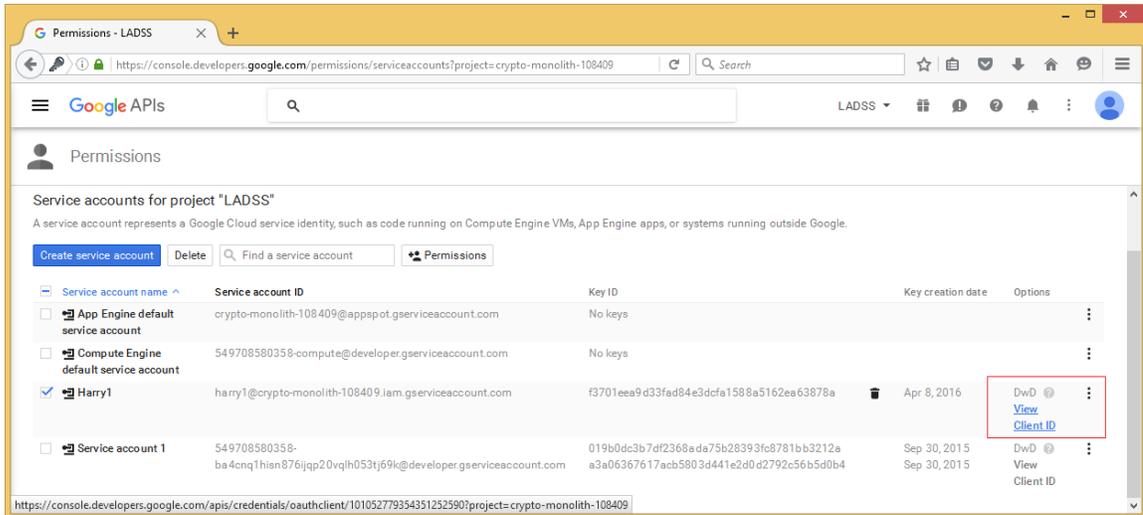
18. Select the service account created and click edit from options (three vertical dots at the end of account record).



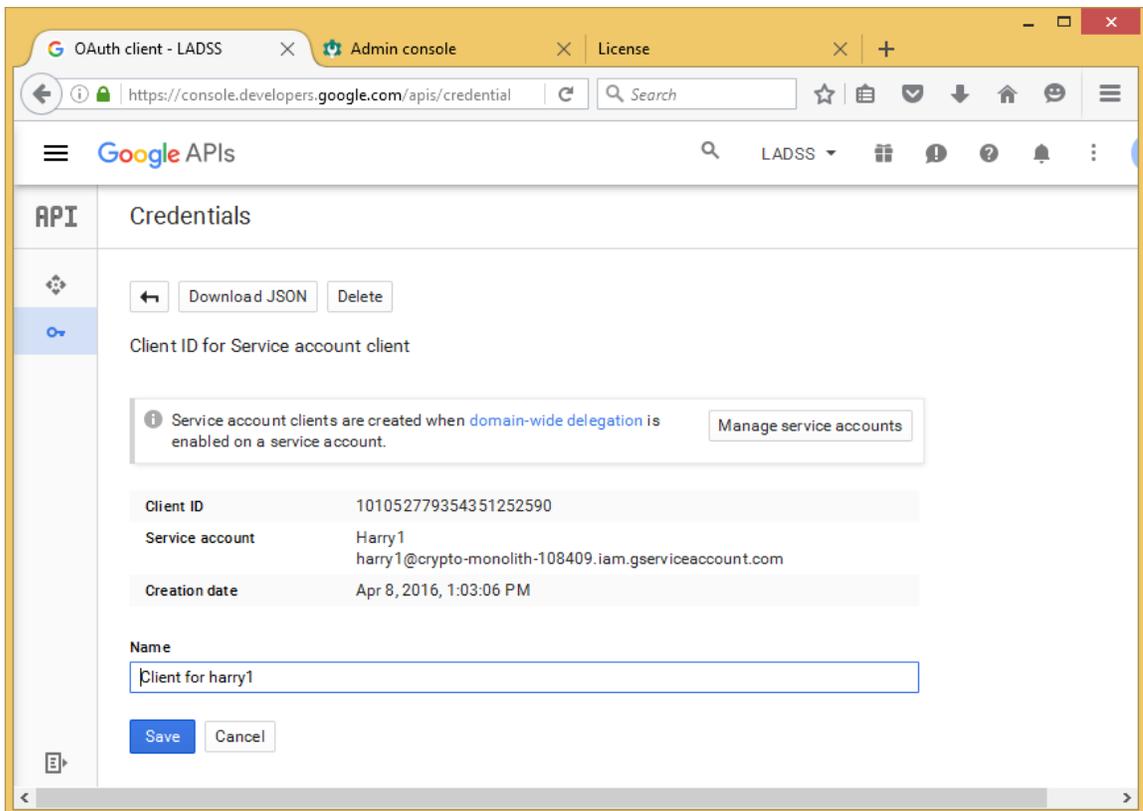
19. Select "Enable-Google Apps Domain-wide Delegation" to grant domain wide access without logging to the service account. See this link for more details: [https://developers.google.com/admin-sdk/directory/v1/guides/delegation#delegate\\_domain-wide\\_authority\\_to\\_your\\_service\\_account](https://developers.google.com/admin-sdk/directory/v1/guides/delegation#delegate_domain-wide_authority_to_your_service_account)



20. Now click on View Client ID.

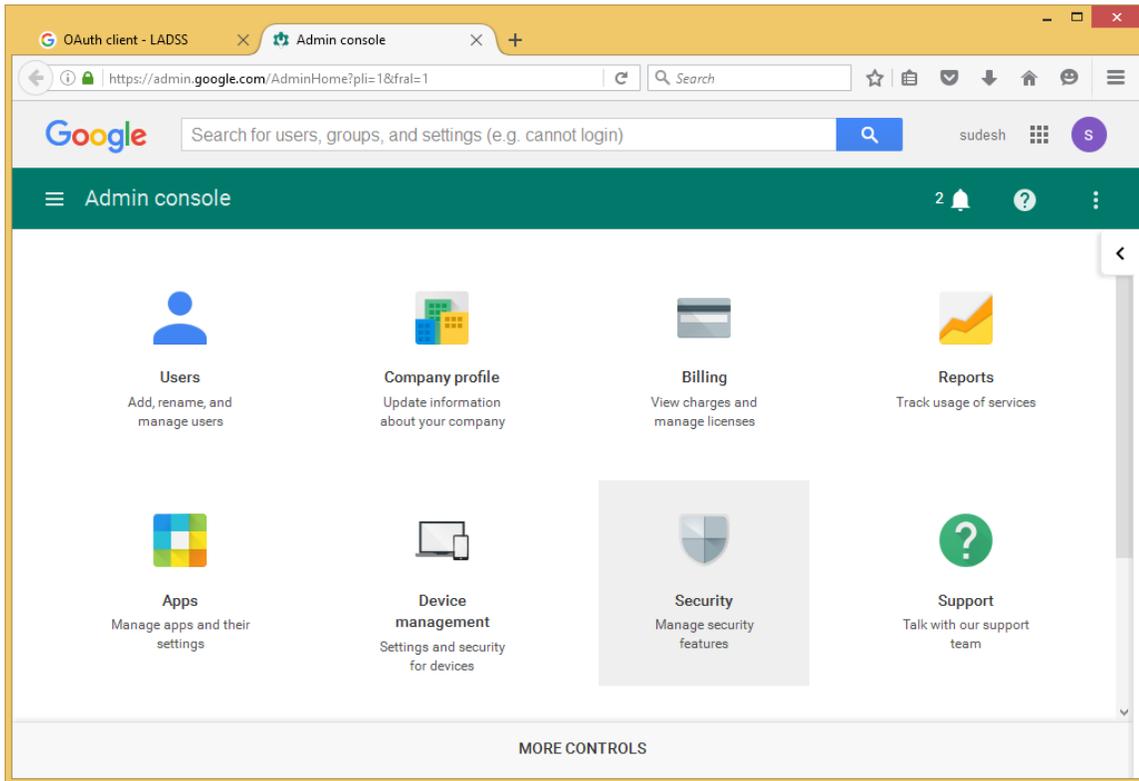


21. Copy the Client ID and Service account email in a safe location for further use.



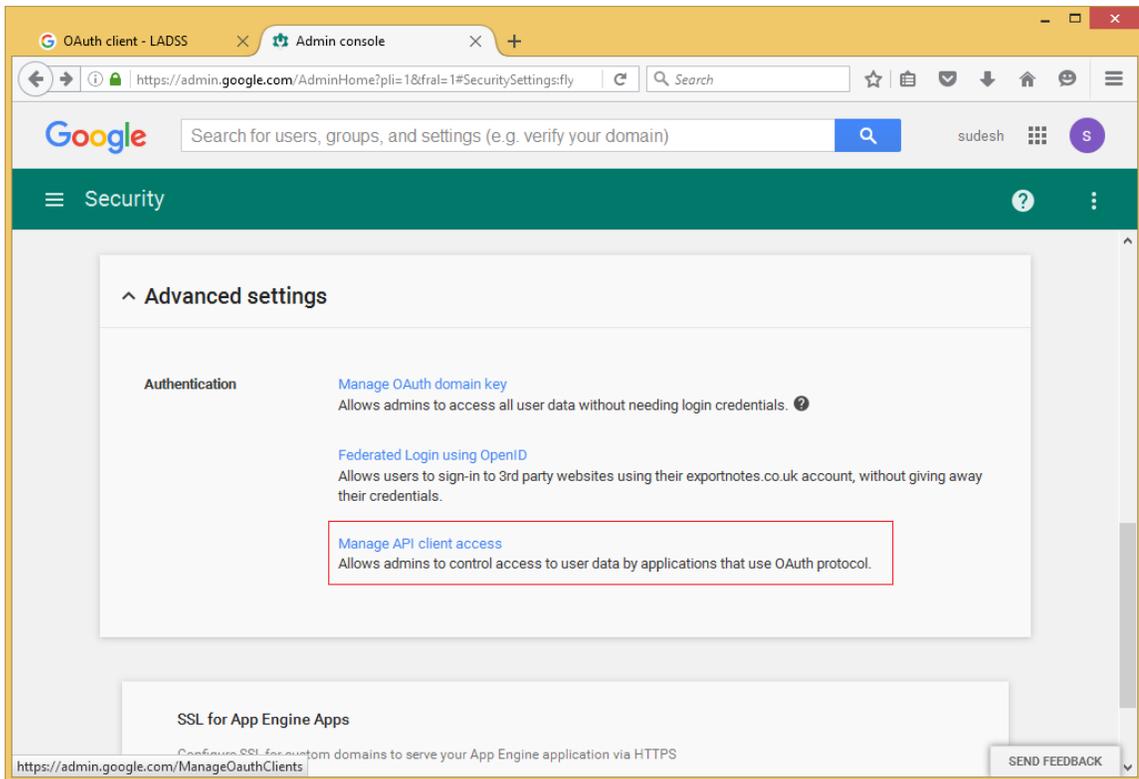
22. Now go to Google Admin Console (<https://admin.google.com>) and login using your Google Apps Administrator account.

23. Click on security tile.



24. Click show more and go to Advanced Setting.

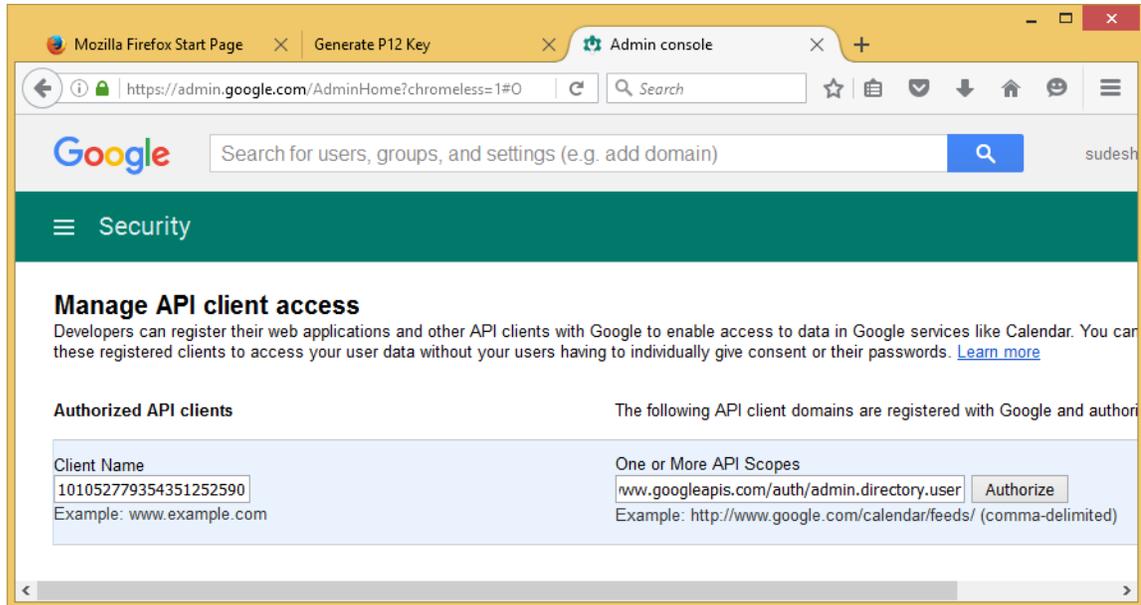
25. Click on Manage API Access.



26. Enter the Client ID generated earlier from developer console in client name and enter the following scopes in the scope field (use comma to separate them).

*<https://www.googleapis.com/auth/admin.directory.user>, <https://www.googleapis.com/auth/admin.directory.group>*

27. Click 'Authorize'.



28. That's it. You can start using the details in the Software.

Once the Password Synchronization settings are completed, Google Apps password resets can be successfully performed by end users.