Sardar Kulveer Singh

Performing API Calls to Salesforce Application Using OAuth 2.0 Authorization with Third-Party Tools like Postman

Table of Contents

**Purpose1**

**Scope1**

**Prerequisite1**

**Creating Connected App (Salesforce) with Oauth2.01**

**Authorizing Postman using Oauth2.05**

**Performing API Calls using Postman6**

# Purpose:

The purpose of this document is to provide instructions on how to perform API calls to a Salesforce application using a third-party tool like Postman, utilizing the OAuth 2.0 authorization method. This guide aims to enable developers or administrators to authenticate and access Salesforce resources programmatically through the Salesforce REST API using Postman as the HTTP client.

# Scope:

This document covers the process of obtaining an access token through the OAuth 2.0 authentication flow, configuring Postman for API calls, and making authenticated requests to Salesforce. It does not cover the specific details of individual API endpoints or specific use cases for Salesforce integration.

# Prerequisite:

Before following the steps outlined in this document, the following prerequisites should be met:

* A Salesforce Developer account or access to a Salesforce org.
* A Connected App created in Salesforce with OAuth 2.0 settings enabled and appropriate scopes defined.
* Postman installed on your machine (available at <https://www.postman.com/downloads/>).

# Procedure:

* Obtain OAuth 2.0 credentials.
* Configure Postman for OAuth 2.0
* Make authenticated API calls.

# Creating connected app (salesforce) with Oauth2.0:

## Obtain OAuth2.0 credentials:

To create a Connected App in Salesforce, follow these steps:

* Log in to your Salesforce account.
* Click on the "App Launcher" (grid icon) at the top-left corner of the Salesforce home page.
* In the App Launcher, search for and select "App Manager”.

A screenshot of a computer

Description automatically generated

* Click on the "New Connected App" button.

A screenshot of a computer

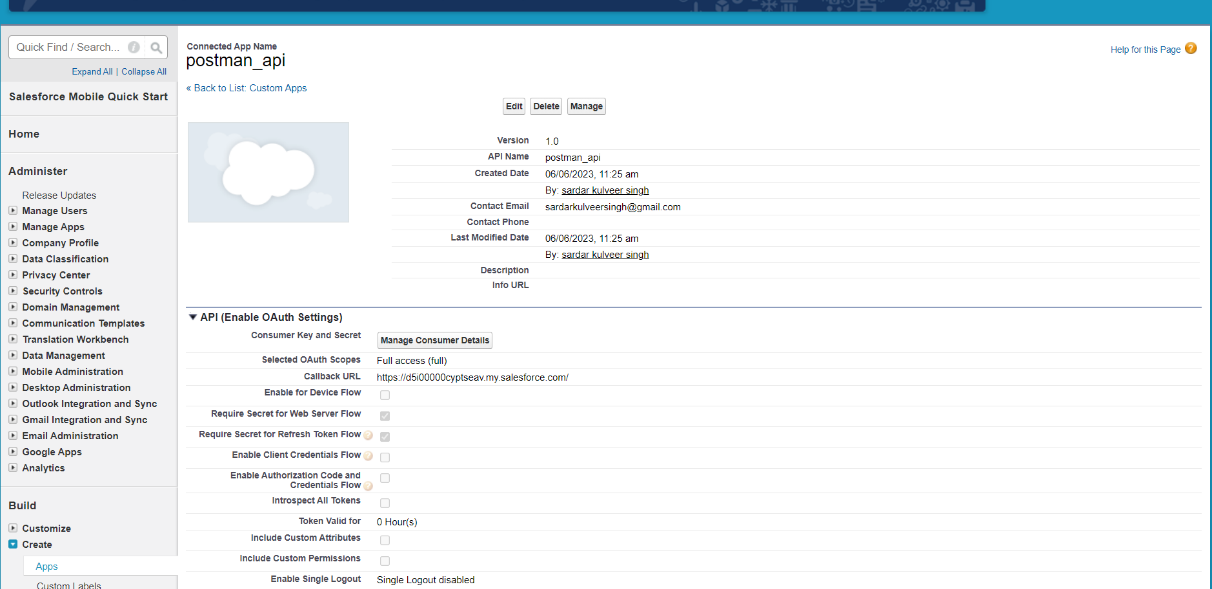
Description automatically generated with medium confidence

* Provide the following details for your Connected App

A screenshot of a computer

Description automatically generated with medium confidence

* Connected App Name: Enter a name for your app.
* API Name: The API name is automatically populated based on the Connected App Name.
* Contact Email: Enter your email address.
* Enable OAuth Settings: Check this checkbox to enable OAuth 2.0 settings for your app.
* Call-back URL: Enter the call back URL where users will be redirected after authorization. This can be a dummy URL for now, as it will be updated later.
* Selected OAuth Scopes: Select the appropriate OAuth scopes based on the level of access your app requires.
* Require Secret for Web Server Flow: Check this checkbox if you want to use the Web Server Flow and require a client secret.
* Use Digital Signatures: Check this checkbox if you want to use digital signatures for requests.
* Click on the "Save" button.
* On the next screen, you'll see the details of your Connected App, including the Consumer Key and Consumer Secret.



A picture containing text, font, screenshot, algebra

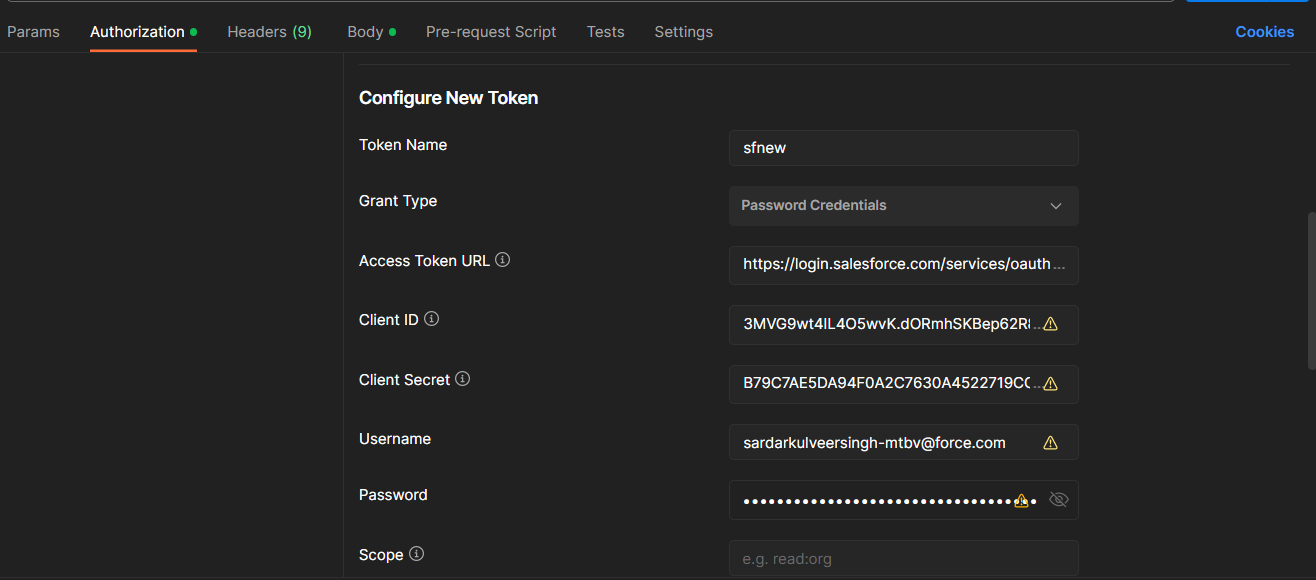
Description automatically generated

* Under the "API (Enable OAuth Settings)" section, click on the "Edit" button to update the call back URL.
* In the "Call-back URL" field, enter the actual call back URL that should receive the authorization response.
* Click on the "Save" button to save the changes.

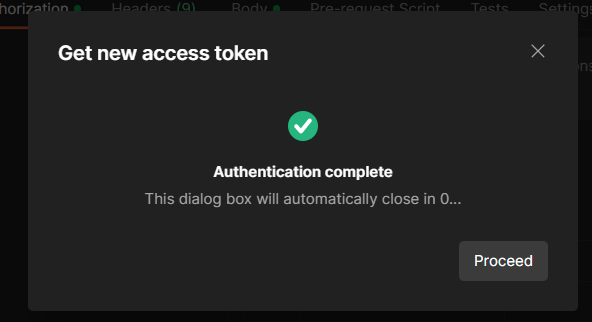
# **Authorizing Postman using Oauth2.0:**

## Configure Postman for OAuth 2.0:

* Open Postman and create a new request.
* In the request body, include the required parameters:



* grant\_type: Set to "password" if using username and password for authentication, or "authorization\_code" if using the authorization code flow.
* client\_id: The Consumer Key of your Connected App.
* client\_secret: The Consumer Secret of your Connected App.
* username (for password flow): The Salesforce username.
* password (for password flow): The Salesforce password concatenated with the security token.
* redirect Uri (for authorization code flow): The call back URL specified in the Connected App settings.
* Send the request to obtain an access token



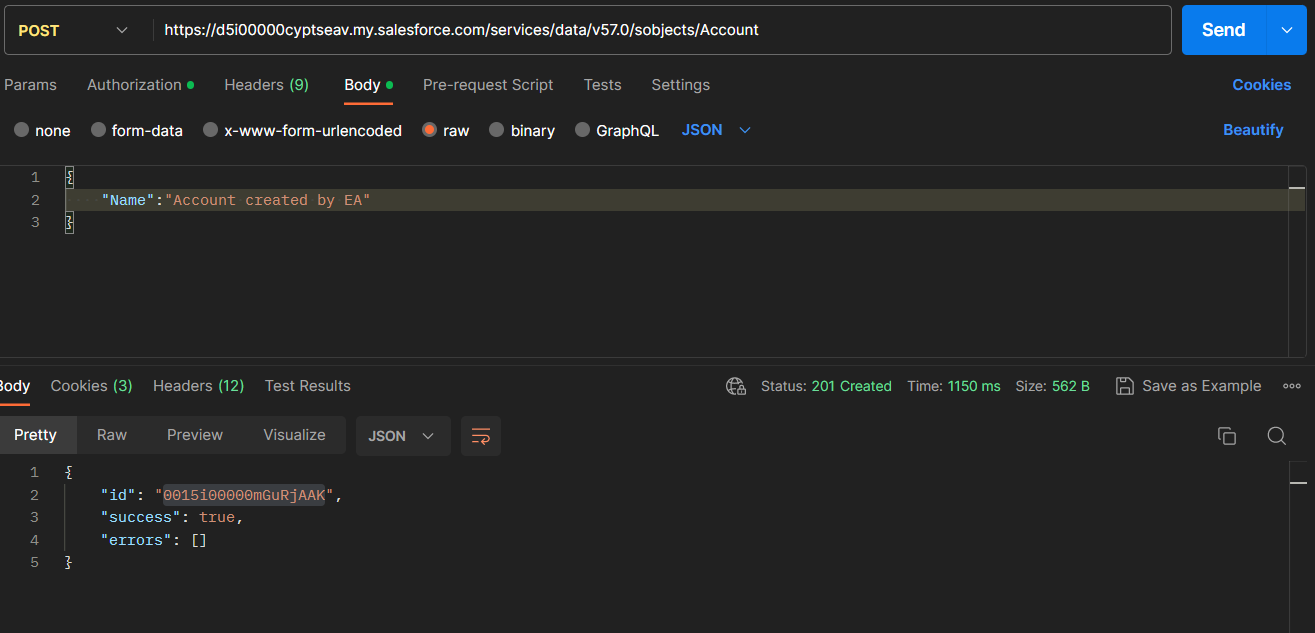
A screenshot of a computer

Description automatically generated

# Performing API Calls using Postman:

Make authenticated API calls:

* Copy the access token received in the response.
* Create a new request in Postman to access the desired Salesforce API endpoint.
* Set the request type, URL, headers, and parameters according to the specific API documentation.
* Include the access token in the Authorization header in the format: Authorization: Bearer <access\_token>.
* Send the request and review the response from Salesforce.



* Above post request creates an account in salesforce using the account create api

A screenshot of a computer

Description automatically generated with medium confidence

* Above get request gets the details of the account created from salesforce using the Api Account/accountid