

# Configure Webex Contact Center APIs with Postman as an Application

## Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Configure](#)

[Verify](#)

## Introduction

This document describes how to configure the Webex Contact Center (WxCC) APIs with Postman as an application

## Prerequisites

## Requirements

Cisco recommends that you have knowledge of these topics:

- Webex Contact Center (WxCC) 2.0
- Customer Administrator account into a your tenant
- Use Postman API platform

## Components Used

The information in this document is based on these software versions:

- WxCC 2.0
- Postman API platform
- Webex Developer portal

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

## Configure

**Step 1:** Log in to the Developer portal with your customer administrator access:

## [Contact Center for Developers](#)

**Step 2:** Select the Create a New App icon.



**Step 3:** Enter a new name for the integration, description, Redirected URI, and the Scope as shown:

If you use Postman as an App, redirect URL for Postman: <https://oauth.pstmn.io/v1/callback>

If you use Postman in the browser, redirect URL for Postman: <https://oauth.pstmn.io/v1/browser-callback>

**Integration Name\***  
Name of your integration  
Partner\_summit Edit

**Description\***  
Provide some details about what your integration does, how it benefits users, and most importantly, how a user can get started using it. The description should be under 1024 characters.  
Partner summit Edit

**Redirect URI(s)\***  
One or more URIs that a user will be redirected to when completing an OAuth grant flow.  
<https://oauth.pstmn.io/v1/browser-callback> Edit

**Scopes\***  
Scopes define the level of access that your integration requires.

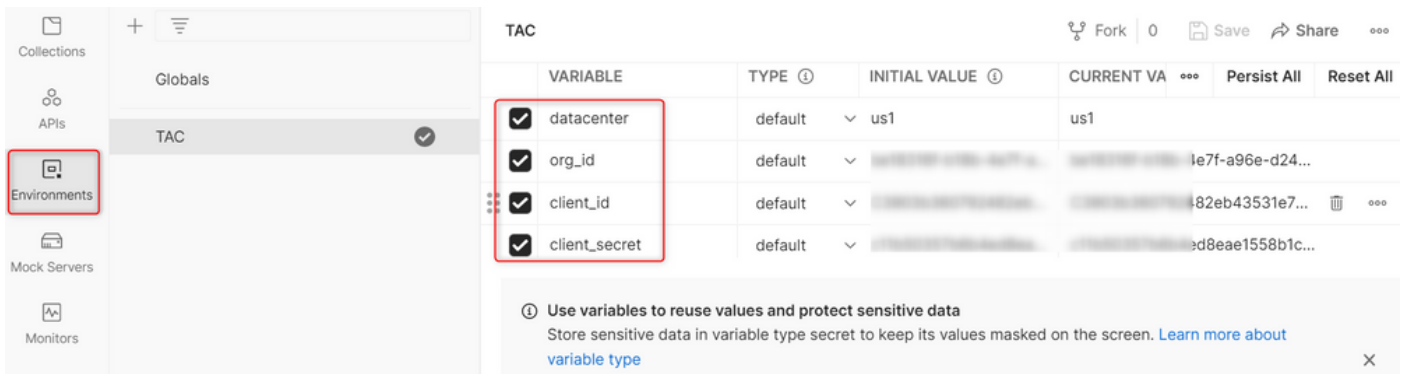
- cjp:config
- cjp:config\_write
- cjp:config\_read
- spark:people\_read
- cjp:user

**Step 4:** On Postman, create the Environment variables.

The variables can be global (applied to all that use these variables) or it can be local. This is the screen shot environments of all the variables created to be used locally:

Variables to be passed are:

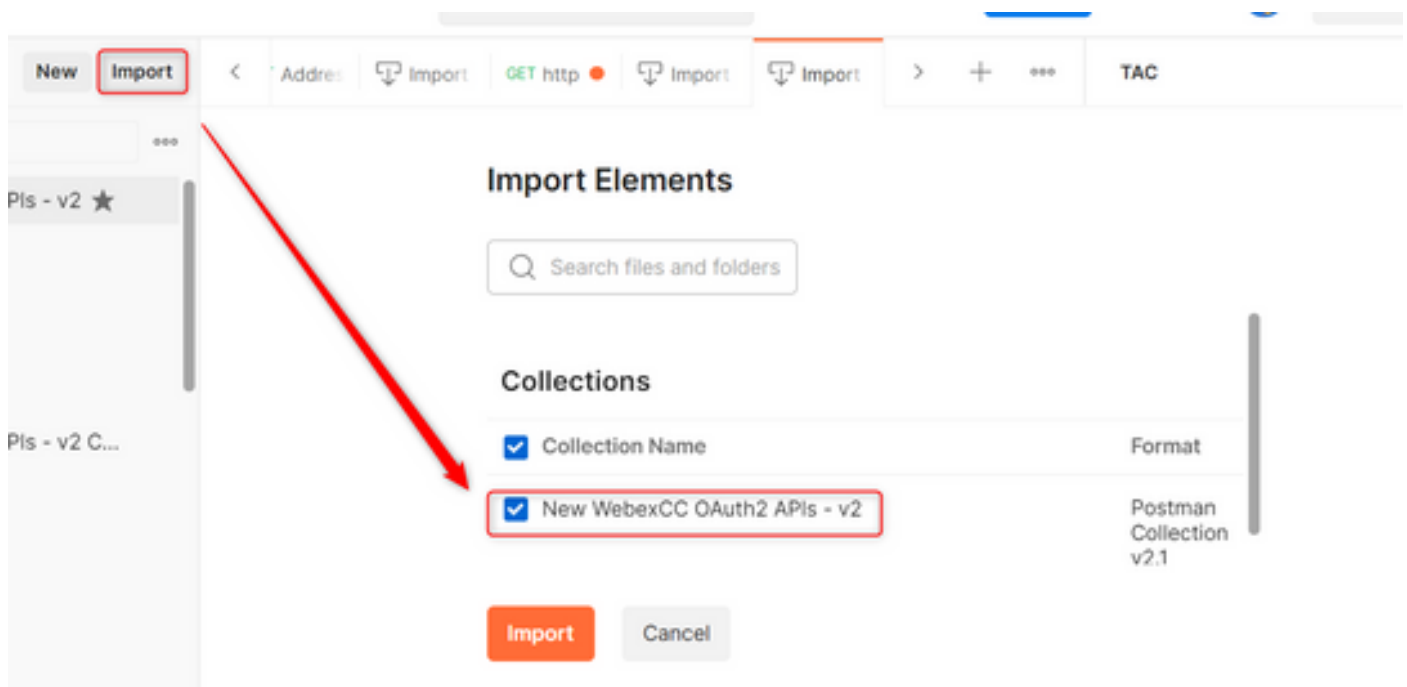
- **datacenter** - The Datacenter is us1 for ProdUS1, eu1 and eu2 for ProdeU1 and ProdeU2, anz1 for ProdANZ1
- **org\_id** - Enter the Org ID of the tenant
- **client\_id** - Enter the client ID generated on the Webex App on the Developer portal
- **client\_secret** - Enter the Client Secret generated on the Webex App on Developer portal



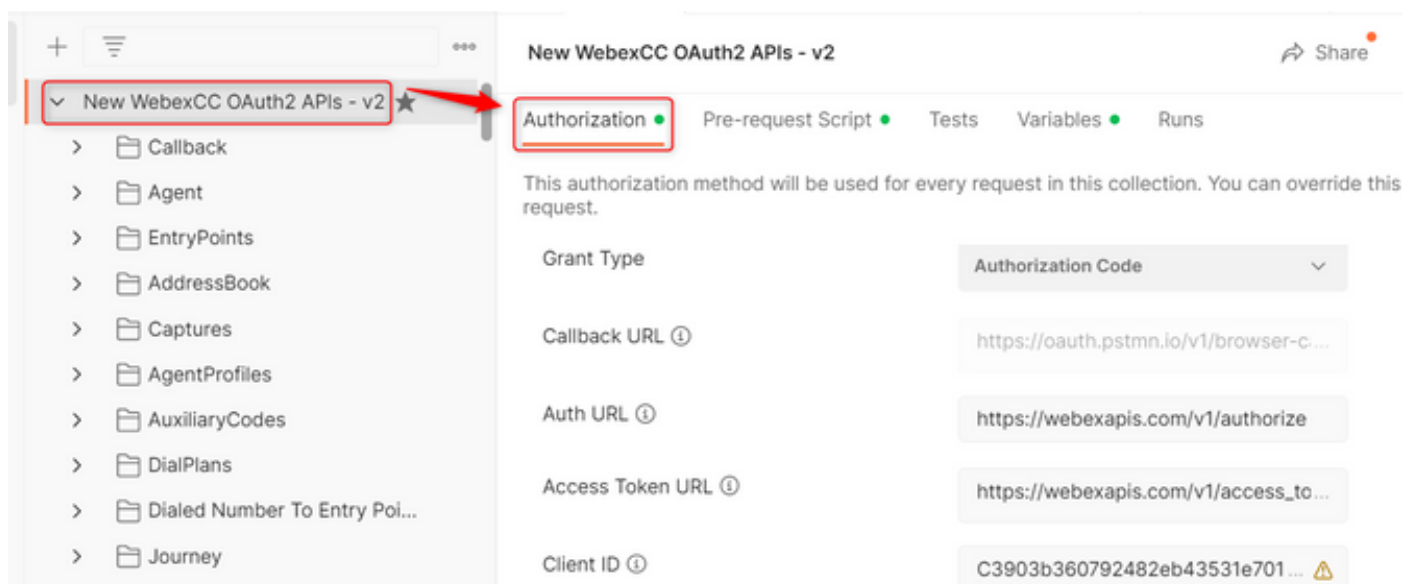
**Step 5:** Download the Post Collection from the link.

To download the collection, please navigate to [Sample Postman APIs](#).

**Step 6:** Import the JSON file into Postman.

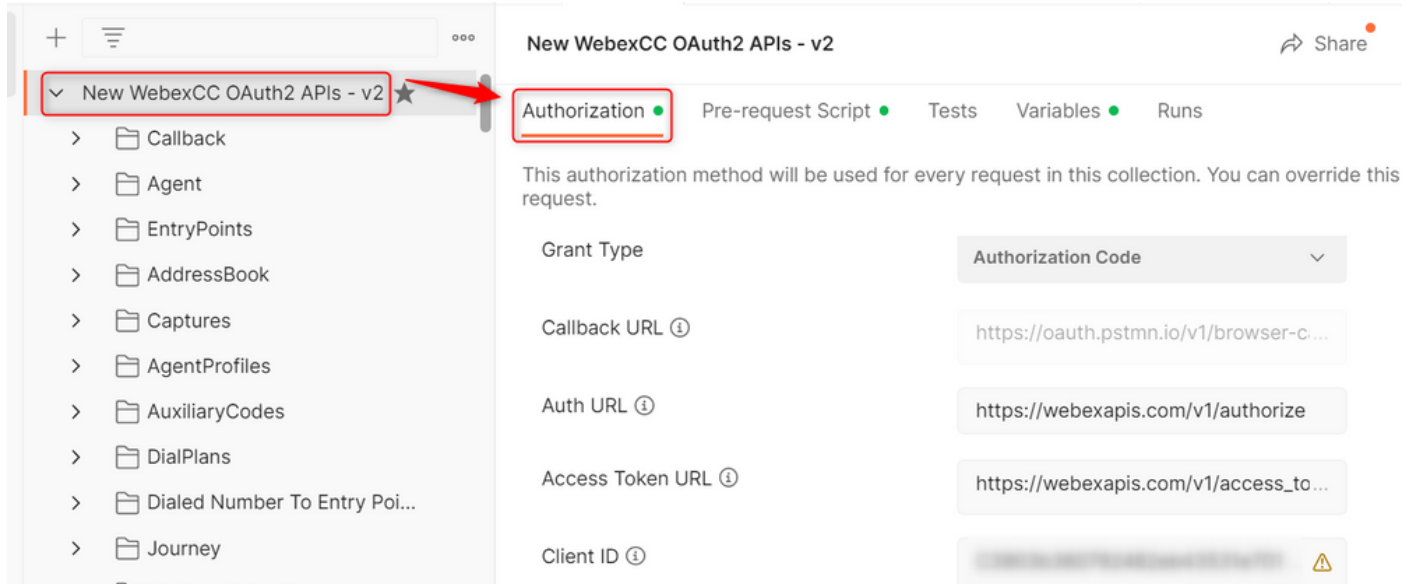


**Step 7:** Once imported, select the collection and navigate to the Authorization page.

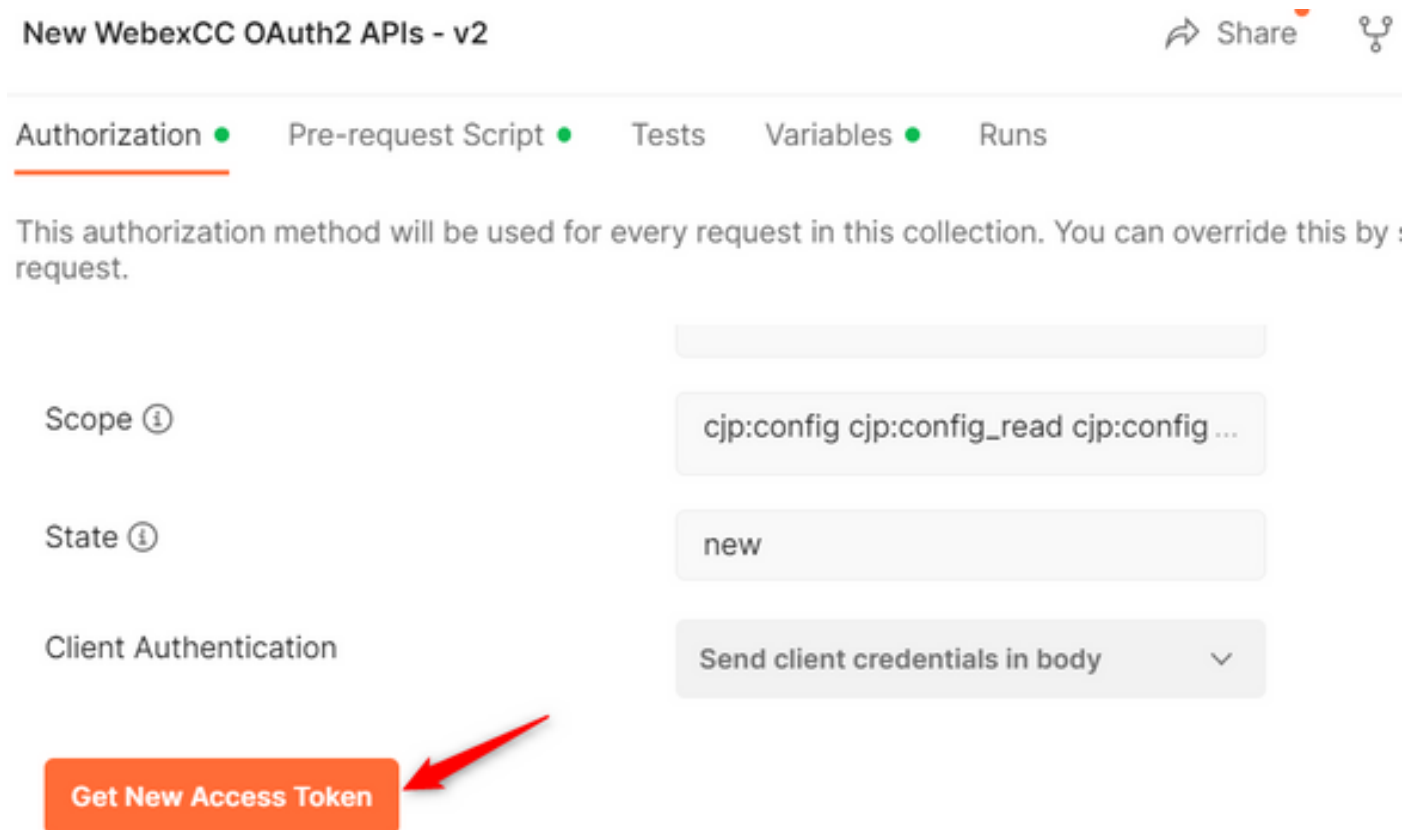


**Step 8:** Select the collection you created and navigate to the Authorization section.

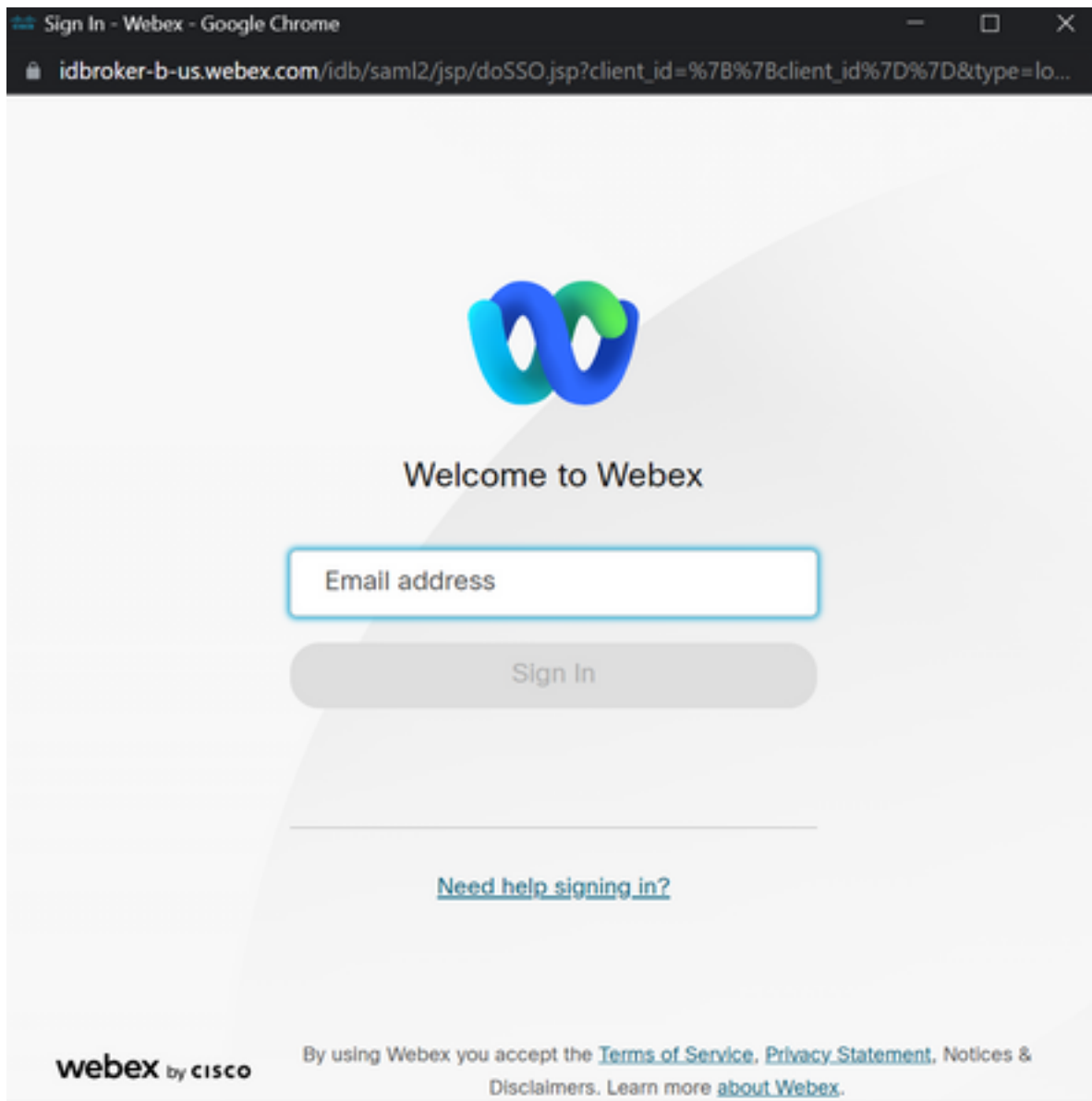
- **Auth\_URL:** <https://webexapis.com/v1/authorize>
- **Access Token URL:** [https://webexapis.com/v1/access\\_token](https://webexapis.com/v1/access_token)
- **Scope:** cjp:config cjp:config\_read cjp:config\_write



**Step 9:** Select **Get New Access Token**.

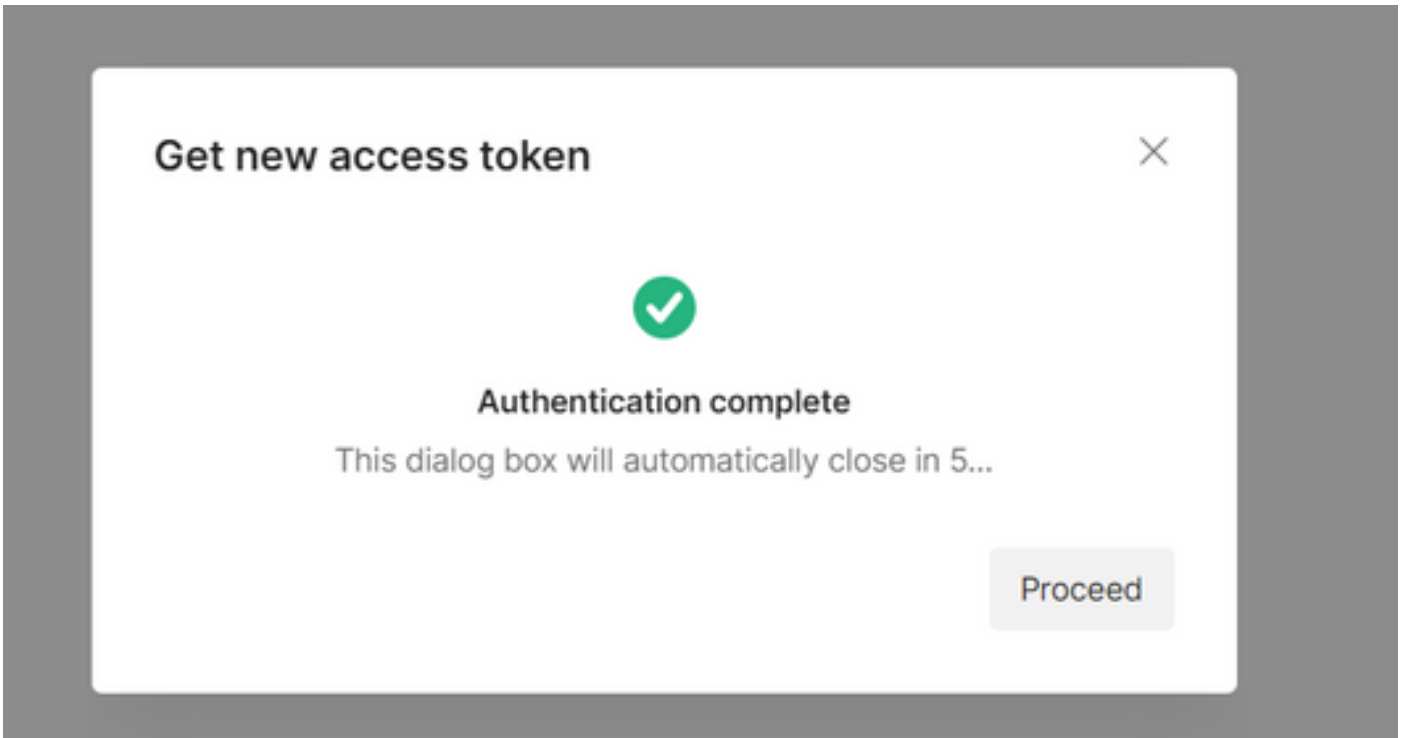


**Step 10:** Once that is finished, you get a prompt to log in to Webex. Sign in with the your administrator access to the customer tenant (the same log in credentials used to log in to the Developer portal in Step 1).

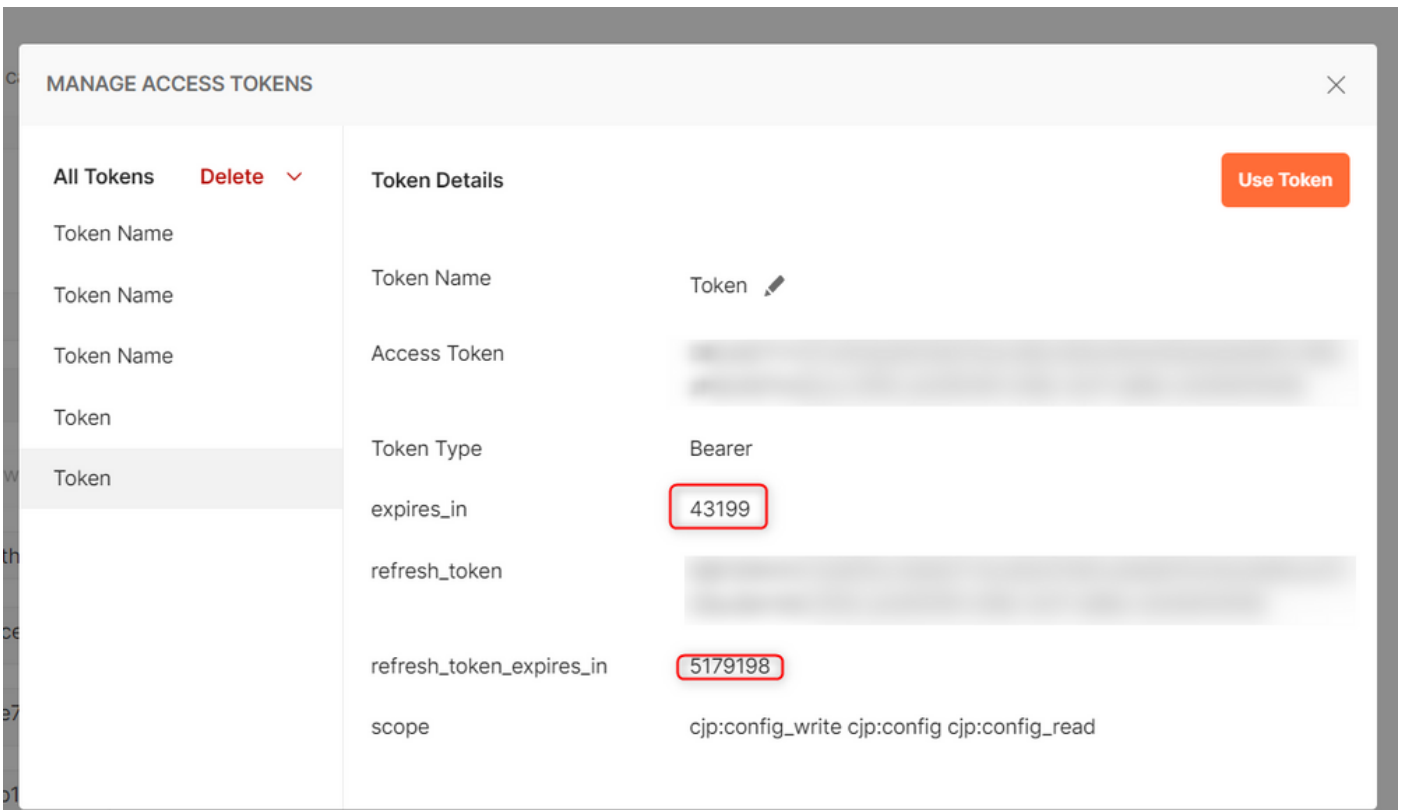


The screenshot shows a web browser window titled "Sign In - Webex - Google Chrome". The address bar contains the URL: `idbroker-b-us.webex.com/idb/saml2/jsp/doSSO.jsp?client_id=%7B%7Bclient_id%7D%7D&type=lo...`. The main content area features the Webex logo (two interlocking blue and green rings) at the top center. Below the logo is the text "Welcome to Webex". Underneath is a text input field with the placeholder "Email address". Below the input field is a grey "Sign In" button. A horizontal line separates this section from a link that says "Need help signing in?". At the bottom left is the "webex by cisco" logo. At the bottom right, there is a disclaimer: "By using Webex you accept the [Terms of Service](#), [Privacy Statement](#), Notices & Disclaimers. Learn more [about Webex](#)."

**Step 11:** Once that is finished, you see the **Success** message:



**Step 12:** After 5 seconds, the access token is generated. Select Use Token.



## Verify

Once this finished, run an API from the list imported.

- Under Auth, please make sure to select **Inherit Auth from Parent**.

- Make sure the Environment created in the previous step is selected. If the variables created were global, ignore this step.

New WebexCC OAuth2 APIs - v2 / AddressBook / All AddressBook

GET https://api.wxcc-{{datacenter}}.cisco.com/organization/{{org\_id}}/address-book/ **Send**

Params **Auth** Headers (6) Body Pre-req. Tests Settings Cookies

Type **Inherit auth from parent**

The authorization header will be automatically generated when you send the request. [Learn more about authorization](#)

This request is using OAuth 2.0 from collection [New WebexCC OAuth2 APIs - v2](#).

200 OK 69.3 ms 571 B **Save Response**

Docu No Environment TAC

Authorization OAuth 2.0

This request is using an authorization helper from collection [New WebexCC OAuth2 APIs - v2](#)

[View complete collection documentation](#)