

## Chapter 2

# Steps for Implementing Subscription Endpoints

from ADP Marketplace Integration - Separate Credentials (OAuth2) Guide

Published on  
Jul 22, 2021 4:08PM

Last modified  
Mar 30, 2023 9:53AM



## ADP Copyright Information

ADP, the ADP logo, and Always Designing for People are trademarks of ADP, Inc.

Windows is a registered trademark of the Microsoft Corporation.

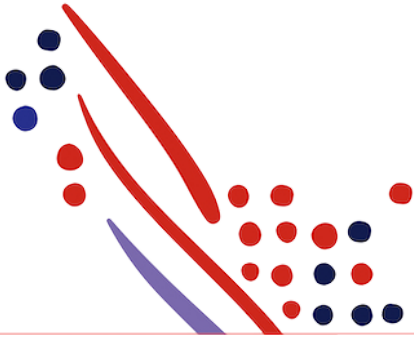
All other trademarks are the property of their respective owners.

Copyright © 2023 ADP, Inc. ADP Proprietary and Confidential - All Rights Reserved. These materials may not be reproduced in any format without the express written permission of ADP, Inc.

These materials may not be reproduced in any format without the express written permission of ADP, Inc. ADP provides this publication "as is" without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability or fitness for a particular purpose. ADP is not responsible for any technical inaccuracies or typographical errors which may be contained in this publication. Changes are periodically made to the information herein, and such changes will be incorporated in new editions of this publication. ADP may make improvements and/or changes in the product and/or the programmes described in this publication.

Published on  
Jul 22, 2021 4:08PM

Last modified  
Mar 30, 2023 9:53AM



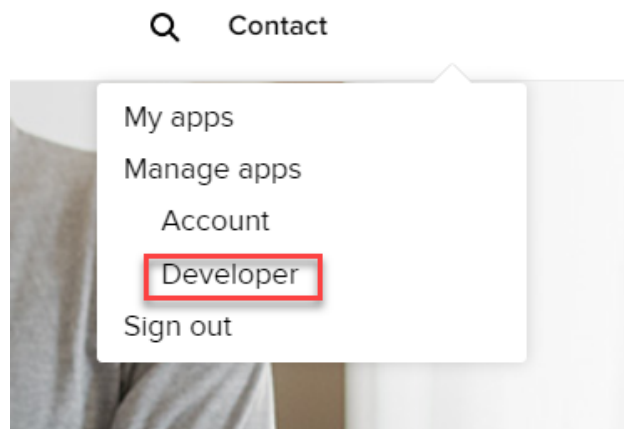
# Chapter Contents

## Chapter 2

### Steps for Implementing Subscription Endpoints

# Steps for Implementing Subscription Endpoints

1. Log into ADP Marketplace and access the developer menu under your profile.



2. If you haven't already created an app listing, create the app listing using the template provided by ADP.
3. Access the menu Products -> {your product listing} -> Edit.
4. From the menu on the left sidebar, navigate to Integration -> Edit Integration.
5. Now, navigate to the "Subscriptions" section under Edit Integration. Developers must register URL endpoints to handle notifications for subscription create, subscription change and subscription cancel. Make sure to append your URL Endpoint with "?eventUrl={eventUrl}", if you do not you will never get the event URL to call later. Sample URLs:  
Subscription create - <https://example.com/adpsubscription/create?eventUrl={eventUrl}>  
Subscription cancel - <https://example.com/adpsubscription/cancel?eventUrl={eventUrl}>  
Subscription change - <https://example.com/adpsubscription/change?eventUrl={eventUrl}>  
Subscription status - <https://example.com/adpsubscription/status?eventUrl={eventUrl}>

A screenshot of the "Edit Integration" form in the ADP Marketplace, specifically the "SUBSCRIPTIONS" section. The form contains four input fields for notification URLs, each highlighted with a red rectangular box. The first field is "Subscription Create Notification URL" with the value "https://example.com/adpsubscription/create?eventUrl={eventUrl}". Below it is a checkbox "This is an interactive endpoint" and a blue button "Add Required Info". The second field is "Subscription Change Notification URL" with the value "https://example.com/adpsubscription/change?eventUrl={eventUrl}". Below it is a checkbox "Additional product settings required during checkout". The third field is "Subscription Cancel Notification URL" with the value "https://example.com/adpsubscription/cancel?eventUrl={eventUrl}". The fourth field is "Subscription Status Notification URL" with the value "https://example.com/adpsubscription/status?eventUrl={eventUrl}". At the bottom, there is a checkbox "Enable upcoming invoice notification".



## Info

Do NOT select "This is an Interactive Endpoint", this option is not supported.

6. Navigate to the **"Credentials"** section under **"Edit Integration"**, and ensure you are selecting **"Separate Credentials"** for the Authorization type. This will configure your listing to utilize OAuth2.0 client credentials grant type flow in order for you to retrieve the subscription event payload from the eventUrl.
7. Generate a client ID and Secret in the **"Inbound Credentials"** section.
8. Under the **"Outbound Credentials"** section select **OAuth2 (client credentials)** as the type. Enter in your OAuth2 server's token endpoint along with client id, client secret, and scope that was generated for ADP Marketplace to authenticate.

**Credentials**

Authorization type

Separate Credentials (selected)

Shared Credentials (Legacy) type uses distinct OAuth 2.0 client keys and secrets to secure distribution event notifications and integration. We recommend that you create products using this model, and that you migrate existing products to this model.

**INBOUND CREDENTIALS (OAUTH 2.0)**

Use the client ID and secret to request an OAuth 2.0 access token that can be used to secure incoming API requests for your product integration (for example, to retrieve details of an event).

[Generate ID and Secret](#)

[Go To Technical Doc](#)

**OUTBOUND CREDENTIALS**

Define the credentials configuration used to secure event notifications sent to your configured distribution endpoints.

Type

OAuth2 (client credentials) (selected)

**Token endpoint**

<https://partnerapplication.com/oauth/token>

Endpoint used to request an OAuth 2.0 access token

**Client ID**

11

Client identifier used to request an OAuth 2.0 access token from the configured token endpoint

**Client secret**

\*\*\*\*\*

Client secret used to request an OAuth 2.0 access token from the configured token endpoint

**Scope (Optional)**

A space-separated list of scope values associated with the requested OAuth 2.0 access token. Leave blank if not needed.

[Save](#)

[Go To Technical Doc](#)

9. There are different subscription events, such as create, update, cancel and status notifications. However, they all follow the same mechanism in terms of how they notify your application.
10. Subscription events should be developed as synchronous events. (ADP Marketplace should receive the response immediately from your application **after you have consumed the payload in the eventUrl.**)