

Custom Auth with Token Refresh

Last Modified on 08/17/2020 2:07 am EDT

OAuth2 with grant type "client_credentials" is a perfect example of where using custom authentication with token refresh is necessary: when the endpoint is not quite OAuth2, but you still need to get an access token and refresh it. Import [Achievers 4.json](#) to see an example where this is implemented.

First, you need to configure the initial token exchange. To do this, create a resource with the **Resource Type** set to **PROVISION AUTH VALIDATION**. Do not use **ON PROVISION**, as it will not set the `authentication.time` property.

Configure parameters to save the `expires_in` and `access_token` that get returned from this call:

Name	Type	Data Type	Source	Description	Required
read	value	string	request	.	<input checked="" type="checkbox"/>
scope	form	string	< vendors Information		<input type="checkbox"/>
token	configuratic	string	response	.	<input checked="" type="checkbox"/>
access_token	bodyField	string	< vendors Information		<input type="checkbox"/>
clientsecret	configuratic	string	request	.	<input checked="" type="checkbox"/>
client_secret	form	string	< vendors Information		<input type="checkbox"/>
clientid	configuratic	string	request	.	<input checked="" type="checkbox"/>
client_id	form	string	< vendors Information		<input type="checkbox"/>
client_credentials	value	string	request	.	<input checked="" type="checkbox"/>
grant_type	form	string	< vendors Information		<input type="checkbox"/>
expires_in	configuratic	string	response	expires_in	<input type="checkbox"/>
expires_in	bodyField	string	< vendors Information		<input type="checkbox"/>

Next create a new resource with the **Resource Type** set to **ON REFRESH**. In this example, the call to refresh the token is the same as the call to get the token the first time, so this resource will be identical to the first one.

How does it work?

In this example, you need the two API calls: one with a **Resource Type** of **PROVISION AUTH VALIDATION**, and another with **ON REFRESH**. Beyond that, it is all about the configuration properties:

Name	Key	Type	Default	Description	Required	Hide UI
expires_in	expires_in	text 128	3600	expires_in	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Client Secret	clientsecret	text 128	Default Value	Client Secret	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Client ID	clientid	text 128	Default Value	Client ID	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Subdomain	subdomain	text 128	https://podmoeight.sar	The subdomain	<input checked="" type="checkbox"/>	<input type="checkbox"/>
token	token	text 128	Default Value	token	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Time of Getting Token or Perform	authentication.time	text 32	Default Value	Time of Getting Token or Performing Authentication	<input type="checkbox"/>	<input checked="" type="checkbox"/>

You need to have `authentication.time` (this should get added for you) and `expires_in`. Make sure these are spelled exactly like this example.

When you provision, it will set the `authentication.time` through the **PROVISION AUTH VALIDATION** resource. The

`expires_in` will be set by the response of that call.

The next time a call is made our system will check if (`expires_in` + `authentication.time` > now). If it is, it will call ON REFRESH, which will restart the process.
