|  |  |
| --- | --- |
| **How to use STS ?** |  |
|  |
|  |  |  |  |
| **Version 1.0** |  |  |  |

Content

[1. Register as a client of STS 2](#_Toc57276799)

[2. Get your first token via Postman 3](#_Toc57276800)

[3. Call Smoney API with your token 4](#_Toc57276801)

# Register as a client of STS

Before attempting to get a token from STS, you need to register as a client. Start by choosing a **client secret**.

For example, client secret is P@rtner0ne!! Do not share the client secret but instead compute its hash value using these 3 steps:

* + Navigate to <https://dotnetfiddle.net/xH3iA3>
	+ [Line 10] Change the input variable to match your chosen secret
	+ Click Run and retrieve your hash from the console



Finally send your **hash of your client secret** to S-Money which will add you as an STS client.

# Get your first token via Postman

Once your registration is confirmed, you will get your **Client ID** (example: partnerOne) to be used with your **Client secret** to get your first token via [Postman](https://www.postman.com/). Create a new request using the following criterias:

* **POST** <STS\_URL>**/connect/token**



* For the Authorization, select **Basic Auth** as a type and use your **Client id** and **client secret** as username and password



* For the body, select **x-www-form-urlencoded** and add the following keyvalue pairs (**grant\_type:client\_credentials**) and (**scope:partner**). Note that the keyvalue pairs are case sensitive !



* Click Send to see a response containing an **access\_token** which **expires** in 3600 seconds:



# Call Smoney API with your token

Thanks to your token, you can start calling smoney APIs in a secure way. Try it out using Postman and the following steps:

* **GET** <SMONEY\_API\_URL>**/api/v1.1/cars**



* For the Authorization, select **Bearer Token** as a type and paste in the previously obtained **access\_token**



* Click send to see a **200 OK reponse**:



You must retrieve an access token from STS before calling smoney API. The token is valid for 1 hour only so do not forget to check if the token has expired before using it. This process can be automated in your application.

Retrieving the token follows OAuth 2 protocol where the application exchanges its credentials (client id & client secret) for temporary access to the API (a Jwt token).

Although you can write your own custom code for oAuth2 protocol, we recommend using a battle tested library adapted for this scenario.

Here is a list depdending on the technical stack: <https://oauth.net/code/>