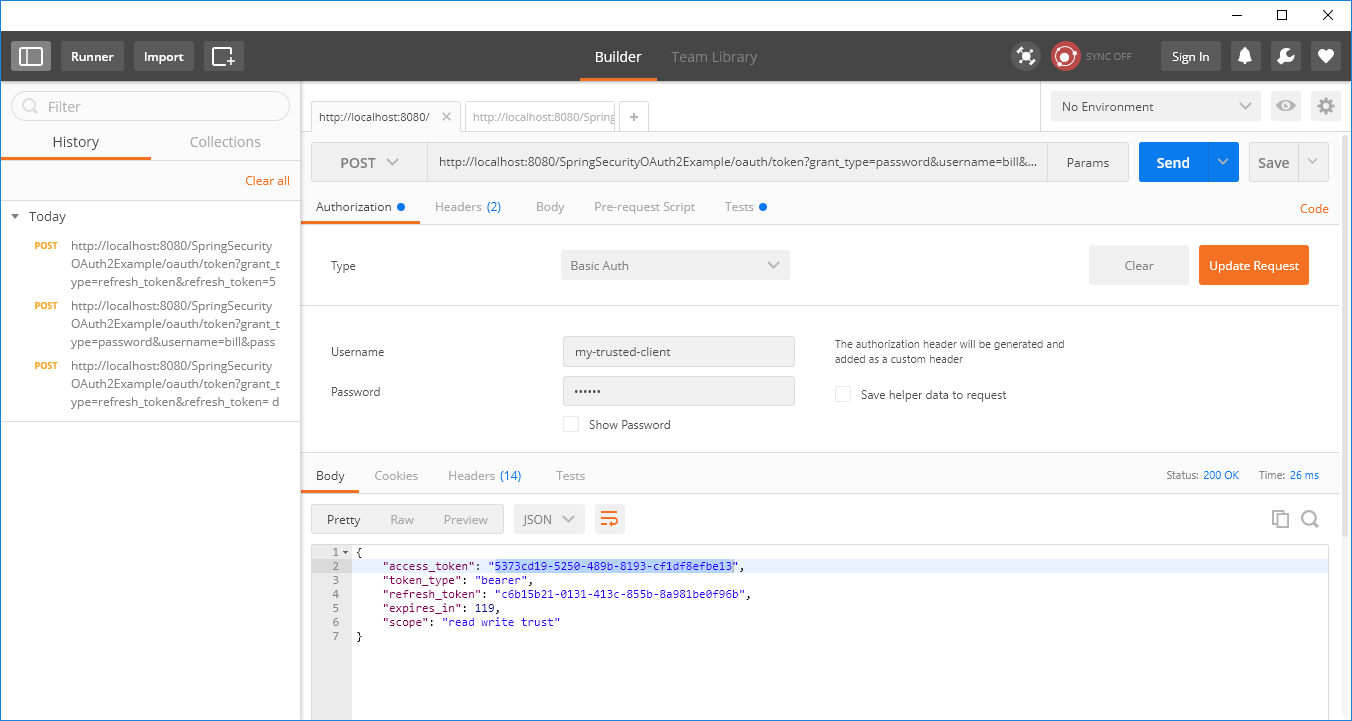
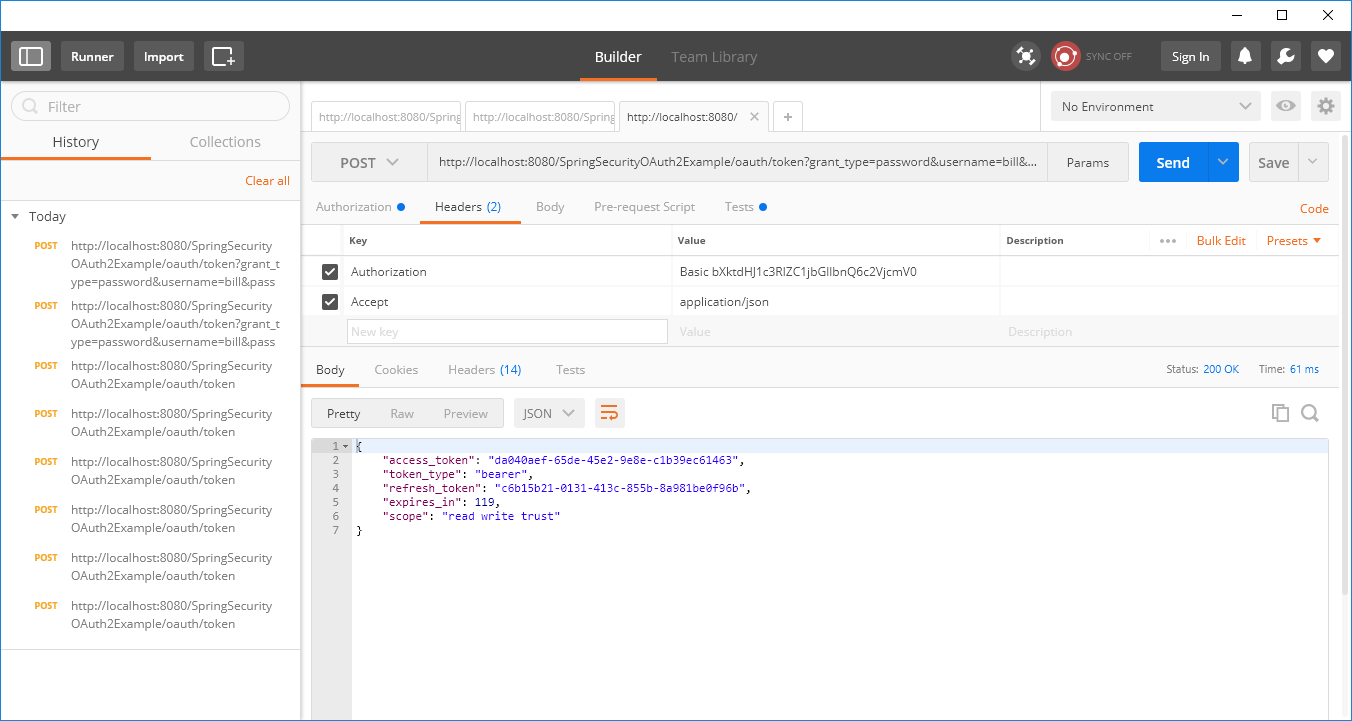
Author: <http://websystique.com/spring-security/secure-spring-rest-api-using-oauth2/>

* Attempt to access resources [REST API] without any authorization [will fail of-course].  
  GET http://localhost:8080/SpringSecurityOAuth2Example/user/
* Ask for tokens[access+refresh] using **HTTP POST** on /oauth/token, with grant\_type=password,and resource owners credentials as req-params. Additionally, send client credentials in Authorization header.

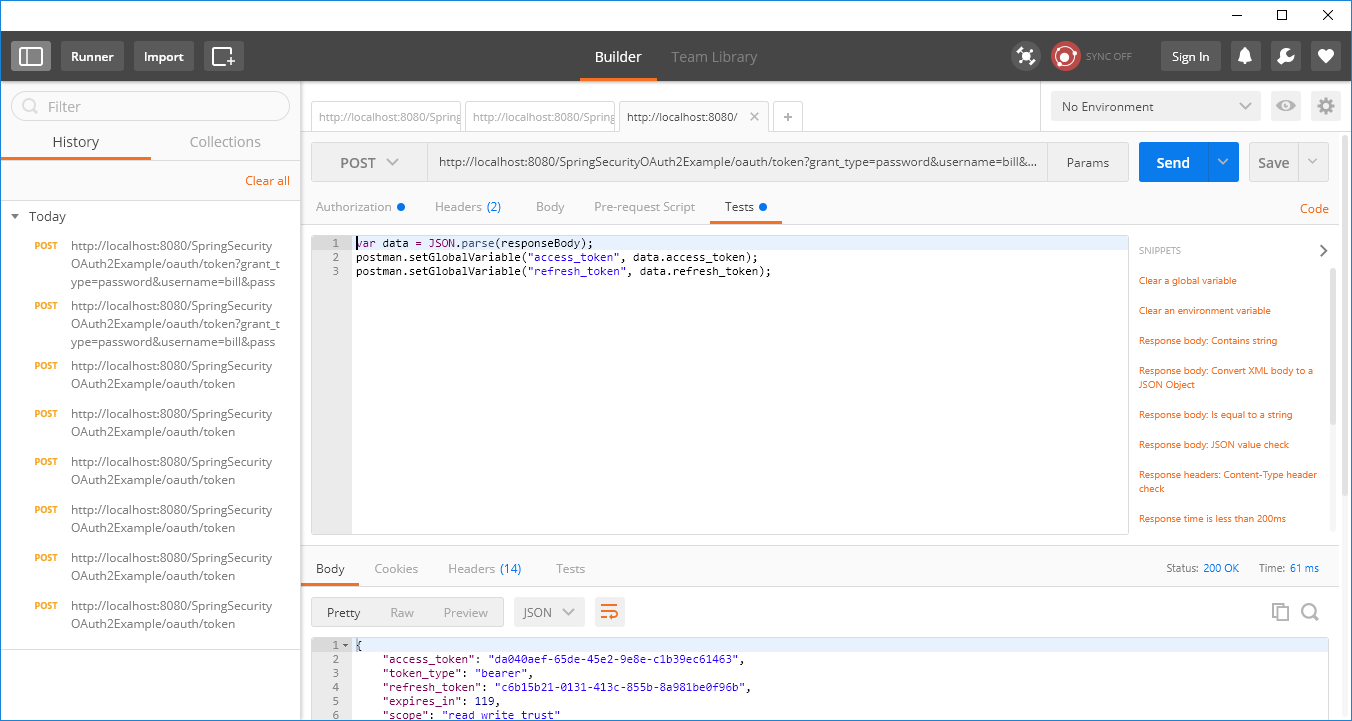
POST http://localhost:8080/SpringSecurityOAuth2Example/oauth/token?grant\_type=password&username=bill&password=abc123

my-trusted-client/secret





opcional



var data = JSON.parse(responseBody);

postman.setGlobalVariable("access\_token", data.access\_token);

postman.setGlobalVariable("refresh\_token", data.refresh\_token);

Response

{

"access\_token": "da040aef-65de-45e2-9e8e-c1b39ec61463",

"token\_type": "bearer",

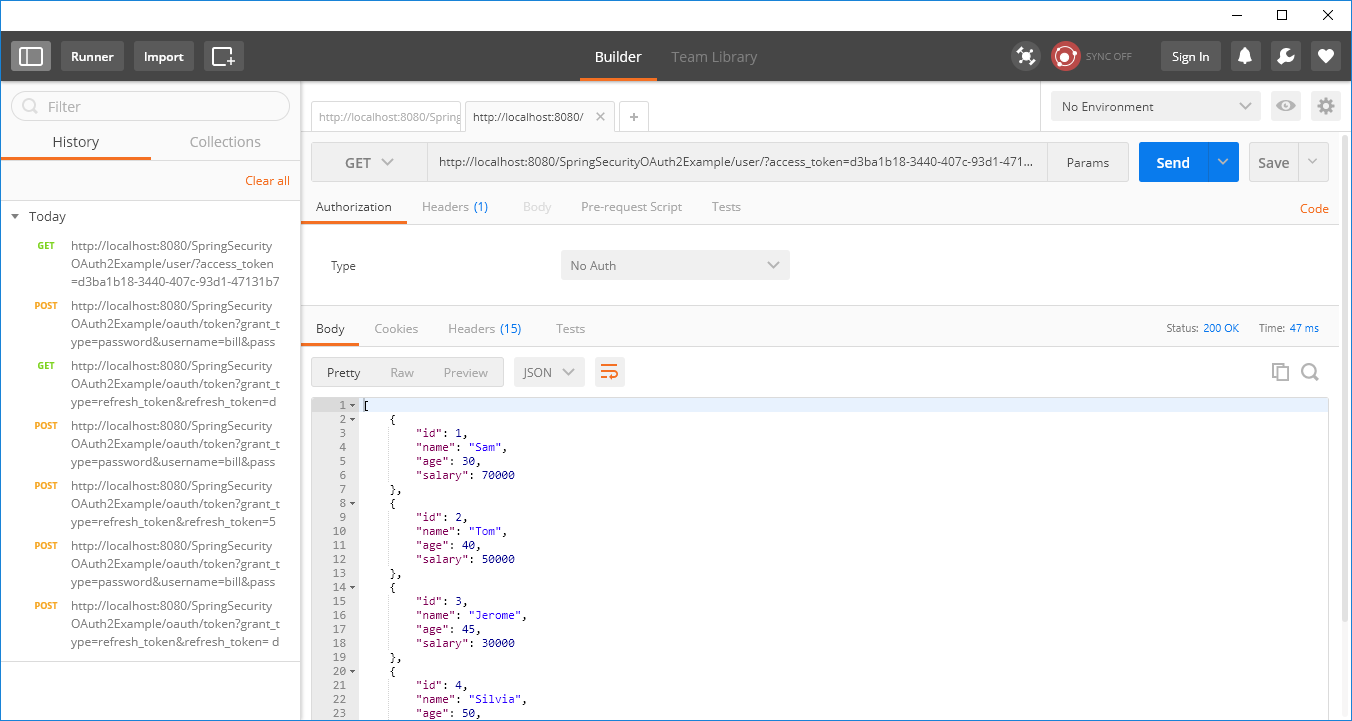
"refresh\_token": "c6b15b21-0131-413c-855b-8a981be0f96b",

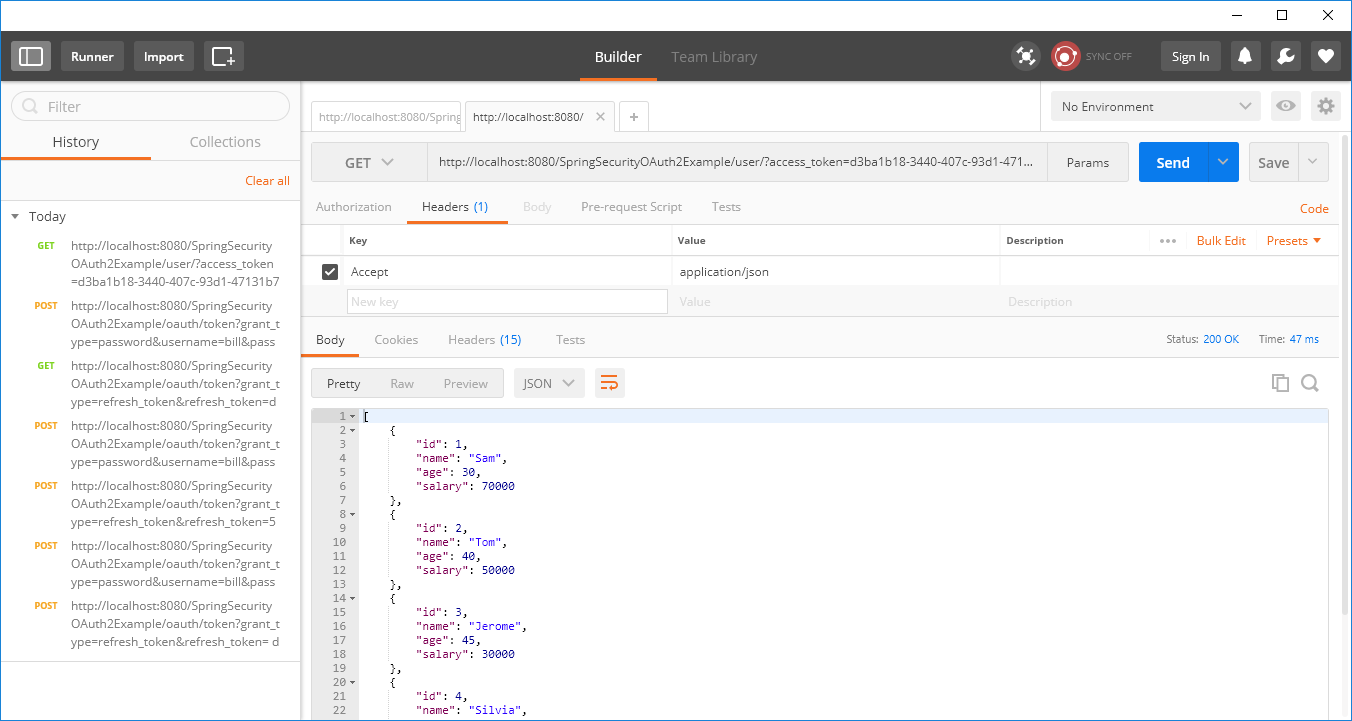
"expires\_in": 119,

"scope": "read write trust"

}

* Access the resource by providing an access token using access\_token query param with request.  
  GET http://localhost:8080/SpringSecurityOAuth2Example/user/?access\_token=3525d0e4-d881-49e7-9f91-bcfd18259109





Response

{

"id": 1,

"name": "Sam",

"age": 30,

"salary": 70000

},

{

"id": 2,

"name": "Tom",

"age": 40,

"salary": 50000

},

{

"id": 3,

"name": "Jerome",

"age": 45,

"salary": 30000

},

{

"id": 4,

"name": "Silvia",

"age": 50,

"salary": 40000

}

]

Response after 2 minutes

{

"error": "invalid\_token",

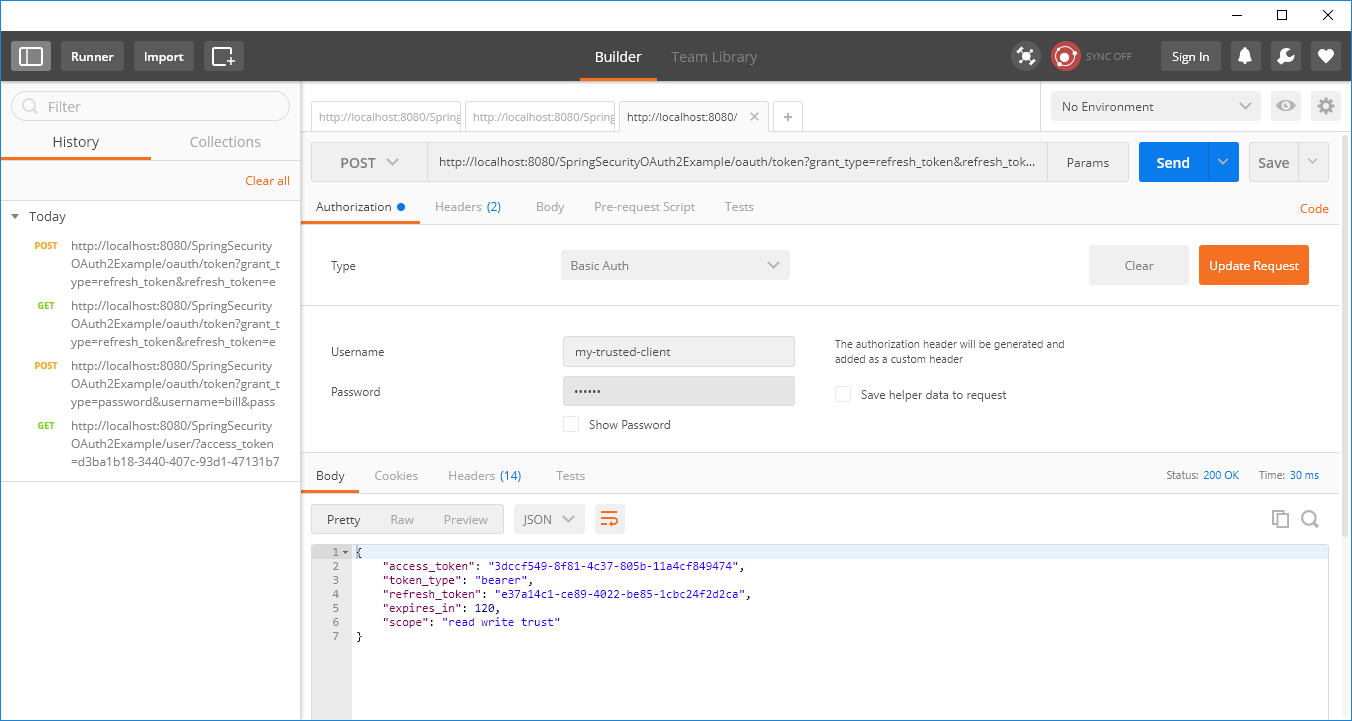
"error\_description": "Invalid access token: d3ba1b18-3440-407c-93d1-47131b74ac7e"

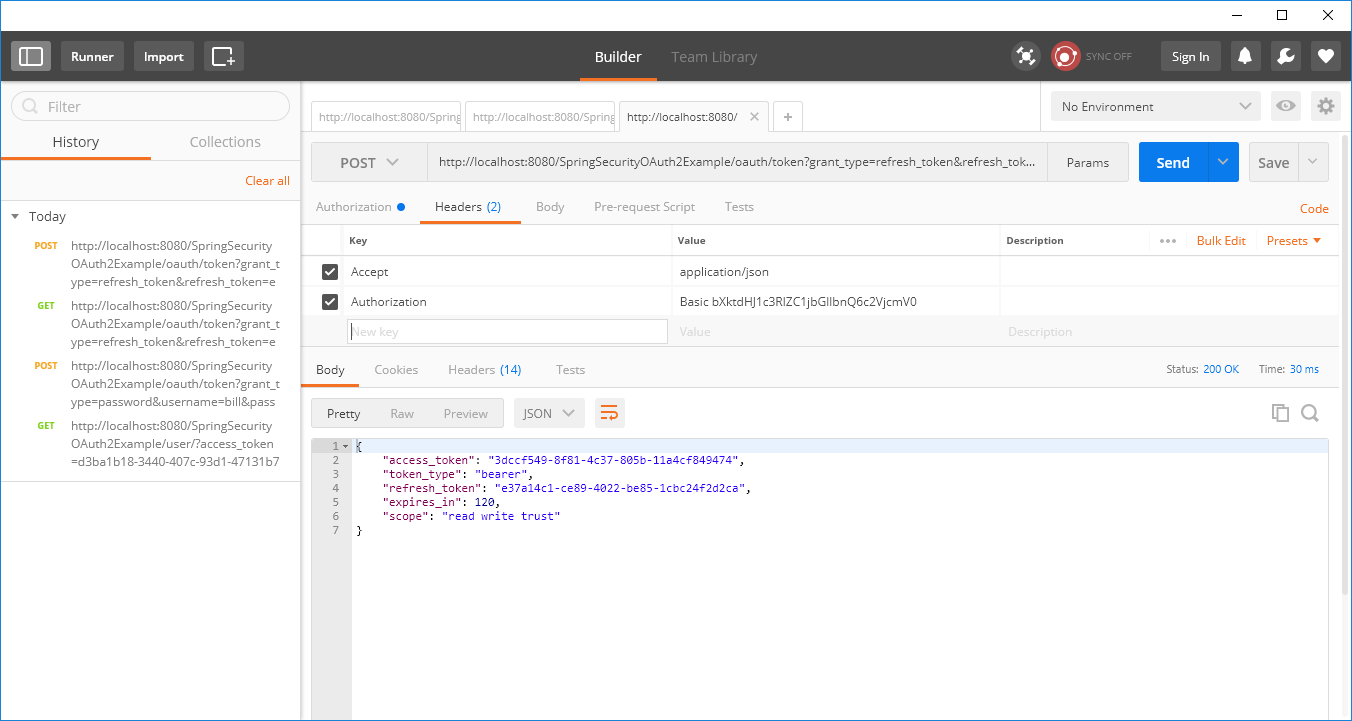
}

* Ask for a new access token via valid refresh-token, using **HTTP POST** on /oauth/token, with grant\_type=refresh\_token,and sending actual refresh token. Additionally, send client credentials in Authorization header.

POST http://localhost:8080/SpringSecurityOAuth2Example/oauth/token?grant\_type=refresh\_token&refresh\_token=094b7d23-973f-4cc1-83ad-8ffd43de1845

Se manda viejo para pedir nuevo refresh\_token





response

{

"access\_token": "3dccf549-8f81-4c37-805b-11a4cf849474",

"token\_type": "bearer",

"refresh\_token": "e37a14c1-ce89-4022-be85-1cbc24f2d2ca",

"expires\_in": 120,

"scope": "read write trust"

}