OAuth authorisation and authentication

OAuth authorisation in transmartApp

OAuth authentication for tranSMART is implemented in transmartApp. When transmartApp is running, the RESTful API is available, which is documented on transmart-rest-api on Github. For authorisation a separate route `/oauth` is available.
Use of the access token in REST requests is described under HTTP exchange details. Here we describe the process of obtaining an access token from transmartApp.

Authorisation Overview

The RESTful API supports authentication with OAuth 2.0, but each end-user first needs to authenticate with transmartApp to get access to tranSMART's resources.
In order to use the API using an OAuth access token in your client application, the following steps are needed:

1. End-users need to be redirected to the following OAuth URI to be visited in a web browser. Here they can authenticate themselves for this client application:
   `{oauthServer}/oauth/authorize?response_type=code&client_id={clientId}&client_secret={clientSecret}&redirect_uri={oauthServer}/oauth/verify`
2. After the end-user has succesfuly authenticated at this URI, a request token is supplied, which the end-user needs to copy and paste as input to your client.
3. Your client needs to exchange this request token for a semi-permanent access token, using the following HTTP request:
   `GET {oauthServer}/oauth/token?grant_type=authorization_code&client_id={clientId}&client_secret={clientSecret}&code={requestToken}&redirect_uri={oauthServer}/oauth/verify`

The response of step 3 will be JSON containing the access token, in addition to its type, a refresh token, and when the access token will expire in seconds:

```
{
    "access_token" : "12345-abcde",
    "token_type" : "bearer",
    "refresh_token" : "67890-fghij",
    "expires_in" : 99999
}
```

If the access token is expired, a new one (and a new refresh token) can be obtained using the refresh token in the following HTTP request:
`GET {oauthServer}/oauth/token?grant_type=refresh_token&client_id={clientId}&client_secret={clientSecret}&refresh_token={refreshToken}&redirect_uri={oauthServer}/oauth/verify`

Information about the the access token, refresh token, expiry, etc. (e.g., to check if an access token is still valid) can be obtained with the HTTP request:
`GET {oauthServer}/transmart/oauth/inspectToken`

For the oauth server configuration and an alternative available authentication workflow, see OAuth client application registration. For further details and other possible authentication workflows see Types of code grants to access the REST API through OAuth.

Explanation of URI variables

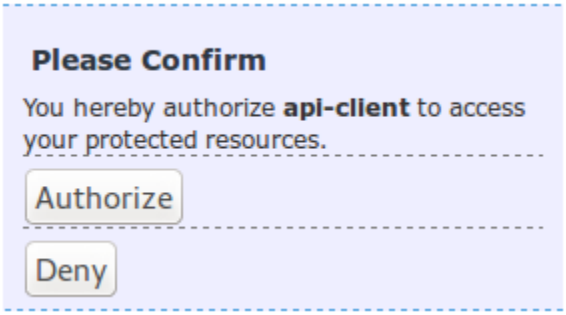
variable	explanation
`{authServer}`	The URI of the OAuth server to be used. By default this will be identical to the URI of your tranSMART server.
`{clientId}`	The client id assigned to your client application after registering it with the OAuth server. (currently: `api-client`)*
`{clientSe****cret}`	The client secret assigned to your client application after registering it with the OAuth server. (currently: `api-client`)*
`{requestToken}`	The temporary token received by your end-user after authenticating, and which needs to be exchanged by your client for an access token.

* The framework supports a workflow in which client applications are registered in tranSMART and use an application specific client id and secret. Currently, for both variables the value `api-client` can be used.

Example

Assuming a running installation of transmartApp on `http://localhost:8080/transmart`, the link for requesting a authorization code is:

http://localhost:8080/transmart/oauth/authorize?response_type=code&client_id=api-client&client_secret=api-client&redirect_uri=http://localhost:8080/transmart/oauth/verify.



Click on `Authorize` and transmartApp will generate an authorization code, e.g., `ABCxyz`.

Using `curl` we can use this to get an access token from transmartApp, to be used in requests:

```
> curl -H "Accept: application/json" "http://localhost:8080/transmart/oauth/token?
grant_type=authorization_code&client_id=api-client&client_secret=api-
client&code=ABCxyz&redirect_uri=http://localhost:8080/transmart/oauth/verify"
```

If all is right, you will get a JSON formatted reply like:

```
{"access_token":"cacwvu5l-yozb-rdsg-fnzskphppamm","token_type":"bearer","refresh_token":"x07euvad-hgix-ww4b-
b9o9irufh6bq","expires_in":25617,"scope":"write read"}
```

This means that the access token, to be used for REST calls is `cacwvu5l-yozb-rdsg-fnzskphppamm`.

Fetch information about the the access token, refresh token, expiry, etc.:

```
> curl -H "Accept: application/json" -H "Authorization: Bearer cacwvu5l-yozb-rdsg-fnzskphppamm" http://localhost:
8080/transmart/oauth/inspectToken
```

Fetch the list of studies:

```
> curl -H "Accept: application/json" -H "Authorization: Bearer cacwvu5l-yozb-rdsg-fnzskphppamm" http://localhost:
8080/transmart/studies
```

Use the refresh token to get a new access token when the current token has expired:

```
> curl -H "Accept: application/json" "http://localhost:8080/transmart/oauth/token?
grant_type=refresh_token&client_id=api-client&client_secret=api-client&refresh_token=x07euvad-hgix-ww4b-
b9o9irufh6bq&redirect_uri=http//localhost:8080/transmart/oauth/verify"
```

Note

This documentation is partially based on the transmart-rest-api documentation on Github.