Web Services: Authentication types

Authentication types and mechanisms for Matrix42 Web Services integration.

GOAL

The SASM application delivers rich REST Services API. And in many cases the existing Web Services need to be requested outside of the UUX application, when OAUTH token is not available. The System provides two alternative approaches for authenticating the Web Service Requests, using Basic Authentication or Web Service API Tokens

Basic Authentication

The approach is very simple to use but highly not recommended, as it is very vulnerable from the Security perspective, because the full login credentials (login name and password) are stored on client side, and could be compromised.

To authenticate with Basic Authentication the HTTP Request Header has to have property `Authorization` with value 'Basic username:password'. The following example demonstrates call of Workflow Resume operation using Basic Authentication

```
var xmlhttp = new XMLHttpRequest();
xmlhttp.open('GET', 'http://localhost/M42Services/api/workflow/resume?objectId=fcfd3ca2-ec21-4846-874c-d6c4f27fb97', true);
xmlhttp.setRequestHeader('Authorization', 'Basic ' + 'XNtXWTZESvdmlldjpnZmhqS20hMQ=='/btoa("username:password")/);
```

Use API Token

The more secure and recommended way to authenticate Web Service call is using API Tokens. The example uses Javascript to demonstrate the flow of Tokens, but absolutely the same ways it can be done with any other programming language

Step 1: Generate API Token

Issue API token in UUX and store it on client. See more details on Web Service API Token.

Step 2: Exchange API token to Access Token

API Token is used only for getting Access Token, which is used for authenticating each Web Service call.

```
var xmlhttp = new XMLHttpRequest();
xmlhttp.open('POST', 'http://{serverurl}/M42Services/api/ApiToken/GenerateAccessTokenFromApiToken/', true);
```

https://help.matrix42.com/030_DWP/030_INT/Business_Processes_and_API_Integrations/Web_Services%3A_Authentication…

Updated: Sat, 18 Jul 2020 00:14:35 GMT

Copyright © 2018. Matrix42 is a registered trademark of Matrix42 AG. All other brands and product names are trademarks or r…
xmlhttp.setRequestHeader("Content-Type", "application/json");
xmlhttp.setRequestHeader('Authorization', 'Bearer ' + 'eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ1bmlxdWVfbmFtZSI6ImltYWdlLWNsZXRlcyJdLCJ1c2VyX2F0ZiI6IjIiLCJleHAiOjE2MDk1NzAwMzB9.KPRlkVooufH0sFVIIHGa38m3kYSNTXrOJdiFB8VSQNM');

Step 3: Use Access Token for Web Service Call

```
xmlhttp.onreadystatechange = function() {
  if (this.readyState == 4 && this.status == 200) {
    if(this.responseText!==null){
      var svchttp = new XMLHttpRequest();
      svchttp.open('GET', 'http://{serverurl}/M42Services/api/workflow/resume?objectId=fcfd3ca2-ec21-4846-874c-d6c4f27ff97', true);
      svchttp.setRequestHeader('Authorization', 'Bearer ' + JSON.parse(this.responseText).RawToken);
    }
    else {
      console.log('Not valid or disabled API token')
    }
};
xmlhttp.send();
```

EXAMPLES

Call Service: Start Workflow

To prepare the web service call to any service, first of all, figure out the contract of the service, which input parameters of the Web Service and how they passed in REST request, as well how the Web Service return data.

You can find this information either in UUX Administration application, management area "Web Services", or find where in standart UUX application this service is activated, and check the contract in the browser Development Tools Network tabulator
The example starts Approval Workflow, which accepts the single input parameter "bookings"

```javascript
var svchttp = new XMLHttpRequest();
svchttp.open('POST', 'http://{serverurl}/M42Services/api/workflow/start/{workflowId}', true);
svchttp.setRequestHeader('Authorization', 'Bearer ' + {token});
svchttp.send('{bookings: [id1, id2]}');
```

Where:

* `workflowId` - object ID of the Workflow (can be found in table PLSLXmlComponentType.ID);
* `token` - authentication token (see above how to obtain it);