

In this assignment, I will cover the implementation of Authentication-Service. This service will allow two major features:

1. Authenticate users via token mechanism, which will later allow to:
2. Authorization access with specific roles to each resource after authenticating.

The implementation of the service along with additional modification at existing services which are involved as resources, such as Organization Service and Asset Service.

Running mvn clean package docker:build :

```
[INFO] -----
[INFO] Reactor Summary:
[INFO]
[INFO] Zuul Proxy Server ..... SUCCESS [ 10.250 s]
[INFO] Config Server ..... SUCCESS [ 4.732 s]
[INFO] Asset Server ..... SUCCESS [ 7.262 s]
[INFO] Organization Service ..... SUCCESS [ 6.508 s]
[INFO] Eureka Server ..... SUCCESS [ 5.694 s]
[INFO] New Version of Organization Service ..... SUCCESS [ 6.441 s]
[INFO] Special Routes Service ..... SUCCESS [ 6.470 s]
[INFO] Authentication Service ..... SUCCESS [ 8.899 s]
[INFO] assignment3_1-parent-pom 0.0.1-SNAPSHOT ..... SUCCESS [ 0.197 s]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 57.165 s
[INFO] Finished at: 2019-05-14T23:40:19+03:00
[INFO] -----
→ Assignment3_1
```

docker-compose -f docker/common/docker-compose.yml up

```
organizationservice_1 | 2019-05-14 20:42:42.581 INFO 22 --- [          main] ration$HystrixMetricsPollerConfiguration : Starting poller
eurekaserver_1        | 2019-05-14 20:42:42.627 INFO 7 --- [nio-8761-exec-8] c.n.e.registry.AbstractInstanceRegistry : Registered instance ORG
also)
organizationservice_1 | 2019-05-14 20:42:42.640 INFO 22 --- [infoReplicator-0] com.netflix.discovery.DiscoveryClient : DiscoveryClient_ORGANI
organizationservice_1 | 2019-05-14 20:42:42.807 INFO 22 --- [          main] s.b.c.e.t.TomcatEmbeddedServletContainer : Tomcat started on port
organizationservice_1 | 2019-05-14 20:42:42.810 INFO 22 --- [          main] c.n.e.EurekaDiscoveryClientConfiguration : Updating port to 8085
organizationservice_1 | 2019-05-14 20:42:42.820 INFO 22 --- [          main] r.t.organization.Application : Started Application in
eurekaserver_1        | 2019-05-14 20:42:43.136 INFO 7 --- [io-8761-exec-10] c.n.e.registry.AbstractInstanceRegistry : Registered instance ORG
rue)
assetsservice_1       | 2019-05-14 20:42:43.222 INFO 22 --- [          main] c.n.e.EurekaDiscoveryClientConfiguration : Registering applicatio
eurekaserver_1        | 2019-05-14 20:42:43.339 INFO 7 --- [nio-8761-exec-1] c.n.e.registry.AbstractInstanceRegistry : Registered instance ASSI
assetsservice_1       | 2019-05-14 20:42:43.340 INFO 22 --- [          main] o.s.c.support.DefaultLifecycleProcessor : Starting beans in phase
assetsservice_1       | 2019-05-14 20:42:43.357 INFO 22 --- [          main] ration$HystrixMetricsPollerConfiguration : Starting poller
assetsservice_1       | 2019-05-14 20:42:43.543 INFO 22 --- [          main] s.b.c.e.t.TomcatEmbeddedServletContainer : Tomcat started on port
assetsservice_1       | 2019-05-14 20:42:43.547 INFO 22 --- [          main] c.n.e.EurekaDiscoveryClientConfiguration : Updating port to 8080
assetsservice_1       | 2019-05-14 20:42:43.556 INFO 22 --- [          main] r.t.a.AssetsServiceApplication : Started AssetsServiceAp
eurekaserver_1        | 2019-05-14 20:42:43.858 INFO 7 --- [nio-8761-exec-2] c.n.e.registry.AbstractInstanceRegistry : Registered instance ASSI
eurekaserver_1        | 2019-05-14 20:43:08.561 INFO 7 --- [a-EvictionTimer] c.n.e.registry.AbstractInstanceRegistry : Running the evict task
```

Use Case 1 – Access is restricted only to authenticated users:

Attempting to get organization details without authenticating.

The screenshot shows a REST client interface with the following details:

- URL:** `http://localhost:8085/api/organization/v1/organizations/org-1`
- Method:** GET
- Params:** A table with columns KEY, VALUE, and DESCRIPTION. It contains one row with 'Key' and 'Value'.
- Body:** The response body is displayed in JSON format, showing an error:

```
1 {  
2   "error": "unauthorized",  
3   "error_description": "Full authentication is required to access this resource"  
4 }
```

The status bar at the bottom indicates: Status: 401 Unauthorized, Time: 201 ms.

Attempting to get token.

Introducing username, password, grant type and scope to the body.

http://localhost:8901/auth/oauth/token

POST http://localhost:8901/auth/oauth/token

Params Authorization Headers (2) Body Pre-request Script Tests

none form-data x-www-form-urlencoded raw binary

	KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/>	username	david.r	
<input checked="" type="checkbox"/>	password	dr123	
<input checked="" type="checkbox"/>	scope	webclient	
<input checked="" type="checkbox"/>	grant_type	password	
<input checked="" type="checkbox"/>	override	true	
	Key	Value	Description

Body Cookies Headers (9) Test Results Status: 200 OK Time: 428 ms

Pretty Raw Preview JSON

```
1 {
2   "access_token": "ed963444-5ef9-402f-a745-5e8572987b86",
3   "token_type": "bearer",
4   "refresh_token": "be572217-bf62-4ca9-a9fa-4753bb563161",
5   "expires_in": 43199,
6   "scope": "webclient"
7 }
```

Setting authentication for getting token using username and password basic authentication:

The screenshot shows a REST client interface with the following components:

- URL Bar:** `http://localhost:8901/auth/oauth/token`
- Method and URL:** POST `http://localhost:8901/auth/oauth/token`
- Tabs:** Params, Authorization (selected), Headers (2), Body, Pre-request Script, Tests
- Authorization Section:**
 - TYPE:** Basic Auth
 - Description:** The authorization header will be automatically generated when you send the request. [Learn more about authorization](#)
 - Preview Request:** A button to preview the request.
 - Username:** eagleeye
 - Password:** thisissecret
 - Show Password:** ☒
- Response Section:**
 - Tabs:** Body (selected), Cookies, Headers (9), Test Results
 - Status:** 200 OK
 - Time:** 428 ms
 - Format:** JSON
 - Response Body:**

```
1 {
2   "access_token": "ed963444-5ef9-402f-a745-5e8572987b86",
3   "token_type": "bearer",
4   "refresh_token": "be572217-bf62-4ca9-a9fa-4753bb563161",
5   "expires_in": 43199,
6   "scope": "webclient"
7 }
```

Introducing a script to refresh token :

The screenshot shows a Postman interface for a POST request to `http://localhost:8901/auth/oauth/token`. The 'Tests' tab is active, displaying a JavaScript script:

```
1 var data = JSON.parse(responseBody);
2 postman.setGlobalVariable("access_token", data.access_token);
3 postman.setGlobalVariable("refresh_token", data.refresh_token);
4
```

Below the script, the 'Body' tab is selected, showing a JSON response:

```
1 {
2   "access_token": "ed963444-5ef9-402f-a745-5e8572987b86",
3   "token_type": "bearer",
4   "refresh_token": "be572217-bf62-4ca9-a9fa-4753bb563161",
5   "expires_in": 3600
6 }
```

The status bar indicates a successful response with `Status: 200 OK` and `Time: 428 ms`.

Getting organization with bearer token.

The screenshot shows a Postman interface for a GET request to `http://localhost:8901/auth/user`. The 'Headers' tab is active, displaying the following headers:

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> Content-Type	application/json	
<input checked="" type="checkbox"/> Authorization	Bearer ed963444-5ef9-402f-a745-5e8572987b86	
<input checked="" type="checkbox"/> Accept	application/json	
Key	Value	Description

Below the headers, the 'Body' tab is selected, showing a JSON response:

```
1 {
2   "user": {
3     "password": null,
4     "username": "david.r",
5     "authorities": [
6       {
7         "authority": "ROLE_USER"
8       }
9     ],
10    "accountNonExpired": true,
11    "accountNonLocked": true,
12    "credentialsNonExpired": true,
13    "enabled": true
14  },
15  "authorities": [
16    "ROLE_USER"
17  ]
18 }
```

The status bar indicates a successful response with `Status: 200 OK` and `Time: 2782 ms`.

Use Case 2: Protect a service via a Specific Role.

User Ester.N has both user and admin roles.

POST

http://localhost:8901/auth/oauth/token

Params

Authorization

Headers (2)

Body

Pre-request Script

Tests

none

form-data

x-www-form-urlencoded

raw

binary

	KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/>	username	ester.n	
<input checked="" type="checkbox"/>	password	en123	
<input checked="" type="checkbox"/>	scope	webclient	
<input checked="" type="checkbox"/>	grant_type	password	
<input checked="" type="checkbox"/>	override	true	
	Key	Value	Description

body

Cookies

Headers (9)

Test Results

Status: 200 OK

Time: 45

Pretty

Raw

Preview

JSON

1

{

2

"access_token": "25efad7e-05a4-4cc0-9a8c-080feaa09038",

3

"token_type": "bearer",

4

"refresh_token": "1482a953-ed87-4433-90a4-7144fbb6b8b2",

5

"expires_in": 43199,

6

"scope": "webclient"


7

}

Calling auth/user :

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> Content-Type	application/json	
<input checked="" type="checkbox"/> Authorization	Bearer 25efad7e-05a4-4cc0-9a8c-080feaa09038	
<input checked="" type="checkbox"/> Accept	application/json	
Key	Value	Description

body Cookies Headers (10) Test Results Status: 200 OK Time: 23

Pretty Raw Preview JSON 

```
1 {
2   "user": {
3     "password": null,
4     "username": "ester.n",
5     "authorities": [
6       {
7         "authority": "ROLE_ADMIN"
8       },
9       {
10        "authority": "ROLE_USER"
11      }
12    ],
13    "accountNonExpired": true,
14    "accountNonLocked": true,
15    "credentialsNonExpired": true,
16    "enabled": true
17  },
18  "authorities": [
19    "ROLE_ADMIN",
20    "ROLE_USER"
21  ]
}
```