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:

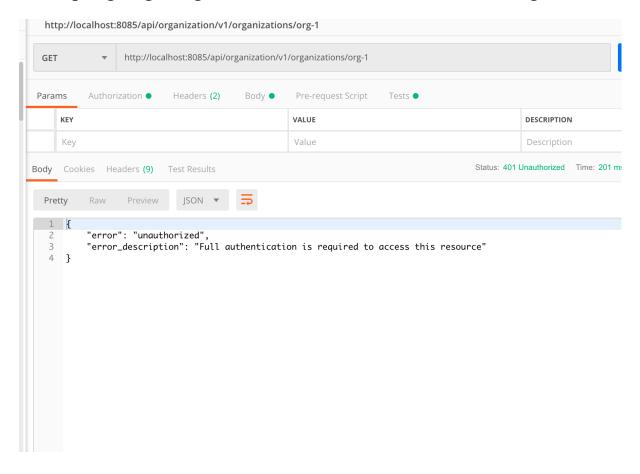
```
[INFO] Reactor Summary:
[INFO]
[INFO] Organization Service ...... SUCCESS [ 6.508 s]
[INFO] New Version of Organization Service ...... SUCCESS [ 6.441 s]
[INFO] assignment3_1-parent-pom 0.0.1-SNAPSHOT ...... SUCCESS [ 0.197 s]
[INFO] --
[INFO] BUILD SUCCESS
[INFO] --
[INFO] Total time: 57.165 s
[INFO] Finished at: 2019-05-14T23:40:19+03:00
→ Assignment3_1
```

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

```
organizationservice 1
                                                        2019-05-14 20:42:42.581 INFO 22 --- [
                                                                                                                                                                main| ration$HvstrixMetricsPollerConfiguration : Starting poller
                                                         | 2019-05-14 20:42:42.627 INFO 7 --- [nio-8761-exec-8] c.n.e.registry.AbstractInstanceRegistry : Registered instance ORG/
     eurekaserver_1
     alse)
                                                         | 2019-05-14 20:42:42.640 INFO 22 --- [nfoReplicator-0] com.netflix.discovery.DiscoveryClient
     organizationservice_1
                                                                                                                                                                                                                                                                  : DiscoveryClient_ORGANI;
                                                         | 2019-05-14 20:42:42.807 INFO 22 --- [ main] s.b.c.e.t.TomcatEmbeddedServletContainer : Tomcat started on port
      organizationservice_1
                                                                                                                                                                  main] c.n.e.EurekaDiscoveryClientConfiguration : Updating port to 8085
      organizationservice_1
                                                         2019-05-14 20:42:42.810 INFO 22 --- [
                                                     2019-05-14 20:42:42.820 INFO 22 --- [
                                                                                                                                                              main] r.t.organization.Application
      organizationservice_1
      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
     eurekaserver 1
     assetsservice 1
     assetsservice 1
                                                     | 2019-05-14 20:42:43.543 | INFO 22 --- [ main] s.b.c.e.t.TomcatEmbeddedServletContainer : Tomcat started on port | 2019-05-14 20:42:43.547 | INFO 22 --- [ main] c.n.e.EurekaDiscoveryClientConfiguration : Updating port to 8080 | 2019-05-14 20:42:43.556 | INFO 22 --- [ main] r.t.a.AssetsServiceApplication : Started AssetsServiceApplication : Started AssetsServiceAp
      assetsservice_1
assetsservice_1
      assetsservice_1
     eurekaserver_1
                                                         | 2019-05-14 20:42:43.858 INFO 7 --- [nio-8761-exec-2] c.n.e.registry.AbstractInstanceRegistry : Registered instance ASSi
                                                     | 2019-05-14 20:43:08.561 INFO 7 --- [a-EvictionTimer] c.n.e.registry.AbstractInstanceRegistry : Running the evict task v
     eurekaserver_1
```

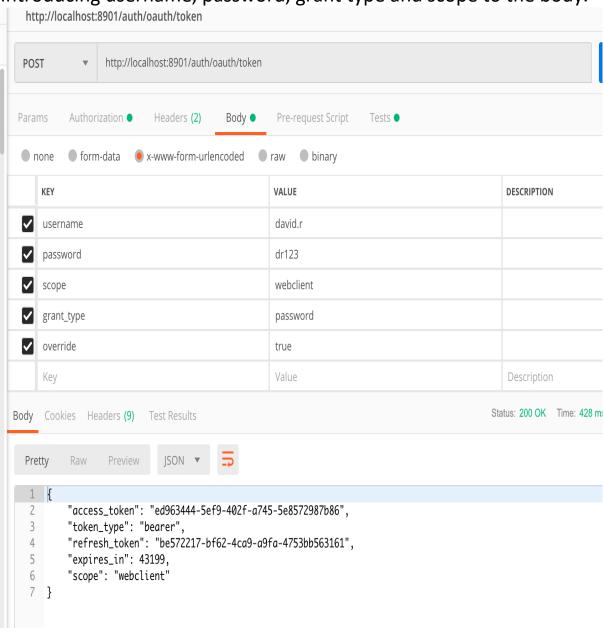
Use Case 1 – Access is restricted only to authenticated users:

Attempting to get organization details without authenticating.

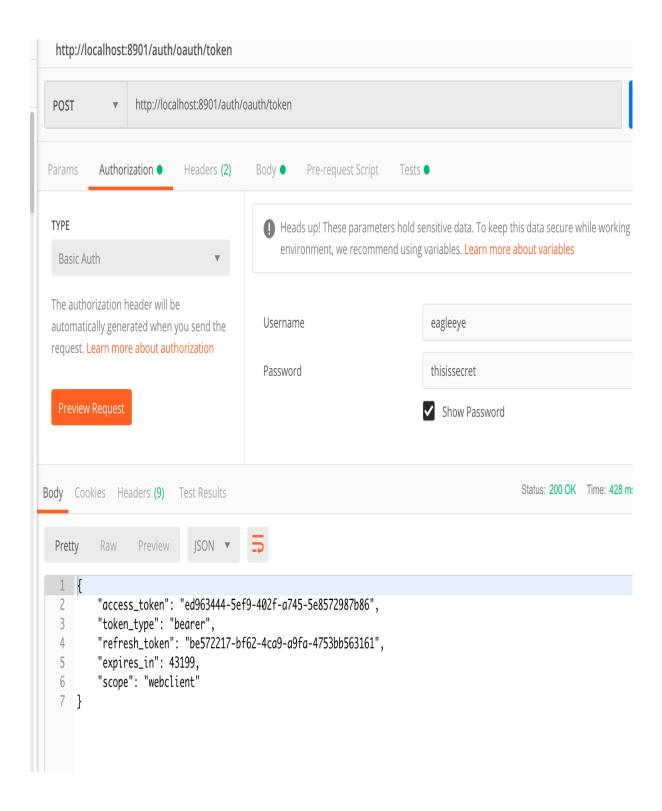


Attempting to get token.

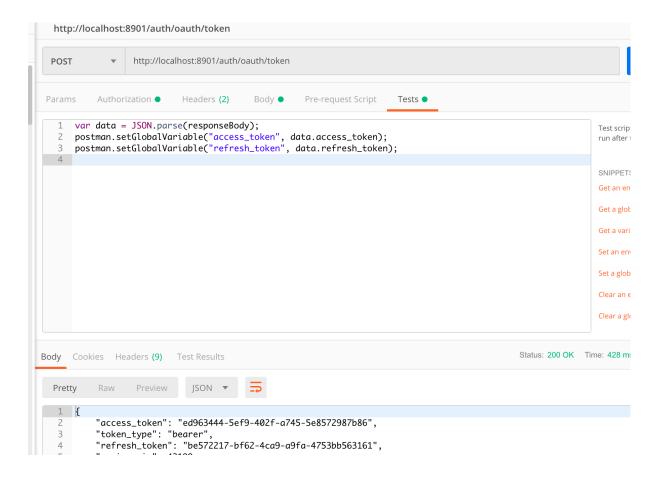
Introducing username, password, grant type and scope to the body.



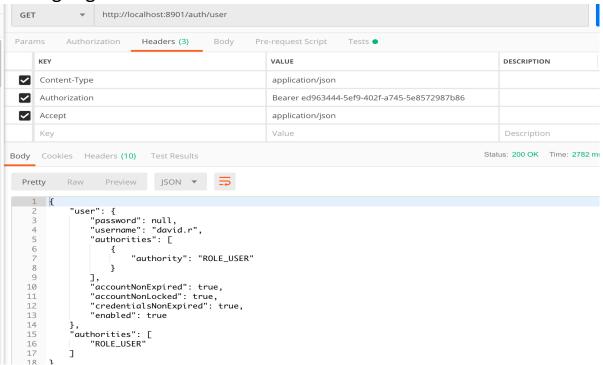
Setting authentication for getting token using username and password basic authentication:



Introducing a script to referesh token:

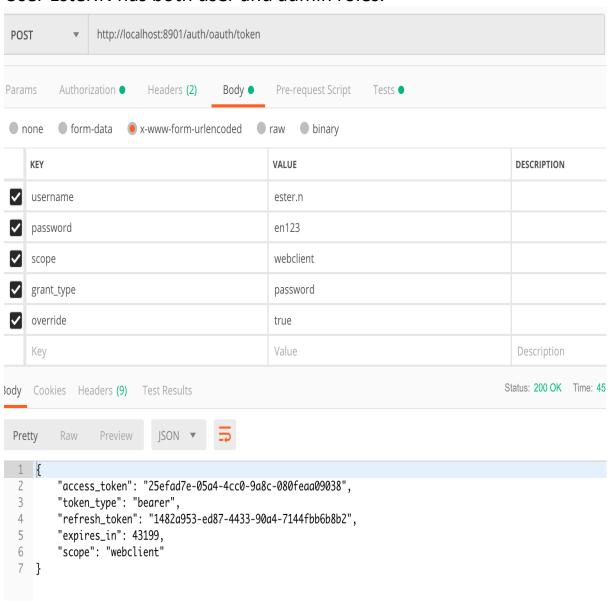


Getting organization with bearer token.



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

User Ester.N has both user and admin roles.



Calling auth/user:

