Especialización en Back End II

Ejemplo

¿Cómo valido la seguridad provista por Keycloak? Vamos a ver cómo crear un endpoint en un controlador (¡aprovechemos a practicar la creación base de una API REST!) para validar el uso de nuestro IAM.

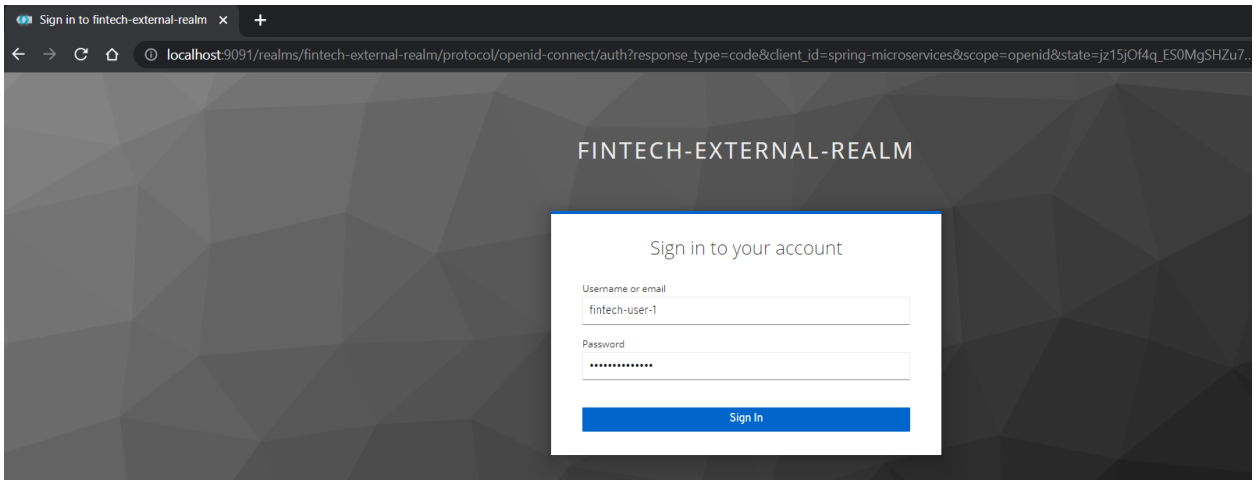
```
@GetMapping(path ="/detail/{id}")
public Map<String, Object>
detailSecure(@RegisteredOAuth2AuthorizedClient
OAuth2AuthorizedClient authorizedClient
   , Authentication auth
   , @PathVariable("id") Long idAccount) {
   Map<String, Object> response = new HashMap<>();
   response.put("clientName",
authorizedClient.getClientRegistration().getClientName());
   response.put("id_account", idAccount);
   response.put("token",
authorizedClient.getAccessToken().getTokenValue());

   return response;
}
```

El endoint, que será de tipo GET, posee dos parámetros (no es necesario que ahondemos en ellos ahora, pero sí es importante que sepamos que provienen de la librería propia de OAuth 2.0 y sirven para autorizar al client creado en el IAM que se estará comunicando con nuestro endpoint) y un path variable, el cual corresponde al ID del cliente.

En el bloque de código, definimos un map que recibirá tres objetos:

- **clientName**: donde guardaremos el nombre del cliente autorizado.
- **id_account**: donde almacenaremos el ID que viene por path variable.
- **token**: donde mantendremos el token que proviene del cliente.

A través del navegador, vamos a intentar acceder al endpoint que definimos previamente: http://localhost:8080/account/detail/1. La aplicación nos pedirá la autenticación en Keycloak:



Luego de autenticarnos, se ejecutará el endpoint inicial, el cual nos devolverá toda la información en formato JSON.