Especialización en Back End II

Ejercicio en vivo

- Ejercitación grupal
- Nivel de complejidad: medio 🔥🔥

Objetivo

Crear una configuración de un realm, cliente y usuarios en Keycloak para luego loguear un usuario y —como administrador— revocar el acceso, así como gestionar los tokens y sus refresh desde el punto de vista del usuario.

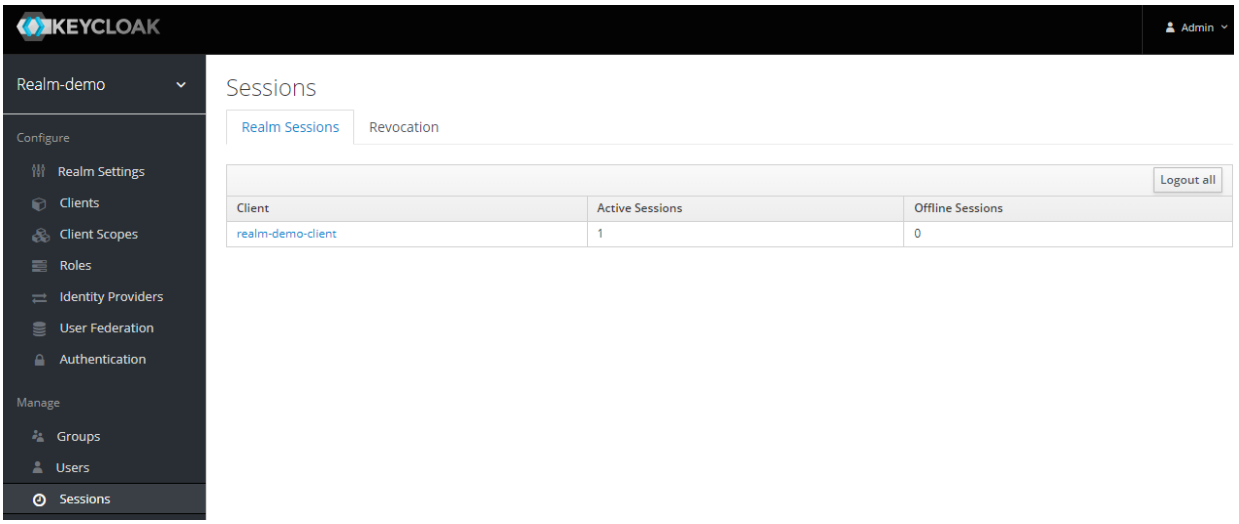
Enunciado

1. Crear un reino indicando en la opción de login que el usuario puede seleccionar la opción de "**Remember me**".
2. Crear un cliente confidencial del tipo OpenID dentro del reino del punto 1 indicando como URL de redirección "**/***".
3. Agregar dentro del cliente un role llamado "**user**".
4. Agregar 2 usuarios: **user1** y **user2**. Los passwords son el mismo nombre de usuario.
5. Utilizar la URL http://{host}:9091/realms/{realm-name}/.well-known/openid-configuration para identificar las URL que disponemos en nuestro nuevo reino que nos permitan autenticarnos.
6. Autenticarnos contra nuestra URL de **authorization_endpoint** del punto anterior:
    a. Para esto ponemos en nuestro navegador la URL de acuerdo a los pasos 1 y 2:

    http://localhost:9091/realms/realm-demo/protocol/openid-connect/auth?response_type=code&client_id=realm-demo-client

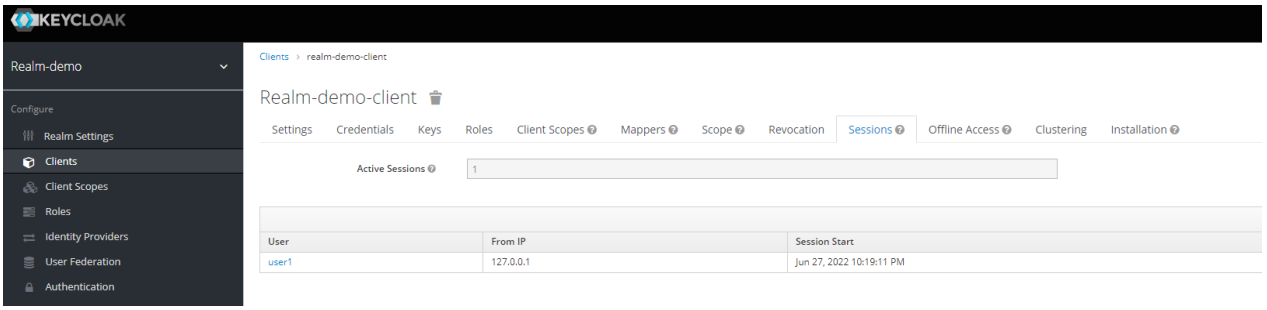
    b. Nos logueamos con **user1**.

7. Como usuarios administradores, vamos a ver los datos de usuarios logueados en nuestro reino desde una nueva pestaña en incógnito (para poder loguearnos como admins):
http://localhost:9091/admin/master/console/#/realms/realm-demo/sessions/realm



8. Como usuarios administradores, vamos a ver las sesiones del cliente creado en el punto 2.

http://localhost:9091/admin/master/console/#/realms/realm-demo/clients/4252e6d9-b144-4f6b-b894-e7ceeb4ab561/sessions

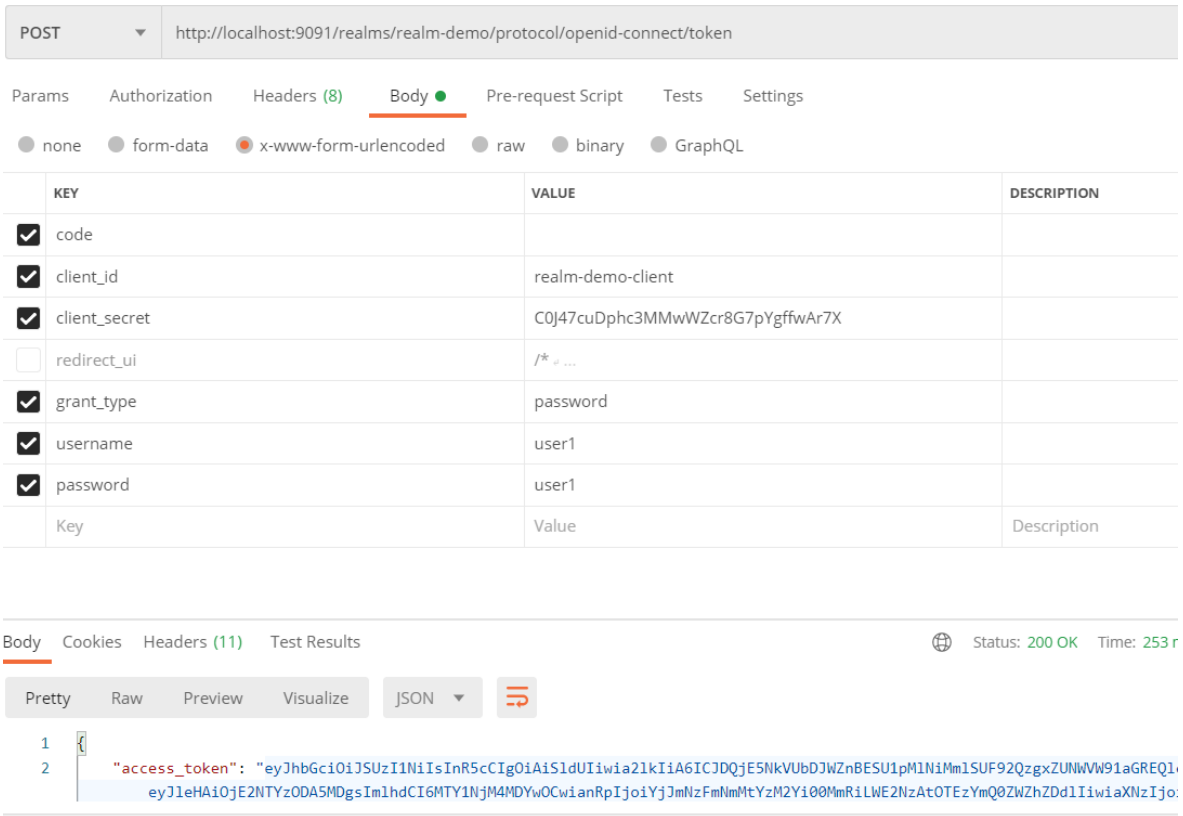


9. Desde la vista de sesiones del punto 7, vamos a desloguear a todos los usuarios activos desde el botón "**Logout all**".

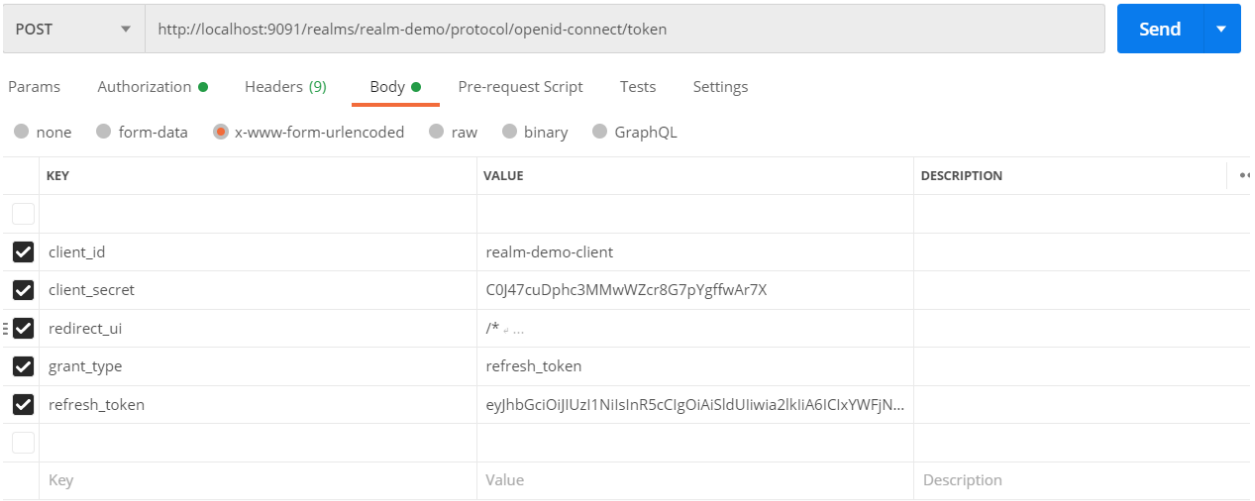
10. Si refrescamos el navegador del punto 6, al perder la sesión, **user1** es deslogueado del IAM.

11. Desde **Postman**, vamos a solicitar un token para el **user1** utilizando la URL con el patrón **{{server}}/auth/realms/{{realm}}/protocol/openid-connect/token**.

    Enviando en el body del request HTTP POST los parámetros que se ven en la siguiente imagen:

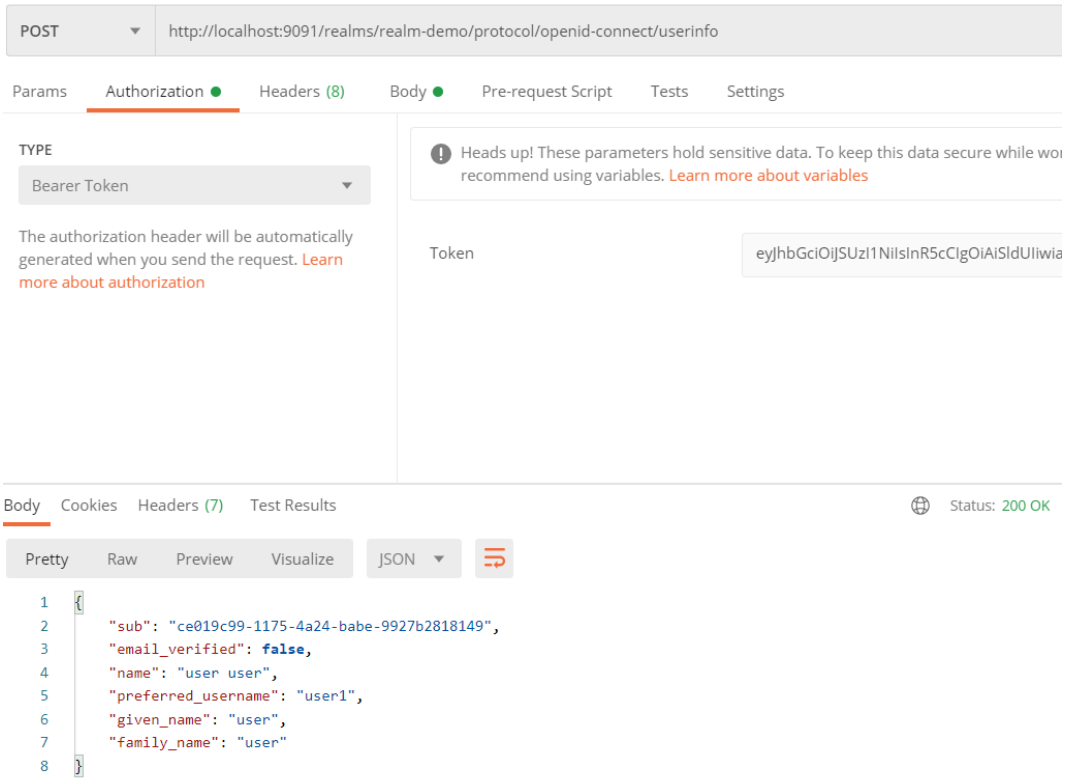


12. La URL del punto 11 nos devuelve un token de refresh para poder obtener un nuevo token de acceso cuando el original expire. Para poder utilizarlo vamos a usar la misma URL de Postman del punto 11 sin especificar usuario y contraseña, aclarando que lo que se solicita es un token de refresh.

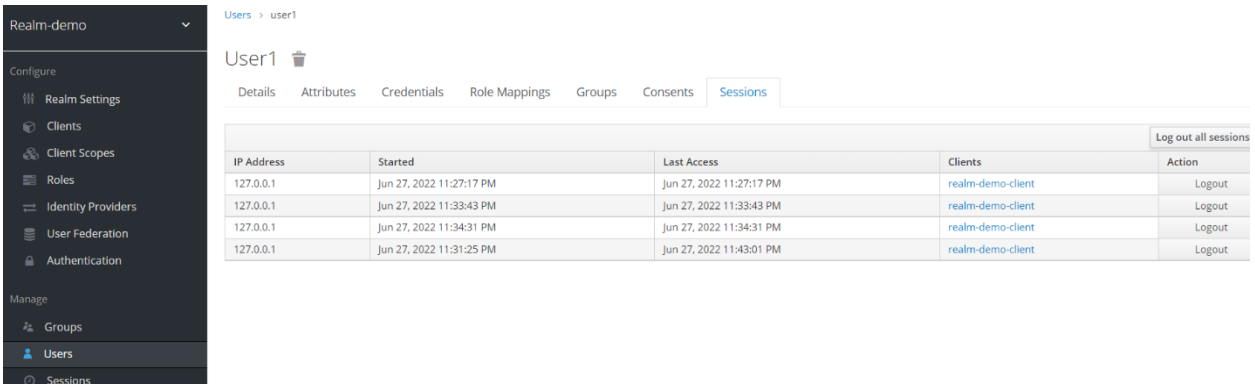


Este método POST lo podemos invocar tantas veces como queramos de acuerdo a la vida útil de nuestro refresh token.

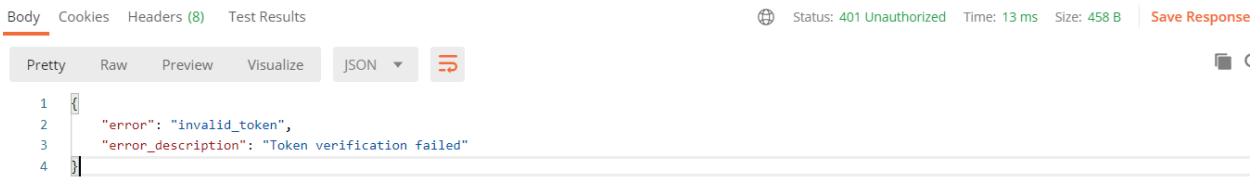
13. Utilizamos desde Postman la URL http://localhost:9091/realms/realm-demo/protocol/openid-connect/userinfo para solicitar información del token actual. Dicho token se enviará como una cabecera de autorización Bearer.



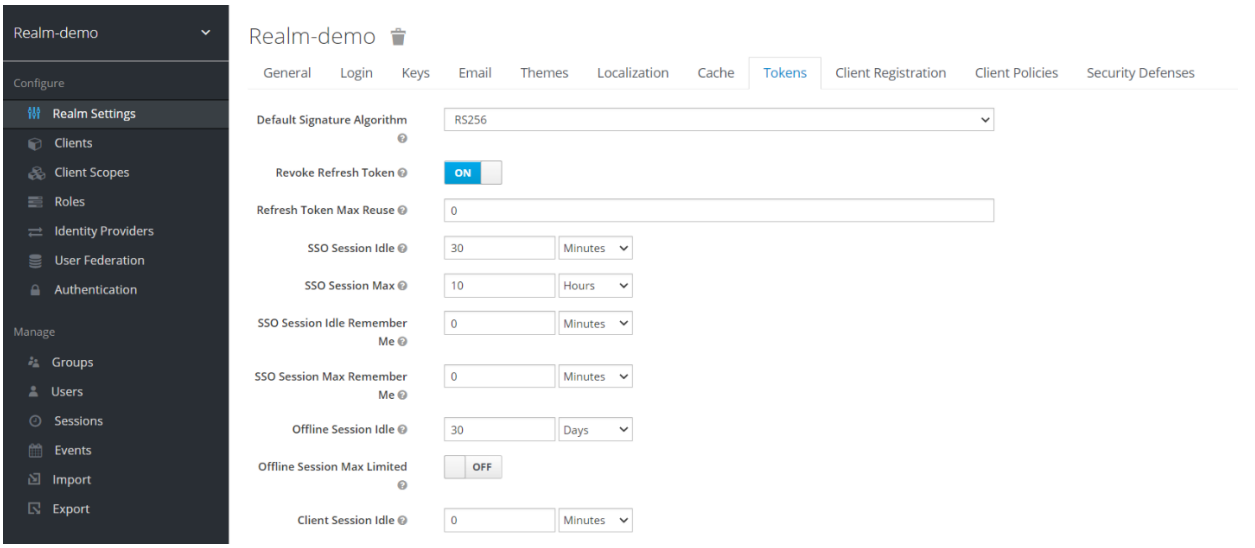
14. Como administradores de Keycloak, accedemos a la sección de usuarios, seleccionamos a **user1** y visualizamos las sesiones activas para todas las asociadas.



15. Si ejecutamos nuevamente el request del punto 12 y 13, vamos a tener un error de token inválido.



16. Vamos a configurar nuestro reino para que utilice rotación de tokens. Para esto, dentro de la configuración de este, vamos a activar la opción de "**Revoke Refresh Token**".



17. Si generamos un token de acceso e intentamos refrescar como en el punto 13, esto será válido solo la primera vez. Si queremos realizarlo más de una vez, debemos tomar el refresh token recibido en el JSON de la respuesta, cada vez que refrescamos el token.