Osserviamo i passi con il secondo programma di esempio

Nella cartella ch4 del repository scaricabile da
https://github.com/PacktPublishing/Keycloak-Identity-and-Access-Management-for-Modern-Applications

Troviamo un programma che può essere avviato con NodeJS:
npm install
npm start
Viene arrivato un servizio che risponde sulla porta 8000
(usare la URL localhost:8000 sul browser)

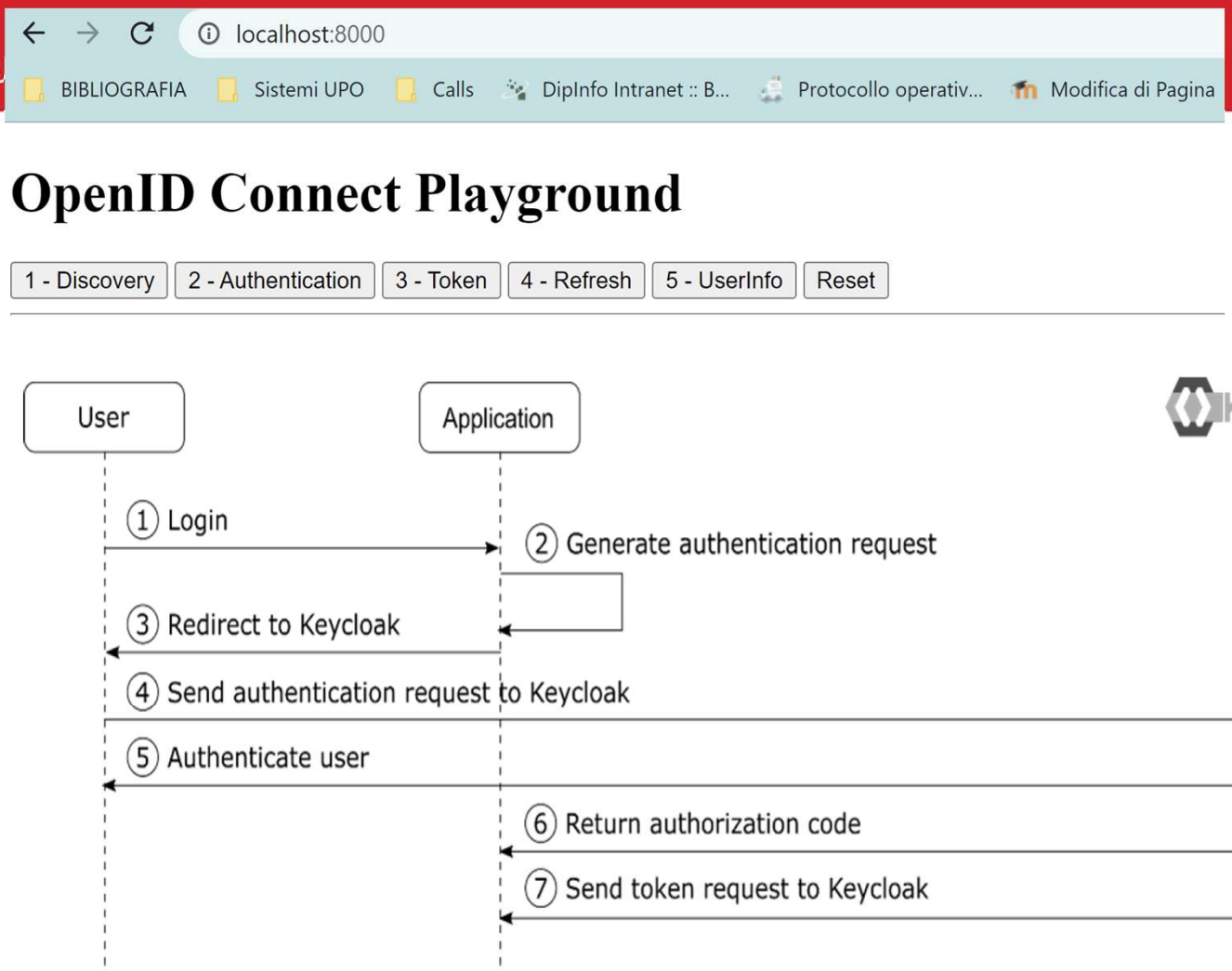
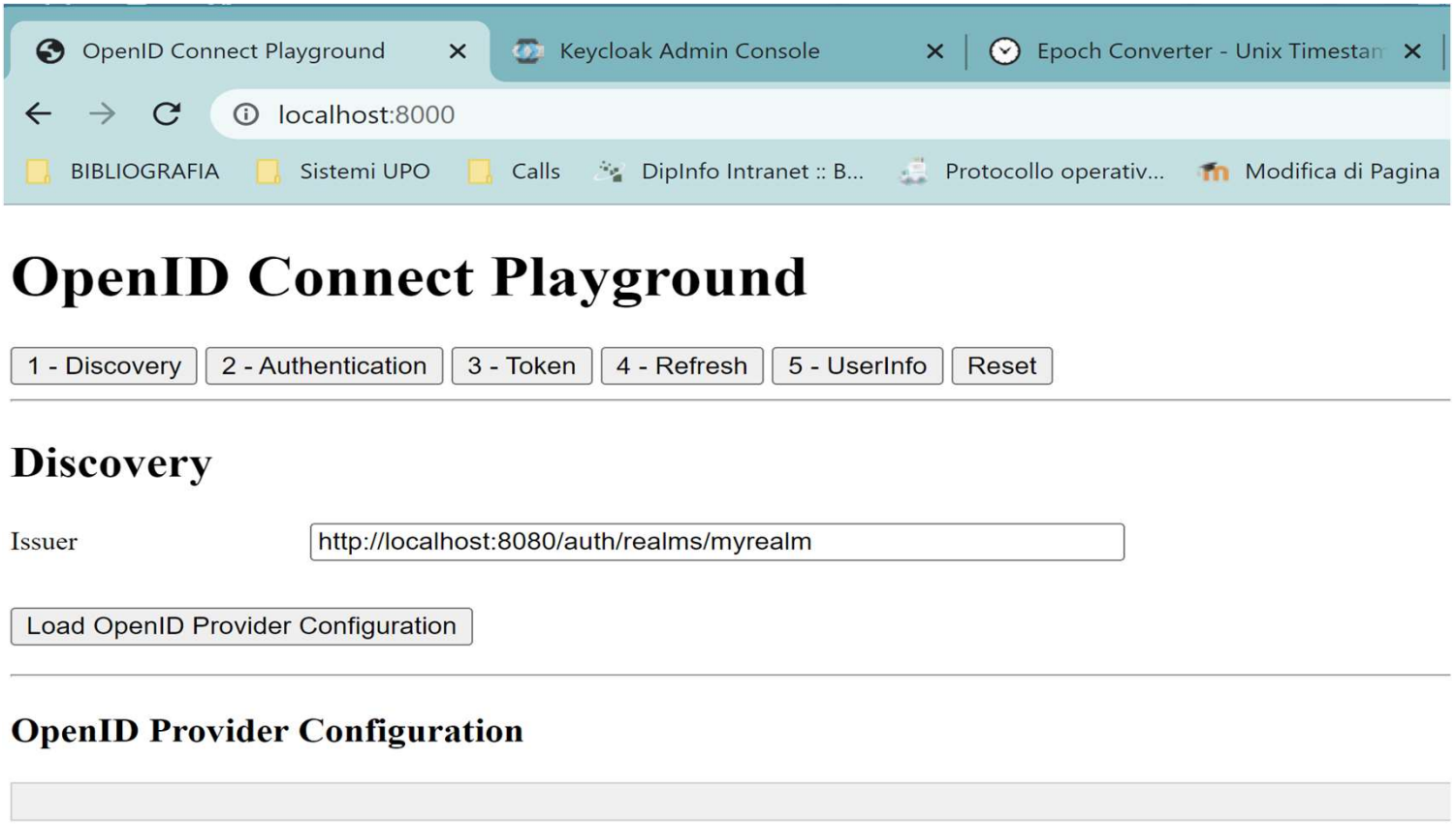


Figure 4.3 – The authorization code flow

Sequenza di passaggi che realizzano la sequenza del grant type authorization code e includono l'autenticazione su keycloak

1. Il primo passo permette di reperire gli endpoint da usare nei vari passaggi
2. Il secondo passo genera una richiesta di inizio della sequenza (che porterà all'autenticazione dell'utente su keycloak)
3. Il terzo passo permette di ottenere il token
4. Il quarto passo permette di «aggiornare» il token (refresh)
5. Nel quinto passo si vede come ottenere informazioni sull'utente tramite l'apposito endpoint

Interfaccia del programma di esempio



Esito del primo passo: endpoint, grant type, algoritmi firma

Gli endpoint che verranno utilizzati nei passi successivi sono:

- "authorization_endpoint": "http://localhost:8080/auth/realms/myrealm/protocol/openid-connect/auth",
- "token_endpoint": "http://localhost:8080/auth/realms/myrealm/protocol/openid-connect/token",

"userinfo_endpoint": "http://localhost:8080/auth/realms/myrealm/protocol/openid-connect/userinfo",

Grant type e response type

"grant_types_supported": ["authorization_code", ...]

- "response_types_supported": [
- "code", <<<<<<<< authorization code restituito
- …
- "id_token", <<<<<<<< token contenente le informazioni sull'utente autenticato
- "token", <<<<<<<< access token – per poter accedere alle risorse dell'utente
- …],

Sequenza di tipo «authorization code»

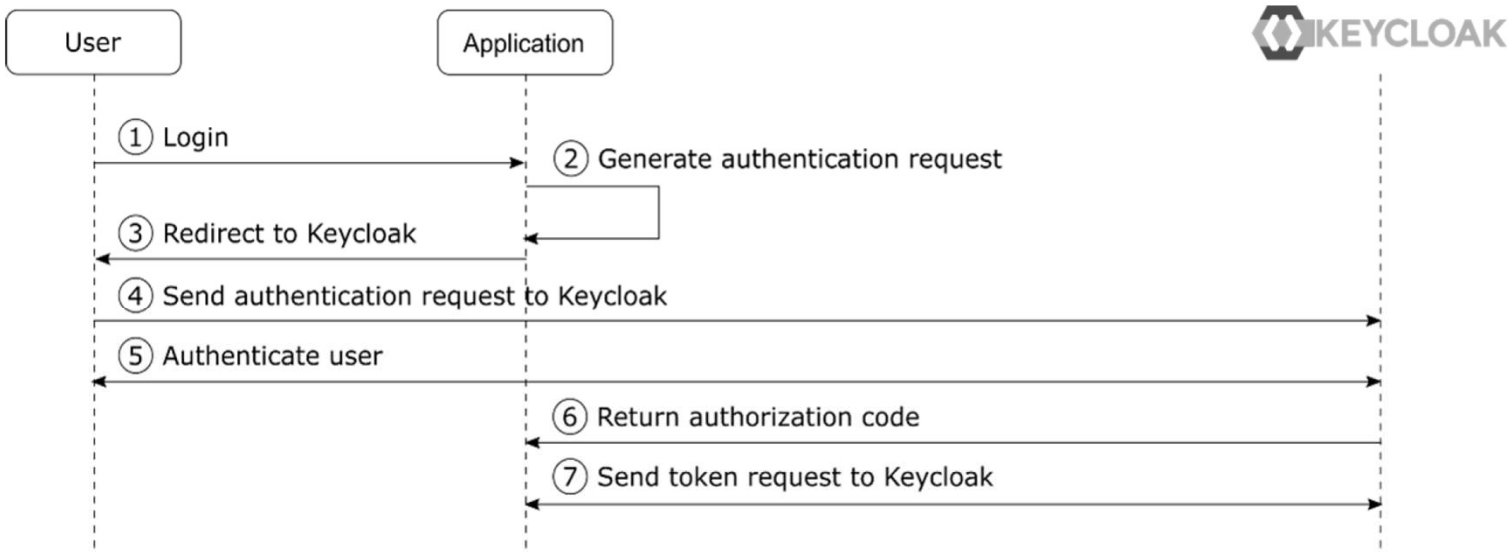


Figure 4.3 – The authorization code flow

UNIVERSITÀ DEL PIEMONTE ORIENTALE

OpenID Connect Playground

1 - Discovery | 2 - Authentication | 3 - Token | 4 - Refresh | 5 - UserInfo | Reset

Authentication

client_id	oidc-playground
scope	openid
prompt	
max_age	
login_hint	

Generate Authentication Request

Authentication Request

Send Authentication Request

Authentication Response

Passi da 2 a 4 della sequenza

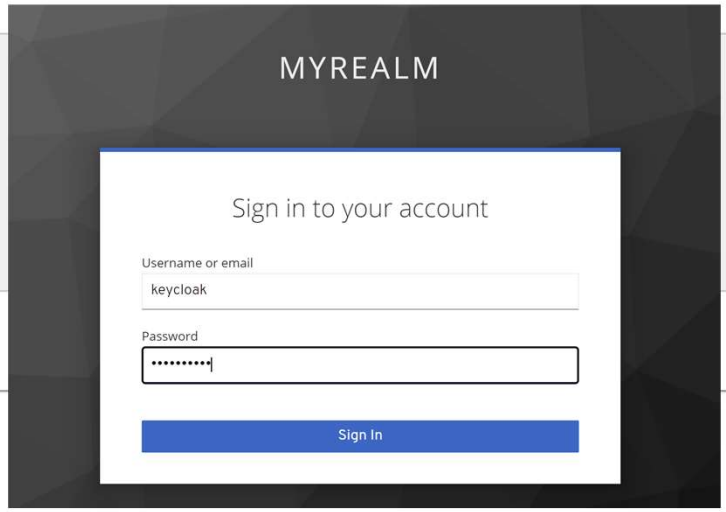
Generate Authentication Request – Send Authentication Request

Authentication Request

```
http://localhost:8080/auth/realms/myrealm/protocol/openid-connect/auth

client_id=oidc-playground
response_type=code
redirect_uri=http://localhost:8000/
scope=openid
```

Send Authentication Request



Authentication Response

```
code=bcbb470a-69d1-4b5d-ad5e-30bdfc7607e4.8f83dc7f-2008-4d81-a54d-604026383e63.2088eac8-450a-4071-a031-76bddf99acb6
```

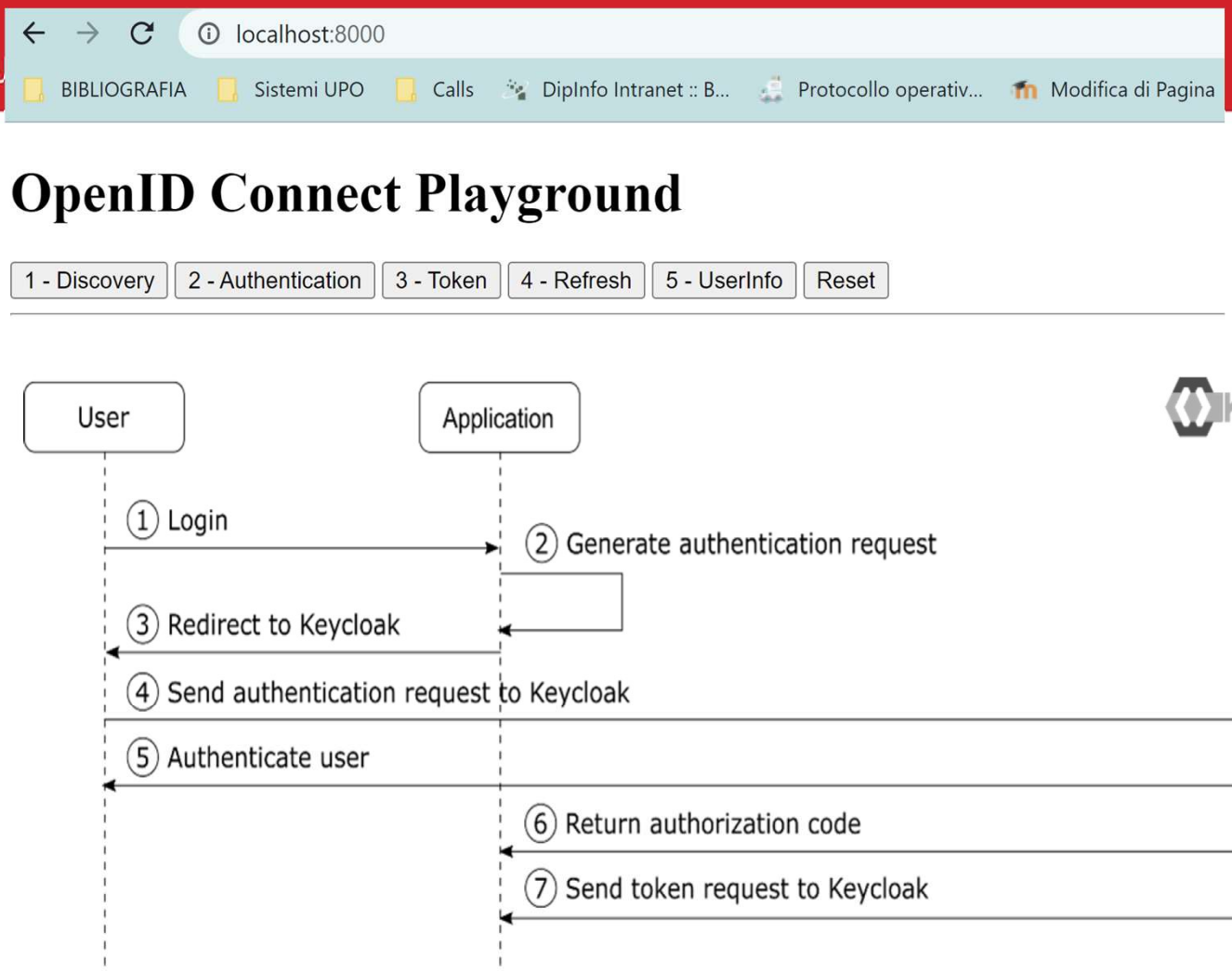


Figure 4.3 – The authorization code flow

PASSI 6-7 – richiesta del token utilizzando l'authorization code

Token

Authorization Code e216d3c9-cd32-419a-a026-73a9f8ad5767.f7c9874f-5829-470f-af12

Auth. Code dal passo precedente

Send Token Request

Token Request

```
http://localhost:8080/auth/realms/myrealm/protocol/openid-connect/token

grant_type=authorization_code
code=e216d3c9-cd32-419a-a026-73a9f8ad5767.f7c9874f-5829-470f-af12-69a31d74a970.2088eac8-450a-4071-a031-76
client_id=oidc-playground
redirect_uri=http://localhost:8000/
```

Token endpoint

Token Response


```
{
  "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICIxLUQyMHhRUnpIMTZuTDdwd1hLWldGbjl1OG
  "expires_in": 300,
  "refresh_expires_in": 1800,
  "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICJhYzI4OTI4OS01OGQzLTRmZDEtYTQ5ZC1iZ
  "token_type": "Bearer",
  "id_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICIxLUQyMHhRUnpIMTZuTDdwd1hLWldGbjl1OG4ySj
```


ID token
Access token
Refresh token

Scadenza del token:

https://www.epochconverter.com/


```
{
  "exp": 1684497331,
  "iat": 1684497031,
  "auth_time": 1684497024,
  "jti": "83e7fdda-1235-444f-a778-df55b2d82897",
  "iss": "http://localhost:8080/auth/realms/myrealm",
  "aud": "oidc-playground",
  "sub": "92777220-d719-431d-a161-faf72862c2ba",
  "typ": "ID",
  "azp": "oidc-playground",
  "session_state": "f7c9874f-5829-470f-af12-69a31d74a970",
  "at_hash": "WNzDwfmFHoE9J22EkP_uyg",
  "acr": "1",
  "yetanotherclaim": "My other claim",
  "email_verified": false,
  "realm_access": {
    "roles": [
      "offline_access",
      "uma_authorization",
      "myrole",
      "newrole"
    ]
  },
  "name": "Giuliana Annamaria Franceschinis",
  "preferred_username": "keycloak",
  "given_name": "Giuliana Annamaria",
  "family_name": "Franceschinis",
  "email": "giuliana.franceschinis@gmail.com",
  "picture": "https://upobook.uniupo.it/Files/People/284/e51940e6-1732-4d2f-b0f9-cf07d25f5bae.png"
}
```

Refresh del token

Refresh

Send Refresh Request

Refresh Request

```
http://localhost:8080/auth/realms/myrealm/protocol/openid-connect/token

grant_type=refresh_token
refresh_token=eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICJhYzI4OTI4OS01OGQzLTRmZDEtYTQ5ZC1iZTkwOT
client_id=oidc-playground
scope=openid
```

Refresh Response


```
{
  "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICIxLUQyMHhRUnpIMTZuTDdwd1hLWldGbjl1O
  "expires_in": 300,
  "refresh_expires_in": 1800,
  "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICJhYzI4OTI4OS01OGQzLTRmZDEtYTQ5ZC1i
  "token_type": "Bearer",
  "id_token": "eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICIxLUQyMHhRUnpIMTZuTDdwd1hLWldGbjl1OG4yS
  "not-before-policy": 0,
  "session_state": "f7c9874f-5829-470f-af12-69a31d74a970",
  "scope": "openid profile email"
}
```

Userinfo endpoint – richiedere dati utente

UserInfo Request

```
http://localhost:8080/auth/realms/myrealm/protocol/openid-c

Authorization: Bearer eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUI
```

Utilizzo il token

UserInfo Response


```
{
  "sub": "92777220-d719-431d-a161-faf72862c2ba",
  "email_verified": false,
  "realm_access": {
    "roles": [
      "offline_access",
      "uma_authorization",
      "myrole",
      "newrole"
    ]
  },
  "name": "Giuliana Annamaria Franceschinis",
  "preferred_username": "keycloak",
  "given_name": "Giuliana Annamaria",
  "family_name": "Franceschinis",
  "email": "giuliana.franceschinis@gmail.com",
  "picture": "https://upobook.uniupo.it/Files/People/284/e5
}
```