Contoh Auth Django Apps dengan Keycloak / RH-SSO

1. Django Project

Di contoh ini, aplikasi Django yang kita buat berjalan di http://localhost:8000

1. Django / Python Library

Menambahkan library django-allauth

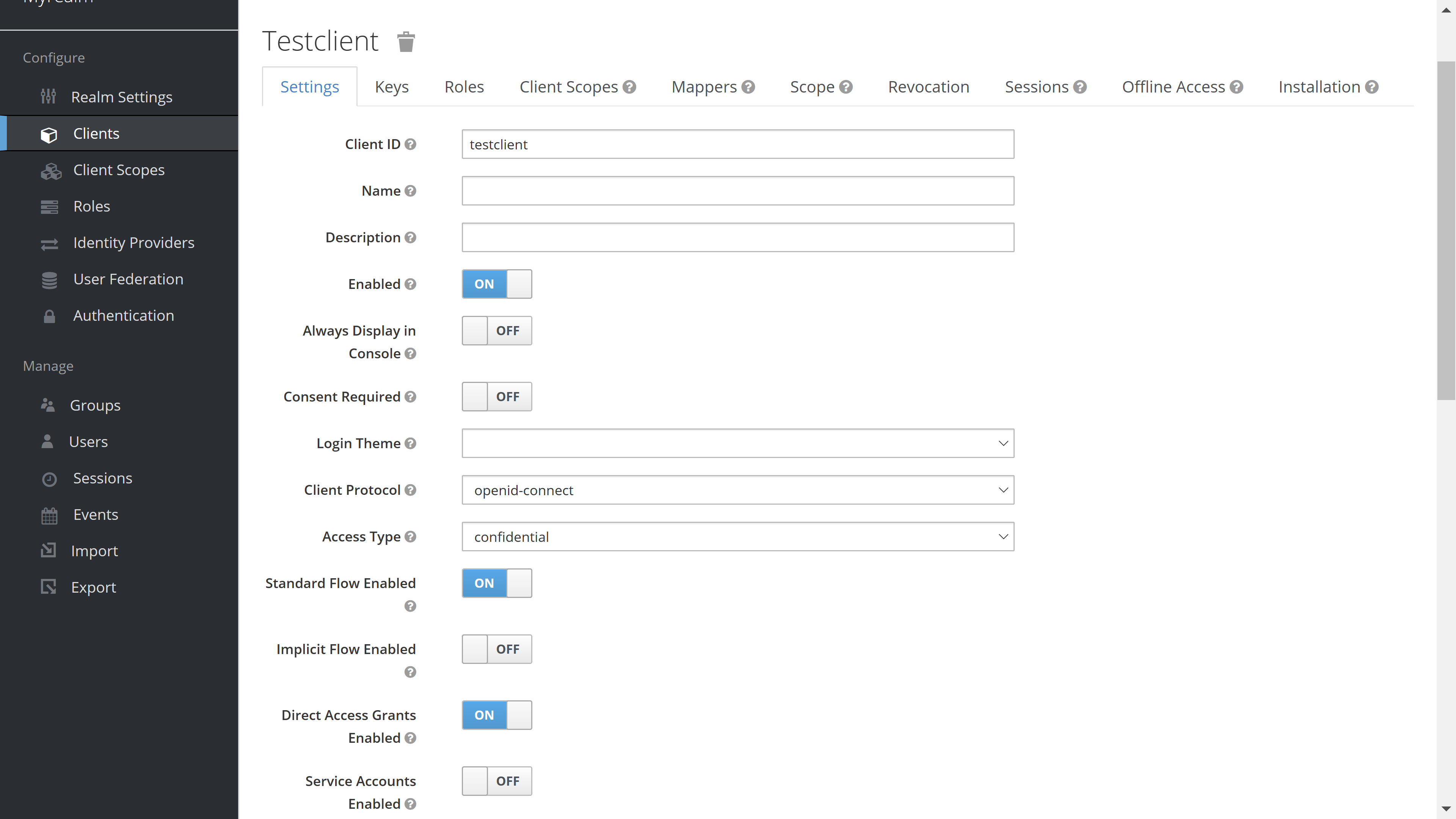
pip install django-allauth

1. Keycloak / RH-SSO Configuration

Buat realm dan client, misal:

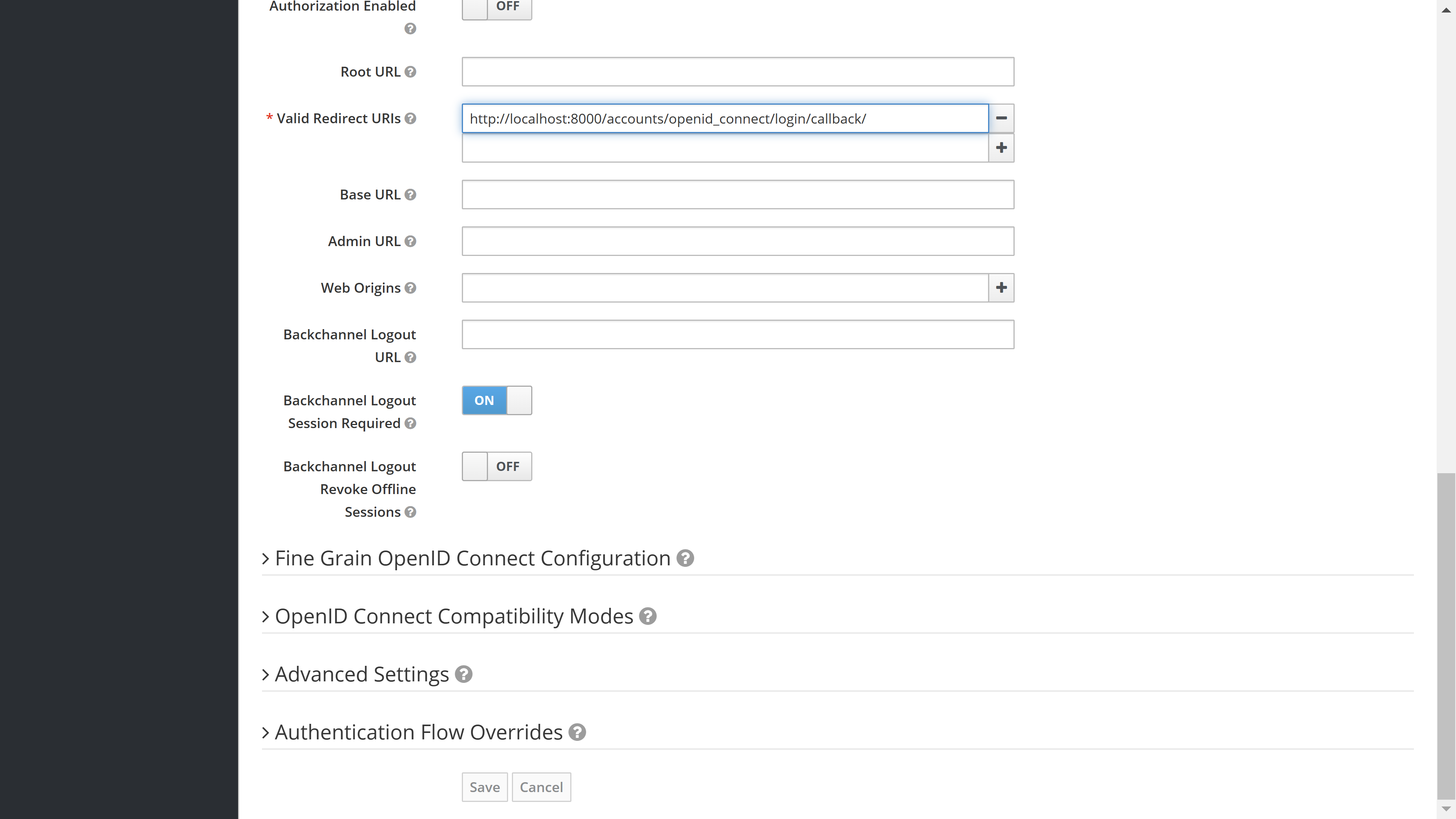
Realm: myrealm

Client: testclient

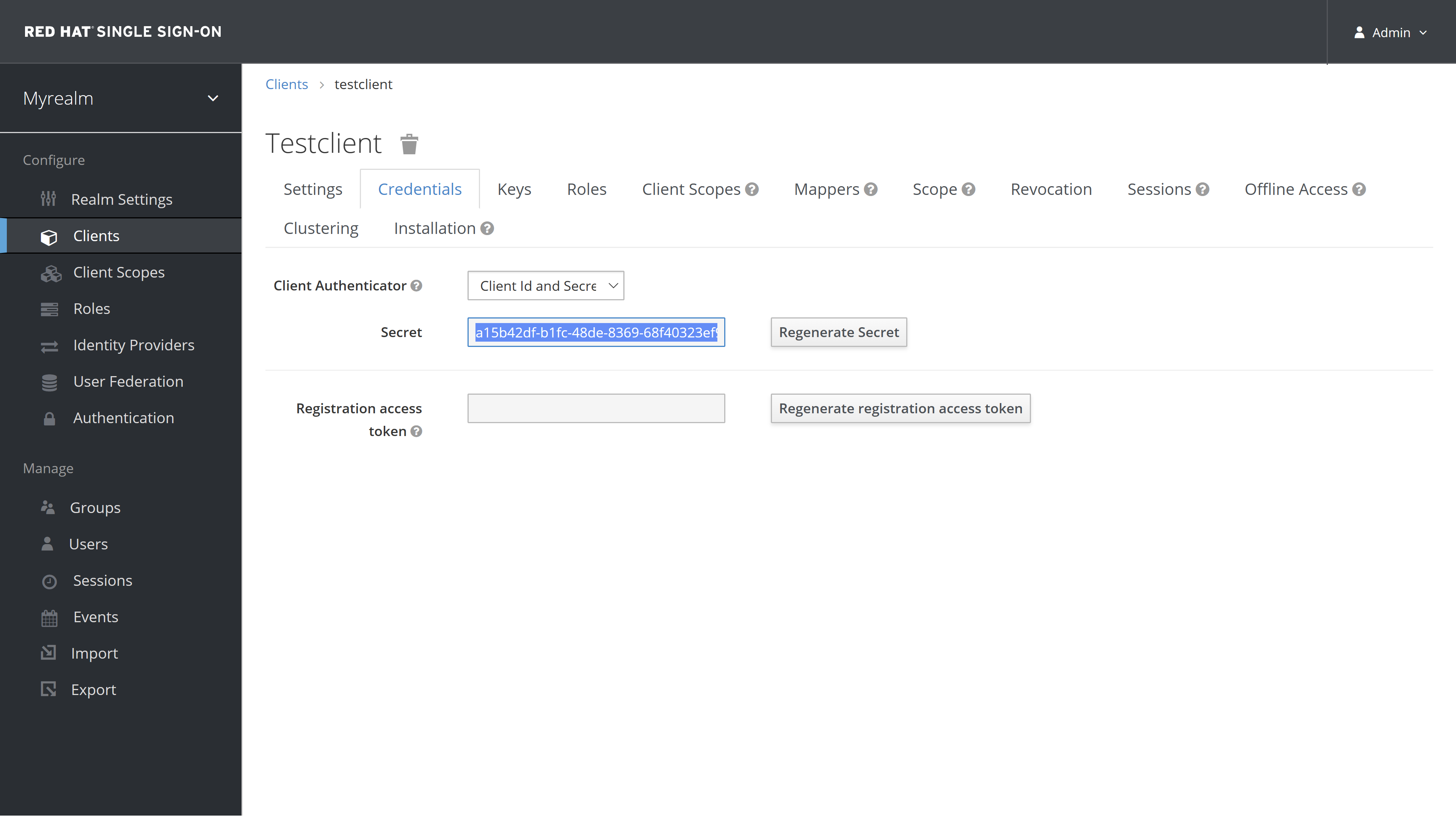


Set Access Type jadi “confidential”

Kemudian tambahkan Valid Redirect Urls: “http://localhost:8000/accounts/openid\_connect/login/callback/”



Kemudian ke tab “Credentials”, silahkan copy Client Secret



1. settings.py Configuration

Ikuti petunjuk di dokumentasi django-allauth

<https://django-allauth.readthedocs.io/en/latest/installation.html>

Tambahkan di bagian INSTALLED\_APPS

INSTALLED\_APPS = [

……

    'allauth',

    'allauth.account',

    'allauth.socialaccount',

    'allauth.socialaccount.providers.keycloak',

]

Tambahkan di bagian AUTHENTICATION\_BACKENDS

AUTHENTICATION\_BACKENDS = [

    # Needed to login by username in Django admin, regardless of `allauth`

    'django.contrib.auth.backends.ModelBackend',

    # `allauth` specific authentication methods, such as login by e-mail

    'allauth.account.auth\_backends.AuthenticationBackend',

]

Tambahkan di bagian SOCIALACCOUNT\_PROVIDERS

SOCIALACCOUNT\_PROVIDERS = { # changed

        'keycloak': {

        'KEYCLOAK\_URL': 'http://localhost:8080/auth',

        'KEYCLOAK\_REALM': 'myrealm',

    },

}

Tambahkan juga SITE\_ID, ini disesuaikan dengan konfigurasi di bagian Django-Admin di bawah nanti

SITE\_ID = 2

Tambahkan juga LOGIN\_REDIRECT\_URL, ini endpoint yang akan dipanggil setelah authentikasi sukses

LOGIN\_REDIRECT\_URL = '/hello'

1. urls.py Configuration

Ikuti petunjuk di dokumentasi django-allauth

<https://django-allauth.readthedocs.io/en/latest/installation.html>

from django.contrib import admin

from django.urls import path, include

from . import hello

urlpatterns = [

    path('admin/', admin.site.urls),

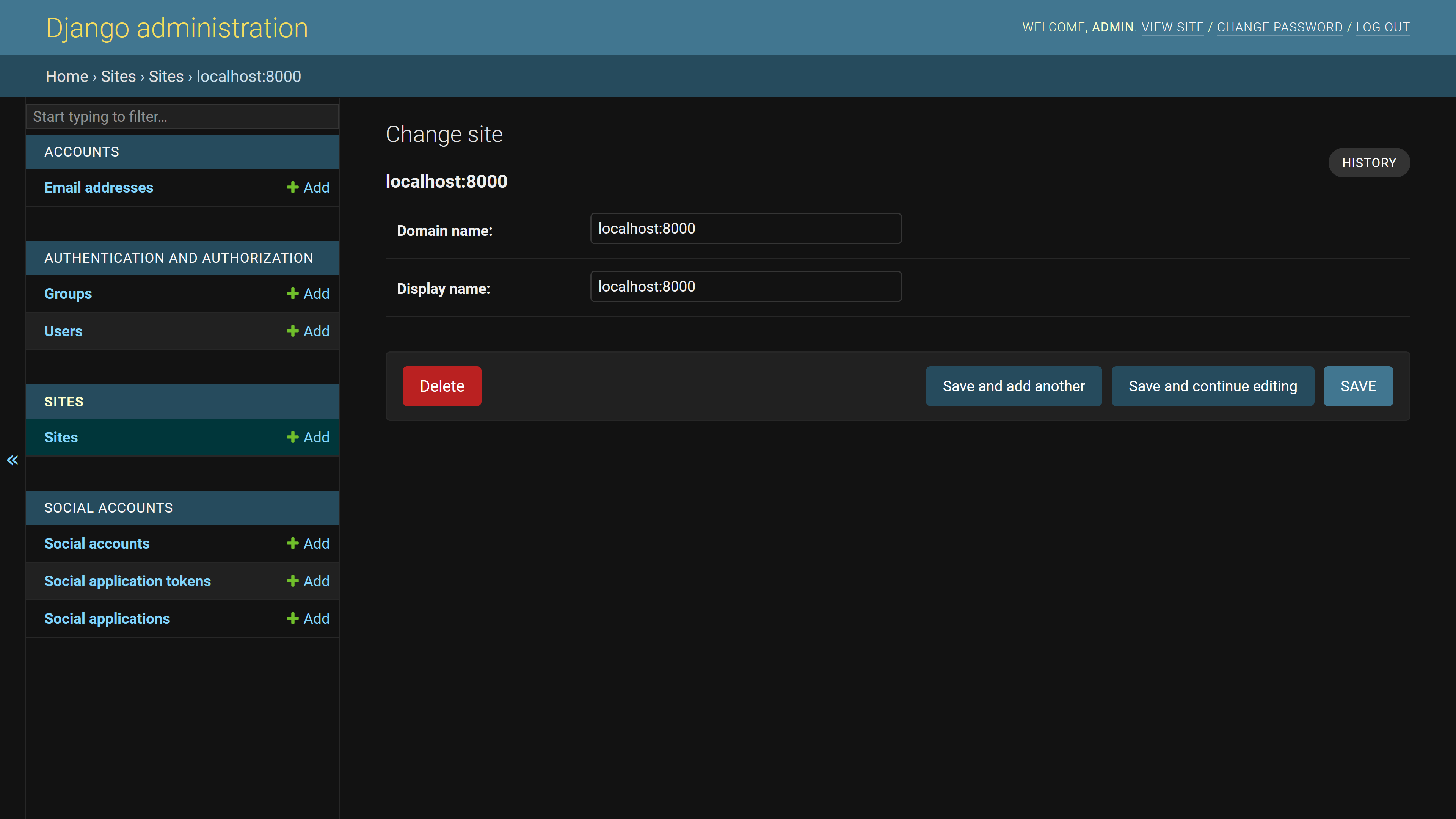
    path('accounts/', include('allauth.urls')),

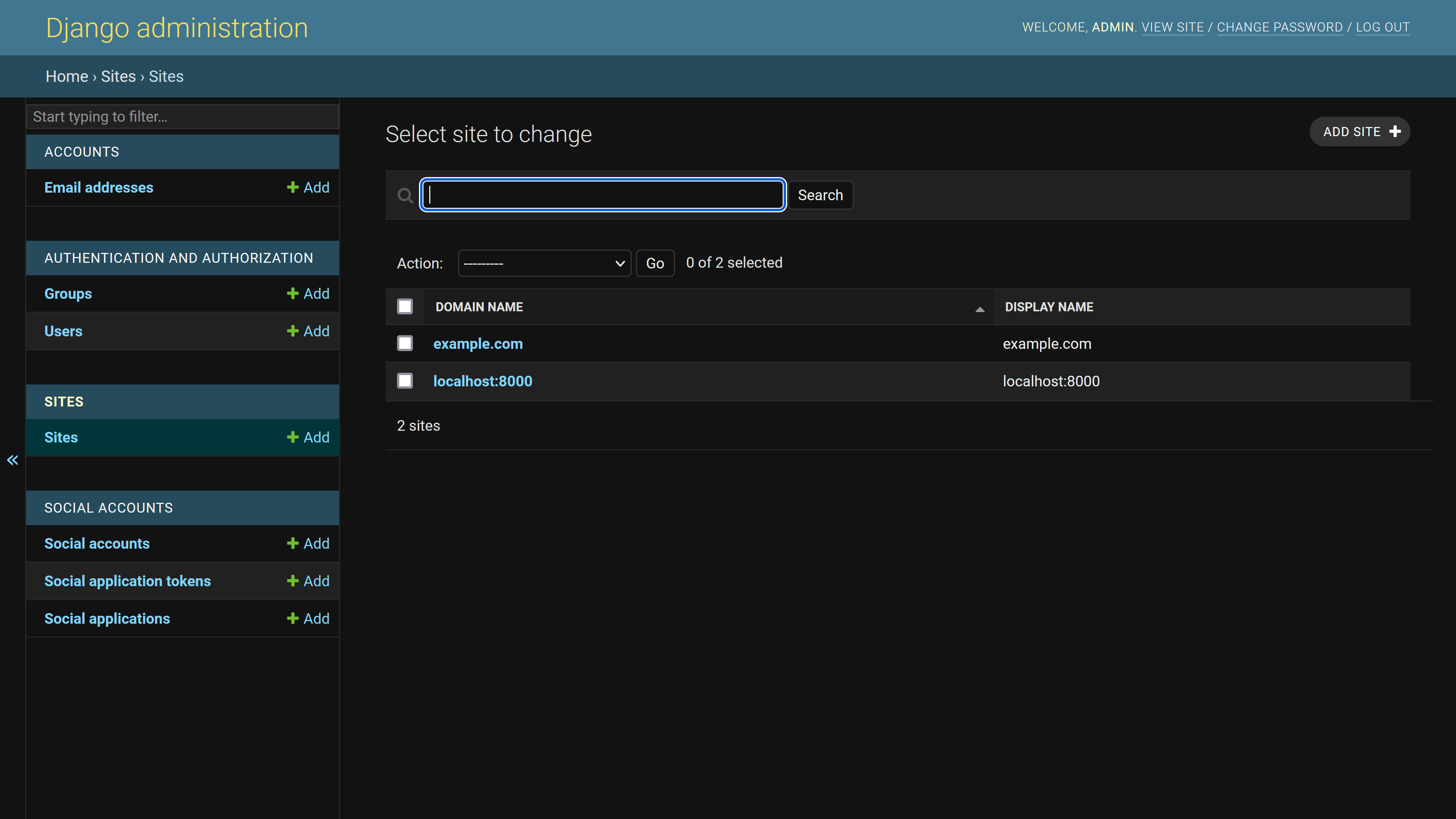
    path('hello/', hello.hello\_world, name='hello\_world'),

]

1. Django-Admin Configuration

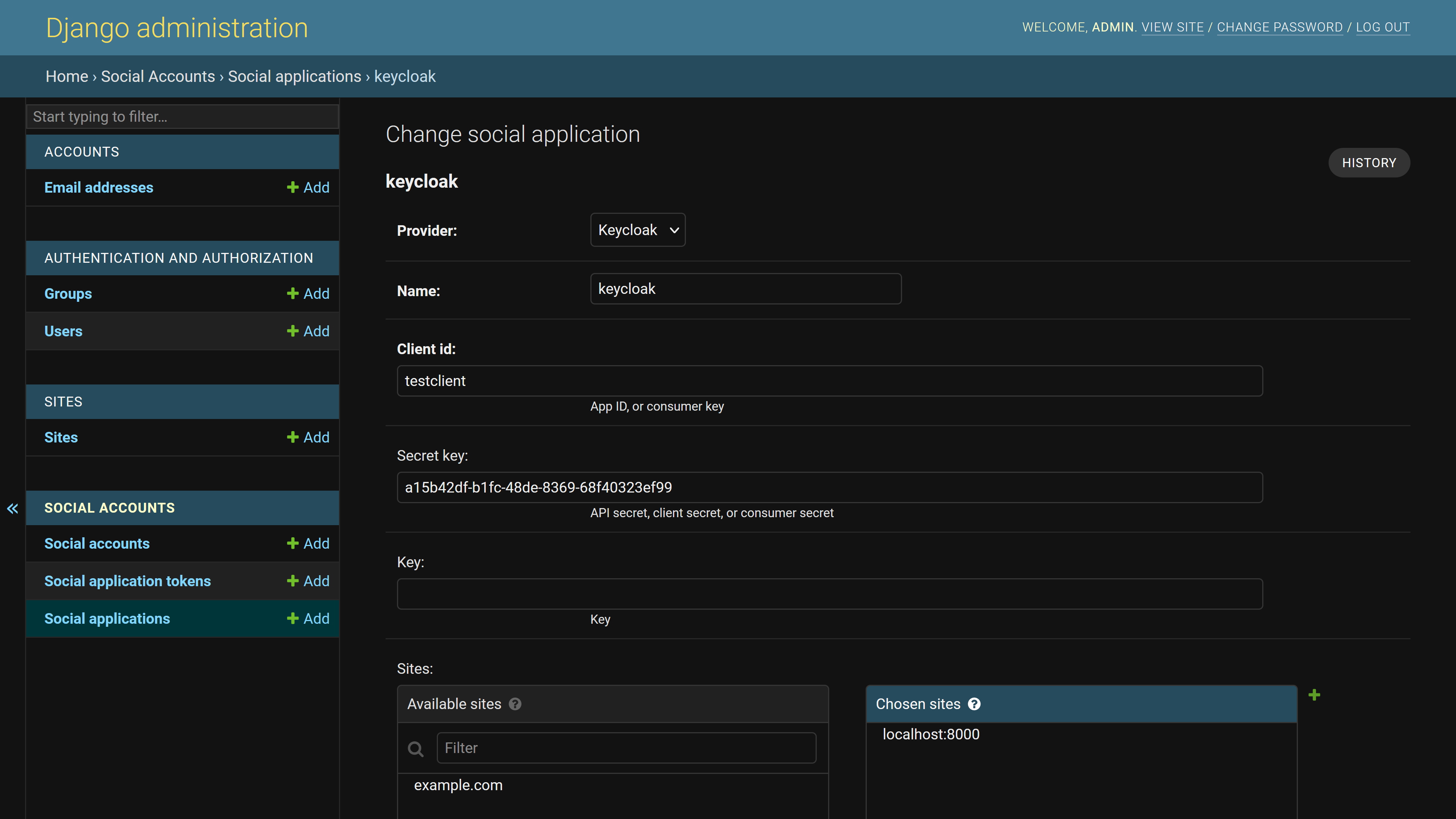
Tambahkan Site: localhost:8000, ini disesuaikan dengan aplikasi django kita





Penting juga untuk diperhatikan, karena sebelumnya sudah ada site example.com (dengan SITE\_ID = 1), maka site yang kita tambahkan akan mendapatkan SITE\_ID = 2. Ini yang kita sesuaikan dengan konfigurasi di settings.py yang dijelaskan di atas

Kemudian tambahkan Social Application



Dengan value sbb:

Provider: Keycloak

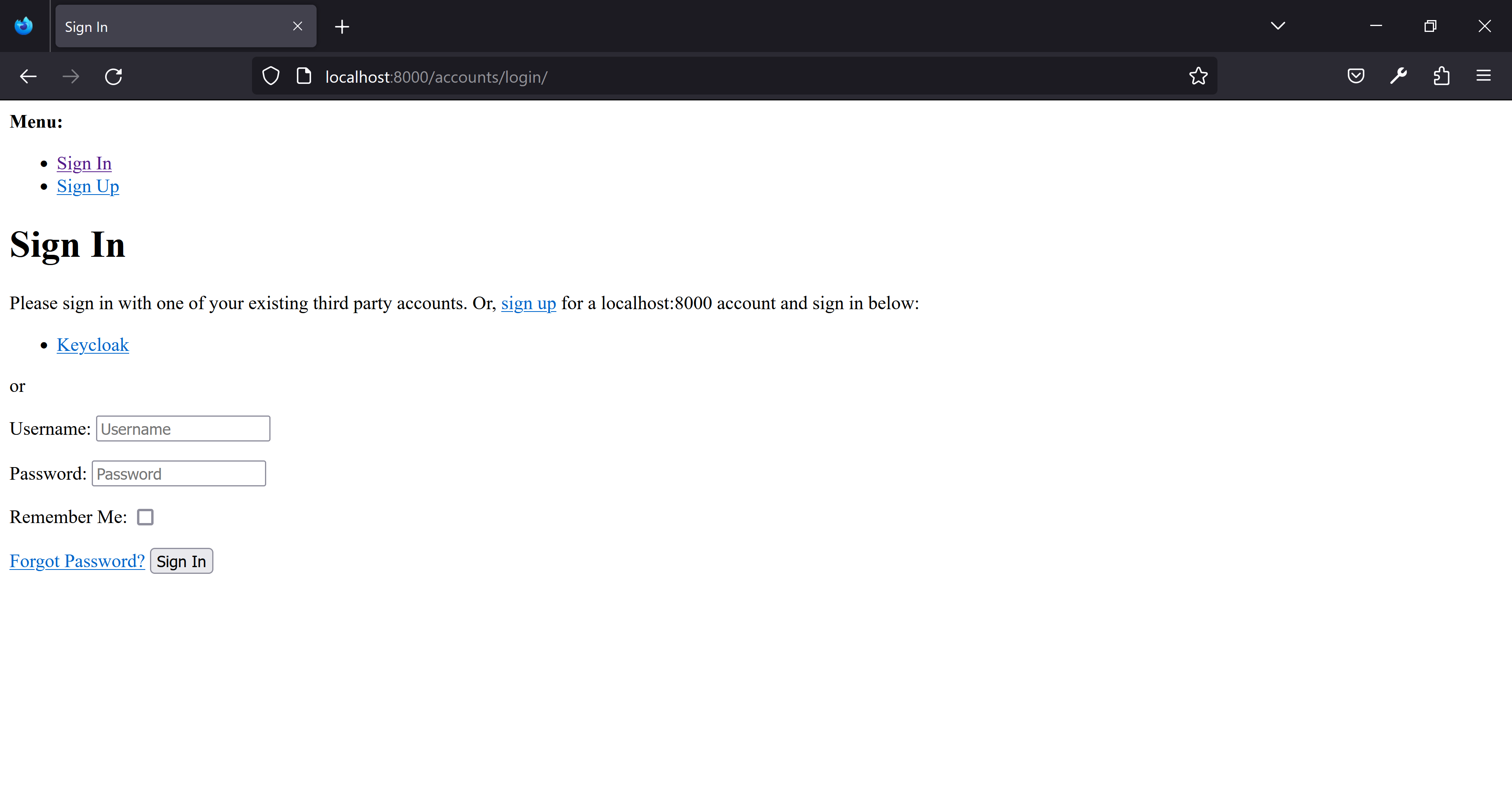
Client Id: testclient 🡪 sesuai dengan konfigurasi client di keycloak admin console

Secret Key: [client secret] 🡪 sesuai dengan konfigurasi client di keycloak admin console

Choosen Sites: 🡪 tambahkan localhost:8000 (aplikasi kita)

1. Test

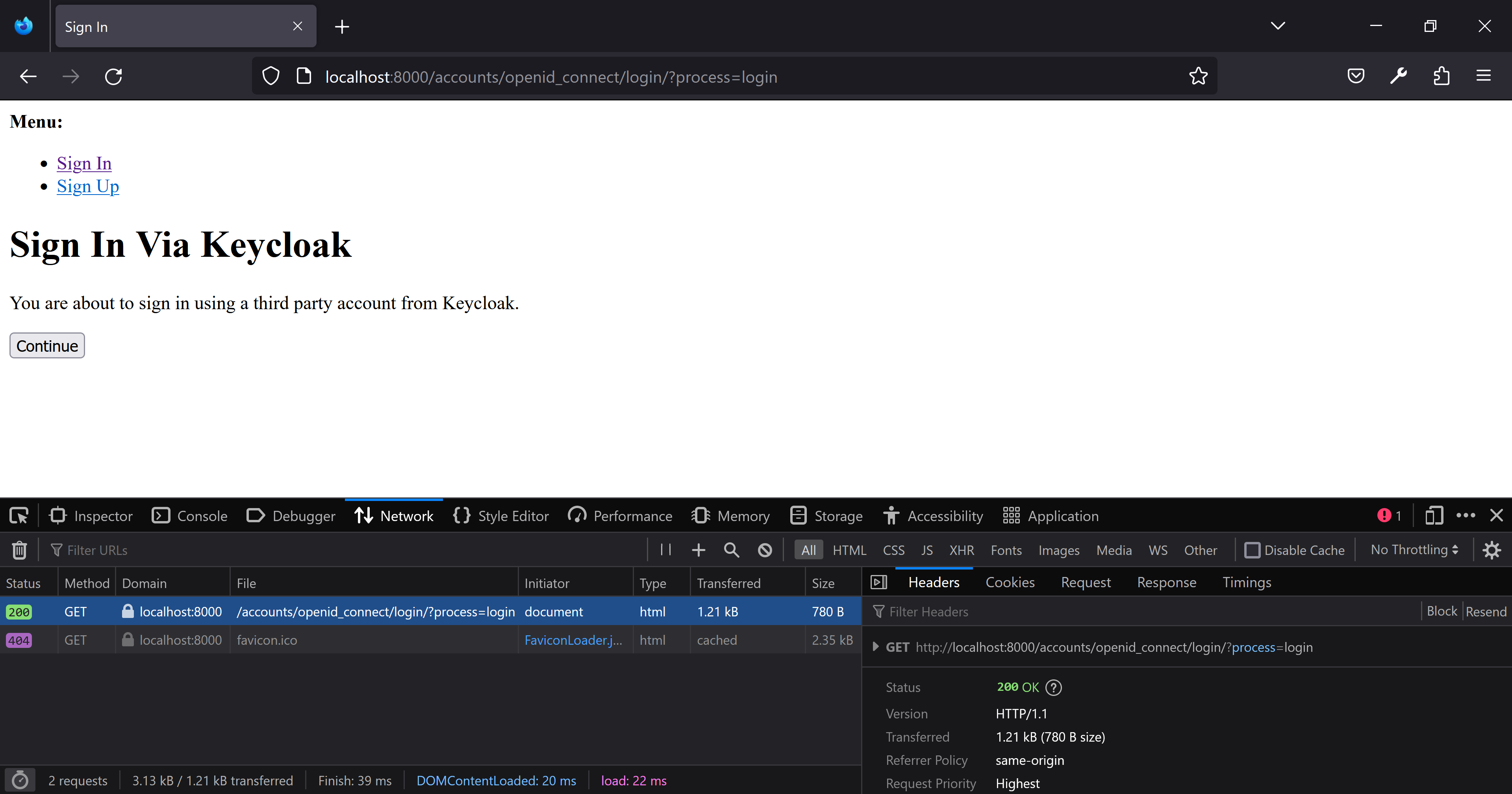
Membuka login page: <http://localhost:8000/accounts/login>



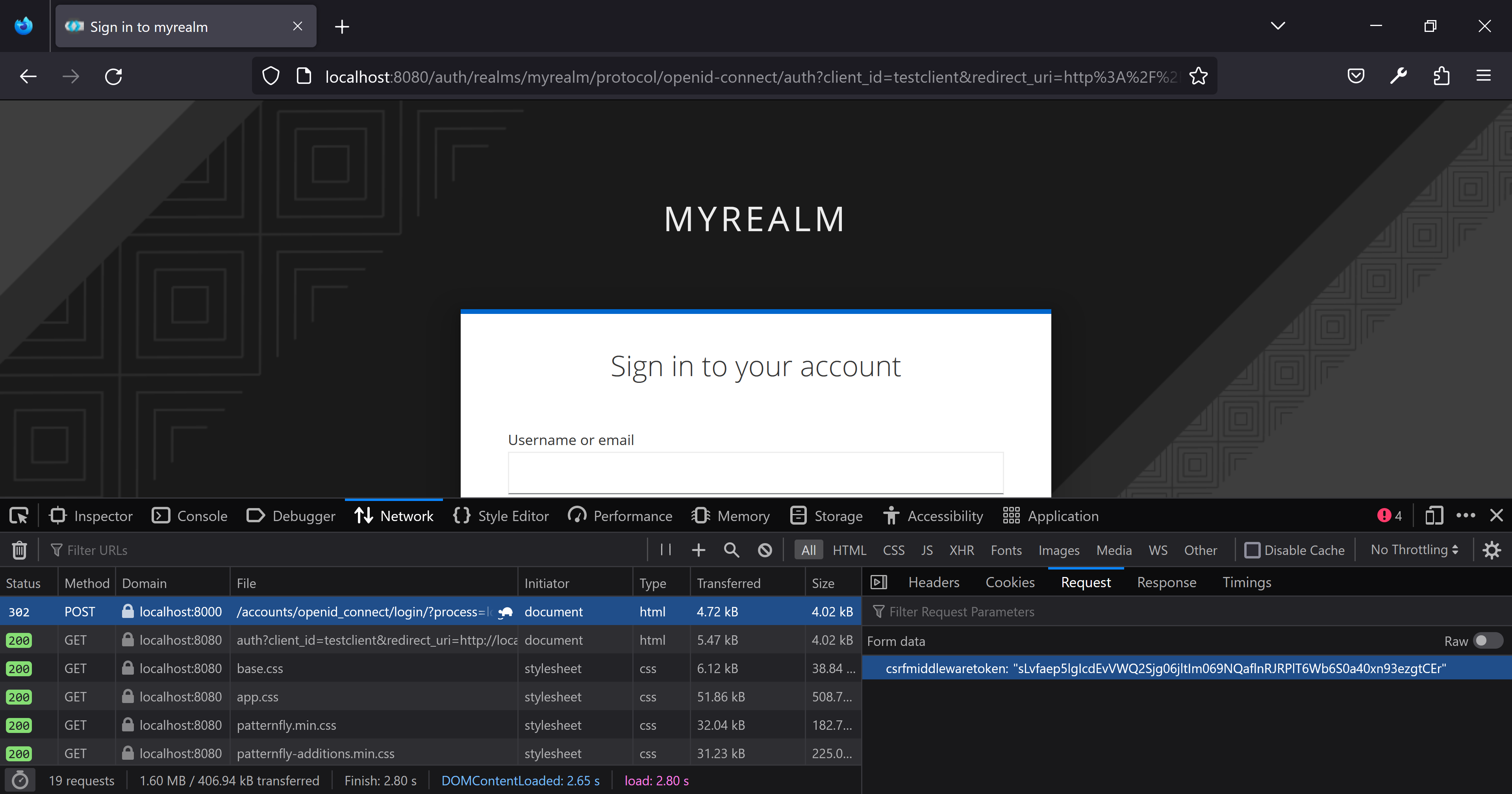
Klik Keycloak, akan membuka halaman selanjutnya

GET

<http://localhost:8000/accounts/openid_connect/login/?process=login>



Klik Continue



POST

<http://localhost:8000/accounts/openid_connect/login/?process=login>

{

"csrfmiddlewaretoken": "sLvfaep5lgIcdEvVWQ2Sjg06jltIm069NQaflnRJRPlT6Wb6S0a40xn93ezgtCEr"

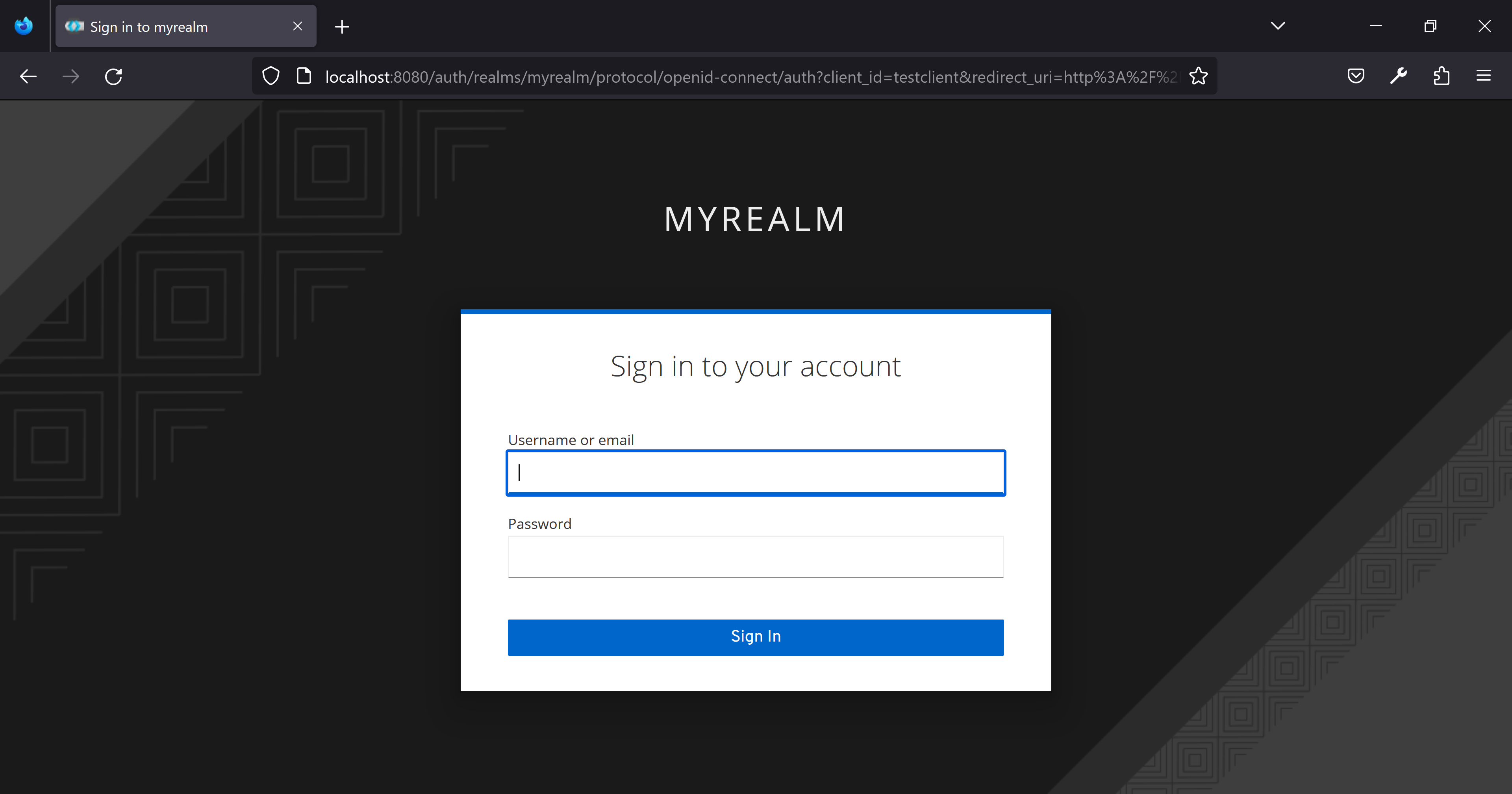
}

Response: 302

Kemudian akan diredirect ke…

GET

<http://localhost:8080/auth/realms/myrealm/protocol/openid-connect/auth?client_id=testclient&redirect_uri=http%3A%2F%2Flocalhost%3A8000%2Faccounts%2Fopenid_connect%2Flogin%2Fcallback%2F&scope=profile+openid+email&response_type=code&state=fgjSloWwTQJL>



Masukkan username dan password sesuai user di realm keycloak, klik Sign In, akan di-redirect ke endpoint yang kita set di LOGIN\_REDIRECT\_URL

\* Alternatif lain: menggunakan provider OpenID Connect (instead of Keycloak)

Di **settings.py**

Di bagian INSTALLED\_APPS, ganti 'allauth.socialaccount.providers.keycloak' dengan 'allauth.socialaccount.providers.openid\_connect'

INSTALLED\_APPS = [

    ……

    'django.contrib.sites',

    'allauth',

    'allauth.account',

    'allauth.socialaccount',

    'allauth.socialaccount.providers.openid\_connect',

]

Di bagian SOCIALACCOUNT\_PROVIDERS

SOCIALACCOUNT\_PROVIDERS = {

    "openid\_connect": {

        "SERVERS": [

            {

                "id": "openid\_connect",  # 30 characters or less

                "name": "OpenId Connect",

                "server\_url": "http://localhost:8080/auth/realms/myrealm",

                # Optional token endpoint authentication method.

                # May be one of "client\_secret\_basic", "client\_secret\_post"

                # If omitted, a method from the the server's

                # token auth methods list is used

                "token\_auth\_method": "client\_secret\_basic",

                "APP": {

                    "client\_id": "testclient",

                    "secret": "dccc6fea-f521-4b08-969d-7a6b0136bdbf",

                },

            },

        ]

    }

}

Referensi:

django-allauth Documentation:

<https://django-allauth.readthedocs.io/en/latest/installation.html>

SSO - Authenticate GeoNode against a keycloack server

<https://gist.github.com/t-book/0fb30804e217bdeb064dd91b5041fbc9>