|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Token Store | | Authentication Provider | | Client Details | |
| Memory | Jdbc / Custom Token Store | Memory | Jdbc / Custom  Authentication Provider | Memory | Jdbc / Custom Client Details Provider |

Beberapa Grant Types:

1. Flow Grant Type Authorization Code
2. Flow Grant Type User Password
3. Flow Grant Type Implicit
4. Flow Grant Type Client Credentials

Details

Grant Types:

1. Flow Grant Type Authorization Code

* Digunakan apa bila kita ingin memverifikasi identitas di aplikasi client,
* Biasanya digunakan apabila aplikasi clientya dapat menyimpan **client secret** 🡺 Client Details (client id & password)

(Misalnya di server PHP,JAVA,NODE JS )

* Dalam implementasinya sebelum menembak callback akan di redirect ke login provider

Setelah login berhasil maka redirect + auth\_code akan di berikan

**Formatnya sbb :**

**http://<ip>:<port>/oauth/authorize?client\_id=<client\_secret>&response\_type=<code>&redirect\_uri=<callback\_yang\_dituju>**

**Ex:**

**http://localhost:8787/oauth/authorize?client\_id=clientauthcode&response\_type=code&redirect\_uri=http://localhost:8712/api/state/new**

Step by stepnya adalah sbb :

1. Buka browser, arahkan ke

http://localhost:8787/oauth/authorize?client\_id=clientauthcode&response\_type=code&redirect\_uri=http://localhost:8787/api/state/new

1. Kita akan diminta login dan kemudian melakukan otorisasi. Setelah login, kita akan disajikan halaman otorisasi, pilih radio button Approve kemudian klik tombol Authorize.
2. Kita akan diredirect ke url yang kita sebutkan pada variabel redirect\_uri di langkah kedua di atas dengan ditambahi parameter authorization code. URL hasil redirectnya seperti ini:

http://localhost:8787/api/state/new?code=8OppiR

Ambil kodenya, yaitu 8OppiR, untuk digunakan pada langkah selanjutnya.

1. Lakukan request dari aplikasi client untuk menukar authorization code dengan access token. Sebagai contoh, kita akan gunakan aplikasi commandline curl atau post, atau method lainnya. Berikut perintahnya

curl -X POST -vu clientauthcode:123456 http://localhost:10000/oauth/token -H "Accept: application/json" -d "grant\_type=authorization\_code&code=8OppiR&redirect\_uri=http://localhost:10001/api/state/new"

1. Kita akan diberikan access token dalam response JSON seperti ini

{

"access\_token":"08664d93-41e3-473c-b5d2-f2b30afe7053",

"token\_type":"bearer",

"refresh\_token":"436761f1-2f26-412b-ab0f-bbf2cd7459c4",

"expires\_in":43199,

"scope":"write read"

}

1. Access token tersebut bisa digunakan aplikasi client untuk mengakses resource terproteksi seperti ini

curl http://localhost:10001/api/admin?access\_token=08664d93-41e3-473c-b5d2-f2b30afe7053

1. Resource server mengecek ke authorization server apakah token tersebut valid atau tidak dengan HTTP request seperti ini

curl -X POST -vu clientauthcode:123456 http://localhost:8787/oauth/check\_token?token=08664d93-41e3-473c-b5d2-f2b30afe7053

1. Authorization server akan membalas request check token dengan response seperti ini

{

"aud": ["aplikasi"],

"exp": 1444158090,

"user\_name": "patar timotius",

"authorities": ["ROLE\_OPERATOR", "ROLE\_SUPERVISOR"],

"client\_id": "clientauthcode",

"scope": ["read", "write"]

}

1. Resource Server Akan memberikan response as proper as it is

{"sukses":true,"page":"admin","user":"endy"}

1. Bila access token expire, kita bisa meminta refresh token sebagai berikut

curl -X POST -vu clientauthcode:123456 http://localhost:8787/oauth/token -d "client\_id=clientauthcode&grant\_type=refresh\_token&refresh\_token=436761f1-2f26-412b-ab0f-bbf2cd7459c4"

1. Authorization server akan memberikan respon access token dan refresh token yang baru

{

"access\_token":"e425cee6-7167-4eea-91c3-2706d01dab7f",

"token\_type":"bearer",

"refresh\_token":"436761f1-2f26-412b-ab0f-bbf2cd7459c4",

"expires\_in":43199,"scope":"write read"

}

1. Flow Grant Type User Password

Grant type ini biasanya digunakan bila pembuat aplikasi client sama dengan pembuat resource server. Sehingga aplikasi client diperbolehkan mengambil data username dan password langsung dari user. Contohnya: aplikasi Twitter android ingin mengakses daftar tweet untuk user tertentu. Walaupun demikian, penggunaan flow type ini tidak direkomendasikan lagi. Sebaiknya gunakan flow type authorization code atau client credentials.

Step by step:

1. Request Token terlebih dahulu:

curl -X POST -vu clientapp:123456 http://localhost:10000/oauth/token -H "Accept: application/json" -d "client\_id=clientapp&grant\_type=password&username=endy&password=123"

1. Ini akan menghasilkan token sejenis diatas

{

"access\_token":"e425cee6-7167-4eea-91c3-2706d01dab7f",

"token\_type":"bearer",

"refresh\_token":"436761f1-2f26-412b-ab0f-bbf2cd7459c4",

"expires\_in":43199,"scope":"write read"

}

1. Akses lagi dengan menggunakan token tersebut

curl -H "Authorization: Bearer e425cee6-7167-4eea-91c3-2706d01dab7f " http://localhost:10001/api/admin

1. Untuk refresh dapat di lakukan dengan

curl -X POST -vu clientauthcode:123456 http://localhost:8787/oauth/token -d "client\_id=clientauthcode&grant\_type=refresh\_token&refresh\_token=436761f1-2f26-412b-ab0f-bbf2cd7459c4"

1. Flow Grant Type Implicit

Pastikan ada jangkar untuk state :

Di state ini akan masuk ke data state degnan memproses client id ke authorization server

1. Flow Grant Type Client Credentials

Pada flow type ini, aplikasi client diberikan akses penuh terhadap resource yang diproteksi tanpa perlu meminta username dan password user. Biasanya digunakan bila aplikasi client dan aplikasi resource server dibuat oleh perusahaan yang sama.

Ask

curl -X POST -vu clientcred:123456 http://localhost:8787/oauth/token -H "Accept: application/json" -d "grant\_type=client\_credentials"

Kita akan mendapatkan response berupa access\_token dalam format JSON

{

"access\_token":"45841c94-8851-4f93-bdb1-7de9519df175",

"token\_type":"bearer",

"expires\_in":43199,"scope":"trust"

}

Gunakan access token ini dalam Authorization header untuk mengakses resource-server

curl -H "Authorization: Bearer 45841c94-8851-4f93-bdb1-7de9519df175" <http://localhost:10001/api/client>

{

"sukses":true,

"page":"client",

"user":"clientcred"

}