

Lumen : OAuth API

Christophe Brun

2016



Création d'un projet

- Création d'un projet vide nommé : test_oauth

```
composer create-project --prefer-dist laravel/lumen test_oauth
```

- Ajout du bundle (extension): lumen-passport

```
composer require dusterio/lumen-passport
```

Création d'une table des utilisateurs

- Création de la table via une migration

```
php artisan make:migration --create=users create_users_table
```

- Contenu du fichier de migration

```
<?php
....
public function up()
{
    Schema::create('users', function (Blueprint $table) {
        $table->increments('id');
        $table->string('name', 32);
        $table->string('username', 32);
        $table->string('email', 320);
        $table->string('password', 64);
        $table->timestamps();

    });
}
...
```

- migrate

```
php artisan migrate
```

Configuration de l'environnement

- Dans le fichier `.env`, création d'une clé pour l'encryption (32 caractères car on va utiliser le cipher 256)

```
APP_ENV=local
APP_DEBUG=true
APP_KEY=aofuejdbsjurm pazxhetidnkfvuajfsh
```

```
DB_CONNECTION=mysql
DB_HOST=172.17.0.2
DB_PORT=3306
DB_DATABASE=getphone
DB_USERNAME=getphone
DB_PASSWORD=getphone
```

Configuration de l'environnement 2

- Configurer l'authentification: fichier config/auth.php

```
<?php
return [
    'defaults' => [
        'guard' => 'api',
    ],
    'guards' => [
        'api' => ['driver' => 'passport',
        'provider' => 'users',
    ],
    ],
    'providers' => [
        //
        'users' => [
            'driver'=> 'eloquent',
            'model'=>\App\User::class,
        ],
    ],
    'passwords' => [
        //
    ],
];
```

Configuration de l'environnement 3

- Configurer les providers : fichier bootstrap/app.php

```
<?php

// Enable Facades
$app->withFacades();

// Enable Eloquent
$app->withEloquent();

// Enable auth middleware (shipped with Lumen)
$app->routeMiddleware([
    'auth' => App\Http\Middleware\Authenticate::class,
]);

// Finally register two service providers - original one and Lumen adapter
$app->register(Laravel\Passport\PassportServiceProvider::class);
$app->register(Dusterio\LumenPassport\PassportServiceProvider::class);
```

Ajouter l'API Token à l'utilisateur

- Fichier app/User.php

```
<?php
....
use Laravel\Passport\HasApiTokens;

class User extends Model implements AuthenticatableContract, AuthorizableContract
{
    use Authenticatable, Authorizable;
    use HasApiTokens;
    ...
}
```

Finalisation installation Passport

- En ligne de commande :

```
php artisan passport:install
```

- En base de données, un truc du genre :

```
MySQL [getphone]> select * from oauth_clients;
```

+-----+-----+-----+-----+-----+				
id	user_id	name	secret	
			redirect	personal_
+-----+-----+-----+-----+-----+				
1	NULL	Personal Access Client	http://localhost	jKl0nsilsBppxnKpZ7KB0UaM5kj0yhhettmX1yUvo
2	NULL	Password Grant Client	http://localhost	FciCiTRfXnrbT0439saAZxUFxoPgvKXxwro8Ykas

Création utilisateur de test

- Fichier database/seeds/DatabaseSeeder.php

```
<?php
use Illuminate\Database\Seeder;
use Illuminate\Support\Facades\Hash;
use App\User;

class DatabaseSeeder extends Seeder
{
    /**
     * Run the database seeds.
     *
     * @return void
     */
    public function run()
    {
        // $this->call('UsersTableSeeder');
        DB::table('users')->delete();
        User::create(array(
            'name' => 'christophe brun',
            'username' => 'cbrun',
            'email' => 'christophe@tilkee.com',
            'password' => Hash::make('cbrun'),
        ));
    }
}
```

- Insertion en base

```
php artisan db:seed
```

Test : Authentication

- Avec postman : POST /oauth/token
 - grant_type : password
 - client_id: 2 [en base de données]
 - client_secret : FciCiTRfXnrbTO439saAZxUFXoPgvKXxwro8Ykas [en base de données]
 - scope : *
 - username: christophe@tilkee.com
 - password : cbrun
- Resultat :

```
{
  "token_type": "Bearer",
  "expires_in": 3155673597,
  "access_token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9",
  "refresh_token": "Bz1ujRLq9MwiY3G+v1spNd3...9iKh1TEAF1riTKgH1b6B7jScVVXdVp0qpkYM10="
}
```

Protection d'un route

- Exemple de protection d'un route : app/Http/routes.php

```
<?php
$app->group(['prefix' => 'api/v1', 'namespace' => 'App\Http\Controllers', 'middleware' =>
    'auth:api'], function($app)
{
    $app->get('content', 'ContentController@index');
    $app->get('content/{id}', 'ContentController@getContent');
    $app->post('content', 'ContentController@createContent');
    $app->put('content/{id}', 'ContentController@updateContent');
    $app->delete('content/{id}', 'ContentController@deleteContent');
});
```

- Appel via postman : GET /api/v1/content
 - Content-type : application/json
 - Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJSU...