

Communauté en ligne pour les développeurs

Obtenir des données utilisateur à l'aide d'un jeton d'accès dans l'application client passeport laravel

J'ai créé avec succès **serveur.app** et **client.app** à l'aide de **Laravel Passport documentation**. Tout fonctionne comme prévu.

client.app de la Route:

```
Route::get('callback', function (Request $request) {
    $http = new GuzzleHttp\Client;
    $response = $http->post('http://server.app/oauth/token', [
        'form_params' => [
            'grant_type' => 'authorization_code',
            'client_id' => 3,
            'client_secret' => 'secret',
            'redirect_uri' => 'http://client.app/callback',
            'code' => $request->code
        ]
    ]);
    return json_decode((string) $response->getBody(), true)['access_token'];
});
```

Par défaut, cette route retourne **access_token** avec qui je peux faire ce que je veux.

Demande:

```
http://server.app/oauth/authorize?client_id=3&redirect_uri=http%3A%2F%2Fclient.app%2Fcallback
```

Retourne:

```
http://client.app/callback?code=access_token
```

Question:

Comment faire correctement demande à **serveur.app** avec **access_token** dans **client.app** pour obtenir par exemple **utilisateur(s) courriel(s)**.

Dois-je utiliser: `http://server.app/api/user` demande pour obtenir des données? Si oui, comment je

✓ 3 réponses

INTÉRESSANT

LES DERNIERS

DOSSIERS

Process.Kill () Accès refusé

Pourquoi autorelease est-il particulièrement dangereux / coûteux pour les applications iPhone?

Image dynamique centrale horizontale en div avec position absolue

Ajouter un index unique. SQLite3

Comment compiler les fichiers Fortran 77 sous Linux?

Lire l'adresse du point de terminaison du service WCF par nom à partir de web.config

peux faire? Si possible, veuillez écrire un code.

Merci pour toute les réponses.

[source d'information](#)

auteur [Tauras](#)

[access-token](#) [json](#) [laravel](#) [laravel-passport](#) [php](#)

28

J'ai absolument fou sur ce problème! Il a juste fait pas flipper sens pourquoi il continuait de cracher la stupide Unauthenticated d'erreur lorsque vous essayez de frapper un itinéraire tel que `/api/user` . Après beaucoup de recherches (BEAUCOUP de recherches), j'ai enfin trouvé la réponse. Si vous voyez cette fero de Laracasts, vous êtes mon héros!

Avez-vous vérifier

`app\Providers\RouteServiceProvider.php` ? dans le `mapApiRoutes()` vous pouvez définir la **middleware**. assurez-vous que son `auth:api` . si ce n'est pas le changer. retirez aussi la `auth` **middleware** de la route `api.php` fichier.

<https://laracasts.com/discuss/channels/laravel/laravel-53-passport-api-unauthenticated-in-postman-using-personal-access-tokens>

Edit:

Une fois que vous avez effectué la modification de la `app\Providers\RouteServiceProvider.php` mentionné ci-dessus, procéder à l'exemple ci-dessous.

Tout d'abord, nous avons besoin de récupérer une nouvelle `access_token` . Pour ce faire, je suis en utilisant le **mot de passe** `grant_type` (plus d'info: <https://laravel.com/docs/5.4/passport#password-grant-tokens>)

Pour récupérer une nouvelle `access_token` j'ai créé une nouvelle route sur `routes/web.php` et l'a appelé

/connect . J'ai ensuite placé le code à partir du lien ci-dessus:

```
Route::get('connect', function (Request $request) {
    $http = new GuzzleHttp\Client;
    $response = $http->post('http://example.com/oauth/token', [
        'form_params' => [
            'grant_type' => 'password',
            'client_id' => $request->client_id,
            'client_secret' => $request->client_secret,
            'username' => $request->username,
            'password' => $request->password,
            'scope' => ''
        ],
    ]);
    return json_decode((string) $response->getBody(), true);
});
```

À l'aide de [Chrome Facteur](#) vous devez:

1. Définir la méthode de GET
2. Entrez la se connecter à l'URL avec les params c'est à dire http://example.com/connect?client_id=1&client_secret=W2zogh7tiBh2jfHleYuzpViv
3. Cliquez sur le **les en-têtes de** onglet (c'est à côté de l'Autorisation), ajouter une touche de Accept et de la valeur de application/json
4. Appuyez sur le bouton Envoyer

Exemple de résultat:

```
{
  "token_type": "Bearer",
  "expires_in": 31535999,
  "access_token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsImp0aSI6ImE0M",
  "refresh_token": "Ar0Ww0g1Hjf1LpL4fK0srNUXT5v91u+CjwcE8LBvH7GJsaM0"
}
```

Mettre en évidence la `access_token` chaîne et le copier dans un éditeur de texte.

Vous aurez besoin de créer une nouvelle route dans `routes/api.php` . L'itinéraire ci-dessous simplement la sortie du courant les utilisateurs de l'API' info:

```
Route::group(['prefix' => 'user'], function() {  
Route::get('/', function() {  
return response()->json(request()->user());  
});  
});
```

Une fois que vous avez fait ci-dessus, procéder à ces modifications pour Postman:

1. Définir la méthode de GET
2. Modifier l'URL pour pointer à l'API de route c'est à dire <http://example.com/api/user>
3. Cliquez sur le **les en-têtes de** onglet de nouveau et ajouter une nouvelle clé de Authorization et de la valeur de Bearer access_token_here (remplacer access_token_here avec le jeton d'accès que vous avez copié précédemment)
4. Appuyez sur le bouton Envoyer

Exemple de sortie:

```
{  
"id": "75ced500-1544-11e7-9a8e-ad5f1aee389e",  
"name": "test test",  
"email": "test@test.com",  
"created_at": "2017-03-30 23:29:03",  
"updated_at": "2017-03-30 23:29:03"  
}
```

4

Avez-vous essayé de courir sur le FACTEUR de chrome app?

Si pas, téléchargez-le.

<https://chrome.google.com/webstore/detail/postman/fhbhl=en>

Remplir les informations.

The screenshot shows a REST client interface with the following details:

- Method: GET
- URL: http://server.app/api/user
- Authorization: Bearer eyJ0eXAI0jKV1QilCJhbGciOiJSUzI1
- Status: 200 OK
- Time: 165 ms
- Response Body (JSON):


```

1 {
2   "id": 90,
3   "name": "John Doe",
4   "email": "john.doe@gmail.com",
5   "verified": 1,
6   "email_token": "pQhSs6DEwT",
7   "account_type": null,
8   "active": "Y",
9   "created_at": "2017-02-23 00:16:02",
10  "updated_at": "2017-02-23 00:16:02"
11 }
```

Après avoir généré les données, cliquez sur le bouton de CODE à partir du haut à droite (j'ai mis en évidence avec la couleur verte).

Enfin, sélectionnez votre langue et copiez le code. Pour le PHP, j'ai choisi php cURL.

Espère que cela fonctionne.

2

Selon Laravel [la documentation](#) vous devez ajouter de la route vers le serveur d'application (routes/api.php): `$response->getBody();`

```
Route::get('/user', function () {
//authenticated user. Use User::find() to get the user from db by
return app()->request()->user();
})->middleware('auth:api');
```

Faire la demande via quuzzle:

```
$response = $client->request('GET', '/api/user', [
'headers' => [
'Accept' => 'application/json',
'Authorization' => 'Bearer '.$accessToken,
],
]);
echo $response->getBody();
```

Vous devez [être connecté](#) pour publier un commentaire.