

Cours Laravel 6 – les données – l’oubli du mot de passe

laravel.sillo.org/cours-laravel-6-les-donnees-loubli-du-mot-de-passe/

bestmomo

August 31,
2019


Un mot de passe est plus facile à oublier qu’à retenir, d’autant qu’on en est littéralement submergé ! Toute application avec une authentification doit prévoir ce genre de situation et donner la possibilité aux utilisateurs de se sortir de ce mauvais pas. Le plus simple est de leur permettre de créer un nouveau mot de passe. Laravel est équipé pour cela et nous allons le voir dans ce chapitre.

Le présent chapitre est une suite du précédent et il faut avoir mis en place ce qui concerne l’authentification.

La base de données

Avec les migrations réalisées dans le chapitre précédent il y a la table **password_resets** dans notre base :

On voit qu’on va mémoriser ici l’adresse **email**, le jeton (**token**) et le **timestamp** (par défaut les jetons sont valables pendant une heure). On trouve ces réglages dans le fichier **config/auth.php** :

#	Nom	Type
1	email 	varchar(255)
2	token	varchar(255)
3	created_at	timestamp

```
'passwords' => [  
    'users' => [  
        'provider' => 'users',  
        'table' => 'password_resets',  
        'expire' => 60,  
    ],  
],
```

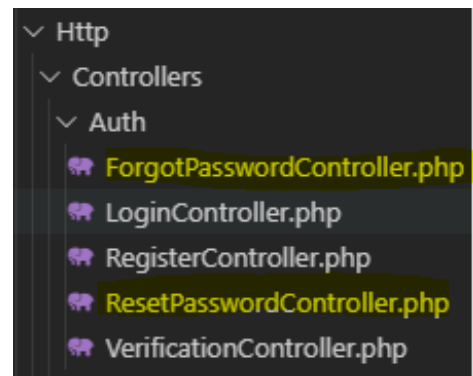
Les routes et les contrôleurs

On a 4 routes pour le renouvellement du mot de passe :

POST	password/email	password.email	App\Http\Controllers\Auth\ForgotPasswordController@sendResetLinkEmail	web,guest
GET HEAD	password/reset	password.request	App\Http\Controllers\Auth\ForgotPasswordController@showLinkRequestForm	web,guest
POST	password/reset	password.update	App\Http\Controllers\Auth\ResetPasswordController@reset	web,guest
GET HEAD	password/reset/{token}	password.reset	App\Http\Controllers\Auth\ResetPasswordController@showResetForm	web,guest

Remarquez l’application du middleware **guest** sur toutes ces routes.

On dispose également de 2 contrôleurs :



Première étape : la demande

Pour l'utilisateur la première étape va consister à cliquer sur le lien « Forgot Your Password? » prévu sur la vue de connexion :

Ce lien correspond à cette route :



```
GET|HEAD | password/reset | password.request | App\Http\Controllers\Auth\ForgotPasswordController@showLinkRequestForm | web,guest
```

On appelle la méthode **showLinkRequestForm** du contrôleur **ForgotPasswordController**. Si vous regardez dans ce contrôleur vous n'allez pas trouver cette méthode mais un trait :

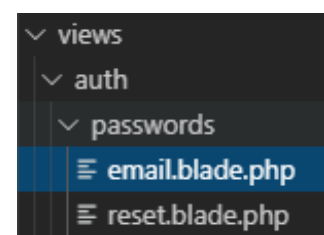
```
use SendsPasswordResetEmails;
```

L'espace de nom complet est **Illuminate\Foundation\Auth\SendsPasswordResetEmails**. Ce trait est dans le framework et voici la méthode **showLinkRequestForm** :

```
public function showLinkRequestForm()
{
    return view('auth.passwords.email');
}
```

On utilise la vue **auth.passwords.email** :

Avec un formulaire qui a cet aspect :



Reset Password

E-Mail Address

Send Password Reset Link

On demande l'adresse email à l'utilisateur. La soumission correspond à cette route :

```
POST | password/email | password.email | App\Http\Controllers\Auth\ForgotPasswordController@sendResetLinkEmail | web,guest
```

On appelle la méthode **sendResetLinkEmail** du contrôleur **ForgotPasswordController**. Cette fois encore la méthode est dans le trait **SendsPasswordResetEmails**.

Dans ce trait il y a la validation :

```
$this->validate($request, ['email' => 'required|email']);
```

On peut vérifier que ça fonctionne :

E-Mail Address

dupont@o.o

✖

Aucun utilisateur n'a été trouvé avec cette adresse email.

Si la validation passe avec succès on envoie un email à l'utilisateur. Voici le message reçu :

Laravel

Hello!

You are receiving this email because we received a password reset request for your account.

Reset Password

This password reset link will expire in 60 minutes.

If you did not request a password reset, no further action is required.

Regards,
Laravel

If you're having trouble clicking the "Reset Password" button, copy and paste the URL below into your web browser: <http://localhost/password/reset/ee88d1627463ee3fb4e7263f95e21a89db08c1f3cc76d6c03c276f7eb25d415b?email=durand%40ui.fr>

L'envoi de l'email se fait avec un système de notification dont je vous parlerai dans la troisième partie de ce cours.

Et si tout s'est bien passé on renvoie le formulaire avec un message :

```
return $response == Password::RESET_LINK_SENT
    ? $this->sendResetLinkResponse($request, $response)
    : $this->sendResetLinkFailedResponse($request, $response);
```

Voici ce que ça donne :

Reset Password

Nous vous avons envoyé par email le lien de réinitialisation du mot de passe !

E-Mail Address

Send Password Reset Link

Dans la table un jeton (token) a été généré :

email	token	created_at
durand@lui.fr	\$2y\$10\$CSV5mq1bGyfkZE5ODL6PTeD270oM95EWzrU59/Ryhr/...	2019-08-31 11:12:19

Il sera ainsi possible de vérifier la provenance de la demande.

Deuxième étape : le renouvellement

Quand on appuie sur le bouton du message ou si on clique sur le lien de secours on utilise cette route :

```
GET | HEAD | password/reset/{token} | password.reset | App\Http\Controllers\Auth\ResetPasswordController@showResetForm | web, guest
```

On appelle la méthode **showResetForm** du contrôleur **ResetPasswordController**. Si vous regardez dans ce contrôleur vous n'allez encore pas trouver cette méthode mais un trait :

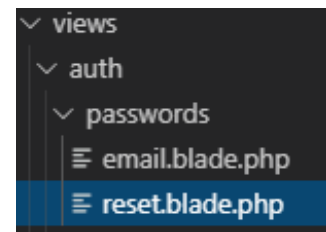
```
use ResetsPasswords;
```

L'espace de nom complet est **Illuminate\Foundation\Auth\ResetsPasswords**. Ce trait est dans le framework et voici la méthode **showResetForm** :

```
public function showResetForm(Request $request, $token = null)
{
    return view('auth.passwords.reset')->with(
        ['token' => $token, 'email' => $request->email]
    );
}
```

On utilise la vue **auth.passwords.reset** en lui transmettant le jeton (token) et l'email :

Voici l'aspect du formulaire :



Reset Password

E-Mail Address

Password

Confirm Password

Reset Password

Dans ce formulaire on retrouve le jeton (token) dans un contrôle caché :

```
<input type="hidden" name="token" value="RHD4IUZLcmKfSVyvzYzpvIZlpZIKhtImZIX6tO2V">
```

La soumission de ce formulaire utilise cette route :

```
POST | password/reset | password.update | App\Http\Controllers\Auth\ResetPasswordController@reset | web,guest
```

On appelle la méthode **reset** du contrôleur **ResetPasswordController**. Cette méthode est aussi dans le trait **ResetsPasswords**.

On commence par une validation :

```
$this->validate($request, $this->rules(), $this->validationErrorMessages());
```

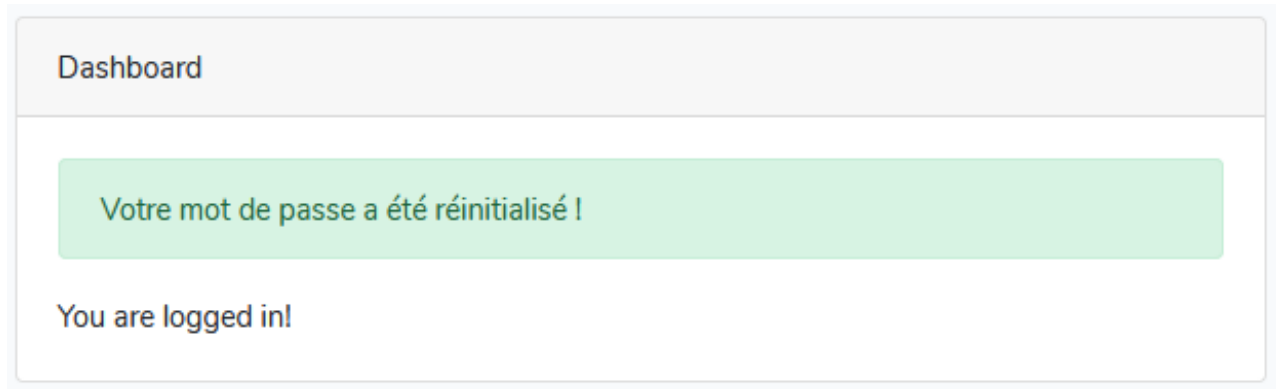
On trouve les règles (rules) dans cette méthode :

```
protected function rules()
{
    return [
        'token' => 'required',
        'email' => 'required|email',
        'password' => 'required|confirmed|min:8',
    ];
}
```

Ensuite on vérifie que l'adresse existe bien, que le jeton est bon et on procède à la mémorisation du nouveau mot de passe :

```
$response = $this->broker()->reset(
    $this->credentials($request), function ($user, $password) {
        $this->resetPassword($user, $password);
    }
);
```

Enfin si tout va bien on connecte l'utilisateur avec un message :



En résumé

- Un système complet de renouvellement du mot de passe est prévu. avec ses routes, ses contrôleurs et ses vues, en ajoutant le package **laravel/ui**.
- Le renouvellement se fait en deux étapes : une demande qui reçoit une réponse sous forme d'email sécurisé avec un jeton (token) puis une soumission du nouveau mot de passe avec un formulaire.