



3



0



16



Laravel Livewire security

**Darius Dauskurdis**

Posted on Feb 3, 2021

Laravel - Livewire security

#laravel #livewire #security

In official Laravel Livewire page <https://laravel-livewire.com> is written "Tools like Vue and React are extremely powerful, but the complexity they add to a full-stack developer's workflow is insane." and I am totally agree. Sometimes we need static pages applications, sometimes with interactive components, but the problem is that we have no time to learn new framework for small or middle size projects. Livewire is framework too, but this framework has strong relations with Laravel, so full-stack developers can create interactive applications on frontend without leaving backend. Livewire can be your best friend when you need to create CRUD pages, eshop cart, review forms with ratings or other interactive objects, but if we are talking about interaction, it means, we need to have in mind that data will be visible on user's web

**Darius Dauskurdis**

Follow

I am lithuanian senior web developer, living in Norway, with more than 12 years of experience in creating web applications based on PHP. My best friend is Laravel! :)

LOCATION

Norway

JOINED

Dec 22, 2020

More from Darius Dauskurdis

Laravel 8 - Customize profile page

#laravel #jetstream #profile #customization

Laravel 8 - Input live formatting and validation with Livewire

#livewire #formatting #validation #input

Laravel 8 - Translate countries list in different languages

#laravel #translation #locale #countries

browser. Let's say we have CRUD created with Livewire. On form submit in web browser's console we will see that we sending many variables with values. No problem if this data is just blog title or description, but what if we have companies or users with sensitive data, or field with company id which can be easily changed in web browser by changing value of inspected object. It means user can change company id and he will be able to get information about another company. Of course we need extra security on backend side, to check if this user belongs to this company or not, maybe we can implement Laravel role-based access control package - Laratrust. But what we can do in easy way is to encrypt our information. So let's say we have button in Laravel blade template, which deletes company user.

```
<button wire:click="deleteUser({{$user->id}})">Delete user</button>
```

But in web browser you will see:

```
<button wire:click="deleteUser(12)">Delete user</button>
```

And this 12 can be changed to another value. Luckily blade supports Laravel encryption service like "Crypt". So in blade file we can add:

```
<button wire:click="deleteUser('{{Crypt::encrypt($user->id)}}')">Delete user</button>
```

So in web browser we will see something like this:

```
<button wire:click="deleteUser('t0JpdiI6InZQcEpmSW1NdkNjK3pyd1lWM2...')">Delete user</button>
```

Don't forget to add single quotes for function parameter, because after encryption this parameter becomes string

encryption this parameter becomes string.

In our Livewire component class we can add "deleteUser" method.

```
public function deleteUser($id)
{
    try {
        $id = Crypt::decrypt($id);

        ...
        Remove user from company by decrypted id
        ...

    } catch (DecryptException $e) {
        ...
        Return error
        ...
    }
}
```

Don't forget to add "DecryptException" contract and "Crypt" facade

```
use Illuminate\Contracts\Encryption\DecryptException;
use Illuminate\Support\Facades\Crypt;
```

Another important thing is what kind of information we have in our component's class public variables. Maybe it is not the best solution to loop Model items, but it can be very helpful to filter information we don't want to show.

```
$this->company_users = [];
foreach ($company->users as $key => $company_user) {
    $temp_user = [
        'id' => Crypt::encrypt($company_user->id),
        'name' => $company_user->name,
        'surname' => $company_user->surname
    ];
}
```

```
},  
$this->company_users[] = $temp_user;  
}
```

So it means we will add to `$this->company_users` variable only information we can show on web browser and user id will be encrypted.

Discussion (-20)

[Subscribe](#)

Add to the discussion

[Code of Conduct](#) • [Report abuse](#)

Read next



New Things Added - Laravel 9.16 Released

Morcos Gad - Jun 9



Laravel dd() alternative in cakephp

Chidiebere Chukwudi - Jun 9



Top 10 Reasons Why Laravel 9 Is The Most Popular PHP Framework-Codexashish

Ashish - May 22



The Benefits of Custom CRM Software Development

sumyya khan - Jun 9

[DEV Community](#) — A constructive and inclusive social network for software developers. With you every step of your journey.

Built on [Forem](#) — the [open source](#) software that powers [DEV](#) and other inclusive communities.

Made with love and [Ruby on Rails](#). DEV Community © 2016 - 2022.

