

## Logout from every devices except the currently logged in one in Laravel

Amit Merchant · May 5, 2019 · [Laravel](#)

Every robust web app has this feature where you're provided with a setting through which you can choose to logout from all the devices you've been logged in from previously, except the current one. This is certainly a nice security feature that you can provide to your user without them even asking for it and if your app is built on top of Laravel 5.6, this comes out-of-the-box.

Here's how you can accomplish this. First of all, you need to uncomment the line `\Illuminate\Session\Middleware\AuthenticateSession::class` from `$middlewareGroups` property in `app/Http/Kernel.php` file because this is the middleware that manages the user sessions in Laravel.

```
/**
 * The application's route middleware groups.
 *
 * @var array
 */
protected $middlewareGroups = [
    'web' => [
        ...
        \Illuminate\Session\Middleware\AuthenticateSession::class,
        ...
    ],
    ...
];
```

Next, you can use `logoutOtherDevices` method on the `Auth` facade which takes user's password as its only argument. You can collect the password through an input form.

```
use Illuminate\Support\Facades\Auth;

Auth::logoutOtherDevices($password);
```

This is it! This will get your job done of logging out the user from all its active logged in devices except the the current one.

### Behind the scenes

Now, Let's take a look at the `logoutOtherDevices` method's implementation which you can find in `\Illuminate\Auth\SessionGuard` class.

```
/**
 * Invalidate other sessions for the current user.
 *
 * The application must be using the AuthenticateSession middleware.
 *
 * @param string $password
 * @param string $attribute
 * @return bool|null
 */
public function logoutOtherDevices($password, $attribute = 'password')
{
    if (! $this->user()) {
        return;
    }

    $result = tap($this->user()->forceFill([
        $attribute => Hash::make($password),
    ])->save());

    $this->queueRecallerCookie($this->user());

    $this->fireOtherDeviceLogoutEvent($this->user());

    return $result;
}
```

Let's break down the function in chunks to understand what it's doing.

As you can see, the method accepts two arguments. The first one being the password of the logged in user and the second one being an optional one which is an attribute called 'password'.

```
if (! $this->user()) {
    return;
}
```

The method first checks if the user is logged in or not. If it's not logged in, it returns immediately and continue execution if otherwise.

```
$result = tap($this->user()->forceFill([
    $attribute => Hash::make($password),
]))->save();
```

The `forceFill` method here effectively ignores `$fillable` array on the `User` model and will update the model with the array key-value pairs that you pass to it which is in our case is 'password'. Then the result will be caught by `tap` method which is then used to save it onto the `User` model.

```
$this->queueRecallerCookie($this->user());
```

It will then create a cookie for the existing session using `queueRecallerCookie` method and queue it into the cookie jar.

```
$this->fireOtherDeviceLogoutEvent($this->user());
```

This will fire the other device logout event if the dispatcher is set and this will invalidate user's other sessions entirely, meaning they will be "logged out" of all guards they were previously authenticated by. You can check this by looking at the `fireOtherDeviceLogoutEvent` method's implementation.

```
/**
 * Fire the other device logout event if the dispatcher is set.
 *
 * @param \Illuminate\Contracts\Auth\Authenticatable $user
 * @return void
 */
protected function fireOtherDeviceLogoutEvent($user)
{
    if (isset($this->events)) {
        $this->events->dispatch(new Events\OtherDeviceLogout(
            $this->name, $user
        ));
    }
}
```

Learn the fundamentals of **PHP 8 (and 8.1)**, the latest version of PHP, and how to use it today with my new book **PHP 8 in a Nutshell**. It's a no-fluff and easy-to-read guide to the latest features and nitty-gritty details of PHP 8. So, if you're looking for a quick and easy way to PHP 8, this is the book for you.

Grab your copy now!

Read a sample

» Share: [Twitter](#), [Facebook](#), [Hacker News](#)

Beep! Beep! I'm also running a [YouTube channel](#) which I hope you're going to love!



Hi there! I'm [Amit](#). I write articles about all things web development. If you like what I write and want me to continue doing the same, I would like you buy me some coffees. I'd highly appreciate that. Cheers!

 Buy me a coffee

 Join my newsletter

### Comments?

More on similar topics

Browse articles by category

Discard Eloquent model changes in Laravel 9.x

Why Developers Choose Cloudways For Laravel Hosting

Accessing Laravel documentation from the terminal

The artisan about command in Laravel 9.x



**Focus on Laravel**  
**Not on Laravel**  
**Hosting**

GET STARTED

PROMO CODE: **AMIT**  
WITH 20\$ FREE  
HOSTING CREDIT

Download my eBook



Newsletter

Get the latest articles delivered right to your inbox!  
(Read a [sample](#).)

Subscribe via email

Subscribe

By subscribing, you agree with Reue's [Terms of Service](#) and [Privacy Policy](#).

Follow me everywhere



MY TINY YOUTUBE CHANNEL



More in "Laravel"

Discard Eloquent model changes in Laravel 9.x

Why Developers Choose Cloudways For Laravel Hosting

Accessing Laravel documentation from the terminal

The artisan about command in Laravel 9.x

New invokable rule objects in Laravel 9.x

Recently Published

Toggleing state of the checkbox based on the viewport width in JavaScript **NEW**

Discard Eloquent model changes in Laravel 9.x

Conditionally spreading objects in JavaScript

Converting PNG images to WebP using PHP

Running shell commands is dangerously simple in PHP

Top Categories

LaravelPHPJavaScript

MiscellaneousCSS

ReactGitJekyll

VS Code

[Home](#) [About](#) [Contact](#)  
[Categories](#) [Guestbook](#)



[About](#) [Contact](#) [Categories](#) [Guestbook](#)