Práctica 3. Subida de archivos

Taller de Desarrollo Seguro con PHP

Mayo 2019

1. Objetivo de la práctica

El participante deberá proteger un componente de código PHP que permite la subida de archivos de imágenes al servidor. Deberá aplicar técnicas de programación segura para:

- Validar la extensión y tipo de archivo
- Validación del tamaño del archivo
- Recodificación de imágenes utilizando la librería GD
- Sanear el nombre de archivos o renombrar los archivos.

2. Validar la extensión y tipo de archivo

La validación de los tipos de archivos no es un proceso trivial. Existen muchas maneras en que se puede falsificar tanto la extensión como el tipo mime.

El tipo que llega en el arreglo $_FILES no es confiable ya que es un tipo asignado por el navegador. En su lugar se recomienda utilizar otros mecanismos para validar que el archivo sea del tipo adecuado. En el caso de las imágenes, podemos utilizar la librería GD para obtener información de la imagen utilizando la función getimagesize()

Listing 1: Función getimagesize()

```php
$imagesize = getimagesize($uploaded_tmp);
```

También se recomienda valida la extensión del archivo:

Listing 2: Obtener la extensión del archivo.

```php
$ext = substr($uploaded_name, strrpos($uploaded_name, '.') + 1);
```

3. Validar el tamaño de archivo

Para validar el tamaño del archivo se puede utilizar el valor que se encuentra dentro de $_FILES, el tamaño está indicado en bytes:

```php
$_FILES[ 'archivo' ][ 'size' ];
```

Es importante recordar que aunque se defina en el formulario un campo oculto con el tamaño máximo permitido, esto no representa una validación, ya que al estar definido en el cliente se considera inseguro.

4. Recodificación de imágenes utilizando la librería GD

En el caso de las imágenes lo que se recomienda es recodificarlas, es decir, volverlas a crear utilizando librerías como GD. Esto lo que realiza es que quita cualquier información que no es propia de una imagen, incluyendo los metadatos del archivo.

Listing 3: Recodificación de imágenes

```php
if ($uploaded_type == 'image/jpeg') {
   $img = imagecreatefromjpeg($uploaded_tmp);
   imagejpeg($img, $temp_file, 100);
} else {
   $img = imagecreatefrompng($uploaded_tmp);
   imagepng($img, $temp_file, 9);
}
imagedestroy($img);
```

5. Sanear el nombre de archivos o renombrar los archivos.

Para evitar problemas por caracteres inválidos o peligrosos como ../, se recomienda que los archivos que se suban se renombren utilizando alguna nomenclatura definida. Una opción es dar un nombre aleatorio al archivo, por ejemplo:

Listing 4: Generar nombre aleatorio de archivo

```php
$target_file   =  md5(uniqid() . $uploaded_name) . '.' . $ext;
```