

*php*

# uploading files

**PHP web development 2019/2020**

Milena Tomova  
Vratsa Software

<https://vratsasoftware.com/>

# Table of Contents

1. the HTML
2. \$\_FILES
3. Saving uploaded file
4. Handling file upload
  - validating files through \$\_FILES elements
  - using php functions
5. Security
6. Configuring file uploads



# the HTML



```
<form method="post" action="" enctype="multipart/form-data">
  <div class="form-group">
    <label class="control-label">Enter product name</label>
    <input type="text" name="unit_name" class="form-control">
  </div>
  <div class="form-group">
    <label for="product-image">Product image</label>
    <input type="file" id="product-image"
name="product_image">
  </div>
  <button type="submit" class="btn btn-default">Save</button>
</form>
```



**\$\_FILES**

# \$\_FILES superglobal variable

\$\_FILES

## \$\_FILES

This is a 'superglobal', or automatic global, variable.

This simply means that it is available in all scopes throughout a script.

There is no need to do **global \$variable;** to access it within functions or methods.

*same like \$\_POST, \$\_GET superglobals*



After form submission all the uploaded files and their specifics  
**are stored in `$_FILES` superglobal.**

Do not look for them in `$_POST`  
superglobal!

After form submission  
submitted data is saved in two  
superglobals

- files data goes to **\$\_FILES superglobal.**
- rest of the data is in \$\_POST superglobal.



- \$\_FILES splits file`s data in predefined elements with indexes -
  - ["name"] - *filename*
  - ["type"] - *filetype*
  - ["tmp\_name"] - temprary name used to store file in the temporary file upload directory
  - ["error"] - index of error occured during upload
  - ["size"] - file size in bytes

[see detailed info](#)



**saving  
uploaded  
file**



# Saving uploaded file

after form`s submission, the file /its content/ is stored in a temporary directory, with a temporary name, waiting to be moved to the desired server`s or project`s folder



# Saving uploaded file

## *Step 1 - moving the file to desired location*

```
$uploaddir = '../uploads/';  
$uploadfile = $uploaddir . basename($_FILES['product_image']['name']);  
  
if (move_uploaded_file($_FILES['product_image']['tmp_name'], $uploadfile)) {  
    echo "File is valid, and was successfully uploaded.\n";  
} else {  
    echo "Possible file upload attack!\n";  
}
```

## Step 2 - saving file info in data base

- *filename - and repetitive data*
  - *if all files are saved in the same folder - save folder name as part of the data about the file*

```
$product_name = $_POST['product_name'];
```

```
$image = $uploadfile;
```

```
$insert_query = "INSERT INTO `products` (`product_name`, `image`) VALUES ('$product_name', '$uploadfile')";
```





# handling file uploads



# Handling file uploads

- *check for file upload errors*

```
if(!isset($_FILES['product_image']['error'])){  
    // code to execute if no errors occur  
}
```

*List of error values and messages explained*

# Handling file uploads

- *check filesize - always set limit of allowed upload filesize*

```
if ($_FILES['product_image']['size'] > 1000000) {  
    //code to execute if file exceeds the allowed limit  
}
```



# Handling file uploads

- Check MIME Type - **DO NOT TRUST `$_FILES['product_image']['mime']` VALUE**
- make your own mime type validation solutions
- [some useful examples here](#)



# Handling file uploads

- *Avoid using `$_FILES['product_image']['name']` as filename for saving the uploaded file on project's server*
- *DO NOT USE `$_FILES['upfile']['name']` WITHOUT ANY VALIDATION !!*
- *You should name the file uniquely.*



# **Securing file uploads**



## What are the File Upload Risks?

*some of them are*

- Attacking your infrastructure:
  - Overwriting an existing file
  - Malicious content
- Very large file upload



## How to Prevent File Upload Attacks

- Only allow specific file types
- Verify file types
- Remove possible embedded threats
- Set a maximum name length and maximum file size
- Randomize uploaded file names
- Store uploaded files outside web root folder
- Use simple error messages



# **configuring file uploads**



# Configuring file uploads

*File uploads directives in php.ini*

file\_uploads

upload\_max\_filesize

upload\_tmp\_dir

post\_max\_size

max\_input\_time



# Questions?



Гнездото  
Coworking

Цялостен  
курс по  
програми  
ране

Дизайн  
курс

Курс по  
дигит.  
маркетинг

MindHub



# Partners



**Telerik  
Academy**



**MindHub**

**ПРОМЯНАТА**

# Trainings @ Vratsa Software



- Vratsa Software – High-Quality Education, Profession and Jobs
  - [www.vratsasoftware.com](http://www.vratsasoftware.com)
- The Nest Coworking
  - [www.nest.bg](http://www.nest.bg)
- Vratsa Software @ Facebook
  - [www.fb.com/VratsaSoftware](http://www.fb.com/VratsaSoftware)
- Slack Channel
  - [www.vso.slack.com](http://www.vso.slack.com)

