Uploading Files

It may be useful to allow your users to upload images, sounds, or other types of files. These files will then be stored on your server and you can make them available throughout the website.

With allowing file uploads, you need to take care to not allow malicious files to be uploaded, such as php scripts or otherwise. When a file is uploaded, it is stored in the /tmp/ directory with a random (i.e., hashed) name and you will want to check its size, type, error codes, etc. before moving it to the directory where you want to store these files.

The form (XHTML)

In addition to having the form have the method "post", you will also want to add an **enctype:**

```
<form method="post" enctype="multipart/form-data">
```

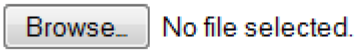
Enctype stands for "encoded type", and it needs to be set to "multipart/form-data" when you are handling file uploads.

Once the submit button is hit, you will end up with a $_POST array and a separate $_FILES array to work with.

Inside of the form, you will have your standard form elements, and to upload files you will use an input element of type "file".

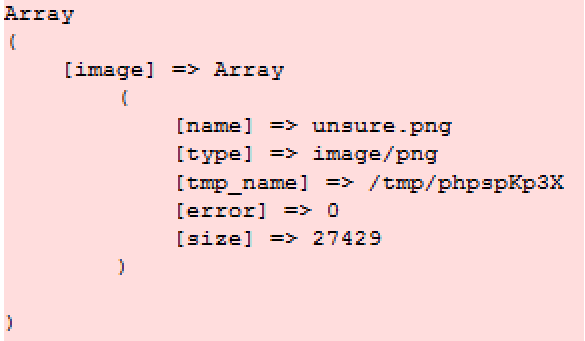
```
<form method="post" enctype="multipart/form-data" action="gallery.php5">
      <input type="file" name="image" />
      <br/>
      <input type="submit" name="new-image" value="Add Image"/>
</form>
```

This will show up as a "Browse" button with a field for the file name:



Validating and Moving the file (PHP)

Once a file is selected and the submit button is pressed, your $_FILES array will look something like:



```
Array
(
    [image] => Array
        (
            [name] => unsure.png
            [type] => image/png
            [tmp_name] => /tmp/phpspKp3X
            [error] => 0
            [size] => 27429
        )

)
```

You will want to make sure that:

1. The “type” is a file type you allow on the server
2. The “name” includes the correct extension to use on your server (no images named image.php!)
3. The “error” code is 0
4. The “size” is both below the set max file size (in a config file on the server) and below a file size that you choose to be maximum.

If any of these fail, you should log and display some error messages.

If they all pass, then you will copy the file from “tmp_name” to a path that you want, using the file's original “name”. We can copy this file using PHP's **move_uploaded_file(...)** function:

```
$destination = "images/" . $file["name"];
move_uploaded_file( $file["tmp_name"], $destination );
```

Note that move_uploaded_file may return **false** if there is an error, so this is another place you will need to have error checking and an error message display.

Once this is done, refresh that directory in your FTP client to see if the images got uploaded.