Passage walkthrough

Index

List of pictures

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:

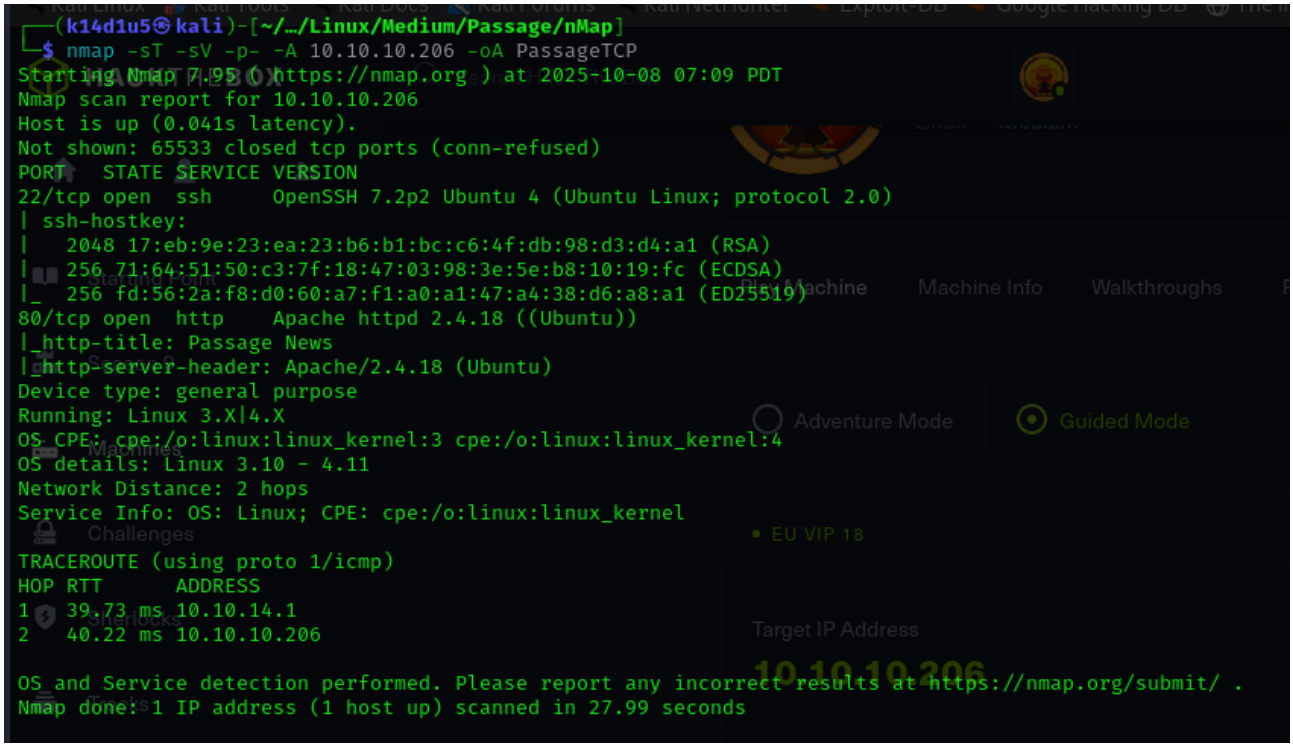

Figure 1 - nMap scan results

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 Linuz 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:

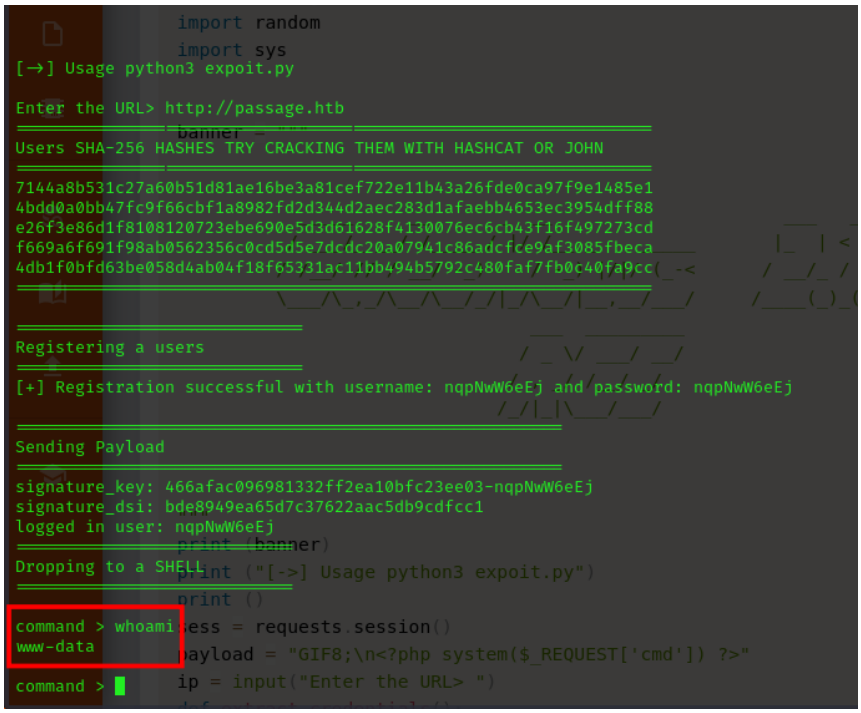

Figure 2 - First shell on the target

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

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:

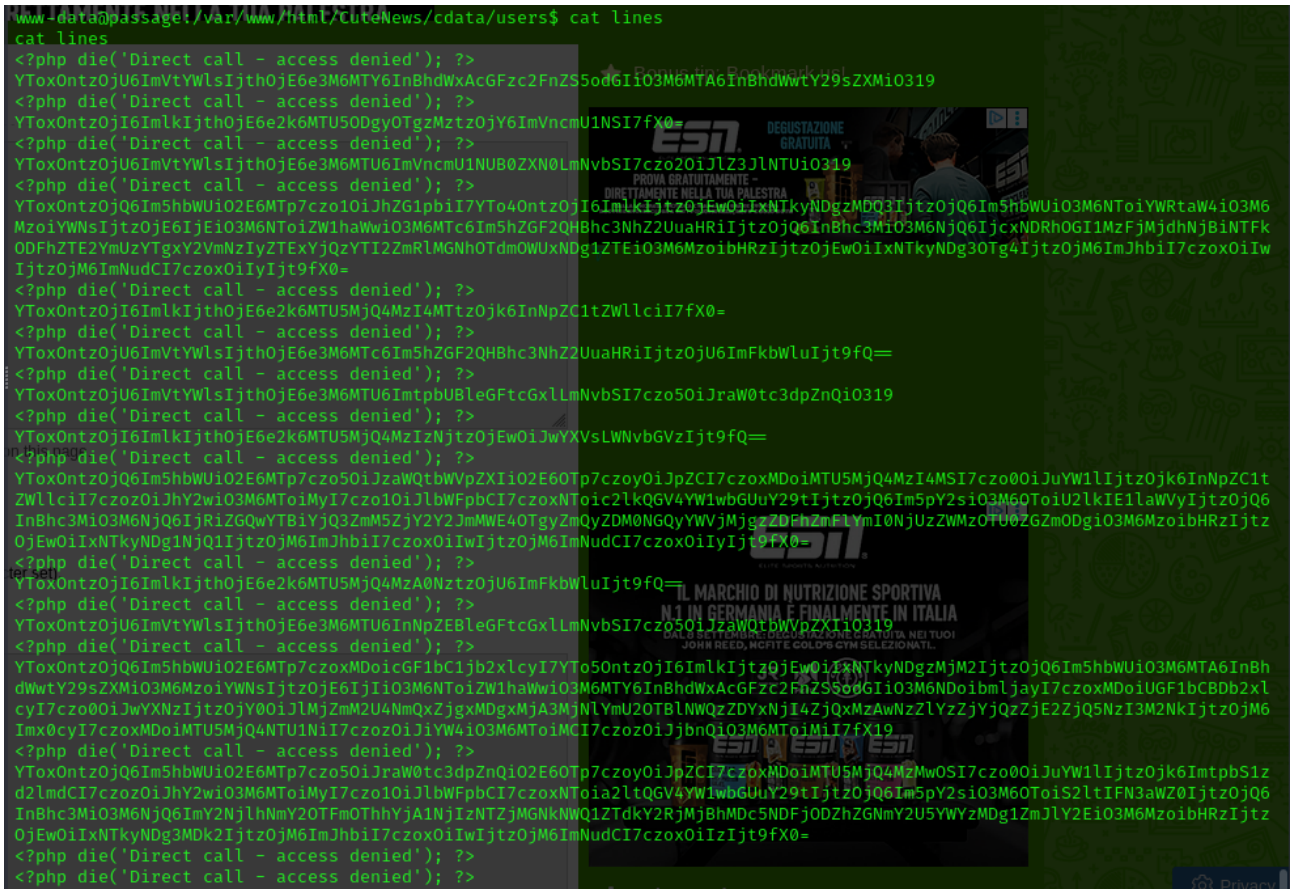

Figure 4 - Interesting file

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:

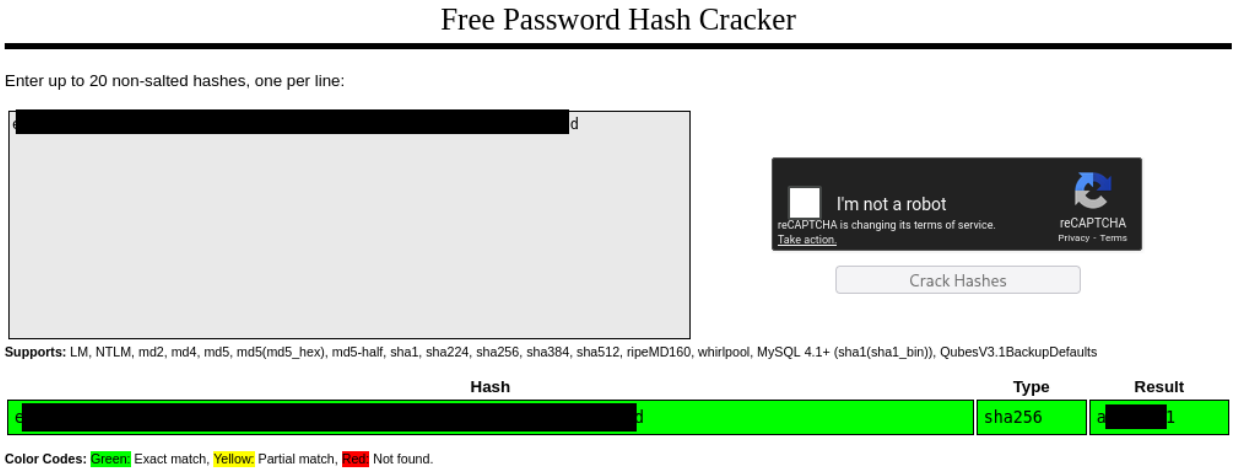

Figure 5 - Password cracked

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

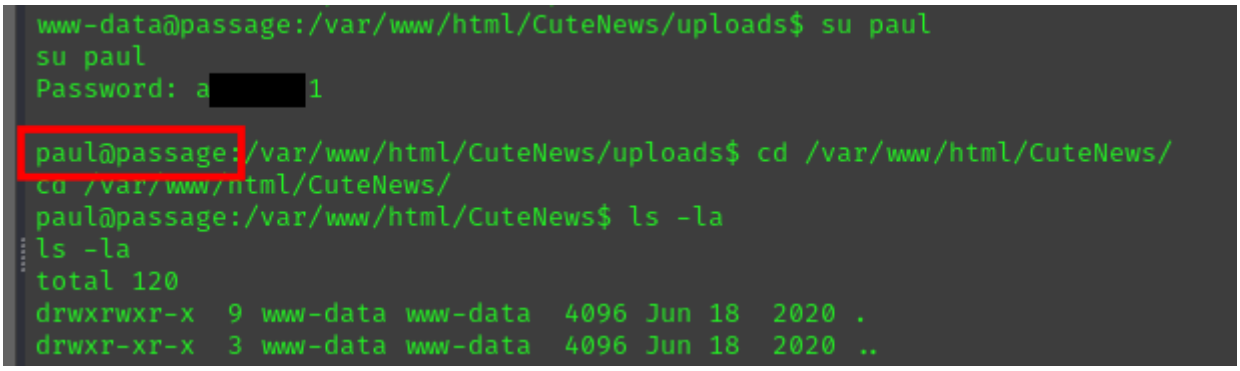

Figure 6 - Lateral movement

Finally, I was able to retrieve di user flag:

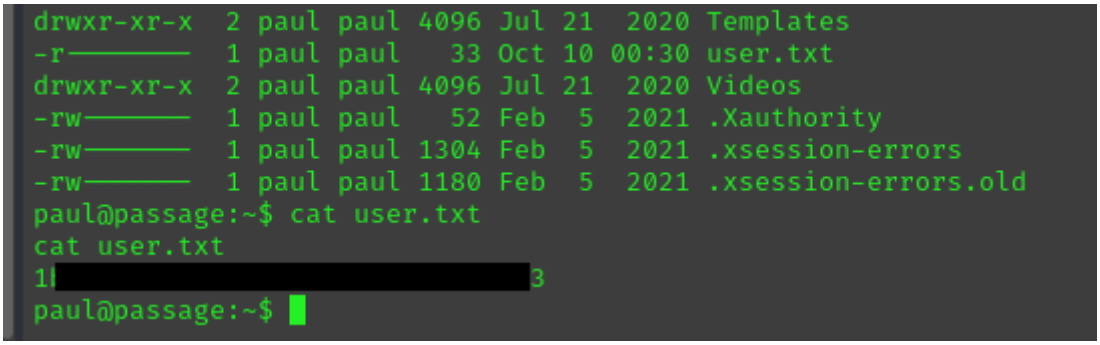

Figure 7 - User flag

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:

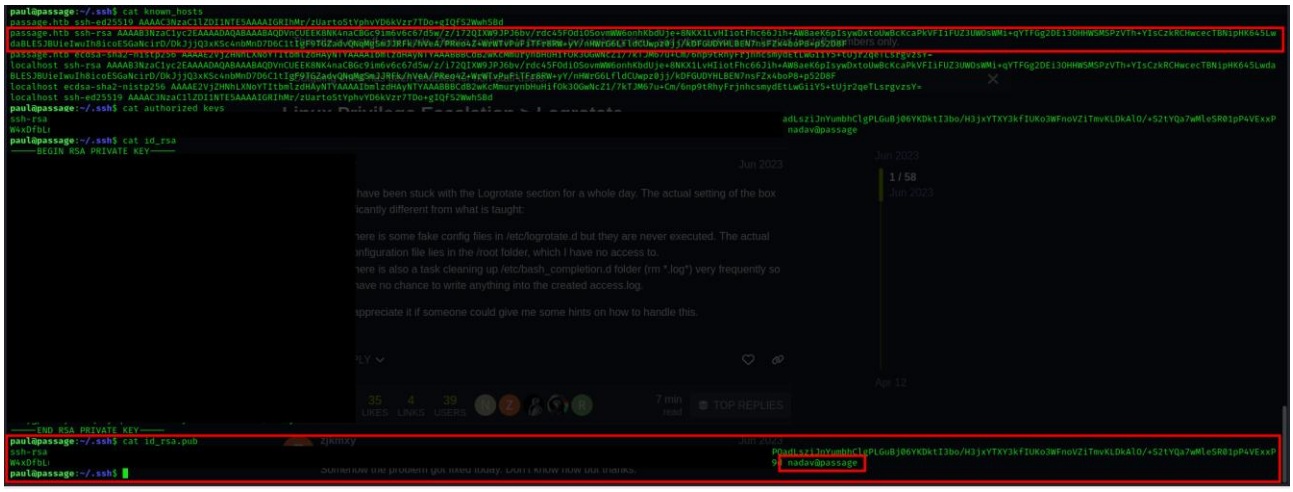

Figure 8 - Nadav SSH keys

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

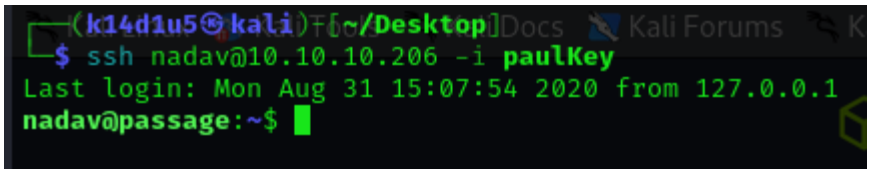

Figure 9 - Login as Nadav user

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:

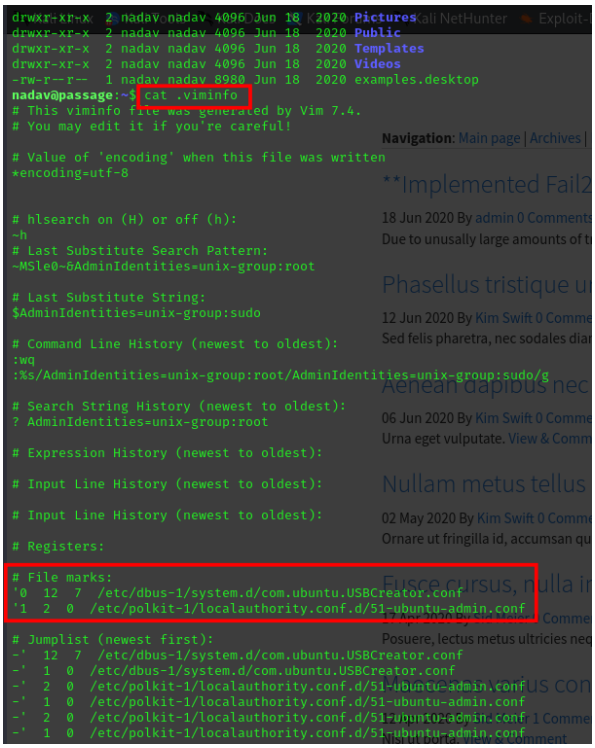

Figure 10 - Modified files

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:

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:

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.