**Mirai walkthrough**

# **Index**

[Index 1](#_Toc177651237)

[List of pictures 1](#_Toc177651238)

[Disclaimer 2](#_Toc177651239)

[Reconnaissance 2](#_Toc177651240)

[Initial foothold 2](#_Toc177651241)

[User flag 3](#_Toc177651242)

[Privilege escalation 4](#_Toc177651243)

[Personal comments 5](#_Toc177651244)

# **List of pictures**

[Figure 1 - nMap scan results 2](#_Toc177651262)

[Figure 2 - ffuf scan results 3](#_Toc177651263)

[Figure 3 - SSH login 3](#_Toc177651264)

[Figure 4 - Privilege escalation 4](#_Toc177651265)

[Figure 5 - Devices mounted 4](#_Toc177651266)

[Figure 6 - Note about file deleting 4](#_Toc177651267)

[Figure 7 - Rough file recover 5](#_Toc177651268)

[Figure 8 - Root flag 5](#_Toc177651269)

# **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’re willing and curious to know and learn about Ethical Hacking, Security and Penetration Testing.

Just as note: 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:

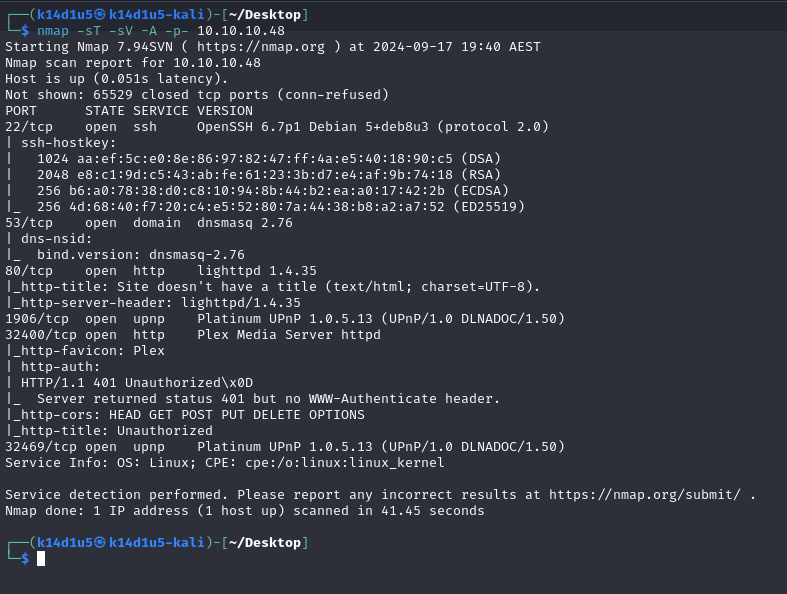


Figure 1 - nMap scan results

Open ports are 22, 53, 80, 1906, 32400 and 32469. NMap identified SSH service on port 22, DNS service on port 53, a web application server on port 80, Platinum UPnP service on ports 1906 and 32469 and Plex media server service on port 32400. Also, nMap identified Linux as system operating, but it didn’t provide more details about it.

# **Initial foothold**

The first thing I did was trying to access to web application server and Plex Media Server services I found via browser. The web application running on port 80 has a blank page. The Plex service required credentials to log in. At this point I started to search some hidden content on web application running on port 80, using ffuf tool:

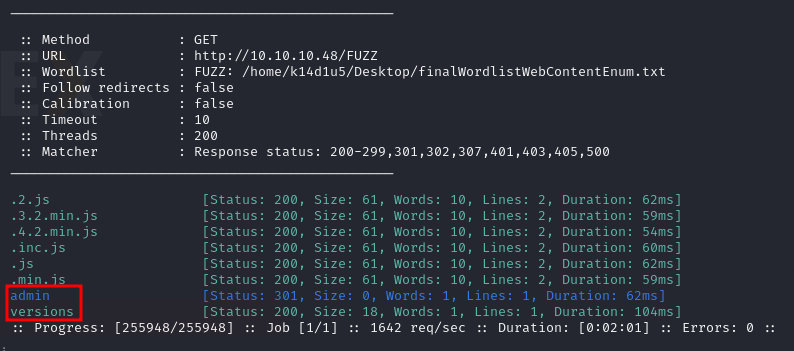


Figure 2 - ffuf scan results

I founded the admin and versions path. The latter one allowed me to download a file which contains versions numbers. The former one allowed me to access to a new dashboard.

# **User flag**

On the admin dashboard I found out that the application running there was **Pi-hole**. Looking for it on the Internet, I learnt this application has a distribution for Raspberry devices. At this point, I kept to search some interesting information on the target. For example, I tried to analyzing the DNS service. It let me to find a new domain. I added this domain to the file on my Kali and I browsed to it. However, it informed me that the navigation is blocked. Also, I found some interesting CVEs, but any of them fitted to this box scenario, at this stage at least. After a lot of time, I tried to use the Raspberry default credential on the login pages. However, them doesn’t work. So, my last chance and idea was to try these credentials to log in via SSH:

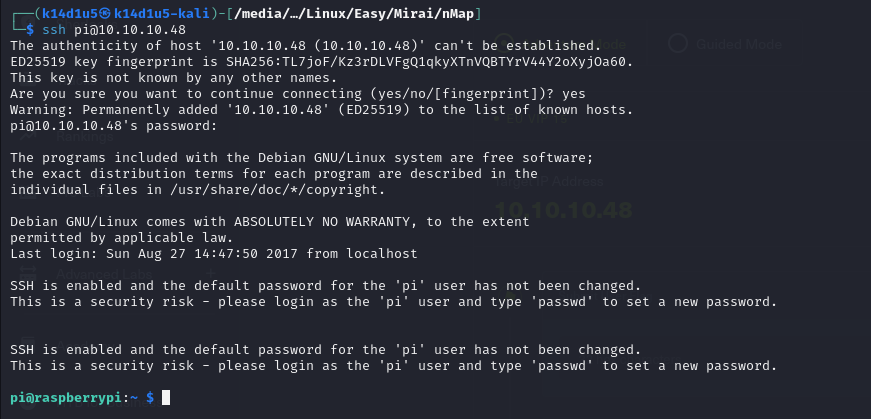


Figure 3 - SSH login

I was very surprised that them worked. Anyway, I am on the target machine and more surprising, this user has the user flag I retrieved (I forgot a screenshot about the user flag).

# **Privilege escalation**

At this point I need to escalate my privileges. The first thing I usually check are the sudoers. It was incredible that this was the right way to escalate privileges. In fact, the pi user had the following configuration:

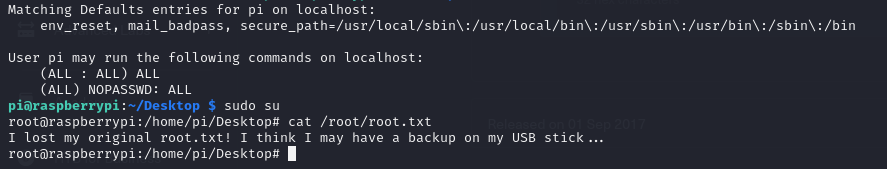


Figure 4 - Privilege escalation

As you can see in the previous picture, I just needed to run the command to became root. However, I was not able to retrieve the root flag yet. The root.txt existed, but it informed me that the file was deleted and can exist a copy on a USB stick. Due to this clue, I checked the devices mounted:

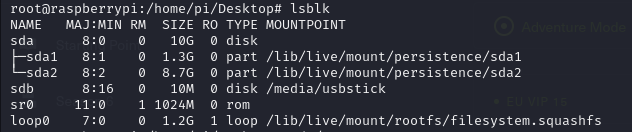


Figure 5 - Devices mounted

As I expected, one of these devices is a USB stick. So, I simply checked in it. However, it didn’t contain the root flag, but I found a new note:

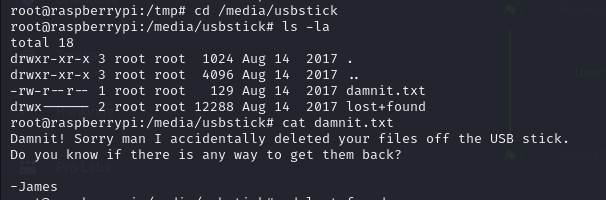


Figure 6 - Note about file deleting

I didn’t still find the root flag. The last note I found made me think about I have to recover the root flag from the USB stick. I looked on the Internet some way to do it and several of them require a no built-in tool. So, this searching took a bit of time. Finally, I found a rough way to do it:



Figure 7 - Rough file recover

So, looking for the flag in all output, I finally found the root flag:

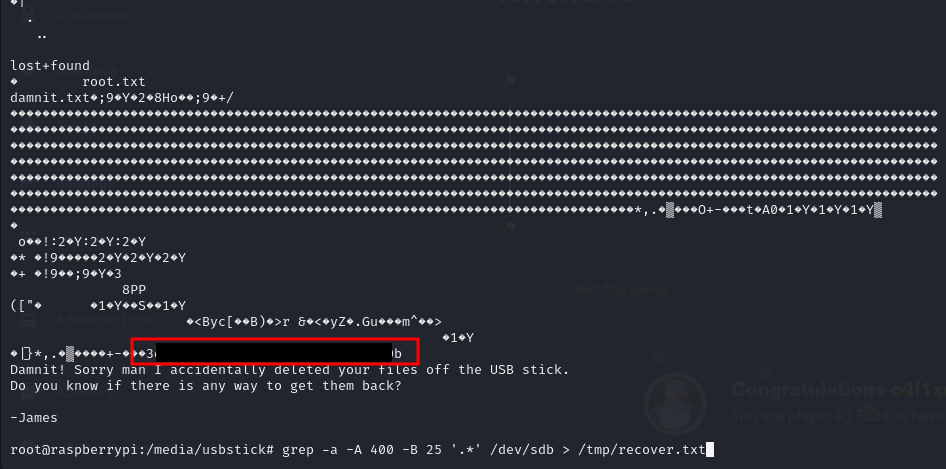


Figure 8 - Root flag

# **Personal comments**

I consider this box very funny and basic. However, I liked the searching of the root flag, similar to a treasure hunting. However, since I can’t install anything on a box, it was challenging to find the right command to use to recover files from the USB stick. Also, I understand the password reuse matter, but, at least in this case, it is not easy to think that the Raspberry default password can be used to log in via SSH. In fact, there is any clue to hypnotize that a user named pi, or relative to Pi-hole or Raspberry, exist. In conclusion, I rated this box as easy.