**Seal walkthrough**

# **Index**

[Index 1](#_Toc199175042)

[List of pictures 1](#_Toc199175043)

[Disclaimer 2](#_Toc199175044)

[Reconnaissance 2](#_Toc199175045)

[Initial foothold 3](#_Toc199175046)

[User flag 4](#_Toc199175047)

[Privilege escalation 6](#_Toc199175048)

[Personal comments 7](#_Toc199175049)

[References 7](#_Toc199175050)

# **List of pictures**

[Figure 1 - nMap scan results 1 2](#_Toc199175030)

[Figure 2 - nMap scan results 2 2](#_Toc199175031)

[Figure 3 - nMap scan results 3 3](#_Toc199175032)

[Figure 4 - Credentials found 3](#_Toc199175033)

[Figure 5 - Tomcat manager GUI 4](#_Toc199175034)

[Figure 6 - First user shell 4](#_Toc199175035)

[Figure 7 - Dangerous configuration 4](#_Toc199175036)

[Figure 8 - Extracting Luis SSH private key 5](#_Toc199175037)

[Figure 9 - Backup copleted check 5](#_Toc199175038)

[Figure 10 - User shell and flag 6](#_Toc199175039)

[Figure 11 - Malicious reverse shell 6](#_Toc199175040)

[Figure 12 - Root shell and flag 7](#_Toc199175041)

# **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:

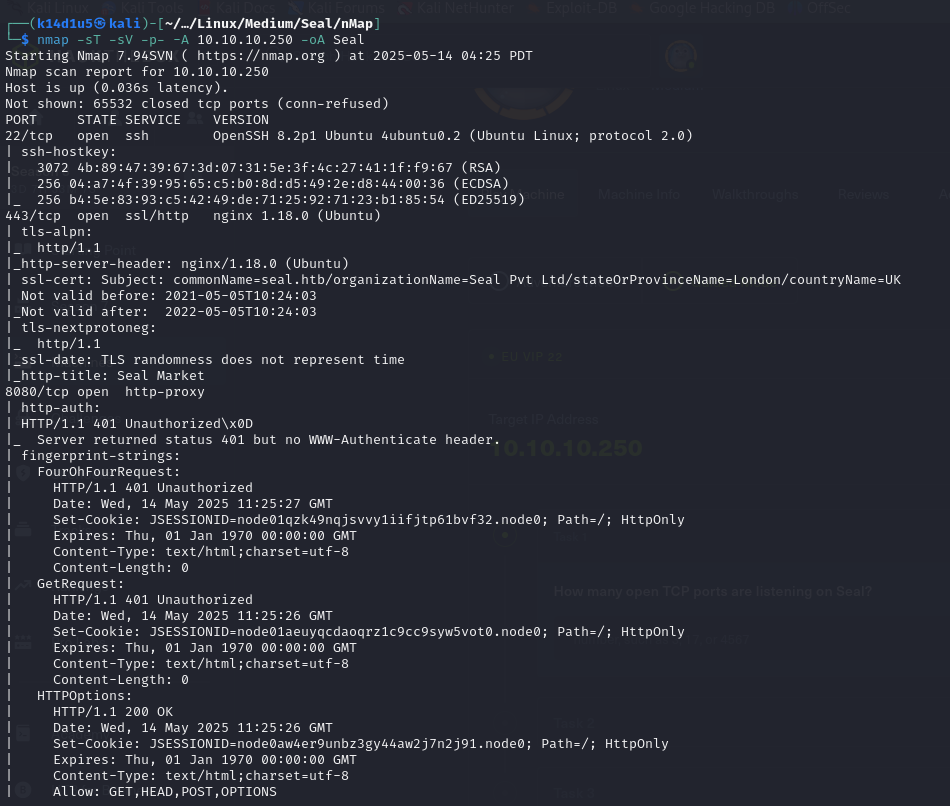


Figure - nMap scan results 1

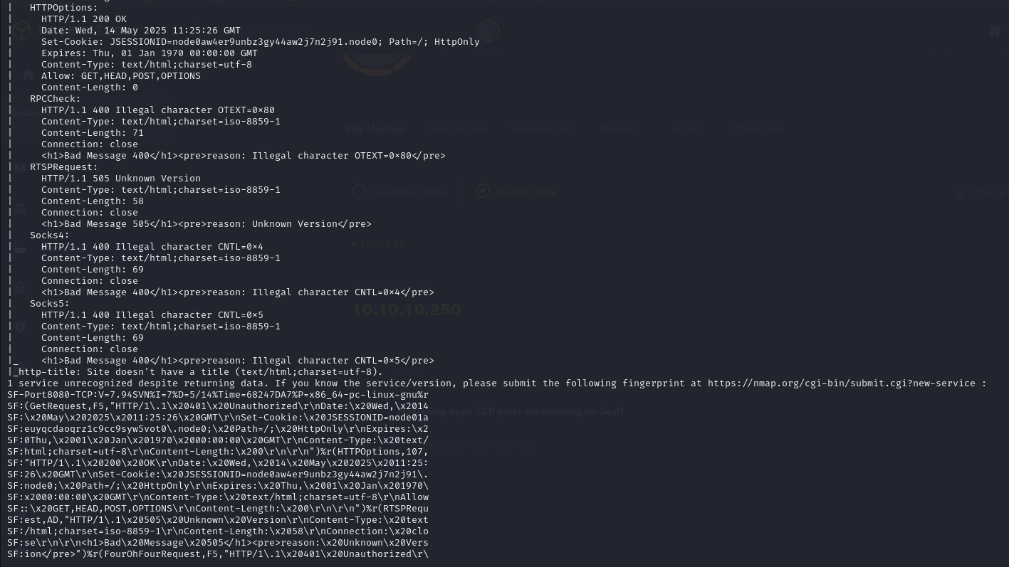


Figure - nMap scan results 2

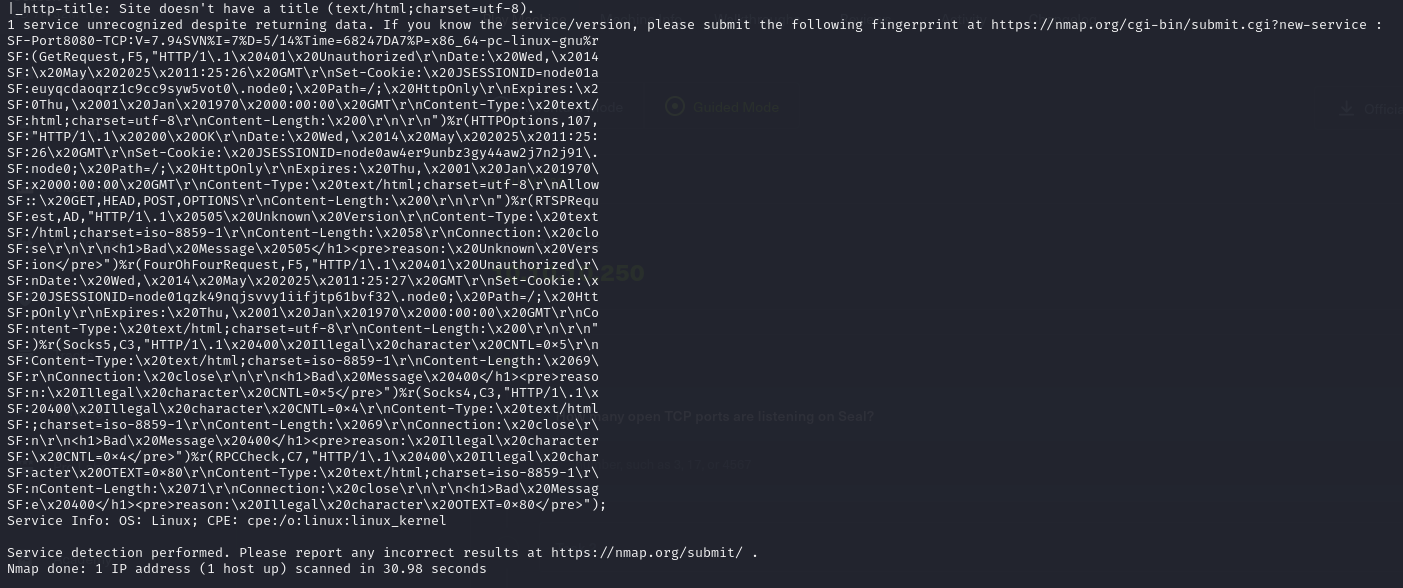


Figure 3 - nMap scan results 3

Open ports are 22, 443 and 8080. Therefore, SSH service was enabled. In addition, a web application was running on port 443 and a reverse proxy was running on port 8080. Lastly, nMap recognized Linuz as operative system, maybe Ubuntu, but it didn’t provide any further information about it.

# **Initial foothold**

As first task, I run FFUF to find some hidden web content on both ports. However, I didn’t find nothing of very important. Therefore, I browsed to the web application on port 8080. This application was gutbucket and I was able to register a new user. I did it and I logged in. At this point, I explored the git repository in which I found server, proxy and application configuration files. However, in the first analysis I didn’t find anything useful. During the investigation, I noted I was able to access to old commits. In this way, I found credentials to access to the Tomcat manager GUI:

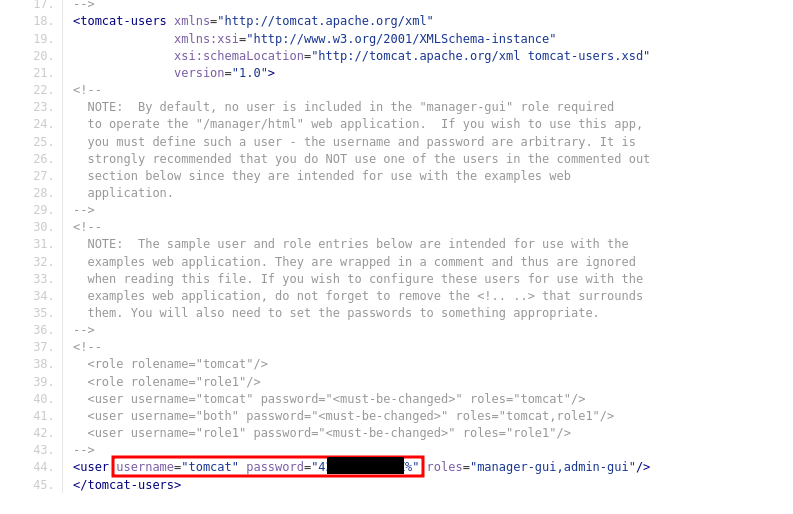


Figure 4 - Credentials found

# **User flag**

Sadly, I was not able to directly access to the https://seal.htb/manager/html/ (I added an entry in the file to use the URL seal.htb). I received a 403 code response from the application. Looking for something useful on the Internet, I learned that Tomcat and nginx could be affected by a vulnerability due to a different way to parse URLs. Therefore, I investigated deeper this condition and I was able to access to the Tomcat manager GUI using the URL https://seal.htb/manager;name=orange/html/:

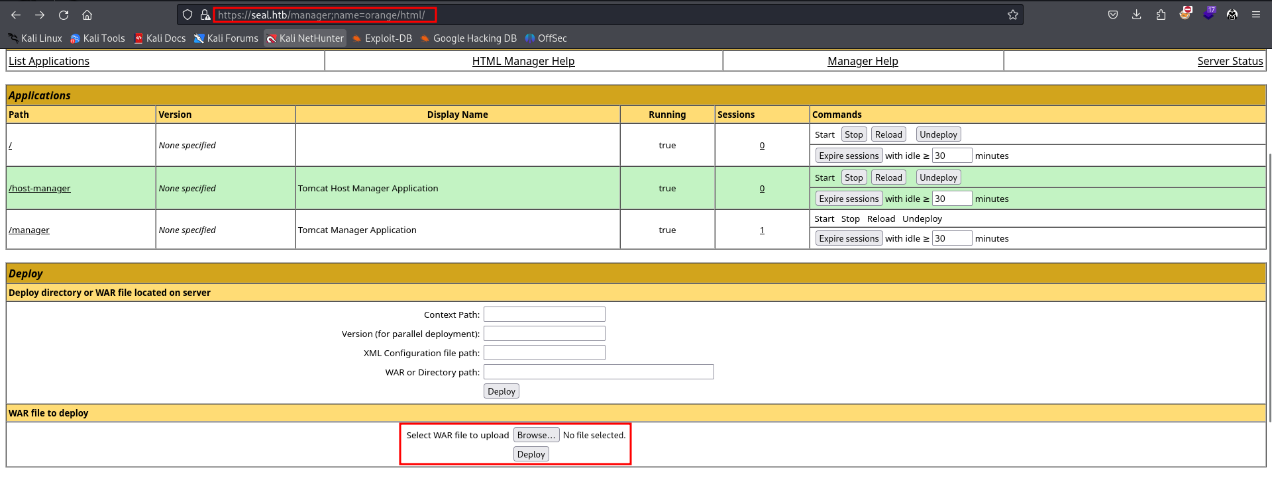


Figure 5 - Tomcat manager GUI

At this point, I easily obtained a user shell via uploading a malicious war file:

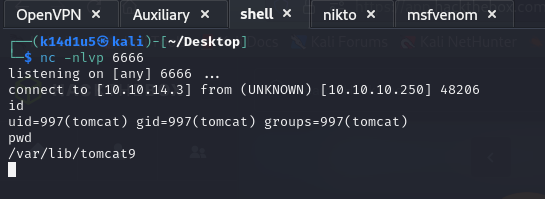


Figure 6 - First user shell

However, the user I had was not useful to retrieve the user flag, so I needed to perform a lateral movement task. Therefore, I explored the file system and I found an interesting file that configure some tasks. These tasks were about a backup and, in particular, the attribute was set (it allowed to copy the file pointed by a link file and not the link file itself):

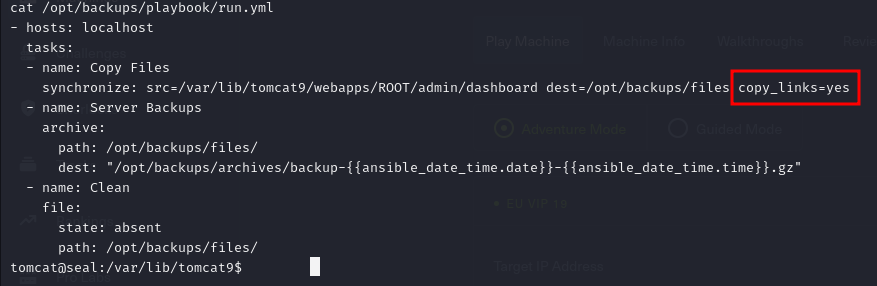


Figure 7 - Dangerous configuration

I kept to investigate these tasks and I found out that the backup was created by Luis. Therefore, I tried to exfiltrate Luis SSH private keys as shown in the following picture:

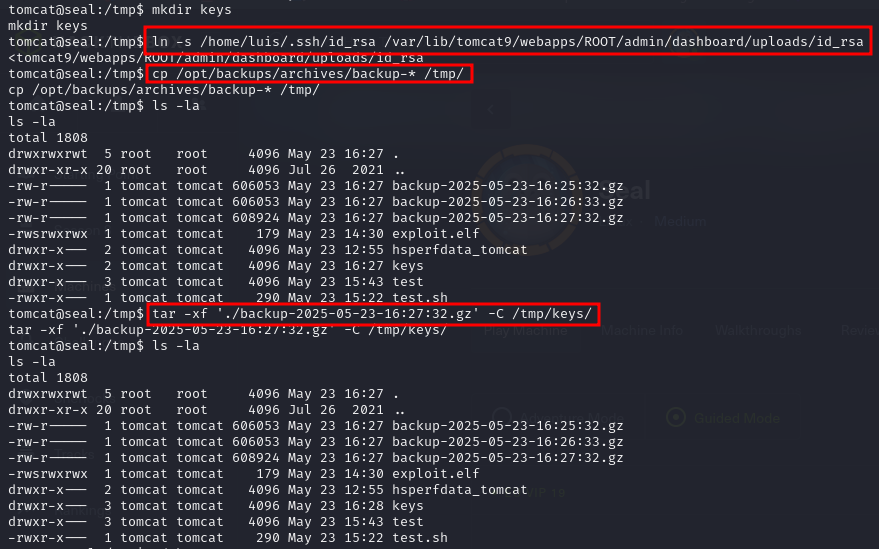


Figure 8 - Extracting Luis SSH private key

I created a simple script to be sure when the backup was completed:

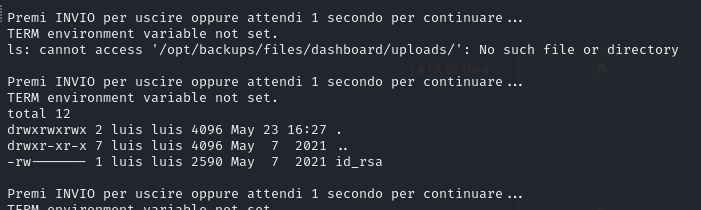


Figure 9 - Backup completed check

At this point, I checked the backup archive, I extracted the private key I tried to involve in the backup and I copied it on my Kali machine. Lastly, I used the key to login in via SSH as Luis and I retrieved the user flag:



Figure 10 - User shell and flag

# **Privilege escalation**

At this point, I just needed to escalate my privileges. To achieve this goal, I checked if Luis account has some sudoers privileges. Luckily, I was able to execute a single script as sudo without using the password. This script executes a yaml playbook, similar to the one that used the attribute. Therefore, I forged a custom malicious playbook, as shown in the following picture:



Figure 11 - Malicious reverse shell

At this point, I just needed to open a lister and execute my malicious playbook to obtain a shell as root and retrieve the root flag, as shown in the following:

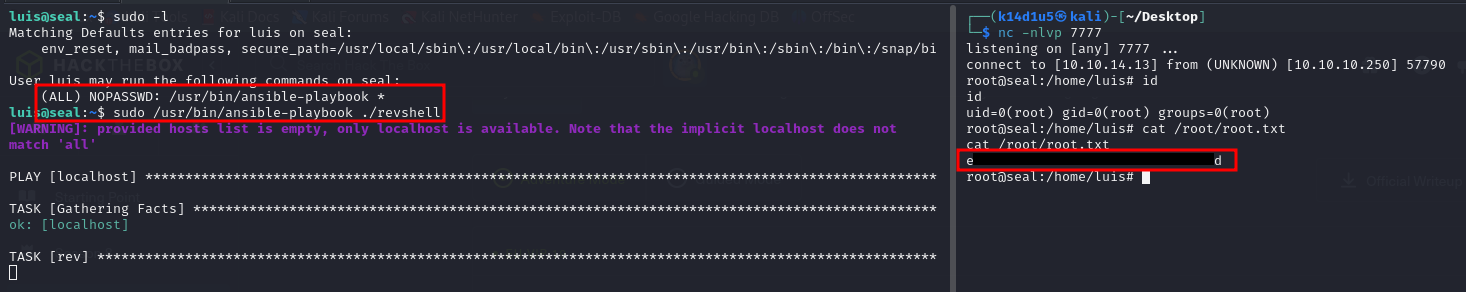


Figure 12 - Root shell and flag

# **Personal comments**

In my opinion, this box is very linear. However, I learned something from it, in particular how I could exploit a yaml configuration if it uses some specific option. Of course, I learned to analyze better these files. I had fun to complete this box If I remember well, I evaluate a little bit less than medium as a global mark on the HackTheBox platform.

# **References**

1. Tomcat and nginx mutual authentication bypass: <https://rioasmara.com/2022/03/21/nginx-and-tomcat-mutual-auth-bypass/>.