TryHackMe Room Write-up

Title: Lo-Fi

Description: Want to hear some lo-fi beats, to relax or study to? We've got you covered!

Author: Igor Buszta

Lo-Fi

Description: *Access this challenge by deploying both the vulnerable machine by pressing the green "Start Machine" button located within this task, and the TryHackMe AttackBox by pressing the "Start AttackBox" button located at the top-right of the page.*

Navigate to the following URL using the AttackBox: http://MACHINE_IP and find the flag in the root of the filesystem.

Note: The web page does load some elements from external sources. However, they do not interfere with the completion of the room.

Objective: Climb the filesystem to find the flag!

Data: IP of webpage.

Tools: macOS, Tryhackme VPN, LFI

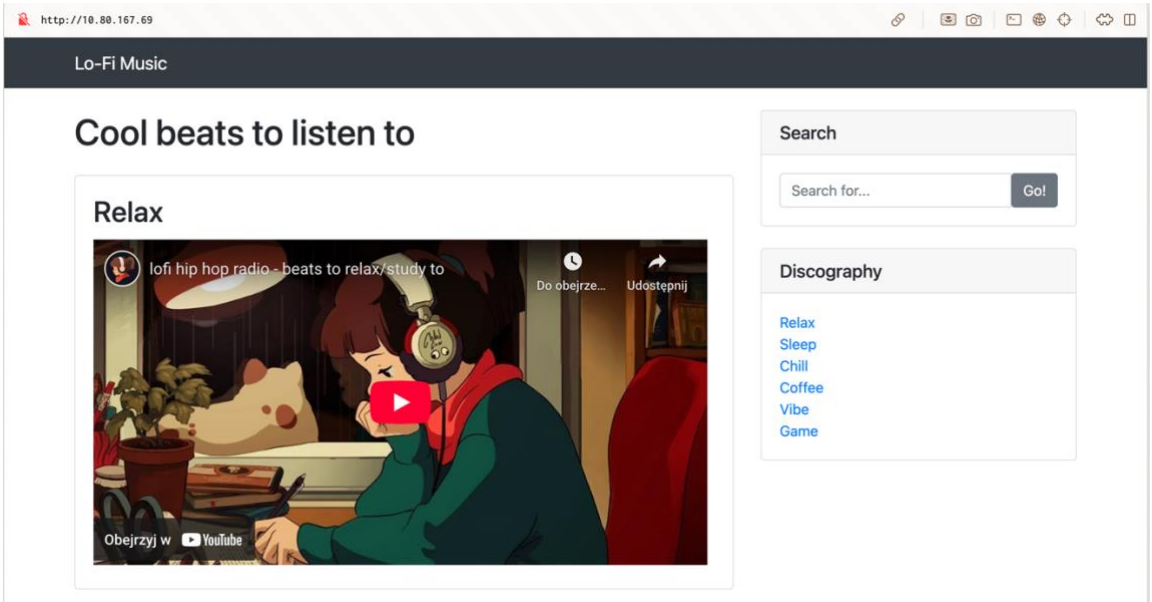
When we look at the similiar rooms in the description, we can be sure, that this room includes Local File Inclusion or Path Traversal. Both methods are based on Linux/Windows directories management. Linux uses '../' to mark a directory one level above. The redundant '../' are later ignored if used to much, so the more the better.

Let's visit our target website (I used VPN this time, AttackBox was loading the page for too long)

To connect to VPN you must download the VPN config file. Go to your terminal and type in command:

```
sudo openvpn --config /Path/To/Your/Config/File/[nickname].ovpn
```

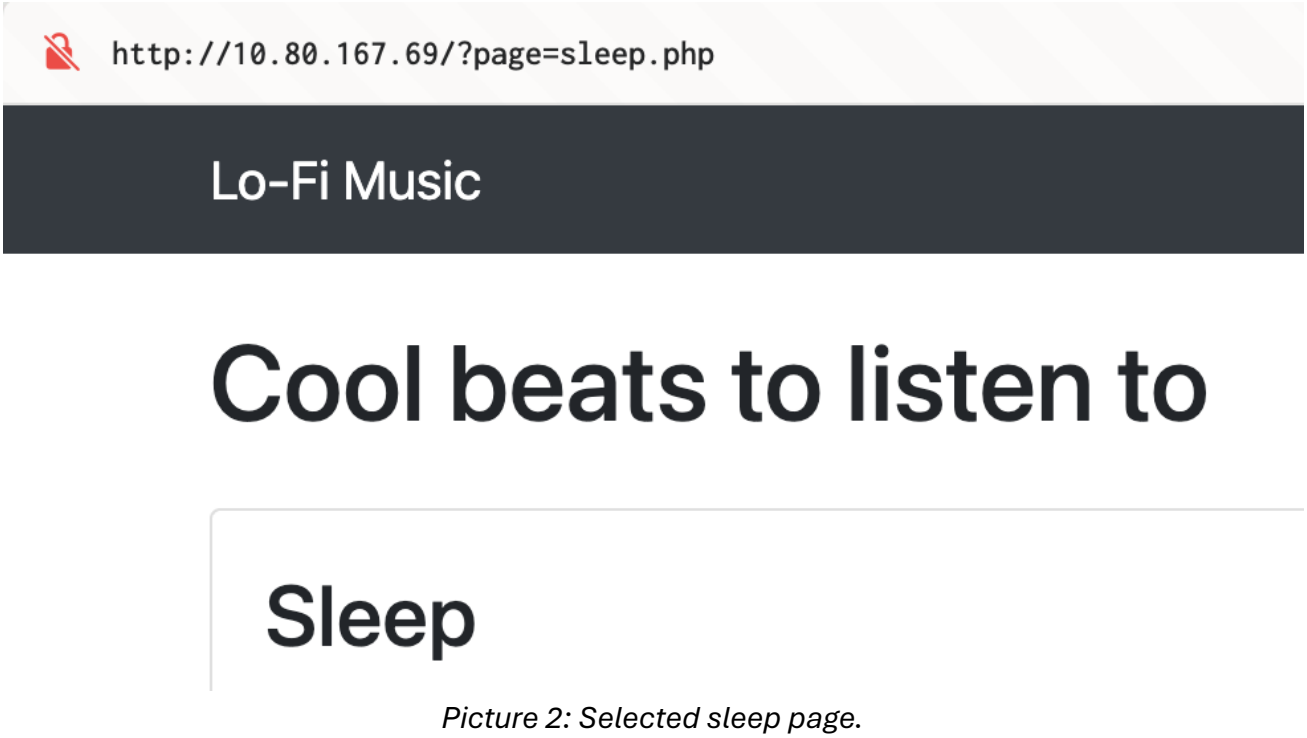
Now let's visit the page from our own browser.



Picture 1: Front page.

We know that we need to manipulate the URL in order to find some information or flag.

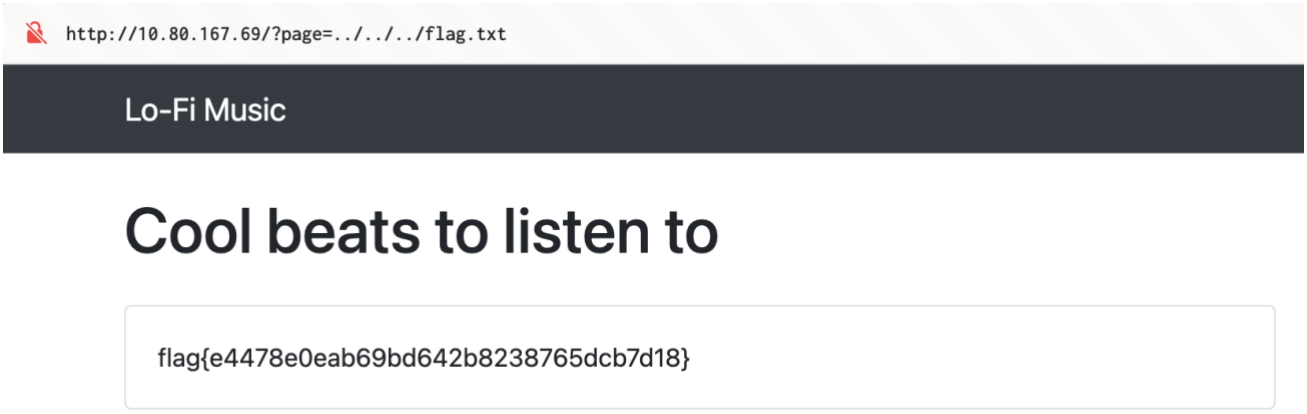
Let's see what are the main requests and semantics. Let's see what's under *Sleep* in page's Discography.



Picture 2: Selected sleep page.

It is quite straight forward. The page shows us a file *sleep.php*. Let's remove this uninteresting file and try the most obvious file and see if there's anything. Let's use semantics we know to try and get to higher directories.

As mentioned before, trying only ../ is most likely not to work. That's why we use multiple '../' to dive deeper. And what do you know, there's our flag we are looking for.



Picture 3: Retrieved flag from flag.txt