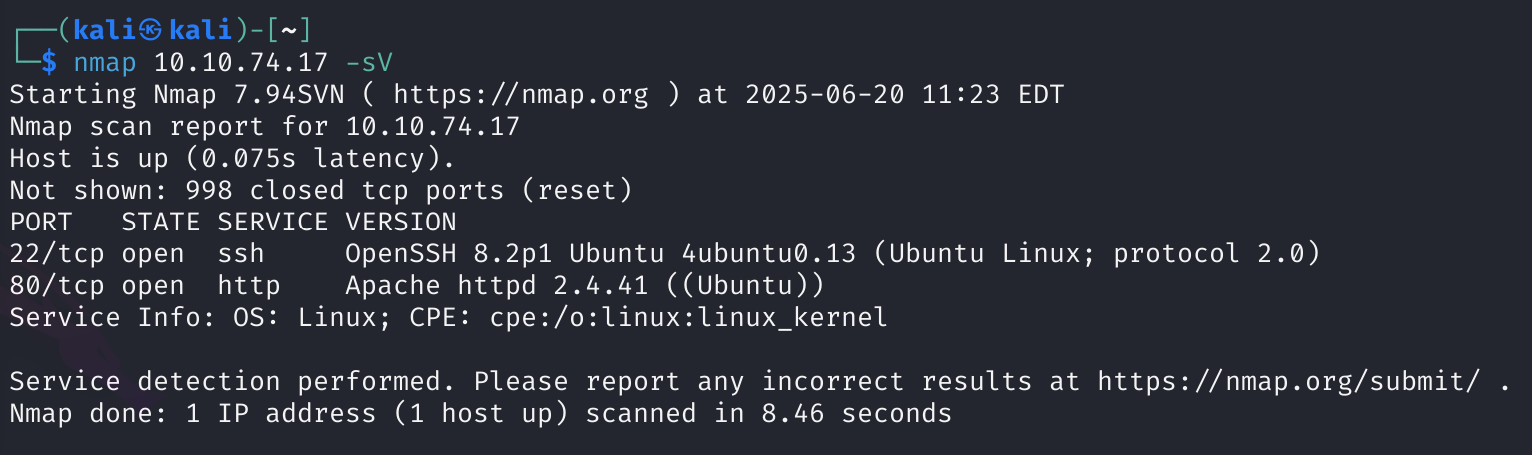
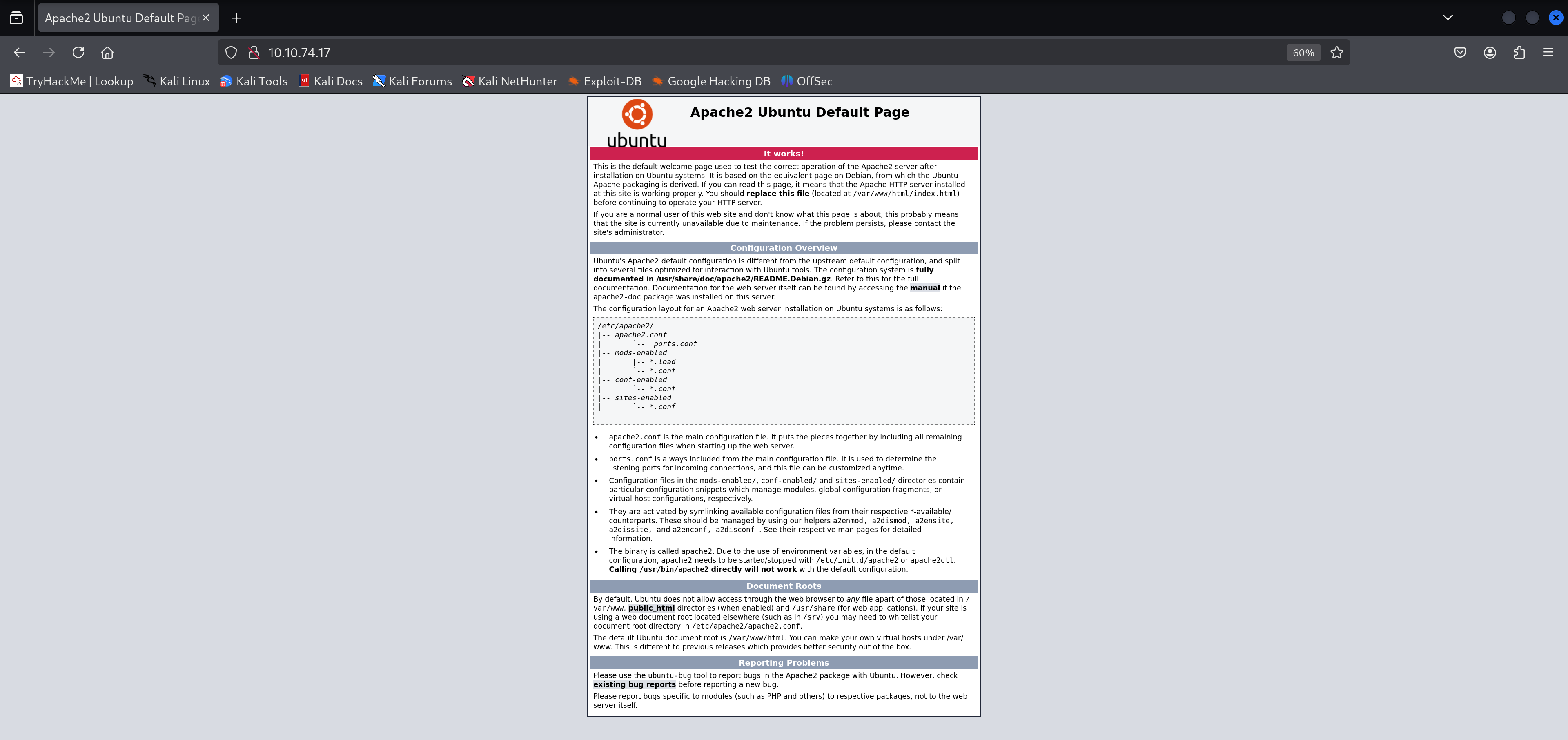
**Dreaming Solution Documentation**

**Written by: Yuval Quina**

1. To begin with, I started by performing a basic Nmap port scanning using this command:

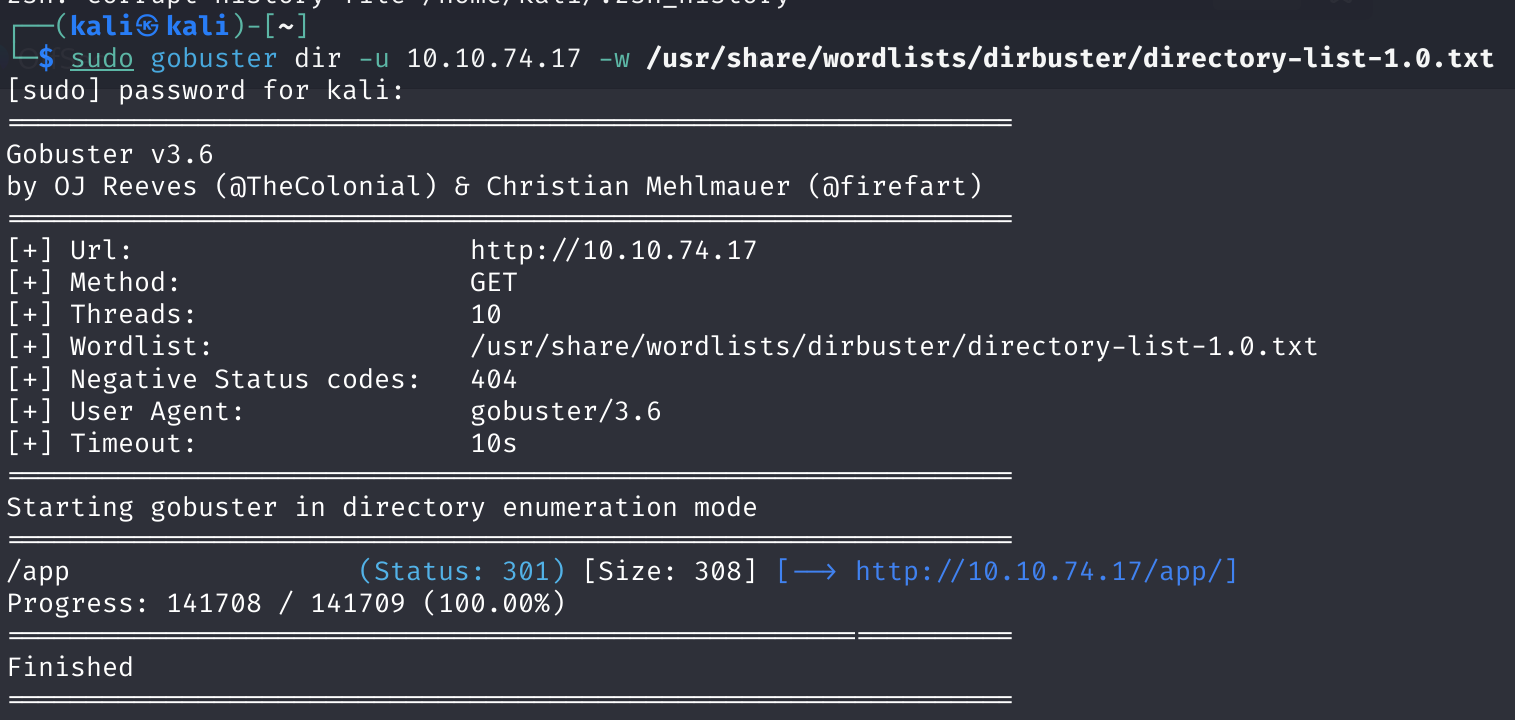


1. So, I opened the website of the machine:

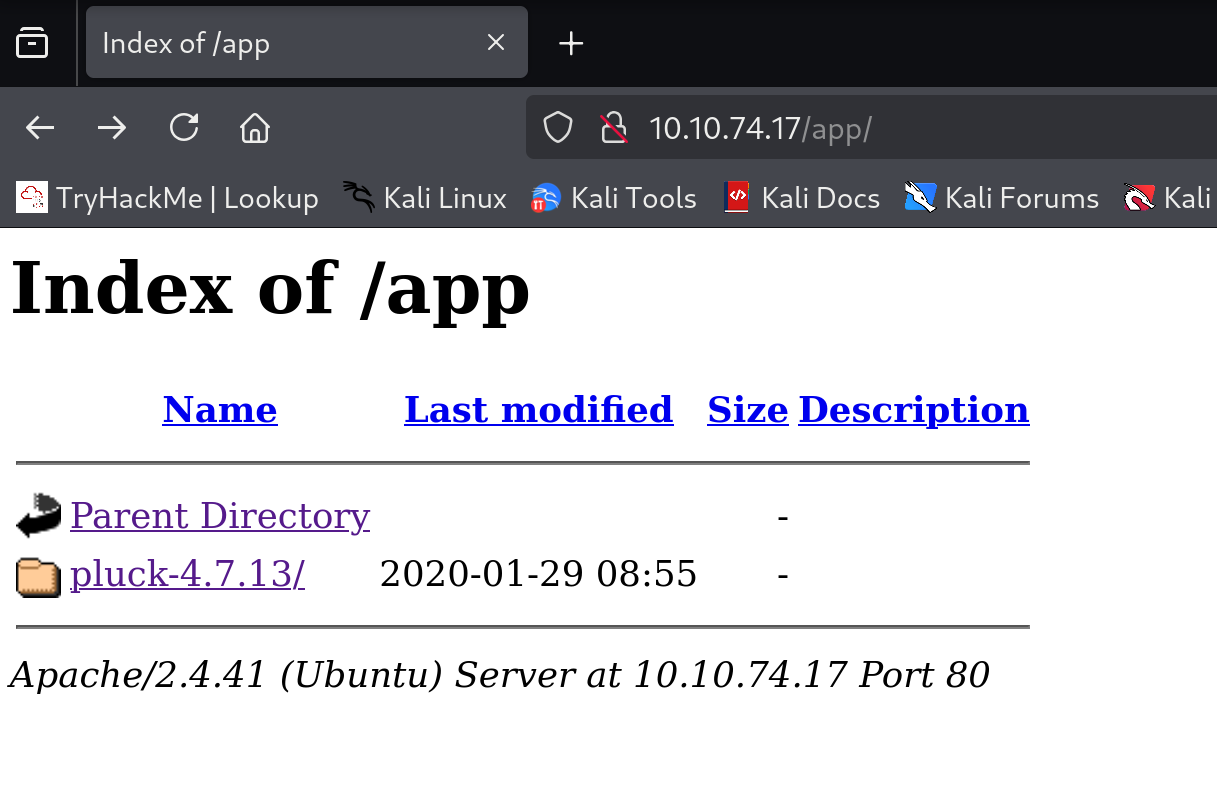


I immediately noticed that this machine running of “Ubuntu” and running “Apache” server.

1. I did a little investigating on this website (such as viewing inspector, trying robots.txt and more) but I found nothing.
2. I chose to perform a directories scan using gobuster and found this directory:



1. So, I opened the directory:



I clicked on the “pluck-4.7.13” directory and it moved me to this page:

תמונה שמכילה טקסט, צילום מסך, תוכנה, תכונות מולטימדיה

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

Now, I started investigating this page.

1. I saw in the inspector that clicking on the “admin” label will move me to a login page:



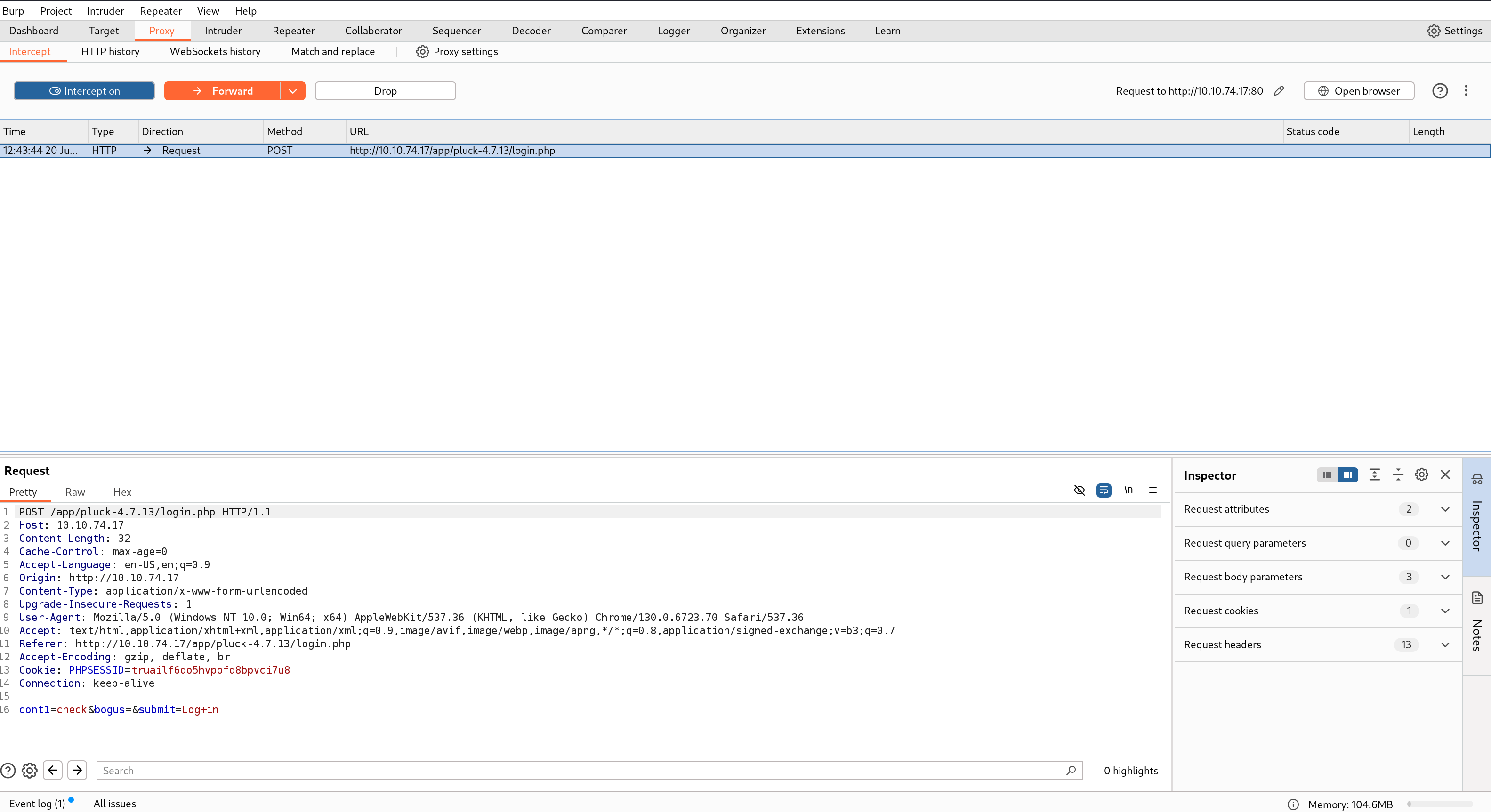
So, of course I clicked on it and this is the login page:

תמונה שמכילה טקסט, צילום מסך, תוכנה, סמל מחשב

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

1. I saw that I needed only password (probably the username is admin) so I used the “burpsuite” tool to bruteforce the right password.

I began with capturing a login packet:



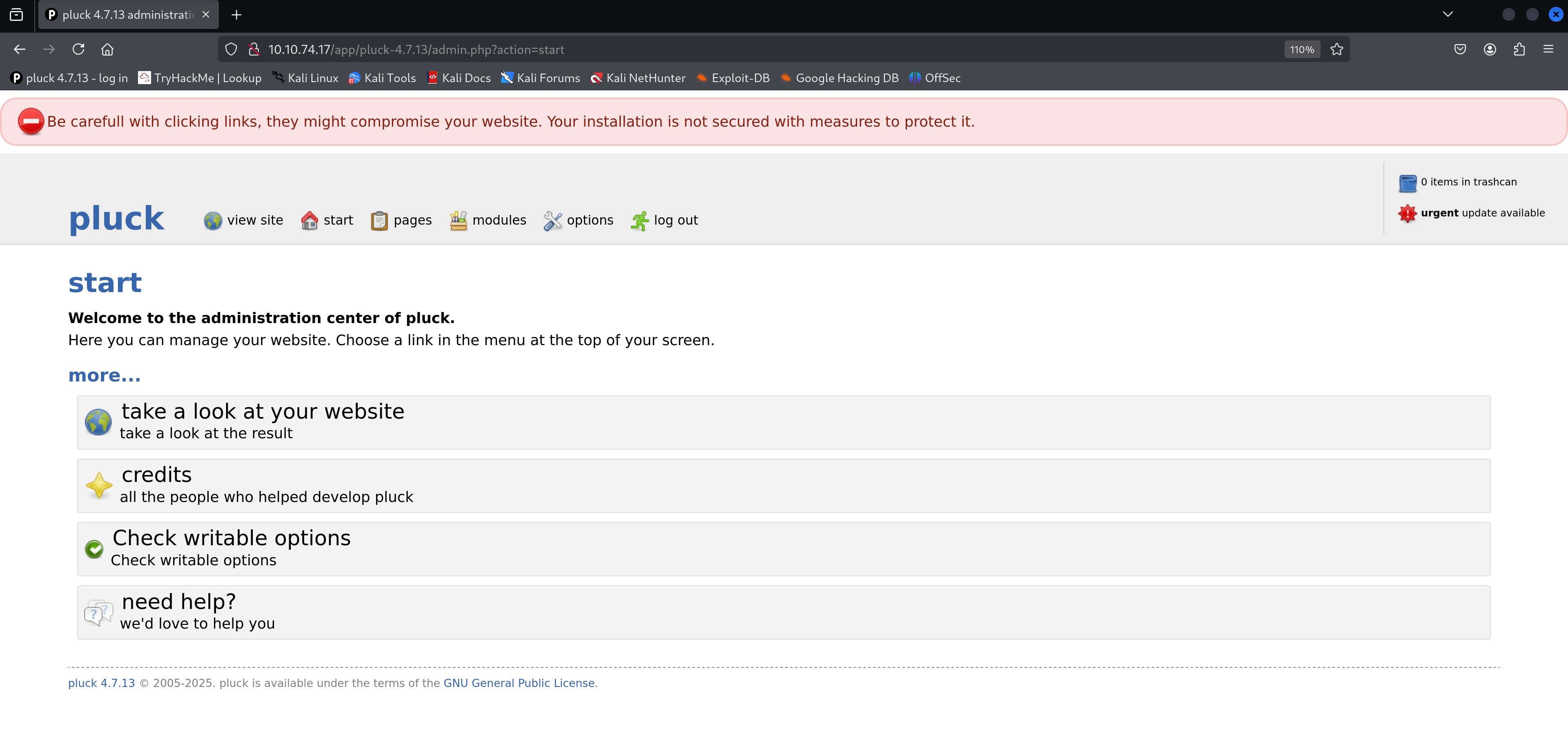
I moved it to the “Repeater”, marked the “contl” header value, select the “/usr/share/wordlists/rockyou.txt” wordlist and runed the bruteforce:

תמונה שמכילה טקסט, צילום מסך, תוכנה, תצוגה

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

I found that the right password is “password”.

1. Now, I just entered it, and this page was loaded:



So, I scrolled through the page and found that I could upload files to the machine.

תמונה שמכילה טקסט, גופן, צילום מסך

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

Therefore, I uploaded this PHP reverse shell file:

תמונה שמכילה טקסט, צילום מסך, גופן

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

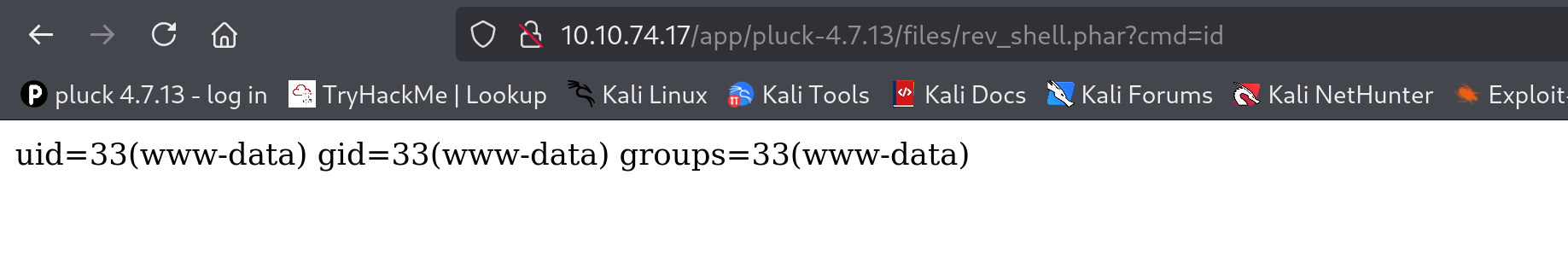
The problem was that when I uploaded this file suddenly the files system transformed it from .php to .txt which makes it inexecutable:



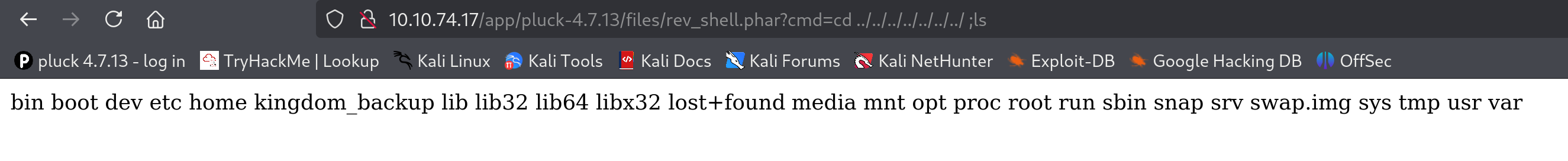
1. This made me think for a moment and after couple of minutes I remembered that there are some alternative executable php formats (like: .phtml and .phar) so I tried both of them I and found that .phar didn’t changed!
2. So, I clicked on the magnifying glass icon (which takes me to the path of the files) and I added the “cmd” header that enable me to run commands over the machine:



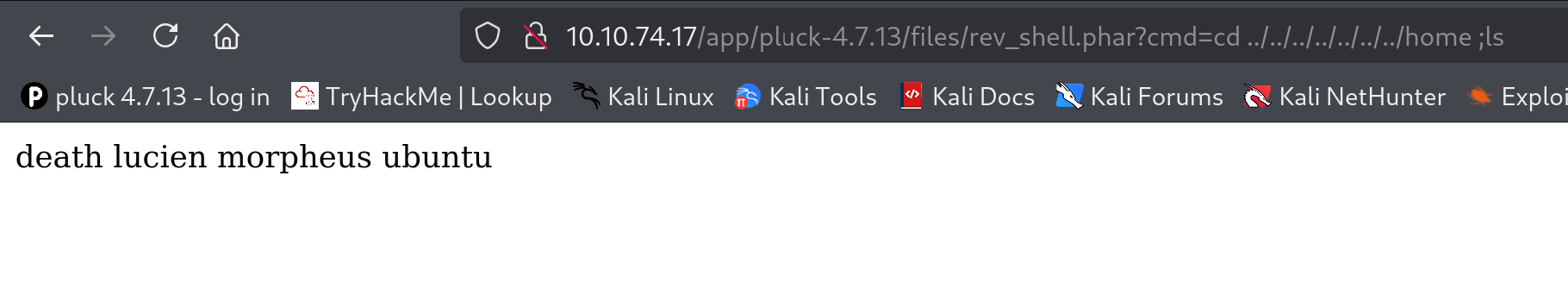
1. After doing all this, I tested it by using the basic “id” command and it worked:



1. Afterwards, I navigated to the root directory:



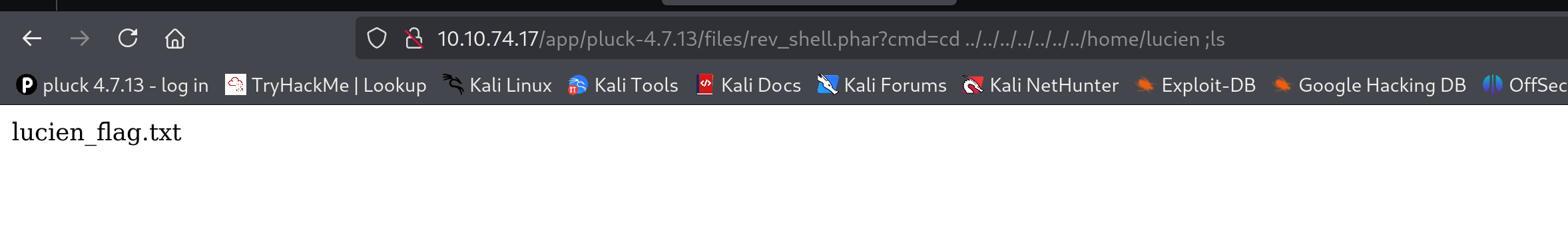
1. After further navigation I found on the home directory the 3 users of the machine that also appears in the flag’s description:



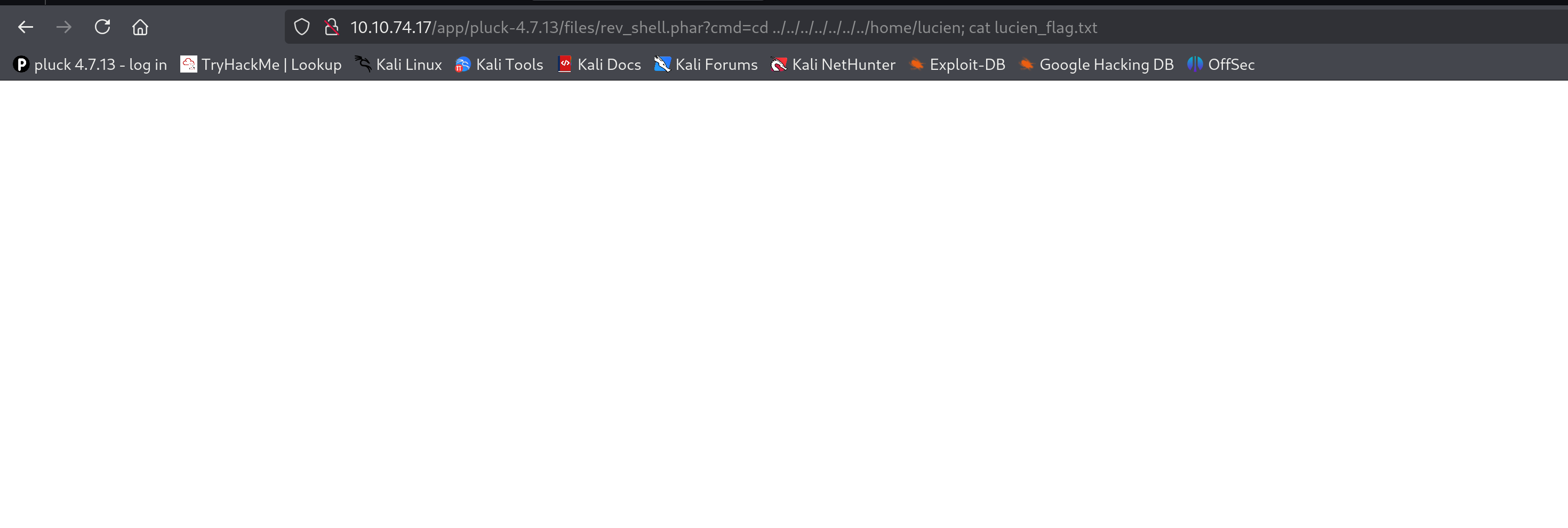
תמונה שמכילה טקסט, צילום מסך, גופן

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

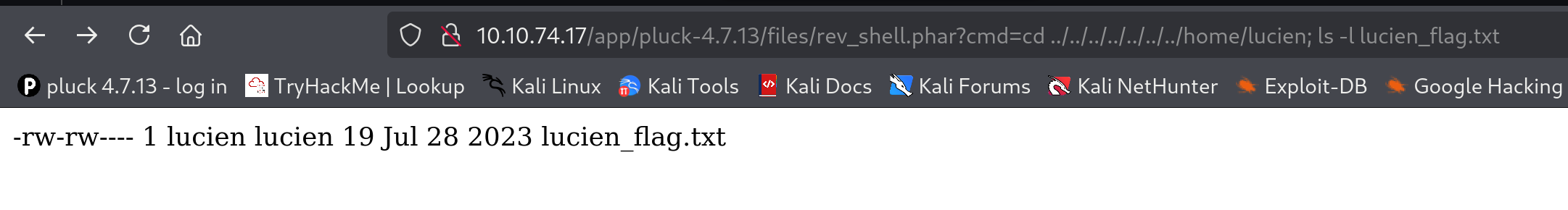
1. The first user that I navigated to was “lucien” and I found in his directory the flag file:



I tried to open it, but it didn’t open:



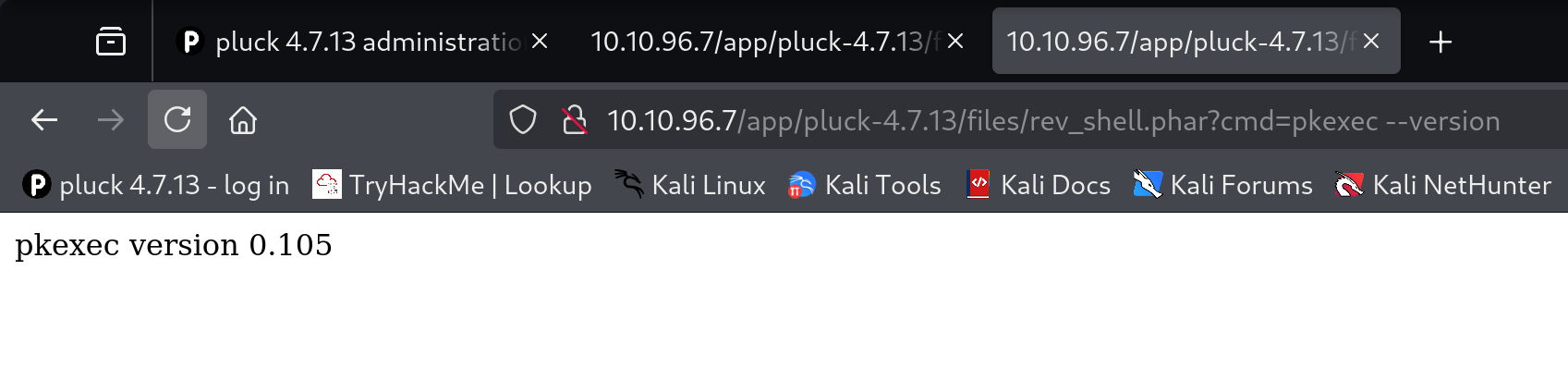
1. I figured it out that I need to somehow get connected as lucient to access the file content:



1. I searched for a vulnerable command, and I found this one:

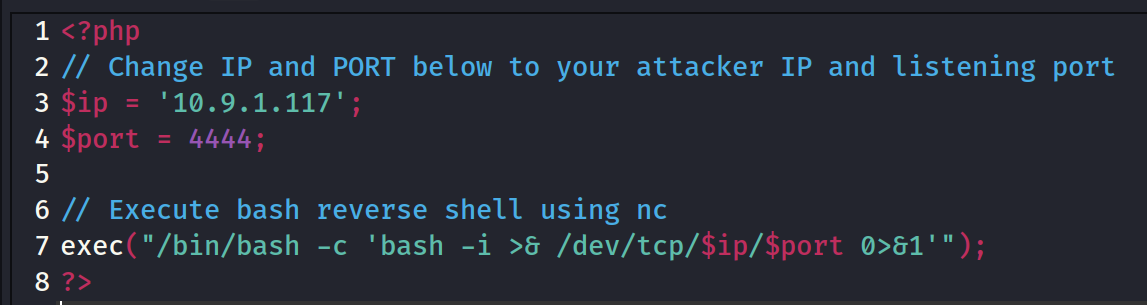


After checking its version:



I found that this command is vulnerable to “CVE-2021-4034” (which names “PwnKit”).

1. The problem was that I got a web shell and not a meterpreter so I could not use msfconsole as usual. Therefore, I chose to first of all change to “netcat” so I will be able to control over the shell more comfortably. I did it by uploading another .phar file that will run the following commands in order to connect to a listener that I’m running on my machine:



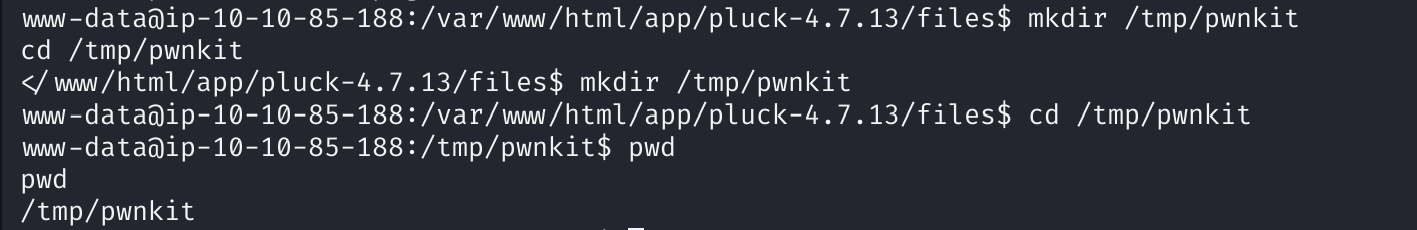
תמונה שמכילה טקסט, צילום מסך, גופן

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

תמונה שמכילה טקסט, צילום מסך, תוכנה, תכונות מולטימדיה

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

1. Now, I needed to perform the execution of the exploit manually. I created a working directory for that attack on the target side:



1. Note: I found that the problem is in the exploit.c in the values of the DIR and SHELL and CHARSET because you should customize them according to machine’s