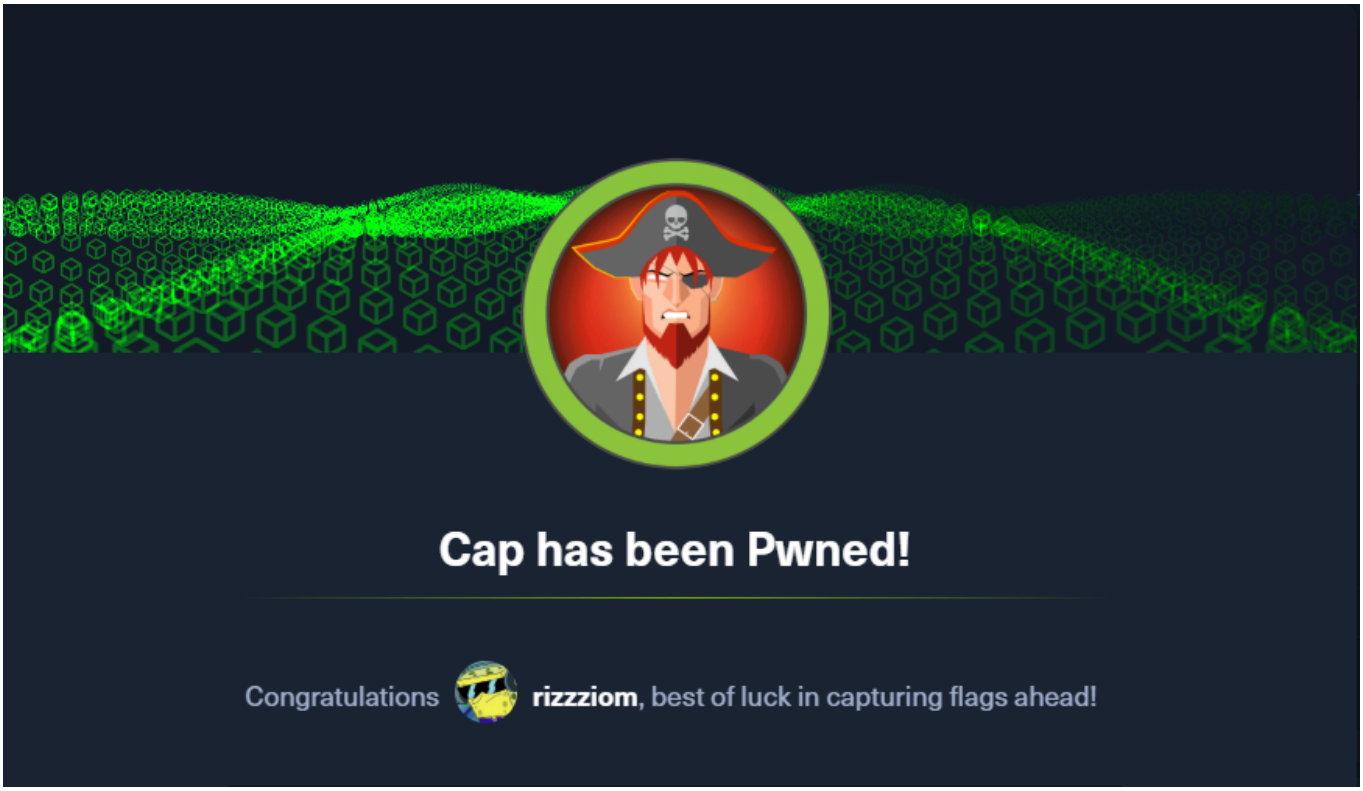
CAP

LINK TO MACHINE: https://app.hackthebox.com/machines/Cap



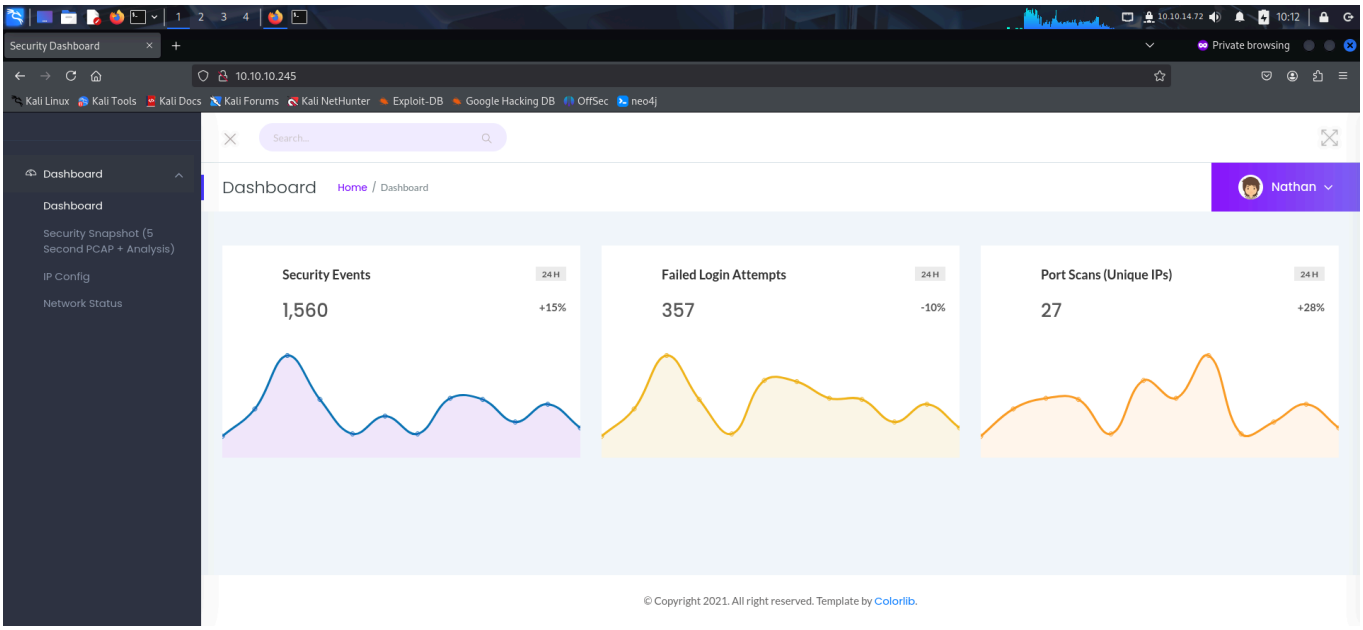
SCANNING

I performed an **nmap** aggressive scan to find open ports and the services running on them.

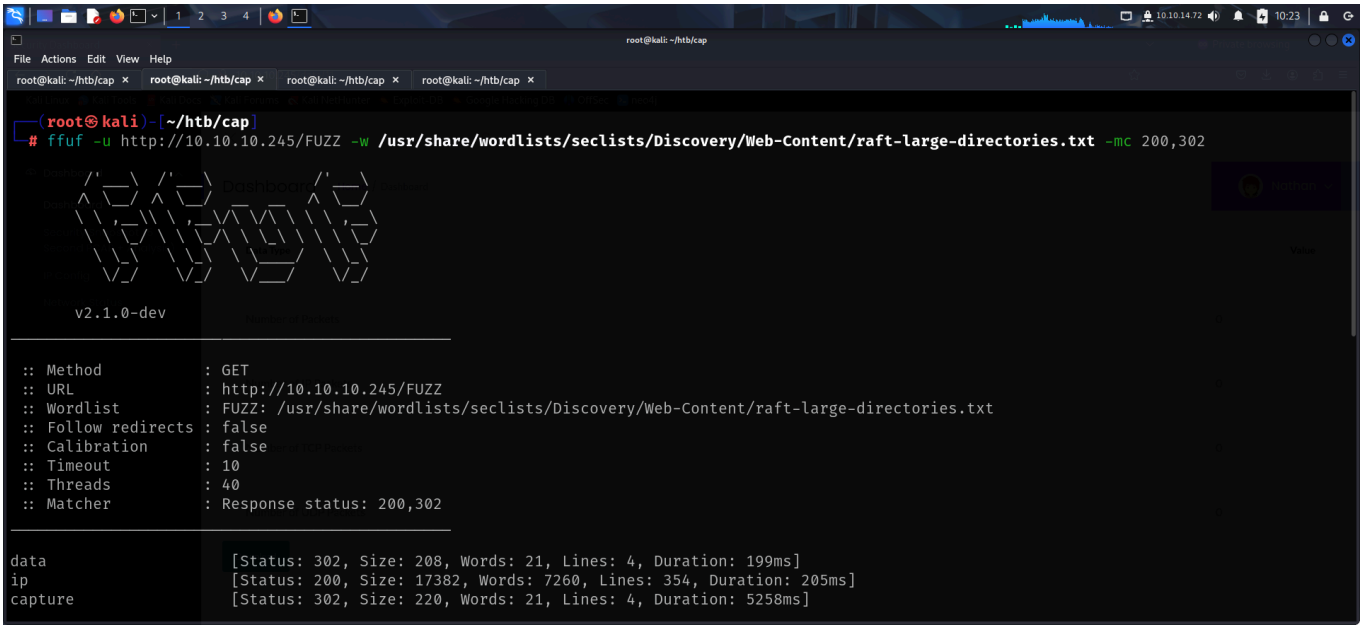
```
┌──(root㉿kali)-[~/htb/cap]
└─# nmap -A -p- 10.10.10.245 --min-rate 10000 -oN cap.nmap
Starting Nmap 7.95 ( https://nmap.org ) at 2025-03-03 10:09 EST
Nmap scan report for 10.10.10.245
Host is up (0.19s latency).
Not shown: 65532 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.2 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 fa:80:a9:b2:ca:3b:88:69:a4:28:9e:39:0d:27:d5:75 (RSA)
|   256 96:d8:f8:e3:e8:f7:71:36:c5:49:d5:9d:b6:a4:c9:0c (ECDSA)
|_  256 3f:d0:ff:91:eb:3b:f6:e1:9f:2e:8d:de:b3:de:b2:18 (ED25519)
80/tcp open  http    Gunicorn
|_http-server-header: gunicorn
|_http-title: Security Dashboard
Device type: general purpose
Running: Linux 5.X
OS CPE: cpe:/o:linux:linux_kernel:5
OS details: Linux 5.0 - 5.14
Network Distance: 2 hops
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using port 8888/tcp)
HOP RTT       ADDRESS
1   184.20 ms 10.10.14.1
```

FOOTHOLD

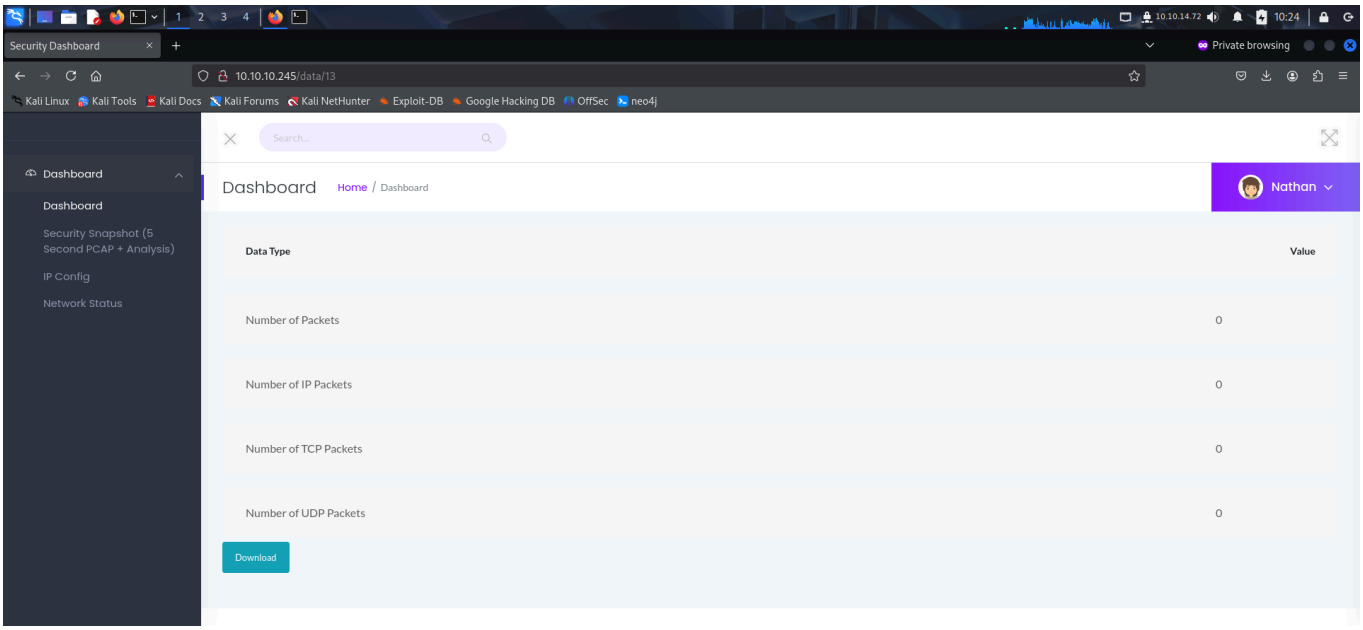
The **nmap** scan identified an **http** server running so I accessed it using my browser and got a dashboard.

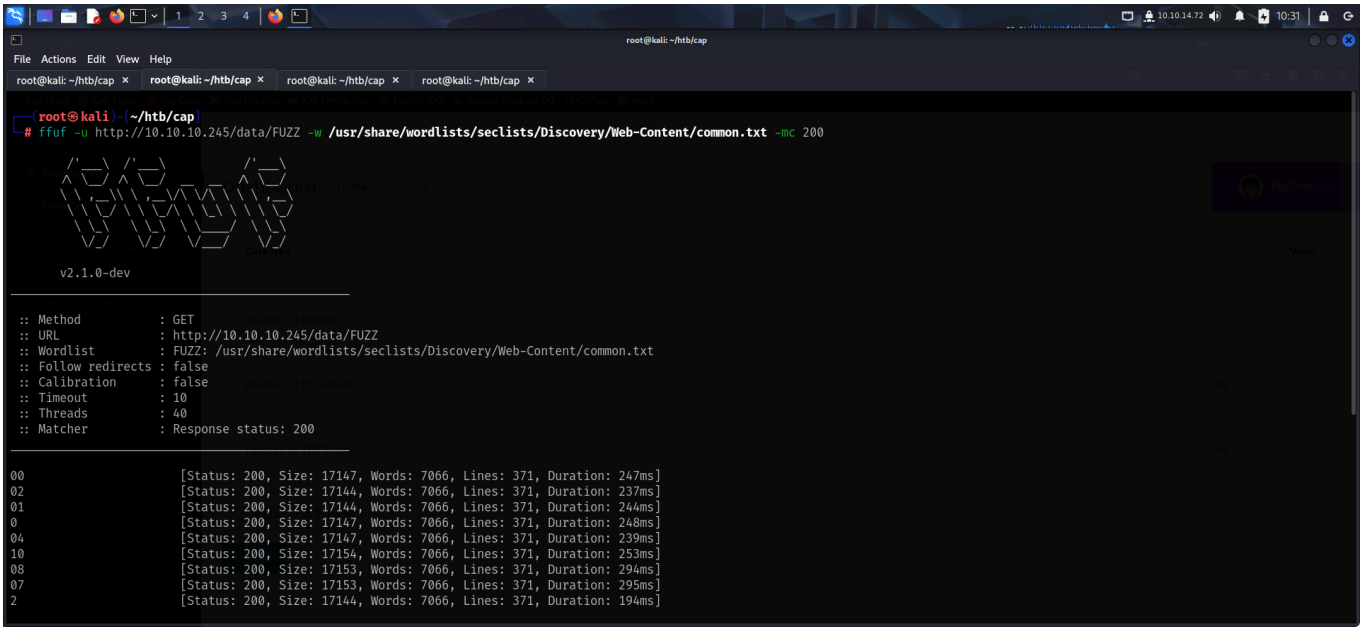


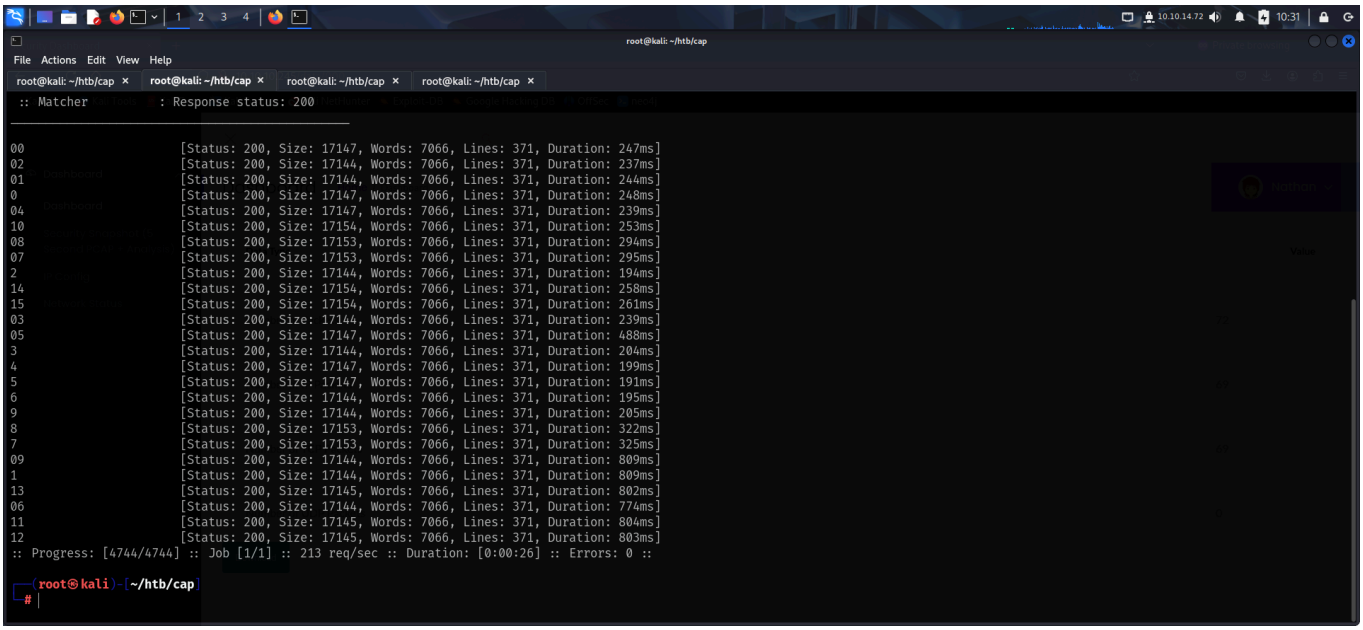
I explored the application and in the meantime, performed a scan to find directories using **ffuf**.



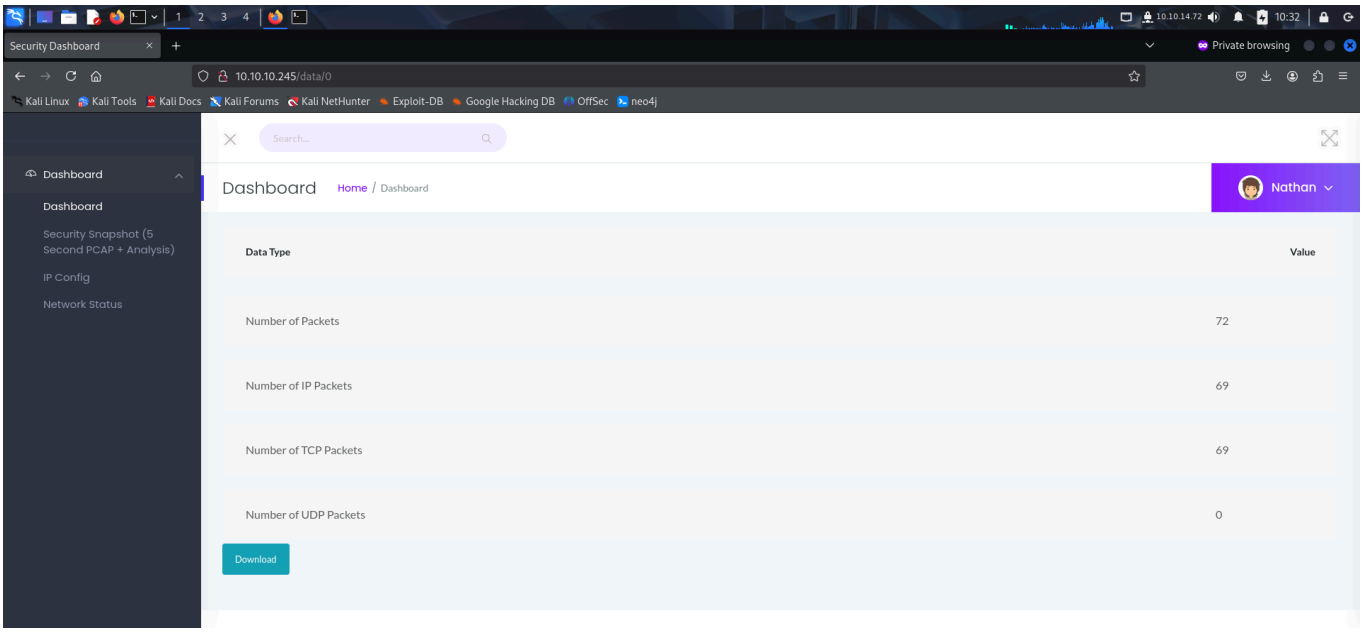
The *data* directory seemed interesting. It contained various capture files.

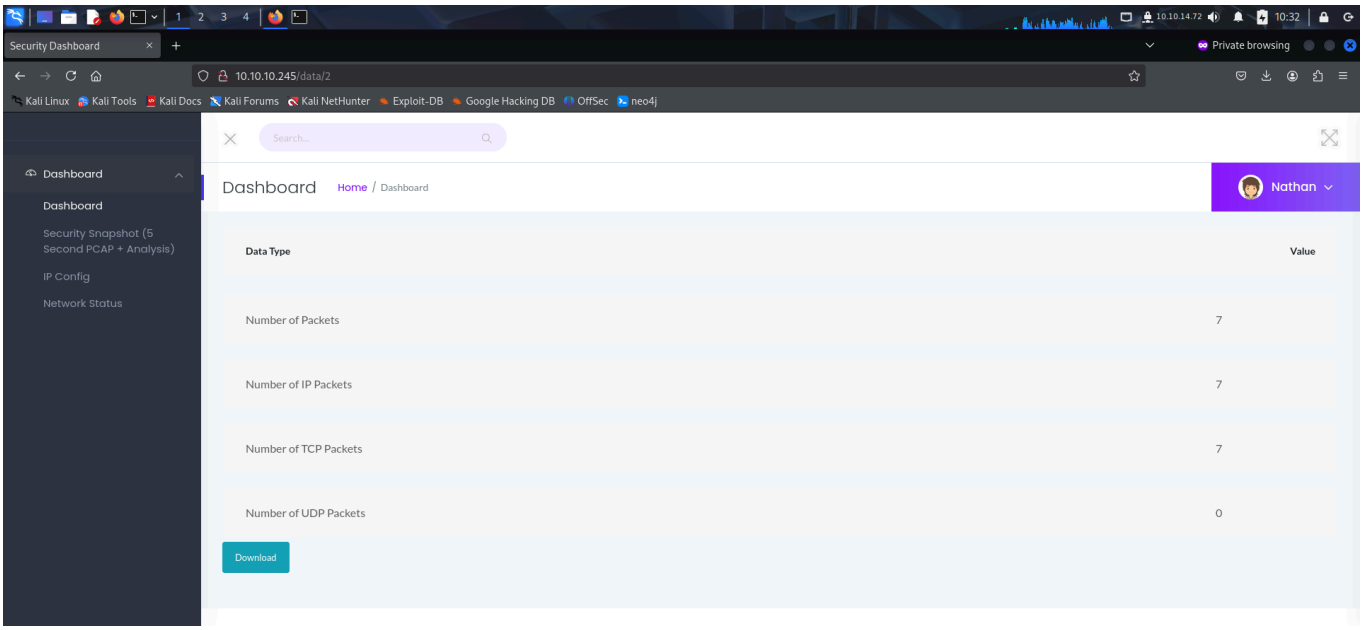

Security Dashboard
10.10.10.245/data/13
Private browsing
Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking DB
OffSec
neo4j
Dashboard
Dashboard
Security Snapshot (5 Second PCAP + Analysis)
IP Config
Network Status
Dashboard
Home / Dashboard
Nathan
Data Type
Value
Number of Packets
0
Number of IP Packets
0
Number of TCP Packets
0
Number of UDP Packets
0
Download

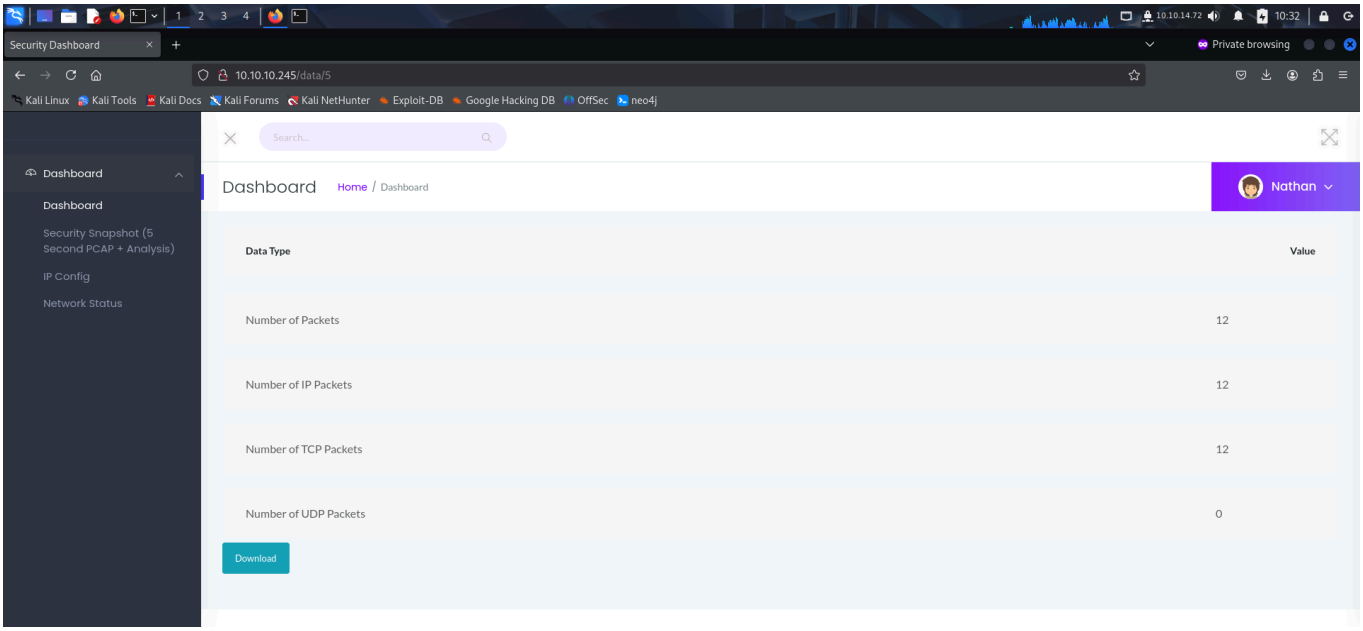

root@kali: ~/htb/cap
File Actions Edit View Help
root@kali: ~/htb/cap
root@kali: ~/htb/cap
root@kali: ~/htb/cap
root@kali: ~/htb/cap
root@kali)-[~/htb/cap]
ffuf -u http://10.10.10.245/data/FUZZ -w /usr/share/wordlists/seclists/Discovery/Web-Content/common.txt -mc 200
v2.1.0-dev
:: Method : GET
:: URL : http://10.10.10.245/data/FUZZ
:: Wordlist : FUZZ: /usr/share/wordlists/seclists/Discovery/Web-Content/common.txt
:: Follow redirects : false
:: Calibration : false
:: Timeout : 10
:: Threads : 40
:: Matcher : Response status: 200
00 [Status: 200, Size: 17147, Words: 7066, Lines: 371, Duration: 247ms]
02 [Status: 200, Size: 17144, Words: 7066, Lines: 371, Duration: 237ms]
01 [Status: 200, Size: 17144, Words: 7066, Lines: 371, Duration: 244ms]
0 [Status: 200, Size: 17147, Words: 7066, Lines: 371, Duration: 248ms]
04 [Status: 200, Size: 17147, Words: 7066, Lines: 371, Duration: 239ms]
10 [Status: 200, Size: 17154, Words: 7066, Lines: 371, Duration: 253ms]
08 [Status: 200, Size: 17153, Words: 7066, Lines: 371, Duration: 294ms]
07 [Status: 200, Size: 17153, Words: 7066, Lines: 371, Duration: 295ms]
2 [Status: 200, Size: 17144, Words: 7066, Lines: 371, Duration: 194ms]

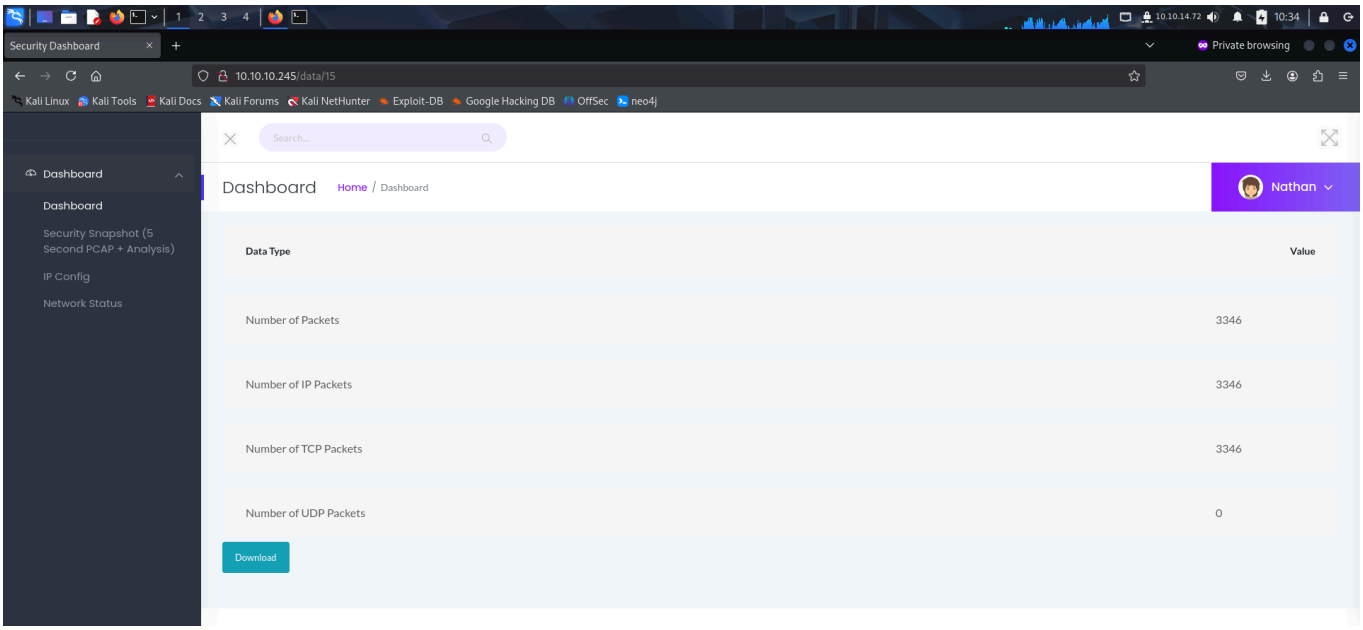


I downloaded the packet capture files.

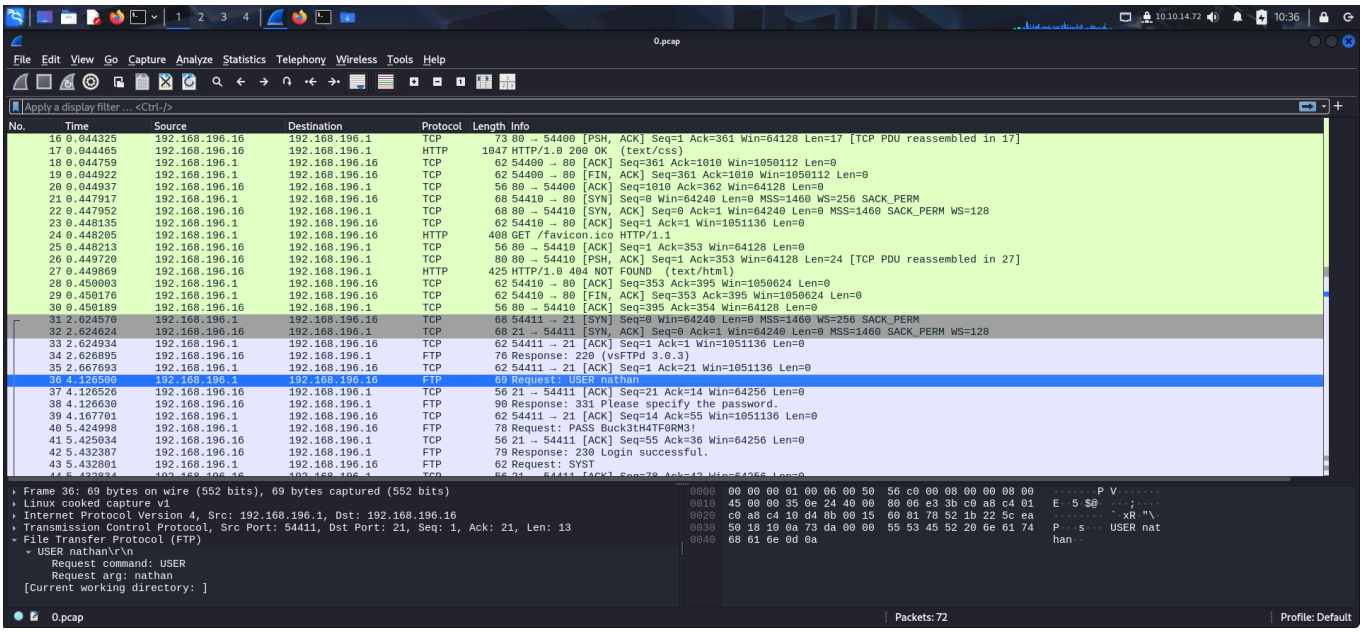


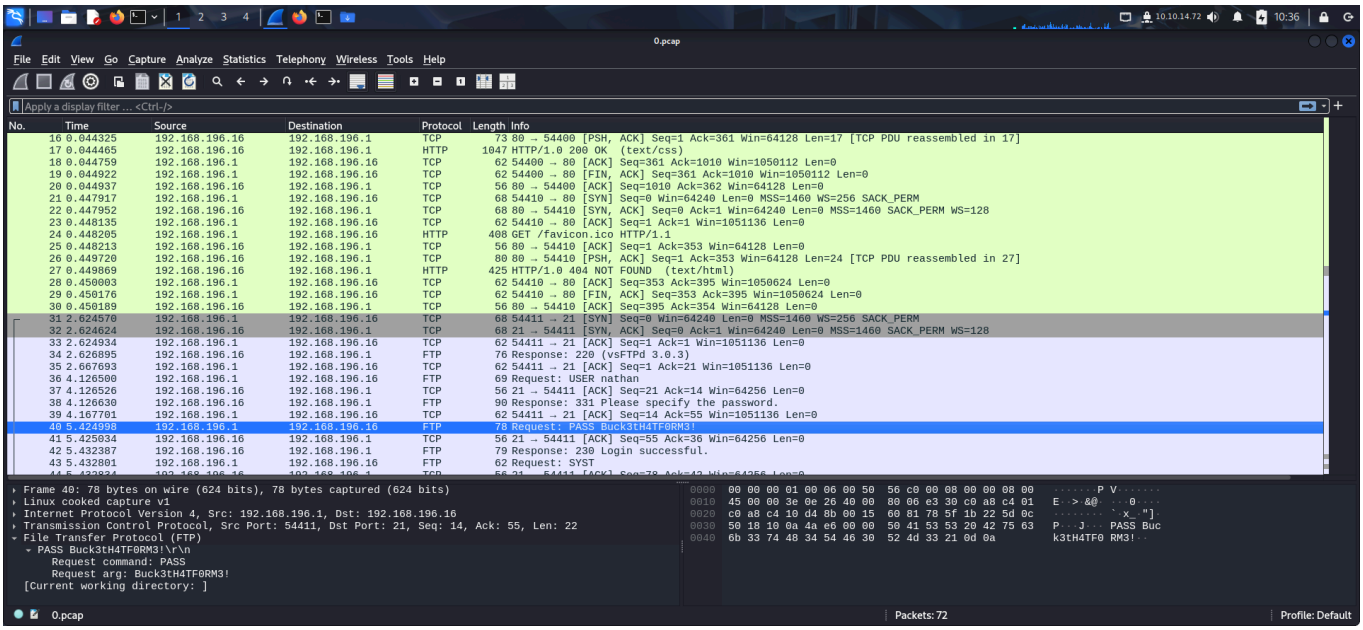

Security Dashboard
10.10.10.245/data/2
Private browsing
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec neo4j
Search...
Dashboard
Dashboard
Security Snapshot (5 Second PCAP + Analysis)
IP Config
Network Status
Dashboard
Home / Dashboard
Nathan
Data Type
Value
Number of Packets
7
Number of IP Packets
7
Number of TCP Packets
7
Number of UDP Packets
0
Download


Security Dashboard
10.10.10.245/data/5
Private browsing
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec neo4j
Search...
Dashboard
Dashboard
Security Snapshot (5 Second PCAP + Analysis)
IP Config
Network Status
Dashboard
Home / Dashboard
Nathan
Data Type
Value
Number of Packets
12
Number of IP Packets
12
Number of TCP Packets
12
Number of UDP Packets
0
Download

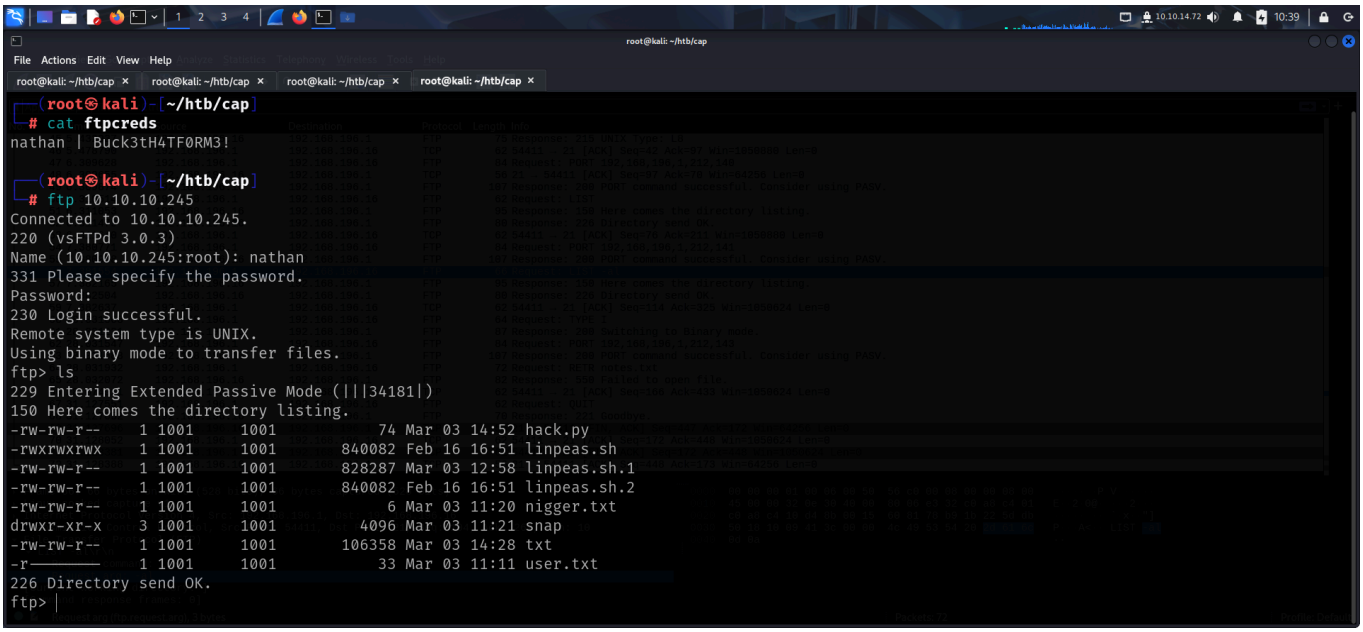


The file *0.pcap* contained the **ftp** credentials.

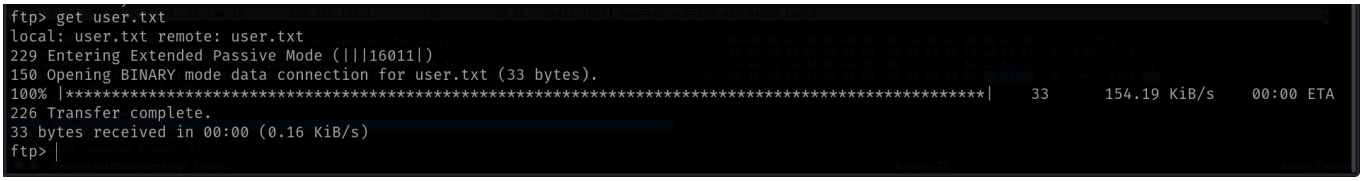


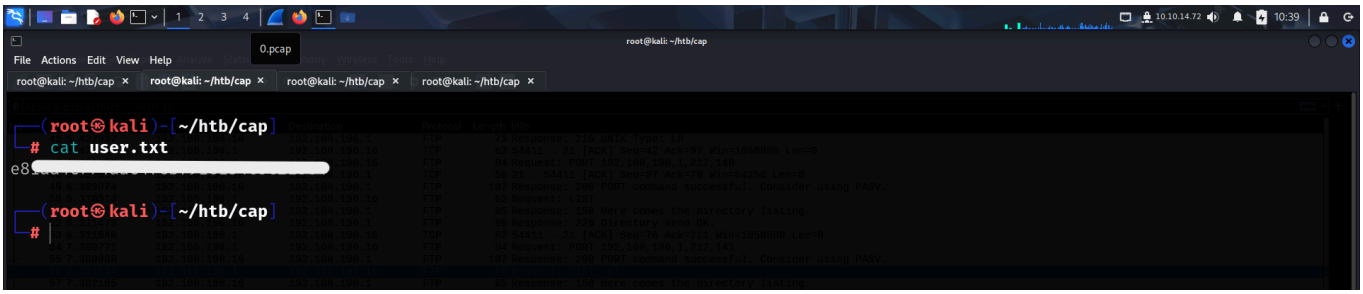


I used those credentials to log into the **ftp** server and found the user flag there.

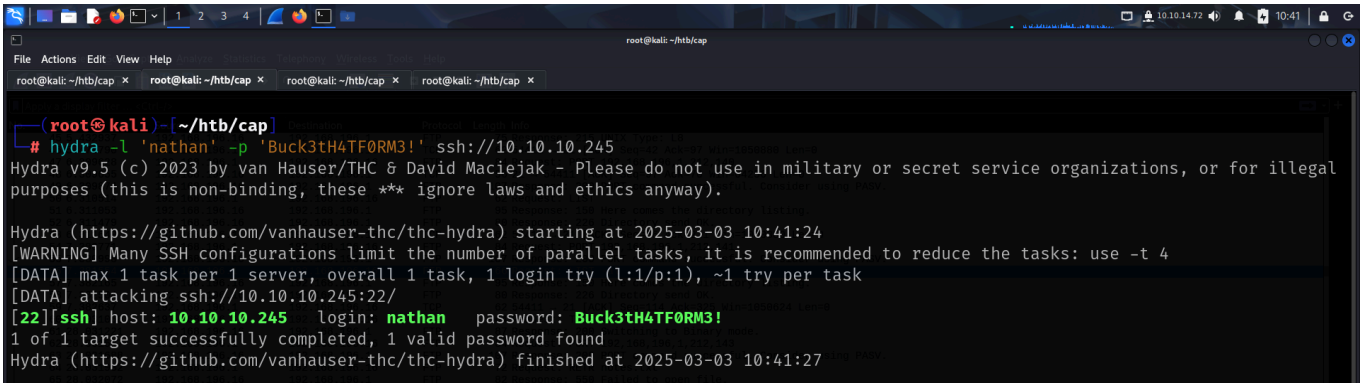


I downloaded the user flag on my system and read the contents.

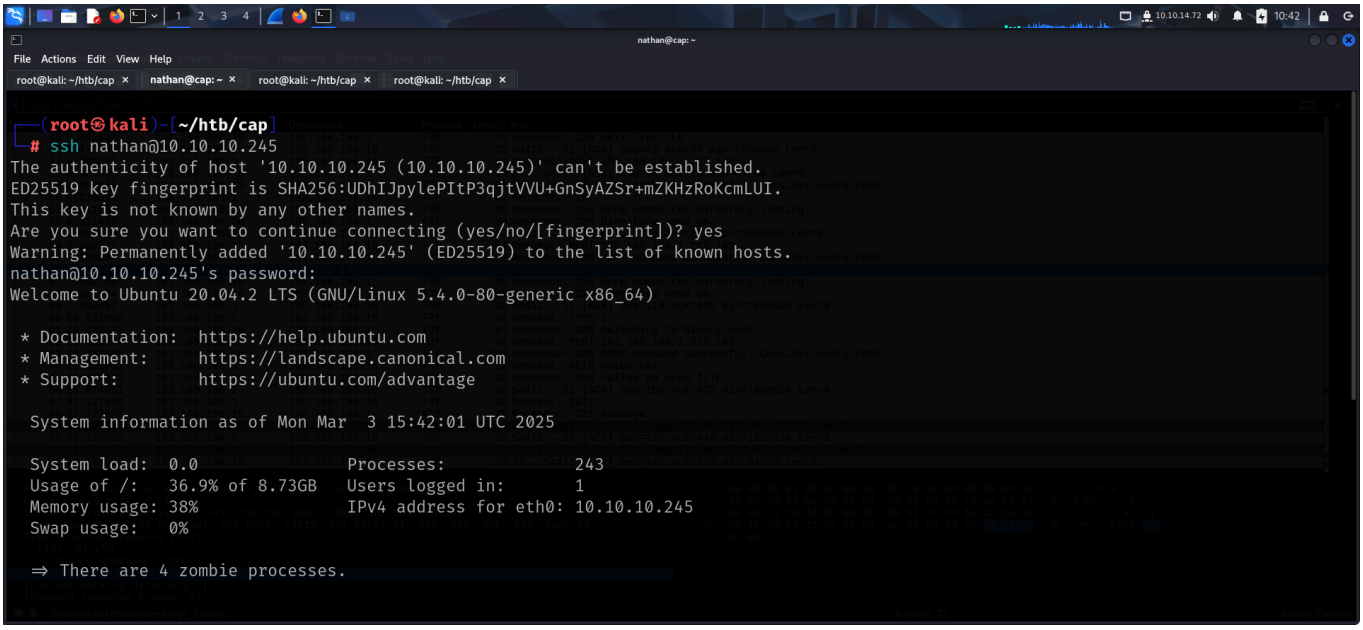


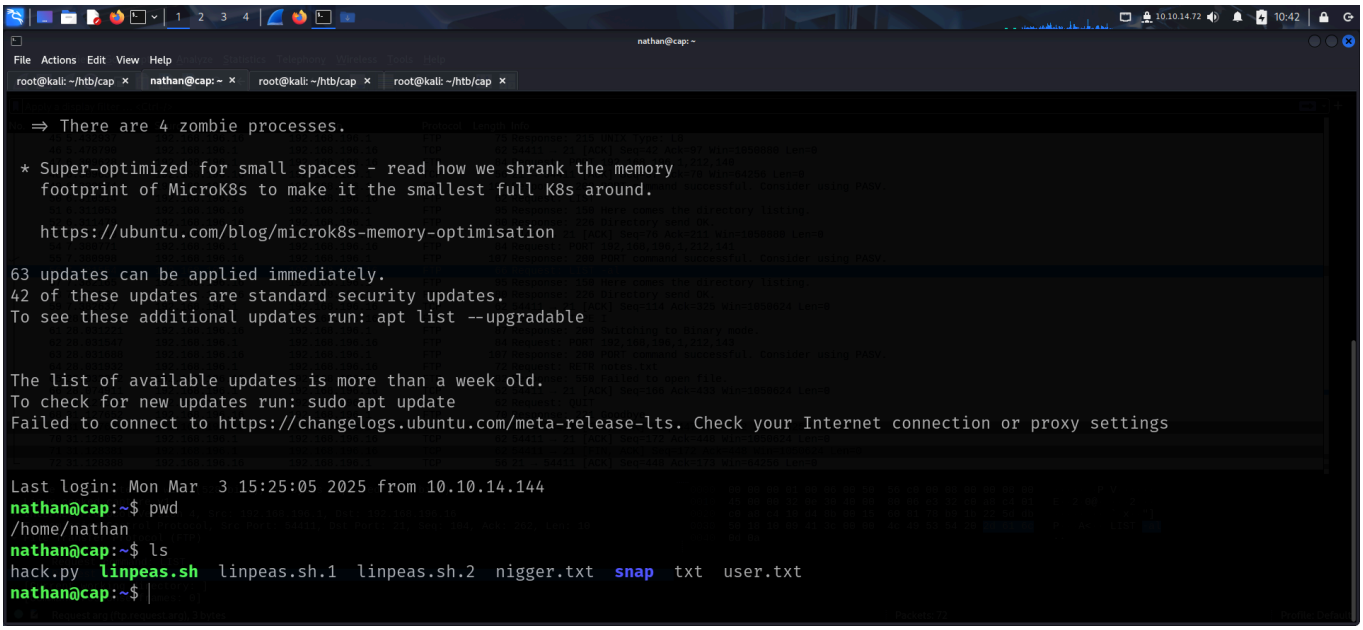


I checked if the existing credentials could also be used for **ssh** and found that the credentials had been reused.



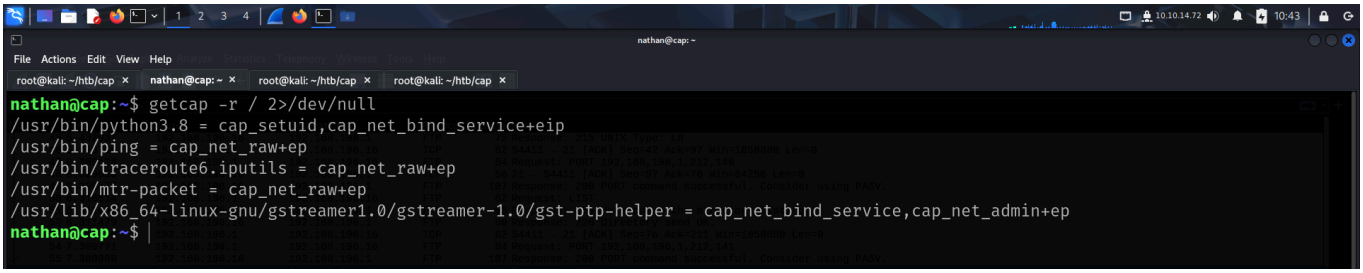
I log into the server using **ssh**.



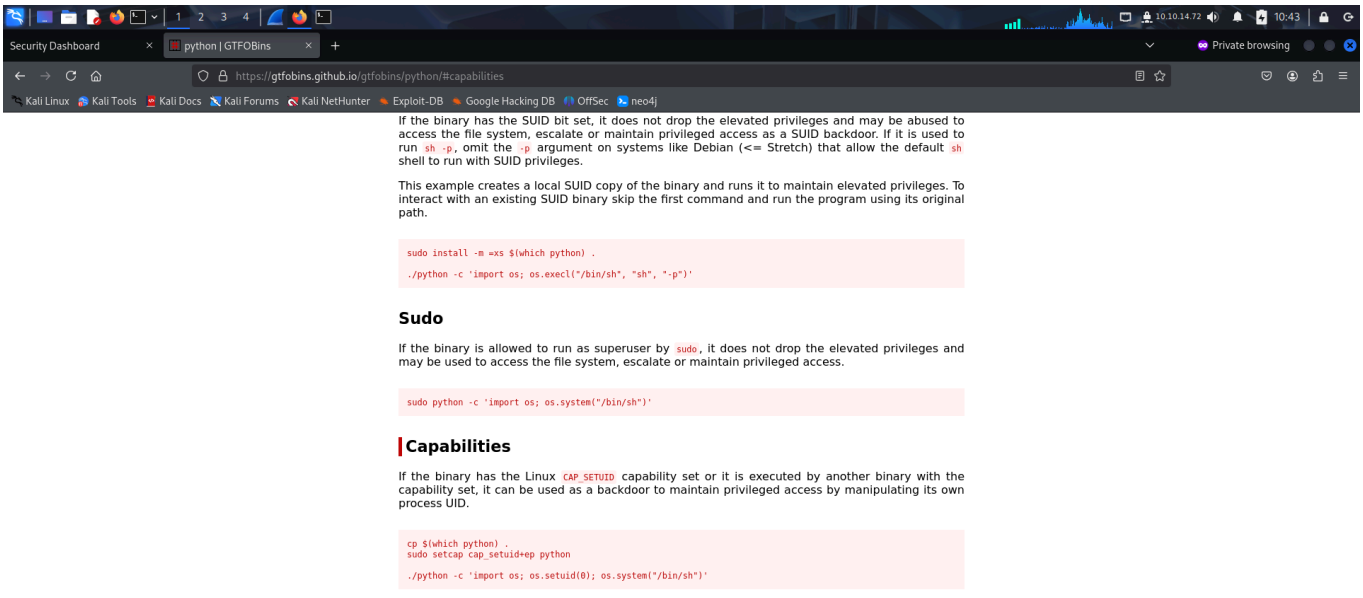


PRIVILEGE ESCALATION

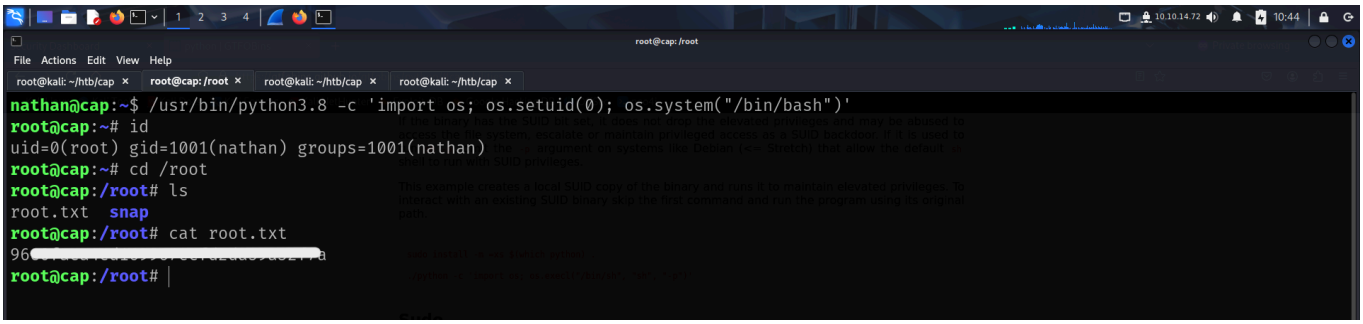
I viewed binaries with **capabilities** and found **python**.



I searched **GTFObins** and found a way to exploit this for privilege escalation.



I followed the commands and got **root** access. I then captured the root flag from */root*



That's it from my side! Until next time :)
